

# Web Application Development

(c) Gilles Everling

2019-11-22

# Table of Contents

1. Introduction	2
2. Operating systems	4
2.1. Ubuntu	4
2.1.1. Installation and configuration	4
.htaccess	5
Missing .Xauthority file	5
2.1.2. Administration	5
Command line	5
Useful commands	5
Security	8
Main directories	9
File system permissions	9
Using USB drives	9
Backup	9
Recover deleted files	10
Mail	10
Remote copy	10
Fail2ban	10
Logwatch	11
Grub2	11
cURL	11
Traceroute	12
Check disk health	12
Measure Internet speed	12
Remove PPA	12
2.2. Windows	12
2.2.1. Server	12
Introduction	12
Planning	13
Installation	14
Upgrade	19
Migration	19
Deployment	19
Security	19
Networking	20
Active Directory Domain Services (AD DS)	20
Group policy	24
Windows Management Instrumentation (WMI) filters	26

PowerShell .....	26
Windows Admin Center .....	26
AppLocker .....	27
Firewall .....	27
BitLocker .....	27
EFS .....	27
ISATAP .....	27
iSCSI .....	27
Resilient File System (ReFS) .....	27
IPsec .....	27
Windows containers .....	27
Remote Desktop Services .....	27
Tips & Tricks .....	27
2.2.2. Client .....	28
Administration .....	28
Tips & Tricks .....	29
3. Tools of the trade .....	31
3.1. Portable work environment .....	31
3.1.1. Vagrant .....	31
3.1.2. Docker .....	31
3.1.3. VirtualBox .....	31
3.2. Integrated Development Environments .....	33
3.2.1. Codiad .....	33
3.2.2. Atom .....	33
Portable installation .....	33
3.2.3. PhpStorm .....	33
Portable installation .....	33
Project setup .....	34
Database connection setup .....	39
Template adjustment .....	40
3.2.4. NetBeans .....	41
Portable installation .....	44
HTML5 Project setup .....	44
PHP Project setup .....	47
Database connection setup .....	52
Template adjustment .....	55
Useful NetBeans shortcuts .....	56
3.3. SSH .....	57
3.3.1. Clients .....	57
3.3.2. SFTP .....	57
3.4. Browsers .....	59

3.4.1. Firefox	59
3.5. Documentation	59
3.5.1. AsciiDoctor	59
Installation on Ubuntu	59
Tips & tricks	60
3.5.2. Markdown	60
3.6. Content Management Systems	60
3.6.1. Wordpress	61
3.6.2. Magento	61
4. Client side programming	62
4.1. HTML5	62
4.1.1. Basic structure	62
<head>	63
<body>	63
Opening and closing tags	63
Tabs, new lines and spaces	63
Comments	64
4.1.2. Validation	65
4.1.3. Planning	66
Brain storming	66
Blueprint	67
Requirements specification	67
4.1.4.  	68
4.1.5. <p>	69
4.1.6. Phrase tags	70
4.1.7. HTML entities	71
4.1.8. <header>	72
<h1> ... <h6>	72
4.1.9. Lists	73
<ul>	73
<ol>	74
Nested lists	75
<dl>	77
4.1.10. <a>	80
Hyperlink inside a hyperlink	82
Hyperlink ping tracking	83
4.1.11. <img>	83
Image formats	85
Image resizing	85
Adding transparency	87
<picture>	87

<map> .....	87
Logo creation .....	87
figure and figcaption .....	92
4.1.12. <nav> .....	92
4.1.13. <main> .....	93
4.1.14. <section> .....	94
4.1.15. <footer> .....	95
4.1.16. <article> .....	96
4.1.17. <aside> .....	97
4.1.18. <div> .....	97
4.1.19. <q> .....	97
4.1.20. <blockquote> .....	98
4.1.21. Firefox console and Firebug .....	98
4.1.22. <base> .....	99
4.1.23. <link> .....	99
Style sheet .....	99
Favicon .....	100
4.1.24. <meta> .....	100
4.1.25. <table> .....	103
4.1.26. Forms .....	112
<input> .....	120
<textarea> .....	121
<label> .....	121
<fieldset> .....	121
<legend> .....	121
<select> .....	122
<optgroup> .....	122
<option> .....	122
<button> .....	123
<datalist> .....	123
<keygen> .....	123
<output> .....	123
4.1.27. Block vs inline elements .....	123
4.1.28. <video> .....	126
Video file formats .....	126
Embedding YouTube videos .....	127
4.1.29. <audio> .....	128
Audio file formats .....	128
4.1.30. Additional elements .....	129
<pre> .....	130
<mark> .....	130

<address>	131
<time>	131
<ins>	131
<del>	131
<iframe>	131
4.1.31. <embed>	134
4.1.32. Global attributes	135
4.1.33. Event attributes	135
4.1.34. Copyright	135
4.1.35. Quiz	135
4.2. CSS3	135
4.2.1. Include CSS3	136
Inline	136
Embedded	136
External	137
Imported	138
Rules of precedence	139
4.2.2. Syntax	140
Comments	140
Variables	141
4.2.3. Units	141
4.2.4. Properties	142
object-fit	142
overflow	142
background	142
content	143
white-space	144
border-image	144
word-wrap	144
word-break	145
hyphens	145
box-sizing	145
contenteditable	145
resize	146
4.2.5. Selectors	146
4.2.6. Box model	147
4.2.7. Layout	149
Normal flow	149
Floats	152
Positioning	157
Resize images automatically while preserving aspect ratio	162

Table layout .....	162
Vertical centering .....	165
Multiple columns .....	167
Flexible box layout .....	167
Grid layout .....	188
Responsive design .....	190
Fluid typography .....	199
Navigation menus .....	199
4.2.8. Make it look good .....	205
Gradients .....	205
Buttons .....	207
Transformation and animation .....	208
Fonts .....	210
4.2.9. Quiz .....	211
4.2.10. Tests .....	211
Computer Shop .....	211
Video Viewer .....	217
4.3. JavaScript .....	222
4.3.1. Introduction to programming .....	222
4.3.2. Getting started with JavaScript .....	223
4.3.3. Adding JavaScript to HTML documents .....	225
External JavaScript .....	225
Embedded JavaScript .....	228
Inline JavaScript .....	229
Combinations .....	229
4.3.4. Comments .....	230
4.3.5. Semicolons .....	230
4.3.6. Basic input and output .....	231
console.log .....	231
document.write .....	231
alert .....	231
confirm .....	232
prompt .....	232
4.3.7. Constants .....	232
4.3.8. Variables .....	233
Variable names .....	233
4.3.9. Data types .....	234
Strings .....	234
Numbers .....	235
Booleans .....	237
Conversions .....	237

Dates .....	238
Math .....	238
Regular expressions .....	238
Maps and sets .....	238
4.3.10. Operators .....	238
eval .....	239
4.3.11. Conditional statements .....	239
if .....	239
switch .....	240
4.3.12. Loops .....	242
for .....	242
while .....	243
for in .....	244
do while .....	245
4.3.13. Jumps and exceptions .....	245
Labeled statements .....	245
break .....	245
continue .....	246
Exception handling .....	247
4.3.14. Functions .....	248
Normal functions .....	248
Arrow functions .....	250
Variable scope .....	252
let vs var .....	253
Anonymous functions .....	255
Self-invoking functions .....	256
Asynchronous programming .....	257
4.3.15. Debugging .....	260
4.3.16. Arrays .....	261
Creating and iterating .....	261
Adding and deleting .....	263
Multidimensional arrays .....	263
Methods .....	264
Turning arrays into parameters .....	264
4.3.17. Template literals .....	264
4.3.18. Objects and classes .....	265
Prototypes .....	265
Object creation .....	265
this .....	272
Getting and setting properties .....	272
Deleting properties .....	274

Testing properties . . . . .	274
Object attributes . . . . .	275
Property attributes . . . . .	276
Closures . . . . .	277
Private instance and prototype members . . . . .	277
<code>call</code> , <code>apply</code> and <code>bind</code> . . . . .	279
<code>Object</code> . . . . .	279
The <code>Proxy</code> object . . . . .	279
ECMAScript 6 . . . . .	279
4.3.19. Events . . . . .	280
Registering event listeners . . . . .	280
Event flow . . . . .	283
Keyboard events . . . . .	287
<code>unload</code> and <code>beforeunload</code> . . . . .	293
<code>error</code> . . . . .	294
4.3.20. ) . . . . .	295
Creating a new DOM document from a string . . . . .	296
Properties and methods of the HTML DOM Element Object . . . . .	296
Selecting DOM elements . . . . .	296
Traversing the DOM . . . . .	302
Getting and setting attributes . . . . .	305
Element content . . . . .	311
Managing nodes . . . . .	314
Determining the dimensions of elements . . . . .	316
Manipulating CSS . . . . .	316
Inserting JS dynamically . . . . .	319
Handling iframes . . . . .	320
Observing DOM mutations . . . . .	320
4.3.21. ) . . . . .	321
Timers . . . . .	321
The <code>location</code> object . . . . .	327
The <code>navigator</code> object . . . . .	327
The <code>history</code> object . . . . .	327
The <code>screen</code> object . . . . .	327
4.3.22. Strict mode . . . . .	328
4.3.23. Dates . . . . .	328
4.3.24. Modules . . . . .	328
4.3.25. Destructuring . . . . .	328
4.3.26. Viewports . . . . .	328
4.3.27. AJAX . . . . .	329
<code>XMLHttpRequest</code> . . . . .	329

Cross-origin requests .....	338
Fetch .....	338
4.3.28. JSON .....	341
4.3.29. Application Programming Interfaces (API) .....	347
File .....	348
Drag and drop .....	349
Web Workers .....	355
Server-Sent Events .....	355
Canvas .....	362
Web Sockets .....	370
WebGL .....	371
Page Visibility .....	395
WebAudio .....	396
Observers .....	397
Web Storage .....	398
IndexedDB .....	398
Cache .....	400
Service Worker .....	400
Progressive web applications (PWA) .....	405
JSON Web Tokens .....	406
Worklets .....	406
Web Animations .....	406
Web Messaging .....	406
Credential management .....	406
4.3.30. Tools .....	406
Dealing with old browsers: graceful degradation, polyfills and transpilers .....	406
Minimizers, optimizers, obfuscators, deobfuscation, compressors and beautifiers .....	407
Editor components and online editors .....	409
Feature detection .....	410
4.3.31. Frameworks .....	410
jQuery .....	410
4.3.32. Libraries .....	416
Web app .....	416
Visualization .....	416
Tables .....	416
GPU acceleration .....	417
Physics .....	417
Tone.js .....	417
4.3.33. JSDoc .....	417
4.3.34. Web Components .....	417
Custom elements .....	417

HTML import .....	417
HTML templates .....	417
4.3.35. Problems .....	417
Show/hide HTML element .....	417
Color preview .....	418
Puzzle .....	418
Path tracker .....	418
Paint app .....	418
4.3.36. Problem solutions .....	419
Show/hide HTML element .....	419
Color preview .....	420
4.3.37. Tests .....	421
Currency Converter .....	421
Space Ship .....	426
Space Circuit .....	429
Targeting Practice .....	431
Hockenheim Ring .....	434
Football Magic .....	436
Football Magic v2 .....	439
Calculator .....	441
Space Clock .....	445
Dog Race .....	449
Crazy Button .....	451
MicroJSON .....	452
Dice .....	456
Test Stats .....	458
Picture Viewer .....	460
4.4. XML .....	461
4.4.1. SVG .....	461
Syntax .....	462
Responsive CSS .....	465
4.4.2. RSS .....	466
4.4.3. Atom .....	466
4.4.4. OPML .....	467
4.4.5. MathML .....	467
4.5. Web Application Programming Interfaces .....	468
4.5.1. CKEditor .....	469
4.5.2. Google Charts .....	470
4.5.3. pdfmake .....	472
4.5.4. Facebook .....	472
4.5.5. Yahoo Query Language (YQL) .....	472

4.5.6. Finance .....	472
Quandl .....	472
Federal Reserve Bank of St. Louis .....	474
World Bank .....	477
Eurostat .....	477
Yahoo! Finance .....	477
IEX .....	477
4.5.7. Shodan .....	478
4.6. Security .....	478
4.6.1. Password hashing .....	478
4.6.2. Cross-Site Request Forgery (CSRF) .....	478
4.6.3. Fiddler .....	478
4.6.4. Google hacking .....	479
4.6.5. HTTPS .....	479
4.6.6. Email .....	479
4.6.7. Tracking .....	479
4.7. Mobile and desktop application development .....	479
4.7.1. Mobile .....	479
Cordova .....	480
NativeScript .....	480
MIT App Inventor 2 .....	480
Titanium .....	480
Ionic .....	480
Framework 7 .....	480
Sencha .....	480
Android SDK .....	480
4.7.2. Desktop .....	481
Electron .....	481
NW.js .....	481
4.8. Other .....	481
4.8.1. Accessibility .....	481
4.8.2. Search engine and social network optimization, eCommerce .....	481
Search engine optimization .....	481
Know your customer .....	482
4.8.3. User agent switching .....	482
5. Server side programming .....	483
5.1. Installing and configuring the tools .....	483
5.1.1. Introduction .....	483
5.1.2. Windows .....	484
Apache, PHP and MySQL/MariaDB .....	484
5.1.3. Ubuntu .....	493

Apache multi-processing modules (MPM) .....	493
Security .....	494
HTTP2 .....	494
Reverse proxy .....	494
Caching .....	495
VirtualBox .....	495
5.2. PHP .....	496
5.2.1. Introduction .....	496
5.2.2. Hello world .....	496
5.2.3. Logging .....	497
5.2.4. Variables and data types .....	498
String concatenation .....	499
5.2.5. Heredoc .....	499
5.2.6. Constants .....	500
5.2.7. Comments .....	501
5.2.8. Operators .....	501
@ operator .....	502
5.2.9. Conditional statements .....	502
if else .....	502
Ternary operator .....	503
5.2.10. Loops .....	503
for .....	503
while .....	503
break and continue .....	504
Nested loops .....	504
5.2.11. Arrays .....	504
Associative arrays .....	506
Looping through arrays with foreach .....	507
Multidimensional arrays .....	508
5.2.12. Functions .....	509
Type declarations .....	509
Variable scope .....	511
Default values for function parameters .....	513
Passing values by reference .....	513
Using a variable number of parameters .....	514
Anonymous functions .....	515
Recursive functions .....	515
5.2.13. Including external scripts .....	515
5.2.14. Superglobals .....	517
\$_GET .....	517
\$_POST .....	518

<code>\$_FILES</code> .....	518
<code>\$_COOKIE</code> .....	519
<code>\$_REQUEST</code> .....	519
<code>\$_SESSION</code> .....	519
<code>\$_SERVER</code> .....	519
<code>\$_ENV</code> .....	520
<code>\$GLOBALS</code> .....	520
5.2.15. Forms .....	520
POST .....	520
GET .....	521
Form validation .....	522
Using arrays to submit large volumes of data to the server .....	528
Using form fields to transfer data between pages .....	530
5.2.16. Useful PHP functions .....	532
<code>isset</code> .....	532
<code>strlen</code> .....	533
<code>exit</code> or <code>die</code> .....	533
<code>header</code> .....	533
<code>mt_rand</code> .....	533
<code>eval</code> .....	533
<code>dirname</code> .....	534
<code>number_format</code> .....	534
<code>strpos</code> .....	534
<code>implode</code> .....	534
<code>explode</code> .....	534
<code>filter_var</code> .....	534
<code>preg_match</code> .....	535
<code>sprintf</code> .....	535
<code>printf</code> .....	535
<code>list</code> .....	535
<code>ini_set</code> .....	535
<code>rawurlencode</code> .....	535
<code>htmlspecialchars</code> .....	535
<code>htmlentities</code> .....	535
<code>html_entity_decode</code> .....	535
<code>serialize</code> and <code>unserialize</code> .....	535
<code>getdate</code> .....	535
<code>date</code> .....	535
<code>nl2br</code> .....	535
Executing system commands .....	536
Password hashing .....	536

5.2.17. Regular expressions .....	536
5.2.18. Cookies and sessions .....	537
Cookies .....	537
Sessions .....	539
5.2.19. Files .....	541
Read file contents as a string .....	541
Read file contents as an array .....	542
Write to a file .....	542
Delete a file .....	542
Create a zip file .....	542
5.2.20. Classes and objects .....	542
Defining a class .....	543
Static properties and functions .....	543
Objects as parameters .....	544
5.2.21. Exceptions .....	544
5.2.22. Generators .....	544
5.2.23. PDF generation .....	544
TCPDF .....	544
FPDF .....	545
dompdf .....	545
5.2.24. WebSockets .....	545
5.2.25. Web scraping .....	545
using PHP .....	545
using JS .....	546
5.2.26. Frameworks .....	547
5.2.27. Tests .....	547
National Elections 2013 .....	547
WMOTU Speed Calc .....	552
WMOTU Shop .....	557
Quiz and Shopping Center .....	559
Roll The Dice .....	562
Message board .....	564
Book Shop .....	566
TicTacToe .....	569
TicTacToe Pro .....	572
WSERS1 Shop .....	575
WMOTU Madhouse .....	577
Operating Systems .....	577
Quiz .....	579
Login .....	580
Computer Shop .....	583

Simple Calculator .....	584
5.3. MySQL .....	587
5.3.1. When and where do we need a database? .....	587
5.3.2. Structured data .....	587
Personal data .....	587
5.3.3. Legal base for storing data .....	587
5.3.4. Introduction to MySQL .....	587
5.3.5. Data types .....	589
5.3.6. Case sensitivity .....	590
5.3.7. Comments .....	590
5.3.8. Naming conventions .....	591
5.3.9. Character sets and collations .....	591
5.3.10. DB Diagram design tools .....	592
5.3.11. Creating and dropping databases and tables .....	593
Via PhpStorm .....	595
Via the MySQL command line .....	597
Via phpMyAdmin .....	600
5.3.12. Work flow in PHP .....	602
5.3.13. Connecting to and disconnecting from a database via PHP .....	602
5.3.14. SQL injection .....	604
5.3.15. Retrieving data .....	608
With internal data .....	609
With user provided data .....	612
5.3.16. Inserting data .....	615
From a CSV file .....	615
Preparing strings for database insertion .....	615
With internal data .....	616
With user provided data .....	616
Storing objects, arrays and other complex data types .....	617
5.3.17. Deleting data .....	617
With internal data .....	618
With user provided data .....	618
5.3.18. Updating data .....	618
With internal data .....	619
With user provided data .....	619
5.3.19. Altering tables .....	620
5.3.20. DB normalization .....	620
5.3.21. Creating a safe registration and login .....	620
Rudimentary version .....	621
User table .....	621
DB class .....	621

Login and registration script .....	622
A more sophisticated approach .....	625
5.3.22. Procedural use of mysqli .....	625
5.3.23. PHP Data Objects (PDO) .....	634
Connection .....	634
Statement execution .....	635
5.3.24. Security .....	636
5.3.25. Tests .....	636
WMOTU Shop DB .....	636
Perfume Shop .....	640
MicroQuack .....	648
5.4. Node.js .....	652
5.4.1. Installation .....	652
5.4.2. NPM .....	653
5.4.3. Node Version Manager (NVM) .....	654
5.4.4. Introduction .....	654
Streams .....	654
Hello world web server .....	656
AJAX and JSON .....	659
Error handling .....	661
Requiring modules .....	662
__dirname .....	662
5.4.5. WebSockets .....	662
ws .....	662
uws .....	675
Socket.io .....	675
WebRTC .....	677
5.4.6. Express.js .....	677
Form handling .....	677
HTTPFormClient1.html .....	678
HTTPFormServer1.js .....	678
HTTPFormClient2.html .....	679
HTTPFormClient2.css .....	680
HTTPFormServer2.js .....	680
HTTPFormServer3.js .....	682
BlackList.js .....	683
HTTPFormServer4.js .....	684
HTTPServer2.js .....	686
5.4.7. Database handling .....	688
MySQL .....	688
CouchDB .....	692

PostgreSQL .....	692
MongoDB .....	693
Redis .....	694
5.4.8. Sessions .....	695
With HTTP(s) .....	695
With HTTP(s) and WebSocket .....	699
5.4.9. Registration and login .....	704
reglogin.html .....	704
reglogin_loggedin.html .....	706
DB2.js .....	706
reglogin.js .....	707
5.4.10. Security .....	709
Content Security Policy (CSP) and HTTP headers .....	709
5.4.11. Performance .....	710
Cache control .....	710
Diffeential serving .....	710
5.4.12. Other useful APIs and packages .....	710
Server Sent Events .....	710
Connect .....	710
Authentication .....	710
HTTP2 .....	710
PDF .....	710
Technical analysis .....	711
Email .....	711
Deployment .....	713
Streams .....	713
Web and image optimization .....	713
Others .....	713
5.4.13. apess.lu .....	713
mysql_options.js .....	713
DB.js .....	714
WebSocketServer.js .....	718
server.js .....	723
WebSocketClient.js .....	726
index.js .....	728
index.html .....	738
index.css .....	740
dossieren.html .....	745
dossieren.css .....	745
editiounen.html .....	746
editiounen.css .....	751

interna.js .....	754
interna.html .....	754
interna.css .....	758
legislatioun.html .....	760
legislatioun.css .....	769
login.html .....	772
login.css .....	772
member.html .....	772
member.css .....	773
presentatioun.html .....	773
presentatioun.css .....	775
profil.html .....	775
profil.css .....	777
ressourcen.html .....	779
ressourcen.css .....	782
6. Game development .....	783
6.1. JS game engines .....	783
6.1.1. Phaser .....	783
6.2. Game assets .....	783
6.3. Game promotion monetization .....	784
7. Machine learning .....	785
7.1. The right model to solve the problem .....	785
7.2. Applications .....	787
7.3. Tools .....	787
7.3.1. Python .....	787
7.3.2. R .....	788
7.4. Regression .....	788
7.4.1. Linear regression .....	788
Univariate linear regression .....	788
Multivariate linear regression .....	791
7.4.2. Polynomial regression .....	791
7.5. Neural networks .....	791
7.6. Reinforcement learning .....	791
7.6.1. Introduction .....	791
7.6.2. Q-learning .....	793
7.6.3. OpenAI Gym .....	796
Installation .....	796
7.6.4. ViZDoom .....	797
7.6.5. Arcade Learning Environment (ALE) .....	797
7.6.6. Tabular solution methods .....	797
7.7. Tensorflow .....	797

7.7.1. Tensors .....	798
7.7.2. Layers .....	800
8. Exercises .....	805
8.1. HTML5 .....	805
8.1.1. Exercise 1 .....	805
8.1.2. Exercise 2 .....	805
8.1.3. Exercise 3 .....	805
8.1.4. Exercise 4 .....	805
8.1.5. Exercise 5 .....	805
8.1.6. Exercise 6 .....	806
8.2. JavaScript .....	806
8.2.1. Exercise 1 .....	806
8.2.2. Exercise 2 .....	806
8.2.3. Exercise 3 .....	806
8.2.4. Exercise 4 .....	806
8.2.5. Exercise 5 .....	806
8.2.6. Exercise 6 .....	806
8.2.7. Exercise 7 .....	807
8.2.8. Exercise 8 .....	807
8.2.9. Exercise 9 .....	807
8.2.10. Exercise 10 .....	808
8.2.11. Exercise 11 .....	808
8.2.12. Exercise 12 .....	808
8.2.13. Exercise 13 .....	808
8.2.14. Exercise 14 .....	809
8.2.15. Exercise 15 .....	809
8.2.16. Exercise 16 .....	809
8.2.17. Exercise 17 .....	809
8.2.18. Exercise 18 .....	809
8.2.19. Exercise 19 .....	809
8.2.20. Exercise 20 .....	809
8.2.21. Exercise 21 .....	809
8.2.22. Exercise 22 .....	810
8.2.23. Exercise 23 .....	810
8.2.24. Exercise 24 .....	810
8.2.25. Exercise 25 .....	810
8.2.26. Exercise 26 .....	810
8.2.27. Exercise 27 .....	810
8.2.28. Exercise 28 .....	810
8.2.29. Exercise 29 .....	810
8.2.30. Exercise 30 .....	810

8.2.31. Exercise 31 .....	811
8.2.32. Exercise 32 .....	811
8.2.33. Exercise 33 .....	811
8.2.34. Exercise 34 .....	811
8.2.35. Exercise 35 .....	811
8.2.36. Exercise 36 .....	811
8.2.37. Exercise 37 .....	811
8.2.38. Exercise 38 .....	811
8.2.39. Exercise 39 .....	811
8.3. Node .....	812
8.3.1. Exercise 1 .....	812
8.3.2. Exercise 2 .....	812
8.3.3. Exercise 3 .....	812
8.3.4. Exercise 4 .....	812
8.3.5. Exercise 5 .....	812
8.3.6. Exercise 6 .....	812
8.3.7. Exercise 7 .....	812
8.3.8. Exercise 8 .....	813
8.4. PHP .....	813
8.4.1. Exercise 1 .....	813
8.4.2. Exercise 2 .....	813
8.4.3. Exercise 3 .....	813
8.4.4. Exercise 4 .....	813
8.4.5. Exercise 5 .....	814
8.4.6. Exercise 6 .....	814
8.4.7. Exercise 7 .....	814
8.4.8. Exercise 8 .....	814
8.4.9. Exercise 9 .....	815
8.4.10. Exercise 10 .....	815
8.4.11. Exercise 11 .....	815
8.4.12. Exercise 12 .....	815
8.4.13. Exercise 13 .....	816
8.4.14. Exercise 14 .....	816
8.4.15. Exercise 15 .....	816
8.4.16. Exercise 16 .....	816
8.4.17. Exercise 17 .....	816
8.4.18. Exercise 18 .....	817
8.4.19. Exercise 19 .....	817
8.4.20. Exercise 20 .....	817
8.4.21. Exercise 21 .....	817
8.4.22. Exercise 22 .....	817

8.4.23. Exercise 23 .....	818
8.4.24. Exercise 24 .....	819
8.4.25. Exercise 25 .....	820
8.4.26. Exercise 26 .....	820
8.4.27. Exercise 27 .....	820
8.4.28. Exercise 28 .....	821
8.4.29. Exercise 29 .....	821
8.5. MySQL .....	821
8.5.1. Exercise 1 .....	821
8.5.2. Exercise 2 .....	822
8.5.3. Exercise 3 .....	822
8.5.4. Exercise 4 .....	822
8.5.5. Exercise 5 .....	822
8.5.6. Exercise 6 .....	822
8.5.7. Exercise 7 .....	822
8.5.8. Exercise 8 .....	823
9. Problems .....	824
9.1. WMOTU Games .....	824
9.2. WMOTU League .....	824
9.2.1. WMOTU League Service .....	824
Solution .....	824
9.2.2. WMOTU League .....	824
Solution .....	824
9.3. WMOTU Sorter .....	826
9.3.1. Solution .....	827
9.4. WMOTU Sub .....	829
9.4.1. Solution .....	830
Standard solution .....	830
Evolved object oriented solution .....	835
9.5. WMOTU Mailer .....	843
9.5.1. WMOTU Mailer v1 .....	843
Solution .....	843
9.5.2. WMOTU Mailer v1++ .....	844
Solution .....	845
9.5.3. WMOTU Mailer v2 .....	846
Solution .....	846
9.5.4. WMOTU Mailer v3 .....	849
9.5.5. WMOTU Mailer v4 .....	849
9.5.6. WMOTU Mailer v5 .....	849
9.6. WMOTU Quack .....	849
9.6.1. Security .....	850

9.6.2. DB	850
9.6.3. db_credentials.php	850
9.6.4. database.php`	850
Constants	850
set_credentials	851
connect	851
get_user_id	851
get_user_name	851
get_user_data	851
get_users	851
get_description	851
update_description	851
is_logged_in	851
update_login_timestamp	852
register_login	852
login	852
activate_user	852
create_user	852
delete_user	852
9.6.5. index.php	852
9.6.6. main.php	853
9.6.7. logout.php	854
9.6.8. updatedescription.php	854
9.6.9. deleteprofile.php	854
9.6.10. deleteprofileimage.php	855
9.6.11. getprofileimage.php	855
9.6.12. getmember.php	856
9.6.13. Solution	856
DB	856
database.php	858
index.php	864
index.js	868
index.css	869
main.php	871
main.js	872
main.css	877
websocket.js	881
checkuser.php	883
logout.php	883
deleteprofile.php	883
deleteprofileimage.php	884

getprofileimage.php .....	884
getmember.php .....	885
getmembers.php .....	885
stillloggedin.php .....	886
updatedescription.php .....	886
upload.php .....	887
bouncer.php .....	888
server.php .....	889
10. Tutorials .....	891
10.1. WMOTU Lab .....	891
10.1.1. index.html .....	891
10.1.2. html5.html .....	893
10.1.3. style.css .....	895
10.2. WMOTU Invaders .....	898
10.2.1. index.html .....	898
10.2.2. index.js .....	899
10.2.3. index.php .....	908
10.3. WMOTU Invaders object-oriented .....	908
10.3.1. index.html .....	908
10.3.2. index.js .....	910
10.3.3. index.php .....	919
10.4. WMOTU Address Book .....	920
10.4.1. createDB.sql .....	920
Using MySQL command line .....	922
Security .....	922
Using phpMyAdmin .....	923
10.4.2. index.php .....	924
10.4.3. db_credentials.php .....	926
10.4.4. database.php .....	926
10.4.5. bouncer.php .....	930
10.4.6. main.php .....	931
10.4.7. logout.php .....	931
10.4.8. header.php .....	932
10.4.9. footer.php .....	932
10.4.10. add.php .....	932
10.4.11. edit.php .....	933
10.4.12. delete.php .....	935
10.4.13. style.css .....	935
10.5. WMOTU Tank .....	937
10.5.1. index.html .....	937
10.5.2. tank.js .....	938

10.6. Web Note .....	944
10.6.1. Requirements specification .....	944
10.6.2. Analysis .....	945
10.6.3. Design and implementation .....	945
DB .....	945
Backend .....	946
API .....	949
Frontend .....	951
11. Art Gallery .....	964
11.1. Bill Brinck .....	964
11.1.1. Publi .....	964
11.2. Marco Campana .....	964
11.2.1. Pacman .....	964
11.3. Nicolas Detombes .....	964
11.3.1. Dodge The Asteroids .....	964
11.3.2. Angry Chickens .....	964
11.3.3. Square Invasion .....	965
11.4. Aleksandar Veselinovic and Steve Weiwert .....	965
11.4.1. TrésBien .....	965
12. Resources .....	966
12.1. Books .....	966
12.2. Editors, IDEs and tools .....	966
12.3. HTML5 .....	966
12.3.1. Web sites .....	966
12.3.2. Books .....	967
12.4. CSS3 .....	967
12.4.1. Web sites .....	967
12.4.2. Books .....	967
12.5. JavaScript .....	968
12.5.1. Web sites .....	968
12.5.2. Books .....	969
12.6. PHP .....	970
12.6.1. Web sites .....	970
12.6.2. Books .....	970
12.7. MySQL .....	970
12.7.1. Web sites .....	970
12.7.2. Books .....	971
12.8. XML .....	971
12.9. Scripts .....	971
12.10. Programming .....	971
12.11. Web apps .....	971

12.12. Artificial intelligence .....	972
12.12.1. Reinforcement learning .....	973
12.12.2. Data sources .....	974
12.13. Internet .....	974
12.14. Security .....	974
12.15. OS .....	975
12.16. Hardware .....	975
12.17. Public domain pictures .....	976
12.18. Public domain music .....	976
12.19. Fonts and icons .....	976
12.20. Internet references .....	977
12.21. Online courses and tutorials .....	977
12.22. Data .....	977
12.23. Search engines .....	977
12.24. Office software .....	977
12.25. Thunderbird .....	977
12.26. Media editing .....	978
12.27. Music .....	978
12.28. Graphics .....	978
12.29. Selfhosted .....	978
12.30. Website builder .....	979
12.31. Tools .....	979

Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License

# Chapter 1. Introduction

From [Michael Nielsen](#) (emphasis mine):

Technologies come and technologies go, but insight is forever.

and

Struggling with a project you care about will teach you far more than working through any number of set problems. **Emotional commitment is a key to achieving mastery.**

Before we learn to develop web applications, we need a basic understanding of the Internet and the World Wide Web. For a more in-depth explanation of these two terms, take a look at [en.wikipedia.org/wiki/Internet](https://en.wikipedia.org/wiki/Internet) and [en.wikipedia.org/wiki/Hypertext\\_Transfer\\_Protocol](https://en.wikipedia.org/wiki/Hypertext_Transfer_Protocol). The Internet is a globally distributed network of networks. It connects billions of computers and devices allowing them to communicate with each other. The World Wide Web (WWW) is just one of many services running on the Internet. It is a huge collection of documents and other resources interlinked via [hyperlinks](#). Each resource has a uniform resource locator (URL), which gives access to this resource. Typically we use browsers (e.g. Mozilla Firefox, Google Chrome, Microsoft Edge, Apple Safari) to access the Internet. Browsers use the Hypertext Transfer Protocol (HTTP) to communicate with other computers, so called web servers, on the Internet. The Internet uses a whole suite of protocols that are split into several layers. At the top level, the application layer, we have HTTP and many other protocols. Below, on the transport layer, we have the Transport Control Protocol (TCP). Beneath this layer we have the Internet Protocol (IP) on the Internet layer.

The WWW has evolved significantly since the early nineties. Today the web browser and related technologies are increasingly becoming the platform of choice for application development, for a number of reasons:

1. Write once run anywhere. A web browser is installed by default on virtually every desktop, tablet, smartphone and other devices. A web application will run on all of these devices without requiring the user to download and install anything or the developer to provide executables for different operating systems.
2. Updates are instantaneous, i.e. the next time the user uses the application, he/she will automatically be using the latest version.
3. The performance of browser JavaScript engines rivals the best Java just in time compilers (JIT) and the gap to compiled C++ and assembler is dwindling. Today's web apps use multithreading, accelerated 3D graphics and many other techniques that make full use of the available hardware.
4. There are a large amount of standard application programming interfaces (API) as well as highly sophisticated open source libraries for all kinds of purposes.
5. A virtually unlimited amount of documentation is available.

The following is a small sample list of web applications to provide a glimpse of what can be done:

1. [codepen.io](https://codepen.io)
2. [mail.google.com](mailto:mail.google.com)
3. [www.draw.io](https://www.draw.io)
4. [www.cubeslam.com/qoxerr](https://www.cubeslam.com/qoxerr)
5. [github.com/kripken/BananaBread](https://github.com/kripken/BananaBread)

# Chapter 2. Operating systems

Some recommendations on how to choose between Linux and Windows as a server platform can be found at [www.singlehop.com/blog/linux-servers-vs-microsoft-windows-servers](http://www.singlehop.com/blog/linux-servers-vs-microsoft-windows-servers) and [www.1and1.com/digitalguide/server/know-how/linux-vs-windows-the-big-server-check](http://www.1and1.com/digitalguide/server/know-how/linux-vs-windows-the-big-server-check) among many others.

## 2.1. Ubuntu

What are Linux and Ubuntu? To find out see [en.wikipedia.org/wiki/Linux\\_distribution](http://en.wikipedia.org/wiki/Linux_distribution), [en.wikipedia.org/wiki/Ubuntu](http://en.wikipedia.org/wiki/Ubuntu) and [help.ubuntu.com/18.04/installation-guide/amd64/install.en.pdf](http://help.ubuntu.com/18.04/installation-guide/amd64/install.en.pdf).

### 2.1.1. Installation and configuration

The [server guide](#) provides the details. Get the ISO from [Ubuntu](#) and create a bootable USB stick using for instance [Rufus](#).

If you have your own domain, use [Certbot](#) to get a free certificate. Use `certbot --apache` to create, `certbot --apache --expand` to expand and `certbot renew` to renew all certificates. Alternatively, for test purposes, you can create a self-signed certificate as shown in [www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-apache-in-ubuntu-18-04](http://www.digitalocean.com/community/tutorials/how-to-create-a-self-signed-ssl-certificate-for-apache-in-ubuntu-18-04).

Follow [SSL v3 goes to the dogs - POODLE kills off protocol](#) and [forward secrecy](#):

```
SSLProtocol All -SSLv2 -SSLv3
SSLHonorCipherOrder on
SSLCipherSuite "EECDH+ECDSA+AESGCM EECDH+aRSA+AESGCM EECDH+ECDSA+SHA384
EECDH+ECDSA+SHA256 EECDH+aRSA SHA384 EECDH+aRSA+SHA256 EECDH+aRSA+RC4 EECDH EDH+aRSA
!RC4 !aNULL !eNULL !LOW !3DES !MD5 !EXP !PSK !SRP !DSS"
```

To enable [HTTP Strict Transport Security \(HTSTS\)](#):

```
a2enmod headers
```

Add the following line to `default-ssl.conf` in the `<VirtualHost default:443>` directive:

```
Header always set Strict-Transport-Security "max-age=63072000; includeSubdomains;
preload"
```

Make sure that the file `headers.load` is in the `mods-enabled` folder. If it isn't, copy it from `mods-available`. Add your site to [hstspreload.org](http://hstspreload.org) if you intend to use HTTPS over the long term.

Test your [server security](#).

To set umask permanently, add `umask 0027` to `/etc/profile` or change the following in `/etc/pam.d/common-session` (cf. [serverfault.com/questions/231717/how-to-get-full-control-of-umask-pam-](http://serverfault.com/questions/231717/how-to-get-full-control-of-umask-pam-)

permissions):

```
session optional pam_umask.so umask=0027
```

If you get the error `apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1` when restarting Apache, edit your `/etc/hosts` file and make sure it contains `127.0.0.1 localhost servername.domain.com servername` (cf. [source](#)).

### `.htaccess`

In order to be able to use `.htaccess` files:

1. The Apache `rewrite` module needs to be enabled, if it isn't already:

```
a2enmod rewrite
apache2ctl restart
```

2. The `AllowOverride All` directive needs to be in your Apache config file (usually in `/etc/apache2`) for the directory tree where your access file is located.

[www.thefanclub.co.za/how-to/how-secure-ubuntu-1604-lts-server-part-1-basics](http://www.thefanclub.co.za/how-to/how-secure-ubuntu-1604-lts-server-part-1-basics)

For the rest see [Security](#).

### Missing `.Xauthority` file

If you log in via SSH and get this message use

```
ssh -X user@host
```

to have the file created ([source](#)).

## 2.1.2. Administration

To simplify package management you might want to install [wajig](#).

### Command line

To learn the Linux command line study [linuxcommand.org](http://linuxcommand.org).

### Useful commands

#### Recursively search for strings in files

```
grep -rHn "string" /path
```

[askubuntu.com/questions/55325/how-to-use-grep-command-to-find-text-including-subdirectories](http://askubuntu.com/questions/55325/how-to-use-grep-command-to-find-text-including-subdirectories)

### Kernel version

```
uname -r
```

### Last logged in users

```
last
```

### Determine Ubuntu version

```
lsb_release -a
```

or

```
cat /etc/issue
```

[help.ubuntu.com/community/CheckingYourUbuntuVersion](https://help.ubuntu.com/community/CheckingYourUbuntuVersion)

### Locate a file

```
whereis <filename>
```

**find** without permission denied messages

```
find / -name <name> -print 2>&-
```

which is equivalent to (cf. [source](#)):

```
find / -name <name> -print 2>/dev/null
```

### chmod all directories but not files

```
find . -type d -exec chmod o+rx {} +
```

[superuser.com/questions/91935/how-to-chmod-all-directories-except-files-recursively](https://superuser.com/questions/91935/how-to-chmod-all-directories-except-files-recursively)

### chmod all executable files

```
find . -executable -type f -exec chmod o+rx {} +
```

[superuser.com/questions/38981/how-can-i-find-only-the-executable-files-under-a-certain-directory-in-linux](https://superuser.com/questions/38981/how-can-i-find-only-the-executable-files-under-a-certain-directory-in-linux)

## Activate root

`sudo passwd root` and give root a password. Afterwards, you can for instance run `su -`.

## Remotely unlocking encrypted Ubuntu servers with Dropbear

## Setting `umask` permanently

[Set system wide umask](#)

## Reconfigure package

`dpkg-reconfigure package`

## Uninstall package completely

`apt purge package`

## Solve dependency problems with `aptitude`

[stackoverflow.com/questions/26571326/how-do-i-resolve-the-following-packages-have-unmet-dependencies/35679746#35679746](https://stackoverflow.com/questions/26571326/how-do-i-resolve-the-following-packages-have-unmet-dependencies/35679746#35679746)

## Nano copy/paste

[askubuntu.com/questions/833102/copy-only-copy-not-cutting-in-nano](http://askubuntu.com/questions/833102/copy-only-copy-not-cutting-in-nano)

## Nano jump to beginning or end of file

[freethegnu.wordpress.com/2007/06/23/nano-shortcuts-syntax-highlight-and-nanorc-config-file-pt1](http://freethegnu.wordpress.com/2007/06/23/nano-shortcuts-syntax-highlight-and-nanorc-config-file-pt1)

To jump to the beginning use `Ctrl+W` followed by `Ctrl+Y`. To jump to the end use `Ctrl+W` followed by `Ctrl+V`.

## Monitor socket connections

`ss`

## Monitor network

`ip`

[www.ghacks.net/2018/04/15/back-to-basics-part-6-using-the-ip-command-in-gnu-linux-for-beginners](http://www.ghacks.net/2018/04/15/back-to-basics-part-6-using-the-ip-command-in-gnu-linux-for-beginners)

## View contents of compressed files

[www.linuxtechi.com/view-contents-compressed-file-folders-linux](http://www.linuxtechi.com/view-contents-compressed-file-folders-linux)

## Exclude files/folders from archives

```
# https://stackoverflow.com/questions/984204/shell-command-to-tar-directory-excluding-
certain-files-folders
# Note the files/folders excluded are relatively to the root of your tar.
tar cfz /media/Backup232/www`date +%a`.tar.gz --exclude='owncloud'
--exclude='everling.lu/WAD' /var/www 2>/dev/null
```

### Redirect output streams

From [askubuntu.com/questions/625224/how-to-redirect-stderr-to-a-file](https://askubuntu.com/questions/625224/how-to-redirect-stderr-to-a-file):

1. Redirect stdout to one file and stderr to another file: `command > out 2>error`
2. Redirect stderr to stdout (&1), and then redirect stdout to a file: `command >out 2>&1`
3. Redirect both to a file: `command &> out`

### Use wget to recursively download all files of a type

[www.garron.me/en/bits/wget-download-image-pdf-type-of-files.html](https://www.garron.me/en/bits/wget-download-image-pdf-type-of-files.html)

### User login history

`lastlog` showing data from `/var/log/lastlog`.

To see detailed info view `/var/log/auth.log`.

### Change hostname

[linuxize.com/post/how-to-change-hostname-on-ubuntu-18-04](https://linuxize.com/post/how-to-change-hostname-on-ubuntu-18-04)

### Display your public IP address

`curl` [icanhazip.com](https://icanhazip.com)

### Get or set timezone

[linuxize.com/post/how-to-set-or-change-timezone-on-ubuntu-18-04](https://linuxize.com/post/how-to-set-or-change-timezone-on-ubuntu-18-04)

### Security

[www.cyberciti.biz/tips/linux-security.html](https://www.cyberciti.biz/tips/linux-security.html)

### SSL/TLS

Proceed as follows to create a self signed certificate to be able to use HTTPS (cf. [websiteforstudents.com/create-ssl-tls-self-signed-certificates-on-ubuntu-16-04-18-04-18-10](https://websiteforstudents.com/create-ssl-tls-self-signed-certificates-on-ubuntu-16-04-18-04-18-10/)):

```
openssl genrsa -aes256 -out server.key 4096
openssl rsa -in server.key -out server.key
openssl req -new -days 2000 -key server.key -out server.csr
openssl x509 -req -sha512 -days 2000 -in server.csr -signkey server.key -out
server.crt
```

## Firewall

[help.ubuntu.com/community/UFW](http://help.ubuntu.com/community/UFW)

## Antivirus

[www.clamav.net](http://www.clamav.net)

## Main directories

[Main directories](#)

## File system permissions

[File system permissions](#)

[Wayne Pollock](#)

[help.ubuntu.com/community/FilePermissionsACLs](http://help.ubuntu.com/community/FilePermissionsACLs)

## Using USB drives

Find out what the drive is called using `fdisk -l`, then mount the drive using `mount <drive> /media/usb`. To unmount use `umount /media/usb`.

To have a drive mounted automatically, add it to `/etc/fstab`. Use `lsblk -o` or `fdisk -l` to get the required information for your drive. After a system reboot, your drive should be available.

## Backup

[System backup](#) is essential. Install [storeBackup](#), create a directory for your backups and add a [crontab](#) task using `crontab -e`. Here is an example crontab entry where an email is sent after backup completion (cf. [how-to-sendmail](#)):

```
* 3 * * 1 /opt/storeBackup/bin/storeBackup.pl --sourceDir /var/www --backupDir
/root/backup
| sed 's/^/To: mail address\nSubject: backup\n\n/' | sendmail -t
```

Alternatively you can set up a [systemd timer](#). A discussion about pros and cons can be found at [cron vs systemd timers](#).

Instead of or in addition to local backup you might consider cloud backup using [Duplicity](#), preferably with [encryption](#).

## Recover deleted files

Install and use `extundelete`.

## Mail

Whilst you may not want to run your own [mail server](#), if you want to enable your server to send emails, install [Postfix](#) or better a complete [mail server](#). For Postfix configuration read [steam.io/2013/04/01/postfix-rate-limiting](#) and [easyengine.io/tutorials/mail/postfix-debugging](#). To send an email, create a file with content structured as in the following example and then use `sendmail recipient <file:`

```
Subject: everling.lu backup job
```

```
Backup has been run
```

For alternative approaches see [serverfault.com/questions/370935/how-to-sendmail-via-command-line-piping](#).

You can manage emails using [mail](#) or more comfortably using [mutt](#).

## Remote copy

```
rcp -prv source target
```

## Fail2ban

From [pbxinaflash.com/community/threads/disable-fail2ban.6972](#):

Alternative to stopping fail2ban:

From the Linux command prompt type: `service fail2ban stop`

To start fail2ban: `service fail2ban start`

To reload fail2ban if you have a banned IP: `service fail2ban restart`

Restarting will `clear` the ban.

To prevent fail2ban from banning IPs on the local network or other places:

Modify `/etc/fail2ban/jail.conf` look for the line:

```
#ignoreip 127.0.0.1 192.168.1.24/24 ....
```

uncomment it by removing the # and then change the IP addresses. To have fail2ban ignore network 192.168.20.0 (255.255.255.0), add `192.168.20.0/24` to the above line. You can add as many networks as you like. Just leave a space.

Just a note, if you have a VPN or a tunnel, you should add its network too. I've had the tunnel banned!

You can see if fail2ban has banned an IP by checking `/var/log/fail2ban.log`. It will indicate banned and unbanned IP addresses.

To unban an IP address, use `fail2ban-client set YOURJAILNAMEHERE unbanip IPADDRESSHERE`

The hard part is finding the right jail: Use `iptables -L -n` to find the rule name then use `fail2ban-client status` to get the actual jail names. The rule name and jail name may not be the same but it should be clear which one is related to which ([source](#)).

Stop Apache DOS attacks as described in [r3dux.org/2013/06/how-to-stop-apache-dos-attacks-with-fail2ban](http://r3dux.org/2013/06/how-to-stop-apache-dos-attacks-with-fail2ban).

## Logwatch

[www.digitalocean.com/community/tutorials/how-to-install-and-use-logwatch-log-analyzer-and-reporter-on-a-vps](http://www.digitalocean.com/community/tutorials/how-to-install-and-use-logwatch-log-analyzer-and-reporter-on-a-vps)

## Grub2

[Grub2](#) is the default boot loader and manager for Ubuntu.

## cURL

```
git clone https://github.com/curl/curl.git
apt install autoconf libtool
./buildconf
./configure
```

## Traceroute

[dnsquery.org/dnstraversal](https://dnsquery.org/dnstraversal)

## Check disk health

```
apt install smartmontools
fdisk -l
smartctl -c /dev/sdX
smartctl -t short /dev/sdX
smartctl -H /dev/sdX
```

See [blog.shadypixel.com/monitoring-hard-drive-health-on-linux-with-smartmontools](https://blog.shadypixel.com/monitoring-hard-drive-health-on-linux-with-smartmontools) for further details.

## Measure Internet speed

```
wget -O - raw.githubusercontent.com/sivel/speedtest-cli/master/speedtest.py | python
```

See [www.shellhacks.com/test-internet-speed-linux-command-line](https://www.shellhacks.com/test-internet-speed-linux-command-line).

## Remove PPA

[itsfoss.com/how-to-remove-or-delete-ppas-quick-tip](https://itsfoss.com/how-to-remove-or-delete-ppas-quick-tip)

# 2.2. Windows

## 2.2.1. Server

### Introduction

From [docs.microsoft.com/en-us/windows-server](https://docs.microsoft.com/en-us/windows-server):

Windows Server is the platform for building an infrastructure of connected applications, networks, and web services, from the workgroup to the data center.

Windows Server is used to serve all kinds of applications, data and services a client might require. Microsoft calls these roles and features. Roles are major capabilities required. For instance **Active Directory** to authenticate users or the **Domain Name System** (DNS) to locate other computers. Features can add additional functionality to the OS or enhance specific roles. [Here](#) and [here](#) you can find a list of the roles and features provided by Windows Server. There you can also see the differences

between **Windows Server Standard** and **Windows Server Datacenter**. One of the major differences is that with the standard version you can only host 2 virtual machines per license.

The two Windows Server versions are available with different footprints either with desktop (**Desktop Experience**) or without. The advantage of the former being ease of use and of the latter performance and storage requirements.



This [article](#) provides much more in-depth information on the different Windows Server versions and is a must read.

Even without desktop experience it is still possible to run a number of GUI applications as illustrated in [medium.com/@RealNetwork/windows-server-core-2019-gui-management-sysinternals-utilities-datacenter-standard-hyper-v-dashboard-265801412c89](https://medium.com/@RealNetwork/windows-server-core-2019-gui-management-sysinternals-utilities-datacenter-standard-hyper-v-dashboard-265801412c89).

[www.youtube.com/playlist?list=PLcRhKiWZmM8L6r2vysrNaIz8inmdNQJk](https://www.youtube.com/playlist?list=PLcRhKiWZmM8L6r2vysrNaIz8inmdNQJk)

[www.microsoft.com/en-us/cloud-platform/windows-server](https://www.microsoft.com/en-us/cloud-platform/windows-server)

[docs.microsoft.com/en-us/windows-server](https://docs.microsoft.com/en-us/windows-server)

[www.microsoft.com/en-us/learning/companion-moc.aspx](https://www.microsoft.com/en-us/learning/companion-moc.aspx)

[www.classcentral.com/course/coursera-windows-server-management-and-security-9320](https://www.classcentral.com/course/coursera-windows-server-management-and-security-9320)

[social.technet.microsoft.com/wiki/contents/articles/11608.e-book-gallery-for-microsoft-technologies-en.aspx](https://social.technet.microsoft.com/wiki/contents/articles/11608.e-book-gallery-for-microsoft-technologies-en.aspx)

[blogs.msdn.microsoft.com/microsoft\\_press/2016/09/26/free-ebook-introducing-windows-server-2016](https://blogs.msdn.microsoft.com/microsoft_press/2016/09/26/free-ebook-introducing-windows-server-2016)

[www.discudemy.com/category/Windows%20Server](https://www.discudemy.com/category/Windows%20Server)

[www.youtube.com/channel/UCP2uPp7TUXwJCicB0byzDnQ](https://www.youtube.com/channel/UCP2uPp7TUXwJCicB0byzDnQ)

[winintro.ru](https://winintro.ru)

## Planning

Starting with Windows Server 2016, we cannot convert between Core and Desktop anymore. Quote from [docs.microsoft.com/en-us/windows-server/get-started/getting-started-with-server-with-desktop-experience](https://docs.microsoft.com/en-us/windows-server/get-started/getting-started-with-server-with-desktop-experience):

Unlike some previous releases of Windows Server, you cannot convert between Server Core and Server with Desktop Experience after installation. If you install Server with Desktop Experience and later decide to use Server Core, you should do a fresh installation.

## Migration

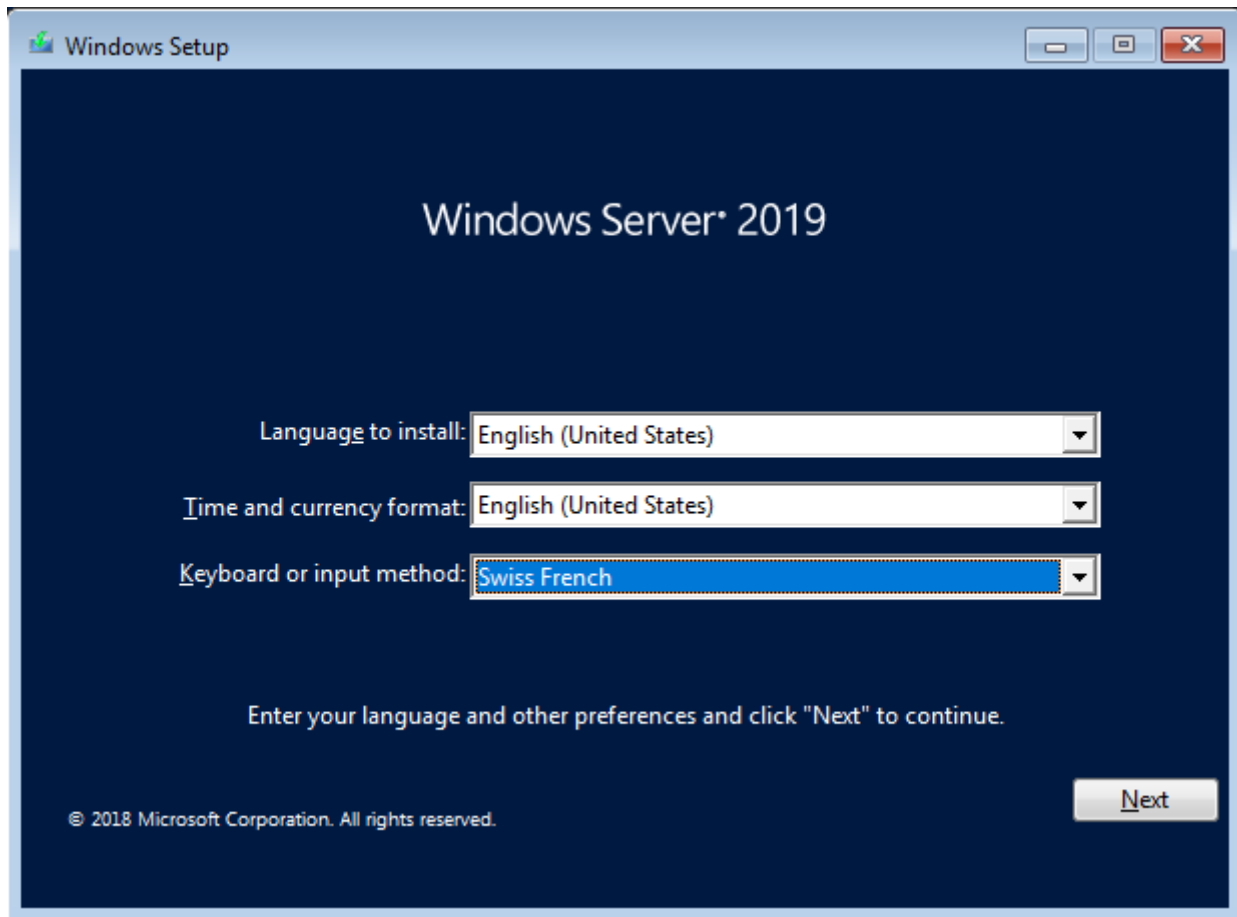
If you are planning to migrate Windows Server be sure to study [docs.microsoft.com/en-us/windows-server/get-started/migrate-roles-and-features](https://docs.microsoft.com/en-us/windows-server/get-started/migrate-roles-and-features).

## Installation

To be able to have several virtual machines communicate set the network mode in VirtualBox to **NAT Network** as described in [www.techrepublic.com/article/how-to-create-multiple-nat-networks-in-virtualbox](http://www.techrepublic.com/article/how-to-create-multiple-nat-networks-in-virtualbox). Also see [www.virtualbox.org/manual/ch06.html](http://www.virtualbox.org/manual/ch06.html) and [www.nakivo.com/blog/virtualbox-network-setting-guide](http://www.nakivo.com/blog/virtualbox-network-setting-guide).



If you discover that your guest OS takes up a full core all the time, try assigning 2 cores to the guest VM via **System** → **Processor** in the VirtualBox Manager (cf. [forums.virtualbox.org/viewtopic.php?f=6&t=87991&start=45](https://forums.virtualbox.org/viewtopic.php?f=6&t=87991&start=45)).



Select the operating system you want to install

Operating system	Architecture	Date modified
Windows Server 2019 Standard Evaluation	x64	1/8/2019
Windows Server 2019 Standard Evaluation (Desktop Experien...	x64	1/8/2019
Windows Server 2019 Datacenter Evaluation	x64	1/8/2019
Windows Server 2019 Datacenter Evaluation (Desktop Experi...	x64	1/8/2019

**Description:**

This option installs the full Windows graphical environment, consuming extra drive space. It can be useful if you want to use the Windows desktop or have an app that requires it.

Which type of installation do you want?

**Upgrade: Install Windows and keep files, settings, and applications**

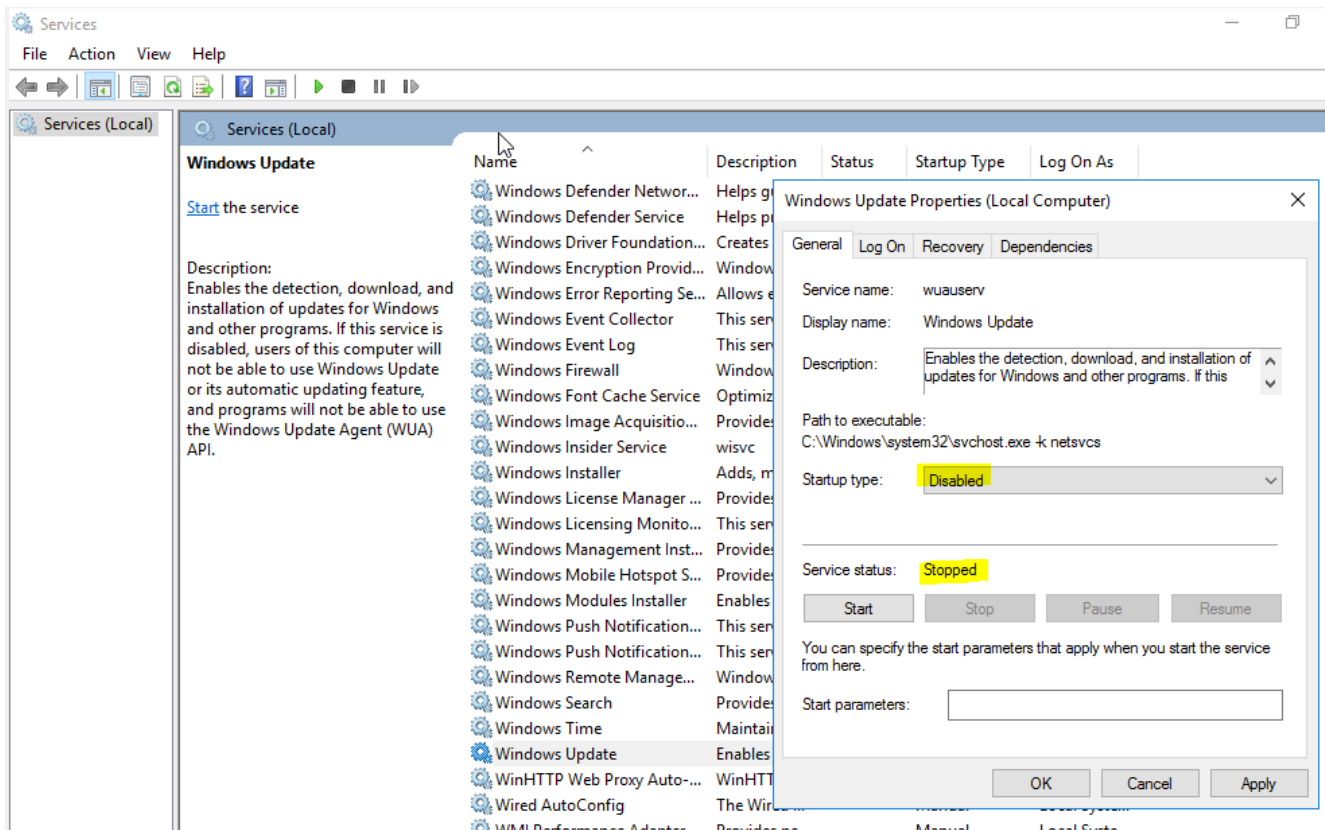
The files, settings, and applications are moved to Windows with this option. This option is only available when a supported version of Windows is already running on the computer.

**Custom: Install Windows only (advanced)**

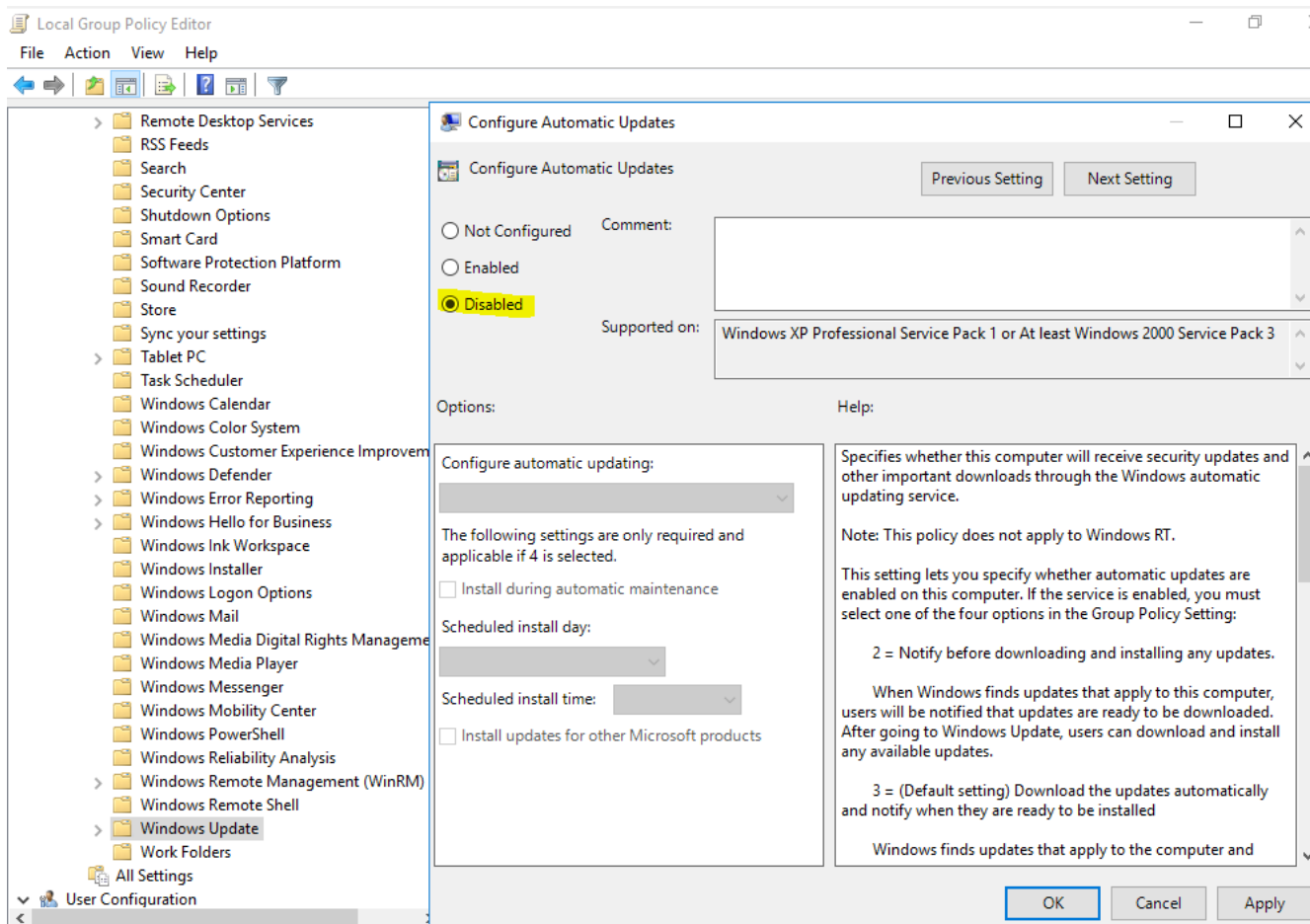
The files, settings, and applications aren't moved to Windows with this option. If you want to make changes to partitions and drives, start the computer using the installation disc. We recommend backing up your files before you continue.

[Help me decide](#)

In previous Windows Server versions updates took ages so it was sometimes useful to disable them. This is not the case for Windows Server 2019 anymore, as the update speed has increased significantly. If you still want to disable updates, for the GUI version launch Services from the search box:



Then run `gpedit.msc` → Computer Configuration → Administrative Templates → Windows Components → Windows Update and disable Configure Automatic Updates.



For the version without GUI, run `sconfig`, select menu 5 and then set updates to manual.

Set the correct timezone:

## Date and time

10:26 AM, Tuesday, January 22, 2019

Set time automatically

☒ On

Set time zone automatically

☐ Off

Change date and time

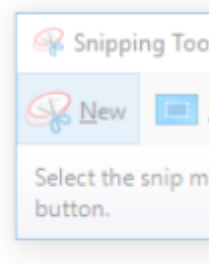
Change

Time zone

(UTC+01:00) Brussels, Copenhagen, Madrid, Paris

Adjust for daylight saving time automatically

☒ On



Now remove the disc from the virtual drive and select Devices → Insert Guest additions CD image... and run the guest additions. From the command line you'll have to change to the CD drive and then run `VBoxWindowsAdditions.exe`.

Install [BgInfo](#).

To see detailed info about the server run `Get-CimInstance Win32_OperatingSystem|fl *`.

To enable ping `netsh advfirewall firewall add rule name="ICMP Allow incoming V4 echo request" protocol=icmpv4:8,any dir=in action=allow`.

To enable RPC communication between server and client start your Windows client computer, run `gpedit.msc` and enable Computer Configuration → Administrative Templates → Network → Network Connections → Windows Firewall → Domain Profile → Windows Firewall: Allow inbound remote administration exception.

Install a real browser on both servers and disable Internet Explorer using `dism /online /Disable-Feature /FeatureName:Internet-Explorer-Optional-amd64` in an elevated command prompt or PowerShell (see [support.microsoft.com/en-us/help/4013567/how-to-disable-internet-explorer-on-windows](https://support.microsoft.com/en-us/help/4013567/how-to-disable-internet-explorer-on-windows)).

Verify that remote management is enabled. This can be done via `sconfig` or `Configure-SMRemoting-get` in PowerShell or via the Server Manager GUI.

If you want to manage Windows Server installations remotely from a Windows client you can use the [Remote Server Administrative Tools \(RSAT\)](#). To connect to a remote server you might need to follow [nokitel.im/index.php/2016/03/11/windows-10-and-server-2012-r2-server-manager-winrm-negotiate-authentication-error](http://nokitel.im/index.php/2016/03/11/windows-10-and-server-2012-r2-server-manager-winrm-negotiate-authentication-error) and study [github.com/MicrosoftDocs/windowsserverdocs/blob/master/WindowsServerDocs/administration/server-manager/add-servers-to-server-manager.md](https://github.com/MicrosoftDocs/windowsserverdocs/blob/master/WindowsServerDocs/administration/server-manager/add-servers-to-server-manager.md).

To execute cmdlets on remote servers, you need to specify the credentials as described in [stackoverflow.com/questions/34768795/pass-password-into-credential](https://stackoverflow.com/questions/34768795/pass-password-into-credential).

To disable the monitor timeout on a server without GUI run `powercfg -change -monitor-timeout-ac 0` (cf. [ss64.com/nt/powercfg.html](http://ss64.com/nt/powercfg.html)).

To see a list of installed roles and features:

```
Get-WindowsFeature|Where Installed or Get-WindowsFeature|Where-Object InstallState -eq Installed
```

You can be more specific, for example select only roles and features that begin with "Print" on computer server1 like this:

```
Get-WindowsFeature -Name Print* -ComputerName server1 -Credential (Get-Credential CORP\Administrator)|Where Installed
```

Install roles and features using `Install-WindowsFeature`, for example:

```
Install-WindowsFeature -Name DHCP -ComputerName server1
```

To remove use `Uninstall-WindowsFeature`.

To get a list of all PowerShell commands use `Get-Command`.

To get help on a command use `help`.

To run background jobs use `Start-Job`:

```
Start-Job {Get-WindowsFeature}
```

`Get-Job` allows us to see the state of a job.

To run a background job and see its output use `Receive-Job`:

```
$job=Start-Job {Get-WindowsFeature}
```

```
Receive-Job -Job $job
```

A new job can be scheduled:

```
$trigger=New-JobTrigger -Once -At 8:05AM Register-ScheduledJob -Name ListFeatures {Get-WindowsFeature} -Trigger $trigger $job=Start-Job {Get-WindowsFeature}
```

We can create persistent sessions:

```
$script1={Start-Job {Install-WindowsFeature -Name Windows-Server-Backup}}
```

```
$script2={Start-Job {Uninstall-WindowsFeature -Name Windows-Server-Backup}}
```

```
$srv1=New-PSSession -ComputerName server1  
  
Invoke-Command $srv1 $script1  
  
Disconnect-PSSession -Name $srv1  
  
Get-WindowsFeature -Name Windows-Server-Backup -ComputerName server1  
  
Connect-PSSession -Name $srv1  
  
Invoke-Command $srv1 $script2  
  
Get-PSSession|Remove-PSSession
```

To see which user you are logged in with use `set` or `whoami`.

To extend the Windows Server evaluation period, use `slmgr -dlv` to see the current status and `slmgr -rearm` to rearm.

To disable IPv6 using PowerShell see [giritharan.com/disable-ipv6](http://giritharan.com/disable-ipv6).

## Upgrade

[docs.microsoft.com/en-us/windows-server/upgrade/upgrade-overview](https://docs.microsoft.com/en-us/windows-server/upgrade/upgrade-overview)

[www.microsoft.com/upgradecenter](https://www.microsoft.com/upgradecenter)

## Migration

[download.microsoft.com/download/0/F/B/0FBFAA46-2BFD-478F-8E56-7BF3C672DF9D/Migrate\\_Roles\\_and\\_Features\\_to\\_Windows\\_Server\\_2012\\_R2\\_or\\_Windows\\_Server\\_2012.pdf](https://download.microsoft.com/download/0/F/B/0FBFAA46-2BFD-478F-8E56-7BF3C672DF9D/Migrate_Roles_and_Features_to_Windows_Server_2012_R2_or_Windows_Server_2012.pdf)

## Microsoft Assessment and Planning Toolkit

[www.microsoft.com/en-us/download/details.aspx?id=8781](https://www.microsoft.com/en-us/download/details.aspx?id=8781)

## Deployment

The easiest way to deploy desktops and servers is [Microsoft Deployment Toolkit \(MDT\)](#).

Make sure to install both the ADK and the Windows PE add-on for the ADK (cf. [osddeployment.dk/2018/12/30/unable-to-open-the-specified-wim-file-error-in-mdt-after-upgrading-to-adk-1809](https://osddeployment.dk/2018/12/30/unable-to-open-the-specified-wim-file-error-in-mdt-after-upgrading-to-adk-1809)).

[en.wikipedia.org/wiki/Preboot\\_Execution\\_Environment](https://en.wikipedia.org/wiki/Preboot_Execution_Environment)

SCCM [docs.microsoft.com/en-us/sccm/core/understand/introduction](https://docs.microsoft.com/en-us/sccm/core/understand/introduction) [www.microsoft.com/en-us/evalcenter/evaluate-system-center-configuration-manager-and-endpoint-protection](https://www.microsoft.com/en-us/evalcenter/evaluate-system-center-configuration-manager-and-endpoint-protection)

## Security

### Security principals

[docs.microsoft.com/en-us/windows/security/identity-protection/access-control/security-principals](https://docs.microsoft.com/en-us/windows/security/identity-protection/access-control/security-principals)

## Networking

[www.slideshare.net/FabioAlmeida8/98-366-mva-slides-lesson-1](http://www.slideshare.net/FabioAlmeida8/98-366-mva-slides-lesson-1)

[en.wikipedia.org/wiki/Private\\_network](http://en.wikipedia.org/wiki/Private_network)

[www.lifewire.com/what-is-a-private-ip-address-2625970](http://www.lifewire.com/what-is-a-private-ip-address-2625970)

[www.microsoft.com/en-us/download/details.aspx?id=8781](http://www.microsoft.com/en-us/download/details.aspx?id=8781)

[simplifiedns.com/private-ipv6](http://simplifiedns.com/private-ipv6)

### Private IP addresses

We can use the following IP address ranges as private or local IP addresses, which will not be visible on the internet:

1. 10.0.0.0 to 10.255.255.255
2. 172.16.0.0 to 172.31.255.255
3. 192.168.0.0 to 192.168.255.255

### Networking tools

1. `ping`
2. `tracert`
3. `pathping`
4. `Test-Connection` (PowerShell)
5. `telnet`
6. `Test-NetConnection` (PowerShell)

### Network access protection (NAP)

[docs.microsoft.com/en-us/windows/win32/nap/network-access-protection-start-page](https://docs.microsoft.com/en-us/windows/win32/nap/network-access-protection-start-page)

### Active Directory Domain Services (AD DS)

According to this recommended [introduction](#) by Microsoft:

AD DS provides a centralized system for managing users, computers, and other resources on a network.

[Wikipedia](#) provides an excellent overview of AD DS. Thereafter you should study the Microsoft documentation at [docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc781408\(v=ws.10\)](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc781408(v=ws.10)), [docs.microsoft.com/en-us/windows-server/manage/windows-admin-center/understand/windows-admin-center](https://docs.microsoft.com/en-us/windows-server/manage/windows-admin-center/understand/windows-admin-center) and [docs.microsoft.com/en-us/windows-server/identity/identity-and-access](https://docs.microsoft.com/en-us/windows-server/identity/identity-and-access).

From "Installing and Configuring Windows Server 2012 Training Guide" by Mitch Tulloch:

Active Directory Domain Services (AD DS) provides a distributed database and directory service that stores and manages information about the users, computers, groups, shares, printers, and other types of objects that comprise an organization's IT infrastructure.

Windows Server 2019 does not provide any new AD functionality or even forest level (cf. [www.virtualizationhowto.com/2018/12/upgrading-windows-server-2016-domain-controller-dc-to-windows-server-2019](http://www.virtualizationhowto.com/2018/12/upgrading-windows-server-2016-domain-controller-dc-to-windows-server-2019)).

First we need to install the AD DS role via Server Manager or PowerShell. Then the server needs to be promoted to a domain controller. See [blogs.technet.microsoft.com/canitpro/2017/02/22/step-by-step-setting-up-active-directory-in-windows-server-2016](http://blogs.technet.microsoft.com/canitpro/2017/02/22/step-by-step-setting-up-active-directory-in-windows-server-2016) and [www.moderndeployment.com/windows-server-2019-active-directory-installation-beginners-guide](http://www.moderndeployment.com/windows-server-2019-active-directory-installation-beginners-guide).

In PS we can use `Install-ADDSForest` to create a new forest, domain and domain controller.

To join a computer to an AD domain, see [www.businessnewsdaily.com/11030-join-windows-server-2016-to-ad-domain.html](http://www.businessnewsdaily.com/11030-join-windows-server-2016-to-ad-domain.html).

To do it remotely via PowerShell:

```
Install-WindowsFeature -name AD-Domain-Services -ComputerName winsecore -Credential (Get-Credential CORP\Administrator)
```

```
Invoke-Command -ComputerName winsecore -credential (get-credential CORP\Administrator) {Import-Module ADDSDeployment; Install-ADDSDomainController -NoGlobalCatalog:$False -CreateDNSDelegation:$False -Credential (Get-Credential CORP\Administrator) -CriticalReplicationOnly:$False -DatabasePath 'C:\Windows\NTDS' -DomainName 'corp.contoso.com' -InstallDNS:$True -LogPath 'C:\Windows\NTDS' -NoRebootOnCompletion:$False -SiteName 'Default-First-Site-Name' -SysVolPath 'C:\Windows\SysVol' }
```

Don't forget to set the DNS server.

If you encounter an error mentioning a duplicate SID, for instance because you cloned a VM, you need to generate a new SID (cf. [www.mustbegeek.com/generate-new-sid-server-2012](http://www.mustbegeek.com/generate-new-sid-server-2012)), which will reset a lot of settings.



Excellent explanations of Active Directory names can be found at [docs.microsoft.com/en-us/windows/win32/ad/naming-properties](https://docs.microsoft.com/en-us/windows/win32/ad/naming-properties) and [social.technet.microsoft.com/wiki/contents/articles/1773.ldap-path-active-directory-distinguished-and-relative-distinguished-names.aspx](https://social.technet.microsoft.com/wiki/contents/articles/1773.ldap-path-active-directory-distinguished-and-relative-distinguished-names.aspx).

Useful commands include `Get-Command -AD`, `Get-ADDomain | fl name, DomainMode`, `Get-ADForest | fl name, DomainMode`.

You can navigate the AD using PowerShell like so:

```
CD AD:
```

```
DIR
```

```
CD "DC=corp,DC=contoso,DC=com"
```

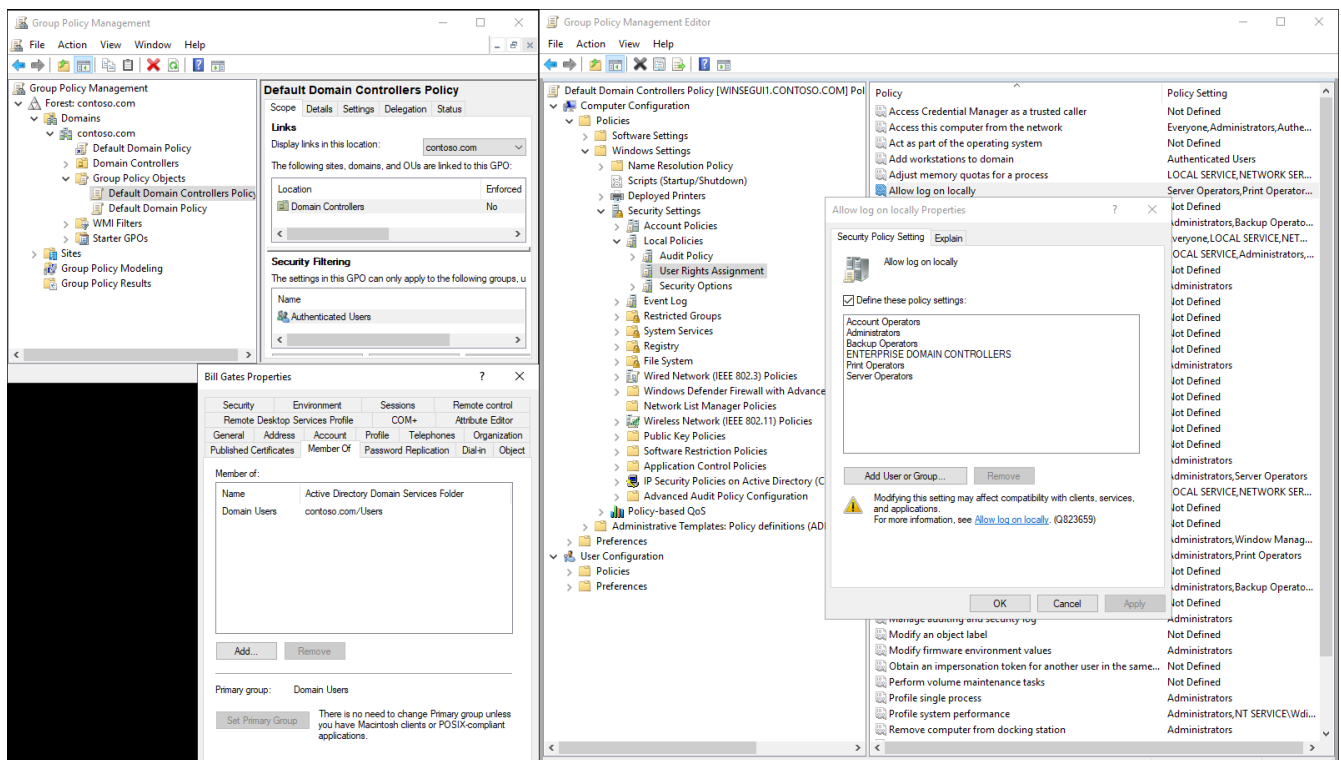
DIR

CD CN=Users

DIR | FT -a

Get-ADUser -Filter {name -like "\*"}

When you add a new non-admin user and try to log them in, you're likely to get the following error message: "The sign-in method you are trying to use isn't allowed. For more info, contact your network administrator". This is due to the default domain controllers policy. To solve this problem, open the Default Domain Controllers Policy and check who is allowed to log on locally:



In this case you can see, that the user is only member of the group Domain Users which is not allowed to log on locally. So either you add the user to a group that is allowed to log on locally or you add at least one of the user's groups to the allowed ones. Then you run `gpupdate /force` and the user should be able to log in.

If you cannot contact a domain controller, see [theitbros.com/active-directory-domain-controller-could-not-be-contacted](http://theitbros.com/active-directory-domain-controller-could-not-be-contacted).

[cp-mlxprod-static.microsoft.com/05329-1025/en-us/content/content\\_8xaxrrjy\\_1704984382/04232015094701.pptx](http://cp-mlxprod-static.microsoft.com/05329-1025/en-us/content/content_8xaxrrjy_1704984382/04232015094701.pptx)

[channel9.msdn.com/Series/Using-PowerShell-for-Active-Directory](http://channel9.msdn.com/Series/Using-PowerShell-for-Active-Directory)

[channel9.msdn.com/Series/Active-Directory-Video-Series/ADs-Reliance-on-DNS](http://channel9.msdn.com/Series/Active-Directory-Video-Series/ADs-Reliance-on-DNS)

[www.varonis.com/blog/top-10-active-directory-tutorials-web](http://www.varonis.com/blog/top-10-active-directory-tutorials-web)

[www.slideshare.net/nishadsukumaran/active-directory-training](http://www.slideshare.net/nishadsukumaran/active-directory-training)

[www.youtube.com/watch?v=nKcrVtvZvpk](http://www.youtube.com/watch?v=nKcrVtvZvpk)

[www.youtube.com/watch?v=J8uw3GNZxzQ](http://www.youtube.com/watch?v=J8uw3GNZxzQ)

## Group types and scopes

[www.tech-faq.com/understanding-group-types-and-scopes.html](http://www.tech-faq.com/understanding-group-types-and-scopes.html)

[docs.microsoft.com/en-us/windows/security/identity-protection/access-control/active-directory-security-groups](https://docs.microsoft.com/en-us/windows/security/identity-protection/access-control/active-directory-security-groups)

[www.serverbrain.org/active-directory-infrastructure-2003/understanding-group-types-and-scopes.html](http://www.serverbrain.org/active-directory-infrastructure-2003/understanding-group-types-and-scopes.html)

## DNS

[docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/reviewing-dns-concepts](https://docs.microsoft.com/en-us/windows-server/identity/ad-ds/plan/reviewing-dns-concepts)

## Kerberos

[www.youtube.com/watch?v=VpBCJ8vS7T0](https://www.youtube.com/watch?v=VpBCJ8vS7T0)

[www.youtube.com/watch?v=kp5d8Yv3-0c](https://www.youtube.com/watch?v=kp5d8Yv3-0c)

[web.mit.edu/kerberos](http://web.mit.edu/kerberos)

[msdn.microsoft.com/en-us/library/bb742516.aspx](https://msdn.microsoft.com/en-us/library/bb742516.aspx)

[technet.microsoft.com/en-us/library/hh831553\(v=ws.11\).aspx](https://technet.microsoft.com/en-us/library/hh831553(v=ws.11).aspx)

[web.mit.edu/kerberos/dialogue.html](http://web.mit.edu/kerberos/dialogue.html)

## Replication

To list commands relevant to replication: **Get-Command Replication** To list replication partners: **Get-ADReplicationPartnerMetadata -target corp.contoso.com**

To register the AD Schema MMC snap-in, see [www.briandesmond.com/active-directory/how-to-register-active-directory-schema-mmc-snap-in](http://www.briandesmond.com/active-directory/how-to-register-active-directory-schema-mmc-snap-in).

## Flexible Single Master Operation (FSMO)

[techgenix.com/fsmo-roles-in-active-directory](http://techgenix.com/fsmo-roles-in-active-directory)

[blogs.technet.microsoft.com/canitpro/2017/05/24/step-by-step-migrating-active-directory-fsmo-roles-from-windows-server-2012-r2-to-2016](https://blogs.technet.microsoft.com/canitpro/2017/05/24/step-by-step-migrating-active-directory-fsmo-roles-from-windows-server-2012-r2-to-2016)

[social.technet.microsoft.com/wiki/contents/articles/6736.powershell-move-transferring-or-seizing-fsmo-roles-to-another-domain-controller.aspx](https://social.technet.microsoft.com/wiki/contents/articles/6736.powershell-move-transferring-or-seizing-fsmo-roles-to-another-domain-controller.aspx)

[support.microsoft.com/en-gb/help/197132/active-directory-fsmo-roles-in-windows](https://support.microsoft.com/en-gb/help/197132/active-directory-fsmo-roles-in-windows)

[theitbros.com/ntdsutil](http://theitbros.com/ntdsutil)

[support.microsoft.com/en-us/help/255504/using-ntdsutil-exe-to-transfer-or-seize-fsmo-roles-to-a-domain-control](https://support.microsoft.com/en-us/help/255504/using-ntdsutil-exe-to-transfer-or-seize-fsmo-roles-to-a-domain-control)

## Fine-grained password policy (FGPC)

[docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc770394\(v=ws.10\)](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc770394(v=ws.10))

[docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc770842\(v=ws.10\)](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc770842(v=ws.10))

## Group policy



Study [4sysops.com/archives/understanding-group-policy-order](https://4sysops.com/archives/understanding-group-policy-order) to understand group policy order. Also see [serverfault.com/questions/510624/gpo-enforced-precedence](https://serverfault.com/questions/510624/gpo-enforced-precedence).



The official group policy settings reference can be found at [tinyurl.com/policysettings-xls](https://tinyurl.com/policysettings-xls).

[4sysops.com/archives/four-ways-to-search-for-group-policy-settings](https://4sysops.com/archives/four-ways-to-search-for-group-policy-settings)

[www.linkedin.com/learning/windows-server-2012-r2-manage-group-policy/configure-group-policy-preferences](https://www.linkedin.com/learning/windows-server-2012-r2-manage-group-policy/configure-group-policy-preferences)

[docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc786524\(v%3dws.10\)](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc786524(v%3dws.10))

[docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/ee461027\(v=technet.10\)](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/ee461027(v=technet.10))

[docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc733087\(v%3dws.10\)](https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2008-R2-and-2008/cc733087(v%3dws.10))

[social.technet.microsoft.com/wiki/contents/articles/51876.group-policy-filtering-and-permission.aspx](https://social.technet.microsoft.com/wiki/contents/articles/51876.group-policy-filtering-and-permission.aspx)

[www.infoworld.com/article/3117286/tutorial-the-joys-of-windows-servers-group-policies.html](https://www.infoworld.com/article/3117286/tutorial-the-joys-of-windows-servers-group-policies.html)

[www.grouppolicy.biz/tutorials](https://www.grouppolicy.biz/tutorials)

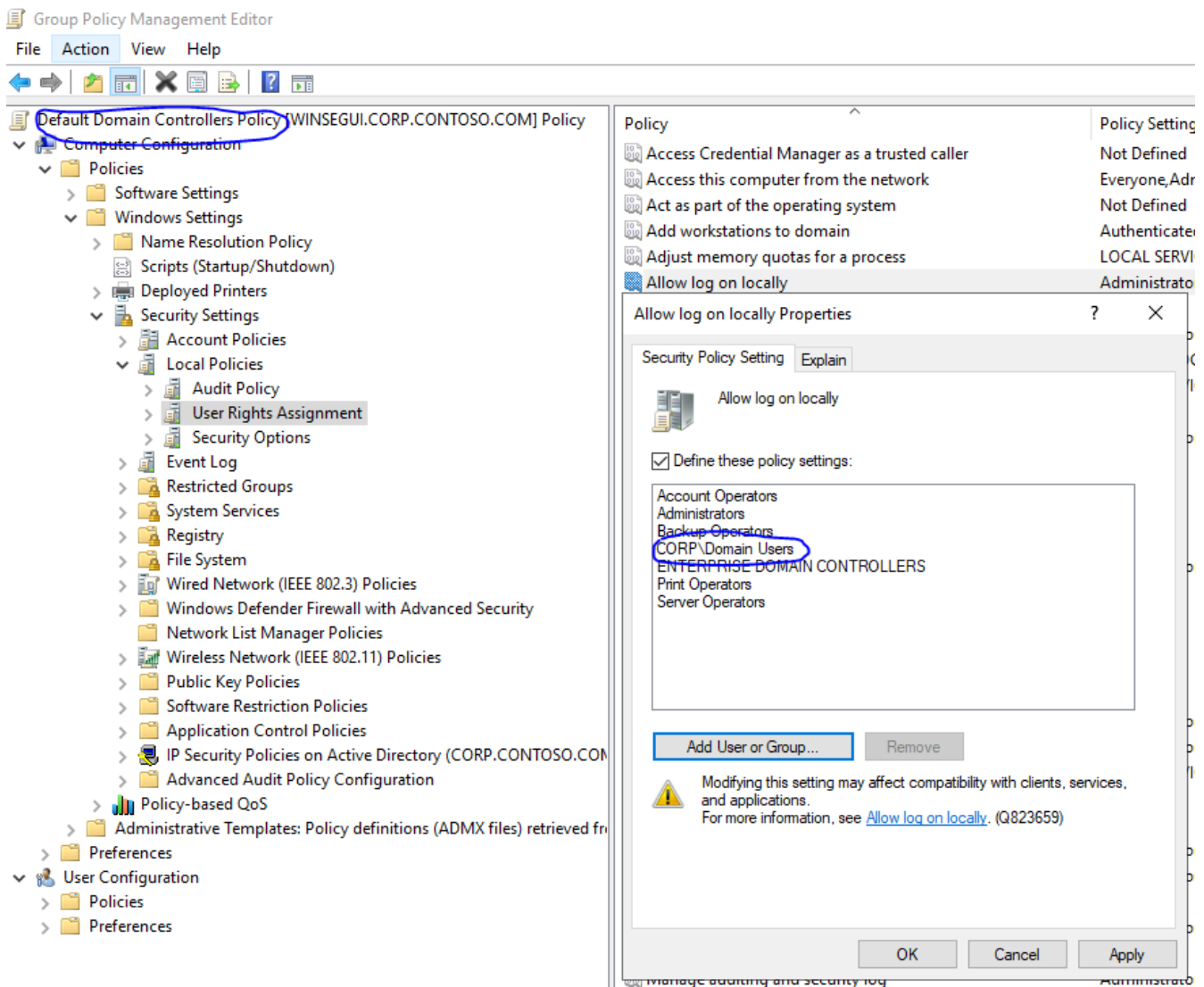
[www.mdmandgpanswers.com/blogs](https://www.mdmandgpanswers.com/blogs)

[sdmsoftware.com/gpoguy](https://sdmsoftware.com/gpoguy)

[sdmsoftware.com/gpoguy/free-tools/video-training](https://sdmsoftware.com/gpoguy/free-tools/video-training)

[www.heidelbergit.dk](https://www.heidelbergit.dk)

If you're getting "The sign-in method you're trying to use isn't allowed" for a new domain user, see [link](#). You might need to add the user/group to the "Allow log on locally" policy, run `gpupdate /force` and reboot.



To redirect the default location of new users of new users and computers, see [support.microsoft.com/kb/324949](https://support.microsoft.com/kb/324949).

### Software restriction policies and AppLocker

[docs.microsoft.com/en-us/windows-server/identity/software-restriction-policies/administer-software-restriction-policies](https://docs.microsoft.com/en-us/windows-server/identity/software-restriction-policies/administer-software-restriction-policies)

[docs.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/applocker-policies-deployment-guide](https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/applocker-policies-deployment-guide)

[social.technet.microsoft.com/wiki/contents/articles/5211-how-to-configure-applocker-group-policy-to-prevent-software-from-running.aspx](https://social.technet.microsoft.com/wiki/contents/articles/5211-how-to-configure-applocker-group-policy-to-prevent-software-from-running.aspx)

[blogs.technet.microsoft.com/askpfeplat/2016/06/27/applocker-another-layer-in-the-defense-in-depth-against-malware](https://blogs.technet.microsoft.com/askpfeplat/2016/06/27/applocker-another-layer-in-the-defense-in-depth-against-malware)

[www.petri.com/setting-application-control-policies-microsofts-applocker](https://www.petri.com/setting-application-control-policies-microsofts-applocker)

[michaelfirsov.wordpress.com/applocker-in-windows-10-enterprise](https://michaelfirsov.wordpress.com/applocker-in-windows-10-enterprise)

[docs.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/configure-the-application-identity-service](https://docs.microsoft.com/en-us/windows/security/threat-protection/windows-defender-application-control/applocker/configure-the-application-identity-service)

Policy can be configured in GPME: Computer Configuration → Policies → Windows Settings →

Security Settings → Application Control Policies → AppLocker

To force the download of the latest domain policies from the domain controllers use `gpupdate /force`.

### Group Policy Management Console Scripting Samples

[gallery.technet.microsoft.com/Group-Policy-Management-17a5f840](https://gallery.technet.microsoft.com/Group-Policy-Management-17a5f840)

### Microsoft Security Compliance Toolkit

[docs.microsoft.com/en-us/windows/security/threat-protection/security-compliance-toolkit-10](https://docs.microsoft.com/en-us/windows/security/threat-protection/security-compliance-toolkit-10)

### Windows Management Instrumentation (WMI) filters

[docs.microsoft.com/en-us/windows/win32/wmisdk/wmi-start-page](https://docs.microsoft.com/en-us/windows/win32/wmisdk/wmi-start-page)

[docs.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/scripting-articles/ms974579\(v=msdn.10\)](https://docs.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/scripting-articles/ms974579(v=msdn.10))

[docs.microsoft.com/en-us/biztalk/core/technical-reference/viewing-the-wmi-core-server-classes-in-cim-studio](https://docs.microsoft.com/en-us/biztalk/core/technical-reference/viewing-the-wmi-core-server-classes-in-cim-studio)

[gallery.technet.microsoft.com/scriptcenter/Scriptomatic-21-aka-shame-9cdc28b5](https://gallery.technet.microsoft.com/scriptcenter/Scriptomatic-21-aka-shame-9cdc28b5)

### PowerShell

[docs.microsoft.com/en-us/powershell](https://docs.microsoft.com/en-us/powershell)

[www.infoworld.com/article/3000612/power-of-powershell-essential-tips-every-windows-admin-will-love.html](http://www.infoworld.com/article/3000612/power-of-powershell-essential-tips-every-windows-admin-will-love.html)

[channel9.msdn.com/Series/Using-PowerShell-for-Active-Directory](http://channel9.msdn.com/Series/Using-PowerShell-for-Active-Directory)

[www.powertheshell.com/topic/learnpowershell](http://www.powertheshell.com/topic/learnpowershell)

[www.powershellgallery.com](http://www.powershellgallery.com)



Study [www.powertheshell.com/topic/learnpowershell/firststeps/firststeps!](http://www.powertheshell.com/topic/learnpowershell/firststeps/firststeps!)

To determine the PowerShell version use `$PSVersionTable` (cf. [www.powertheshell.com/topic/learnpowershell/firststeps/psversion](http://www.powertheshell.com/topic/learnpowershell/firststeps/psversion)).

To enable remote PowerShell access, see [www.powertheshell.com/topic/learnpowershell/firststeps/configuration\[^\]](http://www.powertheshell.com/topic/learnpowershell/firststeps/configuration[^]).

To reset a computer machine password, see [docs.microsoft.com/en-us/powershell/module/microsoft.powershell.management/reset-computermachinepassword](https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.management/reset-computermachinepassword).

### Windows Admin Center

[www.microsoft.com/en-us/cloud-platform/windows-admin-center](https://www.microsoft.com/en-us/cloud-platform/windows-admin-center)

## AppLocker

[blogs.technet.microsoft.com/askpfeplat/2016/06/27/applocker-another-layer-in-the-defense-in-depth-against-malware](https://blogs.technet.microsoft.com/askpfeplat/2016/06/27/applocker-another-layer-in-the-defense-in-depth-against-malware)

## Firewall

```
`Get-NetFirewallProfile -Name Domain -PolicyStore ActiveStore` grep 443
```

## BitLocker

## EFS

## ISATAP

[directaccess.richardhicks.com/category/isatap](https://directaccess.richardhicks.com/category/isatap)

## iSCSI

[www.it-zeugs.de/iscsi-target-unter-windows-server-2016-einrichten.html](http://www.it-zeugs.de/iscsi-target-unter-windows-server-2016-einrichten.html)

## Resilient File System (ReFS)

[docs.microsoft.com/en-us/windows-server/storage/refs/refs-overview](https://docs.microsoft.com/en-us/windows-server/storage/refs/refs-overview)

## IPsec

[docs.microsoft.com/en-us/windows/desktop/fwp/ipsec-configuration](https://docs.microsoft.com/en-us/windows/desktop/fwp/ipsec-configuration)

## Netdom

```
netdom query fsmo
```

## Windows containers

[docs.microsoft.com/en-us/virtualization/windowscontainers/about/index](https://docs.microsoft.com/en-us/virtualization/windowscontainers/about/index)

[docs.microsoft.com/en-us/virtualization/windowscontainers/quick-start/quick-start-windows-server](https://docs.microsoft.com/en-us/virtualization/windowscontainers/quick-start/quick-start-windows-server)

## Remote Desktop Services

[www.youtube.com/watch?v=oIos0TbZfjY](https://www.youtube.com/watch?v=oIos0TbZfjY)

## Tips & Tricks

### Remove Internet Explorer

[www.tecklyfe.com/remove-internet-explorer-ie-windows-server](http://www.tecklyfe.com/remove-internet-explorer-ie-windows-server)

**See computer name**

```
hostname
```

**Rename computer using PS**

```
Rename-Computer -NewName <xyz> -Restart
```

**Get current IP configuration**

```
Get-NetIPAddress
```

**Assign a new IP address**

```
New-NetIPAddress -InterfaceIndex <x> -IPAddress <y> -PrefixLength <z> -DefaultGateway <abc>
```

**Renew IP configuration**

```
ipconfig /renew
```

**Restart computer**

```
shutdown /r /t 0
```

## 2.2.2. Client

### Administration

#### DNS

Display data in the DNS resolver cache:

```
ipconfig /displaydns
```

Flush DNS resolver cache:

```
ipconfig /flushdns
```

Renew all DHCP leases and reregister all DNS names:

```
ipconfig /registerdns
```

Trace a route to a server:

```
tracert hostname  
pathping hotname
```

## Check disk health

[www.maketecheasier.com/check-hard-disk-health-windows](http://www.maketecheasier.com/check-hard-disk-health-windows)

## Robocopy

See [technet.microsoft.com/en-us/library/cc733145%28WS.10%29.aspx#BKMK\\_remarks](http://technet.microsoft.com/en-us/library/cc733145%28WS.10%29.aspx#BKMK_remarks).

Useful options include `/S /PURGE /COPYALL /DCOPY:T /R:2 /W:1 /A-:SH /xj`.

To get rid of undeletable recursive directories (cf. [answers.microsoft.com/en-us/windows/forum/windows\\_7-files/windows-7-infinite-loop-while-using-robocopy/20f32f0c-4cb9-4125-923d-6a57e4d27232](http://answers.microsoft.com/en-us/windows/forum/windows_7-files/windows-7-infinite-loop-while-using-robocopy/20f32f0c-4cb9-4125-923d-6a57e4d27232)) we create an empty dir and then `robocopy empty dest /MIR`.

## Tips & Tricks

Get rid of language bar icon	<a href="http://www.askvg.com/fix-input-indicator-icon-comes-back-in-taskbar-notification-area-after-restarting-windows[^]">www.askvg.com/fix-input-indicator-icon-comes-back-in-taskbar-notification-area-after-restarting-windows[^]</a>
	<a href="http://www.howtogeek.com/howto/windows/bring-misplaced-off-screen-windows-back-to-your-desktop-keyboard-trick">www.howtogeek.com/howto/windows/bring-misplaced-off-screen-windows-back-to-your-desktop-keyboard-trick</a>
Fix DPC WATCHDOG VIOLATION	<a href="http://www.youtube.com/watch?v=VXgAFmPI21g">www.youtube.com/watch?v=VXgAFmPI21g</a>
How to Add Programs, Files, and Folders to System Startup in Windows	<a href="http://www.howtogeek.com/208224/how-to-add-programs-files-and-folders-to-system-startup-in-windows-8.1">www.howtogeek.com/208224/how-to-add-programs-files-and-folders-to-system-startup-in-windows-8.1</a>
List of Windows tools	<a href="http://www.ghacks.net/2017/06/11/list-of-windows-tools">www.ghacks.net/2017/06/11/list-of-windows-tools</a>
	<a href="http://windowsreport.com/weather-app-live-tile-not-working-windows-10">windowsreport.com/weather-app-live-tile-not-working-windows-10</a>
How to access the BIOS	<a href="http://www.addictivetips.com/windows-tips/access-bios-pc">www.addictivetips.com/windows-tips/access-bios-pc</a>
Get rid of hiberfil.sys	<a href="http://www.howtogeek.com/howto/15140/what-is-hiberfil.sys-and-how-do-i-delete-it">www.howtogeek.com/howto/15140/what-is-hiberfil.sys-and-how-do-i-delete-it</a>
How to convert Windows installations to virtual machine images	<a href="http://lt3000.blogspot.com/2018/09/on-chinas-putative-real-estate.html">lt3000.blogspot.com/2018/09/on-chinas-putative-real-estate.html</a>
15 Windows 10 Run Commands Everyone Should Learn	<a href="http://helpdeskgeek.com/windows-10/windows-10-run-commands-everyone-should-learn">helpdeskgeek.com/windows-10/windows-10-run-commands-everyone-should-learn</a>
How to launch apps automatically during startup on Windows 10	<a href="http://www.windowscentral.com/how-launch-apps-automatically-during-startup-windows-10">www.windowscentral.com/how-launch-apps-automatically-during-startup-windows-10</a>
	<a href="http://stackoverflow.com/questions/8976287/recursive-unzipping-with-7z-exe">stackoverflow.com/questions/8976287/recursive-unzipping-with-7z-exe</a>

How To Pin Shortcut To Taskbar When There's No "Pin To Taskbar" Option	<a href="https://media-moon.com/blog/windows-10-how-to-pin-shortcut-to-taskbar-when-theres-no-pin-to-taskbar-option">media-moon.com/blog/windows-10-how-to-pin-shortcut-to-taskbar-when-theres-no-pin-to-taskbar-option</a>
Disk drive defragmentation	<a href="https://www.tenforums.com/performance-maintenance/118761-optimize-drives-says-i-have-ssd-when-i-have-hdd.html">www.tenforums.com/performance-maintenance/118761-optimize-drives-says-i-have-ssd-when-i-have-hdd.html</a>

# Chapter 3. Tools of the trade

## 3.1. Portable work environment

### 3.1.1. Vagrant

**Vagrant** serves to isolate dependencies and their configuration within a single disposable, consistent environment, without sacrificing any of the tools you are used to working with (editors, browsers, debuggers, etc.). Once you or someone else creates a single Vagrantfile, you just need to vagrant up and everything is installed and configured for you to work. Other members of your team create their development environments from the same configuration, so whether you are working on Linux, Mac OS X, or Windows, all your team members are running code in the same environment, against the same dependencies, all configured the same way. Say goodbye to "works on my machine" bugs.

### 3.1.2. Docker

**Docker** containers wrap a piece of software in a complete filesystem that contains everything needed to run: code, runtime, system tools, system libraries – anything that can be installed on a server. This guarantees that the software will always run the same, regardless of its environment.

#### Cheat sheet

If you run into `docker: Got permission denied while trying to connect to the Docker daemon socket at unix:///var/run/docker.sock: run usermod -a -G docker <user>` (cf. [techoverflow.net/2017/03/01/solving-docker-permission-denied-while-trying-to-connect-to-the-docker-daemon-socket](https://techoverflow.net/2017/03/01/solving-docker-permission-denied-while-trying-to-connect-to-the-docker-daemon-socket)).

Kill all docker containers at once: [gist.github.com/evanscottgray/8571828](https://gist.github.com/evanscottgray/8571828)

### 3.1.3. VirtualBox

The techniques and features that [VirtualBox](#) provides are useful for several scenarios:

- Running multiple operating systems simultaneously. VirtualBox allows you to run more than one operating system at a time. This way, you can run software written for one operating system on another (for example, Windows software on Linux or a Mac) without having to reboot to use it. Since you can configure what kinds of "virtual" hardware should be presented to each such operating system, you can install an old operating system such as DOS or OS/2 even if your real computer's hardware is no longer supported by that operating system.
- Easier software installations. Software vendors can use virtual machines to ship entire software configurations. For example, installing a complete mail server solution on a real machine can be a tedious task. With VirtualBox, such a complex setup (then often called an "appliance") can be packed into a virtual machine. Installing and running a mail server becomes as easy as importing such an appliance into VirtualBox.
- Testing and disaster recovery. Once installed, a virtual machine and its virtual hard disks can be considered a "container" that can be arbitrarily frozen, woken up, copied, backed up, and transported between hosts.
- On top of that, with the use of another VirtualBox feature called "snapshots", one can save a particular state of a virtual machine and revert back to that state, if necessary. This way, one can freely experiment with a computing environment. If something goes wrong (e.g. after installing misbehaving software or infecting the guest with a virus), one can easily switch back to a previous snapshot and avoid the need of frequent backups and restores.
- Any number of snapshots can be created, allowing you to travel back and forward in virtual machine time. You can delete snapshots while a VM is running to reclaim disk space.
- Infrastructure consolidation. Virtualization can significantly reduce hardware and electricity costs. Most of the time, computers today only use a fraction of their potential power and run with low average system loads. A lot of hardware resources as well as electricity is thereby wasted. So, instead of running many such physical computers that are only partially used, one can pack many virtual machines onto a few powerful hosts and balance the loads between them.

[druss.co/2015/06/fix-vt-x-is-not-available-verr\\_vmx\\_no\\_vmx-in-virtualbox](http://druss.co/2015/06/fix-vt-x-is-not-available-verr_vmx_no_vmx-in-virtualbox)

[www.nakivo.com/blog/virtualbox-network-setting-guide](http://www.nakivo.com/blog/virtualbox-network-setting-guide)

[www.oracle.com/technical-resources/articles/it-infrastructure/admin-manage-vbox-cli.html](http://www.oracle.com/technical-resources/articles/it-infrastructure/admin-manage-vbox-cli.html)

## 3.2. Integrated Development Environments

### 3.2.1. Codiad

**Codiad** is a web-based IDE framework with a small footprint and minimal requirements.

### 3.2.2. Atom

Get the editor from [atom.io](http://atom.io). Installation instructions can be found in the [flight manual](#). It is recommended to install the two Asciidoc packages for Atom as mentioned in [Asciidoc live preview](#). Activate live preview with `ctrl-shift-a` as described in [asciidoc-preview](#).

#### Portable installation

Follow Portable Mode in [flight-manual.atom.io/getting-started/sections/installing-atom](http://flight-manual.atom.io/getting-started/sections/installing-atom). To create the `.atom` folder for the portable installation see [gist.github.com/ozh/4131243](https://gist.github.com/ozh/4131243).

### 3.2.3. PhpStorm

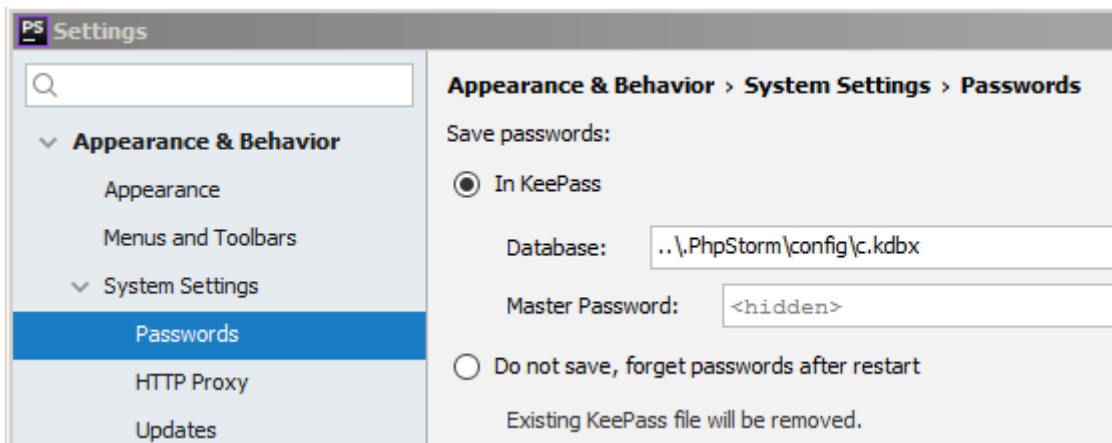
PhpStorm is the ideal IDE for web app development. It provides full database and server integration.

#### Portable installation

To install PhpStorm on a portable drive, go to [JetBrains](#) and click the **Download** button. Cancel the automatic download of the `.exe` file and right click **direct link**, select **Copy Link Location**, paste the link into a new tab and replace the `exe` extension with `zip`, then press enter. This will download and open the zipped version of PhpStorm. Extract it to your portable drive.

Open the file `bin/idea.properties`, replace the line starting with `#idea.config.path` with  
`idea.config.path=${idea.home}/.WebIde/config,` `#idea.system.path` with  
`idea.system.path=${idea.home}/.WebIde/system,` `#idea.plugins.path` with  
`idea.plugins.path=${idea.config.path}/plugins` and `#idea.log.path` with  
`idea.log.path=${idea.system.path}/log.`

In order to avoid having to reenter your SSH password after every logout set the following:

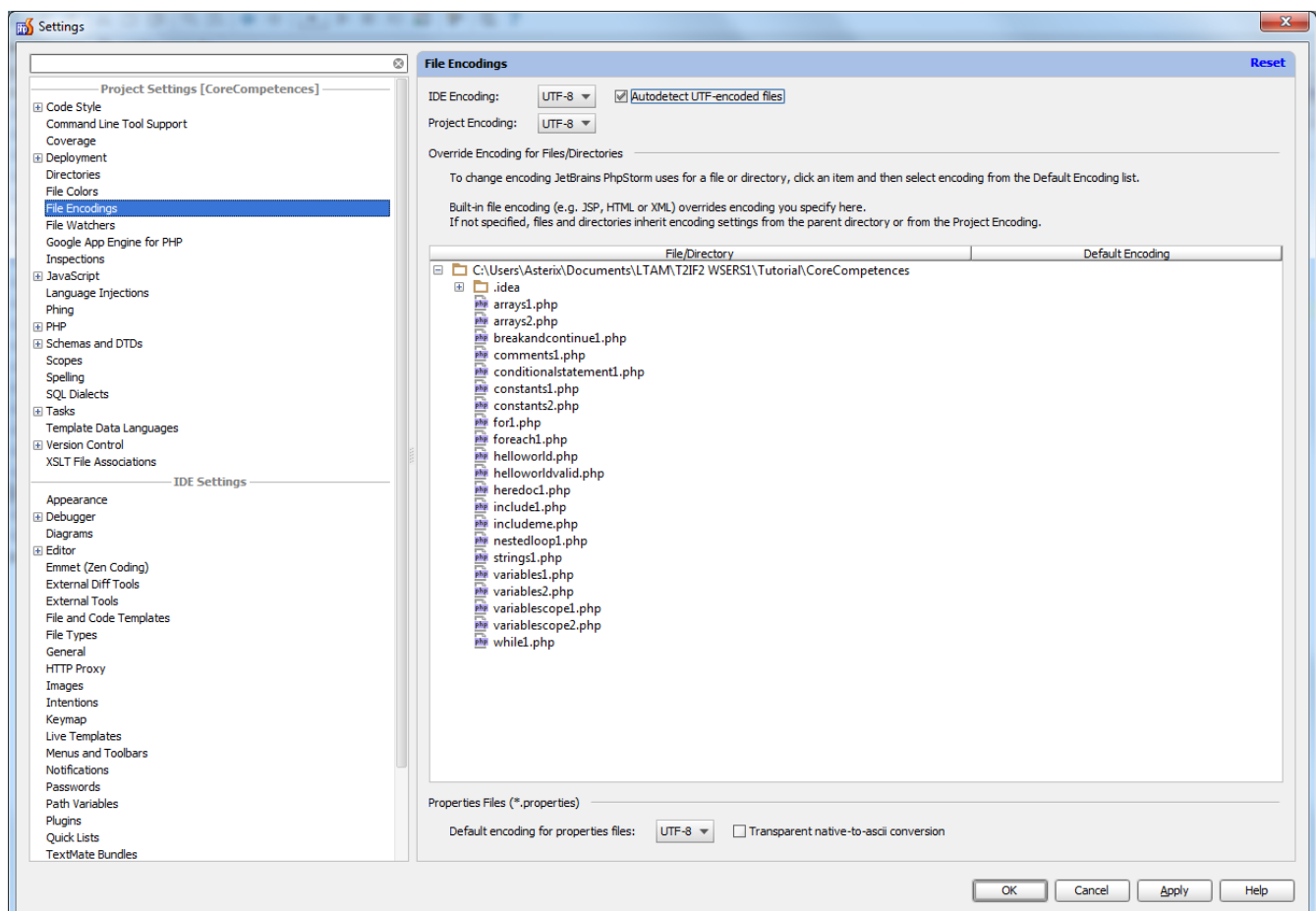


This works only for project deployment servers, not for global ones. Unfortunately this does not work for DB sources for which you'll have to reenter the SSH password.

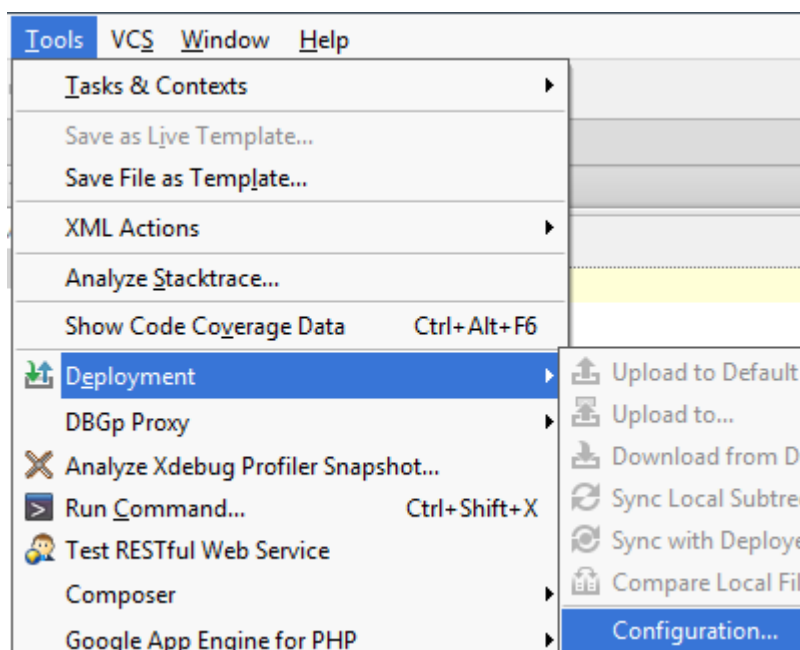
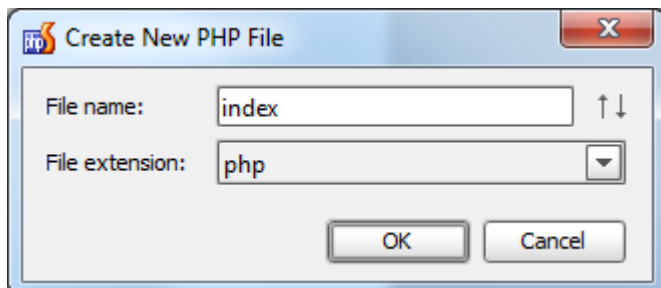
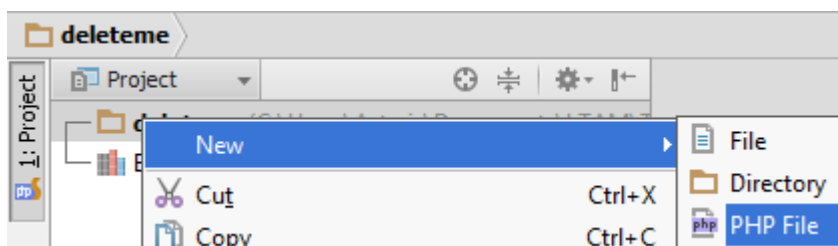
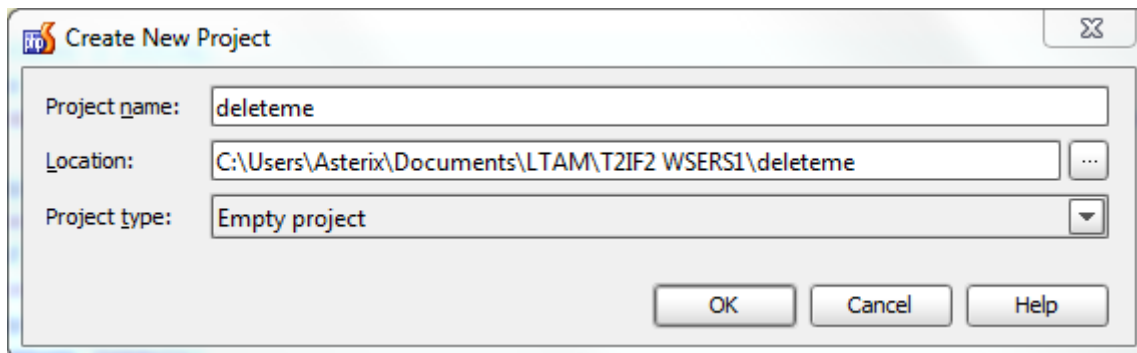
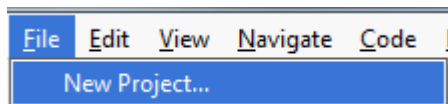
If you have settings from another PhpStorm installation that you'd like to import, you can do this via **File → Import Settings...**

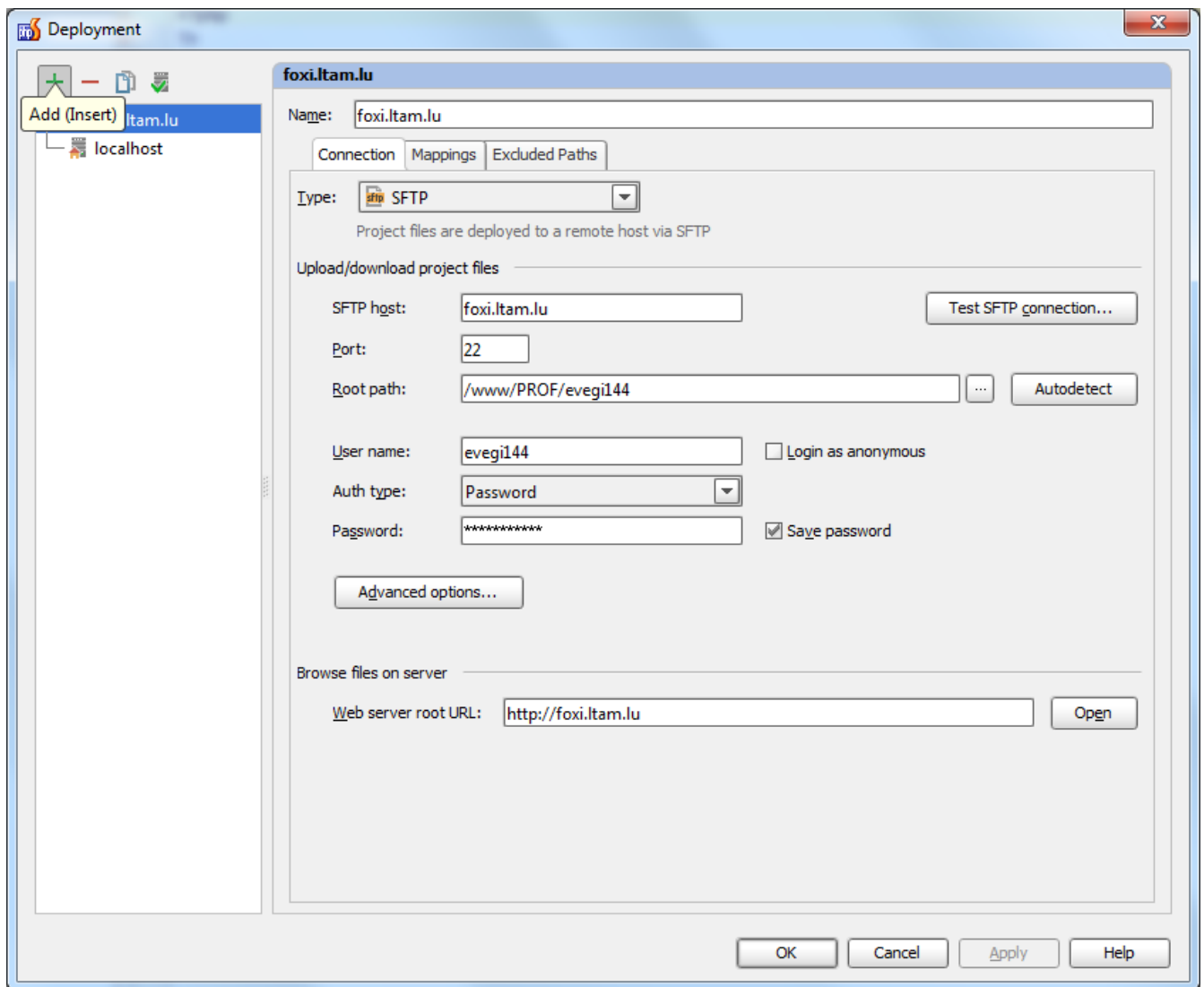
## Project setup

First, we set all file encodings to UTF-8 in order to avoid any problems with special characters. Search for file encodings in the search box. Given the constant PhpStorm UI changes, your file encoding settings may be located in a different place:



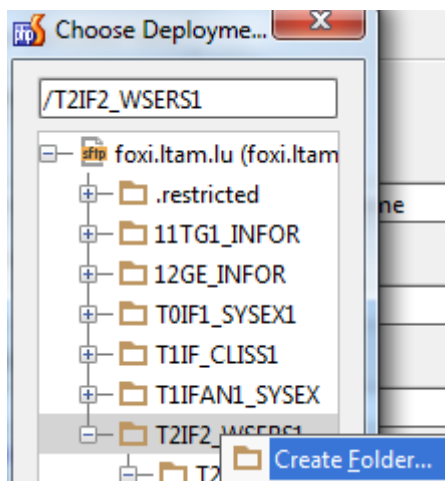
Now we configure a new project:





Deployment path on server 'foxi.ltam.lu':

Shift+Er



**foxi.ltam.lu**

Name:

Connection Mappings Excluded Paths





Local path:

Deployment path on server 'foxi.ltam.lu':


Web path on server 'foxi.ltam.lu':

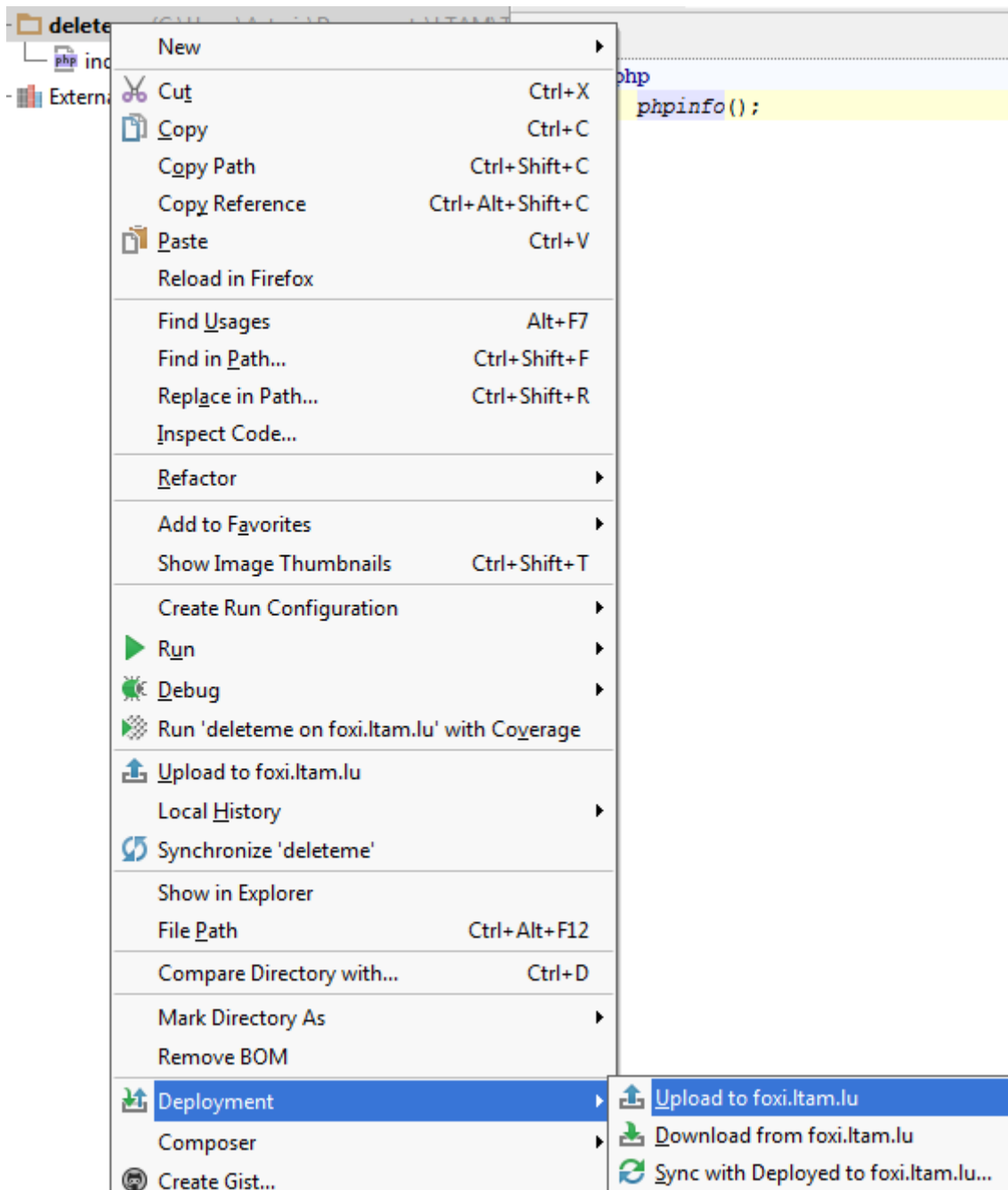
Project URL: [http://foxi.ltam.lu/PROF/evengi144/T2IF2\\_WSERS1/deleteme](http://foxi.ltam.lu/PROF/evengi144/T2IF2_WSERS1/deleteme)

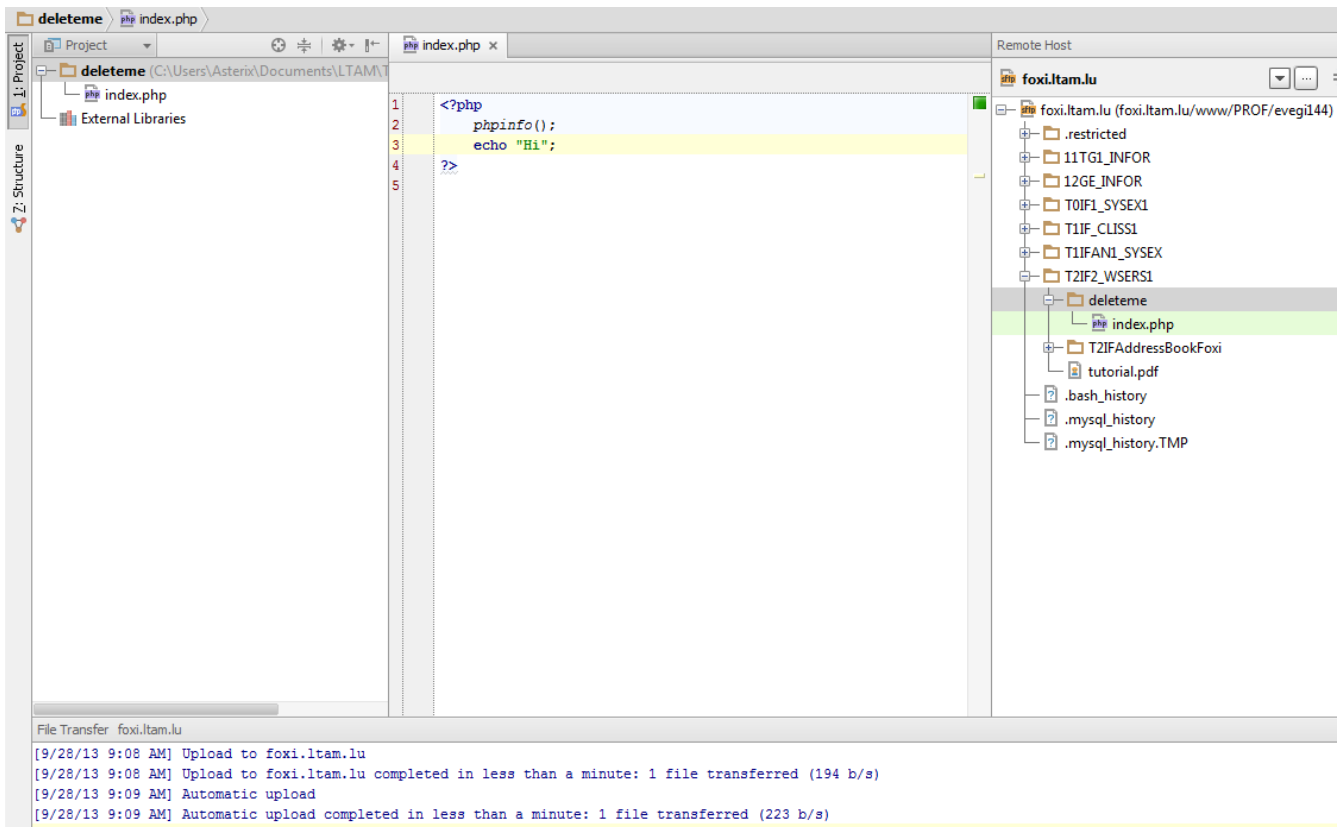
Tools VCS Window Help

- Tasks & Contexts
- Save as Live Template...
- Save File as Template...
- XML Actions
- Analyze Stacktrace...
- Show Code Coverage Data Ctrl+Alt+F6
-  **Deployment**
  - Upload to Default S
  - Upload to...
  - Download from De
  - Sync Local Subtree
  - Sync with Deployed
  - Compare Local File
  - Configuration...
  - Options...
  - Automatic Upload**
- DBGp Proxy
-  Analyze Xdebug Profiler Snapshot...
-  Run Command... Ctrl+Shift+X
-  Test RESTful Web Service
- Composer
- Google App Engine for PHP

Choose default server for actions

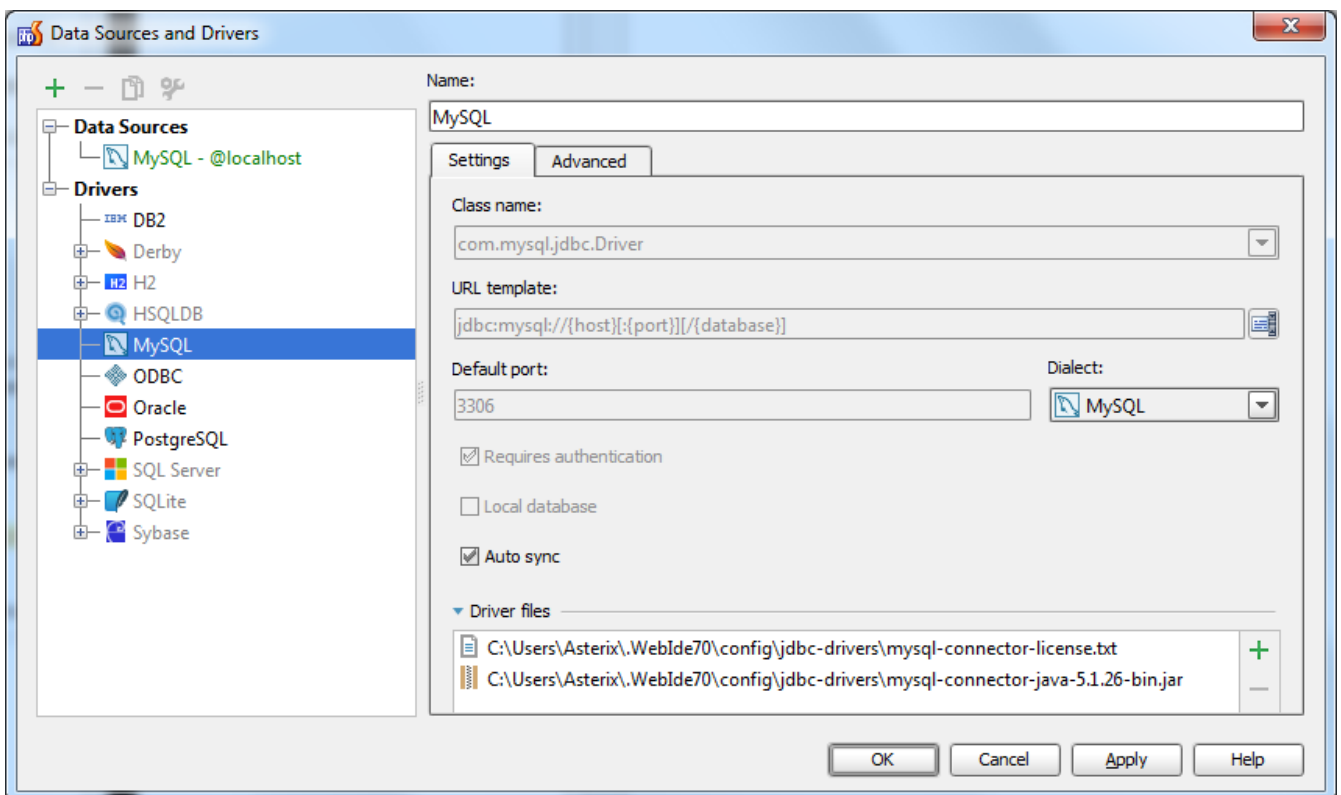
 1. foxi.ltam.lu



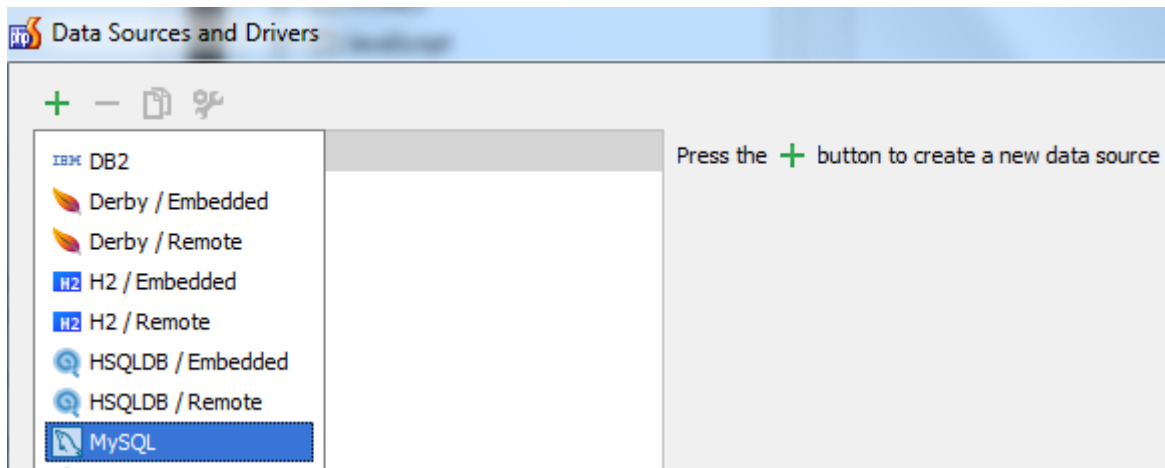


## Database connection setup

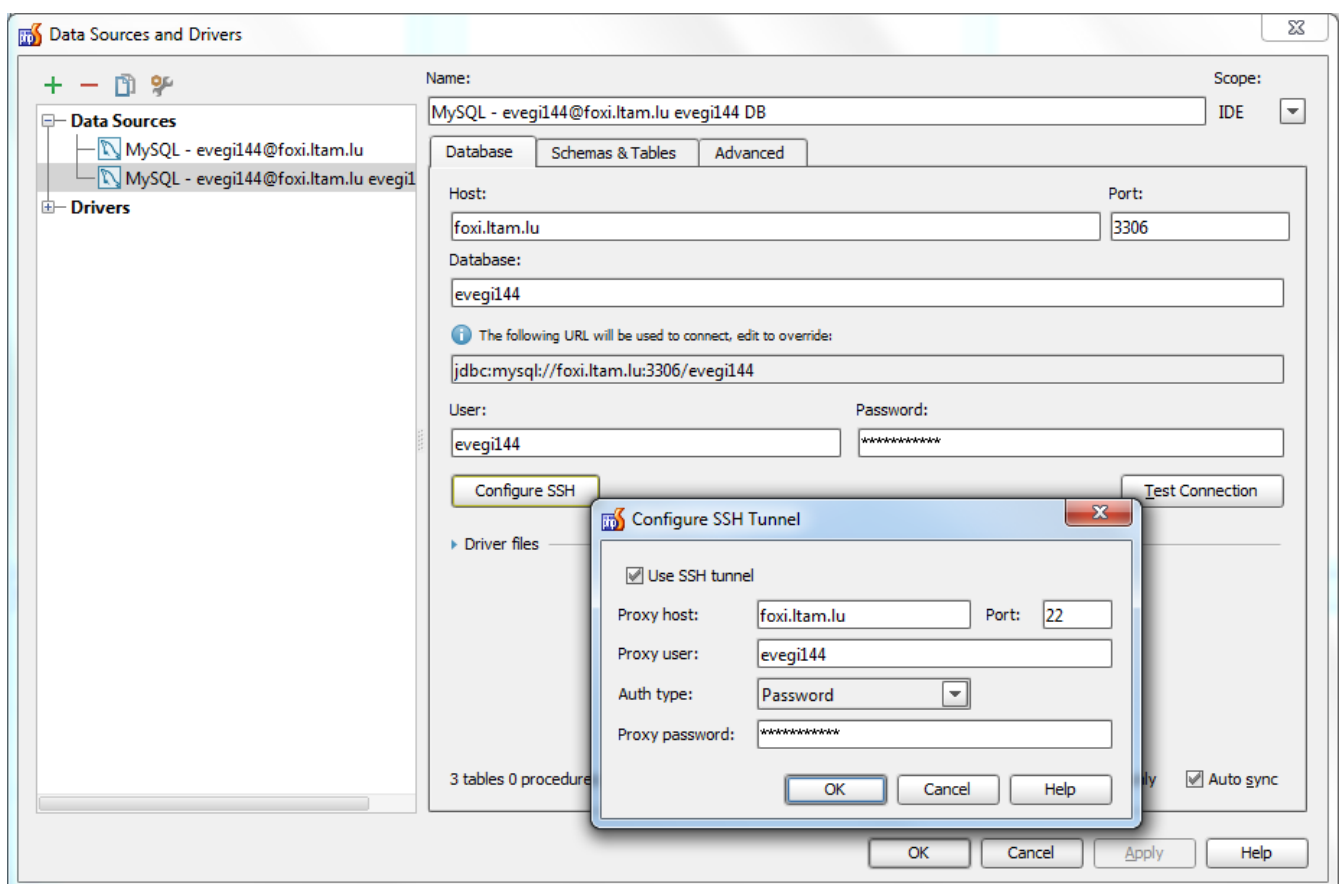
First we need to make sure that the drivers are loaded:



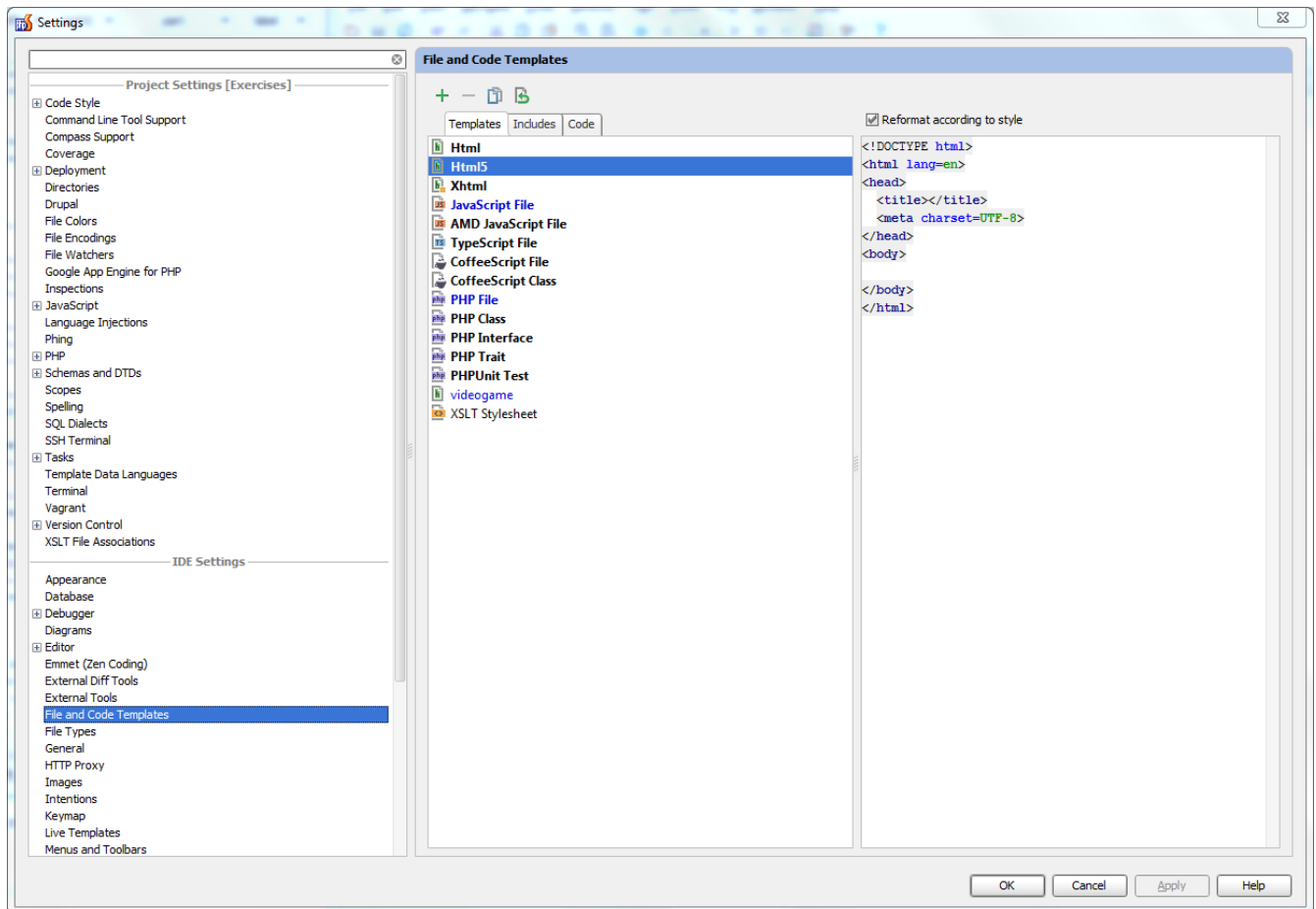
Then we need to create a data source:



Make sure to right click the connection and select **Make Global** so that you don't need to configure it for each project:



## Template adjustment



To improve the performance of PhpStorm see [dev.to/adammcquiff/improve-the-performance-of-webstorm-and-other-jetbrains-ides-11bc](https://dev.to/adammcquiff/improve-the-performance-of-webstorm-and-other-jetbrains-ides-11bc).

### 3.2.4. NetBeans

Download [NetBeans](#).

https://netbeans.org/downloads/ HOME / Download

## NetBeans IDE 8.0.2 Download

8.0.1 | 8.0.2 | Development | Archive

Email address (optional):

Subscribe to newsletters: ☒ Monthly ☐ Weekly ☒ NetBeans can contact me at this address

IDE Language: **English** Platform: **Windows** Note: Greyed out technologies are not supported for this platform

### NetBeans IDE Download Bundles

Supported technologies *	Java SE	Java EE	C/C++	HTML5 & PHP	All
NetBeans Platform SDK	•	•			•
Java SE	•	•			•
Java FX	•	•			•
Java EE		•			•
Java ME					•
HTML5		•		•	•
Java Card™ 3 Connected					•
C/C++			•		•
Groovy					•
PHP				•	•
Bundled servers					
GlassFish Server Open Source Edition 4.1		•			•
Apache Tomcat 8.0.15		•			•
	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>	<a href="#">Download</a>
	Free, 90 MB	Free, 186 MB	Free, 63 MB	Free, 63 MB	Free, 205 MB

\* You can add or remove packs later using the IDE's Plugin Manager (Tools | Plugins).

Java 7 and later versions are required for installing and running the PHP and C/C++ NetBeans Bundles. You can download the latest Java at [java.com](#).

JDK 7 and later versions are required for installing and running the Java SE, Java EE and All NetBeans Bundles. You can download [standalone JDK](#) or download the latest [JDK with NetBeans IDE Java SE bundle](#).

You can start developing applications based on the NetBeans Platform using the NetBeans IDE for Java SE. Learn more about the [NetBeans Platform](#). NetBeans source code and binary builds without bundled runtimes are also available in [zip file](#) format. See also instructions on how to build the IDE from sources or installation instructions.

**Important Legal Information:**

NetBeans Community Distributions are available under a Dual License consisting of the Common Development and Distribution License (CDDL) v1.0 and GNU General Public License (GPL) v2. Such distributions include additional components under separate licenses identified in the License file. See the Third Party License file for external components included in NetBeans and their associated licenses.

If this is the first time you install NetBeans on your device, you need to install the Java Development Kit (JDK) first (point 1 on the screenshot). This will open the following screen:

http://www.oracle.com/technetwork/java/javase/downloads/index.html

Oracle Technology Network > Java > Java SE > Downloads

Overview Downloads Documentation Community Technologies Trainin

### Java SE Downloads

Java SE

Java EE

Java ME

Java SE Support

Java SE Advanced & Suite

Java Embedded

Java DB

Web Tier

Java Card



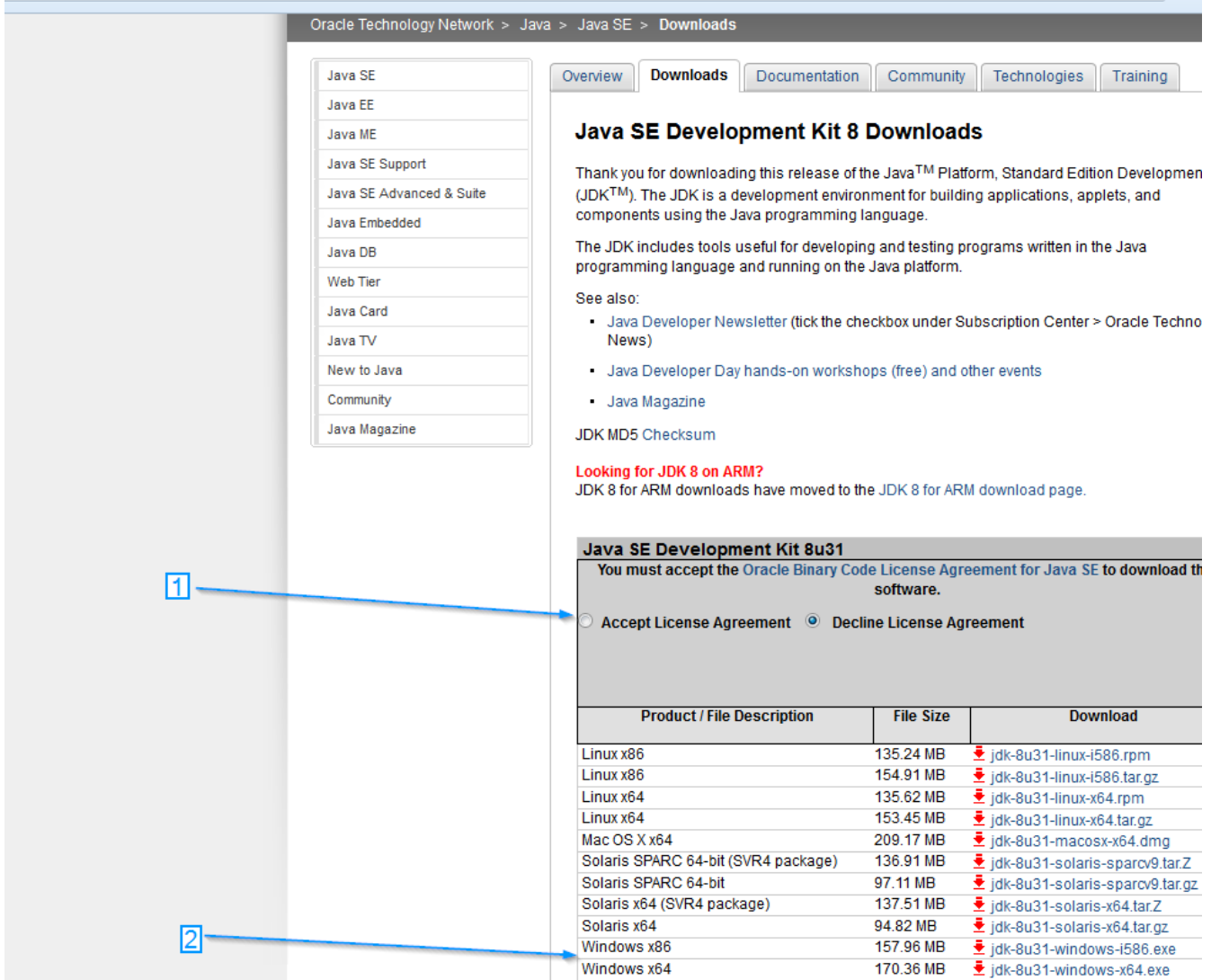
[DOWNLOAD](#)

Java Platform (JDK) 8u31



[DOWNLOAD](#)

NetBeans with JDK 8



Oracle Technology Network > Java > Java SE > Downloads

**Java SE Development Kit 8 Downloads**

Thank you for downloading this release of the Java™ Platform, Standard Edition Development (JDK™). The JDK is a development environment for building applications, applets, and components using the Java programming language.

The JDK includes tools useful for developing and testing programs written in the Java programming language and running on the Java platform.

See also:

- Java Developer Newsletter (tick the checkbox under Subscription Center > Oracle Techno News)
- Java Developer Day hands-on workshops (free) and other events
- Java Magazine

JDK MD5 Checksum

**Looking for JDK 8 on ARM?**  
JDK 8 for ARM downloads have moved to the JDK 8 for ARM download page.

**Java SE Development Kit 8u31**

You must accept the [Oracle Binary Code License Agreement for Java SE](#) to download the software.

☐ Accept License Agreement ☒ Decline License Agreement

Product / File Description	File Size	Download
Linux x86	135.24 MB	<a href="#">jdk-8u31-linux-i586.rpm</a>
Linux x86	154.91 MB	<a href="#">jdk-8u31-linux-i586.tar.gz</a>
Linux x64	135.62 MB	<a href="#">jdk-8u31-linux-x64.rpm</a>
Linux x64	153.45 MB	<a href="#">jdk-8u31-linux-x64.tar.gz</a>
Mac OS X x64	209.17 MB	<a href="#">jdk-8u31-macosx-x64.dmg</a>
Solaris SPARC 64-bit (SVR4 package)	136.91 MB	<a href="#">jdk-8u31-solaris-sparcv9.tar.Z</a>
Solaris SPARC 64-bit	97.11 MB	<a href="#">jdk-8u31-solaris-sparcv9.tar.gz</a>
Solaris x64 (SVR4 package)	137.51 MB	<a href="#">jdk-8u31-solaris-x64.tar.Z</a>
Solaris x64	94.82 MB	<a href="#">jdk-8u31-solaris-x64.tar.gz</a>
Windows x86	157.96 MB	<a href="#">jdk-8u31-windows-i586.exe</a>
Windows x64	170.36 MB	<a href="#">jdk-8u31-windows-x64.exe</a>

Accept the license agreement and select the right JDK version for your operating system.

When the JDK is installed you can install NetBeans. For our purposes we only need the HTML5 + PHP version (point 2).

Alternatively you can install the NetBeans Java SE bundle (point 3), which includes the JDK and NetBeans. This will take you to the following screen:

ORACLE

Sign In/Register Help Country Communities I am a... I want to... Search

Products Solutions Downloads Store Support Training

Oracle Technology Network > Java > Java SE > Downloads

Overview Downloads Documentation Community Technologies Training

### JDK 7u75 with NetBeans 8.0.2

This distribution of the JDK includes the Java SE bundle of NetBeans IDE, which is a powerful integrated development environment for developing applications on the Java platform. [Learn more](#)

You must accept the JDK 7u75 and NetBeans 8 Cobundle License Agreement to download software.

☐ Accept License Agreement ☒ Decline License Agreement

Java SE and NetBeans Cobundle (JDK 7u75 and NB 8.0.2)		
Product / File Description	File Size	Download
Linux x86	210.98 MB	<a href="#">jdk-7u75-nb-8_0_2-linux-i586.sh</a>
Linux x64	207.26 MB	<a href="#">jdk-7u75-nb-8_0_2-linux-x64.sh</a>
Mac OS X x64	290.39 MB	<a href="#">jdk-7u75-nb-8_0_2-macosx-x64.dmg</a>
Windows x86	233.28 MB	<a href="#">jdk-7u75-nb-8_0_2-windows-i586.exe</a>
Windows x64	236.08 MB	<a href="#">jdk-7u75-nb-8_0_2-windows-x64.exe</a>

Accept the license agreement and select the right JDK version for your operating system.

## Portable installation

If you want to install NetBeans on a portable device, you can download it as a zip file (point 4 on the first screenshot in the previous subsection.) This will take you to the following screen:

NetBeans NetBeans IDE NetB

HOME / Download

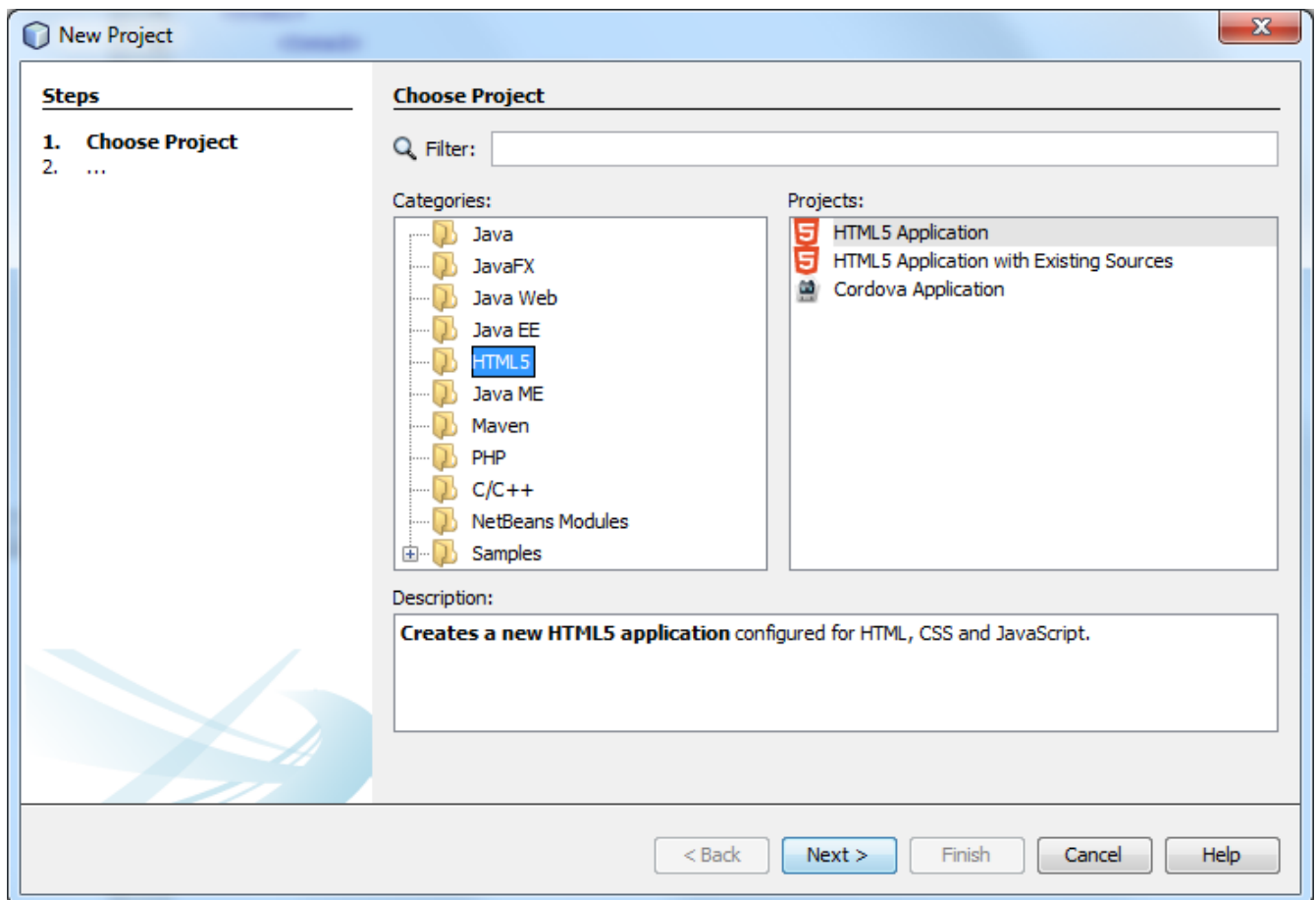
## NetBeans IDE 8.0.2 Download

### Zip Files

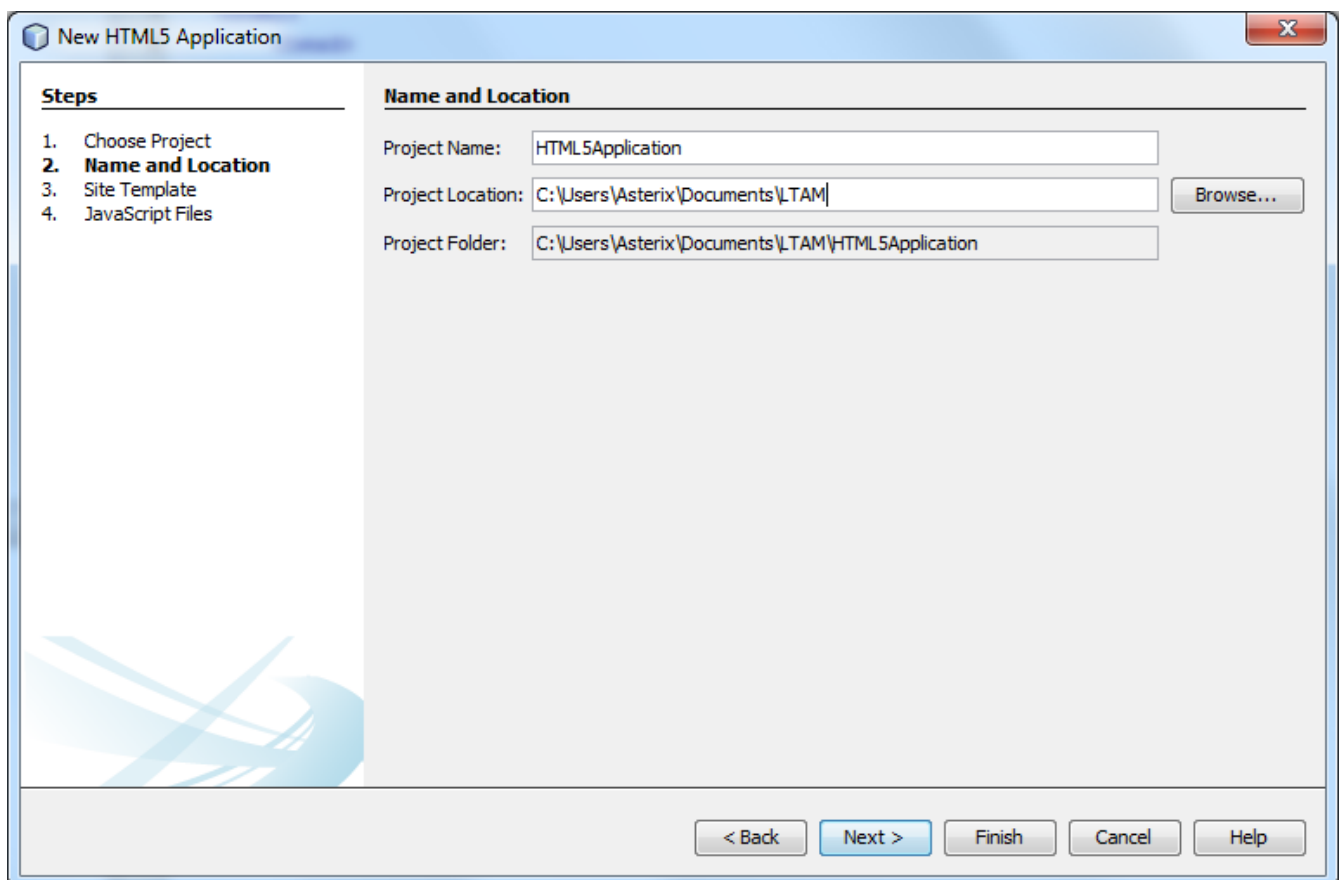
- [hg-l10n-201411181905.zip](#)
- [ide-l10n-201411181905.zip](#)
- [netbeans-8.0.2-201411181905-cpp.zip](#)
- [netbeans-8.0.2-201411181905-javaee.zip](#)
- [netbeans-8.0.2-201411181905-javase.zip](#)
- [netbeans-8.0.2-201411181905-php.zip](#)

## HTML5 Project setup

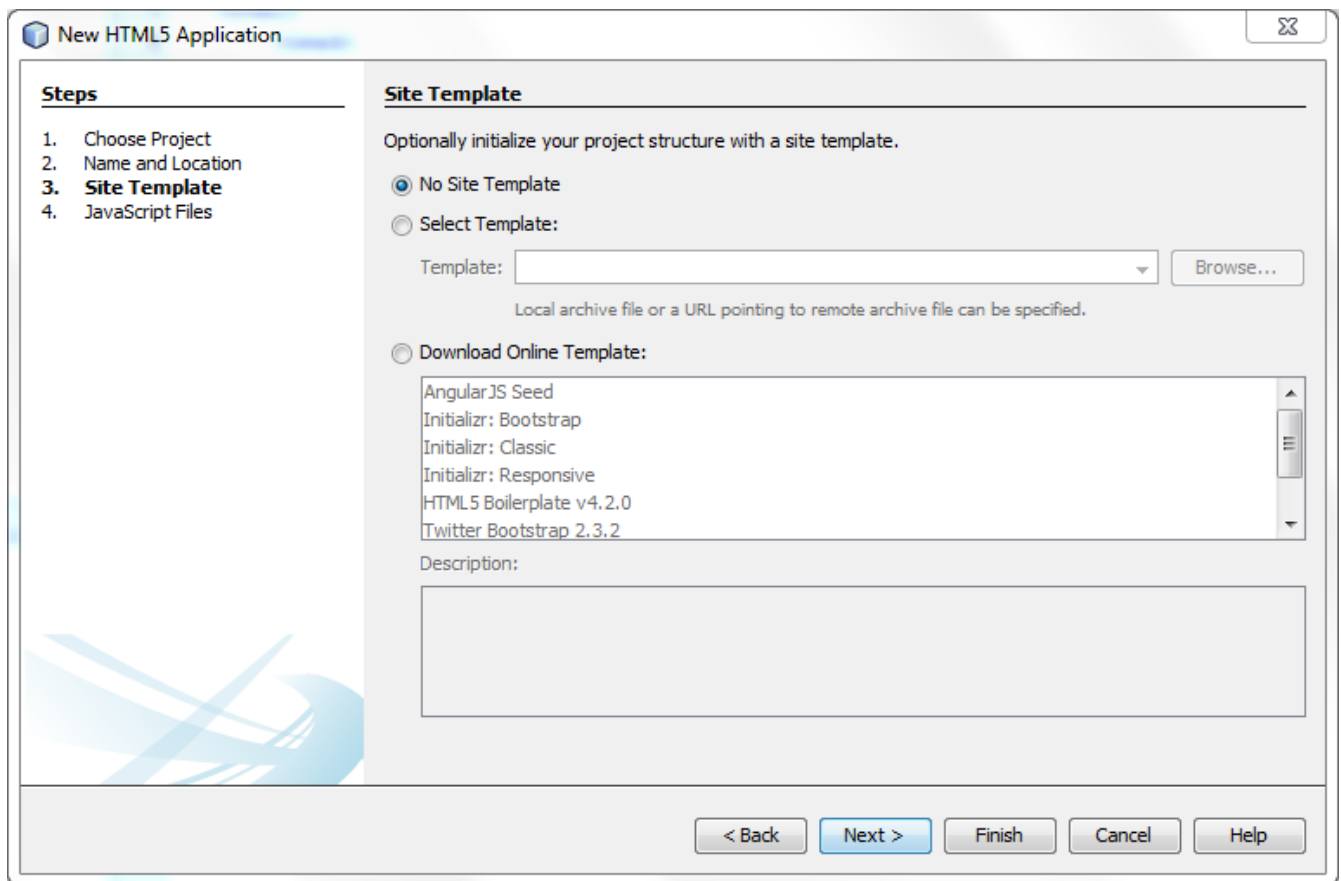
Click on  or File → New Project... or Ctrl + Shift + N:



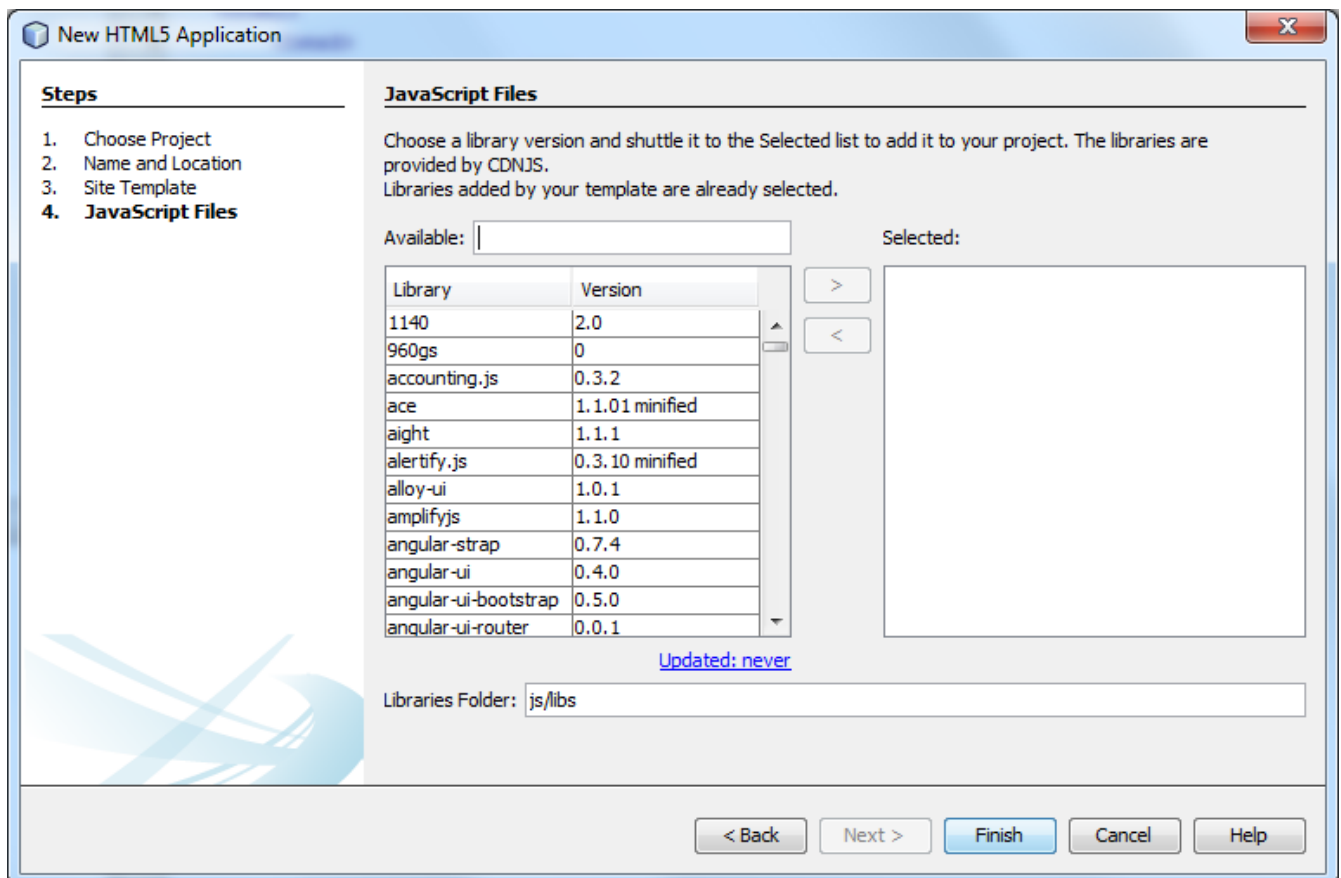
Specify the name of your project and where you'd like to save it:



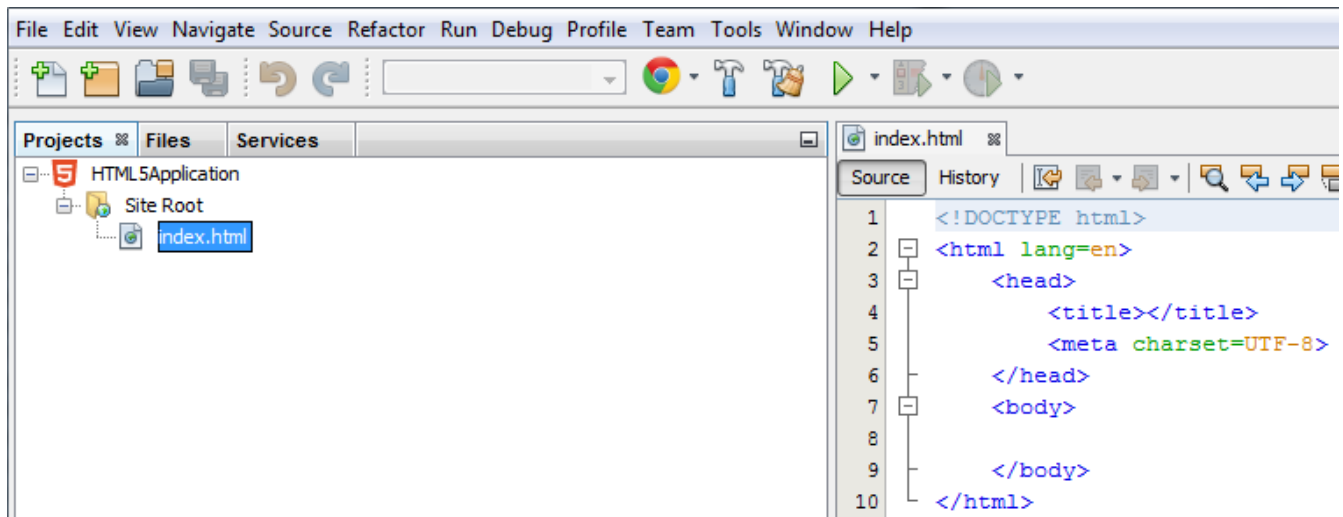
We don't use site templates:



We also don't use a JavaScript library, so we can just click on Finish:

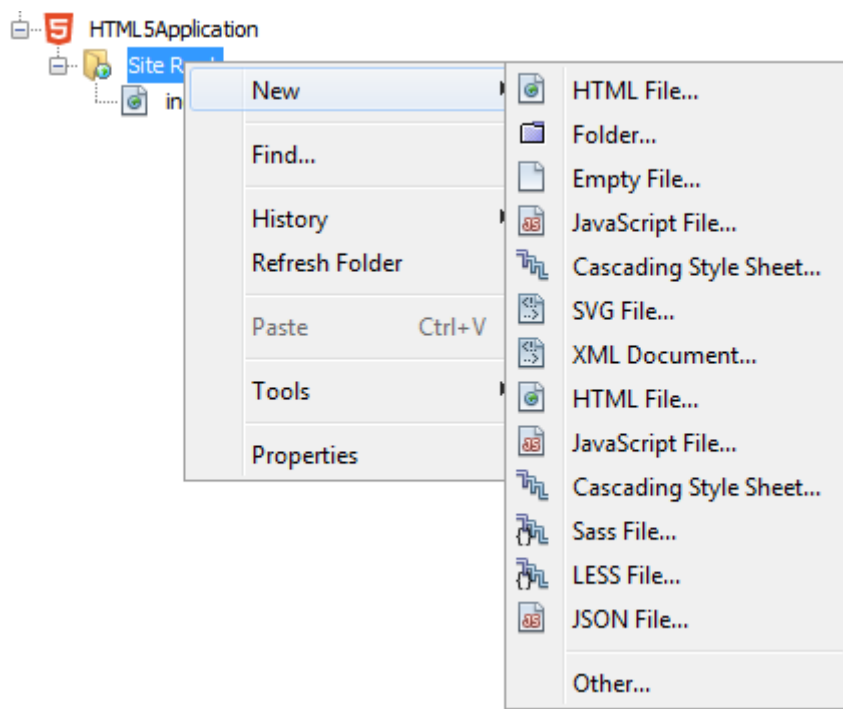


You can now see your new HTML5 project structure in the upper left corner. NetBeans has also opened the `index.html` file, which is the default name of the main project HTML5 file:




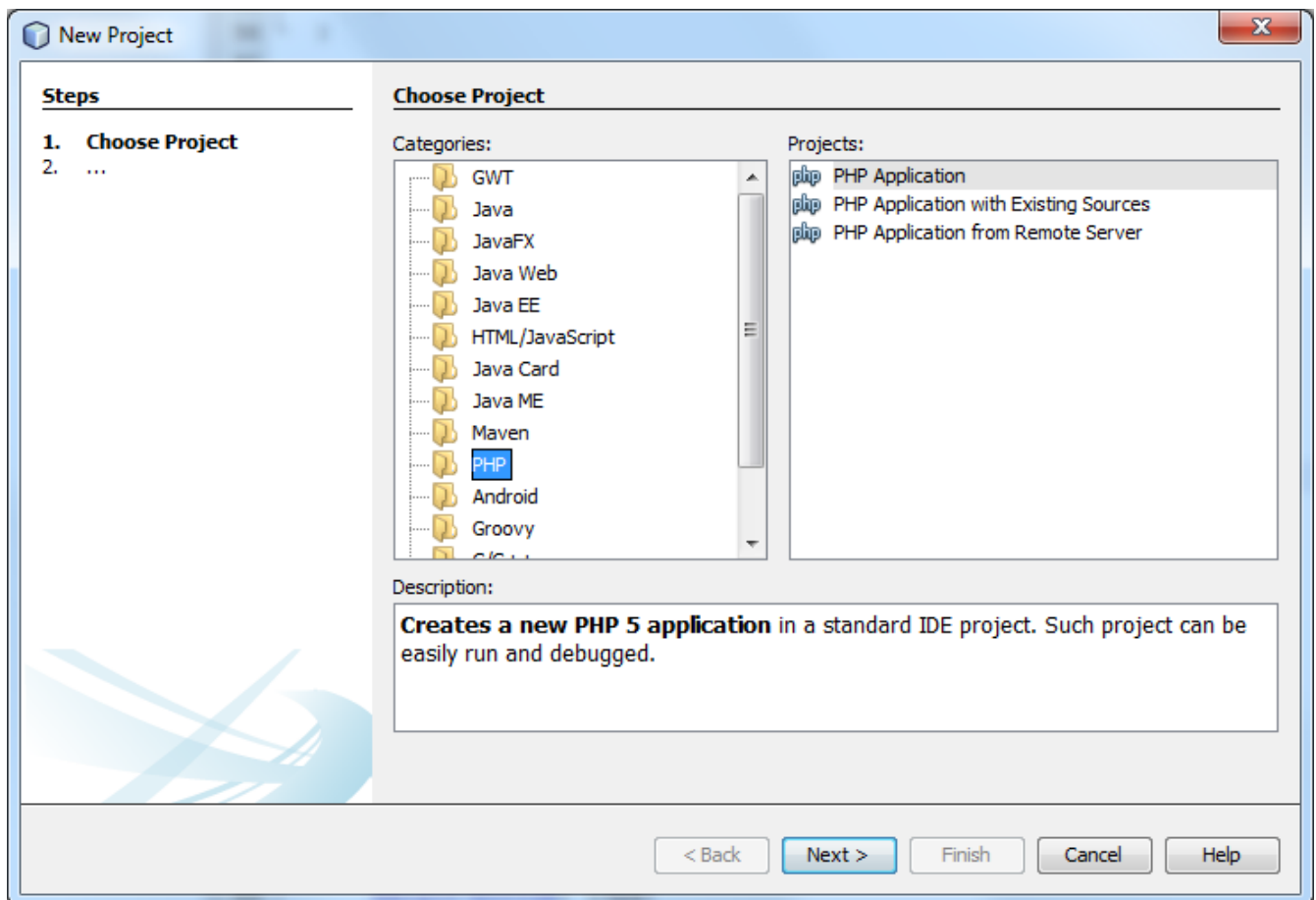
The content of this file is defined in the corresponding template. See [NetBeans templates](#) for guidance on how to change this.

In order to add a new file to your project, right click on Site Root and select the file type:

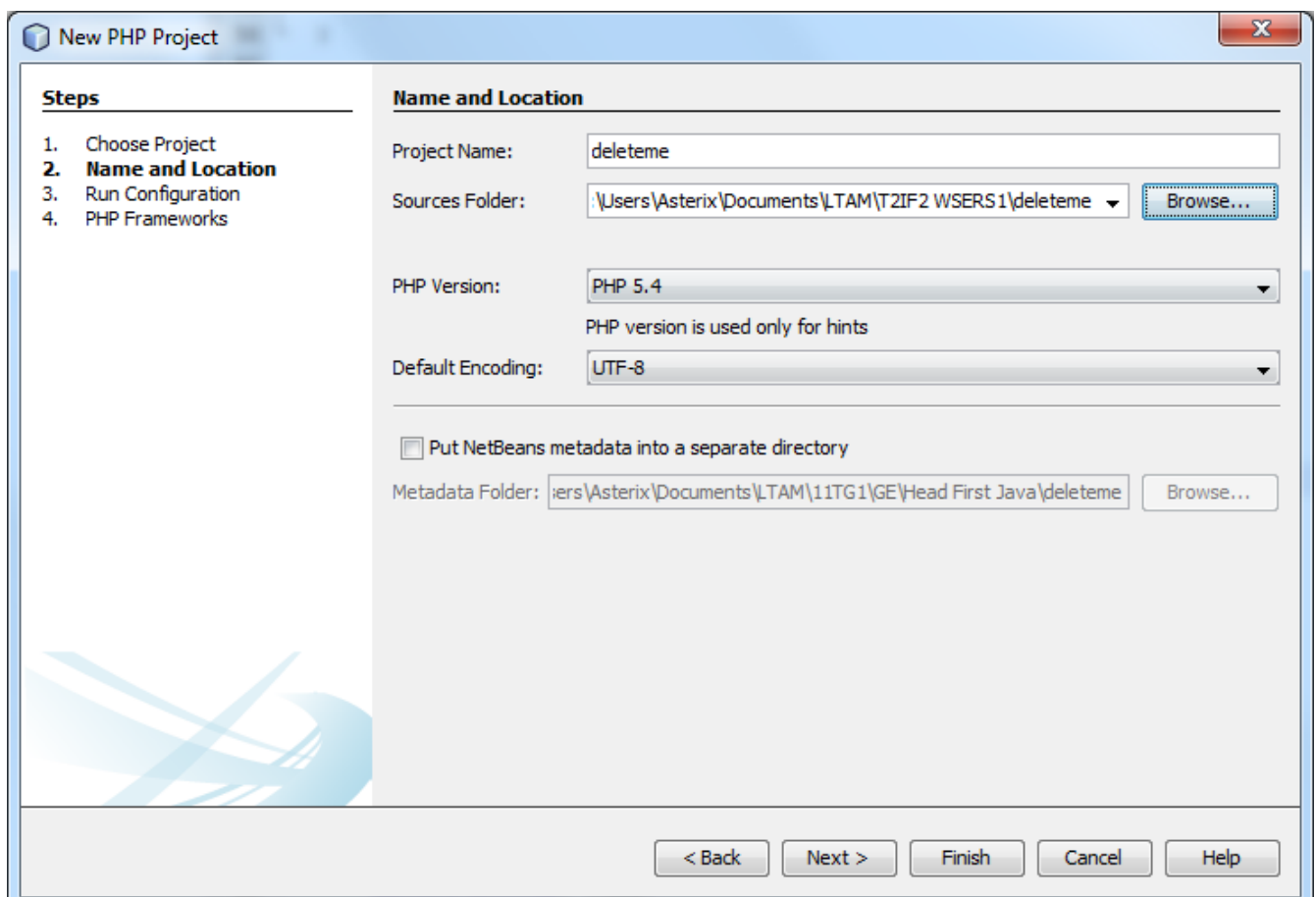


## PHP Project setup

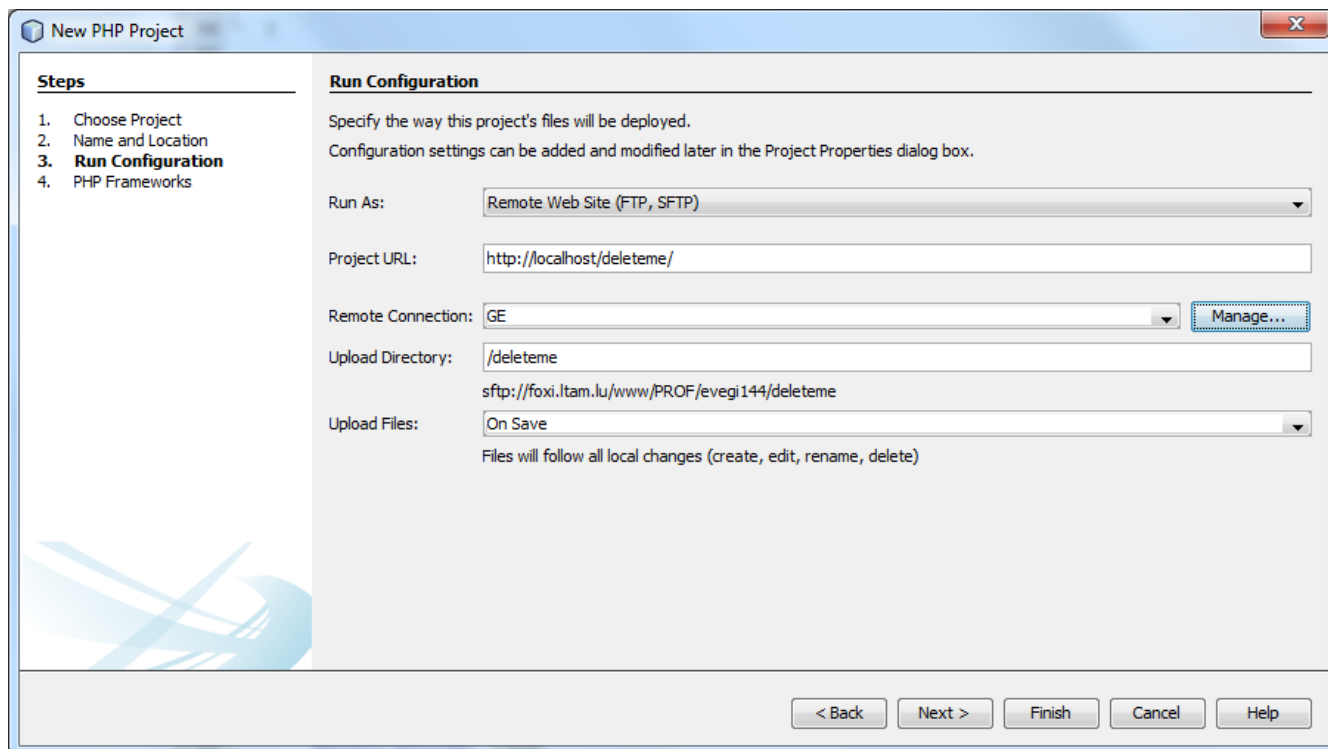
Click on  or File → New Project... or Ctrl + Shift + N. If you are using an older version of NetBeans (< 7.4), you may need to install the PHP plugin via Tools → Plugins → Available Plugins and select PHP. After restarting NetBeans, you should get the following screen when creating a new project:



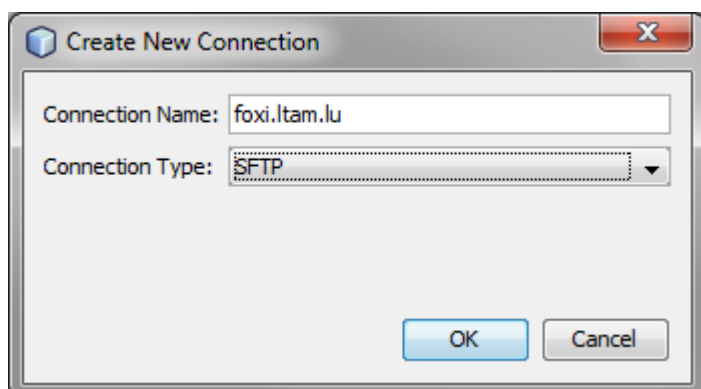
Specify the name of your project and where you'd like to save it. Choose the latest PHP version and keep UTF-8 as the default encoding. The latter makes sure, that non-English characters such as é or ä are handled correctly:



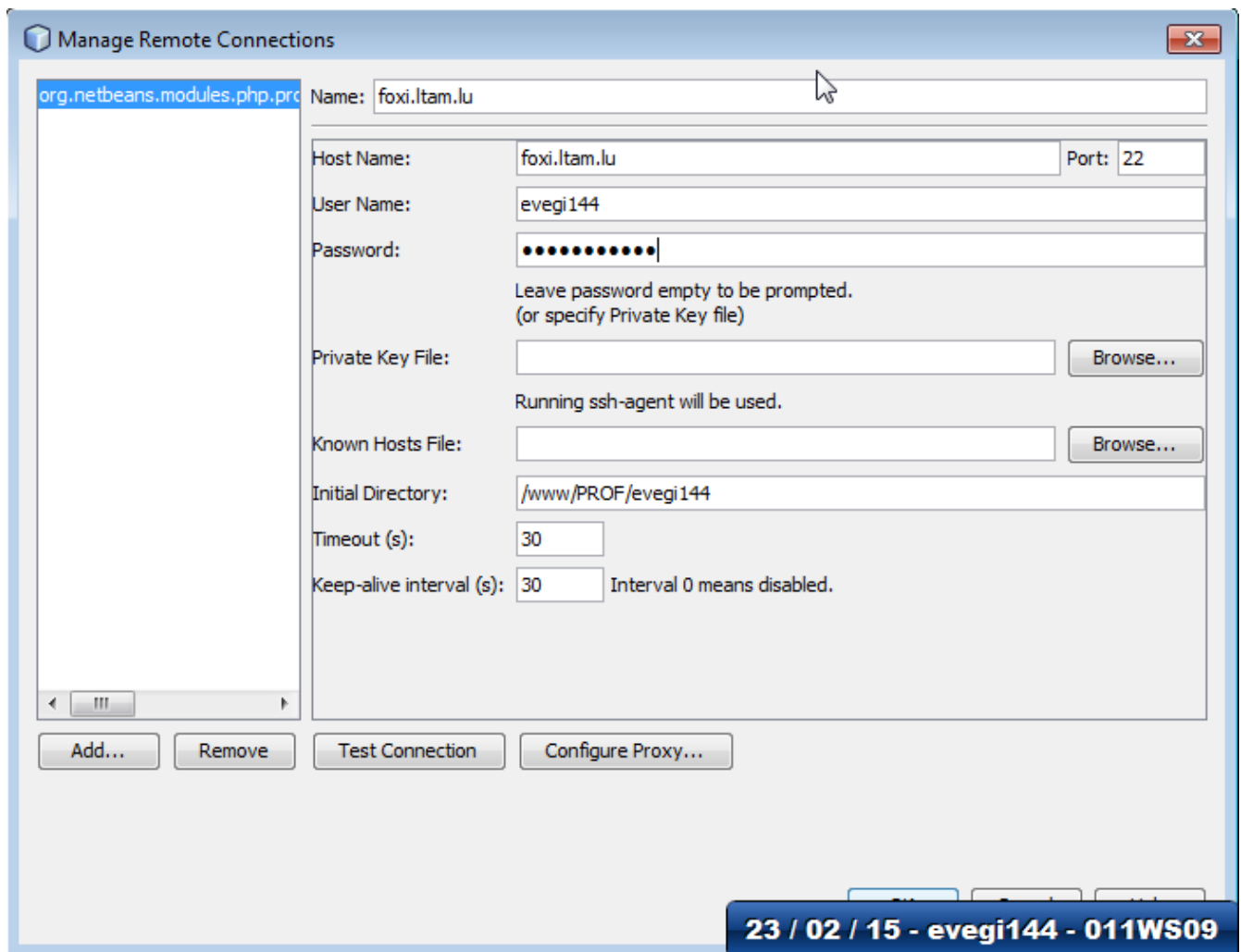
If you want to run your project on your local web server, select the corresponding option. We'll run our projects on Foxi, thus we specify **Remote Web Site (FTP, SFTP)**:



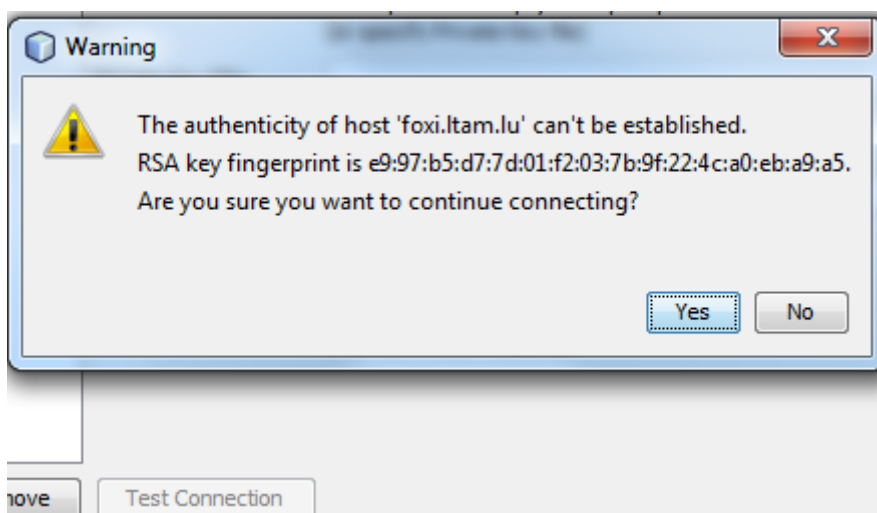
In order to be able to upload our files to Foxi, we need to define a remote connection, thus we need to click on Manage....



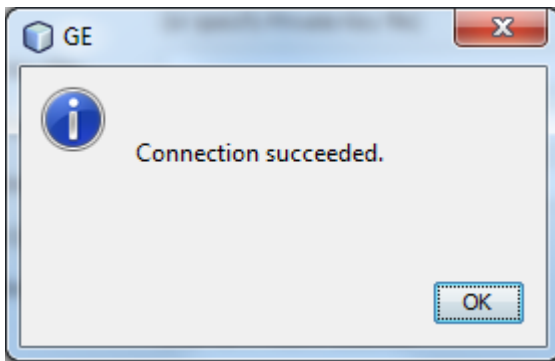
The host name is **foxi.ltam.lu**. The port number needs to be 22. Enter your IAM code as user name and your 11-digit matricule as password (cf. [FOXI\\_login\\_2017.pdf](#)) to learn how to change your password). The initial directory should be set to **/www/your class/your IAM code**:



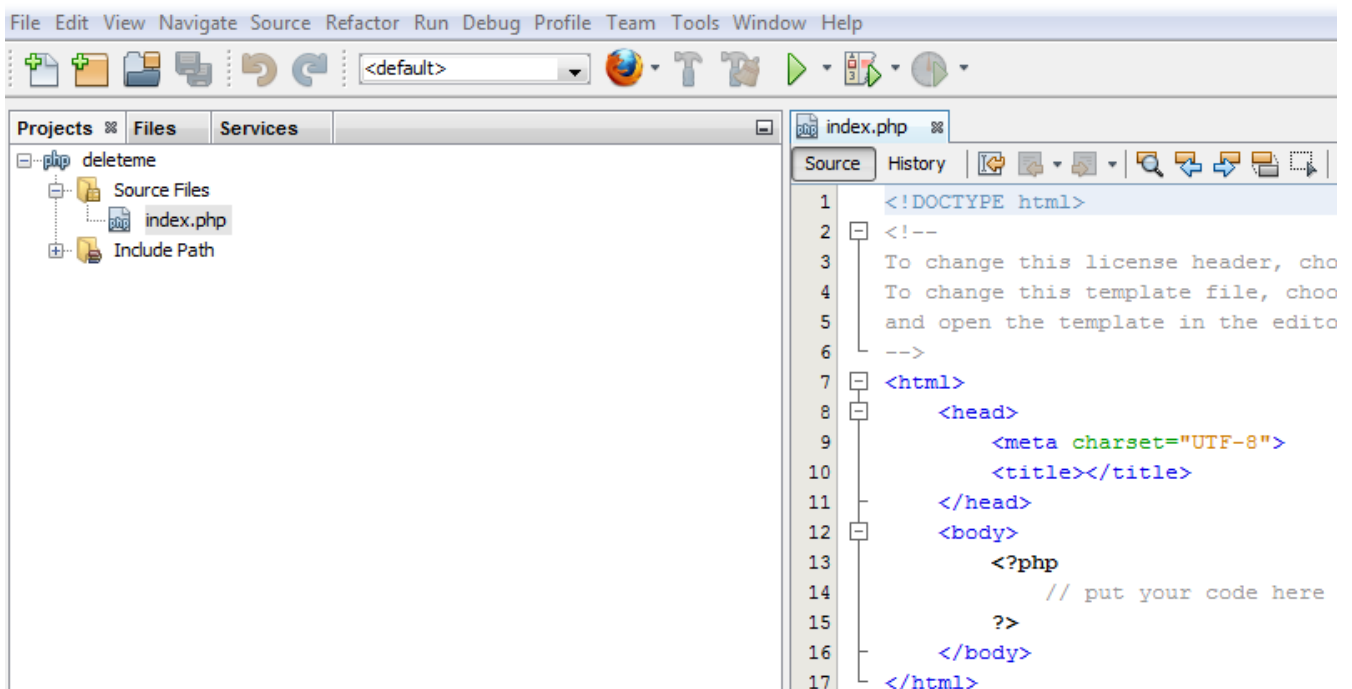
Now click the connection test button. You should get the following prompt, which you should confirm:



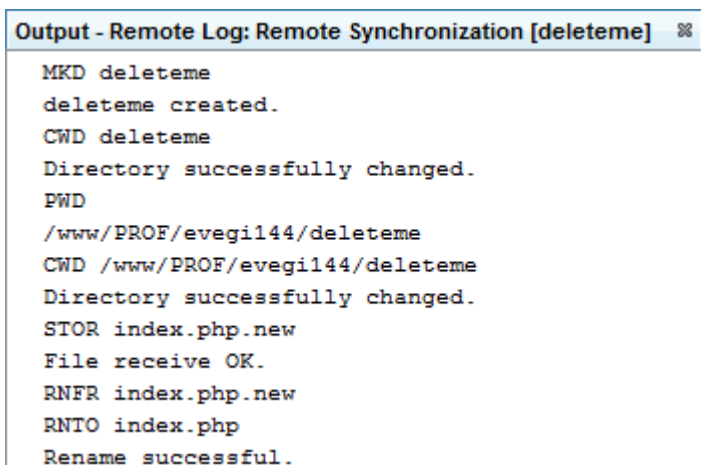
The following dialog should appear:



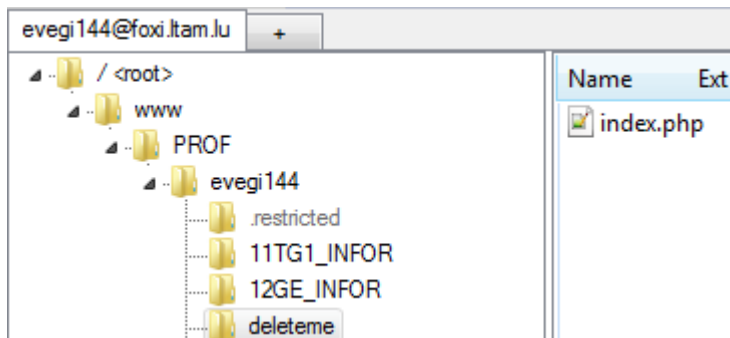
Now you can click on OK. We do not use any PHP framework, so you can click the Finish button in the final dialog. In the projects window (top left) you should now see your new PHP project. You can expand the project structure by clicking on the +-sign in front of it. Under **Source Files** you'll see your new PHP project:



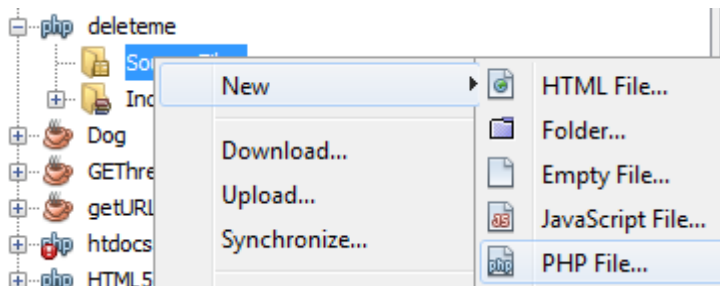
You can now start working on your project. When you save your changes, you'll have to confirm that you want to connect to Foxi. You'll then see confirmation in the output window that the file has been uploaded to Foxi:



We can verify this using our SFTP client (cf. [SFTP](#)):



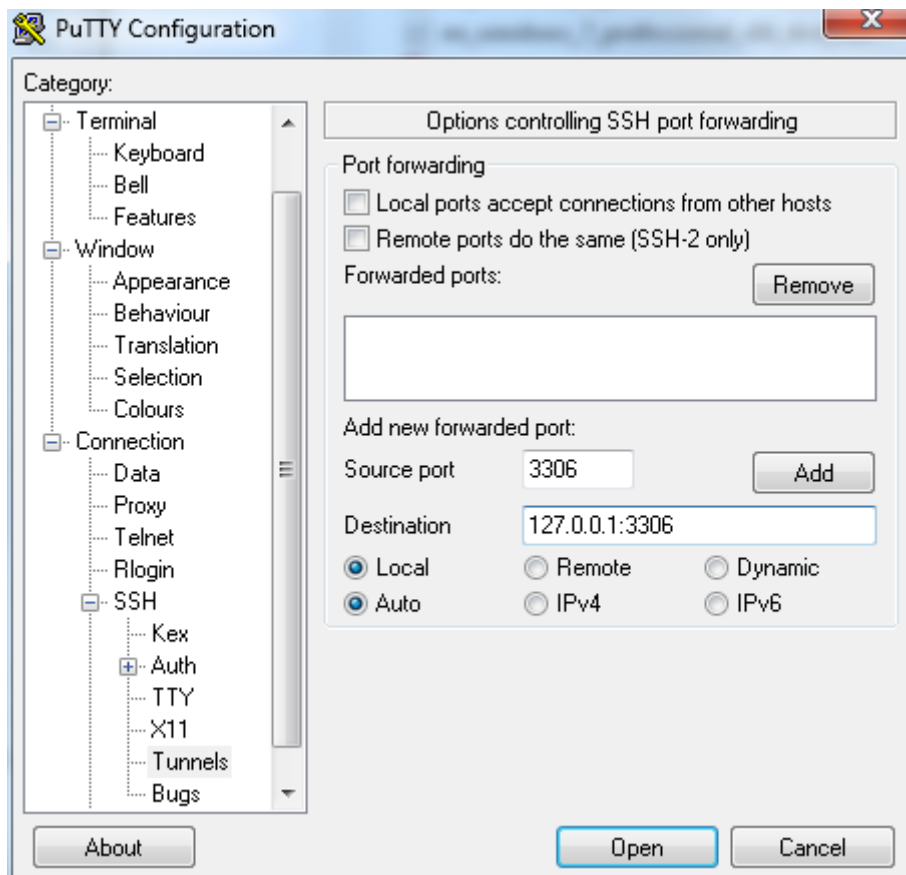
We can add a new file to our project by right clicking on **Source Files** and selecting **New > PHP File...**:



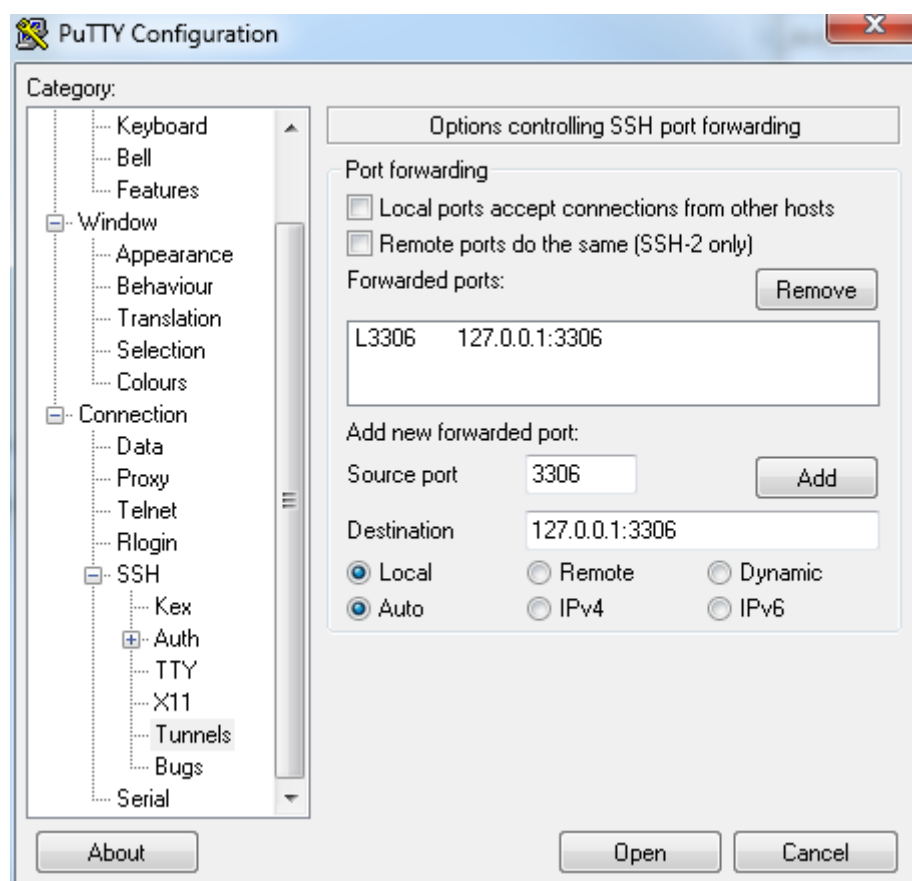
## Database connection setup

In order to connect to MySQL on Foxi we need to open a secure shell (SSH) tunnel. Given that NetBeans does not provide a built-in tunnel functionality, we need to use an external SSH client, for instance [Putty](#).

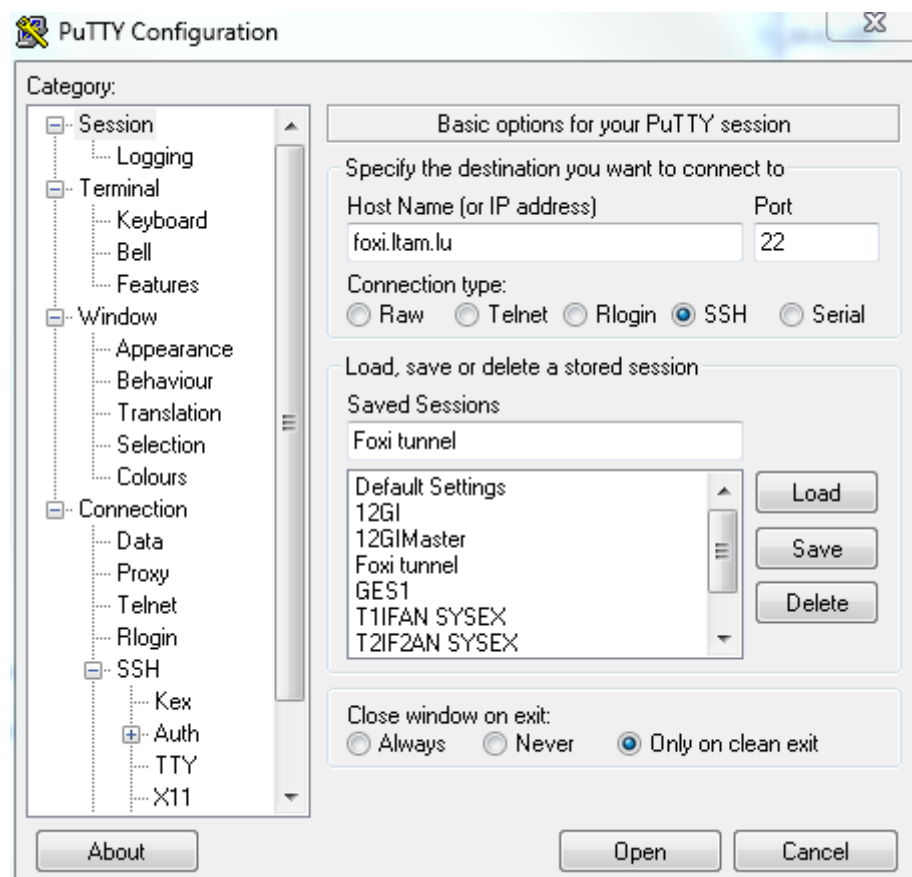
Enter the tunnel data:



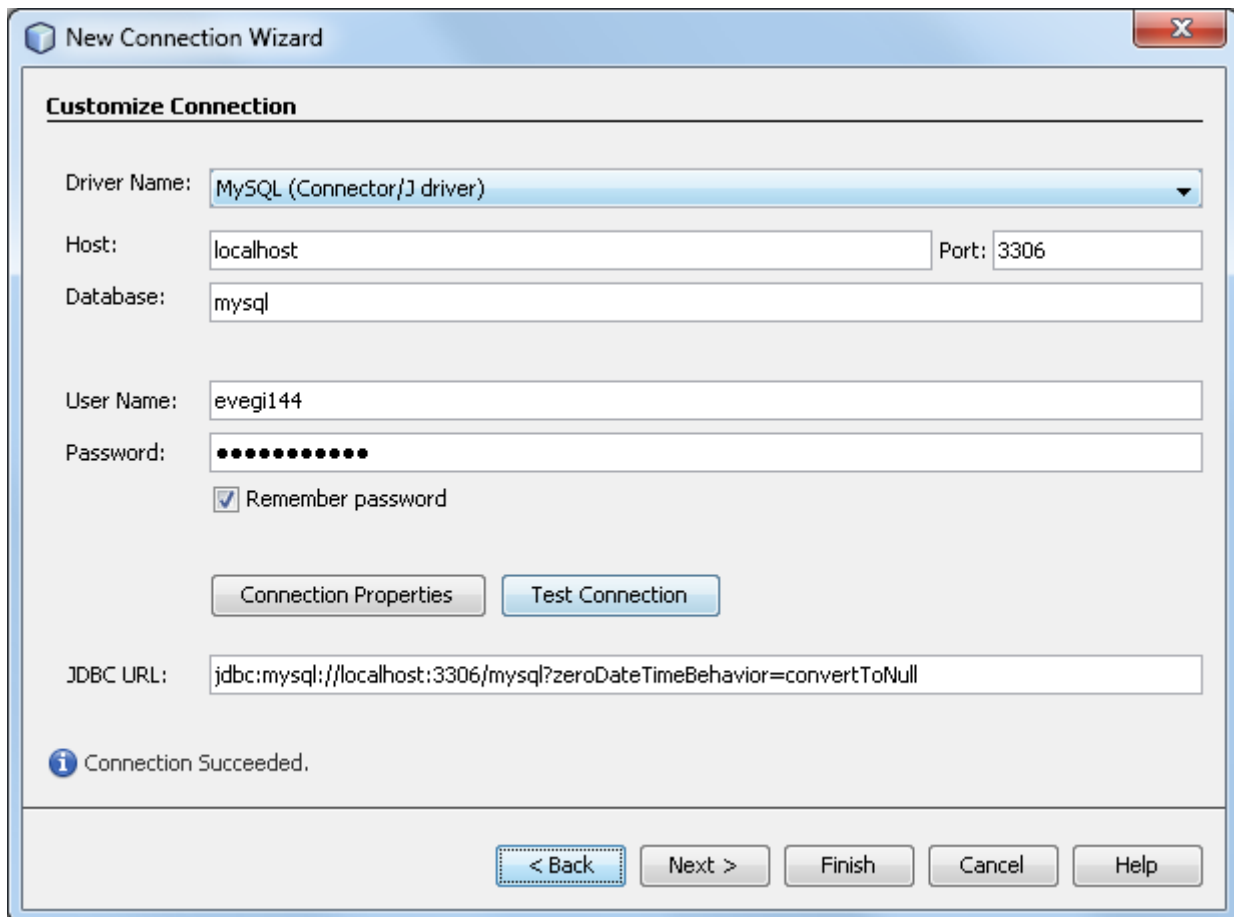
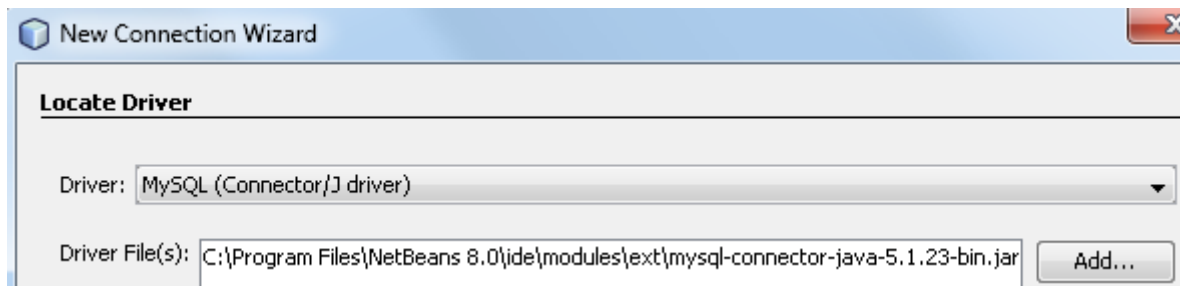
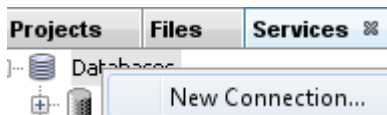
Then add the tunnel:



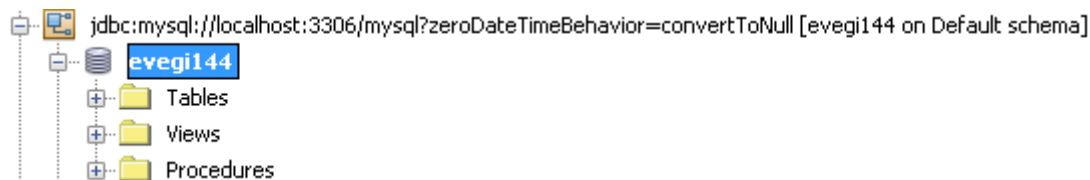
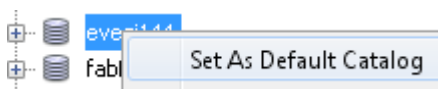
Save the session:



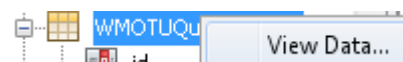
Create a new DB connection:

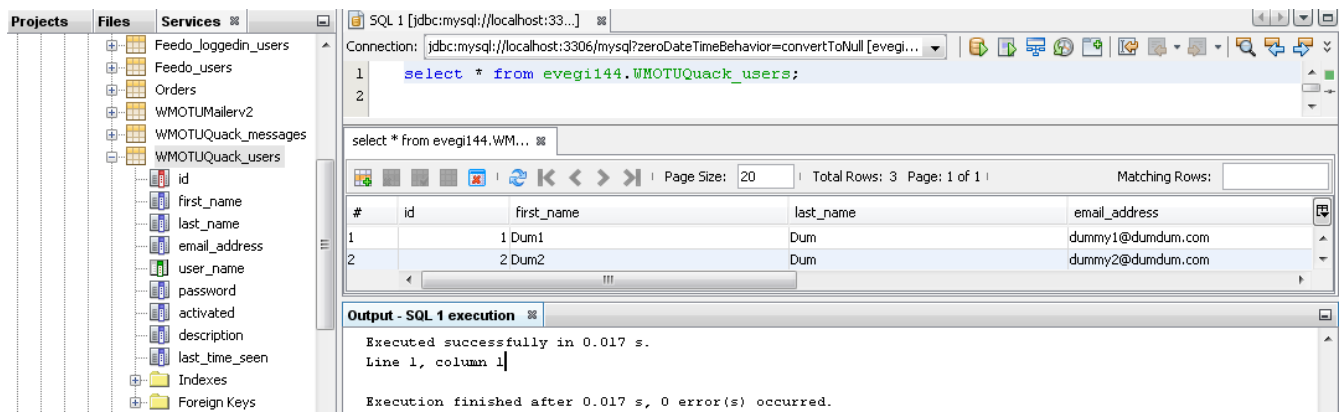


Set the default DB:

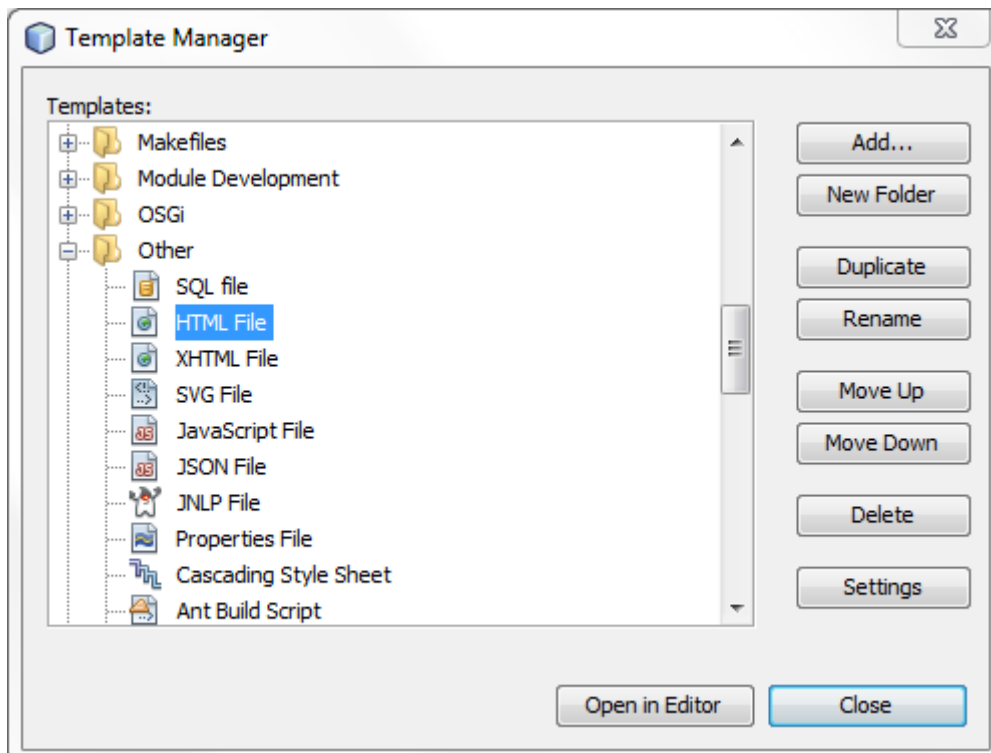
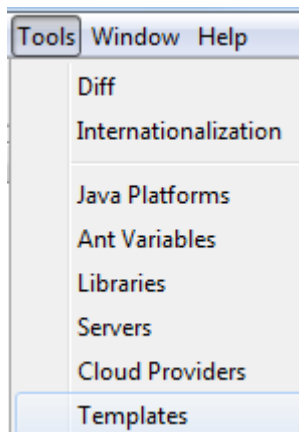


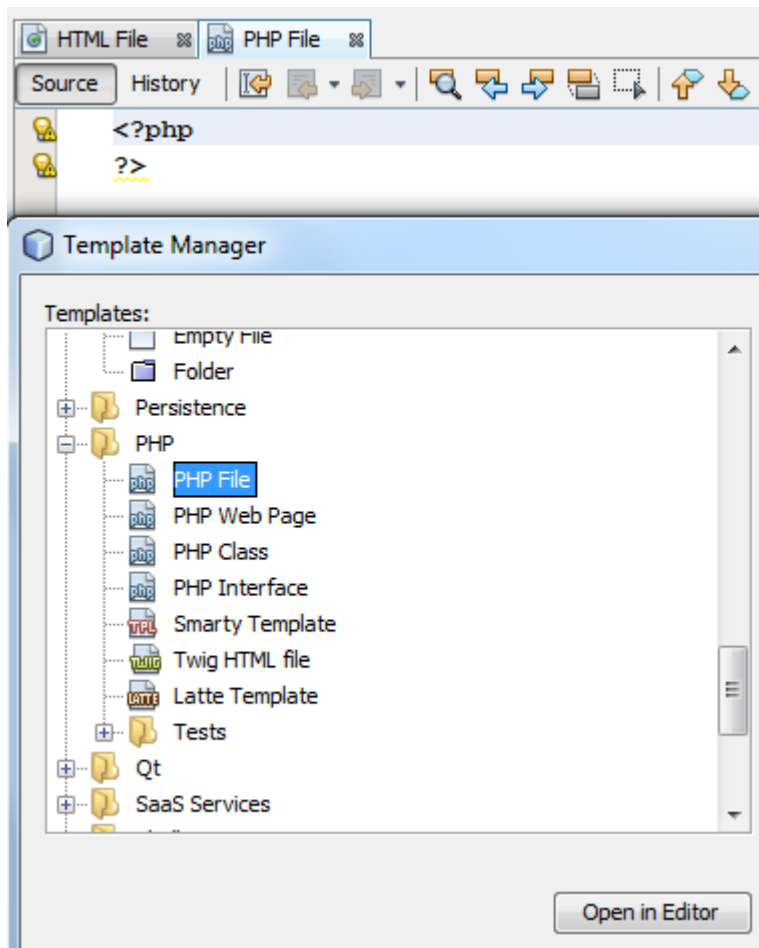
View table data:





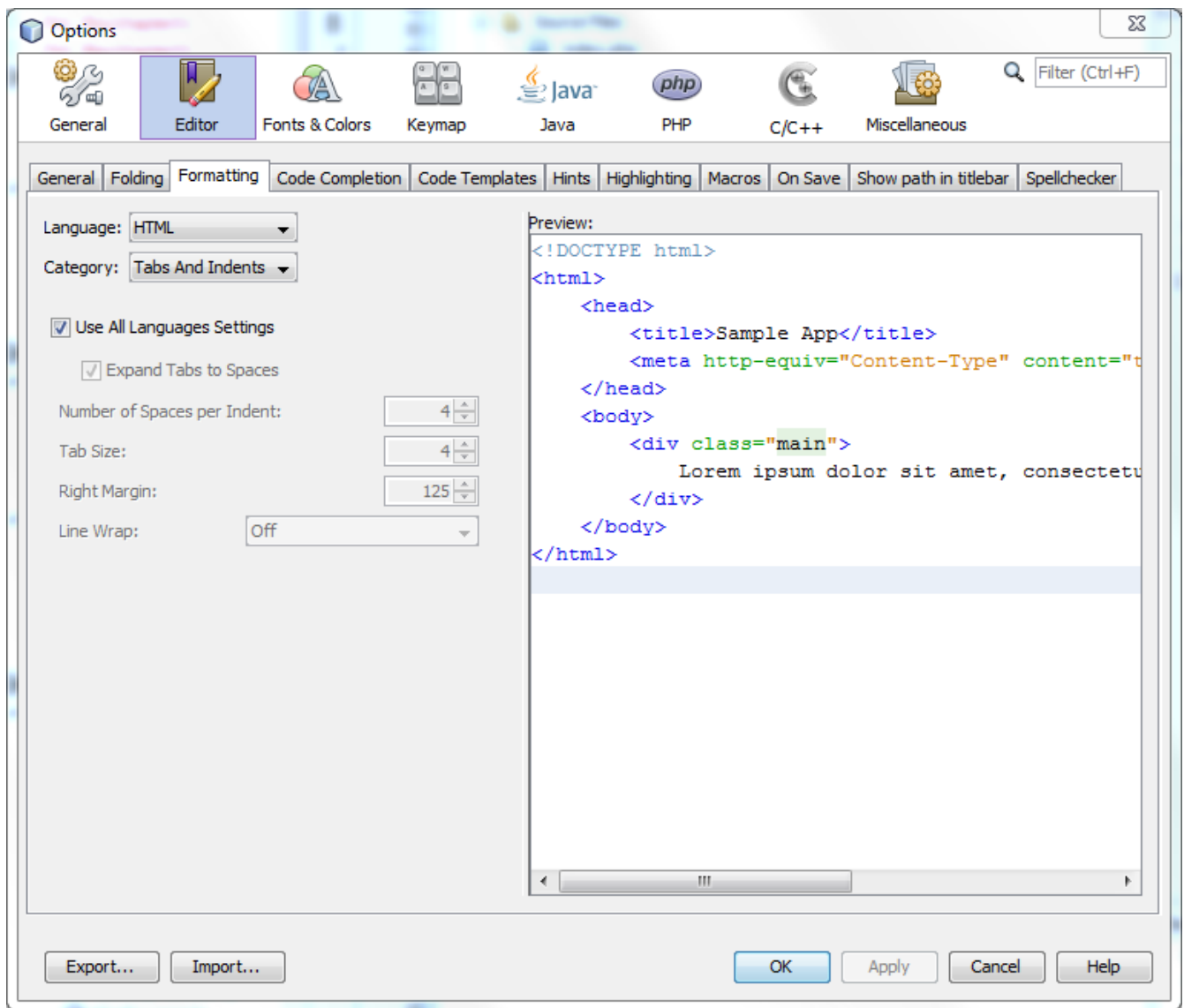
## Template adjustment





## Useful NetBeans shortcuts

Pressing **Alt** + **Shift** + **F** or selecting **Source** › **Format** will reformat the source code according to the settings in **Tools** › **Options**:



## 3.3. SSH

### 3.3.1. Clients

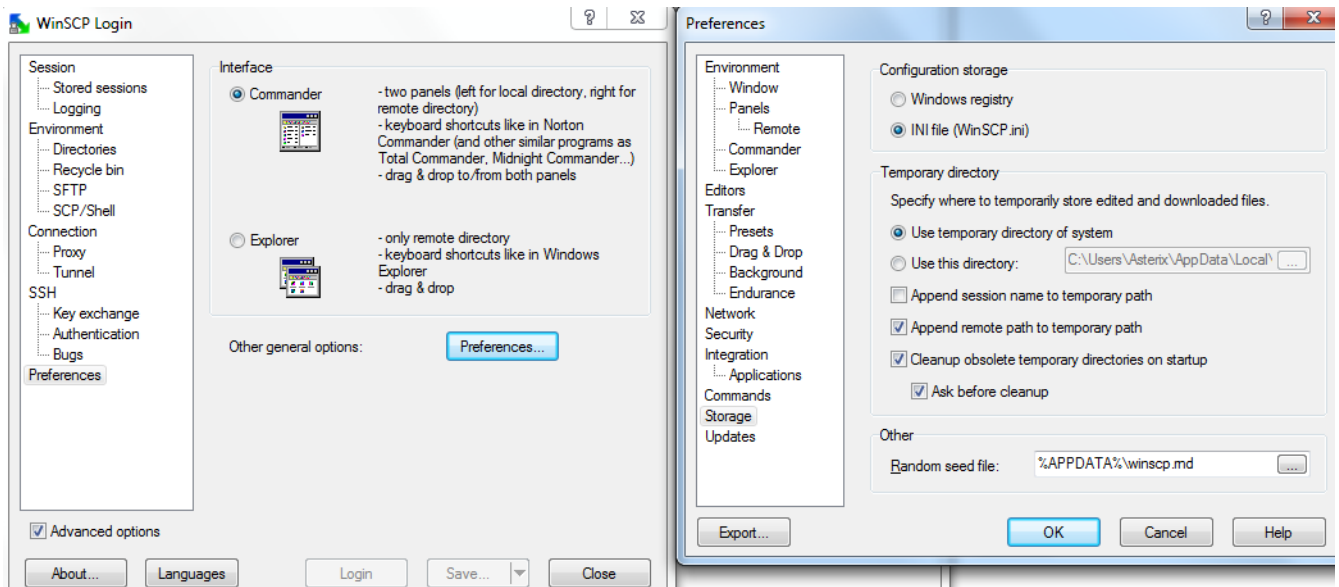
You can connect to your server using a secure shell (SSH) client such as [Putty](#), which is however rather rudimentary. Therefore you might prefer to use more convenient GUIs, most of which use Putty in the background:

1. [MTPuTTY](#)
2. [SmarTTY](#)
3. [WinSSHTerm](#)
4. [MobaXterm](#)

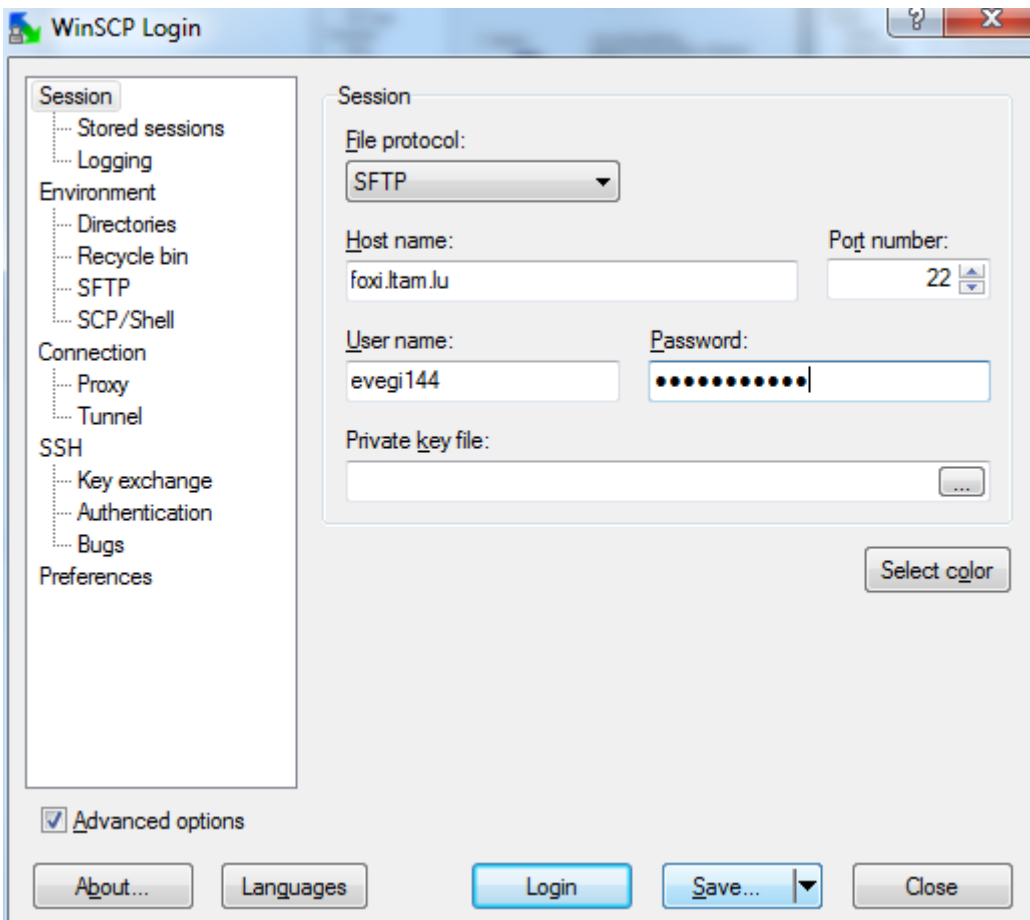
### 3.3.2. SFTP

SFTP is the secure, i.e. encrypted, version of FTP, the file transfer protocol. As its name suggests, it is used to transfer files from our development machine to the server and vice versa. There are a number of SFTP clients freely available, e.g. Filezilla. I prefer [WinSCP](#). Download the portable

executable and unzip it to a folder of your choice. Set the configuration storage option on the storage page in the preferences dialog to **INI file**. This will lead WinSCP to save its configuration in an ini file in the same folder where the program itself is located. If you have stored WinSCP on your USB stick, the configuration will also be stored there, so you won't have to reenter the server data every time you use the program:



Set up your server connection and click **save**. Click **login** to connect to the server and begin transferring your files:



## 3.4. Browsers

### 3.4.1. Firefox

If you incur problems loading a web page that works with other browsers, the source is often to be found with a specific add-on. To confirm this is the case, close Firefox and start it in safe mode, for instance by holding the Shift key pressed whilst launching Firefox (cf. [Safe Mode](#)).

To see all preferences that can be set, enter `about:config` in the address bar.

In case of a corrupt places.sqlite file: [developer.mozilla.org/en-US/docs/Mozilla/Tech/Places/places.sqlite\\_Database\\_Troubleshooting#How\\_to\\_\(try\\_to\)\\_recover\\_from\\_a\\_corrupt\\_places.sqlite](https://developer.mozilla.org/en-US/docs/Mozilla/Tech/Places/places.sqlite_Database_Troubleshooting#How_to_(try_to)_recover_from_a_corrupt_places.sqlite)

## 3.5. Documentation

### 3.5.1. Asciidoctor

#### Installation on Ubuntu

```
apt install asciidoctor
gem install asciidoctor-pdf --pre
gem install coderay
gem install rouge
apt install ruby-dev # required for pygments
gem install pygments.rb
```

[Asciidoctor](#)[Asciidoctorlive](#)[Asciidoctor PDF](#)[Cheatsheet](#)[Awesome Asciidoctor Notebook](#)[Syntax quick reference](#)[Expandable TOC](#)[Asciidoctor IntelliJ plugin](#)[Asciidoctor Diagram](#)[math.meta.stackexchange.com/questions/5020/mathjax-basic-tutorial-and-quick-reference](https://math.meta.stackexchange.com/questions/5020/mathjax-basic-tutorial-and-quick-reference)[asciimath.org](https://asciimath.org)[tinyapps.org/blog/misc/201509260700\\_asciidoctor.html](https://tinyapps.org/blog/misc/201509260700_asciidoctor.html)[dzone.com/articles/publishing-our-asciidoc-developer-guide-as-a-book](https://dzone.com/articles/publishing-our-asciidoc-developer-guide-as-a-book)[pandoc.org](https://pandoc.org)



File names may not contain spaces.

### Tips & tricks

- In order to have `` + `` displayed as intended, use `` {plus} `` (cf. [github.com/asciidoctor/asciidoctor/issues/759](https://github.com/asciidoctor/asciidoctor/issues/759)).
- To have images embedded and produce a stand-alone HTML document use the `-a data-uri` option with asciidoctor.
- To use AsciiDoc in tables, use `a` as explained in [discuss.asciidoctor.org/Lists-inside-of-table-cells-td3938.html](https://discuss.asciidoctor.org/Lists-inside-of-table-cells-td3938.html) and [mrhaki.blogspot.com/2014/11/awesome-asciidoctor-using-asciidoc-in.html](https://mrhaki.blogspot.com/2014/11/awesome-asciidoctor-using-asciidoc-in.html).
- To get hyperlinks rendered correctly, see [github.com/asciidoctor/asciidoctor/issues/625](https://github.com/asciidoctor/asciidoctor/issues/625).

### 3.5.2. Markdown

[en.support.wordpress.com/markdown-quick-reference](https://en.support.wordpress.com/markdown-quick-reference)

## 3.6. Content Management Systems

According to [Wikipedia](#):

A content management system (CMS) manages the creation and modification of digital content. These systems typically support multiple users in a collaborative environment, allowing to perform document management with different styles of governance and workflows. Usually the content is a website (or part of it) and the term commonly refers to web content management systems. Web content may include text and embedded graphics, photos, video, audio, maps, and program code (such as for applications) that displays content or interacts with the user. By their nature, CMSs support the separation of content and presentation.

Basically a CMS aims to enable the creation and maintenance of a web site without any knowledge of HTML, CSS, JavaScript etc. There are many CMS out there, but Wordpress seems to be the most popular by a wide margin. According to [W<sup>3</sup>Techs](#) as of September 20, 2019:

43.7% of the websites use none of the content management systems that we monitor. WordPress is used by 34.6% of all the websites, that is a content management system market share of 61.3%.

However, it is important to choose a CMS based on your specific requirements. As this [article](#) argues, there's no "best" CMS, each CMS has its strengths and weaknesses and is best suited for specific types of projects and environments.

Using an open-source CMS such as WordPress may not always be the best solution, see for instance [7 Solid Reasons not to Use WordPress](#).

### **3.6.1. Wordpress**

[Wordpress](#) provide detailed installation instructions. After installation you should study [first steps](#) and [wordpress.org/support/article/roles-and-capabilities](https://wordpress.org/support/article/roles-and-capabilities)

### **3.6.2. Magento**

[magento.com/products/community-edition](https://magento.com/products/community-edition)

# Chapter 4. Client side programming

## 4.1. HTML5

HTML stands for Hyper Text Markup Language. HTML5 is the latest version of this markup language for describing web documents or pages. It consists of a series of *tags*. An element or tag is like a command or instruction that tells the browser about the structure and meaning (also called semantics) of the content of a specific part of our web page.



[Here](#) is a great listing and explanation of all tags with examples. The official standard can be found [here](#) and the complete set of HTML elements [here](#). [Cheat sheets](#) can be very helpful.

### 4.1.1. Basic structure

<https://students.btsi.lu/evegi144/WAD/HTML5/skeleton1.html>

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>HTML5 Skeleton</title>
    <meta charset=UTF-8>
    <link href=style1.css rel=stylesheet>
    <script src=script1.js></script>
  </head>
  <body>

  </body>
</html>
```

<students.btsi.lu/evegi144/WAD/HTML5/style1.css>

```
body {
  background-color: lightseagreen;
}
```

<students.btsi.lu/evegi144/WAD/HTML5/script1.js>

```
window.alert("Our first JavaScript has been loaded and executed!");
```

The first line of an HTML5 document should tell the browser how to process the document by specifying the Document Type Definition (DTD).

After the DTD comes the `<html>` tag, which specifies the language using the `lang` attribute. See [tag list](#) for a list of available language codes. The `<html>` tag encompasses the whole HTML document

consisting of a `<head>` and a `<body>` part.

## `<head>`

In the head part we specify the title and the character encoding, which for our purposes will be **UTF-8**. UTF-8 has the advantage that it handles special characters, e.g. ö and é, correctly. To learn more about character encodings, see [www.w3.org/International/tutorials/tutorial-char-enc](http://www.w3.org/International/tutorials/tutorial-char-enc). Then we include our external CSS and JavaScript files (more on those in the following chapters).

From [www.w3schools.com/tags/tag\\_head.asp](http://www.w3schools.com/tags/tag_head.asp):

The following elements can go inside the `<head>` element:

- `<title>` (**this element is required in the head section**)
- `<style>`
- `<base>`
- `<link>`
- `<meta>`
- `<script>`
- `<noscript>`

## `<body>`

The body part contains the actual page content.

### Opening and closing tags

For most, but not all, of the HTML5 tags, there is an opening and a closing tag, as in `<body></body>`. There are a few standalone tags, such as `<hr>` to display a horizontal line.

### Tabs, new lines and spaces

**Browsers ignore tabs, new lines and most spaces.** For instance, the following two HTML documents produce exactly the same output except for the words "a well", which have been replaced with "an ill":

[students.btsi.lu/evegi144/WAD/HTML5/spacedemo1.html](http://students.btsi.lu/evegi144/WAD/HTML5/spacedemo1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Example of a well formatted HTML document</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      This is a well formatted HTML document.
    </main>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/HTML5/spacedemo2.html](https://students.btsi.lu/evegi144/WAD/HTML5/spacedemo2.html)

```
<!DOCTYPE html>
<html

                                lang=en>

<head><title>Example

  of      an ill      formatted
HTML document</title>
  <meta charset=UTF-8></head><body><main>This is   an
ill formatted HTML document.</main></body></html>
```

## Comments

In order to help others (and ourselves) understand our HTML documents, it is a good idea to include comments where appropriate. Comments are embedded between `<!--` and `-->`. A comment can span several lines and is not displayed by the browser.

Example: [students.btsi.lu/evegi144/WAD/HTML5/comments1.html](https://students.btsi.lu/evegi144/WAD/HTML5/comments1.html)


```

<!DOCTYPE html> <!-- The document type is compulsory. -->
<html lang=en> <!-- Don't forget to specify the language. -->
  <head>
    <!--
      This is a very simple illustration of comment usage in HTML5.
      You do not have to use comments.
      Use them where it makes sense.
    -->
    <title>This is an example of comment usage in HTML5</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <!-- The main part is currently empty. If you have a good idea
        on how to fill it, let me know! :D
      -->
    </main>
  </body>
</html>

```

### 4.1.2. Validation



In order to be sure that our HTML5 document complies with the official standard and should thus run according to our plans in all compliant browsers, we need to validate our HTML5 and CSS3 files either using the official validators at [validator.w3.org](http://validator.w3.org) and [jigsaw.w3.org/css-validator](http://jigsaw.w3.org/css-validator) or using the Firefox extension [users.skynet.be/mgueury/mozilla/index.html](http://users.skynet.be/mgueury/mozilla/index.html). The latter will install a validation button in the Firefox add-on bar. A simple double click will display the source code of the current page and run it automatically through the official HTML5 validator. With a right click on the button we launch the CSS3 validator via the **Advanced** menu. Here are the outputs of the validators for the solution of the next exercise:

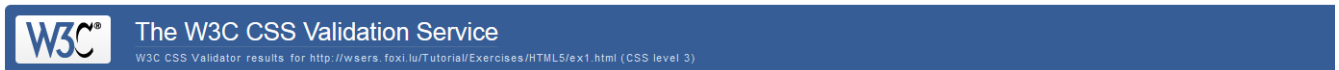
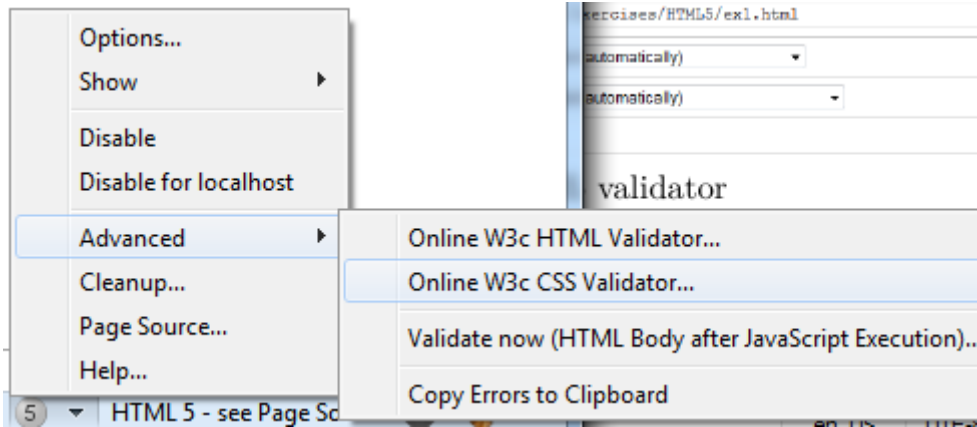

**Markup Validation Service**  
Check the markup (HTML, XHTML, ...) of Web documents

[Jump To:](#)
[Notes and Potential Issues](#)
[Congratulations · Icons](#)

This document was successfully checked as HTML5!

<b>Result:</b>	Passed, 1 warning(s)	
<b>Address :</b>	<input type="text" value="http://users.foxi.lu/Tutorial/Exercises/HTML5/ex1.html"/>	
<b>Encoding :</b>	utf-8	(detect automatically) ▼
<b>Doctype :</b>	HTML5	(detect automatically) ▼
<b>Root Element:</b>	html	

Type	Line	Column	HTML errors and warnings
 Result	0 errors / 2 warnings		
 Info	W3c Online Validation		



W3C CSS Validator results for <http://wsers.foxi.lu/Tutorial/Exercises/HTML5/ex1.html> (CSS level 3)

### Congratulations! No Error Found.

This document validates as [CSS level 3](#) !

To show your readers that you've taken the care to create an interoperable Web page, you may display this icon on any page that validates. Here is the XHTML you could use to add this icon to your Web page:



The warning messages can be safely ignored. They just tell us that the validator is still experimental. Given that the HTML5 standard is not expected to be finalized for many years, this is unlikely to change any time soon.

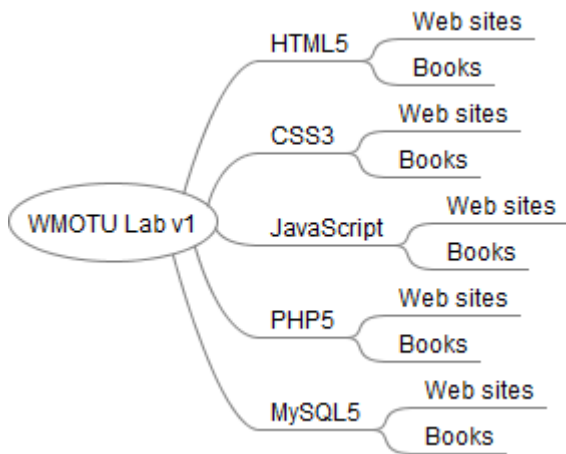
## 4.1.3. Planning

In order to produce a top notch web site, we need to plan our work carefully.

### Brain storming

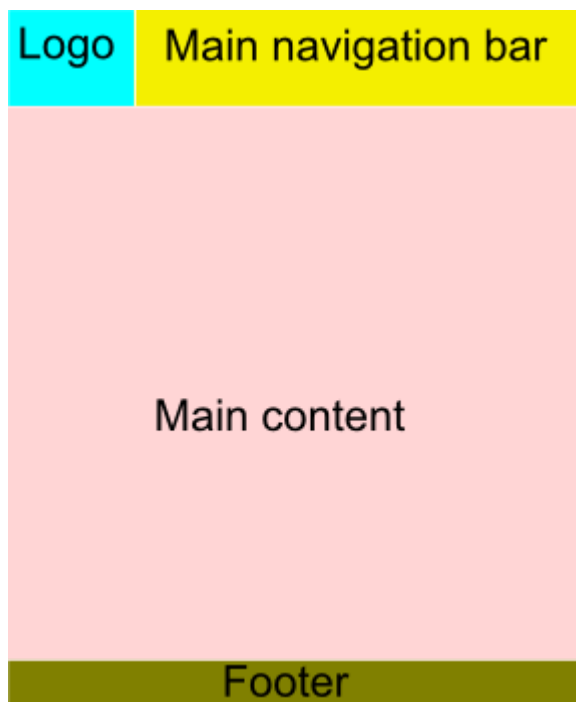
First we need to think about the purpose of our web site. What are the big concepts and ideas that will drive our content? A useful tool in this respect is a mind map. We'll be using the open source FreeMind software, which is available from [freemind.sourceforge.net/wiki/index.php/Main\\_Page](http://freemind.sourceforge.net/wiki/index.php/Main_Page).

As an example, here's the mind map for WMOTU Lab v1:



## Blueprint

Now that we have clarified the big picture content of our site, it's time to sketch out the rough and basic structure. For this purpose we'll use an open source vector drawing software named [Inkscape](#).



## Requirements specification

The standard professional approach to project planning is to produce a requirements specification. Such a document specifies the project requirements, including:

1. Functionality
2. Prototype/model
3. Logical site structure
4. Physical site structure
5. Time plan
6. Development environment and technologies

Here is a minimalist example for the WMOTU Address Book app developed in [WMOTU Address Book](#):

### Functionality

The app serves as an electronic address book. New users need to sign up by providing a login name and password. After logging in, the user enters the main page, where he can logout and view a listing of all his addresses. He can delete or edit each address as well as add a new one. All addresses are stored in a MySQL database on the server.

### Prototype/model

[foxi.ltam.lu/PROF/evegi144/T2IF2\\_WSERS/WAD/HTML5/WMOTUAddressBookPrototype.html](http://foxi.ltam.lu/PROF/evegi144/T2IF2_WSERS/WAD/HTML5/WMOTUAddressBookPrototype.html)

### Logical site structure

[webstyleguide.com/wsg3/3-information-architecture/3-site-structure.html](http://webstyleguide.com/wsg3/3-information-architecture/3-site-structure.html)



### Physical site structure

WMOTUAddressBook  
├── protected

### Time plan

The final product will be delivered electronically on 24.6.14.

### Development environment and technologies

Development will be done mainly with PhpStorm. Main technologies used will be HTML5, CSS3, PHP5 and MySQL5.

#### 4.1.4. `<br>`

As we have seen, new lines in our source code are converted to a single space by the browser. To split our text into different lines, we use the `<br>` tag, which inserts a line break. As already mentioned in this chapter, for most, but not all, HTML5 tags there is an opening and a closing tag. `<br>` is one of the exceptions. It is a so called empty tag, meaning it has no closing tag. This tag should not be used to separate paragraphs. For the latter purpose we use the `<p>` tag, cf. `<p>`.

Example:

This is the first line. This should be the second one.  
This is the second one.

[students.btsi.lu/evegi144/WAD/HTML5/br1.html](http://students.btsi.lu/evegi144/WAD/HTML5/br1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Break row example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      This is the first line.
      This should be the second one.<br>
      This is the second one.
    </main>
  </body>
</html>

```

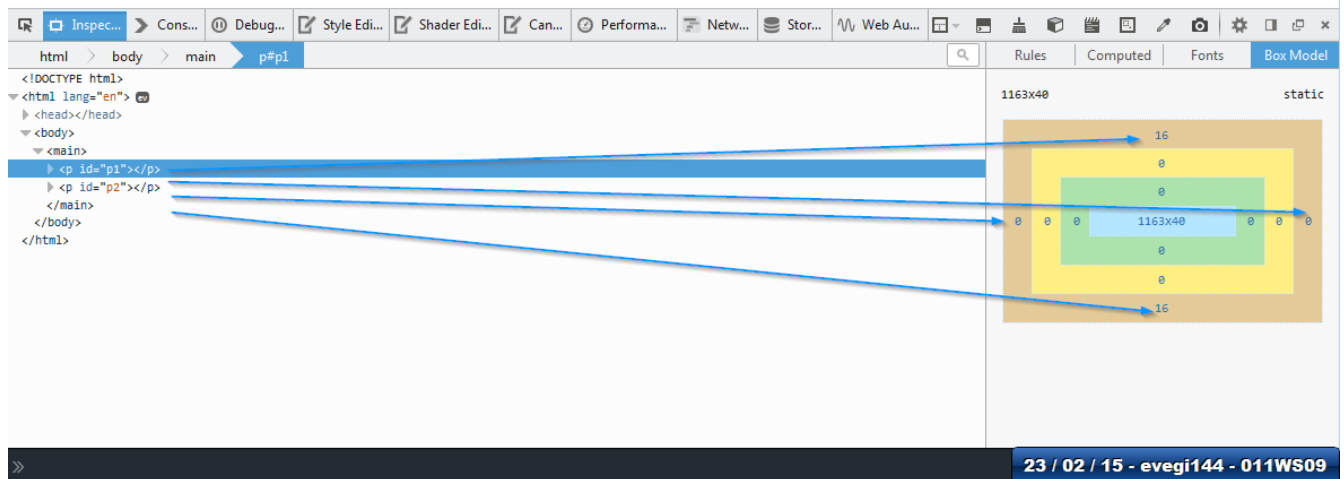
#### 4.1.5. <p>

This tag is used to mark up a paragraph. The browser automatically adds margin above and below each paragraph (see section [Block vs inline elements](#)).

Example:

This is the first paragraph. Note that the text always occupies the full width of the browser window. If you change the width of your browser window, the number of lines that your paragraph occupies changes too.

This is the second paragraph. Use the Firefox console to inspect the margins used by your browser.



[students.btsi.lu/evegi144/WAD/HTML5/p1.html](http://students.btsi.lu/evegi144/WAD/HTML5/p1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Paragraph example</title>
    <meta charset=UTF-8>
    <style>
      #p1 {
        background-color: green;
      }

      #p2 {
        background-color: gold;
      }
    </style>
  </head>
  <body>
    <main>
      <p id=p1>
        This is the first paragraph. Note that the text always occupies the full
        width of the browser window. If you change the width of your browser window,
        the number of lines that your paragraph occupies changes too.
      </p>
      <p id=p2>
        This is the second paragraph. Use the Firefox console to inspect the margins
        used by your browser.
      </p>
    </main>
  </body>
</html>

```

Resize your browser and observe the behavior of your paragraphs. Note that we've given each paragraph an id attribute. This allows us to style each paragraph's background color individually using CSS. More on this in [CSS3](#).

#### 4.1.6. Phrase tags

Phrase tags are used to convey special meaning to text:

Name	Description
<em>	emphasized text
<strong>	important text
<dfn>	definition term
<code>	computer code
<samp>	sample output from a computer program
<kbd>	keyboard input
<var>	variable

Here is a simple application:

I'd like to emphasize *the following*. This is particularly **important**.

*HTML5* is the definition of greatness!

The secret of the universe looks like this: `if (sunshine === true) window.alert("Smile!");`  
This will hopefully produce this output: *Smile!*

Enter `WMOTU` as your user name. It will be stored in *userName*

[students.btsi.lu/evegi144/WAD/HTML5/phrasetags1.html](https://students.btsi.lu/evegi144/WAD/HTML5/phrasetags1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Phrase tags example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <p>I'd like to emphasize <em>the following</em>. This is particularly
      <strong>important</strong>.</p>
      <p><dfn>HTML5</dfn> is the definition of greatness!</p>
      <p>The secret of the universe looks like this:
        <code>if (sunshine === true) window.alert("Smile!");</code>
        This will hopefully produce this output: <samp>Smile!</samp></p>
      <p>Enter <kbd>WMOTU</kbd> as your user name. It will be stored in
        <var>userName</var></p>
    </main>
  </body>
</html>
```

#### 4.1.7. HTML entities

Some characters, such as `<` or `>`, are reserved in HTML. We thus cannot use them directly in our text as the browser would try to interpret them as part of a tag.

To get around this problem, we use character entities. A character entity has the form `&entity_name` or `&#entity_number`. The following table lists the reserved characters and their corresponding entities (see [dev.w3.org/html5/html-author/charref](https://dev.w3.org/html5/html-author/charref) for the complete list and [digitalmediamminute.com/reference/entity](https://digitalmediamminute.com/reference/entity) for the Unicode codes):

Character	Entity number	Entity name	Description
"	<code>&amp;#34;</code>	<code>&amp;quot;</code>	quotation mark
'	<code>&amp;#39;</code>	<code>&amp;apos;</code>	apostrophe
&	<code>&amp;#38;</code>	<code>&amp;amp;</code>	ampersand

Character	Entity number	Entity name	Description
<	&#60;	&lt;	less than
>	&#62;	&gt;	greater than

Application example:

[students.btsi.lu/evegi144/WAD/HTML5/entities1.html](https://students.btsi.lu/evegi144/WAD/HTML5/entities1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Entities usage example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      The HTML expert said &#34;Use HTML entities to display special characters in
      HTML&quot;. As you know, HTML tags start with an &lt; and close with a &gt;, as
in
      <code>&lt;a&gt;</code>.
    </main>
  </body>
</html>
```

#### 4.1.8. <header>

The <header> tag specifies a header for a document or section. It should be used for introductory content or navigation elements. You can have several of these in one document, but they cannot be placed within a <footer>, <address> or another <header> element.

<h1> ... <h6>

These tags specify headings at different levels:

[students.btsi.lu/evegi144/WAD/HTML5/heading1.html](https://students.btsi.lu/evegi144/WAD/HTML5/heading1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Heading Example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <header>
      <h1>Heading level 1</h1>
      <h2>Heading level 2</h2>
      <h3>Heading level 3</h3>
      <h4>Heading level 4</h4>
      <h5>Heading level 5</h5>
      <h6>Heading level 6</h6>
    </header>
  </body>
</html>
```

**Heading level 1**

**Heading level 2**

**Heading level 3**

**Heading level 4**

**Heading level 5**

**Heading level 6**

#### 4.1.9. Lists

We can choose between unordered and ordered lists. In each case, every list item is enclosed in `<li></li>` tags.



A paragraph may not contain lists.

`<ul>`

[students.btsi.lu/evegi144/WAD/HTML5/ul1.html](https://students.btsi.lu/evegi144/WAD/HTML5/ul1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Unordered List Example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <header>
        <h1>Today's shopping list</h1>
      </header>
      <ul>
        <li>Meat</li>
        <li>Cheese</li>
        <li>Vegetables</li>
        <li>Water</li>
        <li>Bin bags</li>
      </ul>
    </main>
  </body>
</html>

```

## Today's shopping list

- Meat
- Cheese
- Vegetables
- Water
- Bin bags

<ol>

This element supports the following particular attributes:

Name	Value	Description
reversed		descending list order
start	number	start value
type	1, A, a, I, i	list marker

## Today's shopping lists

- |               |               |               |
|---------------|---------------|---------------|
| 1. Meat       | 5. Meat       | C. Meat       |
| 2. Cheese     | 4. Cheese     | D. Cheese     |
| 3. Vegetables | 3. Vegetables | E. Vegetables |
| 4. Water      | 2. Water      | F. Water      |
| 5. Bin bags   | 1. Bin bags   | G. Bin bags   |

[students.btsi.lu/evegi144/WAD/HTML5/ol1.html](https://students.btsi.lu/evegi144/WAD/HTML5/ol1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Ordered List Example</title>
    <meta charset=UTF-8>
    <style>
      #l3 {
        list-style-type: upper-roman;
      }
    </style>
  </head>
  <body>
    <main>
      <header>
        <h1>Today's shopping lists</h1>
      </header>
      <ol style="float: left">
        <li>Meat</li>
        <li>Cheese</li>
        <li>Vegetables</li>
        <li>Water</li>
        <li>Bin bags</li>
      </ol>
      <ol style="float: left" reversed>
        <li>Meat</li>
        <li>Cheese</li>
        <li>Vegetables</li>
        <li>Water</li>
        <li>Bin bags</li>
      </ol>
      <ol id=l3 style="float: left" start=3 type=A>
        <li>Meat</li>
        <li>Cheese</li>
        <li>Vegetables</li>
        <li>Water</li>
        <li>Bin bags</li>
      </ol>
    </main>
  </body>
</html>

```

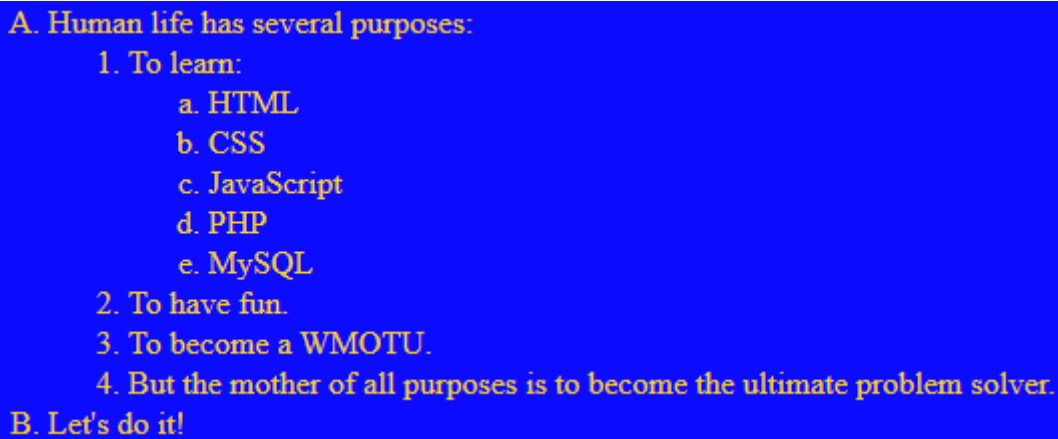
The **style** attribute value of **float: left** means that the list is floated left. As a result, the following element is placed right to the list instead of underneath. Remove the three style attributes and compare the result.

## Nested lists

We can nest lists. This means we can have a list inside a list inside a list inside a list ... as many times as we want. The only thing we need to watch is the correct nesting, i.e. we need to close the

last opened list tag before we close the second last etc.

Example:



A. Human life has several purposes:

1. To learn:
  - a. HTML
  - b. CSS
  - c. JavaScript
  - d. PHP
  - e. MySQL
2. To have fun.
3. To become a WMOTU.
4. But the mother of all purposes is to become the ultimate problem solver.

B. Let's do it!

[students.btsi.lu/evegi144/WAD/HTML5/nestedlists1.html](https://students.btsi.lu/evegi144/WAD/HTML5/nestedlists1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Nested list example</title>
    <meta charset=UTF-8>
    <style>
      body {
        background-color: black;
        color:          gold;
      }

      main {
        position: relative;
        width:    550px;
        height:   230px;
        animation: move 30s linear 0s infinite alternate;
      }

      /* cf. http://www.w3schools.com/css/tryit.asp?filename=trycss3_animation5 */
      @keyframes move {
        0% {
          background-color: blue;
          left:            0px;
          top:             0px;
        }
        25% {
          background-color: white;
          left:            550px;
          top:             0px;
        }
        50% {
          background-color: red;
```

```

        left:          550px;
        top:           230px;
    }
    75% {
        background-color: darkgray;
        left:           0px;
        top:            230px;
    }
    100% {
        background-color: red;
        left:           0px;
        top:            0px;
    }
}
</style>
</head>
<body>
    <main>
        <ul type=A>
            <li>Human life has several purposes:
                <ol>
                    <li>To learn:
                        <ol type=a>
                            <li>HTML</li>
                            <li>CSS</li>
                            <li>JavaScript</li>
                            <li>PHP</li>
                            <li>MySQL</li>
                        </ol>
                    </li>
                    <li>To have fun.</li>
                    <li>To become a WMOTU.</li>
                    <li>But the mother of all purposes is to become the ultimate problem
                        solver.
                    </li>
                </ol>
            </li>
            <li>Let's do it!</li>
        </ul>
    </main>
</body>
</html>

```

Like the animation? We'll do plenty of these in section [Transformation and animation](#).

**<dl>**

From [www.w3.org/TR/html51/semantics.html#the-dl-element](http://www.w3.org/TR/html51/semantics.html#the-dl-element):

The **dl** element represents an association list consisting of zero or more name-value groups (a description list). A name-value group consists of one or more names (**dt** elements) followed by one or more values (**dd** elements), ignoring any nodes other than **dt** and **dd** elements. Within a single **dl** element, there should not be more than one **dt** element for each name.

#### HTML5

HTML5 is a markup language used for structuring and presenting content for the World Wide Web and a core technology of the Internet (cf. <http://en.wikipedia.org/wiki/HTML5>).

#### CSS3

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation semantics (the look and formatting) of a document written in a markup language. Its most common application is to style web pages written in HTML and XHTML, but the language can also be applied to any kind of XML document, including plain XML, SVG and XUL (cf. <https://en.wikipedia.org/wiki/CSS>).

#### JavaScript

JavaScript (JS) is an interpreted computer programming language. As part of web browsers, implementations allow client-side scripts to interact with the user, control the browser, communicate asynchronously, and alter the document content that is displayed. It has also become common in server-side programming, game development and the creation of desktop applications (cf. <https://en.wikipedia.org/wiki/JavaScript>).

#### PHP5

PHP is a server-side scripting language designed for web development but also used as a general-purpose programming language. PHP is now installed on more than 244 million websites and 2.1 million web servers (cf. <https://en.wikipedia.org/wiki/PHP5>).

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Description List Example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <dl>
        <dt>HTML5</dt>
        <dd>HTML5 is a markup language used for structuring and presenting content for
the
        World Wide Web and a core technology of the Internet (cf.
        <a
href=http://en.wikipedia.org/wiki/HTML5>http://en.wikipedia.org/wiki/HTML5</a>).
        </dd>
        <dt>CSS3</dt>
        <dd>Cascading Style Sheets (CSS) is a style sheet language used for describing
the
        presentation semantics (the look and formatting) of a document written in a
markup language. Its most common application is to style web pages written
in
        HTML and XHTML, but the language can also be applied to any kind of XML
document, including plain XML, SVG and XUL (cf. <a
        href=https://en.wikipedia.org/wiki/CSS>
        https://en.wikipedia.org/wiki/CSS</a>).
        </dd>
        <dt>JavaScript</dt>
        <dd>JavaScript (JS) is an interpreted computer programming language. As part
of
        web browsers, implementations allow client-side scripts to interact with the
user, control the browser, communicate asynchronously, and alter the
document
        content that is displayed. It has also become common in server-side
programming, game development and the creation of desktop applications
(cf. <a href=https://en.wikipedia.org/wiki/JavaScript>https://en.wikipedia
.org/wiki/JavaScript</a>).
        </dd>
        <dt>PHP5</dt>
        <dd>PHP is a server-side scripting language designed for web development but
also used as a general-purpose programming language. PHP is now installed on
more than 244 million websites and 2.1 million web servers (cf. <a
        href=https://en.wikipedia.org/wiki/PHP5>
        https://en.wikipedia.org/wiki/PHP5</a>).
        </dd>
      </dl>
    </main>
  </body>
</html>

```

#### 4.1.10. <a>

From [www.w3schools.com/tags/tag\\_a.asp](http://www.w3schools.com/tags/tag_a.asp):

The `<a>` tag defines a hyperlink, which is used to link from one page to another.

The most important attribute of the `<a>` element is the `href` attribute, which indicates the link's destination.

By default, links will appear as follows in all browsers:

1. unvisited link
2. visited link
3. active link

This element supports the following particular attributes:

Name	Value	Description
<code>download</code>	<code>filename</code>	target will be downloaded instead of opened. If filename is omitted it will be saved under the original filename. This works only for files located on the same server than the current page.
<code>href</code>	URL	URL of the page
<code>hreflang</code>	<code>language_code</code>	language of the linked document
<code>media</code>	<code>media_query</code>	the medium that the document is optimized for
<code>rel</code>	<code>alternate</code> , <code>author</code> , <code>bookmark</code> , <code>help</code> , <code>license</code> , <code>next</code> , <code>nofollow</code> , <code>noreferrer</code> , <code>noopener</code> , <code>prefetch</code> , <code>prev</code> , <code>search</code> , <code>tag</code>	Relationship between the current and the linked document. <code>rel=noreferrer</code> will prevent the destination page from learning which page you came from. See <a href="https://developer.mozilla.org/en-US/docs/Web/HTML/Link_types">developer.mozilla.org/en-US/docs/Web/HTML/Link_types</a> for an excellent overview of all link types. Why are <code>noreferrer</code> or <code>noopener</code> important? See <a href="https://mathiasbynens.github.io/rel-noopener">mathiasbynens.github.io/rel-noopener</a> .
<code>target</code>	<code>_blank</code> , <code>_parent</code> , <code>_self</code> , <code>_top</code> <code>framename</code>	where to open the linked document

Here is an example that illustrates different values for the `href` attribute:

# Welcome to LAM.

[Contact us](#)

<students.btsi.lu/evegi144/WAD/HTML5/a1.html>

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>This is a simple hyperlink example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <header>
      <h1>Welcome to <a href=http://www.ltam.lu target=_blank>LAM</a>.</h1>
    </header>
    <main>
      <a href=a1contact.html>Contact us</a>
    </main>
  </body>
</html>
```

<students.btsi.lu/evegi144/WAD/HTML5/a1contact.html>

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>This is a simple hyperlink example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <a href="http://www.ltam.lu/index.php?portal=26" target=_blank>How to find
us</a>
      <a href=a1.html>Home</a>
      <a
href="mailto:gilles.everling@education.lu?cc=everybody@world.com&bcc=spy@NSA.gov&subje
ct=Top%20Secret%20Message&body=This%20is%20the%20message%20body">Email the author</a>
    </main>
  </body>
</html>
```

Note that we need to provide the correct path to the file that the hyperlink is linking to via the **href** attribute. If we are linking to a page on the Internet, we need to specify the complete Unified Resource Locator (URL, cf. [en.wikipedia.org/wiki/URL](https://en.wikipedia.org/wiki/URL)), which consists of the protocol, a colon, two slashes, a host, normally given as a domain name but sometimes as a literal Internet Protocol (IP)

address, optionally a port number and finally the full path of the resource. The protocol used to access Internet pages is called Hypertext Transfer Protocol (HTTP).

If we link to a file within our web site, we use a relative URL, as shown in line 12 of [a1.html](#). If we link to a file on another server, we need to provide an absolute URL as shown in line 9.

If we want to allow the user to send an email by clicking on a hyperlink, we use [mailto](#), as shown in lines 11 to 12 of [a1contact.html](#). Note that we can pass additional parameters such as carbon copy ([cc](#)) and black carbon copy ([bcc](#)) as well as the subject and body. We put a [?](#) in front of the parameters. We assign a value to a parameter using [=](#) and each parameter/value pair is separated by a [&](#). Also note that we need to encode spaces using [%20](#), which is the corresponding hexadecimal (32 decimal) ASCII code (cf. [www.w3schools.com/charsets/ref\\_html\\_ascii.asp](#)). The example above gives a validation error because of the new lines in the [href](#) attribute string. The whole string should be in a single line, but this would not allow the clean printing of the code in the book.

A hyperlink can also point to another place on the same page. For this purpose, we can use the [id](#) attribute on any tag. This is useful if our page contains a huge amount of content and we want to give the user the option to jump directly to a specific location on the page, instead of having to scroll down manually.

Here is an example: [www.w3schools.com/tags/tryit.asp?filename=tryhtml5\\_a\\_href\\_anchor](#)

### **Hyperlink inside a hyperlink**

To nest a hyperlink inside another hyperlink we can use the [<object>](#) tag:

[students.btsi.lu/evegi144/WAD/HTML5/a2.html](#)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>This is a nested hyperlink example</title>
    <meta charset=UTF-8>
    <style>
      main > a > div {
        margin:          10px;
        padding:          20px;
        background-color: greenyellow;
      }

      main > a > div > object > a {
        background-color: yellowgreen;
      }
    </style>
  </head>
  <body>
    <main>
      <a href=https://students.btsi.lu target=_blank>
        <div>students.btsi.lu (<object type=text/html><a
          href=https://students.btsi.lu/evegi144/WAD/WAD.html
target=_blank>WAD</a>)
        </object>
      </div>
    </a>
  </main>
</body>
</html>

```

## Hyperlink ping tracking

[www.ghacks.net/2019/04/20/mozilla-plans-to-enable-hyperlink-ping-tracking-by-default-in-firefox](http://www.ghacks.net/2019/04/20/mozilla-plans-to-enable-hyperlink-ping-tracking-by-default-in-firefox)

### 4.1.11. <img>

The <img> tag is used to insert an image. This element supports the following particular attributes:

Name	Value	Description
alt	text	alternate text for image, <b>required for successful validation</b>
crossorigin	anonymous, use-credentials	use third-party site images with canvas (cf. <a href="https://developer.mozilla.org/en-US/docs/HTML/CORS_settings_attributes">developer.mozilla.org/en-US/docs/HTML/CORS_settings_attributes</a> )
height	pixels	image height
ismap	ismap	image is a server-side map (cf. <a href="http://www.w3schools.com/tags/att_img_ismap.asp">www.w3schools.com/tags/att_img_ismap.asp</a> )
src	URL	image URL

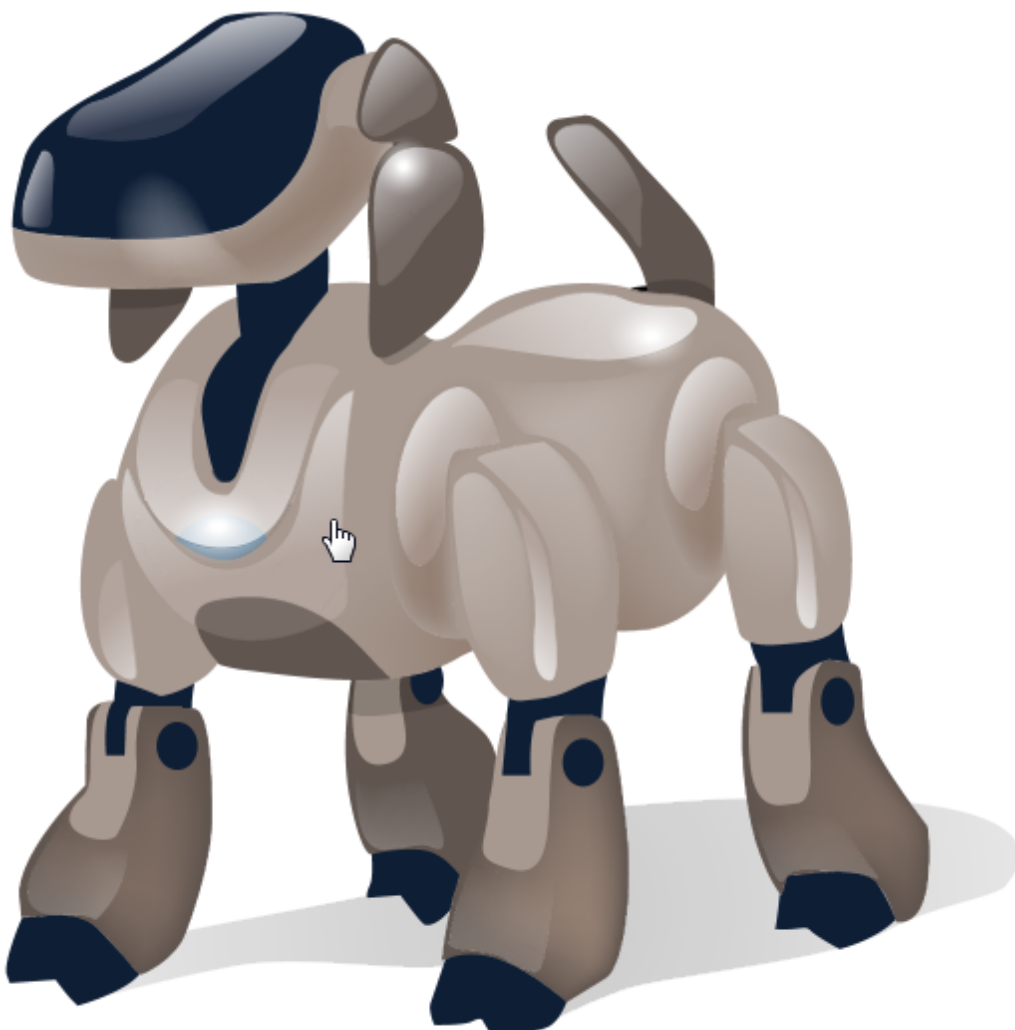
Name	Value	Description
usemap	#mapname	image is a client-side map (cf. <a href="http://www.w3schools.com/tags/att_img_usemap.asp">www.w3schools.com/tags/att_img_usemap.asp</a> )
width	pixels	image width

The alt attribute is required for successful validation. It can be used by screen readers, search engines and others. It will also be displayed by the browser in case the image cannot be displayed.



We should always specify the exact width and height of an image, as they allow the browser to allocate the space required to display the image before the image is loaded. Without them, the browser will have to adjust the page layout after the image has finished loading.

We do not use the width and height attributes to change the size of an image, as the full image will still be loaded by the browser. In order to produce a thumbnail for instance, we use a drawing program such as GIMP ([www.gimp.org](http://www.gimp.org)) or an online editor, e.g. [pixlr.com/editor](http://pixlr.com/editor) (cf. [Image resizing](#)). Be careful to specify the correct path and name of the image. Use `..` to go up to the parent folder and `/` to separate folder and file names. We can embed images inside hyperlinks, like so:



[https://www.iconfinder.com/icons/131462/automatic\\_auto...og\\_machine\\_machine\\_gun\\_robot\\_shadow\\_with\\_icon#size=512](https://www.iconfinder.com/icons/131462/automatic_auto...og_machine_machine_gun_robot_shadow_with_icon#size=512)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Image example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <a
href="https://www.iconfinder.com/icons/131462/automatic_automatic_machine_automaton_dog_machine_machine_gun_robot_shadow_with_icon#size=512">
        <img src=dog_robot_with_shadow.png width=512 height=512

alt="https://www.iconfinder.com/icons/131462/automatic_automatic_machine_automaton_dog_machine_machine_gun_robot_shadow_with_icon#size=512">
      </a>
    </main>
  </body>
</html>
```

Notice that we have to use quotes here for the value of the `href` and `alt` attributes, as they span several lines. The validator won't like this.

## Image formats

The three main image formats used on the Web are JPEG, PNG and GIF. There is a new kid on the block, called Scalable Vector Graphics (SVG), which requires a much deeper understanding to handle but offers a number of advantages, that we'll look into in section [SVG](#).

The key characteristics of the main image formats are summarized in the following table:

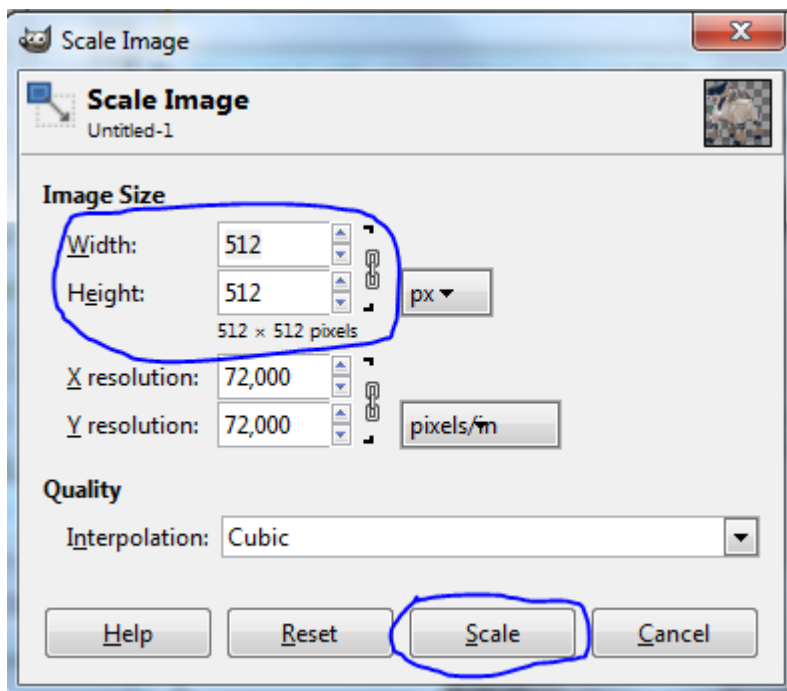
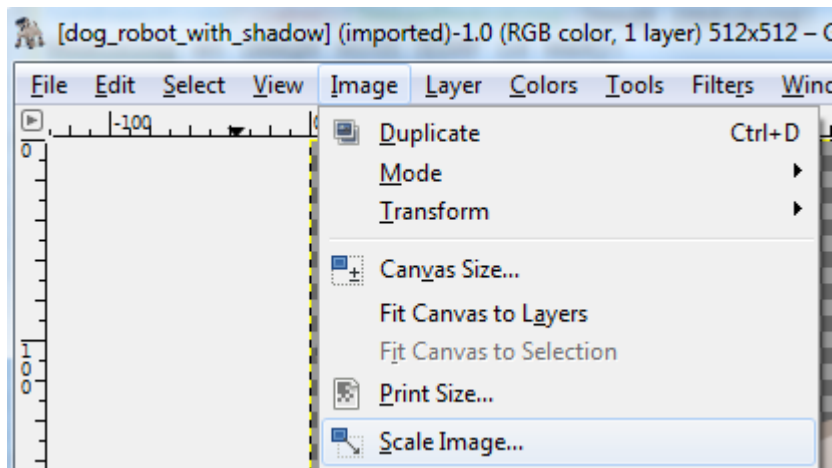
Format	Compression	Colors	Transparency	Animation
PNG	lossless	256 (8 bit), 16.7 million (24 bit) or 4.3 billion (32 bit)	yes	no
JPEG	lossy	16.7 million (24 bit)	no	no
GIF	lossless	256 (8 bit)	yes	yes

If you don't need animation, PNG is the preferred format, particularly for web graphics.

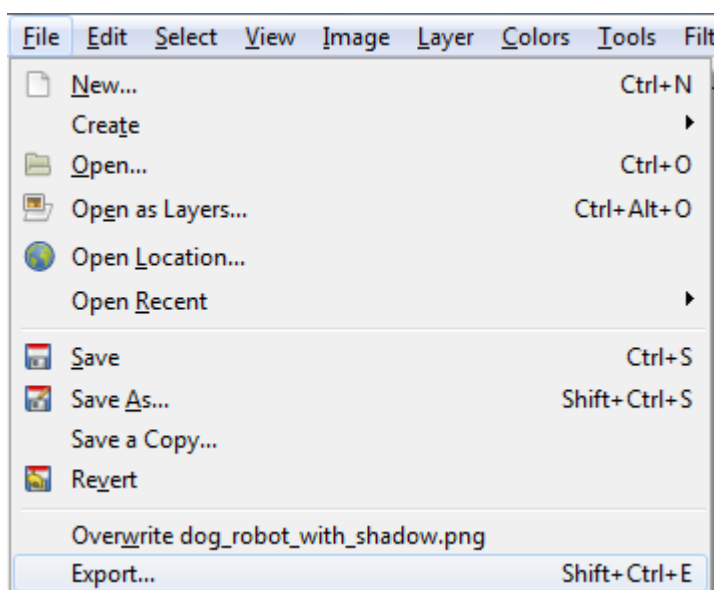
If you want to create animated GIFs, take a look at GifCam ([blog.bahraniapps.com/gifcam](http://blog.bahraniapps.com/gifcam)), which allows you create and edit screencasts and save them as compact GIF images that can be easily embedded in your HTML5, like so:

## Image resizing

Resizing an image with GIMP is easy:



After you've resized the image, export it under a new name:



## Adding transparency

We can either use a program such as Online Image Editor (cf. [www.online-image-editor.com/help/transparency](http://www.online-image-editor.com/help/transparency)) or use GIMP ([www.bogotobogo.com/Gif/gimp-tutorial-transparency.php](http://www.bogotobogo.com/Gif/gimp-tutorial-transparency.php)).

<picture>

[www.w3schools.com/tags/tag\\_picture.asp](http://www.w3schools.com/tags/tag_picture.asp)

<map>

The <map> tag is used to define a client-side image-map. See [www.w3schools.com/tags/att\\_img\\_usemap.asp](http://www.w3schools.com/tags/att_img_usemap.asp) for the details.

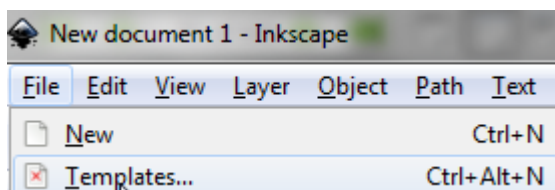
Here is a simple example application:

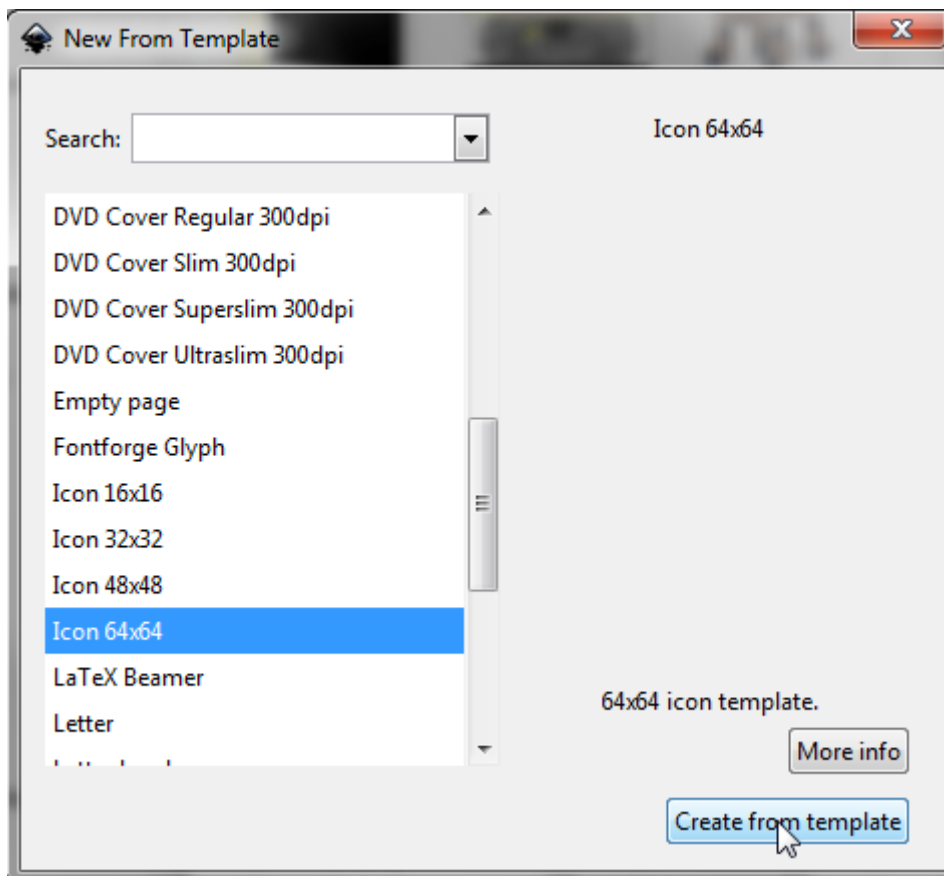
[students.btsi.lu/evegi144/WAD/HTML5/map1.html](http://students.btsi.lu/evegi144/WAD/HTML5/map1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Image map example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <img src=dog_robot_with_shadow.png width=512 height=512 usemap=#dogmap
        alt="https://www.iconfinder.com/icons/131462#size=512">
      <map name=dogmap>
        <area shape=rect coords=10,0,250,150 href=map1head.html alt=Head>
        <area shape=rect coords=50,151,400,340 href=map1body.html alt=Body>
        <area shape=rect coords=0,350,512,512 href=map1leg.html alt=Leg>
      </map>
    </main>
  </body>
</html>
```

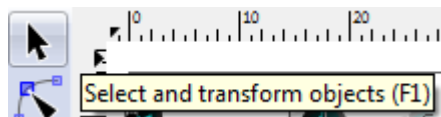
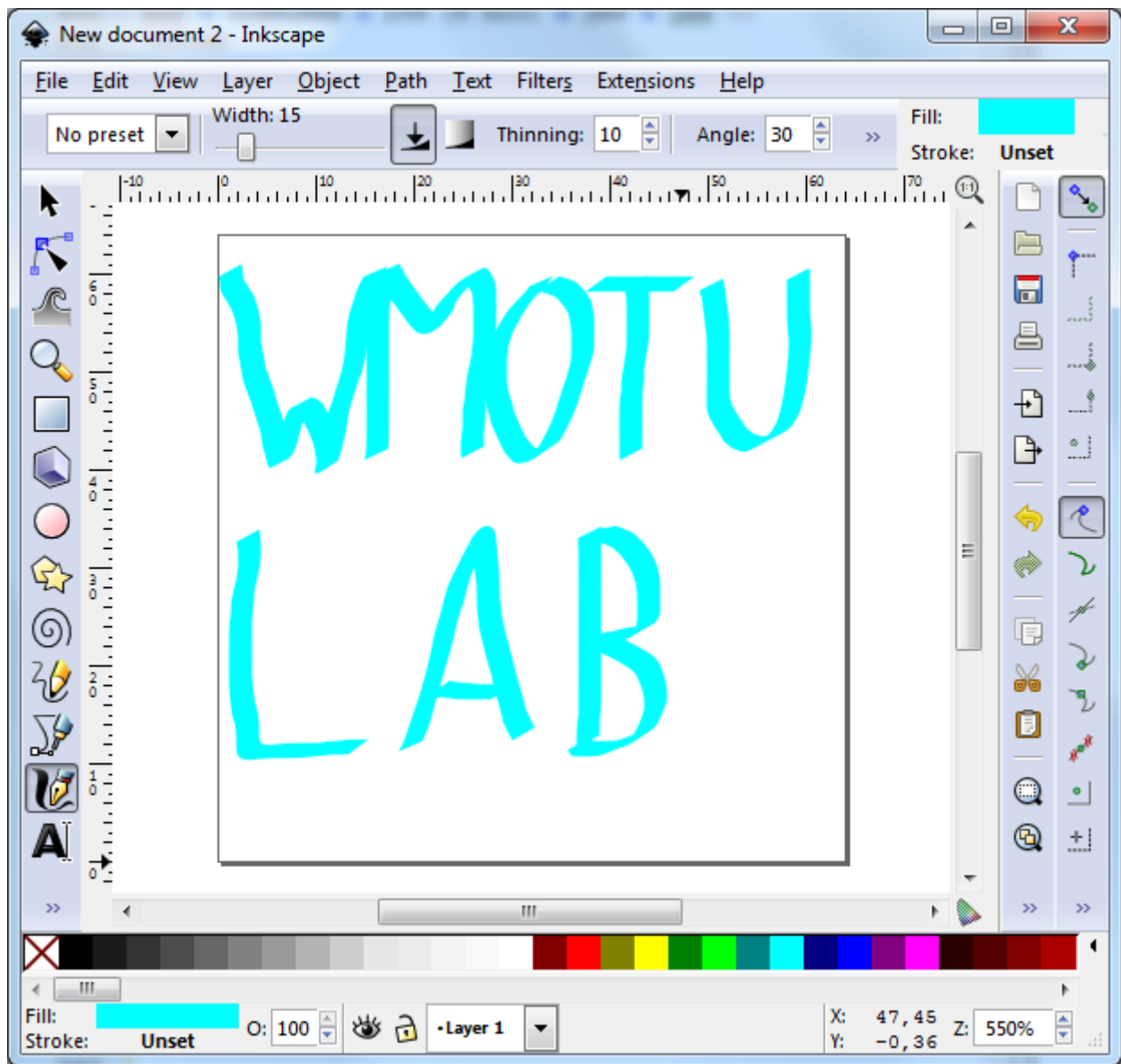
## Logo creation

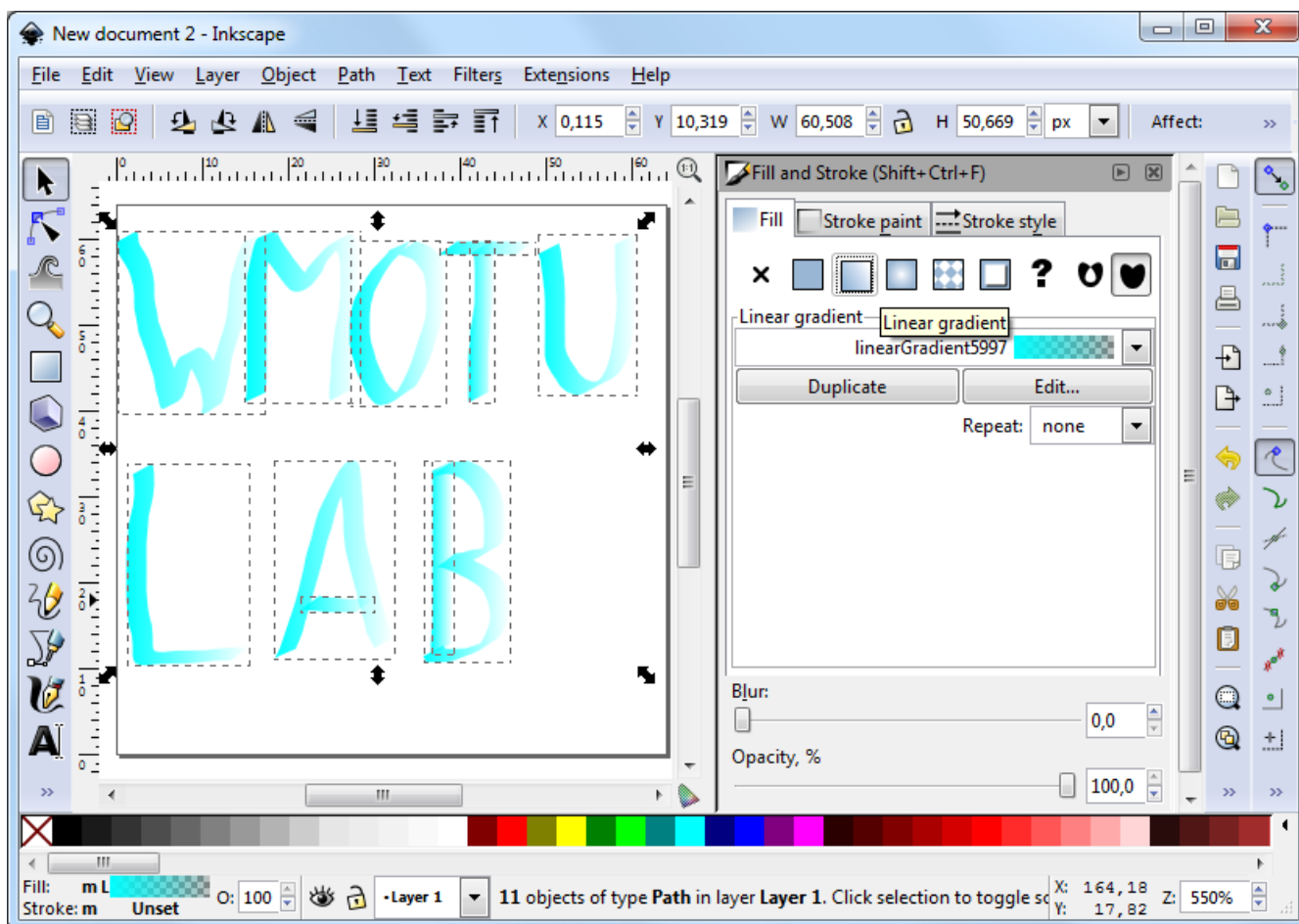
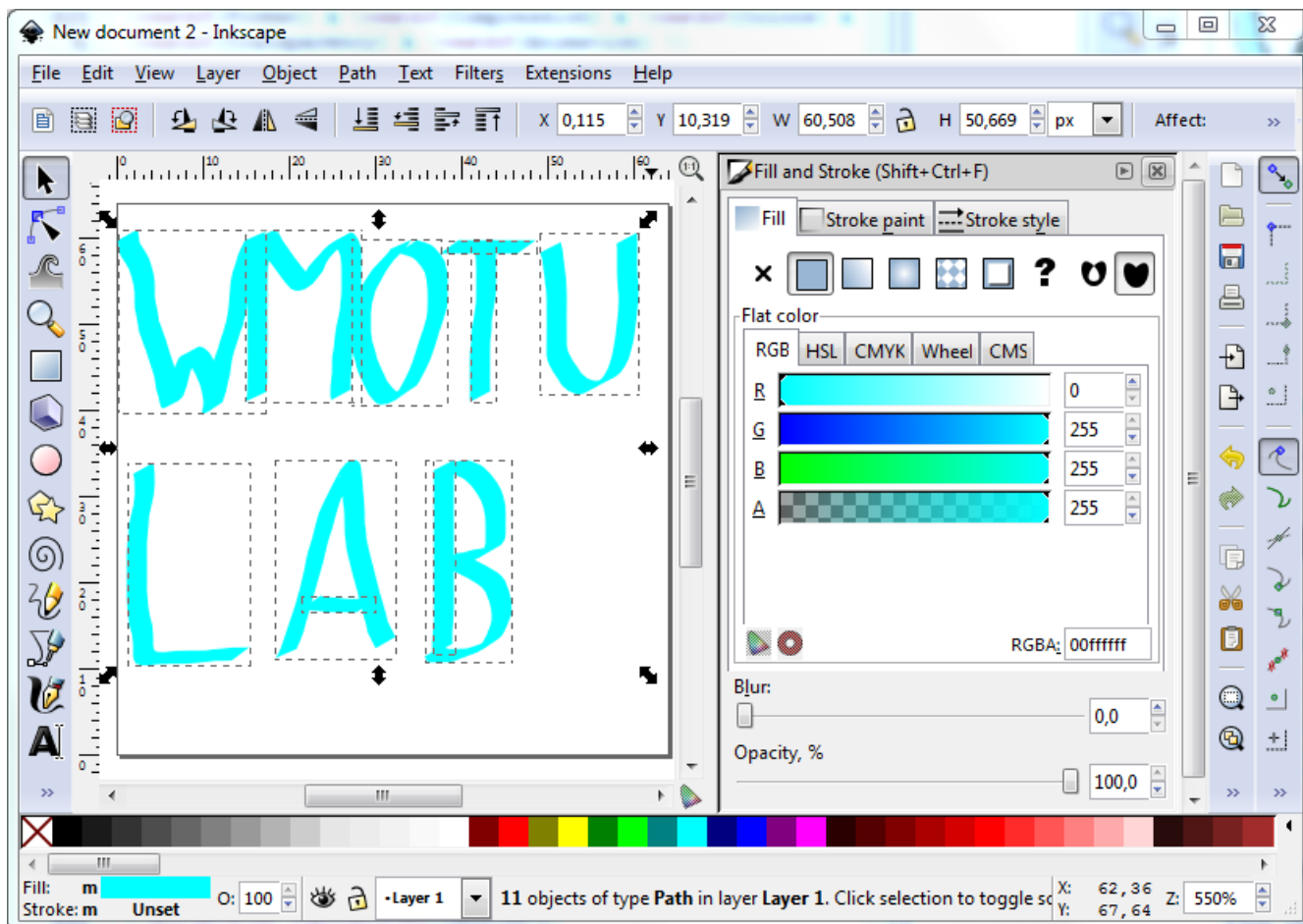
With Inkscape we can create a logo very easily.

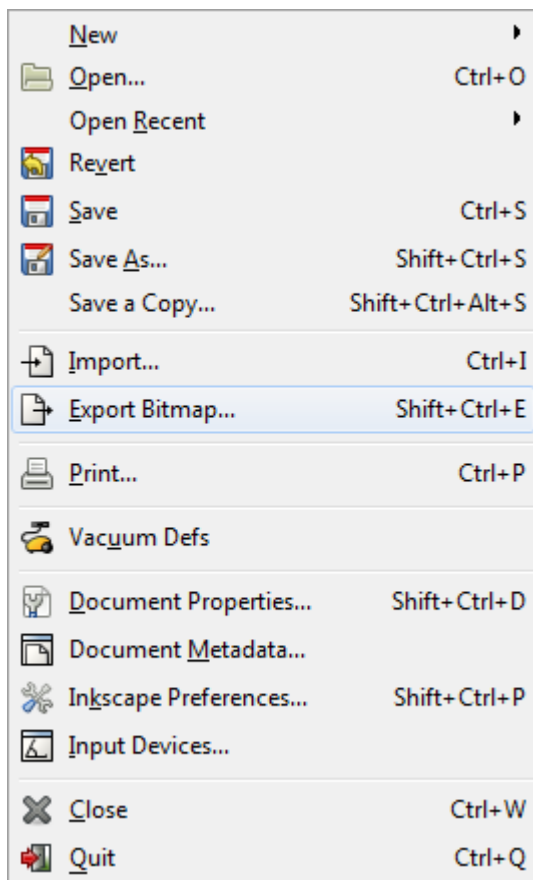
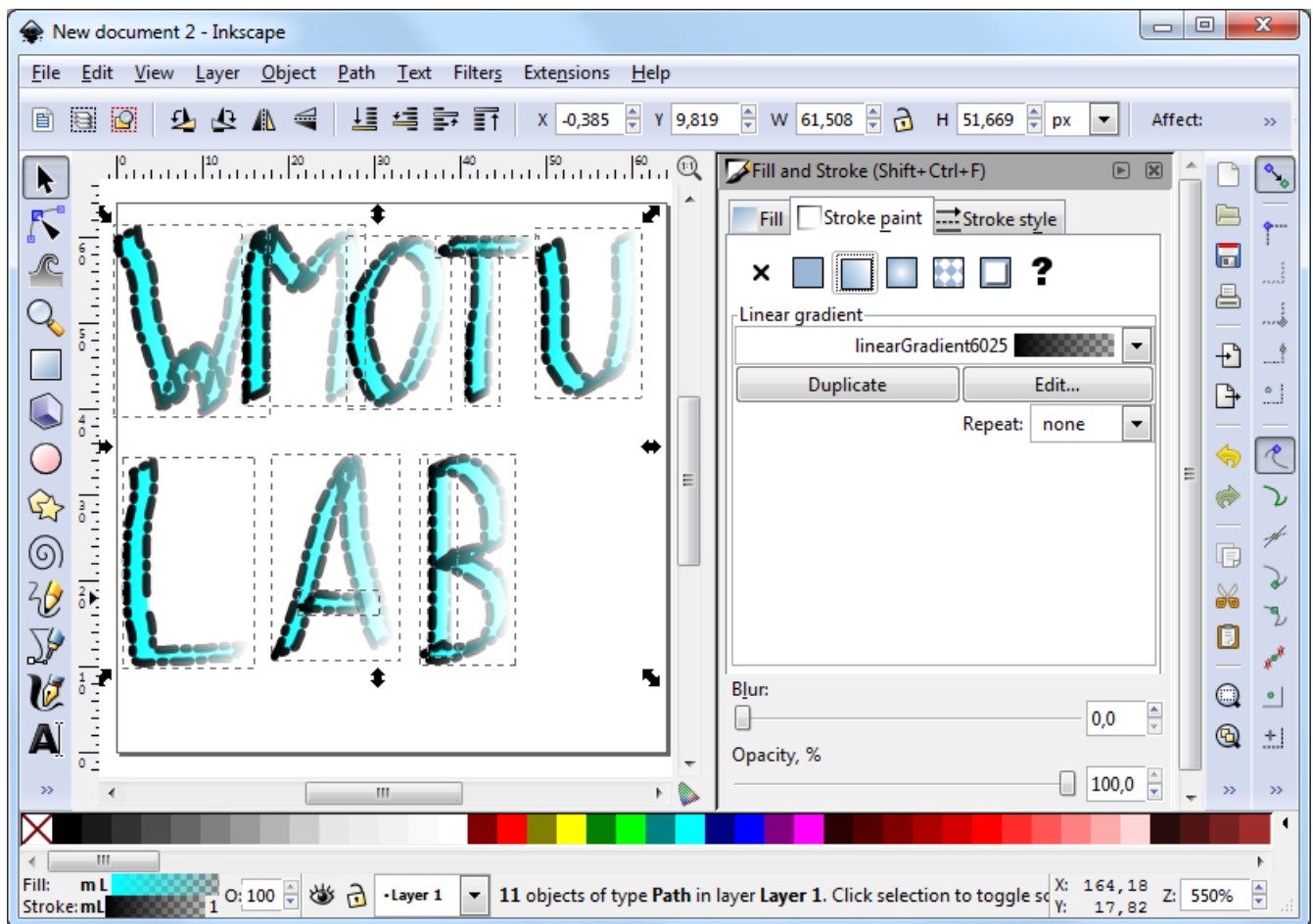




Draw calligraphic or brush strokes (Ctrl+F6)







Export the bitmap using **Ctrl + Shift + E** and select page as export area.

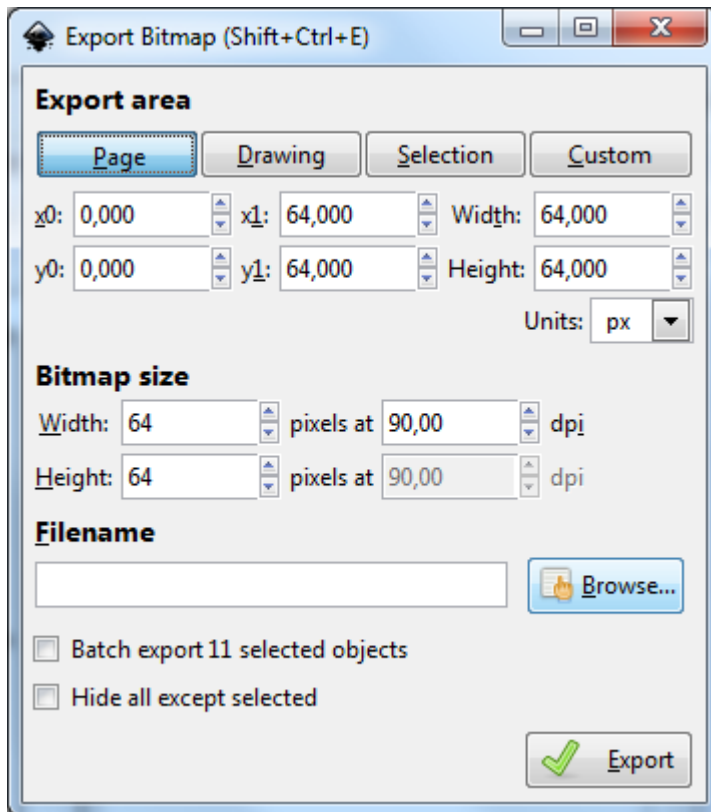


figure and figcaption

[www.w3schools.com/tags/tag\\_figcaption.asp](http://www.w3schools.com/tags/tag_figcaption.asp)

#### 4.1.12. <nav>

This element is used to create the main navigation on a site:



[students.btsi.lu/evegi144/WAD/HTML5/navigation1.html](http://students.btsi.lu/evegi144/WAD/HTML5/navigation1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Navigation</title>
    <meta charset=UTF-8>
    <link href=navigation1.css rel=stylesheet>
  </head>
  <body>
    <header>
      <h1>Navigation</h1>
      <nav>
        <ul>
          <li><a href=#>Home</a></li>
          <li><a href=#>About</a></li>
          <li><a href=#>Team</a></li>
          <li><a href=#>Shareholders</a></li>
          <li><a href=#>Contact</a></li>
        </ul>
      </nav>
    </header>
  </body>
</html>

```

Here the CSS3 file (don't worry, we'll address that topic in the next chapter):

[students.btsi.lu/evegi144/WAD/HTML5/navigation1.css](https://students.btsi.lu/evegi144/WAD/HTML5/navigation1.css)

```

body {
  background-color: lightseagreen;
}

h1 {
  margin: 0;
}

ul {
  list-style: none;
  padding: 0;
  margin: 0;
}

li {
  display: inline;
}

```

#### 4.1.13. <main>

From the W3C working group ([www.w3.org/TR/html51/semantics.html#the-main-element](http://www.w3.org/TR/html51/semantics.html#the-main-element)):

The **main** element represents the main content section of the body of a document or application. The main content section consists of content that is directly related to or expands upon the central topic of a document or central functionality of an application.

Note: the **main** element is not sectioning content and has no effect on the document outline.

The main content section of a document includes content that is unique to that document and excludes content that is repeated across a set of documents such as site navigation links, copyright information, site logos and banners and search forms (unless the document or applications main function is that of a search form).

Authors MUST NOT include more than one **main** element in a document.

Authors MUST NOT include the **main** element as a child of an **article**, **aside**, **footer**, **header** or **nav** element.

#### 4.1.14. <section>

From [www.w3.org/TR/html51/semantics.html#the-section-element](http://www.w3.org/TR/html51/semantics.html#the-section-element):

The section element represents a generic section of a document or application. A section, in this context, is a thematic grouping of content, typically with a heading.

Example:

## Section 1

This is a section with some content.

## Section 2

This is another section with some content.

[students.btsi.lu/evegi144/WAD/HTML5/section1.html](http://students.btsi.lu/evegi144/WAD/HTML5/section1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Section example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <header>
    </header>
    <main>
      <section>
        <h1>Section 1</h1>
        This is a section with some content.
      </section>
      <section>
        <h1>Section 2</h1>
        This is another section with some content.
      </section>
    </main>
  </body>
</html>

```

#### 4.1.15. <footer>

From [www.w3.org/TR/html51/semantics.html#the-footer-element](http://www.w3.org/TR/html51/semantics.html#the-footer-element):

The footer element represents a footer for its nearest ancestor sectioning content or sectioning root element. A footer typically contains information about its section such as who wrote it, links to related documents, copyright data, and the like.

When the footer element contains entire sections, they represent appendices, indexes, long colophons, verbose license agreements, and other such content.

Example: [students.btsi.lu/evegi144/WAD/HTML5/footer1.html](http://students.btsi.lu/evegi144/WAD/HTML5/footer1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Footer example</title>
    <meta charset=utf-8>
  </head>
  <body>
    <header>
      <nav></nav>
    </header>
    <main>

    </main>
    <footer>&copy; 2014 WMOTU</footer>
  </body>
</html>

```

#### 4.1.16. <article>

From [www.w3.org/TR/html51/semantics.html#the-article-element](http://www.w3.org/TR/html51/semantics.html#the-article-element):

The article element represents a complete, or self-contained, composition in a document, page, application, or site and that is, in principle, independently distributable or reusable, e.g. in syndication. This could be a forum post, a magazine or newspaper article, a blog entry, a user-submitted comment, an interactive widget or gadget, or any other independent item of content.

Example: [students.btsi.lu/evegi144/WAD/HTML5/article1.html](http://students.btsi.lu/evegi144/WAD/HTML5/article1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Article example</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <article>
        <h1>HTSTA</h1>
        <p>HTSTA is the first step to a fulfilling web developer career.</p>
      </article>
    </main>
  </body>
</html>

```

#### 4.1.17. <aside>

From [www.w3.org/TR/html51/semantics.html#the-aside-element](http://www.w3.org/TR/html51/semantics.html#the-aside-element):

The aside element represents a section of a page that consists of content that is tangentially related to the content around the aside element, and which could be considered separate from that content. Such sections are often represented as sidebars in printed typography.

The element can be used for typographical effects like pull quotes or sidebars, for advertising, for groups of nav elements, and for other content that is considered separate from the main content of the page.

The link above provides usage examples.

#### 4.1.18. <div>

From [www.w3.org/TR/html51/semantics.html#the-div-element](http://www.w3.org/TR/html51/semantics.html#the-div-element):

The div element has no special meaning at all. It represents its children. It can be used with the class, lang, and title attributes to mark up semantics common to a group of consecutive elements.

**Note:** Authors are strongly encouraged to view the div element as an element of last resort, for when no other element is suitable. Use of more appropriate elements instead of the div element leads to better accessibility for readers and easier maintainability for authors.

#### 4.1.19. <q>

To mark up a short quotation, we use the <q> tag. This element has one special attribute, cite, that can be used to specify the source URL of the quote.

Example:

[students.btsi.lu/evegi144/WAD/HTML5/q1.html](http://students.btsi.lu/evegi144/WAD/HTML5/q1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Quote example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      George Orwell's <q>Politics and the English Language</q> from 1946
    </main>
  </body>
</html>
```

#### 4.1.20. <blockquote>

To mark up a longer quotation from another source, use the <blockquote> tag:

George Orwell, in his “Politics and the English Language” from 1946, provided the following insight:

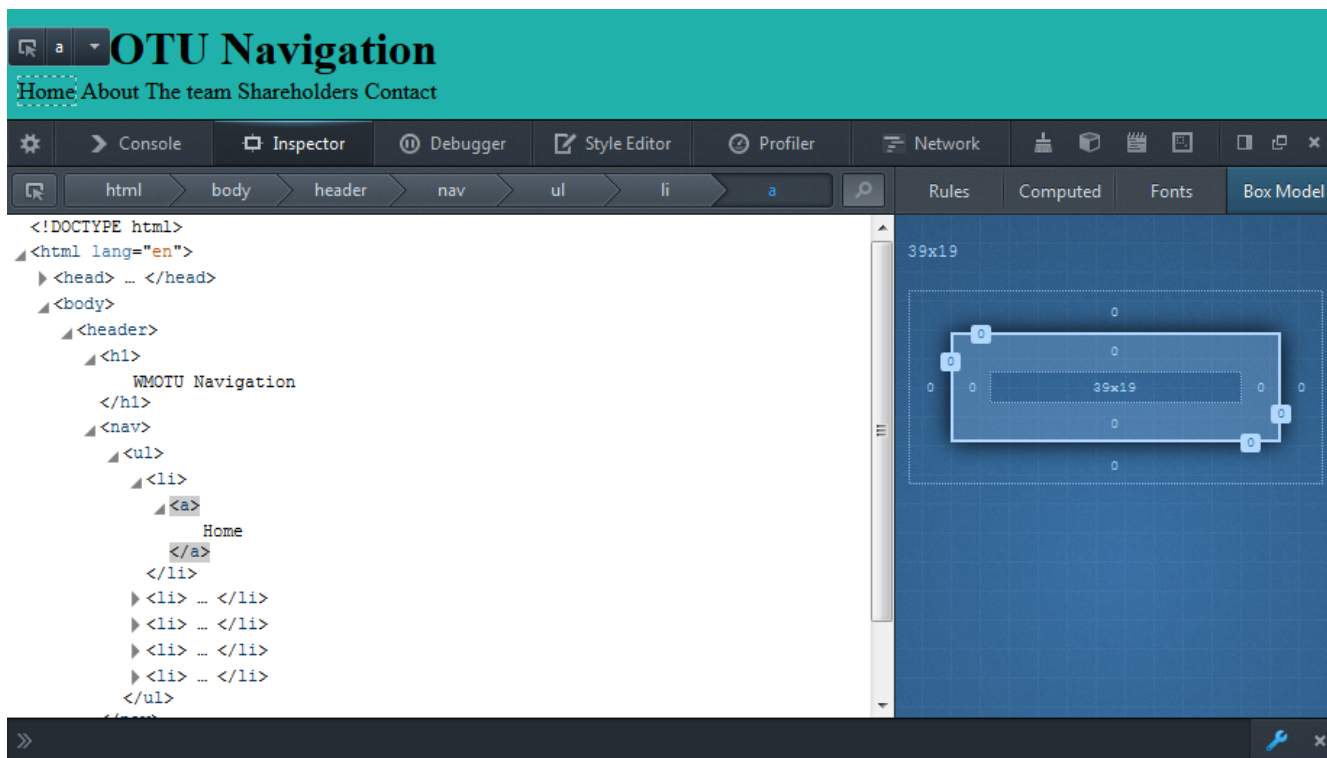
Political language... is designed to make lies sound truthful and murder respectable, and to give an appearance of solidity to pure wind.

[students.btsi.lu/evegi144/WAD/HTML5/blockquote1.html](https://students.btsi.lu/evegi144/WAD/HTML5/blockquote1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Blockquote example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      George Orwell, in his <q>Politics and the English Language</q> from 1946,
      provided the following insight:
      <blockquote>
        Political language... is designed to make lies sound truthful and murder
        respectable,
        and to give an appearance of solidity to pure wind.
      </blockquote>
    </main>
  </body>
</html>
```

#### 4.1.21. Firefox console and Firebug

By pressing **Shift** + **F2** you open the Firefox console, which is a great tool to analyse web pages:



The console tab displays information, warning and error messages and will be one of our most important development tools throughout our web app development journey. The inspector enables us to take a closer look at the styling of a particular element. The debugger will be very helpful to track errors in our JavaScript adventures. The style editor permits the real time changing of the current web page's styles. Try it! The profiler serves to analyse the performance of our app and detect bottleneck. The network tab displays detailed information about what happens on the network. This will be very helpful once we start using HTTP forms and Ajax.

By pressing **F12** you open Firebug, if this plugin is installed in your Firefox browser. If not, you can install it by selecting **Tools > Add-ons**. Search for Firebug and install it.

Firebug provides even more advanced analysis functionality than the console.

#### 4.1.22. `<base>`

[www.w3schools.com/tags/tag\\_base.asp](http://www.w3schools.com/tags/tag_base.asp)

#### 4.1.23. `<link>`

[www.w3schools.com/tags/tag\\_link.asp](http://www.w3schools.com/tags/tag_link.asp)

### Style sheet

We have already seen at the beginning of this chapter an example of an external style sheet inclusion. We just have to include a link with the correct relationship attribute in the head of our document:

```
<link rel=stylesheet href=style.css>
```

## Favicon

[en.wikipedia.org/wiki/Favicon](https://en.wikipedia.org/wiki/Favicon)



[students.btsi.lu/evegi144/WAD/HTML5/favicon1.html](http://students.btsi.lu/evegi144/WAD/HTML5/favicon1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Favicon example 1</title>
    <meta charset=UTF-8>
    <link rel=icon href=favicon.ico>
  </head>
  <body>
    <main>
    </main>
  </body>
</html>
```

We can even have animated favicons, at least in Firefox:

[students.btsi.lu/evegi144/WAD/HTML5/favicon2.html](http://students.btsi.lu/evegi144/WAD/HTML5/favicon2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Favicon example 2</title>
    <meta charset=UTF-8>
    <link rel=icon href=bear.gif type=image/gif>
  </head>
  <body>
    <main>
    </main>
  </body>
</html>
```

Icon files can be 16×16, 32×32, 48×48, or 64×64 pixels in size, and 8-bit, 24-bit, or 32-bit in color depth.

### 4.1.24. <meta>

The <meta> tag is used to provide metadata, i.e. data that describes the document. This data is not displayed on the page, but can be processed by the browser, search engines or other web services (cf. [developer.mozilla.org/en-US/docs/Learn/HTML/Introduction\\_to\\_HTML/The\\_head\\_metadata\\_in\\_HTML](https://developer.mozilla.org/en-US/docs/Learn/HTML/Introduction_to_HTML/The_head_metadata_in_HTML)).

This element supports the following particular attributes:

Name	Value	Description
charset	character set	character encoding for the document, we use utf-8
content	text	value associated with the http-equiv or name attribute
http-equiv	content-type, default-style, refresh	create HTTP header for content attribute
name	application-name, author, description, generator, keywords	name for the metadata

We use the charset attribute to specify the character encoding of our document. This should be set to Unicode, i.e. **utf-8**, as it allows us to use language specific characters such as é and ä. A list of all available character encodings can be found at [www.iana.org/assignments/character-sets/character-sets.xhtml](http://www.iana.org/assignments/character-sets/character-sets.xhtml).

Before the advent of HTML5, **http-equiv** was used to set the character encoding, but no more. The value **default-style** can be used to specify the preferred stylesheet from a selection of link or style elements in case there are several in your document. The value **refresh** can be used to specify after how many seconds a page should be automatically refreshed (i.e. reloaded) or if it should redirect to another page. This can be useful for a site whose content changes rapidly and where you don't want to use JavaScript. For instance, let's assume you have a new web site and you want users to be automatically transferred from your old site to the new one. On your old site, you'd have the following main page: [students.btsi.lu/evegi144/WAD/HTML5/metaredirect1.html](http://students.btsi.lu/evegi144/WAD/HTML5/metaredirect1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Redirection example</title>
    <meta charset=UTF-8>
    <meta http-equiv=refresh content="3; url=metaredirect2.html">
  </head>
  <body>
    <main>
      <header>
        <h1>You'll be redirected to my new site in 3 seconds.</h1>
      </header>
    </main>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/HTML5/metaredirect2.html](http://students.btsi.lu/evegi144/WAD/HTML5/metaredirect2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Redirection example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <header>
        <h1>Welcome to my new site.</h1>
        Hope you like it!
      </header>
    </main>
  </body>
</html>
```

See also [www.w3.org/TR/WCAG20-TECHS/H76.html](http://www.w3.org/TR/WCAG20-TECHS/H76.html).

The **name** attribute can take one of the following values:

Value	Description
<b>application-name</b>	name of the Web application
<b>author</b>	document author
<b>description</b>	description of the page content, can be used by search engines
<b>generator</b>	if the page was generated by a specific software
<b>keywords</b>	comma-separated list of keywords relevant to the page content targeted at search engines

Let's look at an example:

[students.btsi.lu/evegi144/WAD/HTML5/metaname1.html](http://students.btsi.lu/evegi144/WAD/HTML5/metaname1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Meta name usage example</title>
    <meta charset=UTF-8>
    <meta name=application-name content="Meta name usage example">
    <meta name=author content="Gilles Everling">
    <meta name=description content="Meta name usage">
    <meta name=keywords content=meta,name,HTML5>
  </head>
  <body>
    <main>
      <header>
        <h1>A simple application of the <code>meta name</code> attribute.</h1>
      </header>
    </main>
  </body>
</html>
```

#### 4.1.25. <table>

Tables are used to display tabular data, for instance the current national football league rankings. HTML5 tables must not be used for layout purposes. We'll look at a number of appropriate ways to create a tabular layout later on.

A table consists of rows (<tr>) and cells (<td>). Thus, a Tic Tac Toe table would look like this:

```
XXO
XXO
OOX
```

[students.btsi.lu/evegi144/WAD/HTML5/table1.html](https://students.btsi.lu/evegi144/WAD/HTML5/table1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>A simple Tic Tac Toe table</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <table>
        <tr>
          <td>X</td>
          <td>X</td>
          <td>O</td>
        </tr>
        <tr>
          <td>X</td>
          <td>X</td>
          <td>O</td>
        </tr>
        <tr>
          <td>O</td>
          <td>O</td>
          <td>X</td>
        </tr>
      </table>
    </main>
  </body>
</html>

```

The `<table>` tag supports only one attribute, `border`. It can have no value, `""` or `1`. In each of these cases, the table and each cell will have a border 1 pixel wide. Without this attribute the table and cells will have no border.

In most cases it is useful to have a table header, which contains the name or description of the data for each column. We may also need a table footer (cf. [www.w3.org/TR/html51/tabular-data.html#the-tfoot-element](http://www.w3.org/TR/html51/tabular-data.html#the-tfoot-element)). For this purpose, we split the table into a head, a body and a footer part, using the `<thead>`, `<tbody>` and `<tfoot>` tags. Header cells are displayed centered and bold using the `<th>` tag instead of `<td>`. We can add a caption using the `<caption>` tag. Here's an example:

**This term's grades**

Subject	Grade
SYSEX1	51
MATHE1	45
Average	48

[students.btsi.lu/evegi144/WAD/HTML5/table2.html](http://students.btsi.lu/evegi144/WAD/HTML5/table2.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>A simple table with caption, header, body and footer</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <table border>
        <caption>This term's grades</caption>
        <thead>
          <tr>
            <th>Subject</th>
            <th>Grade</th>
          </tr>
        </thead>
        <tbody>
          <tr>
            <td>SYSEX1</td>
            <td>51</td>
          </tr>
          <tr>
            <td>MATHE1</td>
            <td>45</td>
          </tr>
        </tbody>
        <tfoot>
          <tr>
            <td>Average</td>
            <td>48</td>
          </tr>
        </tfoot>
      </table>
    </main>
  </body>
</html>

```

We can have cells span several columns and/or several rows, using the **colspan** and **rowspan** attributes. Example:

---

Name	Village	Phone numbers	
Asterix	Gaul	123 456	621 123 456
Obelix		123 457	621 123 457

[students.btsi.lu/evegi144/WAD/HTML5/table3.html](https://students.btsi.lu/evegi144/WAD/HTML5/table3.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>colspan and rowspan example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <table border>
        <thead>
          <tr>
            <th>Name</th>
            <th>Village</th>
            <th colspan=2>Phone numbers</th>
          </tr>
        </thead>
        <tbody>
          <tr>
            <td>Asterix</td>
            <td rowspan=2>Gaul</td>
            <td>123 456</td>
            <td>621 123 456</td>
          </tr>
          <tr>
            <td>Obelix</td>
            <td>123 457</td>
            <td>621 123 457</td>
          </tr>
        </tbody>
      </table>
    </main>
  </body>
</html>

```

The `<th>` and `<td>` tags can have a `headers` attribute. It links a cell to a given header cell. For this to work, the header cell needs an id. This has no impact on the page display, but may be used by screen readers. See [www.w3schools.com/tags/att\\_th\\_headers.asp](http://www.w3schools.com/tags/att_th_headers.asp) for an example.

The `<th>` tag can have a `scope` attribute, which indicates whether a header cell is a header for a column, row, or group of columns or rows. See [www.w3schools.com/tags/att\\_th\\_scope.asp](http://www.w3schools.com/tags/att_th_scope.asp) and [developer.mozilla.org/en-US/docs/Learn/HTML/Tables/Advanced](http://developer.mozilla.org/en-US/docs/Learn/HTML/Tables/Advanced).

For formatting purposes, we can use the `<colgroup>` tag, see [www.w3schools.com/tags/tag\\_colgroup.asp](http://www.w3schools.com/tags/tag_colgroup.asp).

Soon, we'll see how we can style tables with CSS. Here's a little foretaste:

# TOIF1 HTSTA

## Nationale Fußballtabelle

22.5.13

Rang	Mannschaft	Punkte	Tore A	Tore B	Diff.	Spiele
1.	Fola	53	58	20	38	24
2.	F91	49	45	17	28	24
3.	Jeunesse	47	50	25	25	24
4.	RM Hamm Benfica	42	47	34	13	24
5.	Déifferdeng03	41	46	31	15	24
6.	Gréiwemaacher	36	40	32	8	24
7.	Kanech	34	38	33	5	24
8.	Käerjéng	31	42	48	-6	24
9.	RFCUL	30	38	37	1	24
10.	Wolz 71	30	39	61	-22	24
11.	Progrès Nidderkuer	24	26	41	-15	24
12.	Etzella	24	35	55	-20	24
13.	Peiteng	14	20	50	-30	24
14.	Kayl/Teiteng Union05	12	27	67	-40	24

[students.btsi.lu/evegi144/WAD/HTML5/table4.html](http://students.btsi.lu/evegi144/WAD/HTML5/table4.html)

```
<!DOCTYPE html>
<html lang=de>
  <head>
    <title>HTML and CSS Table Demo</title>
    <meta charset=UTF-8>
    <style>
      body {
        background: radial-gradient(rgb(200, 50, 20), rgb(255, 255, 60), rgb(255, 50,
20),
        black) no-repeat fixed;
        overflow: hidden;
      }

      h1 {
        font: 6em impact fantasy;
```

```
text-shadow: 3px 3px white;
animation: introAnimation 7s;
-webkit-animation: introAnimation 7s;
overflow: hidden;
text-align: center;
position: absolute;
}
```

```
@keyframes introAnimation {
  0% {
    top: 400px;
    font-size: 0.1em;
    transform: rotate(0deg);
  }
  100% {
    transform: rotate(720deg);
    top: 0;
  }
}
```

```
@-webkit-keyframes introAnimation {
  0% {
    top: 400px;
    font-size: 0.1em;
    transform: rotate(0deg);
  }
  100% {
    transform: rotate(720deg);
    top: 0;
  }
}
```

```
table {
  position: absolute;
  overflow: hidden;
  top: 200px;
  left: 0;
  color: white;
  border: 2px ridge red;
  border-spacing: 0;
  transition: left 5s;
  text-shadow: 1px 1px black;
}
```

```
table:hover {
  color: gold;
  left: 5000px;
}
```

```
table caption {
  font-size: 2em;
}
```

```

}

th {
    background-color: lightblue;
    text-align: left;
    padding: 5px;
    margin: 0;
}

td {
    padding: 5px;
}

tr {
    padding: 5px;
}

tr:nth-of-type(odd) {
    background-color: green;
}
</style>
</head>
<body>
<header>
    <h1>T0IF HTSTA</h1>
</header>
<main>
    <table>
        <caption>Nationale Fußballtabelle</caption>
        <thead>
            <tr>
                <th></th>
                <th>Rang</th>
                <th>Mannschaft</th>
                <th>Punkte</th>
                <th>Tore A</th>
                <th>Tore B</th>
                <th>Diff.</th>
                <th>Spiele</th>
            </tr>
        </thead>
        <tbody>
            <tr>
                <td rowspan=14 style="padding: 0; transform: rotate(-90deg);
                font-size: 3em;">22.5.13
                </td>
                <td>1.</td>
                <td>Fola</td>
                <td>53</td>
                <td>58</td>
                <td>20</td>

```

<td>38</td>	
<td>24</td>	

<td>2.</td>	
<td>F91</td>	
<td>49</td>	
<td>45</td>	
<td>17</td>	
<td>28</td>	
<td>24</td>	

<td>3.</td>	
<td>Jeunesse</td>	
<td>47</td>	
<td>50</td>	
<td>25</td>	
<td>25</td>	
<td>24</td>	

<td>4.</td>	
<td>RM Hamm Benfica</td>	
<td>42</td>	
<td>47</td>	
<td>34</td>	
<td>13</td>	
<td>24</td>	

<td>5.</td>	
<td>Déifferdeng03</td>	
<td>41</td>	
<td>46</td>	
<td>31</td>	
<td>15</td>	
<td>24</td>	

<td>6.</td>	
<td>Gréiwemaacher</td>	
<td>36</td>	
<td>40</td>	
<td>32</td>	
<td>8</td>	
<td>24</td>	

<td>7.</td>	
<td>Kanech</td>	

```

        <td>34</td>
        <td>38</td>
        <td>33</td>
        <td>5</td>
        <td>24</td>
    </tr>
    <tr>
        <td>8.</td>
        <td>Käerjéng</td>
        <td>31</td>
        <td>42</td>
        <td>48</td>
        <td>-6</td>
        <td>24</td>
    </tr>
    <tr>
        <td>9.</td>
        <td>RFCUL</td>
        <td>30</td>
        <td>38</td>
        <td>37</td>
        <td>1</td>
        <td>24</td>
    </tr>
    <tr>
        <td>10.</td>
        <td>Wolz 71</td>
        <td>30</td>
        <td>39</td>
        <td>61</td>
        <td>-22</td>
        <td>24</td>
    </tr>
    <tr>
        <td>11.</td>
        <td>Progrès Nidderkuer</td>
        <td>24</td>
        <td>26</td>
        <td>41</td>
        <td>-15</td>
        <td>24</td>
    </tr>
    <tr>
        <td>12.</td>
        <td>Etzella</td>
        <td>24</td>
        <td>35</td>
        <td>55</td>
        <td>-20</td>
        <td>24</td>
    </tr>

```

```

        <tr>
            <td>13.</td>
            <td>Peiteng</td>
            <td>14</td>
            <td>20</td>
            <td>50</td>
            <td>-30</td>
            <td>24</td>
        </tr>
        <tr>
            <td>14.</td>
            <td>Kayl/Teiteng Union05</td>
            <td>12</td>
            <td>27</td>
            <td>67</td>
            <td>-40</td>
            <td>24</td>
        </tr>
    </tbody>
</table>
</main>
<script>
    document.querySelector("h1").style.left = (window.innerWidth - 566) / 2 + "px";
    addEventListener('resize', () => {
        document.querySelector("h1").style.left = (window.innerWidth - 566) / 2 +
"px";
    });
</script>
</body>
</html>

```

#### 4.1.26. Forms

HTML forms ([www.w3.org/TR/html51/semantics.html#forms](http://www.w3.org/TR/html51/semantics.html#forms)) are used to pass data to a server. They are the building blocks that allow the user to provide data to our application. When you use a search engine or buy something in an online shop, you use forms to enter your information.

An HTML form can consist of different input elements such as text fields, check boxes, radio-buttons, submit buttons, selection lists, text areas, labels etc.

Here's the full list of form tags to be used within `<form>`:

Tag	Description
<code>&lt;input&gt;</code>	single line text field
<code>&lt;textarea&gt;</code>	multi line text area
<code>&lt;label&gt;</code>	label, i.e. a text to be displayed next to a form element
<code>&lt;fieldset&gt;</code>	groups related elements
<code>&lt;legend&gt;</code>	caption (short description) for a field set

Tag	Description
<select>	drop-down list
<optgroup>	group of related options in a drop-down list
<option>	option in a drop-down list
<button>	clickable button
<datalist>	option list
<keygen>	key-pair generator field
<output>	result of a calculation

All form elements are enclosed within the `<form>` tag. Here's a simple example:

First name:

Last name:

[students.btsi.lu/evegi144/WAD/HTML5/forms1.html](https://students.btsi.lu/evegi144/WAD/HTML5/forms1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>A first form example using post</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <form method=post action=forms1.php>
        First name: <input name=first_name required><br>
        Last name: <input name=last_name required><br>
        <input type=submit>
      </form>
    </main>
  </body>
</html>
```

When you submit your input, the `forms1.php` script gets executed on the server:

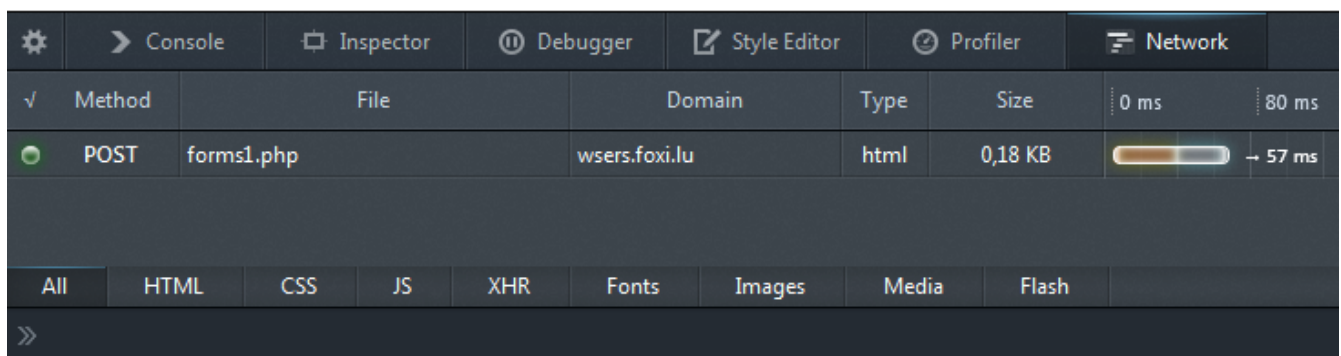
```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>A first form example using post</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <?php
        if (isset($_POST['first_name'], $_POST['last_name']))
          echo '<p>Hi ' . $_POST['first_name'] . ' ' . $_POST['last_name'] . '</p>';
      ?>
    </main>
  </body>
</html>

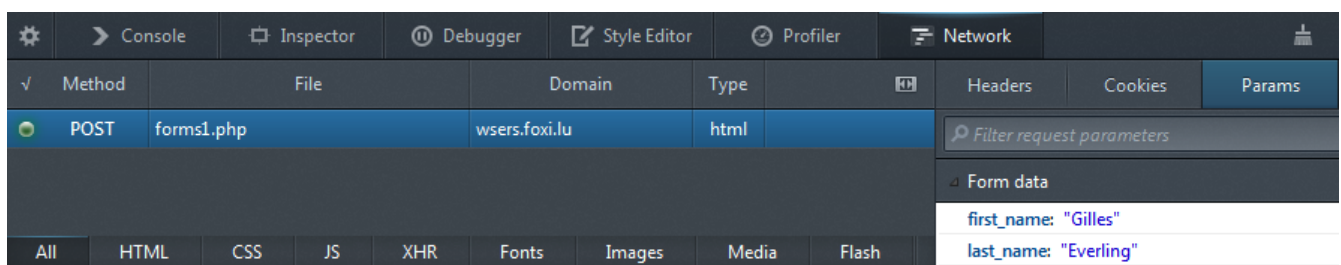
```

Don't worry about the PHP part, we'll get into that later. Our example uses two of the most important `<form>` tag attributes: `action` and `method`. The former specifies the script on the server that should receive the form data. The latter indicates the method that should be used to send the data to the server, either `GET`, which sends the data via the URL or `POST`, which sends the data embedded within the HTTP request. Run our first form example, open the Firefox console and select the **Network** tab. Now enter your first and last names and submit the form. You should see something similar to this:

**Hi Gilles Everling**



Now click on the `forms1.php` line and select the Params tab on the right:



As you can see, your input has been sent as form data, i.e. embedded inside the form.

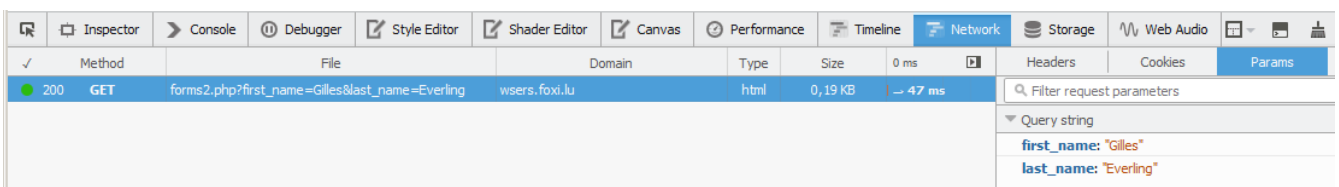
Now let's check what happens if we use the `GET` method: [students.btsi.lu/evegi144/WAD/HTML5/forms2.html](https://students.btsi.lu/evegi144/WAD/HTML5/forms2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>A first form example using get</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <form action=forms2.php>
        First name: <input name=first_name required><br>
        Last name: <input name=last_name required><br>
        <input type=submit>
      </form>
    </main>
  </body>
</html>
```

When you submit your input, the `forms2.php` script gets executed on the server:

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>A first form example using get</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <?php
        if (isset($_GET['first_name']) && isset($_GET['last_name']))
          echo '<p>Hi ' . $_GET['first_name'] . ' ' . $_GET['last_name'] . '</p>';
      ?>
    </main>
  </body>
</html>
```

If you perform the same network analysis as before, you get this:



You should notice two changes:

1. The URL contains a `?` and a `&` as well as the data you entered in the form.
2. The parameter tab says that the data has been sent as query string, i.e. as part of the URL and not embedded as form data.

This means that the form data is visible to everyone and can be easily intercepted, whereas for the **POST** method this is a little bit more difficult. **GET** submissions can be bookmarked and URLs are usually stored in log files on the server, whereas the body of HTTP requests usually is not and can also not be bookmarked. We therefore prefer to use the **POST** method.

Here is an overview of all the attributes specific to the **<form>** tag:

Attribute	Value	Description
<b>accept-charset</b>	character set	character encoding to be used for form submission
<b>action</b>	URL	script to receive the form data
<b>autocomplete</b>	on/off	turn autocomplete on or off
<b>enctype</b>	<b>application/x-www-form-urlencoded, multipart/form-data, text/plain</b>	how the data should be encoded (only for <b>POST</b> method)
<b>method</b>	<b>GET</b> or <b>POST</b>	HTTP method to be used
<b>name</b>	text	name of the form
<b>novalidate</b>		the form should not be validated upon submission
<b>target</b>	<b>_blank, _self, _parent, _top</b>	display response from server in a new, the current, the parent or the top window/tab

Given the excellent documentation on W3Schools regarding all form elements, I will not repeat the details here but instead refer you to their site [www.w3schools.com/html/html\\_forms.asp](http://www.w3schools.com/html/html_forms.asp). **Go through the examples and get a feel for forms. We'll use them throughout our Web app development journey.**

Here is a more complete example:

### Personal data

First name:

Last name:

Password:

Email:

☒ Male

☐ Female

☐ I have a bike

☐ I have a car

### Other data

Saab

Color:

Number (1-10):

Number range:

Submit

Reset

0  +

50  =

[students.btsi.lu/evegi144/WAD/HTML5/forms3.html](https://students.btsi.lu/evegi144/WAD/HTML5/forms3.html)

```
<!DOCTYPE html>
<html lang=en>
<head>
  <title>A more advanced form example</title>
  <meta charset=utf-8>
  <style>
    form {
      width:      350px;
      margin-left: auto;
      margin-right: auto;
    }

    form label {
      float:      left;
      width:      150px;
```

```

        text-align:    right;
        padding-right: 10px;
        margin-top:    10px;
    }

    form input {
        margin-top:    10px;
        text-shadow:    1px 1px 1px white;
        border-radius: 5px;
    }

    form input:focus {
        background-color: yellow;
    }

    form input[type=submit], form input[type=button], form input[type=reset] {
        background: linear-gradient(to bottom right, yellow, red);
        margin-left: 160px;
        width:       190px;
    }

    form input[type=submit]:focus, form input[type=button]:focus,
    form input[type=reset]:focus {
        border: 2px solid grey;
    }

    form input::-moz-focus-inner {
        border: 0;
    }

    form legend {
        font-weight: bold;
    }

    form select {
        border-radius: 5px;
    }

    form select optgroup {
        background-color: yellow;
    }

    form select optgroup option {
        background-color: greenyellow;
    }

    form fieldset {
        border: 1px solid black;
    }
</style>
</head>
<body>

```

```

<main>
  <form method=post>
    <fieldset>
      <legend>Personal data</legend>
      <label for=first_name>First name:</label>
      <input id=first_name list=names name=first_name placeholder=Pitty required
autofocus>
      <datalist id=names>
        <option>Donald</option>
        <option>Mickey</option>
      </datalist><br>
      <label>Last name:</label>
      <input name=last_name required><br>
      <label>Password:</label>
      <!-- oncopy, onpaste and oncut are not part of the HTML5 standard,
so they should not be used! -->
      <input type=password name=password required oncopy='return false;'
        onpaste='return false;' oncut='return false;'><br>
      <label>Email:</label>
      <input type=email name=email required><br>
      <label></label> <!-- Just used for layout purposes. -->
      <input type=radio name=sex value=male checked>Male<br>
      <label></label>
      <input type=radio name=sex value=female>Female<br>
      <label></label>
      <input type=checkbox name=bike value=Bike>I have a bike<br>
      <label></label>
      <input type=checkbox name=car value=Car>I have a car
    </fieldset>
    <fieldset>
      <legend>Other data</legend>
      <select>
        <optgroup label="Swedish Cars">
          <option value=volvo>Volvo</option>
          <option value=saab selected>Saab</option>
        </optgroup>
        <optgroup label="German Cars">
          <option value=mercedes>Mercedes</option>
          <option value=audi>Audi</option>
        </optgroup>
      </select>
      <textarea rows=3 cols=50 name=my_area maxlength=500
        placeholder="Short description of yourself (max. 500 chars)">
      </textarea><br>
      <label>Color:</label>
      <input type=color name=color><br>
      <label>Number (1-10):</label>
      <input type=number name=number min=1 max=10><br>
      <label>Number range:</label>
      <input type=range name=range min=1 max=100 value=50 step=5><br>
    </fieldset>
  </form>

```

```

        <input type=submit value=Submit>
        <input type=reset>
    </form>
    <form oninput="x.value = parseInt(a.value) + parseInt(b.value)">0
        <input type=range id=a value=50>
        + <input type=number id=b value=50>
        =
        <output name=x for="a b"></output>
    </form>
</main>
</body>
</html>

```

## <input>

From [www.w3.org/TR/html51/semantics.html#the-input-element](http://www.w3.org/TR/html51/semantics.html#the-input-element):

The **input** element represents a typed data field, usually with a form control to allow the user to edit the data.

Excellent examples can be found at [www.w3schools.com/tags/tag\\_input.asp](http://www.w3schools.com/tags/tag_input.asp).

See [www.w3schools.com/html/html\\_form\\_attributes.asp](http://www.w3schools.com/html/html_form_attributes.asp) for details on input attributes.

To capture photos using the user's camera you can use the **capture** attribute, like so:

[students.btsi.lu/evegi144/WAD/HTML5/forms4.html](http://students.btsi.lu/evegi144/WAD/HTML5/forms4.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Camera capture example</title>
  </head>
  <body>
    <form>
      <input type=file accept=image/* capture>
    </form>
  </body>
</html>

```

Please note that you can prevent the copying, pasting and cutting of input field contents by setting the corresponding event handlers (cf. [Events](#)) as shown in the password input in the example above.

## Input validation with regular expressions

[www.htmlgoodies.com/html5/markup/telerik-takes-cross-platform-mobile-development-to-the-next-level-for-.net-developers-with-latest-telerik-devcraft-release.html#fbid=WNBB-6q5PXU](http://www.htmlgoodies.com/html5/markup/telerik-takes-cross-platform-mobile-development-to-the-next-level-for-.net-developers-with-latest-telerik-devcraft-release.html#fbid=WNBB-6q5PXU)

[www.htmlgoodies.com/html5/javascript/understanding-how-html5-validation-works.html#fbid=pLIOaPZ67cu](http://www.htmlgoodies.com/html5/javascript/understanding-how-html5-validation-works.html#fbid=pLIOaPZ67cu)

`<textarea>`

From [www.w3.org/TR/html51/semantics.html#the-textarea-element](http://www.w3.org/TR/html51/semantics.html#the-textarea-element):

The `textarea` element represents a multiline plain text edit control for the element's raw value. The contents of the control represent the control's default value.

The raw value of a `<textarea>` control must be initially the empty string.

For examples see [www.w3schools.com/tags/tag\\_textarea.asp](http://www.w3schools.com/tags/tag_textarea.asp).

`<label>`

From [www.w3.org/TR/html51/semantics.html#the-label-element](http://www.w3.org/TR/html51/semantics.html#the-label-element):

The `label` element represents a caption in a user interface. The caption can be associated with a specific form control, known as the label element's labeled control, either using the `for` attribute, or by putting the form control inside the label element itself.

For examples see [www.w3schools.com/tags/tag\\_label.asp](http://www.w3schools.com/tags/tag_label.asp).

`<fieldset>`

From [www.w3.org/TR/html51/semantics.html#the-fieldset-element](http://www.w3.org/TR/html51/semantics.html#the-fieldset-element):

The `fieldset` element represents a set of form controls optionally grouped under a common name.

The name of the group is given by the first `legend` element that is a child of the `fieldset` element, if any. The remainder of the descendants form the group.

For examples see [www.w3schools.com/tags/tag\\_fieldset.asp](http://www.w3schools.com/tags/tag_fieldset.asp).

`<legend>`

From [www.w3.org/TR/html51/semantics.html#the-legend-element](http://www.w3.org/TR/html51/semantics.html#the-legend-element):

The `legend` element represents a caption for the rest of the contents of the `legend` element's parent `fieldset` element, if any.

For examples see [www.w3schools.com/tags/tag\\_legend.asp](http://www.w3schools.com/tags/tag_legend.asp).

## <select>

From [www.w3.org/TR/html51/semantics.html#the-select-element](http://www.w3.org/TR/html51/semantics.html#the-select-element):

The **select** element represents a control for selecting amongst a set of options.

The **multiple** attribute is a boolean attribute. If the attribute is present, then the **select** element represents a control for selecting zero or more options from the list of options. If the attribute is absent, then the select element represents a control for selecting a single option from the list of options.

The size attribute gives the number of options to show to the user. The size attribute, if specified, must have a value that is a valid non-negative integer greater than zero.

The display size of a **select** element is the result of applying the rules for parsing non-negative integers to the value of element's **size** attribute, if it has one and parsing it is successful. If applying those rules to the attribute's value is not successful, or if the **size** attribute is absent, then the element's display size is 4 if the element's **multiple** content attribute is present, and 1 otherwise.

The list of options for a **select** element consists of all the **option** element children of the **select** element, and all the **option** element children of all the **optgroup** element children of the **select** element, in tree order.

For examples see [www.w3schools.com/tags/tag\\_select.asp](http://www.w3schools.com/tags/tag_select.asp).

## <optgroup>

From [www.w3.org/TR/html51/semantics.html#the-optgroup-element](http://www.w3.org/TR/html51/semantics.html#the-optgroup-element):

The **optgroup** element represents a group of **option** elements with a common label.

For examples see [www.w3schools.com/tags/tag\\_optgroup.asp](http://www.w3schools.com/tags/tag_optgroup.asp).

## <option>

From [www.w3.org/TR/html51/semantics.html#the-option-element](http://www.w3.org/TR/html51/semantics.html#the-option-element):

The **option** element represents an option in a **select** element or as part of a list of suggestions in a **datalist** element.

For examples see [www.w3schools.com/tags/tag\\_option.asp](http://www.w3schools.com/tags/tag_option.asp).

### <button>

From [www.w3.org/TR/html51/semantics.html#the-button-element](http://www.w3.org/TR/html51/semantics.html#the-button-element):

The **button** element represents a button labeled by its contents.

For examples see [www.w3schools.com/tags/tag\\_button.asp](http://www.w3schools.com/tags/tag_button.asp).

See [developer.mozilla.org/en-US/docs/Web/HTML/Element/button](http://developer.mozilla.org/en-US/docs/Web/HTML/Element/button) for interesting details.

### <datalist>

From [www.w3.org/TR/html51/semantics.html#the-datalist-element](http://www.w3.org/TR/html51/semantics.html#the-datalist-element):

The **datalist** element represents a set of option elements that represent predefined options for other controls.

For examples see [www.w3schools.com/tags/tag\\_datalist.asp](http://www.w3schools.com/tags/tag_datalist.asp).

### <keygen>

From [www.w3.org/TR/html51/semantics.html#the-keygen-element](http://www.w3.org/TR/html51/semantics.html#the-keygen-element):

The **keygen** element represents a key pair generator control. When the control's form is submitted, the private key is stored in the local keystore, and the public key is packaged and sent to the server.

For examples see [www.w3schools.com/tags/tag\\_keygen.asp](http://www.w3schools.com/tags/tag_keygen.asp).

### <output>

From [www.w3.org/TR/html51/semantics.html#the-output-element](http://www.w3.org/TR/html51/semantics.html#the-output-element):

The **output** element represents the result of a calculation or user action.

For examples see [www.w3schools.com/tags/tag\\_output.asp](http://www.w3schools.com/tags/tag_output.asp).

## 4.1.27. Block vs inline elements

From [www.w3schools.com/css/css\\_display\\_visibility.asp](http://www.w3schools.com/css/css_display_visibility.asp):

A block element is an element that takes up the full width available, and has a line break before and after it.

Examples of block elements:

- `<h1>`
- `<p>`
- `<div>`

An inline element only takes up as much width as necessary, and does not force line breaks.

Examples of inline elements:

- `<span>`
- `<a>`

**This is a heading level 1, a block element.**

This is a `<span>` which is an **inline** element.

Sometimes it is useful to change the default display settings. For example, we might want list items to be stacked horizontally instead of vertically. So instead of this:

- Item1
- Item2
- Item3
- Item4

We might prefer this:

Item1 Item2 Item3 Item4

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Inline vs block display example</title>
    <meta charset=utf-8>
    <style>
      main {
        border: 2px solid red;
        padding: 5px;
      }

      h1 {
        border: 1px inset blue;
        padding: 3px;
      }

      span {
        border: 1px dashed green;
      }

      .inlineList {
        display: inline;
      }
    </style>
  </head>
  <body>
    <main>
      <h1>This is a heading level 1, a <strong>block</strong> element.</h1>
      This is a <span>span</span>, which is an <strong>inline</strong> element.
      <p>Sometimes it is useful to change the default display settings. For
      example, we might want list items to be stacked horizontally instead of
      vertically. So instead of this:
      </p>
      <ul>
        <li>Item1</li>
        <li>Item2</li>
        <li>Item3</li>
        <li>Item4</li>
      </ul>
      We might prefer this:
      <ul>
        <li class=inlineList>Item1</li>
        <li class=inlineList>Item2</li>
        <li class=inlineList>Item3</li>
        <li class=inlineList>Item4</li>
      </ul>
    </main>
  </body>
</html>
```

A complete list of block elements can be found at [https://developer.mozilla.org/en-US/docs/Web/HTML/Block-level\\_elements](https://developer.mozilla.org/en-US/docs/Web/HTML/Block-level_elements). Likewise a list of inline elements is at [developer.mozilla.org/en-US/docs/Web/HTML/Inline\\_elements](https://developer.mozilla.org/en-US/docs/Web/HTML/Inline_elements).

#### 4.1.28. <video>

The <video> tag does what its name suggests, i.e. it defines video. It supports the following particular attributes:

Name	Value	Description
autoplay		play video automatically
controls		display video controls
height	pixels	height of the video player
loop		video will loop indefinitely (if supported by browser)
muted		muted video output
poster	URL	image to be shown while video is downloading and playback has not started
preload	auto, metadata, none	how the video should be loaded (cf. <a href="http://www.w3schools.com/tags/att_audio_preload.asp">www.w3schools.com/tags/att_audio_preload.asp</a> )
src	URL	video URL
width	pixels	width of the video player

#### Video file formats

There are three video file formats currently supported. They are MP4, WebM and OGG. Here is an overview of browser support:

	MP4	WebM	OGG
Firefox	yes	yes	yes
Chrome	yes	yes	yes
Internet Explorer	yes	no	no
Safari	yes	no	no
MIME type	video/mp4	video/webm	video/ogg

Within the <video> tag we use the <source> tag to specify multiple video sources that the browser can choose from, based on its file format support.

If we need to convert between the different video file formats we can use a number of free tools, such as Miro Video Converter ([www.mirovideoconverter.com](http://www.mirovideoconverter.com)) or Handbrake ([handbrake.fr](http://handbrake.fr)).

Here is a simple example of the `<video>` tag in action:



[students.btsi.lu/evegi144/WAD/HTML5/video1.html](http://students.btsi.lu/evegi144/WAD/HTML5/video1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Video example</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <video controls autoplay loop
        src=http://clips.vorwaerts-gmbh.de/big_buck_bunny.mp4>
        Your browser does not support the video tag.
      </video>
    </main>
  </body>
</html>
```

Text inside the `<video>` opening and closing tag will be shown in browsers that do not support the tag.

### Embedding YouTube videos

An excellent explanation can be found at [https://www.htmlgoodies.com/tutorials/web\\_graphics/article.php/3480061/How-To-Add-a-YouTube-Video-to-Your-Web-Site.htm](https://www.htmlgoodies.com/tutorials/web_graphics/article.php/3480061/How-To-Add-a-YouTube-Video-to-Your-Web-Site.htm).

#### 4.1.29. <audio>

The <audio> tag does what its name suggests, i.e. it defines sound. It supports the following particular attributes:

Name	Value	Description
autoplay		play audio automatically
controls		display audio controls
loop		audio will loop indefinitely (if supported by browser)
muted		muted audio output
preload	auto, metadata, none	how the audio should be loaded (cf. <a href="http://www.w3schools.com/tags/att_audio_preload.asp">www.w3schools.com/tags/att_audio_preload.asp</a> )
src	URL	audio URL

#### Audio file formats

There are three audio file formats currently supported. They are MP3, WAV and OGG. Here is an overview of browser support and main format features:

	MP3	OGG	WAV
Firefox	yes	yes	yes
Chrome	yes	yes	yes
Internet Explorer	yes	no	no
Safari	yes	yes	no
MIME type	audio/mpeg	audio/ogg	audio/wav
lossless	no	no	usually yes
compressed	yes	yes	usually no

If we need to convert between the different audio file formats we can use a number of free tools, such as Audacity ([audacity.sourceforge.net](http://audacity.sourceforge.net)). To produce our own music, we can use the outstanding LMMS ([lmms.sourceforge.net](http://lmms.sourceforge.net)) or AudioTool ([www.audiotool.com](http://www.audiotool.com)).

Here is a simple example of the <audio> tag in action:

[students.btsi.lu/evegi144/WAD/HTML5/audio1.html](http://students.btsi.lu/evegi144/WAD/HTML5/audio1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Audio example</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <audio controls autoplay loop>
        <source src=http://api.audiotool.com/track/ge_trance_1/9/mixdown.ogg
              type=audio/ogg>
        <source src=http://api.audiotool.com/track/ge_trance_1/9/mixdown.mp3
              type=audio/mpeg>
        Your browser does not support the audio tag.
      </audio>
    </main>
  </body>
</html>

```

Note that the browser will use the first file format that it supports from those listed. The current versions of the main browsers all support MP3. If we only want to support those we can omit the `<source>` tag:

[students.btsi.lu/evegi144/WAD/HTML5/audio2.html](http://students.btsi.lu/evegi144/WAD/HTML5/audio2.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Audio example</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <audio controls autoplay loop
        src=http://api.audiotool.com/track/ge_trance_1/9/mixdown.mp3>
        Your browser does not support the audio tag.
      </audio>
    </main>
  </body>
</html>

```

Text inside the `<audio>` opening and closing tag will be shown in browsers that do not support the tag.

#### 4.1.30. Additional elements

## <pre>

The `<pre>` tag defines preformatted text. Text is displayed in a fixed-width font and preserves both spaces and line breaks (cf. [www.w3schools.com/tags/tag\\_pre.asp](http://www.w3schools.com/tags/tag_pre.asp)).

Example:

```
This    is          preformatted text.  
Spaces and linebreaks are preserved.  
The font used by the browser has fixed width.
```

[students.btsi.lu/evegi144/WAD/HTML5/pre1.html](http://students.btsi.lu/evegi144/WAD/HTML5/pre1.html)

```
<!DOCTYPE html>  
<html lang=en>  
  <head>  
    <title>Preformatted Text Example</title>  
    <meta charset=UTF-8>  
  </head>  
  <body>  
    <pre>  
This    is          preformatted text.  
Spaces and linebreaks are preserved.  
The font used by the browser has fixed width.  
    </pre>  
  </body>  
</html>
```

## <mark>

This element is used to highlight text (cf. [www.w3schools.com/tags/tag\\_mark.asp](http://www.w3schools.com/tags/tag_mark.asp)). Example:

The **mark element** is used to highlight text in HTML5.

[students.btsi.lu/evegi144/WAD/HTML5/mark1.html](http://students.btsi.lu/evegi144/WAD/HTML5/mark1.html)

```
<!DOCTYPE html>  
<html lang=en>  
  <head>  
    <title>Mark example</title>  
    <meta charset=UTF-8>  
  </head>  
  <body>  
    <main>  
      <p>The <mark>mark element</mark> is used to highlight text in HTML5.</p>  
    </main>  
  </body>  
</html>
```

## <address>

This element is used to display contact information for a person, people or organization. It should include physical and/or digital location/contact information and a means of identifying a person(s) or organization the information pertains to. (cf. [www.w3.org/TR/html5/grouping-content.html#elementdef-address](http://www.w3.org/TR/html5/grouping-content.html#elementdef-address)).

Example:

[Gilles Everling](mailto:gilles.everling@education.lu)

[students.btsi.lu/evegi144/WAD/HTML5/address1.html](http://students.btsi.lu/evegi144/WAD/HTML5/address1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Address example</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <footer>
      <address>
        <a href=mailto:gilles.everling@education.lu>Gilles Everling</a>
      </address>
    </footer>
  </body>
</html>
```

## <time>

## <ins>

## <del>

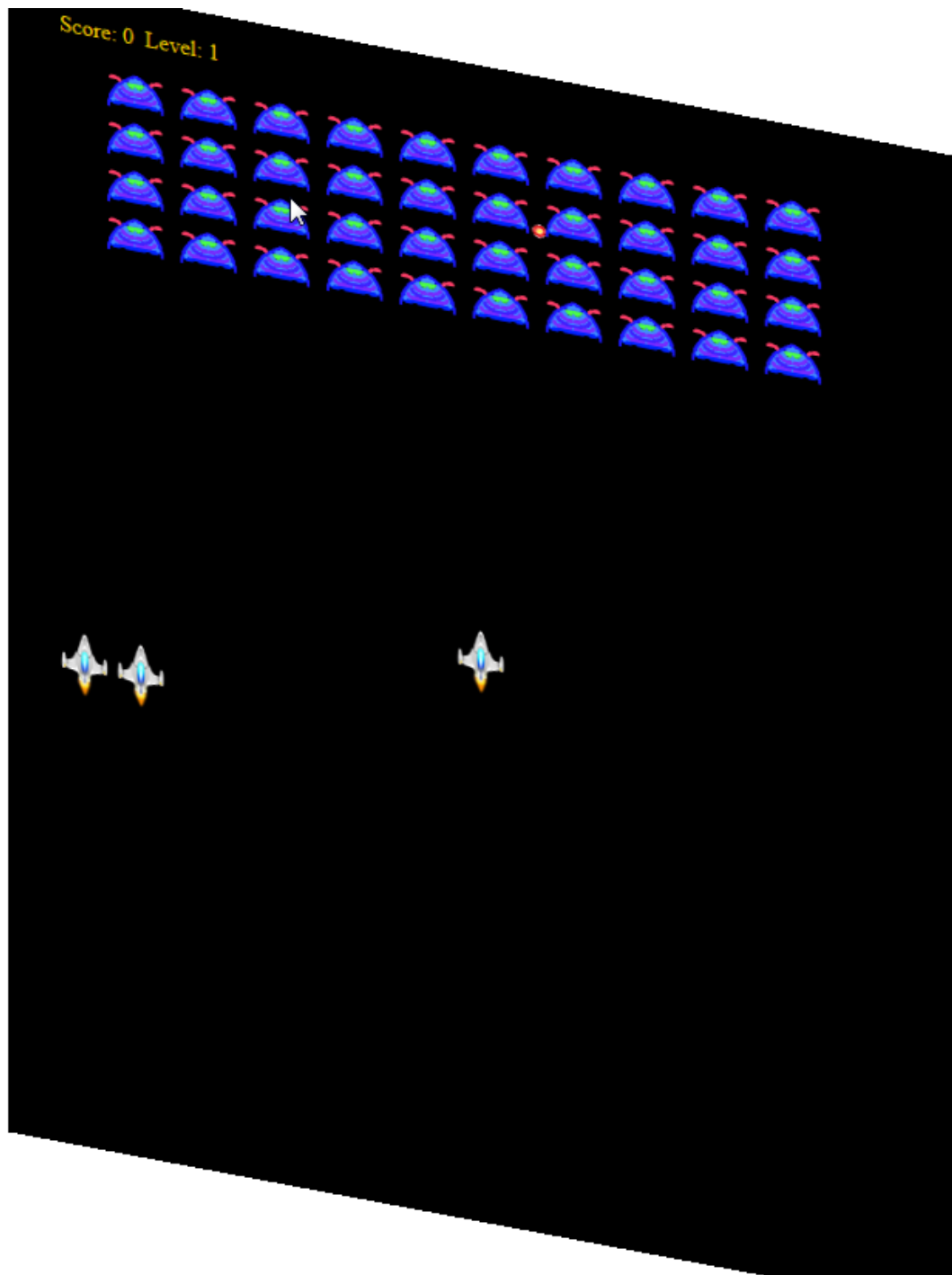
## <iframe>

[www.w3schools.com/tags/tag\\_iframe.asp](http://www.w3schools.com/tags/tag_iframe.asp)

[www.w3.org/TR/html5/embedded-content-0.html#the-iframe-element](http://www.w3.org/TR/html5/embedded-content-0.html#the-iframe-element)

[www.w3.org/wiki/HTML/Elements/iframe](http://www.w3.org/wiki/HTML/Elements/iframe)

WMOTU  
Invaders  
WMOTU  
Quack



[students.btsi.lu/evegi144/WAD/HTML5/iframe1.html](https://students.btsi.lu/evegi144/WAD/HTML5/iframe1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>iframe example</title>
    <meta charset=utf-8>
    <style>
      nav {
        position:      fixed;
        left:           0;
        top:            0;
        bottom:         0;
        width:          100px;
```

```

    background-color: lightgoldenrodyellow;
}

main {
    position:      absolute;
    left:          100px;
    top:           0;
    bottom:        0;
    right:         0;
    background-color: lightblue;
    animation:      animate 60s linear 5s infinite;
}

iframe {
    display: block;
    width:  100%;
    height: 100%;
    border: none;
}

@keyframes animate {
    to {
        transform: rotateX(360deg) rotateY(360deg);
    }
}

ul {
    list-style-type: none;
    padding-left:    10px;
}

li {
    padding-top: 5px;
}
</style>
</head>
<body>
    <nav>
        <ul>
            <li><a href=https://students.btsi.lu/evegi144/WAD/WMOTUInvaders00
                target=myFrame>WMOTU Invaders</a></li>
            <li><a href=https://students.btsi.lu/evegi144/WAD/WMOTUQuack
                target=myFrame>WMOTU Quack</a></li>
        </ul>
    </nav>
    <main>
        <iframe name=myFrame></iframe>
    </main>
</body>
</html>

```



[students.btsi.lu/evegi144/WAD/HTML5/iframe2.html](https://students.btsi.lu/evegi144/WAD/HTML5/iframe2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title></title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <iframe width=560 height=315 src=https://www.youtube.com/embed/-H2x_tGAxSM?rel=0
        allowfullscreen></iframe>
    </main>
  </body>
</html>
```

#### 4.1.31. **<embed>**

This tag creates a container for a plug-in, such as Adobe Reader:

12e Sem1

### Allgemeine Anmerkungen

- In diesem Modul geht es hauptsächlich um PHP, was aber nicht b vernachlässigt werden sollen. Größere JS Skripte sind aber zu ve
- Die Schüler benutzen oder erstellen ein „PHP Cheat Sheet“.
- Der Code muss kommentiert, leserlich und sauber formatiert sein

### Lehrplan

[students.btsi.lu/evegi144/WAD/HTML5/embed1.html](https://students.btsi.lu/evegi144/WAD/HTML5/embed1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Embed example</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <embed src=T-IF-WEB2-WSERS1_LP.pdf width=1000 height=800>
    </main>
  </body>
</html>
```

#### 4.1.32. Global attributes

[www.w3schools.com/tags/ref\\_standardattributes.asp](http://www.w3schools.com/tags/ref_standardattributes.asp)

#### 4.1.33. Event attributes

[www.w3schools.com/tags/ref\\_eventattributes.asp](http://www.w3schools.com/tags/ref_eventattributes.asp)

#### 4.1.34. Copyright

[creativecommons.org](http://creativecommons.org)

#### 4.1.35. Quiz

Take the w3schools quiz at [www.w3schools.com/quiztest/quiztest.asp?qtest=HTML5](http://www.w3schools.com/quiztest/quiztest.asp?qtest=HTML5) as a fun way to check you are as good as you think you are.

## 4.2. CSS3

From [www.w3.org/TR/CSS21](http://www.w3.org/TR/CSS21):

CSS is a style sheet language that allows authors and users to attach style (e.g., fonts and spacing) to structured documents (e.g., HTML documents and XML applications). By separating the presentation style of documents from the content of documents, CSS simplifies Web authoring and site maintenance.

The official CSS3 specifications can be found at [www.w3.org/Style/CSS/current-work](http://www.w3.org/Style/CSS/current-work).

What can be done with CSS? [Here](#)'s an example.

### 4.2.1. Include CSS3

There are four ways we can style our web page with CSS3.

#### Inline

We use the `style` attribute of a specific element to style it.

Example:

[students.btsi.lu/evegi144/WAD/CSS3/inline1.html](https://students.btsi.lu/evegi144/WAD/CSS3/inline1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Inline styling with CSS3</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main style='background-color: gold; color: black'>
      This element is style with inline CSS3.
    </main>
  </body>
</html>
```

#### Embedded

We use the `<style>` tag to embed CSS in the head of our document.

Example:

[students.btsi.lu/evegi144/WAD/CSS3/embedded1.html](https://students.btsi.lu/evegi144/WAD/CSS3/embedded1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Embedded CSS3</title>
    <meta charset=utf-8>
    <style>
      main {
        background-color: lightcoral;
      }

      p {
        border: 2px dashed green;
        color: lime;
      }
    </style>
  </head>
  <body>
    <main>
      <p>All elements styled by embedded CSS3.</p>
    </main>
  </body>
</html>
```

## External

We use the `<link>` tag to include an external CSS style sheet in the head of our document.

Example:

[students.btsi.lu/evegi144/WAD/CSS3/external1.html](https://students.btsi.lu/evegi144/WAD/CSS3/external1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>External CSS3 style sheet</title>
    <meta charset=utf-8>
    <link href=external1.css rel=stylesheet>
  </head>
  <body>
    <main>
      <p>All elements styled via an external CSS3 style sheet.</p>
    </main>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/CSS3/external1.css](https://students.btsi.lu/evegi144/WAD/CSS3/external1.css)

```
main {
  background-color: lightcoral;
}

p {
  border: 2px dashed green;
  color: lime;
}
```

## Imported

We use the `@import` rule to include an external CSS style sheet in the current style sheet. See [developer.mozilla.org/en-US/docs/Web/CSS/@import](https://developer.mozilla.org/en-US/docs/Web/CSS/@import) and [www.w3.org/TR/CSS2/cascade.html#import](http://www.w3.org/TR/CSS2/cascade.html#import).

Example:

[students.btsi.lu/evegi144/WAD/CSS3/imported1.html](http://students.btsi.lu/evegi144/WAD/CSS3/imported1.html)

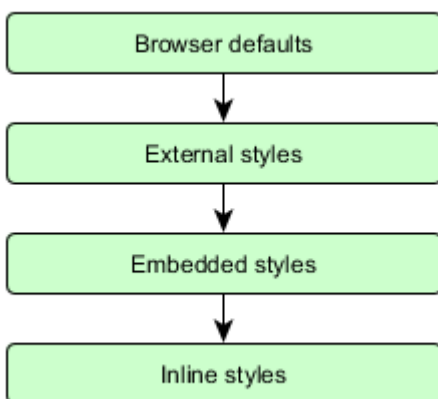
```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Imported CSS3 style sheet</title>
    <meta charset=utf-8>
    <style>
      @import url('external1.css');
      aside {
        padding: 10px;
        background-color: mediumorchid;
      }
    </style>
  </head>
  <body>
    <main>
      <p>Styled via an imported CSS3 style sheet.</p>
      <aside>Styled via embedded CSS.</aside>
    </main>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/CSS3/external1.css](http://students.btsi.lu/evegi144/WAD/CSS3/external1.css)

```
main {  
  background-color: lightcoral;  
}  
  
p {  
  border: 2px dashed green;  
  color: lime;  
}
```

## Rules of precedence

A browser processes styles in the order described at [www.w3.org/TR/css3-cascade/#cascading-origins](http://www.w3.org/TR/css3-cascade/#cascading-origins). A somewhat simplified illustration of the cascade looks like this:



However, we can influence rule precedence by either changing the order of inclusion or by using the **!important** annotation.

If we include an external style sheet after the declaration of the embedded style sheet in the head of our document, the external styles will overwrite the embedded ones.

In the following example, the paragraph background color would normally be green, given that inline styling takes precedence over embedded styling. However, the **!important** annotation gives the embedded style a higher priority:

[students.btsi.lu/evegi144/WAD/CSS3/important1.html](http://students.btsi.lu/evegi144/WAD/CSS3/important1.html)

```

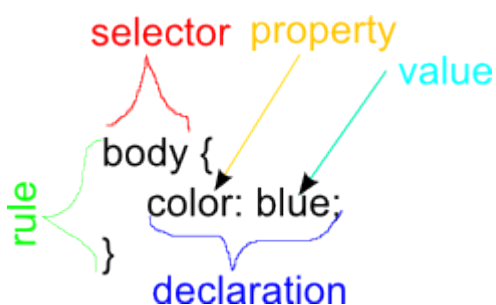
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Changing the cascade with !important</title>
    <meta charset=utf-8>
    <style>
      p {
        background-color: gold !important;
      }
    </style>
  </head>
  <body>
    <main>
      <p style='background-color: green;'>Some text</p>
    </main>
  </body>
</html>

```

You can find the default style sheets used by the main browsers at the following links (cf. [stackoverflow.com/questions/6867254/browsers-default-css-for-html-elements](https://stackoverflow.com/questions/6867254/browsers-default-css-for-html-elements)):

Firefox	<a href="https://mxr.mozilla.org/mozilla-central/source/layout/style/html.css">mxr.mozilla.org/mozilla-central/source/layout/style/html.css</a>
Internet Explorer	<a href="http://www.iecss.com">www.iecss.com</a>
Chrome	<a href="https://trac.webkit.org/browser/trunk/Source/WebCore/css/html.css">trac.webkit.org/browser/trunk/Source/WebCore/css/html.css</a>
Opera	<a href="http://www.iecss.com/opera-10.51.css">www.iecss.com/opera-10.51.css</a>
HTML5 recommendation	<a href="http://www.w3.org/TR/html5/rendering.html">www.w3.org/TR/html5/rendering.html</a>

#### 4.2.2. Syntax



This site is a CSS3 learner's paradise: [www.w3schools.com/css/css\\_examples.asp](http://www.w3schools.com/css/css_examples.asp)

#### Comments

You can comment out parts of a line or even several lines by enclosing them between `/*` and `*/`:

[students.btsi.lu/evegi144/WAD/CSS3/comment1.html](https://students.btsi.lu/evegi144/WAD/CSS3/comment1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>CSS comment example 1</title>
    <meta charset=utf-8>
    <style>
      body {
        background-color: green;
        /*display: none;
        opacity: 0.5;*/
      }
    </style>
  </head>
  <body>
  </body>
</html>
```

Comments cannot be nested, i.e. this won't work:

[students.btsi.lu/evegi144/WAD/CSS3/comment2.html](https://students.btsi.lu/evegi144/WAD/CSS3/comment2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>CSS comment example 1</title>
    <meta charset=utf-8>
    <style>
      body {
        background-color: green;
        /*display: /*block*/none;
        opacity: 0.5;*/
      }
    </style>
  </head>
  <body>
  </body>
</html>
```

## Variables

Variables are available since Firefox 29 and enabled by default starting in Firefox 31. Here are the details: [developer.mozilla.org/en-US/docs/Web/CSS/Using\\_CSS\\_variables](https://developer.mozilla.org/en-US/docs/Web/CSS/Using_CSS_variables)

### 4.2.3. Units

[www.w3schools.com/cssref/css\\_units.asp](http://www.w3schools.com/cssref/css_units.asp)

## 4.2.4. Properties



At [www.w3schools.com/cssref/default.asp](http://www.w3schools.com/cssref/default.asp) you can find easy to understand explanations as well as examples for pretty much every CSS property. This is an extremely useful resource that you should use.

[www.w3.org/TR/css-2010/#properties](http://www.w3.org/TR/css-2010/#properties) and [meiert.com/en/indices/css-properties](http://meiert.com/en/indices/css-properties) provide an almost complete list of CSS properties with links to the specifications and in-depth explanations, which are useful for WMOTUs.

### object-fit

Used to specify how an `<img>` or `<video>` should be resized to fit its container. See [www.w3schools.com/css/css3\\_object-fit.asp](http://www.w3schools.com/css/css3_object-fit.asp). Also take a look at [object-position](#).

### overflow

To disable scrolling on mobile devices, see [stackoverflow.com/questions/10592411/disable-scrolling-in-all-mobile-devices](http://stackoverflow.com/questions/10592411/disable-scrolling-in-all-mobile-devices).

### background

See [www.w3schools.com/cssref/css3\\_pr\\_background.asp](http://www.w3schools.com/cssref/css3_pr_background.asp). For multiple backgrounds, see [www.css3.info/preview/multiple-backgrounds](http://www.css3.info/preview/multiple-backgrounds).

Scaling an image to its maximum displayable size without distorting the proportions can be done like this:



[students.btsi.lu/evegi144/WAD/CSS3/backgroundmaxsize1.html](http://students.btsi.lu/evegi144/WAD/CSS3/backgroundmaxsize1.html)


```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Maximum size background image scaling without distortion</title>
    <meta charset=utf-8>
    <style>
      main {
        position: fixed;
        top: 0;
        left: 0;
        width: 100%;
        height: 100%;
        background-image: url(DSC00538.JPG);
        background-repeat: no-repeat;
        background-size: contain;
      }
    </style>
  </head>
  <body>
    <main>
    </main>
  </body>
</html>

```

## content

A great list of Unicode symbols can be found at [inamidst.com/stuff/unidata](http://inamidst.com/stuff/unidata).

 Email us

[students.btsi.lu/evegi144/WAD/CSS3/content1.html](http://students.btsi.lu/evegi144/WAD/CSS3/content1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Content demo</title>
    <meta charset=UTF-8>
    <style>
      button::before {
        content: '\2709';
      }
    </style>
  </head>
  <body>
    <button> Email us</button>
  </body>
</html>

```

white-space

[www.w3schools.com/cssref/pr\\_text\\_white-space.asp](http://www.w3schools.com/cssref/pr_text_white-space.asp)

border-image

[www.w3schools.com/cssref/css3\\_pr\\_border-image.asp](http://www.w3schools.com/cssref/css3_pr_border-image.asp) border-image.com

word-wrap

See [www.w3schools.com/cssref/css3\\_pr\\_word-wrap.asp](http://www.w3schools.com/cssref/css3_pr_word-wrap.asp) and [www.w3.org/TR/css-text-3/#word-wrap](http://www.w3.org/TR/css-text-3/#word-wrap).

**word-wrap** allows us to solve some space constraint problems, as shown in this example:

## Section 1

dass

## Section 2

```
dasssss
ssssssss
ssssssss
ssssssss
ssssssss
sssss
```

[students.btsi.lu/evegi144/WAD/CSS3/word-wrap1.html](https://students.btsi.lu/evegi144/WAD/CSS3/word-wrap1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Word wrap example 1</title>
    <meta charset=utf-8>
    <style>
      section {
        width: 50px;
        background-color: blueviolet;
      }

      #section2 {
        word-wrap: break-word;
      }
    </style>
  </head>
  <body>
    <main>
      <section>
        <h1>Section 1</h1>
        dasssssssssssssssssssssssssssssssssssssssssssssssss
      </section>
      <hr>
      <section id=section2>
        <h1>Section 2</h1>
        dasssssssssssssssssssssssssssssssssssssssssssssss
      </section>
    </main>
  </body>
</html>

```

## word-break

In a table you might have to use **word-break** instead of **word-wrap** and set it to **break-word** or **break-all**. You also must set the **width** or **max-width** of the cell for this to work.

See [developer.mozilla.org/en-US/docs/Web/CSS/word-break](https://developer.mozilla.org/en-US/docs/Web/CSS/word-break) and [css-tricks.com/where-lines-break-is-complicated-heres-all-the-related-css-and-html](https://css-tricks.com/where-lines-break-is-complicated-heres-all-the-related-css-and-html).

## hyphens

[developer.mozilla.org/en-US/docs/Web/CSS/hyphens](https://developer.mozilla.org/en-US/docs/Web/CSS/hyphens)

## box-sizing

[www.w3schools.com/cssref/css3\\_pr\\_box-sizing.asp](http://www.w3schools.com/cssref/css3_pr_box-sizing.asp)

## contenteditable

From [www.w3schools.com/tags/att\\_global\\_contenteditable.asp](http://www.w3schools.com/tags/att_global_contenteditable.asp):

The `contenteditable` attribute specifies whether the content of an element is editable or not.

### resize

The `resize` ([www.cssportal.com/css-properties/resize.php](http://www.cssportal.com/css-properties/resize.php) and [developer.mozilla.org/en-US/docs/Web/CSS/resize](http://developer.mozilla.org/en-US/docs/Web/CSS/resize)) property allows us to make an element resizable. Note that in order for the resizing to work, you need to set the `overflow` property to something different from the standard `visible`.

## 4.2.5. Selectors



Study the excellent table at [www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp) to learn the power and expressiveness of CSS selectors.

Here is a simple example:

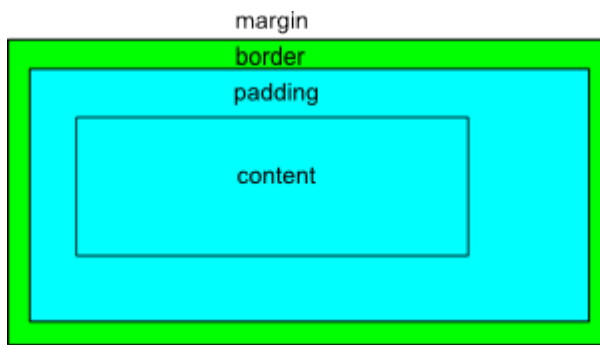


[students.btsi.lu/evegi144/WAD/CSS3/hover1.html](http://students.btsi.lu/evegi144/WAD/CSS3/hover1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Hover demo</title>
    <meta charset=utf-8>
    <style>
      p {
        display: none;
      }

      main {
        width: 500px;
        height: 500px;
        background-color: black;
      }
      main:hover > p {
        background-color: red;
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <main>
      <p>Paragraph</p>
    </main>
  </body>
</html>
```

#### 4.2.6. Box model



Cf. [www.w3schools.com/css/css\\_boxmodel.asp](http://www.w3schools.com/css/css_boxmodel.asp).



[students.btsi.lu/evegi144/WAD/CSS3/boxmodel1.html](http://students.btsi.lu/evegi144/WAD/CSS3/boxmodel1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>CSS3 box model illustration</title>
    <meta charset=utf-8>
    <style>
      body {
        background-color: black;
      }

      header {
        background-color: green;
        border: 20px groove red;
        text-align: center;
      }

      h1 {
        background-color: gold;
        margin: 50px;
        padding: 30px;
        border: 10px dotted steelblue;
      }

      section {
        background-color: khaki;
        text-align: center;
        margin: 20px;
        padding: 40px;
        border: 10px double blue;
      }
    </style>
  </head>
  <body>
    <header>
      This is the header.
      <h1>This is the heading (h1).</h1>
    </header>
    <main>
      <section>
        This is a section.
      </section>
    </main>
  </body>
</html>

```

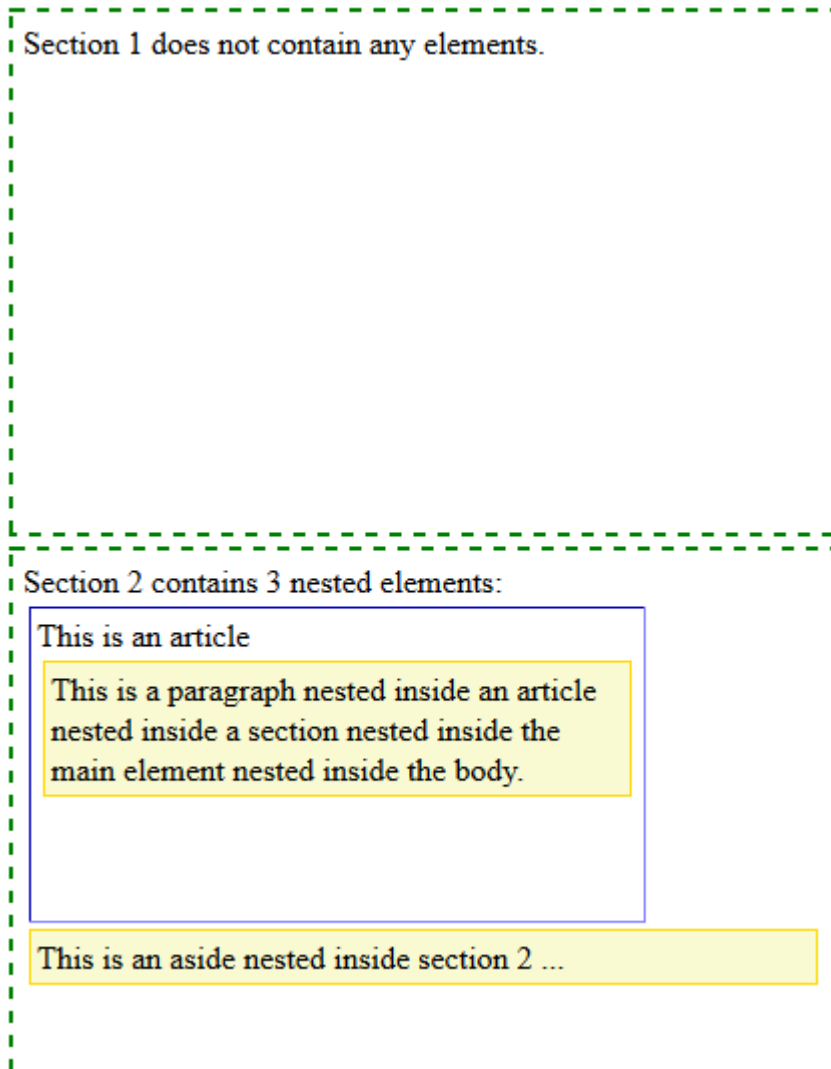


By default, i.e. if you have not changed **box-sizing**, the **width** and **height** properties in CSS3 do NOT include the padding, border and margin! Thus, if an element has content width of 200px, padding of 5px, a border width of 2px and a margin of 10px, the real width of the element will be  $200 + 2 * 5 + 2 * 2 + 2 * 10 = 234\text{px}$ .

## 4.2.7. Layout

### Normal flow

A browser renders our HTML code line by line in the order it appears in our HTML document. This is called **normal flow**. The following is a simple example, illustrating normal flow and the nesting of elements:



[students.btsi.lu/evegi144/WAD/CSS3/normalflow1.html](https://students.btsi.lu/evegi144/WAD/CSS3/normalflow1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Normal flow example 1</title>
    <meta charset=utf-8>
    <style>
      section {
        width: 400px;
        height: 250px;
        border: 2px dashed green;
        margin: 5px;
        padding: 5px;
      }

      article {
        width: 300px;
        height: 150px;
        border: 1px inset blue;
        margin: 3px;
        padding: 3px;
      }

      p, aside {
        background-color: lightgoldenrodyellow;
        border: 1px groove gold;
        margin: 3px;
        padding: 3px;
      }
    </style>
  </head>
  <body>
    <main>
      <section>
        Section 1 does not contain any elements.
      </section>
      <section>
        Section 2 contains 3 nested elements:
        <article>
          This is an article
          <p>This is a paragraph nested inside an article nested inside a
            section nested inside the main element nested inside the body.</p>
        </article>
        <aside>This is an aside nested inside section 2 ...</aside>
      </section>
    </main>
  </body>
</html>

```

Now let's try to build the following layout:

# Header

Navigation

Section 1

Section 2

Footer

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Normal flow example 2</title>
    <meta charset=utf-8>
    <style>
      nav {
        border: 2px dotted blueviolet;
        width: 10%;
        height: 80px;
        margin: 5px;
        padding: 5px;
      }

      section {
        width: 40%;
        border: 2px solid #90ff65;
        margin: 5px;
        padding: 5px;
      }

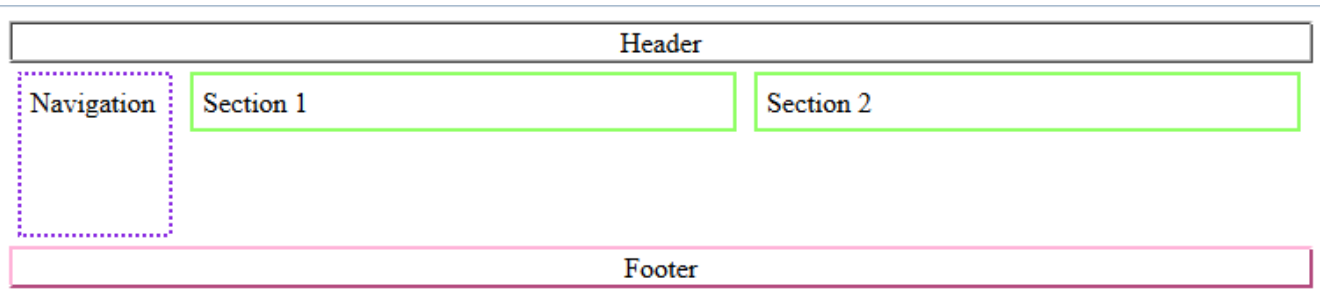
      header {
        border: 2px ridge black;
        text-align: center;
      }

      footer {
        border: 2px outset hotpink;
        text-align: center;
      }
    </style>
  </head>
  <body>
    <header>Header</header>
    <nav>Navigation</nav>
    <main>
      <section>Section 1</section>
      <section>Section 2</section>
    </main>
    <footer>Footer</footer>
  </body>
</html>
```

Unfortunately, this is not exactly what we want.

## Floats

To solve our layout problem from the previous subsection, we need to take the navigation and two section elements out of the normal flow. We can do this using the `float` and `clear` properties:



[students.btsi.lu/evegi144/WAD/CSS3/float1.html](http://students.btsi.lu/evegi144/WAD/CSS3/float1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Float example 1</title>
    <meta charset=utf-8>
    <style>
      nav {
        float: left;
        border: 2px dotted blueviolet;
        width: 10%;
        height: 80px;
        margin: 5px;
        padding: 5px;
      }

      section {
        float: left;
        width: 40%;
        border: 2px solid #90ff65;
        margin: 5px;
        padding: 5px;
      }

      header {
        border: 2px ridge black;
        text-align: center;
      }

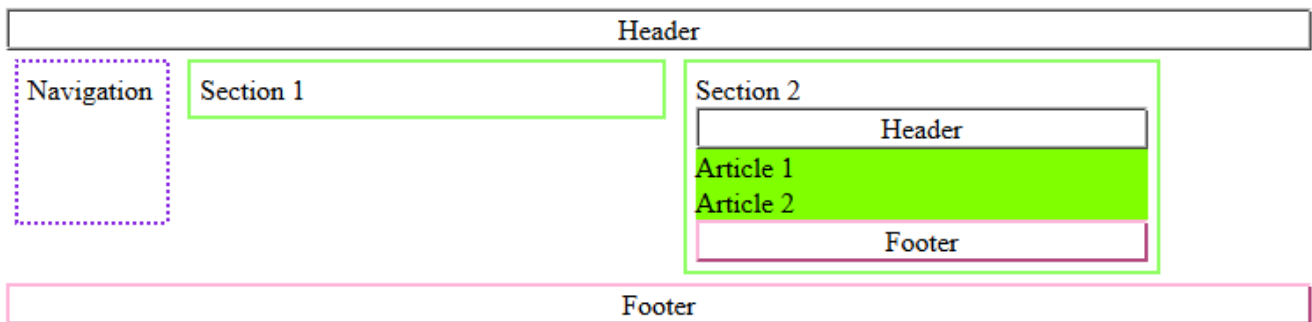
      footer {
        clear: left;
        border: 2px outset hotpink;
        text-align: center;
      }
    </style>
  </head>
  <body>
    <header>Header</header>
    <nav>Navigation</nav>
    <main>
      <section>Section 1</section>
      <section>Section 2</section>
    </main>
    <footer>Footer</footer>
  </body>
</html>

```

It is important to understand that **float** takes the floated element out of the normal flow and allows to float it on the left or right side of its container. Only block elements can be floated. To return to the normal flow, we use the **clear** property. We can clear only the left, only the right or both sides.

Remove the `clear` property from the footer rule and see what happens.

Instead of using `clear` we can use the `overflow` property on the containing element:



[students.btsi.lu/evegi144/WAD/CSS3/float2.html](https://students.btsi.lu/evegi144/WAD/CSS3/float2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Float example 2</title>
    <meta charset=utf-8>
    <style>
      nav {
        float: left;
        border: 2px dotted blueviolet;
        width: 10%;
        height: 80px;
        margin: 5px;
        padding: 5px;
      }

      section {
        float: left;
        width: 40%;
        border: 2px solid #90ff65;
        margin: 5px;
        padding: 5px;
      }

      header {
        border: 2px ridge black;
        text-align: center;
      }

      footer {
        border: 2px outset hotpink;
        text-align: center;
      }

      main {
        overflow: auto;
```

```
    }

    article {
        background-color: chartreuse;
    }
</style>
</head>
<body>
    <header>Header</header>
    <nav>Navigation</nav>
    <main>
        <section>Section 1</section>
        <section>Section 2
            <header>Header</header>
            <article>Article 1</article>
            <article>Article 2</article>
            <footer>Footer</footer>
        </section>
    </main>
    <footer>Footer</footer>
</body>
</html>
```

Floats can be very helpful in solving some simple layout problems, like in this vertical alignment example:

[students.btsi.lu/evegi144/WAD/CSS3/float3.html](https://students.btsi.lu/evegi144/WAD/CSS3/float3.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Float example 3</title>
    <meta charset=utf-8>
    <style>
      #a1 {
        float: left;
      }

      span {
        display: inline-block;
      }
    </style>
  </head>
  <body>
    <main> <!-- How do we get link1 to be aligned at the top instead of at the
bottom?-->
      <a>link1</a><span>a<br>b</span>
      <br>
      <a id=a1>link1</a><span>a<br>b</span>
    </main>
  </body>
</html>

```

## Positioning

The **position** property allows us to take an element out of the normal flow and position it exactly as we like (cf. [developer.mozilla.org/en-US/docs/Web/CSS/position](https://developer.mozilla.org/en-US/docs/Web/CSS/position)).

From [www.w3schools.com/css/css\\_positioning.asp](http://www.w3schools.com/css/css_positioning.asp):

It is important to note that an element with position **absolute** is positioned relative to the nearest positioned ancestor (instead of positioned relative to the viewport, like fixed). However; if an absolute positioned element has no positioned ancestors, it uses the document body, and moves along with page scrolling. Note: A "positioned" element is one whose position is anything except **static**.

Let's have a look at some examples:

LAM T0IF WMOTU	
Link 1	
Link 2	
Link 3	
Link 4	

Coram ipsum dolor sit amet, consectetur adipiscing elit. Ut porta tempor consequat. Donec dignissim portitor sapien nec posuere. Nam vel tortor vitae felis sagittis faucibus a sit amet massa. Praesent pellentesque tempus nisi. Nunc sed congue ipsum, sed cursus neque. Morbi semper eros a justo suscipit laoreet. In portitor egestas rutrum. Praesent et semper nisi, ut consequat sem. Donec tincidunt vestibulum bibendum. Maecenas blandit portitor rhoncus neque, et venenatis diam molestie at. Vestibulum nec tincidunt nisi. Vivamus at quam luctus, scelerisque urna in, interdum mi. Nunc ac ipsum sodales, vehicula tortor ut, blandit justo. Aliquam et lorem in sem tempor tempor nec in nulla.

I'm so sticky

Vivamus dignissim sagittis dolor, in aliquet justo ut idem. Sed magna magna idclitidum vivam commodo nec, commodo eget eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae, Integer ut neque neque. Curabitur feugiat augueam interdum. Nam sit amet orci nisl. Nulla sed augit dolor, aliquam. Phasellus sit amet nisl in dictum vehicula aliquam ac sed augue. Aenean molestie viverra tincidunt. Nulla facilis. Maecenas in faucibus orci nisl. Aliquam aliquam pulvinar posuere. Dui adipiscing malesuada luctus, id rutrum ante venenatis in. Cras laoribus euismod sagittis. Donec gravida porta curabitur. Vivamus eu pulvinar dui, id dignissim sagittis. Nulla eleifend justo ac feugiat consequat. Integer viverra felis ac orci tempus, ac convallis magna eleifend. In ullamcorper sit amet ipsum eu volutpat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae. Nunc non sagittis luctus. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut dictum purus eget risus luctus vivamus vitae ac ipsum. Etiam sollicitudin eu sapien et placerat. In viverra tincidunt erat, at ultrices dui.

Nullam sit amet mi is rursus tincidunt lobortis vitae ac. Aliquam cursus grav fames, eu lobortis velat bibendum nec. Pellentesque venenatis dui elit, et fermentum magna aliquet vel. In hac habitasse platea dictumt. Morbi malesuada sagittis dolor, sed aliquam enim lacina ut. In condimentum metus sit amet mauris eleifend sodales. Vivamus in accumsan odio. Integer egestas elit non faucibus facilisis. Fusce magna nisi. Sed lacina purus id dui accumsan ornare. Nulla in lo, mollis quis porta id, imperdiet in quam. Vivamus ullamcorper volutpat eros, in adipiscing mi aliquet sit amet. Malesuada ultrices vel ac consectetur malesuada. Pellentesque eu scelerisque nibh. Nam placera sitis a neque lacinia venenatis sed at nisi. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut diam nibh, varius vel dapibus id, pulvinar ac lue. Ut rutrum odio et ligula gravida, pharetra suscipit quam congue. Suspendisse vestibulum enim et rursus varius suscipit. Ut tristique placerat sapient et blandit. Etiam non iaculis orci, a blandit nibh. Nunc eget frugiat est. Sed nec tellus qui erat pulvinar sodales. Cras pharetra pulvinar auctor. Curabitur placerat nisi non mattis laoreet.

ad mundum et maledusa fama ac ante ipsum trinitus in faucibus. Sed ut loresum dicitur. Nulla faciliis. Duo dignissim magna vitae alacritas elefrenas. Maecenas a erat val diam portitor aliquid et non una. Nullam non odio et nibb adipiscunt tempore. Integer ut loresum lencundum, sclerisque nisi eget, faucibus diu. Pellentesque eget dolor iuris. Sed dictum auctor massa, sed venenatis loresum tristique et. Donec a metus in orci aliquam pellentesque non euismod sea. Curs tristique nisi ac et elefrenum, eget rhoncus massa euismod. Nulla faciliis. Nam sit amet sagittis auctor, gravida ubiubique odio. Donec euismod sagittis orci, non sclerisque nibb pulvinar nec. Donec vel magna euismod, tempor mauris sed, ullamcorper magna. Integer non tortor iudi voluptat alitricus qui sed augue. Vivamus condimentum nulla sit amet mattis convallis. Aenean erat orci, ultricies eget arcu, aliquet pulvinar quam. Sed rhoncus sapient a volutpat semper. Mauris hendrerit odio sit amet orci hendrerit, elementum porta sapien placierat. Aliquam erat volutpat. Nam luctus sit amet massa eget augue. Donec viverra molestie congue. Cursus eget mauris ut aliquet consequat. Curabitur ut sem et neque

malesuada molis non ut neque. Sed laetore lo et erat blandit, quis vulputate. Integer commodo est nec ullamcorper vestibulum. Donec egetis sapiens eu hendrerit tincidunt. Vestibulum nec orci eu sapien dictum interdum. Vivamus at porta libero, sit acut convallis massa. Phasellus ullamcorper fermentum nunc, et aliquam nisi pulvinar quis. Suspendisse tincidunt fermentum interdum. Maecenas eu tincidunt tortor, non ornare erat. Sed hendrerit nibi eget felis imperdiet molestie. Mauris dictum porta sagittis. Nam scelerisque ante nec luctus laetore. Suspendisse ac tellus porta, ornare una nec, ullamcorper erat. Interdum et malesuada fames ac ante ipsum primis in faucibus. Fusce lacus felis, ultrices quis nunc at, ornare pellentesque lo. Nulla quis ante vitae sem adipiscing commodo eget in ligula. Duis quis lo et erat blandit laetore. Quisque sapien luctus, tempus at justo ac, eleifend dignissim tortor. Sed ipsum tellus, ornare et felis viverra, lobortis dignissim lacus. Suspendisse id luctus at nunc mattis vestibulum. Donec fringilla erat malesuada, posuere massa eu, pretium metus. Phasellus imperdiet, diam lo tincidunt egetas, et dolo semper orci, at

consecrator dicit dula d'veta. Etiam volutpat fugiat aliquam. Proin euismod imperet, sit pellentesque velit vestibulum eu. Etiam dui erat, molestie sit amet tuncidunt virga, facillisi ad dolor. Duis vehicula nulla non dui sagittis tempus. Duis posuere passere et lacinia accusam. Suspendisse potenti. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi ac rutrum sapien. Aliquam erat volutpat. Vivamus at dignissim metus. Maecenas scelerisque turpis diam. Praesent tuncidunt posuere arcu, sed laculisi m' vehicula sit amet. Phasellus eu risus enim. Interdum et malesuada fames ac ante ipsum primis in faucibus. Cras congue turpis non metus volutpat, venenatis condignum mauris pellentesque. Curabitur nec fringilla arcu. Donec at lectis vitae magna pulvinar curis. Mauris elefendit lectus eget diam tristique, sit amet posuere risus varius. Vestibulum et ante nulla feugiat ultricies. Ut leo tellus, vestibulum et magna sit amet, interdum tuncidunt ligula. Ut malesuada tortor nisi, eu imperdiet ligula convallis a. Donec et dui tellus enim euismod vulputate. Sed pharetra lectus quis, porta dolor tuncidunt a. In sem eu et scelerisque dapibus. Nulla

facilis. Aliterque variis rursus ne tortor condimentum auctor. Donec fringilla faucibus. Pondum illancipari mihi a ta ducitindend henderit vite sed nulla. In porta dui qua una laoreet elementum. Namis ultrices aliquo elefrend. Etiam fringilla eret in suscipit volutpat. Integer sollicitudin voluptate orci ullamcorper luctus. In henderit augeo non aliquam ruerud. Sed porta faucibus sodales. Vivamus dignissim sagittis dolor, in aliquet justo nulla. Sed magna magna, sollicitudin vitae commodi neque, commodo eget orci. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Integer et neque neque. Curabitur fringilla accumsan interdum. Nam sit amet auctor nisi. Nulla sed auctor diam. Phasellus sit amet nisi in vehicula aliquam ac sed auge. Aenean molestie viverra tincidunt. Nulla facilis. Maecenas eu accumsan risus. Aliquam aliquam pulvinar posuere. Datis adipiscing malesuada luctus, id ruerud ante venenatis in. Cras lobortis auctor faucibus. Donec gravida porta facillit. Vivamus eu pulvinar dui, id dignissim sapien. Nulla elefrend justo ac fringit congue. Integer viverra felis ac orci tempus, ac convallis magna

delefind. In ullancorpori sit ante ipsum uel post. Vestibulum ante ipsum nunc in faucibus orci luctus et ultrices posuere cubilia Curae, Nunc non sagittis lacus. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut dictum purus eget risus luctus faucibus vitae ac ipsum. Etiam sollicitudin eu sapien et placerat. In vitae tincidunt erat, at ultrices du. Vivamus dignissim sagittis dolor, et aliquet justo auctor. Sed magna magna, sollicitudin vitae commodo nec, commodo eget eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae. Interdum e neque neque. Curabitur feugiat accumsan interdum. Nam sit ante auctor nisi. Nulla sed auctor nisi. Phasellus sit ante nisi id dui vehicula aliquam ac sed augue. Aenean molestie vitae tincidunt. Nulla facili. Maecenas ac accumsan risus. Aliquam aliquam pulvinar posuere. Duis adipiscit malesuada lacus, id rutrum ante venenatis. In Cras lobortis euismod faucibus. Donec gravida porta facili. Vivamus eu pulvinar du, id dignissim sapien. Nulla eleferend justo ac feugiat conctetur. Integer viverra felis ac occipit, ac convalis magna eleferend. In ullamcorpi sit amet

ipsum eu voluptat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Nunc non sagittis ligula. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut dictum purus eget risus luctus faucibus vitae ac ipsum. Etiam sollicitudin eu sapien et placerat. In vitae tincidunt erat, at ultrices dui. Phasellus imperdiet, diam id tincidunt egestas, est dolor semper orci, at consectetur dui diam a velit. Etiam volutpat feuziat aliquam. Proin euismod semper elit, id pellentesque velit vestibulum eu. Etiam dui erat, molestie sit amet tincidunt vitae, facilisis ac dolor. ∞

[students.btsi.lu/evegi144/WAD/CSS3/positioning1.html](https://students.btsi.lu/evegi144/WAD/CSS3/positioning1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Positioning example 1</title>
    <meta charset=UTF-8>
    <style>
      body {
        margin: 0;

      header {
        background-color: gold;
        text-align: center;

      nav {
        background-color: navajowhite;

      section {
        background-color: lightgrey;

      footer {
        background-color: aqua;
        text-align: center;

      #article1 {
        background-color: yellow;
        opacity: 0.3;
      }
    }
  }

```

```

#article2 {
    background-color: indianred;
    opacity:          0.3;
}

#article3 {
    background-color: yellowgreen;
}

#article4 {
    background-color: white;
}

aside {
    background-color: red;
}

h1 {
    margin: 0;
}

ul {
    padding: 0;
}
</style>
</head>
<body>
<header>
    <h1>LAM T0IF WMOTU</h1>
</header>
<nav>
    <ul>
        <li><a href=#>Link 1</a></li>
        <li><a href=#>Link 2</a></li>
        <li><a href=#>Link 3</a></li>
        <li><a href=#>Link 4</a></li>
    </ul>
</nav>
<main>
    <section>
        <article id=article1></article>
        <aside>I'm so sticky</aside>
        <article id=article2></article>
        <article id=article3></article>
        <article id=article4></article>
    </section>
</main>
<footer>&copy; 2018 LAM T0IF</footer>
</body>
</html>

```

## LAM T01F WMOTU

In eo atque

Link 1  
Link 2  
Link 3  
Link 4  
rutrum odio et ligula gravida, pharetra suscipit quam congue. Suspendisse vestibulum enim et risus varius suscipit. Ut tristique placerat sapien et blandit. Etiam non iaculis orci, a blandit nibh. Nunc eget feugiat est. Sed nec tellus quis erat pulvinar sodales. Cras pharetra pulvinar auctor. Curabitur placerat nisl non mattis laoreet. Interdum et malesuada fames ac ante ipsum primis in faucibus. Sed ut lorem ante. Nulla facilisi. Duis dignissim massa vitae iaculis eleifend. Maecenas a erat vel diam porttitor aliquet et non urna. Nullam non odio et nibh adipiscing tempus. Integer ut lorem tincidunt, scelerisque nisl eget, faucibus dui. Pellentesque eget dolor risus. Sed dictum auctor massa, sed venenatis lorem tristique et. Donec a metus in orci aliquam pellentesque non euismod sem. Cras tristique nisi ac est elementum, eget rhoncus massa iaculis. Nulla facilisi. Nam sit amet sagittis arcu, gravida bibendum odio. Donec euismod sagittis orci, non scelerisque nibh pulvinar nec. Donec vel magna euismod, tempor mauris sed, ullamcorper magna. Integer non tortor id justo aliquet ultricies quis sed augue. Vivamus condimentum nulla sit amet mattis convallis. Aenean erat eros, ultricies eget arcu non, aliquet pulvinar quam. Sed rhoncus sapien at volutpat semper. Mauris hendrerit odio sit amet eros hendrerit, elementum porta sapien placerat. Aliquam erat volutpat. Nam luctus sit amet massa eget gravida. Donec viverra molestie congue. Cras eget mauris ut erat aliquet consequat. Curabitur ut sem et neque malesuada mollis non ut neque. Sed laoreet leo a erat blandit, quis vulputate augue sollicitudin. Integer commodo est nec ullamcorper vestibulum. Donec egestas sapien eu hendrerit tincidunt. Vestibulum nec orci eu sapien dictum interdum. Vivamus at porta libero, sit amet convallis massa. Phasellus ullamcorper fermentum nunc, et aliquam nisi pulvinar quis. Suspendisse tincidunt fermentum interdum. Maecenas eu tincidunt tortor, non ornare erat. Sed hendrerit nibh eget felis imperdiet molestie. Mauris auctor porta sagittis. Nam scelerisque ante nec luctus laoreet. Suspendisse ac tellus porta, ornare urna nec, ullamcorper erat. Interdum et malesuada fames ac ante ipsum primis in faucibus. Fusce lacus felis, ultrices quis nunc at, ornare pellentesque leo. Nulla quis ante vitae sem adipiscing commodo eget in ligula. Duis quis leo ut erat blandit laoreet. Quisque sapien lectus, tempus at justo ac, eleifend dignissim tortor. Sed ipsum tellus, ornare et felis viverra, lobortis dignissim lacus. Suspendisse id luctus at nunc mattis vestibulum. Donec fringilla erat malesuada, posuere massa eu, pretium metus. Phasellus imperdiet, diam id tincidunt egestas, est dolor semper orci, at consectetur dui diam a velit. Etiam volutpat feugiat aliquam. Proin euismod semper elit, id pellentesque velit vestibulum eu. Etiam dui erat, molestie sit amet tincidunt vitae, facilisis ac dolor. Duis vehicula nulla non dui sagittis tempus. Duis posuere sapien et lacina accumsan. Suspendisse potenti. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi ac rutrum sapien.

Aliquam erat volutpat. Vivamus at dionisius metus. Maecenas scelerisque turpis diam. Praesent tincidunt congue arcu, sed iaculis mi vehicula sit amet. Phasellus eu risus enim. Interdum et malesuada fames ac ante ipsum primis in faucibus. Cras congue turpis non metus volutpat.

Vivamus dignissim sagittis dolor, in aliquet justo auctor ut. Sed magna magna, sollicitudin vitae commodo nec, commodo eget eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Integer eu neque neque. Curabitur feugiat accumsan interdum. Nam sit amet auctor nisl. Nulla sed auctor diam. Phasellus sit amet nisi id diam vehicula aliquam ac sed augue. Aenean molestie viverra tincidunt. Nulla facilisi. Maecenas eu accumsan risus. Aliquam aliquam pulvinar posuere. Duis adipiscing malesuada lacus, id rutrum ante venenatis in. Cras lobortis euismod faucibus. Donec gravida porta facilisis. Vivamus eu pulvinar dui, id dignissim sapien. Nulla eleifend justo ac feugiat consequat. Integer viverra felis ac orci tempus, ac convallis magna eleifend. In ullamcorper sit amet ipsum eu volutpat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Nunc non sagittis ligula. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut dictum purus eget risus luctus faucibus vitae ac ipsum. Etiam sollicitudin eu sapien et placerat. In vitae tincidunt erat, at ultrices dui.

cubilia Curae; Integer eu neque neque. Curabitur feugiat accumsan interdum. Nam sit amet auctor nisl. Nulla sed auctor diam. Phasellus sit amet nisi id diam vehicula aliquam ac sed augue. Aenean molestie viverra tincidunt. Nulla facilisi. Maecenas eu accumsan risus. Aliquam aliquam pulvinar posuere. Duis adipiscing malesuada lacus, id rutrum ante venenatis in. Cras lobortis euismod faucibus. Donec gravida porta facilisis. Vivamus eu pulvinar dui, id dignissim sapien. Nulla eleifend justo ac feugiat consequat. Integer viverra felis ac orci tempus, ac convallis magna eleifend. In ullamcorper sit amet ipsum eu volutpat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Nunc non sagittis ligula. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut dictum purus eget risus luctus faucibus vitae ac ipsum. Etiam sollicitudin eu sapien et placerat. In vitae tincidunt erat, at ultrices dui. Vivamus dignissim sagittis dolor, in aliquet justo auctor ut. Sed magna magna, sollicitudin vitae commodo nec, commodo eget eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Integer eu neque neque. Curabitur feugiat accumsan interdum. Nam sit amet auctor nisl. Nulla sed auctor diam. Phasellus sit amet nisi id diam vehicula aliquam ac sed augue. Aenean molestie viverra tincidunt. Nulla facilisi. Maecenas eu accumsan risus. Aliquam aliquam pulvinar posuere. Duis adipiscing malesuada lacus, id rutrum ante venenatis in. Cras lobortis euismod faucibus. Donec gravida porta facilisis. Vivamus eu pulvinar dui, id dignissim sapien. Nulla eleifend justo ac feugiat consequat. Integer viverra felis ac orci tempus, ac convallis magna eleifend. In ullamcorper sit amet ipsum eu volutpat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Nunc non sagittis ligula. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut dictum purus eget risus luctus faucibus vitae ac ipsum. Etiam sollicitudin eu sapien et placerat. In vitae tincidunt erat, at ultrices dui. Phasellus imperdiet, diam id tincidunt egestas, est dolor semper orci, at consectetur dui diam a velit. Etiam volutpat feugiat aliquam. Proin euismod semper elit, id pellentesque velit vestibulum eu. Etiam dui erat, molestie sit amet tincidunt vitae, facilisis ac dolor. Duis vehicula nulla non dui sagittis tempus. Duis posuere sapien et lacina accumsan. Suspendisse potenti. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Morbi ac rutrum sapien. Aliquam erat volutpat. Vivamus at dignissim metus. Maecenas scelerisque turpis diam. Praesent tincidunt posuere arcu, sed iaculis mi vehicula sit amet. Phasellus eu risus enim. Interdum et malesuada fames ac ante ipsum primis in faucibus. Cras congue turpis non metus volutpat, venenatis condimentum mauris pellentesque. Curabitur nec fringilla arcu. Donec ac tellus vitae magna pulvinar cursus. Mauris eleifend lectus eget diam tristique, sit amet posuere risus varius. Vestibulum ante at nulla feugiat ultricies. Ut leo tellus, pellentesque et magna sit amet, interdum tincidunt ligula. Ut malesuada tortor nisl, eu imperdiet ligula convallis a. Donec et leo eleifend enim euismod vulputate. Sed pharetra lectus metus, quis porta dolor tincidunt a. In in sem eu est scelerisque dapibus. Nulla facilisi. Aliquam varius risus nec tortor condimentum auctor. Donec fringilla aliquam euismod. Proin ullamcorper mi a dui tincidunt hendrerit vitae sed nulla. In porta dui quis urna laoreet elementum. Mauris ultrices aliquet eleifend. Etiam fringilla erat in suscipit volutpat. Integer sollicitudin vulputate orci ullamcorper luctus. In hendrerit augue non aliquam rutrum. Sed porta faucibus sodales. Vivamus dignissim sagittis dolor, in aliquet justo auctor ut. Sed magna magna, sollicitudin vitae commodo nec, commodo eget eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Integer eu neque neque. Curabitur feugiat accumsan interdum. Nam sit amet auctor nisl. Nulla sed auctor diam. Phasellus sit amet nisi id diam vehicula aliquam ac sed augue. Aenean molestie viverra tincidunt. Nulla facilisi. Maecenas eu accumsan risus. Aliquam aliquam pulvinar posuere. Duis adipiscing malesuada lacus, id rutrum ante venenatis in. Cras lobortis euismod faucibus. Donec gravida porta facilisis. Vivamus eu pulvinar dui, id dignissim sapien. Nulla eleifend justo ac feugiat consequat. Integer viverra felis ac orci tempus, ac convallis magna eleifend. In ullamcorper sit amet ipsum eu volutpat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Nunc non sagittis ligula. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut dictum purus eget risus luctus faucibus vitae ac ipsum. Etiam sollicitudin eu sapien et placerat. In vitae tincidunt erat, at ultrices dui. Vivamus dignissim sagittis dolor, in aliquet justo auctor ut. Sed magna magna, sollicitudin vitae commodo nec, commodo eget eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Integer eu neque neque. Curabitur feugiat accumsan interdum. Nam sit amet auctor nisl. Nulla sed auctor diam. Phasellus sit amet nisi id diam vehicula aliquam ac sed augue. Aenean molestie viverra tincidunt. Nulla facilisi. Maecenas eu accumsan risus. Aliquam aliquam pulvinar posuere. Duis adipiscing malesuada lacus, id rutrum ante venenatis in. Cras lobortis euismod faucibus. Donec gravida porta facilisis. Vivamus eu pulvinar dui, id dignissim sapien. Nulla eleifend justo ac feugiat consequat. Integer viverra felis ac orci tempus, ac convallis magna eleifend. In ullamcorper sit amet ipsum eu volutpat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Nunc non sagittis ligula. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut dictum purus eget risus luctus faucibus vitae ac ipsum. Etiam sollicitudin eu sapien et placerat. In vitae tincidunt erat, at ultrices dui. Vivamus dignissim sagittis dolor, in aliquet justo auctor ut. Sed magna magna, sollicitudin vitae commodo nec, commodo eget eros. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Integer eu neque neque. Curabitur feugiat accumsan interdum. Nam sit amet auctor nisl. Nulla sed auctor diam. Phasellus sit amet nisi id diam vehicula aliquam ac sed augue. Aenean molestie viverra tincidunt. Nulla facilisi. Maecenas eu accumsan risus. Aliquam aliquam pulvinar posuere. Duis adipiscing malesuada lacus, id rutrum ante venenatis in. Cras lobortis euismod faucibus. Donec gravida porta facilisis. Vivamus eu pulvinar dui, id dignissim sapien. Nulla eleifend justo ac feugiat consequat. Integer viverra felis ac orci tempus, ac convallis magna eleifend. In ullamcorper sit amet ipsum eu volutpat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Nunc non sagittis ligula. Interdum et malesuada fames ac ante ipsum primis in faucibus. Ut dictum purus eget risus luctus faucibus vitae ac ipsum. Etiam sollicitudin eu sapien et placerat. In vitae tincidunt erat, at ultrices dui.

© 2018 LAM T01F

students.btsi.lu/evegi144/WAD/CSS3/positioning2.html

```
<!DOCTYPE html>
<html lang=en>
<head>
  <title>Positioning example 2</title>
  <meta charset=UTF-8>
  <style>
    body {
      margin: 0;
      overflow: hidden;
    }

    main {
      overflow: auto;
      position: absolute;
      left: 100px;
      top: 40px;
      right: 0;
      bottom: 20px;
    }

    header {
      background-color: gold;
      text-align: center;
      position: fixed;
      top: 0;
      left: 0;
      width: 100%;
      height: 40px;
    }

    nav {
      background-color: navajowhite;
      position: fixed;
```

```
    top:          40px;
    left:         0;
    width:        100px;
    bottom:       20px;
}

section {
    background-color: lightgrey;
}

footer {
    background-color: aqua;
    text-align:      center;
    position:        fixed;
    bottom:          0;
    left:            0;
    height:          20px;
    width:           100%;
}

#article1 {
    background-color: yellow;
    position:         relative;
    left:             50px;
    top:              100px;
}

#article2 {
    background-color: indianred;
    position:         absolute;
    left:             20px;
    top:              420px;
}

#article3 {
    background-color: yellowgreen;
    position:         fixed;
    left:             50px;
    top:              300px;
}

#article4 {
    background-color: white;
}

aside {
    position:         sticky;
    top:              0;
    background-color: red;
}
```

```

    h1 {
        margin: 0;
    }

    ul {
        padding: 0;
    }
</style>
</head>
<body>
    <header>
        <h1>LAM T0IF WMOTU</h1>
    </header>
    <nav>
        <ul>
            <li><a href=#>Link 1</a></li>
            <li><a href=#>Link 2</a></li>
            <li><a href=#>Link 3</a></li>
            <li><a href=#>Link 4</a></li>
        </ul>
    </nav>
    <main>
        <section>
            <article id=article1></article>
            <aside>I'm so sticky</aside>
            <article id=article2></article>
            <article id=article3></article>
            <article id=article4></article>
        </section>
    </main>
    <footer>&copy; 2018 LAM T0IF</footer>
</body>
</html>

```

On scroll-linked effects:

[developer.mozilla.org/en-US/docs/Mozilla/Performance/Scroll-linked\\_effects](https://developer.mozilla.org/en-US/docs/Mozilla/Performance/Scroll-linked_effects)

[staktrace.com/spout/entry.php?id=834](https://staktrace.com/spout/entry.php?id=834)

## Resize images automatically while preserving aspect ratio

[www.kirupa.com/html5/preserve\\_an\\_image\\_aspect\\_ratio\\_when\\_resized.htm](https://www.kirupa.com/html5/preserve_an_image_aspect_ratio_when_resized.htm)

## Table layout

### Scrollable HTML5 table with fixed header

This is a classic problem that has not had many fully satisfying solutions so far. Some interesting discussions on the topic can be found here:

<a href="https://stackoverflow.com/questions/673153/html-table-with-fixed-headers/25902860">stackoverflow.com/questions/673153/html-table-with-fixed-headers/25902860</a>
<a href="https://stackoverflow.com/questions/673153/html-table-with-fixed-headers/25818428">stackoverflow.com/questions/673153/html-table-with-fixed-headers/25818428</a>
<a href="https://stackoverflow.com/questions/17584702/how-to-add-a-scrollbar-to-an-html5-table">stackoverflow.com/questions/17584702/how-to-add-a-scrollbar-to-an-html5-table</a>
<a href="https://stackoverflow.com/questions/8423768/freeze-the-top-row-for-an-html-table-only-fixed-table-header-scrolling">stackoverflow.com/questions/8423768/freeze-the-top-row-for-an-html-table-only-fixed-table-header-scrolling</a>
<a href="http://www.sitepoint.com/community/t/flexible-html-table-with-fixed-header-and-footer-around-a-scrollable-body/271162/31">www.sitepoint.com/community/t/flexible-html-table-with-fixed-header-and-footer-around-a-scrollable-body/271162/31</a>
<a href="https://stackoverflow.com/questions/19559197/how-to-make-scrollable-table-with-fixed-headers-using-css">stackoverflow.com/questions/19559197/how-to-make-scrollable-table-with-fixed-headers-using-css</a>
<a href="http://salzerdesign.com/test/fixedTable.html">salzerdesign.com/test/fixedTable.html</a>
<a href="http://blog.freestylecoding.com/archive/2011/04/19/html-table-with-fixed-header-and-scrollable-body.aspx">blog.freestylecoding.com/archive/2011/04/19/html-table-with-fixed-header-and-scrollable-body.aspx</a>

The following table lists some solutions I've found and gives a brief comment:

<a href="http://students.btsi.lu/evegi144/WAD/CSS3/table1.html">students.btsi.lu/evegi144/WAD/CSS3/table1.html</a>	Does not work if the window is too small and a horizontal scroll bar appears.
<a href="http://students.btsi.lu/evegi144/WAD/CSS3/table2.html">students.btsi.lu/evegi144/WAD/CSS3/table2.html</a>	Same issue.
<a href="http://students.btsi.lu/evegi144/WAD/CSS3/table3.html">students.btsi.lu/evegi144/WAD/CSS3/table3.html</a>	Same issue.
<a href="http://students.btsi.lu/evegi144/WAD/CSS3/table4.html">students.btsi.lu/evegi144/WAD/CSS3/table4.html</a>	Needs setting column width.
<a href="http://students.btsi.lu/evegi144/WAD/CSS3/table5.html">students.btsi.lu/evegi144/WAD/CSS3/table5.html</a>	Simple but does not work.
<a href="http://students.btsi.lu/evegi144/WAD/CSS3/table6.html">students.btsi.lu/evegi144/WAD/CSS3/table6.html</a>	Works very well but requires a huge amount of CSS and divs.
<a href="http://students.btsi.lu/evegi144/WAD/CSS3/table7.html">students.btsi.lu/evegi144/WAD/CSS3/table7.html</a>	Very simple solution that works but not very smoothly on mobiles as for the reasons explained above under scroll-linked effects. Until <code>position: sticky</code> works correctly in the main browsers this will be my first choice.
<a href="http://students.btsi.lu/evegi144/WAD/CSS3/table8.html">students.btsi.lu/evegi144/WAD/CSS3/table8.html</a>	This is the ideal solution.

[doctorDestructo's fiddles](#) are worth having a look.

### CSS tables

CSS allows us to style elements as tables, which opens up a very powerful layout mechanism. See [www.w3.org/TR/CSS21/visuren.html#propdef-display](http://www.w3.org/TR/CSS21/visuren.html#propdef-display) and [www.w3.org/TR/CSS21/tables.html](http://www.w3.org/TR/CSS21/tables.html) for the details.

Example:

<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Etiam porta euismod rhoncus. Donec semper aliquet sollicitudin. Duis non libero ut ligula volutpat mollis. Ut varius nunc diam, sed sollicitudin massa dapibus nec. In eget neque erat. Sed interdum a felis sed ullamcorper. Curabitur molestie arcu vel quam tempor, sed faucibus magna luctus. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Vivamus arcu eros, venenatis sed tortor sed, luctus tristique lacus. Vestibulum cursus gravida enim id consectetur. Nulla laoreet, quam sed mattis pharetra, eros mauris tempus dui, eleifend molestie augue eros sed ligula. Cras tincidunt odio eu dolor egestas, vel fermentum libero placerat. In bibendum vitae lorem ac tristique. Ut laoreet tristique magna quis ornare. Proin mattis augue at mauris consequat, a fermentum metus eleifend. Aliquam tincidunt ullamcorper vehicula. Integer dapibus non ante sed venenatis. Cras cursus, leo quis consectetur placerat, nibh libero luctus eros, a hendrerit turpis metus vel lectus. Etiam suscipit, dolor vel porta blandit, lacus urna condimentum quam, quis rutrum risus urna eu nisi. Etiam lacinia euismod facilisis. Nunc imperdiet at lectus sed sagittis. Curabitur posuere rhoncus diam pellentesque sodales. Maecenas justo lectus, venenatis et malesuada lobortis, tincidunt ac enim. Fusce sapien velit, venenatis non faucibus ac, luctus vel massa. Praesent interdum, eros nec aliquet consequat, nulla urna lobortis lorem, id mollis erat dolor non ipsum. Proin lorem lacus, mattis quis convallis ac, viverra a diam. Sed id posuere justo. Nulla tempor hendrerit eros ut consectetur. Integer commodo purus augue, interdum sollicitudin ante blandit a. Maecenas sed vehicula diam, accumsan pretium lorem. Cras consectetur, urna at euismod eleifend, leo metus blandit enim, eget interdum dui mi eu sapien. Etiam fermentum nibh vel interdum sollicitudin. Suspendisse leo nunc, blandit sit amet purus ac, pulvinar dapibus dolor. Fusce egestas accumsan aliquam. Nunc tempus scelerisque massa, tempus interdum massa rhoncus vestibulum. Ut ac dictum est, nec cursus odio. Cras ut nisl ornare, ullamcorper felis sed, scelerisque erat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Etiam rutrum neque vel nibh adipiscing adipiscing. Fusce vestibulum tempus erat, at tempor nibh rhoncus dictum. Quisque lacinia laoreet nulla, ac convallis justo tristique vel. Vestibulum ipsum eros, tincidunt aliquet faucibus eget, faucibus in mi. Curabitur fringilla, nisi eu ullamcorper tincidunt, magna enim dictum elit, id suscipit mauris mauris nec dolor. Integer leo eros, volutpat a purus a, adipiscing vestibulum augue. Duis id erat ac nisl facilisis iaculis. Curabitur sit amet diam faucibus, tempor orci at, dapibus arcu. Donec aliquet erat quam, in ultrices nisl varius vel. Pellentesque vestibulum blandit mollis. Integer feugiat ante ut fringilla ullamcorper. Nam dignissim pellentesque laoreet. Etiam quis viverra massa, at volutpat erat. Quisque varius lectus vel porta tincidunt. Vivamus ultricies placerat est et convallis. Integer interdum turpis non nunc fermentum, quis laoreet lectus lobortis. Nam ornare tempus nunc, ut ullamcorper justo consequat sit amet. Nulla eu tellus a massa egestas ullamcorper at pharetra tellus. Fusce semper ullamcorper faucibus. Donec tempor turpis vel turpis adipiscing, ac pretium ante luctus. Suspendisse leo nunc, blandit sit amet purus ac, pulvinar dapibus dolor. Fusce egestas accumsan aliquam. Nunc tempus scelerisque massa, tempus interdum massa rhoncus vestibulum. Ut ac dictum est, nec cursus odio. Cras ut nisl ornare, ullamcorper felis sed, scelerisque erat. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Etiam rutrum neque vel nibh adipiscing adipiscing. Fusce vestibulum tempus erat, at tempor nibh rhoncus dictum. Quisque lacinia laoreet nulla, ac convallis justo tristique vel. Vestibulum ipsum eros, tincidunt aliquet faucibus eget, faucibus in mi. Curabitur fringilla, nisi eu ullamcorper tincidunt, magna enim dictum elit, id suscipit mauris mauris nec dolor. Integer leo eros, volutpat a purus a, adipiscing vestibulum augue. Duis id erat ac nisl facilisis iaculis. Curabitur sit amet diam faucibus, tempor orci at, dapibus arcu. Donec aliquet erat quam, in ultrices nisl varius vel. Pellentesque vestibulum blandit mollis. Integer feugiat ante ut fringilla ullamcorper. Nam dignissim pellentesque laoreet. Etiam quis viverra massa, at volutpat erat. Quisque varius lectus vel porta tincidunt. Vivamus ultricies placerat est et convallis. Integer interdum turpis non nunc fermentum, quis laoreet lectus lobortis. Nam ornare tempus nunc, ut ullamcorper justo consequat sit amet. Nulla eu tellus a massa egestas ullamcorper at pharetra tellus. Fusce semper ullamcorper faucibus. Donec tempor turpis vel turpis adipiscing, ac pretium ante luctus.</p>	<p>Pellentesque vestibulum blandit mollis. Integer feugiat ante ut fringilla ullamcorper. Nam dignissim pellentesque laoreet. Etiam quis viverra massa, at volutpat erat. Quisque varius lectus vel porta tincidunt. Vivamus ultricies placerat est et convallis. Integer interdum turpis non nunc fermentum, quis laoreet lectus lobortis. Nam ornare tempus nunc, ut ullamcorper justo consequat sit amet. Nulla eu tellus a massa egestas ullamcorper at pharetra tellus. Fusce semper ullamcorper faucibus. Donec tempor turpis vel turpis adipiscing, ac pretium ante luctus.</p>
afadsfadsfads	afadsfadsadsdsadasdsfads
sadsa	sadsadsasadsadsa

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Table layout example 1</title>
    <meta charset=UTF-8>
    <style>
      main {
        display: table;
      }

      section {
        display: table-row;
      }

      article {
        display: table-cell;
      }
    </style>
  </head>
  <body>
    <main>
      <section>
        <article></article>
        <article></article>
      </section>
      <section>
        <article>afadsfadsfdas</article>
        <article>afadsfadsadasdsadasddsfdas</article>
      </section>
      <section>
        <article>sadsa</article>
        <article>sadsadsasadsadsa</article>
      </section>
    </main>
  </body>
</html>

```

## Vertical centering

[www.w3.org/Style/Examples/007/center](http://www.w3.org/Style/Examples/007/center)

[www.vanseodesign.com/css/vertical-centering](http://www.vanseodesign.com/css/vertical-centering)

Flexbox, as described in a later section, is by far the easiest way to center any element.

Example:



[students.btsi.lu/evegi144/WAD/CSS3/centering1.html](https://students.btsi.lu/evegi144/WAD/CSS3/centering1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Centering example</title>
    <meta charset=UTF-8>
    <style>
      main {
        position:      absolute;
        top:            0;
        left:           0;
        height:         100%;
        width:          100%;
        background-color: lightblue;
        display:        flex;
      }

      img {
        margin: auto;
      }
    </style>
  </head>
  <body>
    <main>
      <img src=logo_ltam.gif alt="LTAM Logo">
    </main>
  </body>
</html>
```

Here a more generic way to center any element without styling it: [students.btsi.lu/evegi144/WAD/CSS3/centering2.html](https://students.btsi.lu/evegi144/WAD/CSS3/centering2.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Centering example</title>
    <meta charset=UTF-8>
    <style>
      main {
        position:      absolute;
        top:           0;
        left:          0;
        height:        100%;
        width:         100%;
        background-color: lightblue;
        display:       flex;
        align-items:    center;
        justify-content: center;
        /*text-align:   center;*/
        flex-flow:      column;
      }
    </style>
  </head>
  <body>
    <main>
      <section>
        <img src=logo_ltam.gif alt="LTAM Logo">
      </section>
      <section>Section 2</section>
    </main>
  </body>
</html>

```

## Multiple columns

[www.w3schools.com/css/css3\\_multiple\\_columns.asp](http://www.w3schools.com/css/css3_multiple_columns.asp)

## Flexible box layout

According to [www.w3.org/TR/css3-flexbox](http://www.w3.org/TR/css3-flexbox), flexible box layout is

a CSS box model optimized for user interface design. In the flex layout model, the children of a flex container can be laid out in any direction, and can “flex” their sizes, either growing to fill unused space or shrinking to avoid overflowing the parent. Both horizontal and vertical alignment of the children can be easily manipulated. Nesting of these boxes (horizontal inside vertical, or vertical inside horizontal) can be used to build layouts in two dimensions.

It is a big step forward in terms of GUI development and will be our preferred approach.

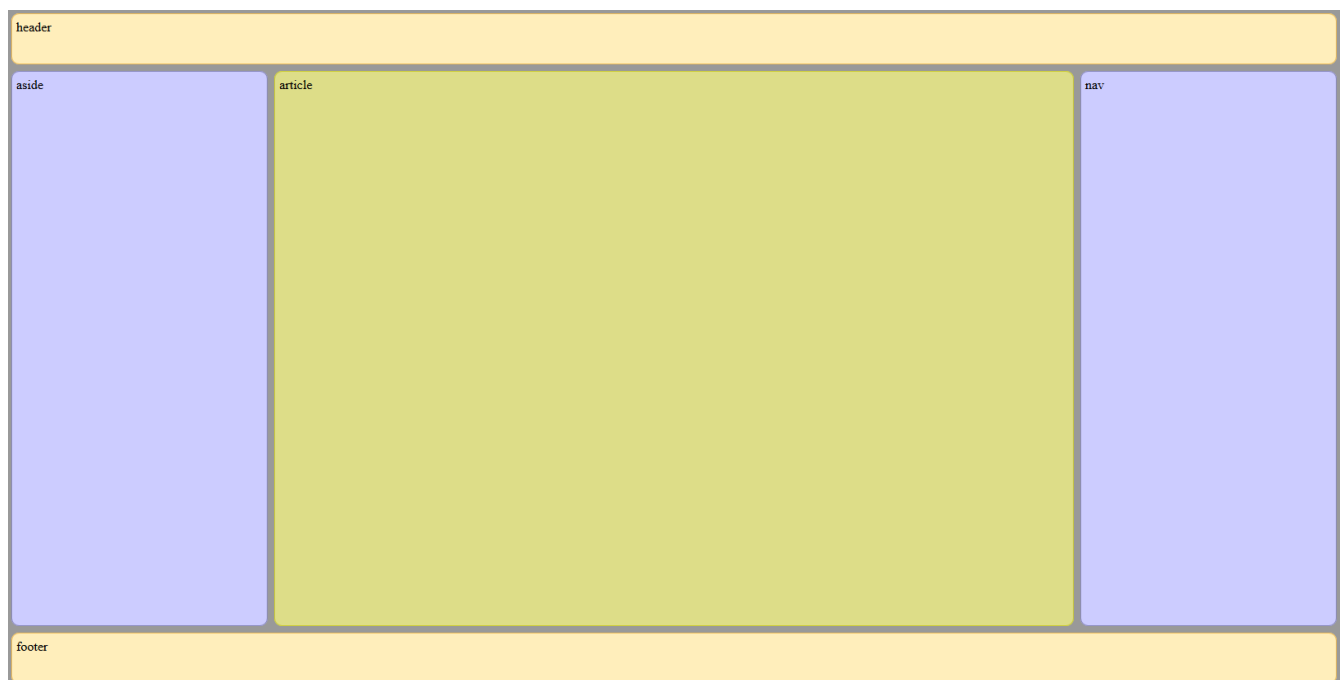
To learn Flexbox in a fun way, take a look at [www.flexboxdefense.com](http://www.flexboxdefense.com) and [flexboxfroggy.com](http://flexboxfroggy.com).

You can find in-depth information and examples at the following links:

<a href="http://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout">developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout</a>
<a href="http://www.w3schools.com/cssref/css3_pr_flex.asp">www.w3schools.com/cssref/css3_pr_flex.asp</a>
<a href="http://bocoup.com/weblog/dive-into-flexbox">bocoup.com/weblog/dive-into-flexbox</a>
<a href="http://www.w3.org/TR/css3-flexbox">www.w3.org/TR/css3-flexbox</a>
<a href="http://css-tricks.com/snippets/css/a-guide-to-flexbox">css-tricks.com/snippets/css/a-guide-to-flexbox</a>
<a href="http://html5please.com/#flexbox">html5please.com/#flexbox</a>
<a href="http://philipwalton.github.io/solved-by-flexbox">philipwalton.github.io/solved-by-flexbox</a>
<a href="http://www.sketchingwithcss.com/samplechapter">www.sketchingwithcss.com/samplechapter</a>
<a href="http://www.sketchingwithcss.com/samplechapter/cheatsheet.html">www.sketchingwithcss.com/samplechapter/cheatsheet.html</a>
<a href="http://www.smashingmagazine.com/2013/05/22/centering-elements-with-flexbox">www.smashingmagazine.com/2013/05/22/centering-elements-with-flexbox</a>
<a href="http://designshack.net/articles/css/build-a-web-page-with-css3-flexbox">designshack.net/articles/css/build-a-web-page-with-css3-flexbox</a>
<a href="http://www.html5rocks.com/en/tutorials/flexbox/quick">www.html5rocks.com/en/tutorials/flexbox/quick</a>

Let's start with a simple header - main - footer layout, where the main part contains a navigation, a section and an aside. Note that the order of the navigation and aside are changed using CSS.

Study the comments in the CSS file together with the links above to gain a deeper understanding of flex layout.



[students.btsi.lu/evegi144/WAD/CSS3/flextest1.html](http://students.btsi.lu/evegi144/WAD/CSS3/flextest1.html)

```
<!DOCTYPE html>
<html lang="en">
<head>
<title>Flexible Box Layout Test 1</title>
<meta charset=utf-8>
<meta name=viewport content="width=device-width, initial-scale=1">
<style>
/* This will make sure that our GUI fills the whole browser window. */
html, body {
    width: 100%;
    height: 100%;
    margin: 0;
}

body {
    background: #999999;
    display: flex; /* This element is a flex container. */
    flex-flow: column; /* Change the main axis to column instead of row. */
    overflow: hidden; /* We don't want scrollbars. */
}

main {
    margin: 0;
    padding: 0;
    display: flex; /* This is another flex container. */
    /* It will grow 3 times faster and shrink at the same speed as the others. */
    flex: auto;
    overflow: hidden; /* We don't want scrollbars. */
}

main > section {
    margin: 4px;
    padding: 5px;
    border: 1px solid #cccc33;
    border-radius: 7pt;
    background: #dddd88;
    flex: 3 1 60%;
    order: 2; /* This element will be displayed in second position. */
    overflow: auto; /* We want scrollbars when required by the content. */
}

main > nav {
    margin: 4px;
    padding: 5px;
    border: 1px solid #8888bb;
    border-radius: 7pt;
    background: #ccccff;
    flex: 1 6 20%;
    order: 3; /* This element will be displayed in third position. */
}
```

```

main > aside {
  margin:      4px;
  padding:     5px;
  border:      1px solid #8888bb;
  border-radius: 7pt;
  background:   #ccccff;
  flex:        1 6 20%;
  order:       1; /* This element will be displayed in first position. */
}

header, footer {
  margin:      4px;
  padding:     5px;
  border:      1px solid #eebb55;
  border-radius: 7pt;
  background:   #ffeebb;
  height:      50px;
}

/* Too narrow to support three columns */
@media all and (max-width: 640px) {
  main {
    flex-flow: column;
  }

  main > section, main > nav, main > aside {
    /* Return them to document order */
    order: 0;
  }

  main > nav, main > aside, header, footer {
    min-height: 50px;
    max-height: 50px;
  }
}

</style>
</head>
<body>
  <header>header</header>
  <main>
    <nav>nav</nav>
    <section>section</section>
    <aside>aside</aside>
  </main>
  <footer>footer</footer>
</body>
</html>

```

Now we let the user resize some of the elements:

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Flexible Box Layout Test 2</title>
    <meta charset=utf-8>
    <meta name=viewport content="width=device-width, initial-scale=1">
    <style>
      body {
        position: absolute;
        top: 0;
        left: 0;
        width: 100%;
        height: 100%;
        background: #999999;
        display: flex;
        flex-flow: column;
        margin: 0;
        overflow: hidden;
      }

      main {
        min-height: 50px;
        margin: 0;
        padding: 0;
        display: flex;
        flex: auto;
      }

      main > article {
        margin: 4px;
        padding: 5px;
        border: 1px solid #cccc33;
        border-radius: 7pt;
        background: #dddd88;
        flex: auto;
        min-width: 50px;
      }

      main > nav {
        margin: 4px;
        padding: 5px;
        border: 1px solid #8888bb;
        border-radius: 7pt;
        background: #ccccff;
        min-width: 150px;
      }

      main > aside {
```

```

margin:      4px;
padding:     5px;
border:      1px solid #8888bb;
border-radius: 7pt;
background:  #ccccff;
min-width:   50px;
flex:        auto;
}

header, footer {
  display:    block;
  margin:     4px;
  padding:    5px;
  min-height: 100px;
  border:     1px solid #eebb55;
  border-radius: 7pt;
  background: #ffeebb;
}

.splitter {
  border-left: 2px solid grey;
  width:       2px;
  min-width:   2px;
  cursor:      col-resize;
}

/* Too narrow to support three columns */
@media all and (max-width: 640px) {
  main {
    flex-flow: column;
  }

  main > nav, main > aside, header, footer {
    min-height: 50px;
    max-height: 50px;
  }
}

</style>
<script src=flextest2.js></script>
</head>
<body>
  <header>header</header>
  <main>
    <nav>nav</nav>
    <article>article</article>
    <div class="splitter"></div>
    <aside>aside</aside>
  </main>
  <footer>footer</footer>
</body>

```

</html>

```
'use strict';

let lastFirstElWidth = 0, lastThirdElWidth = 0;

// https://hacks.mozilla.org/2013/12/application-layout-with-css3-flexible-box-module
class Splitter {
  constructor(handler, leftEl, rightEl) {
    this.lastX = 0;
    this.dragListener = null;
    this.endDragListener = null;
    this.leftEl = leftEl;
    this.rightEl = rightEl;

    handler.addEventListener('mousedown', evt => {
      evt.preventDefault();
      this.lastX = evt.clientX;
      // http://msdn.microsoft.com/en-us/library/windows/apps/hh703713.aspx
      this.dragListener = this.drag;
      this.endDragListener = this.endDrag;
      addEventListener('mousemove', this.dragListener);
      addEventListener('mouseup', this.endDragListener);
    });

    this.drag = evt => {
      let wL, wR;
      const wDiff = evt.clientX - this.lastX;
      wL = getComputedStyle(this.leftEl).width;
      wR = getComputedStyle(this.rightEl).width;
      wL = parseInt(wL) + wDiff;
      wR = parseInt(wR) - wDiff;
      this.leftEl.style.width = wL + 'px';
      this.rightEl.style.width = wR + 'px';
      this.lastX = evt.clientX;
      lastFirstElWidth = wL;
      lastThirdElWidth = wR;
    };

    this.endDrag = () => {
      removeEventListener('mousemove', this.dragListener);
      removeEventListener('mouseup', this.endDragListener);
    };
  };
}

const init = () => {
  const splitter = new Splitter(document.getElementsByClassName('splitter')[0],
    document.getElementsByTagName('article')[0],
    document.getElementsByTagName('aside')[0]);
}
```

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Flexible Box Layout Test 3</title>
    <meta charset=utf-8>
    <meta name=viewport content="width=device-width, initial-scale=1">
    <style>
      body {
        position: absolute;
        top: 0;
        left: 0;
        width: 100%;
        height: 100%;
        background: #999999;
        display: flex;
        flex-flow: column;
        margin: 0;
        overflow: hidden;
      }

      main {
        min-height: 50px;
        margin: 0;
        padding: 0;
```

```
    display:    flex;
    flex:       auto;
}

main > article {
    margin:      4px;
    padding:     5px;
    border:      1px solid #cccc33;
    border-radius: 7pt;
    background:  #dddd88;
    flex:        auto;
    min-width:   50px;
}

main > nav {
    margin:      4px;
    padding:     5px;
    border:      1px solid #8888bb;
    border-radius: 7pt;
    background:  #ccccff;
    min-width:   50px;
}

main > aside {
    margin:      4px;
    padding:     5px;
    border:      1px solid #8888bb;
    border-radius: 7pt;
    background:  #ccccff;
    flex:        auto;
    min-width:   50px;
}

header, footer {
    margin:      4px;
    padding:     5px;
    min-height:  50px;
    border:      1px solid #eebb55;
    border-radius: 7pt;
    background:  #ffeebb;
}

.verticalSplitter {
    border-left: 2px solid grey;
    width:      2px;
    min-width:  2px;
    cursor:     col-resize;
}

.horizontalSplitter {
    border-top: 2px solid grey;
```

```

    height:    2px;
    min-height: 2px;
    cursor:    row-resize;
}

/* Too narrow to support three columns */
@media all and (max-width: 640px) {
    main {
        flex-flow: column;
    }

    main > nav, header, footer {
        height: 50px;
    }

    .horizontalSplitter, .verticalSplitter {
        display: none;
    }
}

</style>
</head>
<body>
  <header>header</header>
  <div id=hs1 class=horizontalSplitter></div>
  <main>
    <nav>nav</nav>
    <article>article</article>
    <div class=verticalSplitter></div>
    <aside>aside</aside>
  </main>
  <div id=hs2 class=horizontalSplitter></div>
  <footer>footer</footer>
  <script src=flextest3.js></script>
</body>
</html>

```

```

'use strict';

let lastFirstElWidth = 0, lastThirdElWidth = 0;

// https://hacks.mozilla.org/2013/12/application-layout-with-css3-flexible-box-module
class Splitter {
  constructor(vertical) {
    this.vertical = vertical;
    this.lastCoord = 0;

    this.init = (splitter, firstEl, thirdEl) =>
    {
      this.firstEl = firstEl;

```

```

    this.thirdEl = thirdEl;
    splitter.addEventListener('mousedown', evt => {
        evt.preventDefault();
        if (vertical) this.lastCoord = evt.clientX;
        else this.lastCoord = evt.clientY;
        this.dragListener = this.drag;
        this.endDragListener = this.endDrag;
        addEventListener('mousemove', this.dragListener);
        addEventListener('mouseup', this.endDragListener);
    });
};

this.drag = evt => {
    let coord1, coord3, coordDiff;
    if (vertical) {
        coordDiff = evt.clientX - this.lastCoord;
        coord1 = getComputedStyle(this.firstEl).width;
        coord3 = getComputedStyle(this.thirdEl).width;
    }
    else {
        coordDiff = evt.clientY - this.lastCoord;
        coord1 = getComputedStyle(this.firstEl).height;
        coord3 = getComputedStyle(this.thirdEl).height;
    }
    coord1 = parseInt(coord1) + coordDiff;
    coord3 = parseInt(coord3) - coordDiff;
    if (vertical) {
        this.firstEl.style.width = coord1 + 'px';
        this.thirdEl.style.width = coord3 + 'px';
        this.lastCoord = evt.clientX;
        lastFirstElWidth = coord1;
        lastThirdElWidth = coord3;
        console.log(lastFirstElWidth + ' ' + lastThirdElWidth);
    }
    else {
        this.firstEl.style.height = coord1 + 'px';
        this.thirdEl.style.height = coord3 + 'px';
        this.lastCoord = evt.clientY;
    }
};

this.endDrag = () => {
    removeEventListener('mousemove', this.dragListener);
    removeEventListener('mouseup', this.endDragListener);
};
}
}

const init = () => {
    const verticalSplitter = new Splitter(true);
    verticalSplitter.init(document.getElementsByClassName('verticalSplitter')[0],

```

```

    document.querySelector('article'), document.querySelector('aside')));
const horizontalSplitter1 = new Splitter(false);
horizontalSplitter1.init(document.getElementById('hs1'),
    document.querySelector('header'), document.querySelector('main'));
const horizontalSplitter2 = new Splitter(false);
horizontalSplitter2.init(document.getElementById('hs2'),
    document.querySelector('main'), document.querySelector('footer'));

/* Our CSS switches flex flow to column if window width <= 640.
   In this case we need to remove the width set by the splitter dragging,
   otherwise the layout won't be right aligned on small screens.
   If window width increases again, we want the previous widths back.
*/
const handleResize = () => {
    if (innerWidth <= 640) {
        document.querySelector('article').style.width = '';
        document.querySelector('aside').style.width = '';
    }
    else {
        document.querySelector('article').style.width = lastFirstElWidth + 'px';
        document.querySelector('aside').style.width = lastThirdElWidth + 'px';
    }
};

window.addEventListener('resize', handleResize);

window.addEventListener('load', init);

```

The following example illustrates how to achieve specific scrollbar behavior:

[students.btsi.lu/evegi144/WAD/CSS3/flextest4.php](https://students.btsi.lu/evegi144/WAD/CSS3/flextest4.php)

```

<!DOCTYPE html>
<html>
  <head>
    <title>Flexible Box Layout Test 4</title>
    <meta charset=utf-8>
    <meta name=viewport content="width=device-width, initial-scale=1">
    <style type="text/css">
      html, body {
        height: 100%;
        width: 100%;
        padding: 0;
        margin: 0;
      }

      body {
        overflow: hidden;
        display: flex;

```

```

    flex-direction: column;
}

header {
    height: 75px;
    min-height: 75px;
}

footer {
    height: 25px;
    min-height: 25px;
}

main {
    display: flex;
    flex: auto;
    border: solid grey;
    border-width: 1px 0;
    overflow: hidden;
}

nav {
    width: 150px;
    min-width: 150px;
}

section {
    border: solid grey;
    border-width: 0 0 0 1px;
    flex: auto;
    overflow: auto;
}
</style>
</head>
<body>
    <header>header</header>
    <main>
        <nav>nav</nav>
        <section>article<br>read <a
            href="https://hacks.mozilla.org/2013/12/application-layout-with-css3-flexible-
box-module/"
            target="_blank">Application Layout with CSS3 Flexible Box Module</a>
        <p style="width:1000px;">
            <?php require_once 'flectest4.txt'; ?>
        </p>
        </section>
    </main>
    <footer>footer</footer>
</body>
</html>

```

Finally a more advanced example combining flexbox, splitters and scrollbars:

[students.btsi.lu/evegi144/WAD/CSS3/flextest5.php](http://students.btsi.lu/evegi144/WAD/CSS3/flextest5.php)

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Flexible Box Layout Test 5</title>
    <meta charset=utf-8>
    <meta name=viewport content="width=device-width, initial-scale=1">
    <style>
      html, body {
        width: 100%;
        height: 100%;
      }

      body {
        background: #999999;
        display: flex;
        flex-direction: column;
        margin: 0;
        overflow: hidden;
      }

      main {
        margin: 0;
        padding: 0;
        display: flex;
        flex: auto;
        overflow: auto;
      }

      main > nav {
        margin: 4px;
        padding: 5px;
        border: 1px solid #8888bb;
        border-radius: 7pt;
        background: #ccccff;
        min-width: 50px;
        flex: auto;
      }

      main > section {
        margin: 4px;
        padding: 5px;
        border: 1px solid #8888bb;
        border-radius: 7pt;
        background: #ccccff;
        display: flex;
        flex-direction: column;
        min-width: 50px;
```

```

        overflow:      hidden;
    }

    main > section > section {
        display:   flex;
        flex-flow: column;
        flex:      auto;
        overflow:  hidden;
    }

    header, footer {
        margin:      4px;
        padding:      5px;
        min-height:   50px;
        border:       1px solid #eebb55;
        border-radius: 7pt;
        background:    #ffeebb;
    }

    #s1 {
        flex:      auto;
        flex-flow: column;
        overflow:  hidden;
    }

    #s1 > article, #s2 {
        overflow: auto;
    }

    .verticalSplitter {
        border-left: 2px solid grey;
        width:       2px;
        min-width:   2px;
        cursor:      col-resize;
    }

    .horizontalSplitter {
        border-top: 2px solid grey;
        height:     2px;
        min-height: 2px;
        cursor:     row-resize;
    }

    /* Too narrow to support three columns */
    @media all and (max-width: 640px) {
        main {
            flex-flow: column;
        }

        .horizontalSplitter, .verticalSplitter {
            display: none;
        }
    }

```

```

    }
  }
</style>
<script src=flextest5.js></script>
</head>
<body>
  <header>header</header>
  <main>
    <nav>nav</nav>
    <div class=verticalSplitter></div>
    <section>
      <section id=s1>
        <header>Header s1</header>
        <article><?php require 'flextest4.txt'; ?></article>
      </section>
      <div id=hs1 class=horizontalSplitter></div>
      <section id=s2><?php require 'flextest4.txt'; ?></section>
    </section>
  </main>
  <footer>footer</footer>
</body>
</html>

```

```

'use strict';

let verticalSplitter, horizontalSplitter;

// https://hacks.mozilla.org/2013/12/application-layout-with-css3-flexible-box-module
class Splitter {
  constructor(vertical) {
    this.vertical = vertical;
    this.lastCoord = 0;
    if (vertical) {
      this.lastFirstElWidth = 0;
      this.lastThirdElWidth = 0;
    }

    this.init = (splitter, firstEl, thirdEl) => {
      {
        this.firstEl = firstEl;
        this.thirdEl = thirdEl;
        splitter.addEventListener('mousedown', evt => {
          evt.preventDefault();
          if (vertical) this.lastCoord = evt.clientX;
          else this.lastCoord = evt.clientY;
          this.dragListener = this.drag;
          this.endDragListener = this.endDrag;
          addEventListener('mousemove', this.dragListener);
          addEventListener('mouseup', this.endDragListener);
        });
      }
    };
  }
}

```

```

};

this.drag = evt => {
  let coord1, coord3, coordDiff;
  if (vertical) {
    coordDiff = evt.clientX - this.lastCoord;
    coord1 = getComputedStyle(this.firstEl).width;
    coord3 = getComputedStyle(this.thirdEl).width;
  }
  else {
    coordDiff = evt.clientY - this.lastCoord;
    coord1 = getComputedStyle(this.firstEl).height;
    coord3 = getComputedStyle(this.thirdEl).height;
  }
  coord1 = parseInt(coord1) + coordDiff;
  coord3 = parseInt(coord3) - coordDiff;
  if (vertical) {
    this.firstEl.style.width = coord1 + 'px';
    this.thirdEl.style.width = coord3 + 'px';
    this.lastCoord = evt.clientX;
    this.lastFirstElWidth = coord1;
    this.lastThirdElWidth = coord3;
  }
  else {
    this.firstEl.style.height = coord1 + 'px';
    this.thirdEl.style.height = coord3 + 'px';
    this.lastCoord = evt.clientY;
  }
};

this.endDrag = () => {
  removeEventListener('mousemove', this.dragListener);
  removeEventListener('mouseup', this.endDragListener);
};
}
}

const init = () => {
  const verticalSplitter = new Splitter(true);
  const horizontalSplitter = new Splitter(false);
  verticalSplitter.init(document.getElementsByClassName('verticalSplitter')[0],
    document.querySelector('nav'), document.querySelector('section'));
  horizontalSplitter.init(document.getElementById('hs1'),
    document.getElementById('s1'), document.getElementById('s2'));

  /* Our CSS switches flex flow to column if window width <= 640.
     In this case we need to remove the width set by the splitter dragging,
     otherwise the layout won't be right aligned on small screens.
     If window width increases again, we want the previous widths back.
  */
  const handleResize = () => {

```

```

    if (innerWidth <= 640) {
        document.querySelector('article').style.width = '';
        document.querySelector('aside').style.width = '';
    }
    else {
        document.querySelector('article').style.width = lastFirstElWidth + 'px';
        document.querySelector('aside').style.width = lastThirdElWidth + 'px';
    }
};

window.addEventListener('resize', handleResize);

window.addEventListener('load', init);

```

A flexible navigation menu:

[students.btsi.lu/evegi144/WAD/CSS3/flextest6.php](http://students.btsi.lu/evegi144/WAD/CSS3/flextest6.php)

```

<!DOCTYPE html>
<html lang=en>
<head>
    <meta charset=UTF-8>
    <title>Flexible Box Layout Test 6</title>
    <style>
        html { /* http://www.paulirish.com/2012/box-sizing-border-box-ftw */
            box-sizing: border-box;
        }
        *, *:before, *:after {
            box-sizing: inherit;
        }

        html, body {
            width: 100%;
            height: 100%;
            margin: 0;
        }

        body {
            display: flex;
            flex-flow: column;
            overflow: hidden;
        }

        main {
            margin: 0;
            overflow: auto;
        }

        nav {

```

```

padding: 0;
}

ul {
list-style: none;
padding: 0;
display: flex;
flex: auto;
background-color: pink;
margin: 0;
}

li {
flex: auto;
}

nav > ul > li > a {
text-decoration: none;
margin: 0;
padding: 0;
display: inline-block;
width: 100%;
text-align: center;
background-color: orange;
border: solid black 1px;
}

nav > ul > li > a:hover {
background-color: yellow;
}

main > section > a {
text-decoration: none;
background-color: tomato;
border-radius: 5px;
padding: 5px;
}
</style>
</head>
<body>
<nav>
<ul>
<li><a href=#about>About</a></li>
<li><a href=#portfolio>Portfolio</a></li>
<li><a href=#contact>Contact</a></li>
</ul>
</nav>
<main>
<section id=about>
<h1>About</h1>
Lorem ipsum dolor sit amet, consectetur adipiscing elit.

```

```
</section>
<section id=portfolio>
  <h1>Portfolio</h1>
  <?php require_once 'flextest4.txt'; ?>
</section>
<section id=contact>
  <h1>Contact</h1>
  <form>
    Name: <input required>
    Email: <input required><button>Send</button>
  </form>
</section>
</main>
</body>
</html>
```

Another layout variation:

[students.btsi.lu/evegi144/WAD/CSS3/flextest7.html](https://students.btsi.lu/evegi144/WAD/CSS3/flextest7.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Flexible Box Layout Test 7</title>
    <style>
      nav {
        display:          flex;
        background-color: #0a73a7;
      }

      ul {
        display:          flex;
        flex:              auto;
        list-style:        none;
        background-color: lightgrey;
      }

      li {
        text-align: center;
        display:          block;
      }

      #l1 {
        justify-content: center;
        background-color: lightgreen;
      }

      #l2 {
        max-width:         130px;
        background-color: lightblue;
      }
    </style>
  </head>
  <body>
    <nav>
      <ul id=l1>
        <li>Item 1</li>
        <li>Item 2</li>
        <li>Item 3</li>
      </ul>
      <ul id=l2>
        <li>Item 4</li>
        <li>Item 5</li>
        <li>Item 6</li>
      </ul>
    </nav>
  </body>
</html>
```

## Grid layout

Grid layout makes it easy to build the big picture aspects of our layout, i.e. the row AND column layout. We can then use Flexbox to manage the horizontal OR vertical alignment of the content inside the grid elements.

<a href="https://css-tricks.com/snippets/css/complete-guide-grid">css-tricks.com/snippets/css/complete-guide-grid</a>
<a href="https://caniuse.com/#search=Grid">caniuse.com/#search=Grid</a>
<a href="https://www.mozilla.org/en-US/developer/css-grid">www.mozilla.org/en-US/developer/css-grid</a>
<a href="https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout">developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout</a>
<a href="https://gridbyexample.com/">gridbyexample.com/</a>
<a href="https://bitsofco.de/how-the-minmax-function-works">bitsofco.de/how-the-minmax-function-works</a>
<a href="https://tutorialzine.com/2017/03/css-grid-vs-flexbox">tutorialzine.com/2017/03/css-grid-vs-flexbox</a>

Here's a very simple example:

[students.btsi.lu/evegi144/WAD/CSS3/grid1.html](https://students.btsi.lu/evegi144/WAD/CSS3/grid1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>CSS Grid example</title>
    <style>
      html, body {
        width: 100%;
        height: 100%;
        margin: 0;
      }

      main {
        height: 100%;
        display: grid;
        grid-template-columns: 3fr 1fr 2fr;
        grid-template-rows: 1fr 3fr;
        /*align-items: stretch;*/
      }

      section {
        background-color: lawngreen;
        border: darkmagenta solid 2px;
      }
    </style>
  </head>
  <body>
    <main>
      <section>One</section>
      <section>Two</section>
      <section>Three</section>
      <section>Four</section>
      <section>Five</section>
      <section>Six</section>
    </main>
  </body>
</html>
```

When should we use Grid and when Flexbox? This [article](#) dives into the question.

To support browsers without grid see:

[www.smashingmagazine.com/2017/11/css-grid-supporting-browsers-without-grid](http://www.smashingmagazine.com/2017/11/css-grid-supporting-browsers-without-grid)

[developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_Grid\\_Layout/CSS\\_Grid\\_and\\_Progressive\\_Enhancement](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Grid_Layout/CSS_Grid_and_Progressive_Enhancement)

[rachelandrew.co.uk/archives/2016/11/26/should-i-try-to-use-the-ie-implementation-of-css-grid-layout](http://rachelandrew.co.uk/archives/2016/11/26/should-i-try-to-use-the-ie-implementation-of-css-grid-layout)

## Responsive design

Responsive web design aims at building websites that work on mobile devices, tablets, and desktop screens.

It is important to set the viewport to optimize the user's experience:

```
<meta name=viewport content="width=device-width, initial-scale=1">
```

For detailed explanations see:



[www.w3schools.com/css/css\\_rwd\\_viewport.asp](http://www.w3schools.com/css/css_rwd_viewport.asp)

[developer.mozilla.org/en-US/docs/Mozilla/Mobile/Viewport\\_meta\\_tag](http://developer.mozilla.org/en-US/docs/Mozilla/Mobile/Viewport_meta_tag)

[www.quirksmode.org/mobile/viewports.html](http://www.quirksmode.org/mobile/viewports.html)

[www.quirksmode.org/mobile/viewports2.html](http://www.quirksmode.org/mobile/viewports2.html)

[www.quirksmode.org/mobile](http://www.quirksmode.org/mobile)

[developer.mozilla.org/en-US/docs/Web/CSS/Media\\_Queries/Using\\_media\\_queries](http://developer.mozilla.org/en-US/docs/Web/CSS/Media_Queries/Using_media_queries)

[drafts.csswg.org/mediaqueries](http://drafts.csswg.org/mediaqueries)

[css-tricks.com/dont-overthink-flexbox-grids](http://css-tricks.com/dont-overthink-flexbox-grids)

[www.w3schools.com/cssref/css3\\_pr\\_mediaquery.asp](http://www.w3schools.com/cssref/css3_pr_mediaquery.asp)

[developer.mozilla.org/en-US/docs/Web\\_Development/Responsive\\_Web\\_design](http://developer.mozilla.org/en-US/docs/Web_Development/Responsive_Web_design)

[blog.teamtreehouse.com/modern-field-guide-responsive-web-design](http://blog.teamtreehouse.com/modern-field-guide-responsive-web-design)

[screensiz.es/phone](http://screensiz.es/phone)

[developer.mozilla.org/en-US/docs/Mozilla/Mobile/Viewport\\_meta\\_tag](http://developer.mozilla.org/en-US/docs/Mozilla/Mobile/Viewport_meta_tag)

[developer.mozilla.org/en-US/docs/Web/HTML/Element/meta](http://developer.mozilla.org/en-US/docs/Web/HTML/Element/meta)

[webdesignerwall.com/tutorials/viewport-meta-tag-for-non-responsive-design](http://webdesignerwall.com/tutorials/viewport-meta-tag-for-non-responsive-design)

[www.smashingmagazine.com/2011/07/22/responsive-web-design-techniques-tools-and-design-strategies](http://www.smashingmagazine.com/2011/07/22/responsive-web-design-techniques-tools-and-design-strategies)

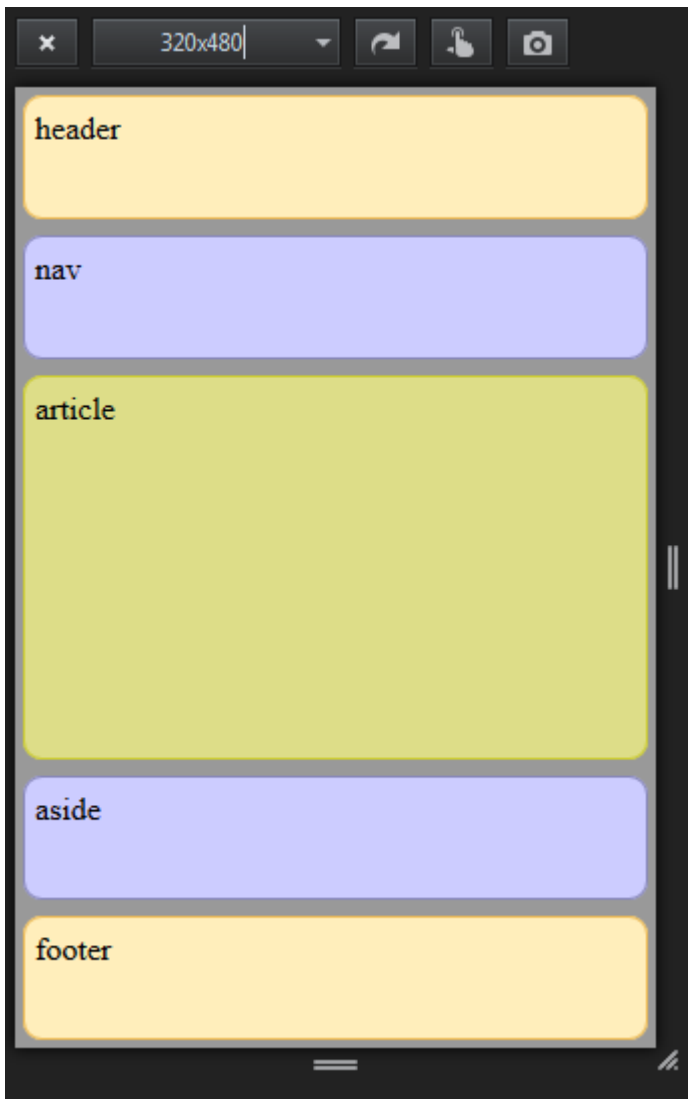
[www.quirksmode.org/mobile/metaviewport](http://www.quirksmode.org/mobile/metaviewport)

[googlewebmastercentral.blogspot.co.uk/2012/04/responsive-design-harnessing-power-of.html](http://googlewebmastercentral.blogspot.co.uk/2012/04/responsive-design-harnessing-power-of.html)

[responsivedesign.is/develop/html/viewport-meta-element](http://responsivedesign.is/develop/html/viewport-meta-element)

[css-tricks.com/accessible-simple-responsive-tables](http://css-tricks.com/accessible-simple-responsive-tables)

In Firefox, pressing **Ctrl** + **Shift** + **M** or **Tools** › **Web Developer** › **Responsive Design View**



In Chrome developer tools, `Ctrl + Shift + M` opens up a similar view. See [developers.google.com/web/tools/chrome-devtools/device-mode/emulate-mobile-viewports](https://developers.google.com/web/tools/chrome-devtools/device-mode/emulate-mobile-viewports) . [www.mobilephoneemulator.com](http://www.mobilephoneemulator.com) is useful to test your layout on mobile devices. See also [www.quora.com/Is-there-an-easy-way-to-get-view-emulate-a-Safari-browser-on-a-Windows-machine](http://www.quora.com/Is-there-an-easy-way-to-get-view-emulate-a-Safari-browser-on-a-Windows-machine)

Let's look at a practical example:

[students.btsi.lu/evegi144/WAD/CSS3/responsive1.html](http://students.btsi.lu/evegi144/WAD/CSS3/responsive1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <title>Responsive layout example 1</title>
    <style>
      html { /* http://www.paulirish.com/2012/box-sizing-border-box-ftw */
        box-sizing: border-box;
      }

      *, *:before, *:after {
        box-sizing: inherit;
      }
    </style>
  </head>
  <body>
    <div>
      <div>header</div>
      <div>nav</div>
      <div>article</div>
      <div>aside</div>
      <div>footer</div>
    </div>
  </body>
</html>
```

```

}

ul {
  display:    flex;
  flex-wrap:  wrap;
  padding:    0;
  list-style: none;
}

ul li {
  flex:       auto;
  text-align: center;
  padding:    2px;
}

@media (min-width: 10em) {
  ul li {
    flex-basis: 33%;
  }
}

@media (min-width: 28em) {
  ul li {
    flex-basis: 0;
  }
}

ul li a {
  display:      block;
  text-decoration: none;
}
</style>
</head>
<body>
  <nav>
    <ul>
      <li><a href=#>Item 1</a></li>
      <li><a href=#>Item 2</a></li>
      <li><a href=#>Item 3</a></li>
      <li><a href=#>Item 4</a></li>
      <li><a href=#>Item 5</a></li>
      <li><a href=#>Item 6</a></li>
    </ul>
  </nav>
  <main>
    Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut et tempus urna. In
sed
    sagittis arcu. Cras in sapien diam. Aenean massa ipsum, rutrum eu facilisis
vitae,
    semper in arcu. Morbi vitae tortor sit amet turpis feugiat aliquam. Aliquam eu
    rhoncus odio, quis rutrum magna.

```

```
</main>
</body>
</html>
```

## Responsive menus

[students.btsi.lu/evegi144/WAD/CSS3/responsive2.php](https://students.btsi.lu/evegi144/WAD/CSS3/responsive2.php)

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Responsive layout example 2</title>
    <meta charset=utf-8>
    <meta name=viewport content="width=device-width, initial-scale=1">
    <style>
      html { /* http://www.paulirish.com/2012/box-sizing-border-box-ftw */
        box-sizing: border-box;
      }

      *, *:before, *:after {
        box-sizing: inherit;
      }

      nav {
        margin:          4px;
        padding:          4px;
        border:           1px solid #8888bb;
        border-radius:    7pt;
        background-color: #ccccff;
        min-width:        50px;
        flex:             auto;
      }

      ul {
        display:    flex;
        flex-wrap:  wrap;
        margin:     0;
        padding:    0;
        list-style: none;
      }

      ul li {
        flex:          auto;
        text-align:    center;
        padding:       2px;
        background-color: #ffeebb;
      }

      @media (min-width: 10em) {
        ul li {
```

```
        flex-basis: 33%;
    }
}

@media (min-width: 28em) {
    ul li {
        flex-basis: 0;
    }
}

ul li a {
    display:          block;
    text-decoration: none;
}

html, body {
    width:  100%;
    height: 100%;
}

body {
    background-color: #999999;
    display:          flex;
    flex-direction:   column;
    margin:            0;
    overflow:          hidden;
}

main {
    margin:  0;
    padding: 0;
    display: flex;
    flex:    auto;
    overflow: auto;
}

main > nav {
    margin:      4px;
    padding:     5px;
    border:      1px solid #8888bb;
    border-radius: 7pt;
    background-color: #ccccff;
    min-width:    50px;
    flex:         auto;
}

main > section {
    margin:      4px;
    padding:     5px;
    border:      1px solid #8888bb;
    border-radius: 7pt;
```

```

    background-color: #ccccff;
    display:         flex;
    flex-direction:  column;
    min-width:       50px;
    overflow:        hidden;
}

main > section > section {
    display:  flex;
    flex-flow: column;
    flex:    auto;
    overflow: hidden;
}

header, footer {
    margin:      4px;
    padding:     5px;
    min-height:  50px;
    border:      1px solid #eebb55;
    border-radius: 7pt;
    background-color: #ffeebb;
}

#s1 {
    flex:      auto;
    flex-flow: column;
    overflow:  hidden;
}

#s1 > article, #s2 {
    overflow: auto;
}

.verticalSplitter {
    border-left: 1px solid grey;
    width:      1px;
    min-width:  1px;
    cursor:     col-resize;
}

.horizontalSplitter {
    border-top: 1px solid grey;
    height:    1px;
    min-height: 1px;
    cursor:    row-resize;
}

/* Too narrow to support three columns */
@media all and (max-width: 640px) {
    main {
        flex-flow: column;
    }
}

```

```

    }

    main > nav, header, footer {
        height: 50px;
    }

    .horizontalSplitter, .verticalSplitter {
        display: none;
    }
}
</style>
<script src=flextest5.js></script>
</head>
<body>
<nav>
    <ul>
        <li><a href=#>Item 1</a></li>
        <li><a href=#>Item 2</a></li>
        <li><a href=#>Item 3</a></li>
        <li><a href=#>Item 4</a></li>
        <li><a href=#>Item 5</a></li>
        <li><a href=#>Item 6</a></li>
    </ul>
</nav>
<main>
    <nav>nav</nav>
    <div class=verticalSplitter></div>
    <section>
        <section id=s1>
            <header>Header s1</header>
            <article><?php require 'flextest4.txt'; ?></article>
        </section>
        <div id=hs1 class=horizontalSplitter></div>
        <section id=s2><?php require 'flextest4.txt'; ?></section>
    </section>
</main>
<footer>footer</footer>
</body>
</html>

```

Here is a hamburger icon menu example:

[students.btsi.lu/evegi144/WAD/CSS3/responsive3.php](https://students.btsi.lu/evegi144/WAD/CSS3/responsive3.php)

```

<!DOCTYPE html>
<html lang=en>
<head>
    <meta charset=UTF-8>
    <meta name=viewport content="width=device-width, initial-scale=1.0">
    <title>Responsive layout example 3</title>

```

```
<style>
/* Inspiration from http://codepen.io/ricardozea/pen/OPaRZO */
html { /* http://www.paulirish.com/2012/box-sizing-border-box-ftw */
  box-sizing: border-box;
}

*, *:before, *:after {
  box-sizing: inherit;
}

#menu-button {
  padding:      0 0.2em 0.2em 0.2em;
  background:   #f6f6f6;
  text-decoration: none;
  color:        #333;
  cursor:       pointer;
  font-size:    3em;
}

#menu-button.active {
  background-color: #333;
  color:           #fff;
}

#menu {
  overflow:    hidden;
  max-height:  0;
  padding:     0;
  clear:       both;
  transition:  all .3s ease-out;
}

#menu.active {
  max-height: 17em;
}

#menu ul {
  margin:      0;
  padding:     0;
  list-style-type: none;
  border:      1px #999 dotted;
  border-bottom: none;
  text-align:  center;
}

#menu li a {
  display:     block;
  padding:     1em;
  border-bottom: 1px #999 dotted;
  text-decoration: none;
  color:       #2963BD;
}
```

```

    background-color: #fff;
}

@media (min-width: 40em) {
    #menu-button {
        display: none;
    }

    #menu {
        max-height: inherit;
    }

    #menu ul {
        background: #fff;
    }

    #menu li {
        display: inline-block;
        margin: 0 .2em;
    }
}
</style>
<script>
    'use strict';

    const init = () => {
        document.querySelector('button').addEventListener('click', () => {
            document.querySelector('button').classList.toggle('active');
            document.querySelector('#menu').classList.toggle('active');
        });
    };

    window.addEventListener('load', init);
</script>
</head>
<body>
    <button id=menu-button></button>
    <nav id=menu>
        <ul>
            <li><a href=#>Item 1</a></li>
            <li><a href=#>Item 2</a></li>
            <li><a href=#>Item 3</a></li>
            <li><a href=#>Item 4</a></li>
            <li><a href=#>Item 5</a></li>
        </ul>
    </nav>
    <main>
        <section><?php require 'flextest4.txt'; ?></section>
    </main>
</body>
</html>

```

## Responsive images

[www.w3schools.com/css/css\\_rwd\\_images.asp](http://www.w3schools.com/css/css_rwd_images.asp)

[css-tricks.com/responsive-images-css](http://css-tricks.com/responsive-images-css)

[developer.mozilla.org/en-US/docs/Learn/HTML/Multimedia\\_and\\_embedding/Responsive\\_images](http://developer.mozilla.org/en-US/docs/Learn/HTML/Multimedia_and_embedding/Responsive_images)

[internetingishard.com/html-and-css/responsive-images](http://internetingishard.com/html-and-css/responsive-images)

## Fluid typography

[www.smashingmagazine.com/2016/05/fluid-typography](http://www.smashingmagazine.com/2016/05/fluid-typography)

[css-tricks.com/viewport-sized-typography](http://css-tricks.com/viewport-sized-typography)

[stackoverflow.com/questions/27992881/max-height-x-doesnt-work-on-chrome](http://stackoverflow.com/questions/27992881/max-height-x-doesnt-work-on-chrome)

## Navigation menus

Menu1 Menu2

Submenu1 Subsubmenu1

Subsubmenu2 Subsubsubmenu1

Subsubsubmenu2 Subsubsubsubmenu1

Subsubsubmenu3

[students.btsi.lu/evegi144/WAD/CSS3/navigationmenu1.html](http://students.btsi.lu/evegi144/WAD/CSS3/navigationmenu1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Navigation Menu</title>
    <meta charset=UTF-8>
    <style>
      nav {
        margin:      0;
        white-space: nowrap;
      }

      nav ul {
        padding:      0;
        margin:      0;
        list-style-type: none;
      }

      nav ul li {
        display: inline-block;
        cursor: pointer;
      }

      nav ul li ul {
        position: absolute;
```

```

        display: none;
    }

    nav > ul > li ul li, nav > ul > li:hover > ul {
        display: block;
    }

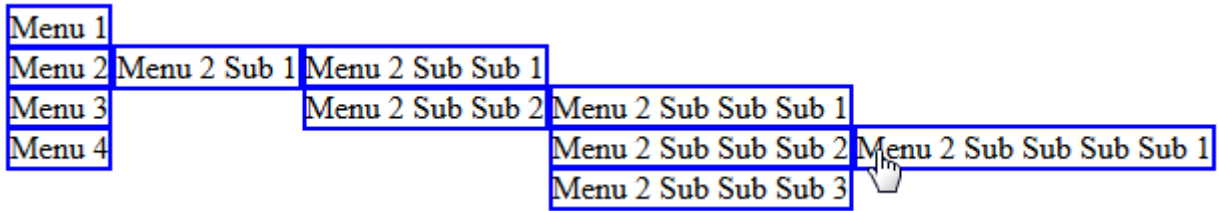
    nav > ul > li ul li:hover > ul {
        display: inline-block;
    }
</style>
</head>
<body>
<nav>
    <ul>
        <li>
            Menu1
            <ul>
                <li>
                    Submenu1
                    <ul>
                        <li>Subsubmenu1</li>
                    </ul>
                </li>
            </ul>
        </li>
        <li>
            Menu2
            <ul>
                <li>
                    Submenu1
                    <ul>
                        <li>Subsubmenu1</li>
                        <li>
                            Subsubmenu2
                            <ul>
                                <li>Subsubsubmenu1</li>
                                <li>
                                    Subsubsubmenu2
                                    <ul>
                                        <li>Subsubsubsubmenu1</li>
                                    </ul>
                                </li>
                                <li>Subsubsubsubmenu3</li>
                            </ul>
                        </li>
                    </ul>
                </li>
            </ul>
        </li>
    </ul>
</nav>

```

```

</nav>
</body>
</html>

```



[students.btsi.lu/evegi144/WAD/CSS3/navigationmenu2.html](http://students.btsi.lu/evegi144/WAD/CSS3/navigationmenu2.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Navigation Menu</title>
    <meta charset=UTF-8>
    <style>
      nav a {
        text-decoration: none;
        cursor:          pointer;
        border:           2px solid blue;
      }

      nav ul {
        padding:    0;
        margin:    0;
        list-style: none;
      }

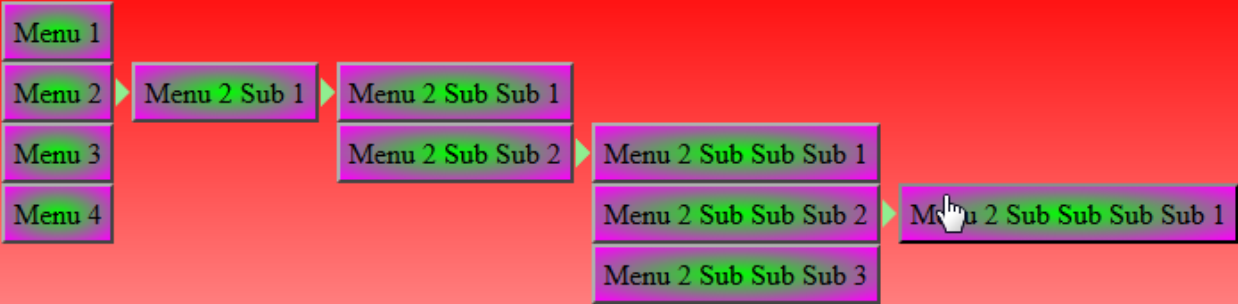
      nav > ul > li {
        float:      left;
        clear:      left;
        white-space: nowrap;
      }

      nav ul li ul {
        position:   absolute;
        display:    inline-block;
        visibility: hidden;
      }

      nav li:hover > ul {
        visibility: visible;
      }
    </style>
  </head>
  <body>
    <nav>

```

```
<ul>
  <li>
    <a>Menu 1</a>
    <ul>
      <li>
        <a>Menu 1 Sub 1</a>
        <ul>
          <li><a>Menu 1 Sub Sub 1</a></li>
        </ul>
      </li>
    </ul>
  </li>
  <li>
    <a>Menu 2</a>
    <ul>
      <li>
        <a>Menu 2 Sub 1</a>
        <ul>
          <li><a>Menu 2 Sub Sub 1</a></li>
          <li>
            <a>Menu 2 Sub Sub 2</a>
            <ul>
              <li><a>Menu 2 Sub Sub Sub 1</a></li>
              <li>
                <a>Menu 2 Sub Sub Sub 2</a>
                <ul>
                  <li><a>Menu 2 Sub Sub Sub Sub 1</a></li>
                </ul>
              </li>
            </ul>
          </li>
          <li><a>Menu 2 Sub Sub Sub 3</a></li>
        </ul>
      </li>
    </ul>
  </li>
  <li>
    <a>Menu 3</a>
  </li>
  <li>
    <a>Menu 4</a>
  </li>
</ul>
</nav>
</body>
</html>
```

[students.btsi.lu/evegi144/WAD/CSS3/navigationmenu3.html](https://students.btsi.lu/evegi144/WAD/CSS3/navigationmenu3.html)

```
<!DOCTYPE html>
<html lang=en>
<head>
<title>Navigation Menu</title>
<meta charset=UTF-8>
<style>
    body {
        background: linear-gradient(red, white, blue) fixed;
    }

    nav {
        position: absolute;
        top: 0;
        left: 0;
        transition: 50s;
    }

    nav:hover {
        left: 500px;
        top: 500px;
    }

    nav a {
        text-decoration: none;
        cursor: pointer;
        padding: 5px;
        background: radial-gradient(rgb(0, 255, 0), rgb(255, 0, 255));
        display: inline-block;
        border: 2px outset black;
        /* stackoverflow.com/questions/826782/css-rule-to-disable-text-selection-highlighting */
        -webkit-touch-callout: none;
        -webkit-user-select: none;
        -khtml-user-select: none;
        -moz-user-select: none;
        -ms-user-select: none;
        /*user-select: none;*/
    }

```

```

}

nav a:hover {
    background-color: green;
}

nav ul {
    padding:    0;
    margin:    0;
    list-style: none;
}

nav > ul ul li {
    padding-left: 10px;
}

nav > ul ul li:before {
    content:    '';
    position:    absolute;
    left:        1px;
    top:         9px;
    border-bottom: 8px solid transparent;
    border-top:   8px solid transparent;
    border-left:  8px solid lightgreen;
}

nav > ul > li {
    float:      left;
    clear:      left;
    white-space: nowrap;
}

nav ul li ul {
    position: absolute;
    display: none;
}

nav > ul li:hover > ul {
    display: inline-block;
}
</style>
</head>
<body>
<nav>
<ul>
<li>
    <a>Menu 1</a>
    <ul>
        <li>
            <a>Menu 1 Sub 1</a>
        </li>
    </ul>

```

```

        <li><a>Menu 1 Sub Sub 1</a></li>
    </ul>
</li>
</ul>
</li>
<li>
    <a>Menu 2</a>
    <ul>
        <li>
            <a>Menu 2 Sub 1</a>
            <ul>
                <li><a>Menu 2 Sub Sub 1</a></li>
                <li>
                    <a>Menu 2 Sub Sub 2</a>
                    <ul>
                        <li><a>Menu 2 Sub Sub Sub 1</a></li>
                        <li>
                            <a>Menu 2 Sub Sub Sub 2</a>
                            <ul>
                                <li><a>Menu 2 Sub Sub Sub Sub 1</a></li>
                            </ul>
                        </li>
                    </ul>
                </li>
                <li><a>Menu 2 Sub Sub Sub 3</a></li>
            </ul>
        </li>
    </ul>
</li>
</ul>
</li>
<li>
    <a>Menu 3</a>
</li>
<li>
    <a>Menu 4</a>
</li>
</ul>
</nav>
</body>
</html>

```

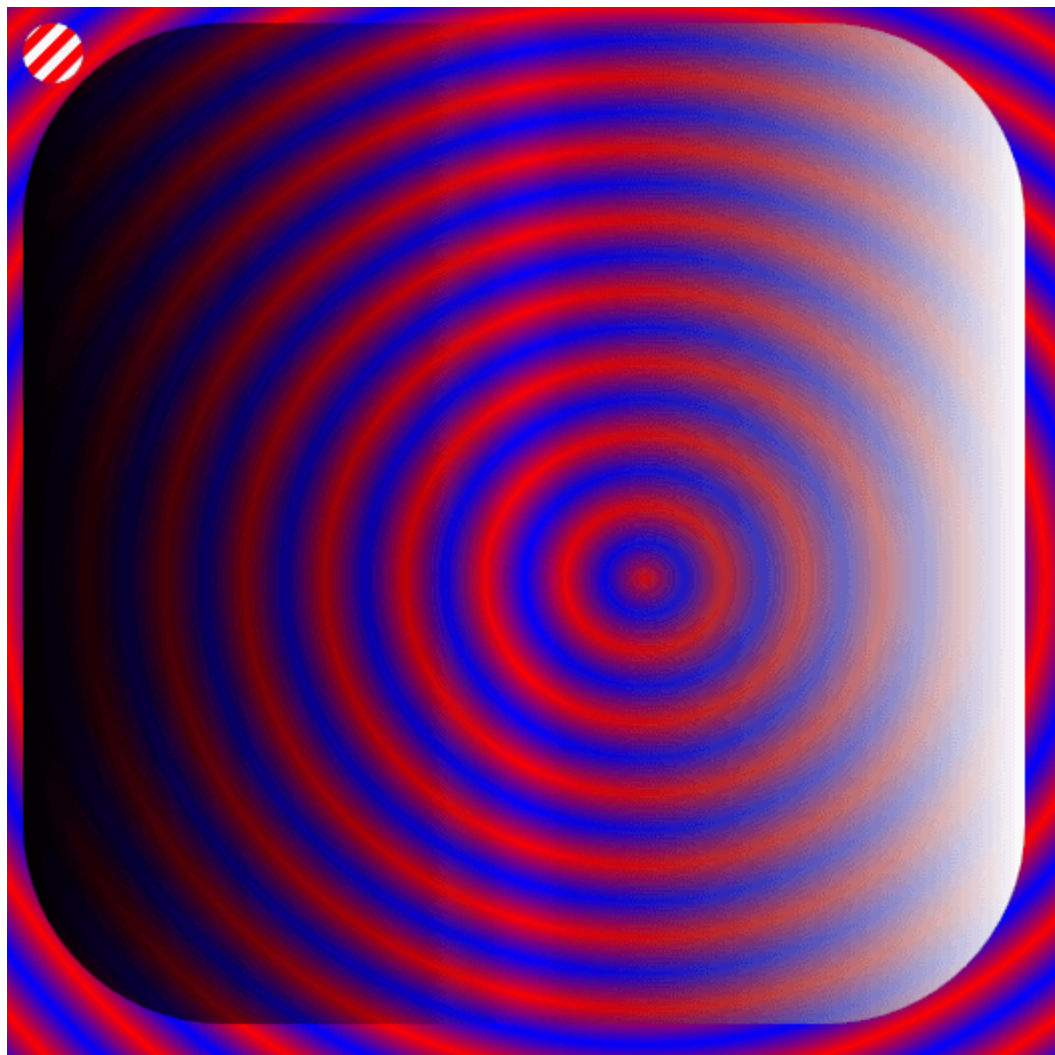
A useful tutorial and templates can be found at [cssmenu.com/blog/responsive-menu-tutorial](http://cssmenu.com/blog/responsive-menu-tutorial).

## 4.2.8. Make it look good

### Gradients

Gradients are a great alternative to standard images. [www.w3schools.com/css/css3\\_gradients.asp](http://www.w3schools.com/css/css3_gradients.asp) and [developer.mozilla.org/en-US/docs/Web/Guide/CSS/Using\\_CSS\\_gradients](http://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Using_CSS_gradients) provide excellent information and examples on the subject. The official specification can be found at [dev.w3.org/csswg/css-images-3](http://dev.w3.org/csswg/css-images-3).

Example:



[students.btsi.lu/evegi144/WAD/CSS3/gradient1.html](http://students.btsi.lu/evegi144/WAD/CSS3/gradient1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Gradient example 1</title>
    <meta charset=UTF-8>
    <style>
      body {
        background: repeating-radial-gradient(red, blue 20px, red 40px) fixed;
      }

      section {
        position: absolute;
        width: 500px;
        height: 500px;
        background: linear-gradient(90deg, black, transparent, white);
        border-radius: 100px;
      }

      article {
        width: 30px;
        height: 30px;
        background: repeating-linear-gradient(-45deg, red, red 5px, white 5px, white
10px);
        border-radius: 15px;
      }
    </style>
  </head>
  <body>
    <main>
      <section>
        <article>

        </article>
      </section>
    </main>
  </body>
</html>

```

An excellent gradient generator can be found at [www.colorzilla.com/gradient-editor](http://www.colorzilla.com/gradient-editor).

## Buttons

With CSS3 it's very easy to create great looking buttons without using pictures. There are many button generators available on the Web that produce nice results, for instance:

[www.cssbuttongenerator.com](http://www.cssbuttongenerator.com)

[dabuttonfactory.com](http://dabuttonfactory.com)

[css3button.net](http://css3button.net)

## Transformation and animation

For examples of what can be done, take a look at the following:

[www.marcofolio.net/css/css\\_animated\\_profile\\_cards.html](http://www.marcofolio.net/css/css_animated_profile_cards.html)

[www.marcofolio.net/css/animated\\_wicked\\_css3\\_3d\\_bar\\_chart.html](http://www.marcofolio.net/css/animated_wicked_css3_3d_bar_chart.html)

[3dtransforms.desandro.com](http://3dtransforms.desandro.com)

Details can be found at [www.w3schools.com/css/css3\\_animations.asp](http://www.w3schools.com/css/css3_animations.asp).

To find out which CSS properties can be animated, take a look at [developer.mozilla.org/en-US/docs/Web/CSS/CSS\\_animated\\_properties](http://developer.mozilla.org/en-US/docs/Web/CSS/CSS_animated_properties).

Here is a simpler example:



[students.btsi.lu/evegi144/WAD/CSS3/animation1.html](http://students.btsi.lu/evegi144/WAD/CSS3/animation1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Animation example 1</title>
```

```

<meta charset=UTF-8>
<style>
  body {
    background: repeating-radial-gradient(red, blue 20px, red 40px) fixed;
  }

  section {
    position:      absolute;
    width:         500px;
    height:        500px;
    background:    linear-gradient(90deg, black, transparent, white);
    animation:     sectionAnimation 5s infinite alternate;
    border-radius: 100px;
  }

  @keyframes sectionAnimation {
    from {
      left: 0;
      top: 0;
    }
    to {
      left: 500px;
      top: 100px;
    }
  }

  aside {
    position:      absolute;
    width:         30px;
    height:        30px;
    background:    repeating-linear-gradient(-45deg, red, red 5px, white 5px,
white 10px);
    border-radius: 15px;
    animation:     asideAnimation 5s infinite alternate;
  }

  /* Standard syntax */
  @keyframes asideAnimation {
    0% {
      left: 500px;
      top: 0;
    }

    50% {
      left: 250px;
      top: 300px;
    }

    100% {
      left: 0px;
      top: 100px;
    }
  }

```

```

    }
  }
</style>
</head>
<body>
  <main>
    <section></section>
    <aside></aside>
  </main>
</body>
</html>

```

## Fonts

[websitesetup.org/web-safe-fonts-html-css](http://websitesetup.org/web-safe-fonts-html-css)

[www.typewolf.com/google-fonts](http://www.typewolf.com/google-fonts)

[www.dvginteractive.com/serif-vs-sans-serif-how-to-increase-your-websites-readability-by-more-than-50-2](http://www.dvginteractive.com/serif-vs-sans-serif-how-to-increase-your-websites-readability-by-more-than-50-2)

@font-face

[www.w3schools.com/css/css3\\_fonts.asp](http://www.w3schools.com/css/css3_fonts.asp)

You can find lots of free fonts at [www.google.com/fonts](http://www.google.com/fonts). Also read [www.creativeblog.com/typography/free-web-fonts-1131610](http://www.creativeblog.com/typography/free-web-fonts-1131610).

[students.btsi.lu/evegi144/WAD/CSS3/font-face1.html](http://students.btsi.lu/evegi144/WAD/CSS3/font-face1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Using the @font-face rule</title>
    <meta charset=utf-8>
    <link href=http://fonts.googleapis.com/css?family=Shadows+Into+Light
rel=stylesheet>
    <style>
      body {
        font-family: 'Shadows Into Light', cursive;
      }
    </style>
  </head>
  <body>
    <main>
      <h1>Test header</h1>
    </main>
  </body>
</html>

```

## 4.2.9. Quiz

Take the w3schools quiz at [www.w3schools.com/quiztest/quiztest.asp?qtest=CSS](http://www.w3schools.com/quiztest/quiztest.asp?qtest=CSS) as a fun way to check you are as good as you think you are.

## 4.2.10. Tests

### Computer Shop



Create the following validated page:

[www.youtube.com/watch?v=C99FqKlnD1s](http://www.youtube.com/watch?v=C99FqKlnD1s)

The following information is not complete:

1. **body**: no margin and padding, background from black to grey, font color white, black shadow of 2 pixels h, v and blur.
2. **header**: height of 100 pixels.
3. **nav**: 150 pixels wide.
4. **main**: right padding of 10 pixels, overflow auto.
5. **footer**: 20 pixels high, font size half normal.
6. **ul** with no margin and padding.
7. **li** with 10 pixels padding above and below.
8. Hyperlinks with color gold and 1 pixel black shadow h, v and blur.
9. Navigation hyperlinks with font size twice normal.

10. **h1** with color gold and font size three times normal.
11. Definition term with color hex 22bb22, bottom border of 2 pixels blueviolet, top margin of 10 pixels and bottom margin of 5 pixels.
12. Table data items padding of 20 pixels.

You can copy paste the following text:

Computer Shop  
Buy

Contact us

Welcome to our shop.

We offer the following:

### Desktops

A desktop computer is a personal computer in a form intended for regular use at a single location desk/table due to its size and power requirements (cf. [http://en.wikipedia.org/wiki/Desktop\\_computer](http://en.wikipedia.org/wiki/Desktop_computer) Wikipedia).

### Laptops

A laptop or a notebook is a portable personal computer with a clamshell form factor, suitable for mobile use. There was a difference between laptops and notebooks in the past, but nowadays it has gradually died away. Laptops are commonly used in a variety of settings, including at work, in education, and for personal multimedia.

A laptop combines the components and inputs of a desktop computer, including display, speakers, keyboard and pointing device (such as a touchpad or a trackpad) into a single device. Most modern-day laptops also have an integrated webcam and a microphone. A laptop can be powered either from a rechargeable battery, or by mains electricity via an AC adapter. Laptop is a diverse category of devices and other more specific terms, such as rugged notebook or convertible, refer to specialist types of laptops, which have been optimized for specific uses. Hardware specifications change significantly between different types, makes and models of laptops (cf. <http://en.wikipedia.org/wiki/Laptop> Wikipedia).

Device

Brand

Price

LTAM

299.99

LTAM

349.99

2015 LTAM T0IF2

## Solution

[students.btsi.lu/evegi144/WAD/CSS3/Tests/ComputerShop/index.html](http://students.btsi.lu/evegi144/WAD/CSS3/Tests/ComputerShop/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Computer Shop</title>
    <meta charset=utf-8>
    <style>
      body {
        margin:      0;
        padding:      0;
        background:   linear-gradient(to bottom right, black, grey) fixed;
        color:        white;
        text-shadow:  2px 2px 2px black;
      }

      header {
        position: fixed;
        width:    100%;
        height:   100px;
      }

      nav {
        position: fixed;
        top:      100px;
        width:    150px;
      }

      main {
        position:    fixed;
        top:         100px;
        left:        150px;
        right:       0;
        bottom:      20px;
        text-align:   justify;
        padding-right: 10px;
        overflow:     auto;
      }

      footer {
        position:    fixed;
        bottom:      0;
        width:       100%;
        height:      20px;
        text-align:   center;
        font-size:    0.5em;
      }

      ul {
```

```

    margin: 0;
    padding: 0;
}

li {
    padding-top: 10px;
    padding-bottom: 10px;
}

a {
    text-decoration: none;
    color: gold;
    text-shadow: 1px 1px 1px black;
}

nav a {
    font-size: 2em;
}

h1 {
    text-align: center;
    font-size: 3em;
    color: gold;
}

dt {
    color: #22bb22;
    border-bottom: 2px dashed blueviolet;
    margin-top: 10px;
    margin-bottom: 5px;
}

th {
    text-align: center;
}

td {
    padding: 20px;
}
</style>
</head>
<body>
    <header>
        <h1>Computer Shop</h1>
    </header>
    <nav>
        <ul>
            <li><a href=#>Buy</a></li>
            <li><a href=#>Contact us</a></li>
        </ul>
    </nav>

```

```

<main>
  <h2>Welcome to our shop.</h2>
  We offer the following:
  <dl>
    <dt>Desktops</dt>
    <dd>A desktop computer is a personal computer in a form intended for regular
use at a
      single location desk/table due to its size and power requirements (cf. <a
      href=http://en.wikipedia.org/wiki/Desktop_computer
target=_blank>Wikipedia</a>).
    </dd>
    <dt>Laptops</dt>
    <dd>A laptop or a notebook is a portable personal computer with a clamshell
form
      factor, suitable for mobile use. There was a difference between laptops and
notebooks in the past, but nowadays it has gradually died away. Laptops are
commonly used in a variety of settings, including at work, in education, and
for
      personal multimedia.
      A laptop combines the components and inputs of a desktop computer, including
display,
      speakers, keyboard and pointing device (such as a touchpad or a trackpad)
into a
      single device. Most modern-day laptops also have an integrated webcam and a
microphone. A laptop can be powered either from a rechargeable battery, or
by mains
      electricity via an AC adapter. Laptop is a diverse category of devices and
other more
      specific terms, such as rugged notebook or convertible, refer to specialist
types of
      laptops, which have been optimized for specific uses. Hardware
specifications change
      significantly between different types, makes and models of laptops (cf. <a
      href=http://en.wikipedia.org/wiki/Laptop target=_blank>Wikipedia</a>).
    </dd>
  </dl>
  <table>
    <tr>
      <th>Device</th>
      <th>Brand</th>
      <th>Price</th>
    </tr>
    <tr>
      <td><img src=1432254774_mycomputer.png alt=Comp1></td>
      <td>LTAM</td>
      <td>299.99 &euro;</td>
    </tr>
    <tr>
      <td><img src=1432254808_Computer2.png alt=Comp2></td>
      <td>LTAM</td>
      <td>349.99 &euro;</td>
    </tr>
  </table>

```

```

    </tr>
  </table>
</main>
<footer>&copy; 2015 LTAM T0IF2</footer>
</body>
</html>

```

## Video Viewer



Create the validated site exactly as shown:

[www.youtube.com/watch?v=mMSt40R0Jgg](http://www.youtube.com/watch?v=mMSt40R0Jgg)

It consists of 2 HTML files (`index.html` and `viewer.html`) and one CSS file.

The following information is not complete:

1. Form data is sent to the file `viewer.html`.
2. The form box has a black shadow of 3 pixels h, v and blur.
3. The user name field is focused automatically when the page is loaded.
4. html and body have no margin and padding and use the whole browser window width and height.
5. `body`: repeating radial gradient from black to yellow 100px to white 200px, white text shadow of

2 pixels h, v and blur.

6. **nav**: 40 pixels high, full width.

7. **main**: display flex, full width. In index.html, there is no navigation, so main starts at the top. Therefore the login box is centered.

8. **footer**: 15 pixels high, font size half normal.

9. **ul** with 10 px horizontal and no vertical margin. No padding.

10. Navigation links with 10 pixel padding, 2 pixel golden border.

11. The form has automatic margin. The animation lasts 5 seconds and starts with 0 opacity.

12. Form inputs have 0.5 opacity and white text shadow of 2 pixels h, v, and blur.

13. The table takes 20% of the total width and has a margin of 5 pixels.

14. Table headings have red text color and a font size 25% bigger than normal.

15. The iframe takes 80% of the total width and also has a margin of 5 pixels.

You can copy paste the following text:

Login

Title

Module

Link

Windows 7 installation

SYSEX1

<https://www.youtube.com/embed/NP3cPmC-08A>

Computer Shop

HTSTA

<https://www.youtube.com/embed/C99FqKlnD1s>

T1IF Invaders

CLISS2

<https://www.youtube.com/embed/c--I9pod00s>

2015 WMOTU

## Solution

[students.btsi.lu/evegi144/WAD/CSS3/Tests/VideoViewer/index.html](https://students.btsi.lu/evegi144/WAD/CSS3/Tests/VideoViewer/index.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Video Viewer</title>
    <meta charset=utf-8>
    <link href=style.css rel=stylesheet>
    <style>
      main {
        top: 0;
      }
    </style>
  </head>
  <body>
    <main>
      <form action=viewer.html>
        <fieldset>
          <legend>Login</legend>
          <input placeholder="user name" required autofocus>
          <input type=password placeholder=password required>
          <button>Login</button>
        </fieldset>
      </form>
    </main>
    <footer>&copy; 2015 WMOTU</footer>
  </body>
</html>

```

[students.btsi.lu/evegi144/WAD/CSS3/Tests/VideoViewer/viewer.html](https://students.btsi.lu/evegi144/WAD/CSS3/Tests/VideoViewer/viewer.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Video Viewer</title>
    <meta charset=utf-8>
    <link href=style.css rel=stylesheet>
  </head>
  <body>
    <nav>
      <ul>
        <li><a href=index.html>Log out</a></li>
        <li><a href="mailto:t0if2@ltam.lu?subject=Information%20request">Contact us</a></li>
      </ul>
    </nav>
    <main>
      <table>
        <thead>
          <tr>

```

```

        <th>Title</th>
        <th>Module</th>
        <th>Link</th>
    </tr>
</thead>
<tbody>
    <tr>
        <td>Windows 7 installation</td>
        <td>SYSEX1</td>
        <td><a href=https://www.youtube.com/embed/NP3cPmC-08A
target=myFrame>View</a></td>
    </tr>
    <tr>
        <td>Computer Shop</td>
        <td>HTSTA</td>
        <td><a href=https://www.youtube.com/embed/C99FqKlnD1s
target=myFrame>View</a></td>
    </tr>
    <tr>
        <td>T1IF Invaders</td>
        <td>CLISS2</td>
        <td><a href=https://www.youtube.com/embed/c--I9pod00s
target=myFrame>View</a></td>
    </tr>
</tbody>
</table>
<iframe name=myFrame></iframe>
</main>
<footer>&copy; 2015 WMOTU</footer>
</body>
</html>

```

[students.btsi.lu/evegi144/WAD/CSS3/Tests/VideoViewer/style.css](https://students.btsi.lu/evegi144/WAD/CSS3/Tests/VideoViewer/style.css)

```

html, body {
    margin: 0;
    padding: 0;
    width: 100%;
    height: 100%;
}

body {
    background: repeating-radial-gradient(black, yellow 100px, white 200px) fixed;
    text-shadow: 2px 2px 2px white;
}

nav {
    position: fixed;
    top: 0;
    width: 100%;

```

```
height: 40px;
}

main {
  position: fixed;
  top: 40px;
  width: 100%;
  bottom: 15px;
  display: flex;
  overflow: auto;
}

footer {
  position: fixed;
  bottom: 0;
  width: 100%;
  height: 15px;
  text-align: center;
  font-size: 0.5em;
}

nav > ul {
  list-style: none;
  margin: 10px 0;
  padding: 0;
}

nav > ul > li {
  display: inline;
}

nav > ul > li > a {
  padding: 10px;
  background-color: black;
  border: 2px inset gold;
  color: gold;
}

nav > ul > li > a:hover {
  background-color: blue;
}

a {
  text-decoration: none;
}

form {
  margin: auto;
  box-shadow: 3px 3px 3px black;
  animation: formAnim 5s;
}
```

```
@keyframes formAnim {
  from {
    opacity: 0;
  }
}

form input {
  opacity: 0.5;
  text-shadow: 2px 2px 2px white;
}

table {
  width: 20%;
  margin: 5px;
}

iframe {
  border: none;
  width: 80%;
  margin: 5px;
}

th {
  color: red;
  font-size: 1.25em;
}
```

## 4.3. JavaScript

Why JS?

### 4.3.1. Introduction to programming

In order to tell a computer what to do, we use a special language, called a "programming" language. Like "human" languages, this is a set of instructions that are translated into codes that have specific meaning to the computer.

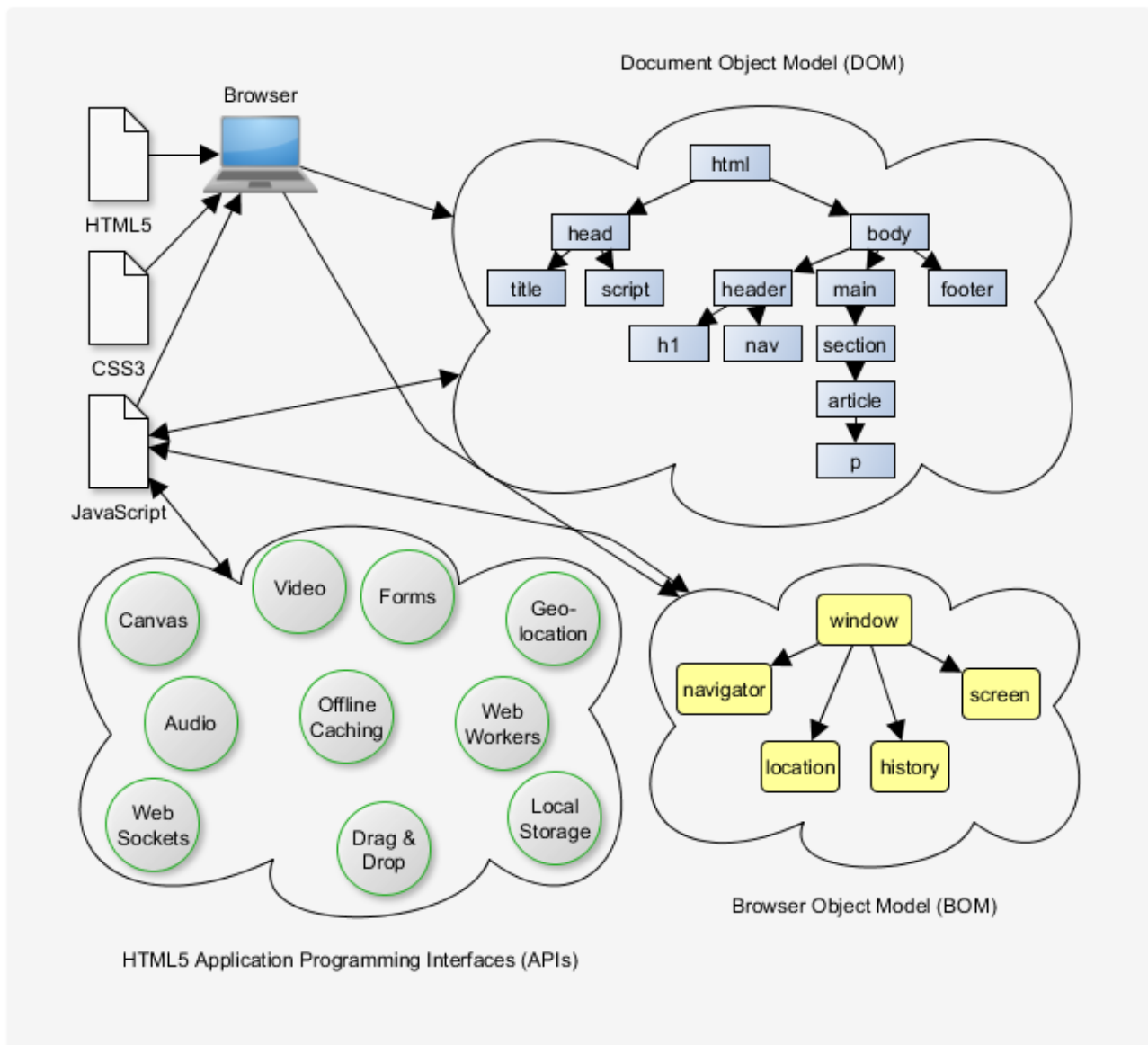
More formally, according to [en.wikipedia.org/wiki/Computer\\_programming](https://en.wikipedia.org/wiki/Computer_programming):

Computer programming (often shortened to programming) is a process that leads from an original formulation of a computing problem to executable programs. It involves activities such as analysis, understanding, and generically solving such problems resulting in an algorithm, verification of requirements of the algorithm including its correctness and its resource consumption, implementation (commonly referred to as coding) of the algorithm in a target programming language. Source code is written in one or more programming languages (such as C, C++, C#, Java, Python, Smalltalk, JavaScript, etc.). The purpose of programming is to find a sequence of instructions that will automate performing a specific task or solve a given problem. The process of programming thus often requires expertise in many different subjects, including knowledge of the application domain, specialized algorithms and formal logic.

#### **4.3.2. Getting started with JavaScript**

JavaScript is the programming language used to interact with the Document Object Model (DOM) that the browser creates from HTML and CSS files. It thereby allows the programmatic control of the web page's appearance and behavior on the client side. It is also increasingly used on the server side with Node.js. It is the foundation for the development of full fledged web applications.

The official standard can be found at [www.ecma-international.org/publications/standards/Ecma-262.htm](http://www.ecma-international.org/publications/standards/Ecma-262.htm).



JavaScript is based on the ECMAScript language specification, the latest version of which can be found at (cf. [www.ecma-international.org/ecma-262](http://www.ecma-international.org/ecma-262)). Browser support is excellent (cf. [kangax.github.io/compat-table/es2016plus](http://kangax.github.io/compat-table/es2016plus)).

A great tutorial is at [www.codecademy.com/courses/introduction-to-javascript](http://www.codecademy.com/courses/introduction-to-javascript).

A number of very insightful articles on JavaScript can be found at [javascript.crockford.com/javascript.html](http://javascript.crockford.com/javascript.html).

To get an in-depth overview of the latest developments in the real world application of JS, have a look at [The state of JavaScript](#).

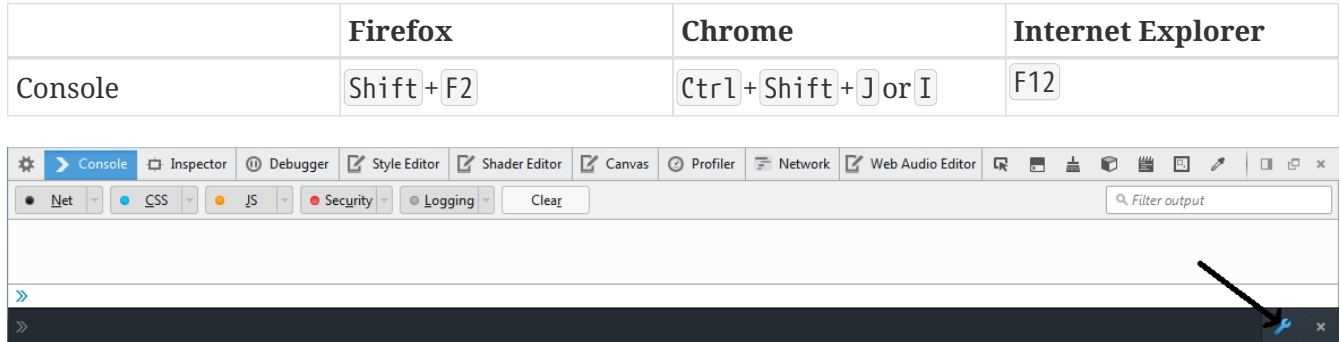
An excellent free online book on ECMAScript 5 can be found at [speakingjs.com/es5/index.html](http://speakingjs.com/es5/index.html).

To find out what's new in ECMAScript 6 take a close look at the following resources:

- [Top 10 ES6 Features Every Busy JavaScript Developer Must Know](#)
- [ES editor and examples](#)
- [Exploring ES6](#)

- [Learn ES6 by doing](#)
- [ES6 interactive guide](#)
- [Understanding ECMAScript 6](#)
- [es6-features.org](http://es6-features.org)

Before we get started, open the console of your browser.



You should always keep the console open, as any error messages will only be visible there.

### 4.3.3. Adding JavaScript to HTML documents

The HTML tag to include a script is, for obvious reasons, the `<script>` tag. The user has the possibility to disable JavaScript in his browser. There's nothing we can do about this except detect it using the `<noscript>` tag and inform the user that our app won't run without JavaScript. In Firefox, you can disable JavaScript by going to the page `about:config` and setting `javascript.enabled` to `false`. Now run the example and turn JavaScript back on and rerun the example:

[students.btsi.lu/evegi144/WAD/JS/noscript1.html](http://students.btsi.lu/evegi144/WAD/JS/noscript1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Detect whether JavaScript is disabled</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <noscript>Sorry, but this application requires JavaScript to run!</noscript>
    </main>
  </body>
</html>
```

### External JavaScript

The most common approach is to create a separate file with the extension `.js` (the extension is not mandatory, but recommended) and include it in the HTML file. Here's an example:

[students.btsi.lu/evegi144/WAD/JS/js1.html](https://students.btsi.lu/evegi144/WAD/JS/js1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Using external JS</title>
    <meta charset=utf-8>
    <script src=js1.js></script>
  </head>
  <body></body>
</html>
```

[students.btsi.lu/evegi144/WAD/JS/js1.js](https://students.btsi.lu/evegi144/WAD/JS/js1.js)

```
"use strict";

alert('Hello world!'); // Display a message box with the text 'Hello world!'.
```

The strict mode `"use strict";` instruction turns on strict mode (cf. [Strict mode](#)), which we should always do in order to make our life easier and to write better quality code. You can put this instruction into the default JavaScript template of your IDE, so that you do not have to type it in every time (cf. [NetBeans templates](#) or [PhpStorm templates](#)).

The `<script>` tag can be placed into the document's head or body. Remember that your browser processes an HTML document from top to bottom. So if the script is put into the head, it is loaded and executed before the HTML body exists. If the script tries to access HTML elements, it will fail, as shown in this example (remember, you need to have the console open to see the error):

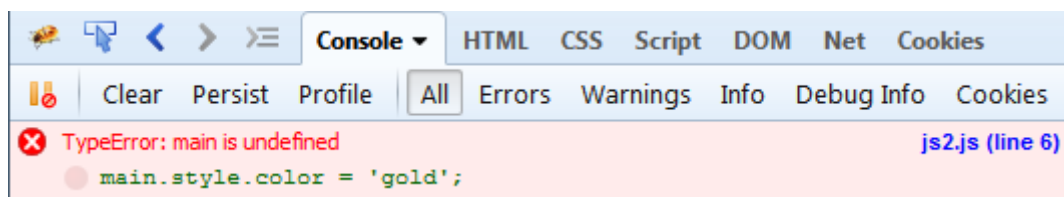
[students.btsi.lu/evegi144/WAD/JS/js2.html](https://students.btsi.lu/evegi144/WAD/JS/js2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Using external JS the wrong way</title>
    <meta charset=utf-8>
    <script src=js2.js defer></script>
  </head>
  <body><main></main></body>
</html>
```

```
"use strict";

// Save the first (and only) main element in a constant named main.
const main = document.getElementsByTagName('main')[0];
// Set the text color of the main element to gold.
main.style.color = 'gold';
// Set the background color of the main element to black.
main.style.backgroundColor = 'black';
// Set the text 'Hello world!' as content of the main element.
main.innerHTML = 'Hello world!';
```

Here is the error message you should see:



One solution to this problem is to include the script at the end of the HTML document, but this is not recommended, as it is difficult to see which script gets included. A cleaner approach is to have the script loaded in the head of the document but put all instructions that should be executed immediately into a function (don't worry, we'll look at those in detail in [JS functions](#), but for now you just need to know that a function is a series of instructions that get executed when we "call" the function using its name) which gets run when the browser has either finished loading and parsing the initial HTML document, without waiting for stylesheets, images, and subframes to finish loading or when the full document has been loaded and parsed. To achieve the former, we use the `DOMContentLoaded` (cf. [developer.mozilla.org/en/docs/Web/Events/DOMContentLoaded](https://developer.mozilla.org/en/docs/Web/Events/DOMContentLoaded)) and for the latter the `load` event (cf. [Events](#) where we talk about events in detail):

[students.btsi.lu/evegi144/WAD/JS/js3.html](https://students.btsi.lu/evegi144/WAD/JS/js3.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Using external JS the right way</title>
    <meta charset=utf-8>
    <script src=js3.js></script>
  </head>
  <body><main></main></body>
</html>
```

[students.btsi.lu/evegi144/WAD/JS/js3.js](https://students.btsi.lu/evegi144/WAD/JS/js3.js)

```

"use strict";

// We declare but do not execute the function.
const init = () => {
  const main = document.querySelector('main'); // Get the main element.
  main.style.color = 'gold'; // Set text color to gold.
  main.style.backgroundColor = 'black'; // Set background color to black.
  main.innerHTML = 'Hello world!'; // Set main content.
};

/* Only after the browser has finished loading and parsing the initial HTML document,
without
waiting for stylesheets, images, and subframes or the whole document,
meaning that the DOM is available, will the init function be called.
There are 2 events we can use and 2 ways to do this. The first one is the preferred
one, as it
allows to install several listeners for the same event. The second one is shorter. */
window.addEventListener('DOMContentLoaded', init);
//window.addEventListener('load', init);
//window.onload = init;

```

### async and defer

From [www.w3schools.com/tags/att\\_script\\_async.asp](http://www.w3schools.com/tags/att_script_async.asp):

The async attribute is only for external scripts (and should only be used if the src attribute is present).

Note: There are several ways an external script can be executed:

- If async is present: The script is executed asynchronously with the rest of the page (the script will be executed while the page continues the parsing).
- If async is not present and defer is present: The script is executed when the page has finished parsing.
- If neither async or defer is present: The script is fetched and executed immediately, before the browser continues parsing the page.

### Embedded JavaScript

If we have only a small script that we want to use in our document, we can embed it directly using the `<script>` tag, like so:

[students.btsi.lu/evegi144/WAD/JS/js4.html](http://students.btsi.lu/evegi144/WAD/JS/js4.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Using embedded JS</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      const sayHello = () => {
        alert('Hello world!');
      };

      window.addEventListener('load', sayHello);
    </script>
  </head>
  <body></body>
</html>

```

We can embed JavaScript in the head and/or in the body part of the document.

### Inline JavaScript

If we need to execute only one or very few instructions, for instance when a link is clicked, we can use JavaScript inline as an event handler (cf. [Events](#)).

Here's a simple illustration using a clickable link:

[students.btsi.lu/evegi144/WAD/JS/js5.html](https://students.btsi.lu/evegi144/WAD/JS/js5.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Using inline JS</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <a href=# onclick="alert('Hello world!');">
        If you click me I'll greet the world!</a>
      </main>
    </body>
  </html>

```

### Combinations

In the real world, you will often use combinations of some or all of the three methods, for instance an external script that defines functions, which are called via embedded or inline script.

Here is an example:

[students.btsi.lu/evegi144/WAD/JS/js6.html](https://students.btsi.lu/evegi144/WAD/JS/js6.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Combining external and inline JS</title>
    <meta charset=utf-8>
    <script src=js6.js></script> <!-- include an external script -->
  </head>
  <body>
    <main>
      <a href=# onclick=modifyMain();>If you click me I'll change the world!</a>
    </main>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/JS/js6.js](https://students.btsi.lu/evegi144/WAD/JS/js6.js)

```
"use strict";

const modifyMain = () => {
  const main = document.getElementsByTagName('main')[0];
  main.style.color = 'gold';
  main.style.backgroundColor = 'black';
  main.innerHTML = 'Hello world!';
};
```

#### 4.3.4. Comments

In order to make our scripts easier to understand, it is a good idea to add comments that explain what the purpose of a piece of code is if it is not obvious. Comments should not repeat the code or explain trivialities. JavaScript provides single line comments using `//` and multiline comments using `/* */`. You have already seen examples of both in [External JavaScript](#).

#### 4.3.5. Semicolons

From [www.ecma-international.org/ecma-262/#sec-automatic-semicolon-insertion](https://www.ecma-international.org/ecma-262/#sec-automatic-semicolon-insertion):

Most ECMAScript statements and declarations must be terminated with a semicolon. Such semicolons may always appear explicitly in the source text. For convenience, however, such semicolons may be omitted from the source text in certain situations. These situations are described by saying that semicolons are automatically inserted into the source code token stream in those situations.

In order to avoid needless errors, we'll always terminate the JS statements listed above with

semicolons.

### 4.3.6. Basic input and output

#### console.log

The `console` object provides access to the browser's debugging console, which we use to detect errors in our programs. For the details see [developer.mozilla.org/en-US/docs/Web/API/console](https://developer.mozilla.org/en-US/docs/Web/API/console) and [developer.mozilla.org/en-US/docs/Tools/Web\\_Console](https://developer.mozilla.org/en-US/docs/Tools/Web_Console).

We can also use colour as shown in [stackoverflow.com/questions/7505623/colors-in-javascript-console](https://stackoverflow.com/questions/7505623/colors-in-javascript-console).

Here is a simple example:

[students.btsi.lu/evegi144/WAD/JS/console1.html](https://students.btsi.lu/evegi144/WAD/JS/console1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Console usage example</title>
    <meta charset=utf-8>
    <script src=console1.js></script>
  </head>
  <body>
  </body>
</html>
```

#### document.write



**DO NOT USE THIS METHOD** (cf. [html.spec.whatwg.org/multipage/webappapis.html#document.write%28%29](https://html.spec.whatwg.org/multipage/webappapis.html#document.write%28%29)).

`document.write` writes a given text verbatim into the document at the current position. Example: [students.btsi.lu/evegi144/WAD/JS/document\\_write1.html](https://students.btsi.lu/evegi144/WAD/JS/document_write1.html)

```
"use strict";

document.write('<table><tr><th>Club</th><th>Score</th></tr>');
document.write('<tr><td>Bayern München</td><td>17</td></tr>');
document.write('<tr><td>FC Liverpool</td><td>15</td></tr></table>');
```

#### alert

In some of the previous examples we have already used the `alert` method. This method is part of the global `window` object, which we'll discuss in ). Normally we would have to write `window.alert` to use this method, but the browser considers `window` as the default object, thereby allowing us to simply write the method name. `alert` takes the text, quoted in `"` or `'`, as parameter and displays it

in a dialog box, which will block the screen until the user confirms the message.

### confirm

The `confirm` method, just like `alert`, displays a dialog box with a specified message, but with an OK and a Cancel button, so that the user has a choice. The method returns `true` if the user clicked OK and `false` if the user clicked Cancel. Example:

[students.btsi.lu/evegi144/WAD/JS/confirm1.html](https://students.btsi.lu/evegi144/WAD/JS/confirm1.html)

```
"use strict";

if (confirm('Do you really want to format your hard drive?') === true)
  console.log('OK, as you wish!');
else {
  console.log('Pweh!');
}
```

### prompt

`prompt` is used to get input from the user. The first parameter specifies the text to display. The second parameter is optional and can be used to display a default value. Take a look at the example and experiment with it. Note the `\'` in the prompt message. This is used to tell the browser that the apostrophe does not terminate the string but is to be displayed as such:

[students.btsi.lu/evegi144/WAD/JS/prompt1.html](https://students.btsi.lu/evegi144/WAD/JS/prompt1.html)

```
"use strict";

const input = prompt('Congratulations, you\'re the master of the universe. ' +
  'Please enter your name:', 'Donald Duck');
if (input) console.log(`Well done ${input}`);
```

## 4.3.7. Constants

If we want to use a specific immutable value throughout our script, we can declare it as a constant using `const`. Constants have block scope, i.e. they are only visible within the block that they are defined. If we define a constant in the main part of our script it will be visible everywhere. The advantage of using the constant compared to using the value directly is the ease with which we can change it.

For instance, if we decide that the player in our latest space shooter should have 5 lives instead of 3, we simply change the value of our constant. Without constant we would have to find all the occurrences of the value in our script and replace them individually, risking to accidentally change other values.

```
const LIVES = 3;
console.log(LIVES);
```

### 4.3.8. Variables

A program spends most of its time manipulating some kind of data, e.g. integer and decimal (or floating point) numbers, text (called string) or other, potentially more complex, data types. The information we need, for internal calculations or for display to the user, has to be stored somehow. To store data, we use **variables**. A variable is simply a place in computer memory where some information is stored.

To work with a variable we first need to declare it using the **let** or **var** keyword. The former has block scope whereas the latter has function scope. For an illustration of the difference, see [let vs var](#). For a more detailed explanation with examples see [developer.mozilla.org/en/docs/Web/JavaScript/Reference/Statements/let](https://developer.mozilla.org/en/docs/Web/JavaScript/Reference/Statements/let). After declaration the variable value is undefined. We can store some information in it using the **=** operator. Let's take a look at an example:

[students.btsi.lu/evegi144/WAD/JS/var1.html](https://students.btsi.lu/evegi144/WAD/JS/var1.html)

```
"use strict";

// We want to recalculate the health of our spaceship after it has been hit by an
// enemy bullet.
// We declare the variable for the initial health level and assign it a value of 100.
let health; // Declare a variable with block scope. Default value undefined.
alert(`Current health: ${health}`); //
health = 100; // We assign the value 100 to the variable.
alert(`Current health: ${health}`);
/* We declare a constant to store the damage caused by an enemy bullet hit. A constant
cannot
be changed. */
const SHOT_IMPACT = 20;

// *** Boom, we have been hit ***
// Now we calculate the new health value
health = health - SHOT_IMPACT;
alert(`We've been hit! Current health: ${health}`);
```

### Variable names

From [www.w3schools.com/js/js\\_variables.asp](http://www.w3schools.com/js/js_variables.asp):

- Variable names can contain letters, digits, underscores, and dollar signs.
- Variable names must begin with a letter.
- Variable names can also begin with **\$** and **\_**.
- Variable names are case sensitive (y and Y are different variables).

- Reserved words (like JavaScript keywords) cannot be used as variable names.

It is recommended to use camelCase, e.g. `firstFraction` instead of `first_fraction` or `firstfraction` etc. Please read the quick overview of JavaScript code conventions at [www.w3schools.com/js/js\\_conventions.asp](http://www.w3schools.com/js/js_conventions.asp).

### 4.3.9. Data types

JavaScript is a dynamically typed language, meaning that we can store any type of data in a variable or pass it as an argument to a function. This is in contrast to traditional programming languages such as C, C++, Java etc. Nevertheless, we need to be aware of the main data types that we will use. In our programs we can use the `typeof` operator to determine the current type of a variable, as in this example:

[students.btsi.lu/evegi144/WAD/JS/types1.html](http://students.btsi.lu/evegi144/WAD/JS/types1.html)

```
"use strict";

let x = 7; // integer number
console.log(`Current type of x: ${typeof x}`);
x = 5.6e-2; // decimal number
console.log(`Current type of x: ${typeof x}`);
x = 'abc sdfsd'; // string
console.log(`Current type of x: ${typeof x}`);
x = false; // boolean
console.log('Current type of x: ' + typeof x);
```

### Strings

A string is simply text, i.e. a sequence of one or more characters. Strings are always embedded within simple (') or double (") quotes. We can combine several strings using the `+` operator, as shown in the examples below. Strings have a number of properties and methods that make working with them much easier. Study [www.w3schools.com/jsref/jsref\\_obj\\_string.asp](http://www.w3schools.com/jsref/jsref_obj_string.asp) for a full reference with many examples.

Examples:

[students.btsi.lu/evegi144/WAD/JS/strings1.html](http://students.btsi.lu/evegi144/WAD/JS/strings1.html)

```

"use strict";

const s1 = 'This is a string';
/* Now we write the combination of two strings, which is a new string, into the body.
   Note that we use the + operator to concatenate, i.e. to combine, three strings. */
document.body.innerHTML = `<p>Content of constant s1: ${s1}</p>`;

let firstName = 'Donald', lastName = 'Duck';
document.body.innerHTML += `<p>Hello ${firstName} ${lastName}, how are you
today?</p>`;
firstName = 'Dagobert'; // We change the content of this variable.
document.body.innerHTML += `<p>Hello ${firstName} ${lastName}, how are you
today?</p>`;
document.body.innerHTML += `<p>Hello ${firstName} ${lastName.toUpperCase()}, how are
you
today?</p>`;

```

## Numbers

JavaScript does not distinguish between integers and decimals. Numbers are always stored as double precision floating point numbers, following the international IEEE 754 standard. This format stores numbers in 64 bits, where the number (the fraction) is stored in bits 0 to 51, the exponent in bits 52 to 62, and the sign in bit 63 (cf. [www.w3schools.com/js/js\\_numbers.asp](http://www.w3schools.com/js/js_numbers.asp)). This has practical implications, some of which are problematic. It leads for instance to mathematical imprecision, which you can confirm by typing `0.1 + 0.2` in the console. To avoid this problem, we can use [decimal.js](#). This problem is however common in almost all programming languages and not limited to JS. For details on the problem in different programming languages see [0.30000000000000004.com](http://0.30000000000000004.com).

Numbers have properties and methods that make working with them much easier. Study [www.w3schools.com/jsref/jsref\\_obj\\_number.asp](http://www.w3schools.com/jsref/jsref_obj_number.asp) for a full reference with many examples.

[students.btsi.lu/evegi144/WAD/JS/numbers1.html](http://students.btsi.lu/evegi144/WAD/JS/numbers1.html)

```

"use strict";

// We declare 4 variables and initialize them with 2 integer and 2 decimal numbers.
const i1 = 5000, i2 = 7345, d1 = 32.456787, d2 = -2.3e5; // exponential notation ->
2.3 * 10^5
/* We display their values as well as the results of simple operations.
   Note that the + operator automatically converts the value after the + into a string
   if
       the value in front of the + is a string.
   On the other hand, if there is nothing in front of the + operator and it is
   followed by a
       string, then this string will be converted to a number, if possible, otherwise to
   NaN
       (not a number).
   */

```

[illegible]

```

console.log(`i: ${i} i.toFixed(1): ${i.toFixed(1)}`);
i = i + 0.2;
console.log(`i: ${i} i.toFixed(1): ${i.toFixed(1)}`);
i = i + 0.2;
console.log(`i: ${i} i.toFixed(1): ${i.toFixed(1)}`);
i = i + 0.2;
console.log(`i: ${i} i.toFixed(1): ${i.toFixed(1)}`);
console.log('The same with a loop:');
// This is a for loop, which we'll discuss later.
for (let i = 0; i < 2; i += .2) console.log(`i: ${i} i.toFixed(1): ${i.toFixed(1)}`);

```

## Booleans

A boolean value is either **true** or **false** (cf. [www.w3schools.com/js/js\\_booleans.asp](http://www.w3schools.com/js/js_booleans.asp)). This data type is used in conditions and comparisons. Example:

[students.btsi.lu/evegi144/WAD/JS/booleans1.html](http://students.btsi.lu/evegi144/WAD/JS/booleans1.html)

```

"use strict";

const flag = true, x = 7, y = 34.6;
console.log(flag);
console.log(`x: ${x} y: ${y} => x is bigger than y: ${x > y}`);
console.log(`x - y + 20 is negative: ${x - y + 20 < 0}`);

```

## Conversions

JavaScript provides several options to convert between data types. To convert strings into numbers, we can put the **+** operator in front of the string. If the string cannot be fully converted into a number, the result will be NaN, meaning not a number. To transform user input into a number, it is usually preferable to use **parseInt** or **parseFloat**. These functions take a string as input and return an integer or a decimal (float, shortcut for floating point number). The advantage of these functions is that even if there are non-numerical characters in the string, as long as there is at least one digit at the beginning of the string, they will return a number and simply ignore the other characters. To convert anything into a string, we can use **toString**, which is a method that every object has by default. Let's look at a couple of examples:

[students.btsi.lu/evegi144/WAD/JS/conversions1.html](http://students.btsi.lu/evegi144/WAD/JS/conversions1.html)

```
"use strict";

const a1 = '66.76', a2 = '5'; // Declare and define 2 string variables.
console.log(a1 + a2); // Concatenation of 2 strings.
console.log(+a1 + +a2); // Conversion to numbers.
const s1 = prompt('Please enter your first number: '); // string 1
const s2 = prompt('Please enter your second number: '); // string 2
console.log(s1 + s2); // Concatenation of 2 strings.
console.log(parseInt(s1) + parseInt(s2)); // Conversion to integers.
console.log(parseFloat(s1) + parseInt(s2)); // Conversion to 1 float + 1 int.
console.log(parseInt(s1) + parseFloat(s2)); // Conversion to 1 int + 1 float.
console.log(parseFloat(s1) + parseFloat(s2)); // Conversion to floats.
const x1 = 123, x2 = 456; // Declare and define 2 number variables.
console.log(x1 + x2); // Addition of 2 numbers.
console.log(x1.toString() + x2); // Conversion to strings.
console.log(x1 + x2.toString()); // Conversion to strings.
console.log(x1.toString() + x2.toString()); // Conversion to strings.
const b1 = false, b2 = true; // Declare and define 2 booleans.
console.log(b1 + b2); // Addition of 2 booleans (0 + 1 -> 1).
console.log(b1.toString() + b2); // Conversion to strings.
console.log(b1 + b2.toString()); // Conversion to strings.
console.log(b1.toString() + b2.toString()); // Conversion to strings.
```

## Dates

JavaScript provides a very useful **Date** object to handle time and date information. Study

[www.w3schools.com/js/js\\_dates.asp](http://www.w3schools.com/js/js_dates.asp)

[www.w3schools.com/jsref/jsref\\_obj\\_date.asp](http://www.w3schools.com/jsref/jsref_obj_date.asp)

[developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Date](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Date)

## Math

The **Math** object offers a number of useful methods. Study [www.w3schools.com/js/js\\_math.asp](http://www.w3schools.com/js/js_math.asp) and [www.w3schools.com/jsref/jsref\\_obj\\_math.asp](http://www.w3schools.com/jsref/jsref_obj_math.asp).

## Regular expressions

Regular expressions provide a very powerful means to search for patterns in strings. Study [www.w3schools.com/js/js\\_regexp.asp](http://www.w3schools.com/js/js_regexp.asp) and [www.w3schools.com/js/js\\_regexp.asp](http://www.w3schools.com/js/js_regexp.asp).

## Maps and sets

### 4.3.10. Operators

Study the JavaScript operator documentation on w3schools: [www.w3schools.com/js/js\\_operators.asp](http://www.w3schools.com/js/js_operators.asp) and [www.w3schools.com/js/js\\_comparisons.asp](http://www.w3schools.com/js/js_comparisons.asp).

`Object.is` is preferable to `===` as explained in [www.jstips.co/en/javascript/why-you-should-use-Object.is\(\)-in-equality-comparison](http://www.jstips.co/en/javascript/why-you-should-use-Object.is()-in-equality-comparison).

## `eval`

`eval` takes a string argument and interprets it as JavaScript code.

[students.btsi.lu/evegi144/WAD/JS/eval1.html](http://students.btsi.lu/evegi144/WAD/JS/eval1.html)

```
"use strict";  
  
console.log(eval('if (5 < 7) true; else false;'));
```



For security and performance reasons usage of `eval` should be avoided (see [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/eval](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/eval)).

### 4.3.11. Conditional statements

See [www.w3schools.com/js/js\\_if\\_else.asp](http://www.w3schools.com/js/js_if_else.asp).

## `if`

Oftentimes the behavior of our program depends on a condition. For example, if we have developed a social network and we want to congratulate the user on his birthday, we first need to verify the condition of the user's birthday being true. Let's look at a couple of examples:

[students.btsi.lu/evegi144/WAD/JS/if1.html](http://students.btsi.lu/evegi144/WAD/JS/if1.html)

```

"use strict";

// We ask the user a very tough question and store his answer.
const correctAnswer = 7;
const userAnswer = parseInt(prompt('3 + 4 = '));
// Now we check whether his answer is correct or not and react accordingly.
if (userAnswer === correctAnswer) document.body.innerHTML += 'Well done!<br>';
else document.body.innerHTML += 'Please check your answer.<br>';

/* Let's assume the user's birthday is 1.2.1983.
   First we need to get the current date.
   Then we need to check whether day and month correspond
   to the user's birthday.
   If that's the case, then we will congratulate him, otherwise
   we'll tell him how far away his birthday is.
*/
// getMonth() returns a number between 0 and 11, so we need to subtract 1!
const userMonth = 1; // February, January would be 0.
const userDay = 1; // User's month and day of birth.
const date = new Date(); // Get the current date.
if (date.getMonth() === userMonth && date.getDate() === userDay)
    alert('Happy birthday!');
/* The else part is optional. In this case we don't need it, as we only want
   to do something if the condition is true.
*/

// Now we create a little script that displays the name of the current month.
const month = date.getMonth(); // We reuse the date variable created above.
if (month === 0) document.body.innerHTML += 'January<br>';
else if (month === 1) document.body.innerHTML += 'February<br>';
else if (month === 2) document.body.innerHTML += 'March<br>';
else if (month === 3) document.body.innerHTML += 'April<br>';
else if (month === 4) document.body.innerHTML += 'May<br>';
else if (month === 5) document.body.innerHTML += 'June<br>';
else if (month === 6) document.body.innerHTML += 'July<br>';
else if (month === 7) document.body.innerHTML += 'August<br>';
else if (month === 8) document.body.innerHTML += 'September<br>';
else if (month === 9) document.body.innerHTML += 'October<br>';
else if (month === 10) document.body.innerHTML += 'November<br>';
else if (month === 11) document.body.innerHTML += 'December<br>';

```

## switch

Another instruction that can be used if there are many alternatives to choose from is **switch**. A detailed explanation can be found at [www.w3schools.com/js/js\\_switch.asp](http://www.w3schools.com/js/js_switch.asp). Here's an example:

[students.btsi.lu/evegi144/WAD/JS/switch1.html](http://students.btsi.lu/evegi144/WAD/JS/switch1.html)

```
"use strict";
```

```
const date = new Date(); // Get the current date.
```

```
const month = date.getMonth(); // We reuse the date variable created above.
```

```
switch (month) {
```

```
  case 0:
```

```
    document.body.innerHTML += 'January<br>';
```

```
    break;
```

```
  /* If we leave this out, the following instructions will be executed,
     even though month is 0 and not 1.
  */
```

```
  */
```

```
  case 1:
```

```
    document.body.innerHTML += 'February<br>';
```

```
    break;
```

```
  case 2:
```

```
    document.body.innerHTML += 'March<br>';
```

```
    break;
```

```
  case 3:
```

```
    document.body.innerHTML += 'April<br>';
```

```
    break;
```

```
  case 4:
```

```
    document.body.innerHTML += 'May<br>';
```

```
    break;
```

```
  case 5:
```

```
    document.body.innerHTML += 'June<br>';
```

```
    break;
```

```
  case 6:
```

```
    document.body.innerHTML += 'July<br>';
```

```
    break;
```

```
  case 7:
```

```
    document.body.innerHTML += 'August<br>';
```

```
    break;
```

```
  case 8:
```

```
    document.body.innerHTML += 'September<br>';
```

```
    break;
```

```
  case 9:
```

```
    document.body.innerHTML += 'October<br>';
```

```
    break;
```

```
  case 10:
```

```
    document.body.innerHTML += 'November<br>';
```

```
    break;
```

```
  case 11:
```

```
    document.body.innerHTML += 'December<br>';
```

```
    break;
```

```
  default:
```

```
    document.body.innerHTML += 'Invalid month!<br>';
```

```
}
```

### 4.3.12. Loops

Loops are used to execute a block of, i.e. one or several, statements, several times. JavaScript provides the `for`, `while`, `for in` and `do while` loops. Each loop consists of a head and a body part.

#### `for`

The `for` loop head specifies the start value, a condition and what should be done after each iteration of the body. This type of loop is the preferred choice if we know the number of iterations in advance. We already saw an example in [Numbers](#). Let's study a couple more examples:

[students.btsi.lu/evegi144/WAD/JS/for1.html](https://students.btsi.lu/evegi144/WAD/JS/for1.html)

```

"use strict";

// A very basic loop.
for (var i = 0; i < 4; i++) {
    // Variable i is defined for the whole function, in this case the whole script as
    // there
    // is no function.
    document.body.innerHTML += `i: ${i}<br>`;
}

/*
    Let's calculate the sum of all numbers between a and b, except those that
    can be divided by 5.
*/
let a = 2, b = 123, sum = 0;
for (i = a; i <= b; i++) { // i was defined using var -> function scope
    if (i % 5 !== 0) {
        sum += i;
    }
}
console.log('sum: ' + sum);

console.log('Here is an illustration of what block scope means.');
```

```

// A very basic loop.
for (let j = 0; j < 4; j++) { // Variable j is only defined for this block.
    document.body.innerHTML += `i: ${i}<br>`;
}

/*
    Let's calculate the sum of all numbers between a and b, except those that
    can be divided by 5.
*/
sum = 0;
for (j = a; j <= b; j++) { // j was defined using let -> block scope
    // To solve this we would have to define j for this block
    using let j = a;
    if (j % 5 !== 0) {
        sum += j;
    }
}
console.log(`sum: ${sum}`);

```

## while

The **while** loop head only specifies the condition that needs to be true for the loop body to be executed. The condition can be any boolean expression, i.e. we do not necessarily need a counter variable. Let's look at an example:

[students.btsi.lu/evegi144/WAD/JS/while1.html](https://students.btsi.lu/evegi144/WAD/JS/while1.html)

```

"use strict";

const result = 2 + 2;
let input, count = 1; // Define variables.
/* As long as the user does not enter the correct answer, we loop.
   Given that prompt returns a string, we need to convert it to an integer.
   */
while ((input = parseInt(prompt('2 + 2 = '))) !== result) {
    document.body.innerHTML += 'Wrong answer, please try again.<br>';
    count++; // Increase the attempt counter.
}
console.log(`Correct, it took you ${count} attempt(s).`);

```

## for in

**for in** loops through the enumerable properties of an object and includes the enumerable properties of the prototype chain (cf. [Objects and classes](#)).

[students.btsi.lu/evegi144/WAD/JS/forin1.html](https://students.btsi.lu/evegi144/WAD/JS/forin1.html)

```

"use strict";

const car = {
    color: 'black',
    weight: 1500,
    length: 4.5
};

for (const prop in car) {
    console.log(`Property name: ${prop} property value: ${car[prop]}`);
}

```

You should not use **for in** to loop through an array, as the following example inspired by "Effective JavaScript" p. 132 shows:

[students.btsi.lu/evegi144/WAD/JS/forin2.html](https://students.btsi.lu/evegi144/WAD/JS/forin2.html)

```

"use strict";

const grades = [23, 45, 56, 43, 32, 29];
let total = 0;
for (const grade in grades) {
    total += grade;
}
let average = total / grades.length;
console.log(`Average: ${average}`); // 2057.5 -> wrong, but why?

// Let's add a debugging instruction to our loop, so that we can see what happens:
total = 0;
for (const grade in grades) {
    total += grade;
    console.log(`${grade} ${total}`);
}

// Now we see that for in loops through the KEYS of the array, not the VALUES.
// The correct approach is to use a for loop:
total = 0;
for (let i = 0; i < grades.length; i++) {
    total += grades[i];
    console.log(`${grades[i]} ${total}`);
}
average = total / grades.length;
console.log(`Average: ${average}`); // 38 -> correct

```

### do while

This is identical to the **while** loop, except that the body of **do while** is always executed at least once, as the condition is only checked after the execution instead of before.

## 4.3.13. Jumps and exceptions

Jump statements instruct the JavaScript interpreter to jump to a different source code location.

### Labeled statements

A JavaScript statement can be labeled by writing the label followed by a colon.

### break

Used alone, **break** causes the innermost enclosing loop or switch statement to exit immediately. If followed by a label, it terminates the enclosing statement that has the specified label. Take a close look at these examples:

[students.btsi.lu/evegi144/WAD/JS/break1.html](https://students.btsi.lu/evegi144/WAD/JS/break1.html)

```

"use strict";

const matrix = [
  [1, 3, 5, 7, 9], // sum 25
  [2, 2, 'a', 2, 2], // sum 8
  [3, 4, 5, 6, 7], // sum 25
  [9, 8, 7, 6, 5], // sum 35
  [4, 4, 4, 4, 4] // 20
];
let sum = 0, error = false;
outerloop: for (let i = 0; i < matrix.length; i++)
  for (let j = 0; j < matrix[0].length; j++) {
    if (isNaN(matrix[i][j])) {
      error = true;
      break outerloop;
    }
    sum += matrix[i][j];
  }
console.log(`Sum: ${sum} error: ${error}`);

sum = 0;
error = false;
for (let i = 0; i < matrix.length; i++)
  for (let j = 0; j < matrix[0].length; j++) {
    if (typeof matrix[i][j] !== 'number') {
      error = true;
      break;
    }
    sum += matrix[i][j];
  }
console.log(`Sum: ${sum} error: ${error}`);

```

## continue

**continue** is similar to **break**, except that it does not exit the loop but restarts it at the next iteration, i.e. it terminates only the current iteration, not the whole loop. Example:

[students.btsi.lu/evegi144/WAD/JS/continue1.html](https://students.btsi.lu/evegi144/WAD/JS/continue1.html)

```

"use strict";

const matrix = [
  [1, 3, 5, 7, 9], // sum 25
  [2, 2, 'a', 2, 2], // sum 8
  [3, 4, 5, 6, 7], // sum 25
  [9, 8, 7, 6, 5], // sum 35
  [4, 4, 4, 4, 4] // 20
];
let sum = 0, error = false;
outerloop: for (let i = 0; i < matrix.length; i++)
  for (let j = 0; j < matrix[0].length; j++) {
    if (typeof matrix[i][j] !== 'number') {
      error = true;
      continue outerloop;
    }
    sum += matrix[i][j];
  }
console.log(`Sum: ${sum} error: ${error}`);

```

## Exception handling

Often our programs rely on the user to provide reasonable input. But what happens if the user provides invalid input? For instance, we have a function that calculates the product of all given parameters, assuming they are all numbers. If one of the parameters is a string, our script might not perform as expected. That's not very professional and chances are, our users won't like it. A better way is to throw an exception using the **throw** instruction and catch it using **try catch finally**. In the **try** block we put the code that is at risk of breaking, for instance on wrong user input. In the **catch** block we put the code that should be executed if something went wrong in the **try** block, i.e. an exception occurred. The **finally** block is optional and will always be executed as the final block. JavaScript has a built-in **Error** object (cf. [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Error](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Error)), that serves to signal an error. Let's study an example:

[students.btsi.lu/evegi144/WAD/JS/exceptions1.html](https://students.btsi.lu/evegi144/WAD/JS/exceptions1.html)

```

"use strict";

const product = () => {
  let prod = 1;
  for (let i = 0; i < arguments.length; i++) {
    if (typeof arguments[i] !== 'number')
      throw new Error('All arguments must be numbers!');
    prod *= arguments[i];
  }
  return prod;
};

try {
  console.log(product(2, 34, 5)); // OK
  console.log(product(2, 'abc', 5)); // Wrong arg -> trouble.
}
catch (error) {
  console.error(error);
}
finally {
  console.log('Job done');
}

```

### 4.3.14. Functions

A function is a block of code that is executed by "calling" the function. Functions are of fundamental importance in software development, as they permit to reuse code, i.e. we solve a problem once and can then use the solution as often as we want without having to reinvent the wheel.

#### Normal functions

A function is defined using the **function** keyword, followed by parentheses **()**, followed by curly brackets **{}**. A function can take one or several parameters, which represent information that is given to the function by the caller.



Parameters of primitive types are passed by value, whereas objects are passed by reference.

Within the curly brackets we find the body of the function, i.e. the code that is executed when the function is called. A function can return an object using the **return** keyword. Please note that **return** does 2 things:

- Returns the given object, or nothing if no object is provided.
- Terminates the function. This means that any instructions following **return** will not be executed.

Normal functions have a built-in **arguments** object, which contains an array of the arguments used when the function was called.

Also note that a normal function can be called before its declaration. This is called hoisting.

Let's study a couple of examples:

[students.btsi.lu/evegi144/WAD/JS/functions1.html](https://students.btsi.lu/evegi144/WAD/JS/functions1.html)

```
"use strict";

// A simple sum function
function sum(a, b) {
  return a + b;
}
const alpha = 34, beta = 67.89;
console.log(`The sum of ${alpha} and ${beta} is ${sum(alpha, beta)}.`);

/* We define a function without parameters that adds a paragraph and returns
nothing.
*/
function sayHelloWorld() {
  const r = Math.floor(Math.random() * 256); // Generate random red component.
  const g = Math.floor(Math.random() * 256); // Random green component.
  const b = Math.floor(Math.random() * 256); // Random blue component.
  const p = document.createElement('p'); // Create new <p> element.
  p.style.color = 'blue'; // Set text and background colors.
  p.style.backgroundColor = 'rgb(' + r + ', ' + g + ', ' + b + ')';
  p.innerHTML = 'Hello world!'; // Set content.
  const main = document.querySelector('main'); // Get main element.
  main.appendChild(p); // Append the new <p> element to <main>.
}

// Now we call the function, i.e. we reuse the code to display the text 5 times.
for (let i = 1; i <= 5; i++) sayHelloWorld();

// Here we call the factorial function even before it's defined -> hoisting.
console.log(`The factorial of 7 is ${factorial(7)}.`);

/* The factorial function takes one parameter, a positive integer, and returns
the number's factorial. Remember the formula:  $n! = n * (n - 1) * \dots * 2$ .
*/
function factorial(x) {
  // If x is not a number, we return false.
  if (isNaN(x)) return false;
  x = x.toFixed(0); // Make sure x is an integer.
  let fact = 1; // In this variable we store the current result.
  for (let i = x; i >= 2; i--) fact *= i; // Loop to calculate factorial.
  return fact; // Return the result.
}

/* We can also define a function that works with a variable number of
parameters, a so called variadic function.
*/
```

```
function printArgs() {
  for (let i = 0; i < arguments.length; i++)
    console.log(`Argument ${i}: ${arguments[i]} of type ${typeof arguments[i]}`);
  console.dir(arguments);
}

printArgs('Hi', 5.6, 5, false);

// Optional named parameters
// cf. http://exploringjs.com/es6/ch_core-features.html#sec_param-defaults-core-feature
function printFromTo(values, {start=0, end=-1, step=1} = {}) {
  console.log(`Start: ${start}`);
  console.log(`Step: ${step}`);

  if (step === 0)
    console.dir(new Error('Infinite loops are not a good idea...'));
  else for (let i = start; i < end; i += step) console.log(values[i]);
}

printFromTo([1,2,3,4,5,6]);
printFromTo([1,2,3,4,5,6], {start: 1, end: 6, step: 0});
printFromTo([1,2,3,4,5,6], {start: 1, end: 6, step: 2});
```

## Arrow functions

From [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Functions/Arrow\\_functions](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Functions/Arrow_functions):

An arrow function expression has a shorter syntax than a function expression and does not bind its own `this`, `arguments`, `super`, or `new.target`. These function expressions are best suited for non-method functions, and they cannot be used as constructors.

An arrow function is created using `⇒`. It means that the value in the parentheses, if any, is passed as parameter to the block of code that the arrow is pointing to. An arrow function is defined like a variable.

Contrary to normal functions, which are read at compile-time, arrow functions are read at run-time. Therefore they must be defined in the code before they can be called.

Also contrary to a function, an arrow function does not have an `arguments` object (cf. point 18.a in [www.ecma-international.org/ecma-262/6.0/#sec-functiondeclarationinstantiation](https://www.ecma-international.org/ecma-262/6.0/#sec-functiondeclarationinstantiation)). Instead we can use so called rest parameters using the `...` operator, which can also be used with normal functions.

[students.btsi.lu/evegi144/WAD/JS/functions1b.html](https://students.btsi.lu/evegi144/WAD/JS/functions1b.html)

```
"use strict";

// A simple sum function
```

```

const sum = (a, b) => a + b;

const a = 34, b = 67.89;
console.log(`The sum of ${a} and ${b} is ${sum(a, b)}.`);

/* We define a function without parameters that adds a paragraph and returns
nothing.
*/
const sayHelloWorld = () => {
  const r = Math.floor(Math.random() * 256); // Generate random red component.
  const g = Math.floor(Math.random() * 256); // Random green component.
  const b = Math.floor(Math.random() * 256); // Random blue component.
  const p = document.createElement('p'); // Create new <p> element.
  p.style.color = 'blue'; // Set text and background colors.
  p.style.backgroundColor = 'rgb(' + r + ', ' + g + ', ' + b + ')';
  p.innerHTML = 'Hello world!'; // Set content.
  const main = document.querySelector('main'); // Get main element.
  main.appendChild(p); // Append the new <p> element to <main>.
};

// Now we call the function, i.e. we reuse the code to display the text 5 times.
for (let i = 1; i <= 5; i++) sayHelloWorld();

/* The factorial function takes one parameter, a positive integer, and returns
the number's factorial. Remember the formula:  $n! = n * (n - 1) * \dots * 2$ .
*/
const factorial = x => {
  // If x is not a number, we return false.
  if (typeof x !== 'number') return false;
  x = x.toFixed(0); // Make sure x is an integer.
  let fact = 1; // In this variable we store the current result.
  for (let i = x; i >= 2; i--) fact *= i; // Loop to calculate factorial.
  return fact; // Return the result.
};

// A function expression can only be called after its definition.
console.log('The factorial of 7 is ' + factorial(7) + '.');

const printArgs = (...args) => {
  for (let i = 0; i < args.length; i++)
    console.log(`Argument ${i}: ${args[i]} of type ${typeof args[i]}`);
  console.dir(args);
};

printArgs('Hi', 5.6, 5, false);

const printFromTo = (values, {start=0, end=-1, step=1} = {}) => {
  if (step === 0)
    console.dir(new Error('Infinite loops are not a good idea...'));
  else for (let i = start; i < end; i += step) console.log(values[i]);
};

```

```
printFromTo([1,2,3,4,5,6]);  
printFromTo([1,2,3,4,5,6], {start: 1, end: 6, step: 0});  
printFromTo([1,2,3,4,5,6], {start: 1, end: 6, step: 2});
```

A major difference between a normal function and an arrow function is that in the case of a normal function, the scope is local whereas in the case of an arrow function, the scope is the surrounding code.

From [hacks.mozilla.org/2015/06/es6-in-depth-arrow-functions/](https://hacks.mozilla.org/2015/06/es6-in-depth-arrow-functions/):



Use non-arrow functions for methods that will be called using the `object.method()` syntax. Those are the functions that will receive a meaningful `this` value from their caller. Use arrow functions for everything else.

The following example illustrates this difference (cf. [this](#)):

[students.btsi.lu/evegi144/WAD/JS/functions1c.html](https://students.btsi.lu/evegi144/WAD/JS/functions1c.html)

```
"use strict";  
  
function localScope() {  
  console.dir(this);  
}  
  
let globalScope = () => console.dir(this);  
  
localScope();  
globalScope();
```

For a more in-depth illustration of why this is very important, study [stackoverflow.com/questions/20279484/how-to-access-the-correct-this-inside-a-callback](https://stackoverflow.com/questions/20279484/how-to-access-the-correct-this-inside-a-callback).

## Variable scope

It is important to understand the concept of variable scope. A local variable, i.e. a variable that is declared inside the body of a function using the `let` or `var` keyword is called a local variable and only exists within the function body. A variable declared outside a function (with or without `let` or `var`) or a variable implicitly declared without `let` or `var` inside a function is a global variable, i.e. it exists even after the function execution has finished. Furthermore, if a global variable `x` is declared and inside a function we declare a local variable `x`, the local `x` will hide the global `x`, meaning that when we use `x` in the function, it will be the local and not the global one.



If you declare a variable without the `var` or `let` keyword, it will always be global, even if declared within a function! This can lead to errors that are hard to detect and should be avoided.

Let's illustrate this with a few examples:

[students.btsi.lu/evegi144/WAD/JS/functions2.html](https://students.btsi.lu/evegi144/WAD/JS/functions2.html)

```
"use strict";

let x = 7; // Declare and initialize a global variable.

f1(4); // We can call the normal function before it is defined -> function hoisting.
console.log('x: ' + x); // x is the GLOBAL variable.

function f1(a) {
  let x = a / 2;
  console.log(`x: ${x}`); // x is the LOCAL variable.
  // The global variable x is not usable here.
}

let f2 = () => {
  x++; // We modify the global variable x, which is a very bad idea.
}

console.log(`x: ${x}`); // x is the GLOBAL variable.
f2();
console.log(`x: ${x}`); // x is the GLOBAL variable.

let f3 = () => {
  let x = 23;
  console.log(`x: ${x}`); // x is the LOCAL variable.
}

f3();
// x is GLOBAL, the local one from f3 does not exist outside of f3.
console.log(`x: ${x}`);
```

Note in the examples above that we called normal function f1 before its declaration. This behavior is possible because of function hoisting, i.e. a normal function is read at run-time, as mentioned above.

**let vs var**

This first example does not work as expected:

[students.btsi.lu/evegi144/WAD/JS/lexical\\_scoping1.html](https://students.btsi.lu/evegi144/WAD/JS/lexical_scoping1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Lexical scoping 1</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const init = () => {
        const buttons = document.querySelectorAll('button');
        for (var i = 0; i < buttons.length; i++) {
          buttons[i].addEventListener('click', () => {
            console.log(`Button ${i + 1} clicked`);
          });
        }
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main>
      <button>B1</button>
      <button>B2</button>
      <button>B3</button>
    </main>
  </body>
</html>
```

The reason is that the scope of `var` is the entire enclosing function, so the value that will be used in the event handler is the last value of `i`, which is 3 in this case.

The second example solves this problem using `let`.

[students.btsi.lu/evegi144/WAD/JS/lexical\\_scoping2.html](https://students.btsi.lu/evegi144/WAD/JS/lexical_scoping2.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Lexical scoping 2</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const init = () => {
        const buttons = document.querySelectorAll('button');
        for (let i = 0; i < 3; i++) {
          buttons[i].addEventListener('click', () => {
            console.log(`Button ${i + 1} clicked`);
          });
        }
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main>
      <button>B1</button>
      <button>B2</button>
      <button>B3</button>
    </main>
  </body>
</html>

```

**let** has block scope, which means that the value used in the event handler is the value of `i` at that point and time in the block, as one would intuitively expect.

### Anonymous functions

We can pass a nameless function as a parameter to a function or assign it to a variable. This will be particularly useful when we deal with event handlers as we'll see later on.

Here is an example of an anonymous function:

[students.btsi.lu/evegi144/WAD/JS/functions3.html](https://students.btsi.lu/evegi144/WAD/JS/functions3.html)

```

"use strict";

// This function takes as parameter a function and a number.
function f1(f, x) {
  // We call the function x times, each time passing i as parameter.
  for (let i = 0; i < x; i++) f(i);
}

// We call f1 with an anonymous function and a number as parameters.
f1(function (x) {
  console.log(x);
}, 5);

// The same using an anonymous function expression.
f1(x => {
  console.log(x);
}, 5);

```

[students.btsi.lu/evegi144/WAD/JS/functions4.html](https://students.btsi.lu/evegi144/WAD/JS/functions4.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>x1</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <script>
      'use strict';

      const init = () => {
        let i = 7;
        setInterval(() => {
          console.log(i--);
        }, 500);
      };

      addEventListener('load', init);
    </script>
  </body>
</html>

```

## Self-invoking functions

[blog.mgechev.com/2012/08/29/self-invoking-functions-in-javascript-or-immediately-invoked-function-expression](https://blog.mgechev.com/2012/08/29/self-invoking-functions-in-javascript-or-immediately-invoked-function-expression)

[stackoverflow.com/questions/34589488/es6-immediately-invoked-arrow-function](https://stackoverflow.com/questions/34589488/es6-immediately-invoked-arrow-function)

## Asynchronous programming

### Promises

From [developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Using\\_promises](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Using_promises):

A Promise is an object representing the eventual completion or failure of an asynchronous operation.

<a href="https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Using_promises">developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Using_promises</a>
<a href="https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Promise">developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Promise</a>
<a href="https://hackernoon.com/transforming-callbacks-into-promises-and-back-again-e274c7cf7293">hackernoon.com/transforming-callbacks-into-promises-and-back-again-e274c7cf7293</a>
^ "

[students.btsi.lu/evegi144/WAD/JS/promise1.html](https://students.btsi.lu/evegi144/WAD/JS/promise1.html)

```

'use strict';

const httpGET = URL => {
  return new Promise((resolve, reject) => {
    const req = new XMLHttpRequest();
    req.addEventListener('load', e => {
      resolve(e.target.response);
    });
    req.addEventListener('error', e => {
      reject(new Error(e.target.statusText));
    });
    req.open('GET', URL);
    req.send();
  });
};

console.log('x1');
httpGET('https://students.btsi.lu').then(res => {
  console.log(`Contents: ${res}`);
},
err => {
  console.log(`Something went wrong: ${err}`);
}
);
console.log('x2');

const myfun4 = (x, resolve, reject) => {
  return new Promise((resolve, reject) => {
    setTimeout(() => {
      if (x > 5) resolve('All good');
      else reject('Not good');
    }, 2000);
  });
};

```

### async and await

From [hackernoon.com/6-reasons-why-javascripts-async-await-blows-promises-away-tutorial-c7ec10518dd9](https://hackernoon.com/6-reasons-why-javascripts-async-await-blows-promises-away-tutorial-c7ec10518dd9):

Async/await makes asynchronous code look and behave a little more like synchronous code. This is where all its power lies.

From [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/async\\_function](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/async_function):

Remember, the await keyword is only valid inside async functions. If you use it outside of an async function's body, you will get a SyntaxError.

[hackernoon.com/6-reasons-why-javascripts-async-await-blows-promises-away-tutorial-c7ec10518dd9](https://hackernoon.com/6-reasons-why-javascripts-async-await-blows-promises-away-tutorial-c7ec10518dd9)

[exploringjs.com/es2016-es2017/ch\\_async-functions.html](https://exploringjs.com/es2016-es2017/ch_async-functions.html)

[developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/async\\_function](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/async_function)

[ponyfoo.com/articles/understanding-javascript-async-await](https://ponyfoo.com/articles/understanding-javascript-async-await)

[tc39.github.io/ecmascript-asyncawait](https://tc39.github.io/ecmascript-asyncawait)

[stackoverflow.com/questions/42964102/syntax-for-async-arrow-function](https://stackoverflow.com/questions/42964102/syntax-for-async-arrow-function)

[medium.com/front-end-hacking/async-await-is-not-about-making-asynchronous-code-synchronous-ba5937a0c11e](https://medium.com/front-end-hacking/async-await-is-not-about-making-asynchronous-code-synchronous-ba5937a0c11e)

[exploringjs.com/es2016-es2017/ch\\_async-functions.html](https://exploringjs.com/es2016-es2017/ch_async-functions.html)

[students.btsi.lu/evegi144/WAD/JS/await1.html](https://students.btsi.lu/evegi144/WAD/JS/await1.html)

```
'use strict';

const httpGET = URL => {
  return new Promise((resolve, reject) => {
    const req = new XMLHttpRequest();
    req.addEventListener('load', e => {
      resolve(e.target.response);
    });
    req.addEventListener('error', e => {
      reject(new Error(e.target.statusText));
    });
    req.open('GET', URL);
    req.send();
  });
};

(async () => {
  console.log('x1');
  await httpGET('https://students.btsi.lu').then(res => {
    console.log(`Contents: ${res}`);
  },
  err => {
    console.log(`Something went wrong: ${err}`);
  }
);
  console.log('x2');
})();
```

[students.btsi.lu/evegi144/WAD/JS/await2.html](https://students.btsi.lu/evegi144/WAD/JS/await2.html)

```

'use strict';

const resolveAfter2Seconds = x => {
  return new Promise((resolve, reject) => {
    setTimeout(() => {
      const rand = Math.random();
      rand < 0.5 ? resolve(x) : reject(x);
    }, 2000);
  });
}

(async () => {
  try {
    console.log('x1');
    console.log(await resolveAfter2Seconds(10));
    console.log(await resolveAfter2Seconds(15));
    console.log('x2');
  } catch (e) {
    console.error(e);
  }
})();

```

### 4.3.15. Debugging

From [en.wikipedia.org/wiki/Debugging](https://en.wikipedia.org/wiki/Debugging):

Debugging is a methodical process of finding and reducing the number of bugs, or defects, in a computer program.

The short video at [youtu.be/0zWiq8FB3Xg](https://youtu.be/0zWiq8FB3Xg) shows how this works in practice using the following HTML document:

[students.btsi.lu/evegi144/WAD/JS/debug1.html](https://students.btsi.lu/evegi144/WAD/JS/debug1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Debugging example</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <script>
        "use strict";

        function f1() {
          // A very basic loop.
          for (let i = 0; i < 4; i++) {
            document.write('i: ' + i + '<br>');
          }
        }

        function f2() {
          // A very basic loop.
          for (var i = 0; i > 4; i++) {
            document.write('i: ' + i + '<br>');
          }
        }

        f1();
        f2();
      </script>
    </main>
  </body>
</html>

```

### 4.3.16. Arrays

Arrays are ordered value collections. Each value or element has its position, which is called index. The index of the first element is 0 and the position of the last element is the length of the array minus 1. The index is a 32 bit number, so the maximum number of elements in an array is  $2^{32} - 1$ . Arrays are dynamic and we do not have to specify any initial size.

It is essential to study [www.w3schools.com/js/js\\_arrays.asp](http://www.w3schools.com/js/js_arrays.asp), [www.w3schools.com/js/js\\_array\\_methods.asp](http://www.w3schools.com/js/js_array_methods.asp) and [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Array](http://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array).

#### Creating and iterating

We create an array either as an array literal, which is the recommended way, or using the Array constructor. The latter method is conducive to errors, as explained in the example:

[students.btsi.lu/evegi144/WAD/JS/array\\_creation.html](http://students.btsi.lu/evegi144/WAD/JS/array_creation.html)

```

"use strict";

const a1 = []; // Create an empty array.
const a2 = [1, 4, 2, 5, 3]; // Create an array with 5 numbers.
const a3 = ['Donald', 45, false]; // Create an array with 3 elements of different
types.
console.log(`Length of a1: ${a1.length}`);
console.log(`Length of a2: ${a2.length}`);
console.log(`Length of a3: ${a3.length}`);
console.log('Elements of a1:');
for (let i = 0; i < a1.length; i++) console.log(a1[i]);
console.log('Elements of a2:');
/* This is a dangerous way of looping through an array, as all enumerable properties
  of the array and its prototype object will be returned and the order is not
  guaranteed
  by the ECMAScript specification.
  */
for (let i in a2) console.log(a2[i]);
console.log('Elements of a3:');
/* If the order of the returned elements is not important, we can use the in operator
  by
  adding a test to exclude inherited enumerable properties.
  */
Object.getPrototypeOf(a3).x=7;
for (let i in a3) {
  //if (!a3.hasOwnProperty(i)) continue;
  console.log(a3[i]);
}

console.log('Elements of a3:');
// The most concise method is for of
for (const elem of a3) console.log(elem);

const a4 = new Array(); // Create an empty array.
/* You need to be careful when creating arrays with new. If there is only one
parameter, it
  indicates the number of elements. If there are several parameters, these will be the
  elements of the array. This adds unneeded complexity and encourages errors.
  */
const a5 = new Array(8); // Create an array with no elements and length 8.
// Create an array with 3 elements of different types.
const a6 = new Array('Donald', 'Duck', 45, true);
console.log(`Length of a4: ${a4.length}`);
console.log(`Length of a5: ${a5.length}`);
console.log(`Length of a6: ${a6.length}`);
console.log('Elements of a4:');
for (const elem of a4) console.log(elem);
console.log('Elements of a5 using a normal for:');
for (let i = 0; i < a5.length; i++) console.log(a5[i]);
// Note the different behavior: for/in simply skips undefined elements.

```

```
console.log('Elements of a5 using for/in:');
for (let i in a5) console.log(a5[i]);
console.log('Elements of a6:');
for (const elem of a6) console.log(elem);
```

It is important to note that arrays are objects. Therefore the **in** operator will use all enumerable properties of the object itself and its prototype, as described in [Testing properties](#). This will include methods that have been added to the prototype unless they are not enumerable. Furthermore, the ECMAScript specification does not fix the order in which the **in** operator returns elements. If element order is important, we should use a standard **for** loop to iterate through an array. If order does not matter, but we cannot guarantee that the array prototype is not polluted with enumerable properties, we should include a test to filter these, as shown in the examples above.



The preferred and most concise method to iterate over arrays is **for of**.

## Adding and deleting

[students.btsi.lu/evegi144/WAD/JS/array\\_add\\_delete.html](https://students.btsi.lu/evegi144/WAD/JS/array_add_delete.html)

```
"use strict";

const a = [];
a[5] = 'Donald';
/* The array has only one element, at index 5, but the length of the array is 6!
   This is called a sparse array. */
console.log(`Contents of array a: ${a} length: ${a.length}`);
a[1] = false;
console.log(`Contents of array a: ${a} length: ${a.length}`);
a.push('Duck'); // Add an element at the end.
console.log(`Contents of array a: ${a} length: ${a.length}`);
a[a.length] = 23e4; // Another way of adding an element at the end.
console.log(`Contents of array a: ${a} length: ${a.length}`);
delete a[a.length-1]; // Delete the last element. This does not alter the array
length!
console.log(`Contents of array a: ${a} length: ${a.length}`);
a.pop(); // Remove the last element and return it. This does adjust the length.
console.log(`Contents of array a: ${a} length: ${a.length}`);
console.log('Just removed ' + a.pop());
console.log(`Contents of array a: ${a} length: ${a.length}`);
```

## Multidimensional arrays

An array element can be another array. So we can easily create multidimensional arrays like so:  
[students.btsi.lu/evegi144/WAD/JS/array\\_multidim.html](https://students.btsi.lu/evegi144/WAD/JS/array_multidim.html)

```

"use strict";

const init = () => {
  const MATRIXLENGTH = 10; // 10 by 10
  // Create an empty 10 x 10 matrix.
  const matrix = new Array(MATRIXLENGTH);
  for (let i = 0; i < MATRIXLENGTH; i++) matrix[i] = new Array(MATRIXLENGTH);
  // Fill the matrix with random integers from [0, 99].
  for (let i = 0; i < MATRIXLENGTH; i++)
    for (let j = 0; j < MATRIXLENGTH; j++) matrix[i][j] = Math.floor(100 *
Math.random());
  // Display the matrix as a table.
  let table = '<table><caption>Random matrix</caption>';
  for (let i = 0; i < MATRIXLENGTH; i++) {
    table += '<tr>';
    for (let j = 0; j < MATRIXLENGTH; j++)
      table += '<td>${matrix[i][j]}</td>';
    table += '</tr>';
  }
  document.body.innerHTML += `${table}</table>`;
};

addEventListener('load', init);

```

## Methods

We have already seen `push` and `pop` methods. A complete reference with examples can be found at [www.w3schools.com/jsref/jsref\\_obj\\_array.asp](http://www.w3schools.com/jsref/jsref_obj_array.asp).

## Turning arrays into parameters

Using the spread operator `...` we can turn arrays into parameters. Why would we want to do this? Let's assume for instance, we want to determine the highest value in an array of numbers. `Math.max()` does not work on arrays. Using the spread operator, this is no problem: [students.btsi.lu/evegi144/WAD/JS/array\\_spread.html](http://students.btsi.lu/evegi144/WAD/JS/array_spread.html)

```

"use strict";

const arr = [];
for (let i = 0; i < 100; i++) arr.push(Math.random());
console.log(`The biggest number is ${Math.max(...arr)}`);

```

For further details see [exploringjs.com/es6/ch\\_parameter-handling.html#sec\\_spread-operator](http://exploringjs.com/es6/ch_parameter-handling.html#sec_spread-operator).

## 4.3.17. Template literals

From [exploringjs.com/es6/ch\\_template-literals.html](http://exploringjs.com/es6/ch_template-literals.html):

A template literal is a new kind of string literal that can span multiple lines and interpolate expressions (include their results).

Example: [students.btsi.lu/evegi144/WAD/JS/template\\_literals1.html](https://students.btsi.lu/evegi144/WAD/JS/template_literals1.html)

```
'use strict';

const firstName = prompt('Please enter your first name');
const lastName = prompt('Please enter your last name');
alert(`Hello ${firstName} ${lastName}, how are you?`);
```

The literal itself is delimited by backticks (`), the interpolated expressions inside the literal are delimited by `\${` and `}`. Template literals always produce strings.

There's much more to template literals, for the details please check the link above.

### 4.3.18. Objects and classes

In JavaScript, anything that is not a string, number, boolean, null or undefined is an object. Objects are collections of properties (cf. [www.ecma-international.org/ecma-262/#sec-object-type](https://www.ecma-international.org/ecma-262/#sec-object-type)). They allow us to regroup data and functions that belong together under one name and to reuse them. One purpose is the representation of real world objects. A good introduction can be found at [developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Working\\_with\\_Objects](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Working_with_Objects).

JS comes with a number of standard built-in objects, details of which can be found at [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects).

#### Prototypes



If you have traditional object-oriented programming experience, e.g. with C++ or Java, it is imperative to understand that JS is not class- but prototype-based. To get a good understanding of what the difference is and why it matters study [You Don't Know JS](#).

Most objects are derived from another object. This other object is called prototype in JavaScript. **Object** is the basic object in JavaScript from which normally all other objects inherit, directly or indirectly. To determine the prototype of an object, we use **Object.getPrototypeOf()**, where we pass the object as parameter.

#### Object creation

There are three ways to create objects.

##### as literals

From "JavaScript The Definitive Guide" p. 117: "An object literal is a comma-separated list of colon-

separated name:value pairs, enclosed within curly braces." **this** refers to the object itself, whose property we want to use. Object literals are often used to avoid declaring a large number of global variables and functions by creating a namespace. This helps to document that a set of properties belong together and serve a common purpose. When we create an object literal, its prototype is **Object.prototype**. [WMOTU Invaders object-oriented](#) provides a sample application.

Examples:

[students.btsi.lu/evegi144/WAD/JS/objects\\_literals.html](https://students.btsi.lu/evegi144/WAD/JS/objects_literals.html)

```
"use strict";

const myMerc = {
  brand: 'Mercedes',
  price: 100000,
  addVAT() {
    this.price = Math.ceil(this.price * 1.15);
  }
};

const myBMW = {
  brand: 'BMW',
  price: 60000,
  addVAT() {
    this.price = Math.ceil(this.price * 1.15);
  }
};

document.body.innerHTML += `Mercedes price without VAT:
${myMerc.price.toLocaleString()}<br>`;
myMerc.addVAT();
document.body.innerHTML += `Mercedes price with VAT:
${myMerc.price.toLocaleString()}<br>`;
document.body.innerHTML += `BMW price without VAT:
${myBMW.price.toLocaleString()}<br>`;

const empty = {}; // We create an empty object the REFERENCE to which is saved in this
// constant.
// We save the REFERENCE to object empty in constant myDog. This means that empty and
myDog
// point to the same object!
const myDog = empty;
myDog.name = 'Idexix'; // We give the myDog, which is the same as the empty, object a
name
myDog.bark = () => { // and a function.
  document.body.innerHTML += 'Wouf wouf!<br>';
};

document.body.innerHTML += `myDog's name is ${myDog.name}<br>`;
myDog.bark();
document.body.innerHTML += `Objects myDog and empty are identical: ${myDog ===`
```

```
empty}<br>`;
document.body.innerHTML += `empty's name is ${empty.name}<br>`;
empty.bark();

const doggy = Object.create(myDog);
doggy.bark();
myDog.name = 'Toto';
document.body.innerHTML += `empty's name is ${empty.name}<br>`;
document.body.innerHTML += `myDog's name is ${myDog.name}<br>`;
document.body.innerHTML += `doggy's name is ${doggy.name}<br>`;
console.log('Prototype of empty: ');
console.dir(Object.getPrototypeOf(empty));
console.log('Prototype of myDog: ');
console.dir(Object.getPrototypeOf(myDog));
console.log('Prototype of doggy: ');
console.dir(Object.getPrototypeOf(doggy));
```

#### from a constructor with **new**

**new** must be followed by a **constructor**, which is a function that initializes the newly created object. As noted above, **this** refers to the object itself, whose property we want to use. The advantage of this approach compared to the previous one is that we can create as many objects as we like using the same constructor. When we use a constructor, the prototype is the **prototype** property of the constructor. Example:

[students.btsi.lu/evegi144/WAD/JS/objects\\_new.html](https://students.btsi.lu/evegi144/WAD/JS/objects_new.html)

```

"use strict";

class Car {
  constructor(brand, price) {
    this.brand = brand;
    this.price = price;
  }

  addVAT() {
    this.price = Math.ceil(this.price * 1.15);
  }
}

const myMerc = new Car('Mercedes', 100000), myBMW = new Car('BMW', 60000);

document.body.innerHTML += `Mercedes price without VAT:
${myMerc.price.toLocaleString()}<br>`;
myMerc.addVAT();
document.body.innerHTML += `Mercedes price with VAT:
${myMerc.price.toLocaleString()}<br>`;
document.body.innerHTML += `BMW price without VAT:
${myBMW.price.toLocaleString()}<br>`;
myMerc.price = 80000; // This only changes the price of myMerc, but not myBMW.
document.body.innerHTML += `Mercedes price without VAT:
${myMerc.price.toLocaleString()}<br>`;
document.body.innerHTML += `BMW price without VAT:
${myBMW.price.toLocaleString()}<br>`;
document.body.innerHTML += `Prototype of myMerc:
${Object.getPrototypeOf(myMerc)}<br>`;
console.log('Prototype of myMerc: ');
console.dir(Object.getPrototypeOf(myMerc));
document.body.innerHTML += `Prototype of myBMW: ${Object.getPrototypeOf(myBMW)}<br>`;
console.log('Prototype of myBMW: ');
console.dir(Object.getPrototypeOf(myBMW));

```

**from a prototype with `Object.create()`**

Almost all JavaScript objects inherit the properties of another object, their prototype. `Object.create` is a method that allows us to create a new object with a given prototype.

[students.btsi.lu/evegi144/WAD/JS/objects\\_prototypes.html](https://students.btsi.lu/evegi144/WAD/JS/objects_prototypes.html)

```

"use strict";

const myMerc = Object.create({
  brand: 'Mercedes',
  price: 100000,
  addVAT() {
    this.price = Math.ceil(this.price * 1.15);
  }
});
const myBMW = Object.create({
  brand: 'BMW',
  price: 60000,
  addVAT() {
    this.price = Math.ceil(this.price * 1.15);
  }
});

const init = () => {
  document.body.innerHTML += `Mercedes price without VAT:
${myMerc.price.toLocaleString()}<br>`;
  myMerc.addVAT();
  document.body.innerHTML += `Mercedes price with VAT:
${myMerc.price.toLocaleString()}<br>`;
  document.body.innerHTML += `BMW price without VAT:
${myBMW.price.toLocaleString()}<br>`;
  myMerc.price = 80000; // This only changes the price of myMerc, but not myBMW.
  document.body.innerHTML += `Mercedes price without VAT:
${myMerc.price.toLocaleString()}<br>`;
  document.body.innerHTML += `BMW price without VAT:
${myBMW.price.toLocaleString()}<br>`;
  document.body.innerHTML += `The prototype of myBMW has the following
properties:<br>`;
  for (let o in Object.getPrototypeOf(myBMW)) document.body.innerHTML += `${o}<br>`;
  document.body.innerHTML += `Prototype of myBMW:
${Object.getPrototypeOf(myBMW)}<br>`;
  console.log('Prototype of myBMW: ');
  console.dir(Object.getPrototypeOf(myBMW));
}

addEventListener('load', init);

```

It is important to understand the difference between a constructor and a prototype. The prototype of an object contains all the properties that are common to all objects that have this prototype. They exist only once in the browser memory. Thus any changes to any of these properties are automatically reflected in all the objects that inherit from this prototype! Take a close look at the following examples:

[students.btsi.lu/evegi144/WAD/JS/objects1.html](https://students.btsi.lu/evegi144/WAD/JS/objects1.html)

```

"use strict";

const animal = {
  speciesName: '',
  sayHello() {
    document.write(`Hi, I am a ${this.speciesName}<br>`);
  }
};

// Object animal is the prototype of objects myDog and myCat.
const myDog = Object.create(animal), myCat = Object.create(animal);
animal.sayHello();
myDog.sayHello();
myCat.sayHello();
// If we change a property of the prototype, it will affect all objects with this
prototype.
animal.speciesName = 'dog';
animal.sayHello();
myDog.sayHello();
myCat.sayHello();

// Now let's use constructors.
// Each object constructed with this constructor will have its own name and bark
properties.
function Dog(name) {
  this.name = name;
  this.bark = () => {
    document.write(`${this.name} says wouf wouf!<br>`);
  };
}

function Cat(name) {
  this.name = name;
  this.miaow = function () {
    document.write(`${this.name} says mmmiiiiiaaaaooowwww!<br>`);
  };
}

class Ape {
  constructor(name) {
    this.name = name;
  }

  scream() {
    document.write(`${this.name} screams Ouaaaaaaaahhhhhhhh!<br>`);
  }

  crackNut() {
    document.write(`${this.name} cracked a nut.<br>`);
  }
}

```

```

class Gorilla extends Ape {
  constructor(name) {
    super(name);
  }

  scream() {
    document.write(`${this.name} screams I am BOSS!!!<br>`);
  }
}

const Toto = new Gorilla('Toto');
Toto.scream();
Toto.crackNut();
Ape.prototype.crackNut = function() {
  document.write(`${this.name} smashed a nut.<br>`);
};
Toto.crackNut();

// Create 2 new dogs using the dog constructor.
let myDog1 = new Dog('Bello'), myDog2 = new Dog('Toro');
// Create 2 new cats using the cat constructor.
let myCat1 = new Cat('Kitty'), myCat2 = new Cat('Tiger');
myDog1.bark();
myDog2.bark();
myCat1.miaow();
myCat2.miaow();
myDog2.name = 'Idefix'; // Change a property of a dog object.
myDog1.bark(); // The change only affects the other dog object.
myDog2.bark();

// Now let's combine constructor and prototype.
// All dogs and cats will have the same species_name and sayHello properties
Dog.prototype = animal;
Cat.prototype = animal;
// Create 2 new dogs using the dog constructor.
myDog1 = new Dog('Bello');
myDog2 = new Dog('Toro');
// Create 2 new cats using the cat constructor.
myCat1 = new Cat('Kitty');
myCat2 = new Cat('Tiger');

// Assign prototype.
myDog1.sayHello();
myDog2.sayHello();
myCat1.sayHello();
myCat2.sayHello();
animal.speciesName = 'bird';
myDog1.sayHello();
myDog2.sayHello();
myCat1.sayHello();

```

```
myCat2.sayHello();
myDog1.bark();
myDog2.bark();
myCat1.miaow();
myCat2.miaow();
console.log('Prototype of myDog1:');
console.dir(Object.getPrototypeOf(myDog1));
console.log('Prototype of myDog2:');
console.dir(Object.getPrototypeOf(myDog2));
console.log('Prototype of myCat1:');
console.dir(Object.getPrototypeOf(myCat1));
console.log('Prototype of myCat2:');
console.dir(Object.getPrototypeOf(myCat2));
```

## this

Mastery of the **this** keyword is a key requirement for understanding JS objects. You should therefore study the excellent explanation at [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/this](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Operators/this).

Do not confuse **this** with **self** (cf. [developer.mozilla.org/en-US/docs/Web/API/Window/self](https://developer.mozilla.org/en-US/docs/Web/API/Window/self) and [stackoverflow.com/questions/16875767/difference-between-this-and-self-in-javascript](https://stackoverflow.com/questions/16875767/difference-between-this-and-self-in-javascript)).

## Getting and setting properties

To get or set properties of an object we can either use the dot (.) or the square bracket ([]) operators. There is an important difference between these two: when using the dot operator, the right-hand must be an identifier, i.e. the name of the property. This cannot be a programmatically generated dynamic value. When using the square bracket operator, the value between the brackets must be an expression that evaluates to a string or to something that can be converted to a string. This opens up endless possibilities, so let's look at a few examples:

[students.btsi.lu/evegi144/WAD/JS/objects\\_access.html](https://students.btsi.lu/evegi144/WAD/JS/objects_access.html)

```

"use strict";

// The [] operator allows us to use names with spaces as property names:
const company = {
  'chief executive officer': 'Donald Duck'
};

//document.innerHTML += `${company.chief executive officer}<br>`; // Does not work!
document.innerHTML += `${company['chief executive officer']}<br>`; // Works!

class Student {
  constructor(name) {
    this.name = name;
    this.branches = {};
    // If the function is already defined, no need to define it again. This check is optional.
    if (!Student.prototype.addBranch)
      Student.prototype.addBranch = (name, average) => {
        this.branches[name] = average;
      };
    // If the function is already defined, no need to define it again. This check is optional.
    if (!Student.prototype.getTotalAverage)
      Student.prototype.getTotalAverage = () => {
        let avg = 0, count = 0;
        for (let branch in this.branches) {
          avg += this.branches[branch];
          count++;
        }
        if (count) return avg / count;
        else return undefined;
      }
  }
}

const st1 = new Student('Bill'), st2 = new Student('Bob');
document.body.innerHTML += `${st1.name} average: ${st1.getTotalAverage()}<br>`;
st1.addBranch("INFOR", 49);
document.body.innerHTML += `${st1.name} average: ${st1.getTotalAverage()}<br>`;
st1.addBranch("MATHE", 34);
document.body.innerHTML += `${st1.name} average: ${st1.getTotalAverage()}<br>`;
st2.addBranch("INFOR", 57);
document.body.innerHTML += `${st2.name} average: ${st2.getTotalAverage()}<br>`;
st2.addBranch("MATHE", 44);
document.body.innerHTML += `${st2.name} average: ${st2.getTotalAverage()}<br>`;

```

## Deleting properties

**delete** removes a property from an object. If you invoke **delete** on a prototype property, then all objects inheriting from this prototype lose this property.

[students.btsi.lu/evegi144/WAD/JS/objects\\_delete.html](https://students.btsi.lu/evegi144/WAD/JS/objects_delete.html)

```
"use strict";

class Dog {
  constructor(name) {
    this.name = name;
    this.bark = () => {
      document.write(`${this.name} says wouf wouf!<br>`);
    };
  }
}

const myDog1 = new Dog('Bello');
myDog1.bark();
delete myDog1.name;
myDog1.bark();
const myDog2 = new Dog('Idefix');
myDog2.bark();
```

## Testing properties

We have different options to test whether an object has a given property:

1. The **in** operator requires the name of a property as a string on its left and an object on its right side. It returns true if the object has an enumerable own or inherited property by that name (cf. [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...in](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/for...in)).
2. **Object.keys(obj)** returns an array of a given object's own enumerable properties (cf. [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Object/keys](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/keys)).
3. **Object.getOwnPropertyNames(obj)** returns an array of all properties (enumerable or not) found directly upon a given object (cf. [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Object/getOwnPropertyNames](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/getOwnPropertyNames)).
4. The **hasOwnProperty()** of an object checks whether the object has a property of the given name. It does not consider inherited properties for which it returns false (cf. [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Object/hasOwnProperty](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/hasOwnProperty)).
5. **propertyIsEnumerable()** returns a Boolean indicating whether the specified property is enumerable (cf. [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Object/propertyIsEnumerable](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/propertyIsEnumerable)).

The following example illustrates the different approaches:

[students.btsi.lu/evegi144/WAD/JS/objects\\_test.html](https://students.btsi.lu/evegi144/WAD/JS/objects_test.html)

```

"use strict";

class Dog {
  constructor(name) {
    this.name = name;
    this.bark = () => {
      document.write(`${this.name} says wouf wouf!<br>`);
    };
  }
}

const myDog1 = new Dog('Bello');
document.body.innerHTML += `toString is a property of myDog1: ${"toString" in
myDog1}<br>`;

let keys = Object.keys(myDog1); // Get the enumerable own properties of the object.
let output = '';
for (let i = 0; i < keys.length; i++) // Display the property/value pairs.
  output += `${keys[i]}: ${myDog1[keys[i]]}<br>`;
document.body.innerHTML += `<p>${output}</p>`;

const s = [1, 2, 3, 4, 5]; // Create an array object.
keys = Object.keys(s); // Get the enumerable own properties of the object.
output = '';
for (let i = 0; i < keys.length; i++) // Display the property/value pairs.
  if (s.propertyIsEnumerable(keys[i])) output += `Enumerable ${keys[i]}:
${s[keys[i]]}<br>`;
document.body.innerHTML += `<p>${output}</p>`;

const props = Object.getOwnPropertyNames(s); // Get all own properties of the object.
output = '';
for (let i = 0; i < props.length; i++) {
  if (s.propertyIsEnumerable(props[i])) output += 'Enumerable ';
  output += `${props[i]}: ${s[props[i]]}<br>`;
}
document.body.innerHTML += `<p>${output}</p>`;

```

## Object attributes

Every object has **prototype**, **class** and **extensible** attributes.

### prototype

This attribute specifies the object's parent, i.e. the object that it inherits from. As we've seen above, for object literals the prototype is **Object.prototype**, for objects created with **new** the prototype is the value of the constructor's prototype property and for objects created with **Object.create**, the prototype is the first parameter passed to this method.

Every object has an **isPrototypeOf** method, which checks whether the object is the prototype of another object passed as parameter.

## class

This internal read-only attribute tells us the type of the object. This attribute can take one of the following values: "Array", "Boolean", "Date", "Error", "Function", "JSON", "Math", "Number", "Object", "RegExp" or "String".

## extensible

This attribute determines whether new properties can be added to the object. `Object.isExtensible()` tells us whether an object is extensible or not. `Object.preventExtensions` allows us to make an object nonextensible.

Here is an example that illustrates the three object attributes:

[students.btsi.lu/evegi144/WAD/JS/objects\\_attributes\\_prototype.html](https://students.btsi.lu/evegi144/WAD/JS/objects_attributes_prototype.html)

```
"use strict";

const animal = {
  species: 'animal',
  displaySpecies () {
    document.body.innerHTML += `I'm a ${this.species}<br>`;
  }
};

const dog = Object.create(animal);
document.body.innerHTML += `Animal is the prototype of dog:
${animal.isPrototypeOf(dog)}<br>`;
document.body.innerHTML += `The class of animal is:
${Object.prototype.toString.call(animal).slice(8, -1)}<br>`;
console.dir(Object.getPrototypeOf(animal));
document.body.innerHTML += `The class of dog is:
${Object.getPrototypeOf(dog).toString().slice(8, -1)}<br>`;
console.dir(Object.getPrototypeOf(dog));
document.body.innerHTML += `animal is extensible: ${Object.isExtensible(animal)}<br>`;
Object.preventExtensions(animal);
document.body.innerHTML += `animal is extensible: ${Object.isExtensible(animal)}<br>`;
document.body.innerHTML += `dog is extensible: ${Object.isExtensible(dog)}<br>`;
```

## Property attributes

Each object property has, in addition to its value, the following three attributes:

1. **configurable**: true if and only if the type of this property may be changed and if the property may be deleted from the corresponding object.
2. **enumerable**
3. **writable**.

Using `Object.defineProperty(obj, prop, descriptor)` we can set these attributes (cf. [https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Object/defineProperty](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object/defineProperty)).

## Closures

From [developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Closures](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Closures):

A closure is a special kind of object that combines two things: a function, and the environment in which that function was created. The environment consists of any local variables that were in-scope at the time that the closure was created.

Study the MDN article, it explains the subject in detail with good examples. Then take a look at [bonsaiden.github.io/JavaScript-Garden](https://bonsaiden.github.io/JavaScript-Garden).

## Private instance and prototype members

[www.crockford.com/javascript/private.html](http://www.crockford.com/javascript/private.html)

Here is a more advanced example, illustrating how we can have both private prototype and private object members and inheritance:

[students.btsi.lu/evegi144/WAD/JS/objects2.html](http://students.btsi.lu/evegi144/WAD/JS/objects2.html)

```
'use strict';

const init = () => {
  class Base {
    constructor() {
      let instance_day = "Tuesday"; // Private instance variable.

      this.getInstanceDay = () => { // Public instance variable getter.
        return instance_day;
      };

      this.setInstanceDay = day => { // Public instance variable setter.
        instance_day = day;
      };

      // Private prototype variable.
      let dayName = "Friday";

      // Private prototype methods
      const getPrototypeDayName = () => {
        return dayName;
      };

      const setPrototypeDayName = day => {
        dayName = day;
      };

      if (!Base.prototype.getDayName)
        Base.prototype.getDayName = () => {
```

```

        return getPrototypeDayName();
    };

    if (!Base.prototype.setDayName)
        Base.prototype.setDayName = day => {
            setPrototypeDayName(day);
        };
    }
}

class SubType extends Base {
    constructor() {
        super();
    }
}

const a = new Base(), b = new Base(), c = new SubType(), d = new SubType();
// Returns "Tuesday"
document.body.innerHTML += `${a.getInstanceDay()}<br>`;

// Returns "Tuesday"
document.body.innerHTML += `${b.getInstanceDay()}<br>`;

// Sets dayName to "Wednesday"
a.setInstanceDay("Wednesday");

// Returns "Wednesday"
document.body.innerHTML += `${a.getInstanceDay()}<br>`;

// Returns "Tuesday"
document.body.innerHTML += `${b.getInstanceDay()}<br>`;

document.body.innerHTML += `${a.getDayName()}<br>`;
document.body.innerHTML += `${b.getDayName()}<br>`;
a.setDayName('Saturday');
document.body.innerHTML += `${a.getDayName()}<br>`;
document.body.innerHTML += `${b.getDayName()}<br>`;

// Returns "Tuesday"
document.body.innerHTML += `${c.getInstanceDay()}<br>`;

// Returns "Tuesday"
document.body.innerHTML += `${d.getInstanceDay()}<br>`;

// Sets dayName to "Wednesday"
c.setInstanceDay("Wednesday");

// Returns "Wednesday"
document.body.innerHTML += `${c.getInstanceDay()}<br>`;

// Returns "Tuesday"

```

```
document.body.innerHTML += `${d.getInstanceDay()}<br>`;

document.body.innerHTML += `${c.getDayName()}<br>`;
document.body.innerHTML += `${d.getDayName()}<br>`;
};

addEventListener('load', init);
```

### call, apply and bind

Given that functions are objects in JavaScript, these three methods allow us to call a function as if it were a method of another object. Detailed descriptions of the **Function** object and examples can be found at [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Function](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Function).

[students.btsi.lu/evegi144/WAD/JS/indirect1.html](https://students.btsi.lu/evegi144/WAD/JS/indirect1.html)

```
"use strict";

function printProperties() {
  document.body.innerHTML += 'Here come the properties...<br>';
  for (let p in this) document.body.innerHTML += `Property ${p}<br>`;
}

const o = {
  x: 1
};

printProperties();
printProperties.call(o);
```

### Object

The global **Object** object has a number of properties. The details can be found at [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Object](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Object). For instance, if we want to determine the number of an object's own enumerable properties, we can use **Object.keys(object)**.

### The Proxy object

[developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/Proxy](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Proxy)

### ECMAScript 6

ECMAScript 6 introduces more convenient syntax to work with objects and classes. Highly recommended introductions can be found at the following links:

[scotch.io/tutorials/better-javascript-with-es6-pt-ii-a-deep-dive-into-classes](https://scotch.io/tutorials/better-javascript-with-es6-pt-ii-a-deep-dive-into-classes)

[developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Classes](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Classes)

[exploringjs.com/es6/ch\\_classes.html](http://exploringjs.com/es6/ch_classes.html)

[www.sitepoint.com/object-oriented-javascript-deep-dive-es6-classes](http://www.sitepoint.com/object-oriented-javascript-deep-dive-es6-classes)

[shiroyasha.io/oop-in-the-age-of-es6.html](http://shiroyasha.io/oop-in-the-age-of-es6.html)

[github.com/getify/You-Dont-Know-JS/blob/master/this%20%26%20object%20prototypes/ch6.md](https://github.com/getify/You-Dont-Know-JS/blob/master/this%20%26%20object%20prototypes/ch6.md)

[www.ibm.com/developerworks/library/wa-ecmascript6-neward-p3](http://www.ibm.com/developerworks/library/wa-ecmascript6-neward-p3)

[students.btsi.lu/evegi144/WAD/WAD.html#\\_call\\_apply\\_and\\_bind](http://students.btsi.lu/evegi144/WAD/WAD.html#_call_apply_and_bind)

### 4.3.19. Events

So far we have seen how to write scripts that execute sequentially, i.e. instructions are processed one after the other. In real life, however, our browser spends most of its time waiting for something to happen, e.g. a key or mouse button is pressed, a picture is loaded, a given timespan has expired, etc. This "something" is called an event in JavaScript. See the references below for the full list of available events.

For our scripts to be able to react to an event, we need to declare functions, called event listeners or handlers, that are invoked by the browser when a specific event occurs. Our handler will be automatically passed an **Event** object (cf. references).

[www.w3schools.com/jsref/dom\\_obj\\_event.asp](http://www.w3schools.com/jsref/dom_obj_event.asp)

[developer.mozilla.org/en-US/docs/Web/Events](https://developer.mozilla.org/en-US/docs/Web/Events)

[developer.mozilla.org/en-US/docs/Web/API/Event](https://developer.mozilla.org/en-US/docs/Web/API/Event)

[www.javascriptkit.com/script/script2/xeye.shtml](http://www.javascriptkit.com/script/script2/xeye.shtml)

### Registering event listeners

In order to tell the browser what it should do if an event occurs, we need to register an event listener. There are 3 ways to do this.

**addEventListener**

The recommended way is to use the **addEventListener** method (cf. <https://developer.mozilla.org/en-US/docs/Web/API/EventTarget.addEventListener>) of the object that will be the target of the event. The first parameter is a string containing the name of the event, the second one will be either an anonymous function or the name (without apostrophes) of the function that will handle the event. The optional third parameter allows to have the listener triggered already in the capture phase as described below. We can register several listeners for the same event using this method:

[students.btsi.lu/evegi144/WAD/JS/event\\_listeners1.html](http://students.btsi.lu/evegi144/WAD/JS/event_listeners1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Handling events using addEventListener</title>
    <meta charset=utf-8>
    <script src=event_listeners1.js></script>
  </head>
  <body>
    <main>
      <button>Click me!</button>
    </main>
  </body>
</html>
```

```

"use strict";

const init = () => {
  const clickHandler1 = () => {
    alert("I'm the first click event handler");
  };

  const clickHandler2 = (event, text) => { // This is the function that will handle
the click
    // event.
    console.dir(event);
    alert(`I'm the second click event handler. You asked me to show this text:
${text}`);
  };

  const myButton = document.querySelector('button'); // Get the button element from
the DOM.
  /* Register the first button click handler.
  Note that clickHandler1 is just a pointer to the function.
  This works because clickHandler1 takes no parameter except the event, which it
receives
  by default. */
  myButton.addEventListener('click', clickHandler1);
  /* Register the button's second click event handler using an anonymous function.
  The function receives the event as a parameter by default.
  Our event handler takes a text to display as second parameter.
  To pass this to our event handler, we need to use an anonymous function.
  */

  myButton.addEventListener('click', event => {
    clickHandler2(event, 'some random text');
  });
};

window.addEventListener('load', init); // Register the event handler for the load
event.

```

### Object property

Another option for event listener registration is the use of the corresponding object property. The name of this property is "on" followed by the event name, e.g. "onclick", "onload" etc. If a handler for this event has already been registered using this approach, it will be replaced:

[students.btsi.lu/evegi144/WAD/JS/event\\_listeners2.html](https://students.btsi.lu/evegi144/WAD/JS/event_listeners2.html)

```

"use strict";

const init = () => {
  const handleClick = event => { // This is the function that will handle the click
    event.
    alert('You clicked me!');
  };

  const myButton = document.querySelector('button'); // Get the button element from
  the DOM.
  myButton.onclick = handleClick; // Register the click event handler as an object
  property.
};

onload = init; // Register the load event handler as an object property.

```

### HTML attribute

The third and least preferable registration method is the use of an HTML attribute. This way should be avoided as it violates the separation of content and behavior:

[students.btsi.lu/evegi144/WAD/JS/event\\_listeners3.html](https://students.btsi.lu/evegi144/WAD/JS/event_listeners3.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Handling events using HTML attributes</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <button onclick="alert('You clicked me!');">Click me!</button>
    </main>
  </body>
</html>

```

### Invocation order

Handlers registered via an object property or HTML attribute are always executed first. Handlers registered with `addEventListener` are called in the order in which they were registered.

### Event flow

The chart at [www.w3.org/TR/DOM-Level-3-Events/#dom-event-architecture](https://www.w3.org/TR/DOM-Level-3-Events/#dom-event-architecture) illustrates the three event phases: capture, target and bubble. If the target of an event is a standalone object (e.g. window), only the event handler on that object will be triggered. If the target object is a tree, for instance the document element, most, but not all, events bubble up the tree as illustrated in the chart. This avoids the need to register event handlers on lots of different elements. We can register

a single event handler at the top of the tree, which can deal with the events triggered on its children. If we set the third parameter of `addEventListener` to true, events are captured before reaching the target object. As seen in the chart, this is the inverse process of bubbling. It provides an opportunity for the parent objects to analyze the event before it reaches the target. This is useful in certain situations, for instance for the implementation of drag and drop, as we'll see in [Drag and drop](#):

[students.btsi.lu/evegi144/WAD/JS/events1.html](https://students.btsi.lu/evegi144/WAD/JS/events1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Event propagation with capturing demo</title>
    <meta charset=utf-8>
    <style>
      body {
        background-color: black;
      }

      main {
        background-color: gold;
      }

      p {
        background-color: green;
      }
    </style>
    <script>
      'use strict';

      /* Note that we use event CAPTURING in all cases here to illustrate the concept.
         We would normally leave out the third parameter of addEventListener.
      */
      const init = () => {
        const mouseHandler = event => {
          console.dir(this);
          console.log(`mouseHandler current event target: ${event.currentTarget}`);
        };

        window.addEventListener('click', mouseHandler, true);
        document.body.addEventListener('click', mouseHandler, true);
        document.getElementsByTagName('main')[0].addEventListener('click',
mouseHandler, true);
        document.getElementsByTagName('p')[0].addEventListener('click', mouseHandler,
true);
      };

      window.addEventListener('load', init, true);
    </script>
  </head>
  <body>
    <main>
      <p>This is a paragraph</p>
    </main>
  </body>
</html>

```

The following example is identical to the previous one, except that the third parameter of `addEventListener` is not set: [students.btsi.lu/evegi144/WAD/JS/events2.html](https://students.btsi.lu/evegi144/WAD/JS/events2.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Event propagation without capturing demo</title>
    <meta charset=utf-8>
    <style>
      body {
        background-color: black;
      }

      main {
        background-color: gold;
      }

      p {
        background-color: green;
      }
    </style>
    <script>
      'use strict';

      // Note that we don't use event CAPTURING.
      const init = () => {
        const mouseHandler = event => {
          console.dir(this);
          console.log(`mouseHandler current event target: ${event.currentTarget}`);
        };

        window.addEventListener('click', mouseHandler);
        document.body.addEventListener('click', mouseHandler);
        document.getElementsByTagName('main')[0].addEventListener('click',
mouseHandler);
        document.getElementsByTagName('p')[0].addEventListener('click', mouseHandler);
      };

      window.addEventListener('load', init);
    </script>
  </head>
  <body>
    <main>
      <p>This is a paragraph</p>
    </main>
  </body>
</html>

```

### Preventing event propagation and/or default actions

In some cases we might want to prevent the default action for a given event to occur, for instance, in order to prevent a form from being submitted when the user clicks the submit button. For this purpose we can call the `preventDefault` method. We can also stop event propagation using

**stopPropagation**. In most cases our event handlers will not return a value. However, if we want to prevent default action and stop propagation, we can return false in our event handler. This works only if the event handler has been registered as an object property or an HTML attribute. With **addEventListener** we can call the **preventDefault** and/or **stopPropagation** methods:

[students.btsi.lu/evegi144/WAD/JS/event\\_cancellation1.html](https://students.btsi.lu/evegi144/WAD/JS/event_cancellation1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Preventing event propagation and default action</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const init = () => {
        document.forms[2].addEventListener('submit', e => {
          e.preventDefault();
          //e.stopPropagation();
          console.log('Form 3 not submitted');
        });
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main>
      <form action=f1.php onsubmit="console.log('Form 1 submitted');">
        <input name=i1>
        <input type=submit name=s1 value=Send>
      </form>
      <form action=f2.php onsubmit="console.log('Form 2 not submitted'); return
false;">
        <input name=i2>
        <input type=submit name=s2 value=Send>
      </form>
      <form action=f3.php>
        <input name=i3>
        <input type=submit name=s3 value=Send>
      </form>
    </main>
  </body>
</html>
```

## Keyboard events

It is important to understand the difference between the three keyboard events **keydown**, **keypress** and **keyup**: When you press a key down, a **keydown** event is triggered, followed by a **keypress** event. If

you keep the key pressed, **keypress** events will continue to be generated at regular time intervals until you release the key, in which case a **keyup** event will be triggered. The following program displays all the attributes of the key events:

[students.btsi.lu/evegi144/WAD/JS/keyevents1.html](https://students.btsi.lu/evegi144/WAD/JS/keyevents1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Key event handler example 1</title>
    <meta charset=utf-8>
    <style>
      table {
        border-collapse: collapse;
        width:          100%;
        padding:         5px;
      }

      table, th, td {
        border:          1px solid red;
        text-align: center;
      }

      table caption {
        font-weight: bold;
        color:         blue;
      }
    </style>
    <script>
      'use strict';

      const init = () => {
        const key = event => {
          let table, i;
          if (event.type === 'keydown') i = 0;
          else if (event.type === 'keypress') i = 1;
          else i = 2;
          table = document.getElementsByTagName('table')[i];
          table.querySelector('caption').innerHTML = event.type +
            ' ' + new Date().toString();
          table.querySelector(`#altKey${i}`).innerHTML = event.altKey;
          table.querySelector(`#char${i}`).innerHTML = event.char;
          table.querySelector(`#charCode${i}`).innerHTML = event.charCode;
          table.querySelector(`#code${i}`).innerHTML = event.code;
          table.querySelector(`#ctrlKey${i}`).innerHTML = event.ctrlKey;
          table.querySelector(`#isComposing${i}`).innerHTML = event.isComposing;
          table.querySelector(`#key${i}`).innerHTML = event.key;
          table.querySelector(`#keyCode${i}`).innerHTML = event.keyCode;
          table.querySelector(`#locale${i}`).innerHTML = event.locale;
          table.querySelector(`#location${i}`).innerHTML = event.location;
```

```

        table.querySelector(`#metaKey${i}`).innerHTML = event.metaKey;
        table.querySelector(`#repeat${i}`).innerHTML = event.repeat;
        table.querySelector(`#shiftKey${i}`).innerHTML = event.shiftKey;
        table.querySelector(`#which${i}`).innerHTML = event.which;
        console.dir(event);
    };

    window.addEventListener('keydown', key);
    window.addEventListener('keypress', key);
    window.addEventListener('keyup', key);
};

window.onload = init;
</script>
</head>
<body>
    <main>
        <table>
            <caption></caption>
            <thead>
                <tr>
                    <th>altKey</th>
                    <th>char</th>
                    <th>charCode</th>
                    <th>code</th>
                    <th>ctrlKey</th>
                    <th>isComposing</th>
                    <th>key</th>
                    <th>keyCode</th>
                    <th>locale</th>
                    <th>location</th>
                    <th>metaKey</th>
                    <th>repeat</th>
                    <th>shiftKey</th>
                    <th>which</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td id=altKey0></td>
                    <td id=char0></td>
                    <td id=charCode0></td>
                    <td id=code0></td>
                    <td id=ctrlKey0></td>
                    <td id=isComposing0></td>
                    <td id=key0></td>
                    <td id=keyCode0></td>
                    <td id=locale0></td>
                    <td id=location0></td>
                    <td id=metaKey0></td>
                    <td id=repeat0></td>

```

```

        <td id=shiftKey0></td>
        <td id=which0></td>
    </tr>
</tbody>
</table>
<table>
    <caption></caption>
    <thead>
        <tr>
            <th>altKey</th>
            <th>char</th>
            <th>charCode</th>
            <th>code</th>
            <th>ctrlKey</th>
            <th>isComposing</th>
            <th>key</th>
            <th>keyCode</th>
            <th>locale</th>
            <th>location</th>
            <th>metaKey</th>
            <th>repeat</th>
            <th>shiftKey</th>
            <th>which</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <td id=altKey1></td>
            <td id=char1></td>
            <td id=charCode1></td>
            <td id=code1></td>
            <td id=ctrlKey1></td>
            <td id=isComposing1></td>
            <td id=key1></td>
            <td id=keyCode1></td>
            <td id=locale1></td>
            <td id=location1></td>
            <td id=metaKey1></td>
            <td id=repeat1></td>
            <td id=shiftKey1></td>
            <td id=which1></td>
        </tr>
    </tbody>
</table>
<table>
    <caption></caption>
    <thead>
        <tr>
            <th>altKey</th>
            <th>char</th>
            <th>charCode</th>

```

```

        <th>code</th>
        <th>ctrlKey</th>
        <th>isComposing</th>
        <th>key</th>
        <th>keyCode</th>
        <th>locale</th>
        <th>location</th>
        <th>metaKey</th>
        <th>repeat</th>
        <th>shiftKey</th>
        <th>which</th>
    </tr>
</thead>
<tbody>
    <tr>
        <td id=altKey2></td>
        <td id=char2></td>
        <td id=charCode2></td>
        <td id=code2></td>
        <td id=ctrlKey2></td>
        <td id=isComposing2></td>
        <td id=key2></td>
        <td id=keyCode2></td>
        <td id=locale2></td>
        <td id=location2></td>
        <td id=metaKey2></td>
        <td id=repeat2></td>
        <td id=shiftKey2></td>
        <td id=which2></td>
    </tr>
</tbody>
</table>
</main>
</body>
</html>

```

Here is the same program, but instead of hard coded HTML tables, they are generated using JavaScript. This shows how we can modify the DOM (cf. [4](#)). The result is 75 lines instead of 176:

[students.btsi.lu/evegi144/WAD/JS/keyevents1\\_optimized.html](http://students.btsi.lu/evegi144/WAD/JS/keyevents1_optimized.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Key event handler example 1 optimized</title>
    <meta charset=utf-8>
    <style>
      table {
        border-collapse: collapse;
        width:          100%;

```

```

padding:      5px;
}

table, th, td {
  border:      1px solid red;
  text-align: center;
}

table caption {
  font-weight: bold;
  color:       blue;
}
</style>
<script>
  'use strict';

  const init = () => {
    const keyAttrs = ['altKey', 'char', 'charCode', 'code', 'ctrlKey',
'isComposing',
    'key', 'keyCode', 'locale', 'location', 'metaKey', 'repeat', 'shiftKey',
    'which'];

    const key = event => {
      let table, i;
      if (event.type === 'keydown') i = 0;
      else if (event.type === 'keypress') i = 1;
      else i = 2;
      table = document.getElementsByTagName('table')[i];
      table.querySelector('caption').innerHTML =
        `${event.type} ${new Date().toLocaleTimeString()}`;
      for (let j = 0; j < keyAttrs.length; j++)
        table.querySelector(`#${keyAttrs[j]}${i}`).innerHTML = event[keyAttrs[j]];
      console.dir(event);
    };

    let table, caption, thead, tbody, tr, th, td;
    for (let i = 0; i < 3; i++) {
      table = document.createElement('table');
      caption = document.createElement('caption');
      thead = document.createElement('thead');
      tr = document.createElement('tr');
      tbody = document.createElement('tbody');
      table.appendChild(caption);
      table.appendChild(thead);
      thead.appendChild(tr);
      for (let j = 0; j < 14; j++) {
        th = document.createElement('th');
        th.innerHTML = keyAttrs[j];
        tr.appendChild(th);
      }
      table.appendChild(tbody);
    }
  };

```

```

        tr = document.createElement('tr');
        tbody.appendChild(tr);
        for (let j = 0; j < 14; j++) {
            td = document.createElement('td');
            td.id = keyAttrs[j] + i;
            tr.appendChild(td);
        }
        document.querySelector('main').appendChild(table);
    }
    window.addEventListener('keydown', key);
    window.addEventListener('keypress', key);
    window.addEventListener('keyup', key);
};

    window.onload = init;
</script>
</head>
<body>
    <main></main>
</body>
</html>

```

To determine the code for a specific key, take a look at the following links:

[www.cambiaresearch.com/articles/15/javascript-char-codes-key-codes](http://www.cambiaresearch.com/articles/15/javascript-char-codes-key-codes)

[developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/keyCode](https://developer.mozilla.org/en-US/docs/Web/API/KeyboardEvent/keyCode)

[msdn.microsoft.com/en-us/library/windows/desktop/dd375731%28v=vs.85%29.aspx](https://msdn.microsoft.com/en-us/library/windows/desktop/dd375731%28v=vs.85%29.aspx)

### unload and beforeunload

If we want to have some code executed when a user navigates away from our page, we can use the **beforeunload** (cf. [developer.mozilla.org/en-US/docs/Web/API/WindowEventHandlers/onbeforeunload](https://developer.mozilla.org/en-US/docs/Web/API/WindowEventHandlers/onbeforeunload)) or the **unload** (cf. <https://developer.mozilla.org/en-US/docs/Web/API/WindowEventHandlers/onunload>) events.

Here is a simple example:

[students.btsi.lu/evegi144/WAD/JS/beforeunload.html](http://students.btsi.lu/evegi144/WAD/JS/beforeunload.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Unload test</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <script>
      'use strict';

      const beforeunload = e => {
        console.dir(e);
        const req = new XMLHttpRequest();
        req.open('POST', 'beforeunload.php', false);
        req.send();
        const confirmationMessage = 'Are you sure you want to leave this shiny page?';
        (e || window.event).returnValue = confirmationMessage; //Gecko + IE
        return confirmationMessage; //Webkit, Safari, Chrome etc.
      };

      addEventListener('load', () => addEventListener('beforeunload', beforeunload));
    </script>
  </body>
</html>

```

```

<?php
  error_log('Before unload');
?>

```

## error

When we include images in our HTML document, which cannot be found by the browser, the browser will display an ugly icon. In Firefox we can solve this issue by setting the alt attribute to the empty string. To solve the issue for all browsers, we can handle the **error** event, like so (test it in Chrome and Firefox to see the difference):

[students.btsi.lu/evegi144/WAD/JS/onerror.html](https://students.btsi.lu/evegi144/WAD/JS/onerror.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Handling the error event</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <img src=dontexist.png alt=''>
      <!-- http://stackoverflow.com/questions/8987428/image-placeholder -->
      <img src=dontexist.png alt='' onerror="this.style.display='none';">
    </main>
  </body>
</html>
```

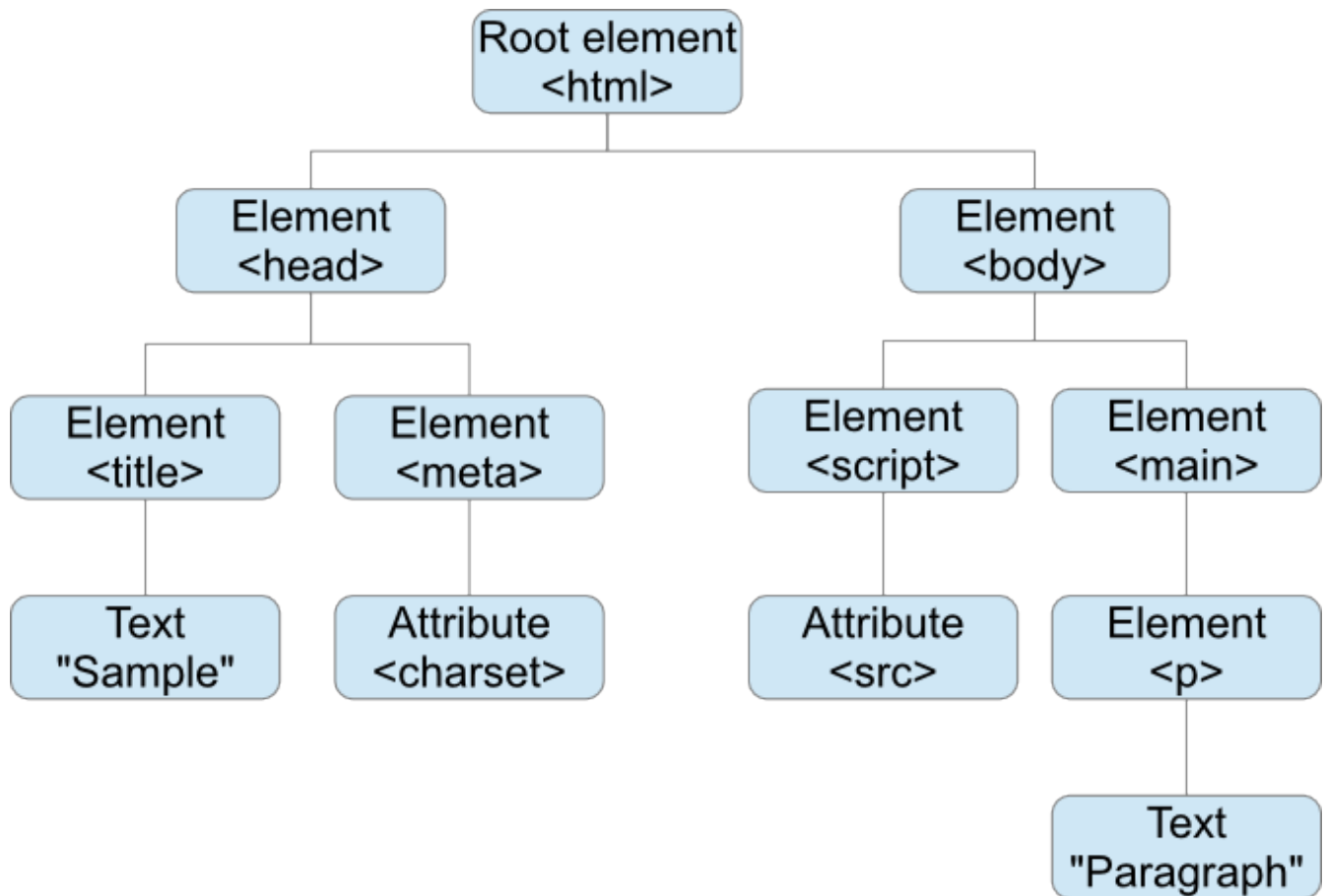
### 4.3.20. )

From [developer.mozilla.org/en-US/docs/Glossary/DOM](https://developer.mozilla.org/en-US/docs/Glossary/DOM):

The Document Object Model (DOM) is an API defined by the W3C to represent and interact with any HTML or XML document. The DOM is a model of an HTML or XML document that is loaded in a web browser. It represents a document as a tree of nodes, where each node represents a portion of the document, such as an element, a portion of text or a comment. The DOM is one of the most used APIs on the web because it allows code running in a web browser to access and interact with every node in the document. Nodes can be created, moved and changed. Event listeners can be added to nodes. Once a given event occurs all of its event listeners are triggered.

We have already looked at the DOM in several instances using the browser console or Firebug.

Here is what the DOM of a very basic HTML document looks like:



A basic introduction can be found at [www.w3schools.com/js/js\\_htmlDOM.asp](http://www.w3schools.com/js/js_htmlDOM.asp).

**HTMLElement** represents any HTML element. Specific elements are children of this object. For instance, a div element is represented via an **HTMLDivElement** object (cf. [developer.mozilla.org/en-US/docs/Web/API/Document\\_Object\\_Model](http://developer.mozilla.org/en-US/docs/Web/API/Document_Object_Model)).

**HTMLDocument** defines some specific properties that are often quite handy. **document.body** is the **<body>**, **document.head** the **<head>** and **document.documentElement** the root, i.e. **<html>** element of the document.

### Creating a new DOM document from a string

From [developer.mozilla.org/en-US/docs/Web/API/DOMParser](http://developer.mozilla.org/en-US/docs/Web/API/DOMParser):

DOMParser can parse XML or HTML source stored in a string into a DOM Document.

### Properties and methods of the HTML DOM Element Object

At [www.w3schools.com/jsref/dom\\_obj\\_all.asp](http://www.w3schools.com/jsref/dom_obj_all.asp) you can find a list of the properties and methods of the HTML DOM Element object with examples, which is very helpful.

### Selecting DOM elements

Before we can change a DOM element in JavaScript, we need to select it, i.e. we need to get a pointer to the element. For this purpose, the **document** object allows us to select an element by id, name, tag, CSS class or selector.

## document.getElementById

This method takes a string parameter specifying the ID of the element that we want.

[students.btsi.lu/evegi144/WAD/JS/getElementById.html](https://students.btsi.lu/evegi144/WAD/JS/getElementById.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>getElementById examples</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      const doSomeWork = () => {
        const myMain = document.getElementById('myMain'); // Store the main element.
        console.dir(myMain); // Display the main element in the console.
        console.dir(document.getElementById('myHeader')); // Display the header.
      };

      // The script should only be executed after the whole document has been loaded,
      otherwise
      // the HTML elements are not yet available.
      window.addEventListener('load', doSomeWork);
    </script>
  </head>
  <body>
    <main id=myMain>
      <header id=myHeader>Header</header>
    </main>
  </body>
</html>
```

## document.getElementsByName

This method takes a string parameter specifying the name of the element that we want and returns a **NodeList** object, which we can access using indices, like arrays (cf. [developer.mozilla.org/en-US/docs/Web/API/NodeList](https://developer.mozilla.org/en-US/docs/Web/API/NodeList)). Remember that the **name** attribute is used during form submission to send data to the server. Not every HTML element can have a **name** attribute.

[students.btsi.lu/evegi144/WAD/JS/getElementsByName.html](https://students.btsi.lu/evegi144/WAD/JS/getElementsByName.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>getElementsByName examples</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      const doSomeWork = () => {
        // Store ALL elements with name myForm in an array.
        const myForms = document.getElementsByName('myForm');
        console.log(`There are ${myForms.length} form elements.<br>`);
        // Display the first (and only) form element in the console.
        console.dir(myForms[0]);
        const myInputs = document.getElementsByName('gender');
        console.log(`There are ${myInputs.length} radio input elements.<br>`);
        // Display the first (and only) header element in the console.
        for (let i = 0; i < myInputs.length; i++) console.dir(myInputs[i]);
      };

      window.addEventListener('load', doSomeWork);
    </script>
  </head>
  <body>
    <main> <!--Note that the main element cannot have a name attribute.-->
      <form name=myForm>
        <input type=radio name=gender>male<br>
        <input type=radio name=gender>female
      </form>
    </main>
  </body>
</html>

```

`document.getElementsByTagName`

This method works like the previous one, except that the parameter is the HTML tag for which we want to select all elements.

[students.btsi.lu/evegi144/WAD/JS/getElementsByTagName.html](https://students.btsi.lu/evegi144/WAD/JS/getElementsByTagName.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>getElementsByTagName examples</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      const doSomeWork = () => {
        // Store ALL main elements in an array.
        const myMains = document.getElementsByTagName('main');
        console.log(`There are ${myMains.length} main elements.<br>`);
        // Store ALL form elements in an array.
        const myForms = document.getElementsByTagName('form');
        console.log(`There are ${myForms.length} form elements.<br>`);
        // Display the first (and only) form element in the console.
        console.dir(myForms[0]);
        const myInputs = document.getElementsByTagName('input');
        console.log(`There are ${myInputs.length} input elements.<br>`);
        // Display the first (and only) header element in the console.
        for (let i = 0; i < myInputs.length; i++) console.dir(myInputs[i]);
      };

      window.addEventListener('load', doSomeWork);
    </script>
  </head>
  <body>
    <main>
      <form>
        <input type=radio name=gender>male<br>
        <input type=radio name=gender>female
      </form>
    </main>
  </body>
</html>

```

### **document.getElementsByClassName**

This method works like the previous two, except that the parameter is the CSS class for which we want to select all elements.

[students.btsi.lu/evegi144/WAD/JS/getElementsByClassName.html](https://students.btsi.lu/evegi144/WAD/JS/getElementsByClassName.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>getElementsByClassName examples</title>
    <meta charset=utf-8>
    <style>
      .magic {
        background-color: lightgreen;
      }
    </style>
    <script>
      "use strict";

      const doSomeWork = () => {
        // Store ALL elements of the magic class in an array.
        const myMagics = document.getElementsByClassName('magic');
        console.log(`There are ${myMagics.length} magic elements.<br>`);
        for (let i = 0; i < myMagics.length; i++) console.dir(myMagics[i]);
      };

      window.addEventListener('load', doSomeWork);
    </script>
  </head>
  <body>
    <main>
      <form>
        <input type=radio><span class=magic>male</span><br>
        <input type=radio>female
      </form>
    </main>
  </body>
</html>

```

and

These methods work like the previous three, except that the parameter is the CSS selector for which we want to select all elements (cf. [www.w3.org/TR/selectors-api](http://www.w3.org/TR/selectors-api)). `querySelector` returns the first element that matches the given CSS selector, whereas `querySelectorAll` returns a `NodeList` object with all the matching elements. These are the most powerful selectors available. We can use the whole gamut of CSS selectors described in [www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp).

[students.btsi.lu/evegi144/WAD/JS/querySelector.html](http://students.btsi.lu/evegi144/WAD/JS/querySelector.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>querySelectorAll examples</title>
    <meta charset=utf-8>
    <style>

```

```

    .magic {
        background-color: lightgreen;
    }
</style>
<script>
    "use strict";

    const doSomeWork = () => {
        // Store the FIRST element of the magic class.
        const myFirstMagic = document.querySelectorAll('.magic');
        console.dir(myFirstMagic);
        // Store all magic articles in an array.
        const myMagicArticles = document.querySelectorAll('.magic article');
        console.log(`There are ${myMagicArticles.length} magic articles.<br>`);
        for (let i = 0; i < myMagicArticles.length; i++)
            console.dir(myMagicArticles[i]);
    };

    window.addEventListener('load', doSomeWork);
</script>
</head>
<body>
    <header><h1>Header</h1></header>
    <main>
        <section>
            <h2>Section 1 header</h2>
            <article>
                <h3>Section 1 article 1 header</h3>
                Section 1 article 1
            </article>
            <article>
                <h3>Section 1 article 2 header</h3>
                Section 1 article 2
            </article>
        </section>
        <section class=magic>
            <h2>Section 2 header</h2>
            <article>
                <h3>Section 2 article 1 header</h3>
                Section 2 article 1
            </article>
            <article>
                <h3>Section 2 article 2 header</h3>
                Section 2 article 2
            </article>
        </section>
    </main>
</body>
</html>

```

## Direct access via the `document` object

We can directly access the following HTML objects (and object collections):

- `document.anchors`
- `document.body`
- `document.documentElement`
- `document.embeds`
- `document.forms`
- `document.head`
- `document.images`
- `document.links`
- `document.scripts`
- `document.title`

## Traversing the DOM

Depending on what we want to do, we can traverse the DOM as a node or as an element tree.

### Node trees

From [www.w3schools.com/jsref/dom\\_obj\\_all.asp](http://www.w3schools.com/jsref/dom_obj_all.asp):

In the HTML DOM, everything is a node:

- The document itself is a document node.
- All HTML elements are element nodes.
- All HTML attributes are attribute nodes.
- Texts inside HTML elements are text nodes.
- Comments are comment nodes.

The link above provides extensive information on the `NodeList` object's methods and properties. In particular, we can determine a node's type via the `nodeType` property (cf. [www.w3schools.com/jsref/prop\\_node\\_nodetype.asp](http://www.w3schools.com/jsref/prop_node_nodetype.asp)).

[students.btsi.lu/evegi144/WAD/JS/DOM\\_nodes.html](http://students.btsi.lu/evegi144/WAD/JS/DOM_nodes.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Illustrates DOM traversal using nodes</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      // We traverse the DOM tree recursively, using all Node elements.
      const traverseDOM = e => {
        for (let child = e.firstChild; child; child = child.nextSibling) {
          console.log(`Node name: ${child.nodeName} value: ${child.nodeValue}`);
          if (child.nodeType === 1) traverseDOM(child);
        }
      };

      window.addEventListener('load', () =>
traverseDOM(document.querySelector('main')));
    </script>
  </head>
  <body>
    <main>
      <header><h1>This is the main header</h1></header>
      <section>
        <header><h2>Section 1 header</h2></header>
        <article>
          <header><h3>Section 1 article 1 header</h3></header>
        </article>
        <article>
          <header><h3>Section 1 article 2 header</h3></header>
        </article>
      </section>
      <section>
        <header><h2>Section 2 header</h2></header>
        <article>
          <header><h3>Section 2 article 1 header</h3></header>
        </article>
        <article>
          <header><h3>Section 2 article 2 header</h3></header>
        </article>
      </section>
    </main>
  </body>
</html>

```

## Element trees

From [developer.mozilla.org/en-US/docs/Web/API/element](https://developer.mozilla.org/en-US/docs/Web/API/element):

The Element interface represents an object within a DOM document.

Text and comment nodes are not treated as objects in this context and are ignored. This API allows us therefore to traverse the DOM element tree, without bothering with text and comments.

The `children` property of an element is particularly useful, as it contains an array of all of its HTML element children.

[students.btsi.lu/evegi144/WAD/JS/DOM\\_elements.html](https://students.btsi.lu/evegi144/WAD/JS/DOM_elements.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Illustrates DOM traversal using elements</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      // We traverse the DOM tree recursively, using only HTML elements.
      const traverseDOM = e => {
        for (let i = 0; i < e.children.length; i++) {
          console.log(e.children[i].nodeName);
          traverseDOM(e.children[i]);
        }
      };

      window.addEventListener('load', () =>
traverseDOM(document.querySelector('main')));
    </script>
  </head>
  <body>
    <main>
      <header><h1>This is the main header</h1></header>
      <section>
        <header><h2>Section 1 header</h2></header>
        <article>
          <header><h3>Section 1 article 1 header</h3></header>
        </article>
        <article>
          <header><h3>Section 1 article 2 header</h3></header>
        </article>
      </section>
      <section>
        <header><h2>Section 2 header</h2></header>
        <article>
          <header><h3>Section 2 article 1 header</h3></header>
        </article>
        <article>
          <header><h3>Section 2 article 2 header</h3></header>
        </article>
      </section>
    </main>
  </body>
</html>

```

## Getting and setting attributes

## as element properties

The attributes of HTML elements are available as properties of the corresponding `HTMLElement` in JavaScript. However, whereas HTML attributes are not case sensitive, JavaScript properties use camel case (cf. [en.wikipedia.org/wiki/CamelCase](https://en.wikipedia.org/wiki/CamelCase)). For instance the `usemap` attribute of an `img` element can be accessed via the `useMap` property. There are two exceptions: given that some attribute names are reserved words in JavaScript, the property name has an `html` prefix, for instance the `for` attribute can be accessed via the `htmlFor` property. The exception to the exception is the `class` attribute, which can be accessed via the `className` property.

[students.btsi.lu/evegi144/WAD/JS/DOM\\_attributes1.html](https://students.btsi.lu/evegi144/WAD/JS/DOM_attributes1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Illustrates access of HTML element attributes using
      element properties</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      const toggleImage = () => {
        let image = document.querySelector('img'); // Get the image element.
        console.log(image.src); // Display the complete URL.
        // Extract the image filename from the complete URL.
        let imageName = image.src.substring(image.src.lastIndexOf('/') + 1);
        // Given that the image is in the same directory as this document,
        // we do not need to specify the complete URL, just the image filename.
        if (imageName === 'camaro256x256.png') {
          image.src = 'ferrari256x256.png';
          image.title = 'https://www.iconfinder.com/icons/67532/'
            + 'car_ferrari_red_small_car_sports_car_icon#size=256';
        }
        else {
          image.src = 'camaro256x256.png';
          image.title = 'https://www.iconfinder.com/icons/67528/'
            + 'camaro_car_sports_car_icon#size=256';
        }
      };
    </script>
  </head>
  <body>
    <button onclick='toggleImage();'>Toggle image</button>
    <br>
    <img src=camaro256x256.png alt=Car

title=https://www.iconfinder.com/icons/67528/camaro_car_sports_car_icon#size=256>
  </body>
</html>
```

Here is an example of how to scroll to the end of an element's content using JS. Note that the height of the element needs to be set to something smaller than the height taken by the content, otherwise this won't work.

[students.btsi.lu/evegi144/WAD/JS/scroll1.html](https://students.btsi.lu/evegi144/WAD/JS/scroll1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Scroll example 1</title>
    <meta charset=utf-8>
    <style>
      main {
        position:      fixed;
        top:            0;
        bottom:         0;
        left:           0;
        right:          0;
        background-color: greenyellow;
        overflow:       auto;
      }
    </style>
    <script>
      'use strict';

      const generateRandomText = (length, lineWidth) => {
        let string = "";
        const charset =
"abcdefghijklmnopqrstuvwxyz0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZ";

        for (let i = 1; i <= length; i++) {
          string += charset.charAt(Math.floor(Math.random() * charset.length));
          if (i % lineWidth === 0) string += '<br>';
        }
        return string;
      };

      const init = () => {
        const main = document.querySelector('main');
        main.innerHTML = generateRandomText(10000, 60);
        main.scrollTop = main.scrollHeight;
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main></main>
  </body>
</html>
```

## using `getAttribute` and `setAttribute`

Instead of using element properties, we can use two `HTMLElement` methods, one to get and one to set the value of an attribute. Note that attribute values are treated as strings, i.e. `getAttribute` always returns and `setAttribute` takes a string. The attribute names used are the standard HTML ones, not the camel case versions used with element properties. These methods can also be used with non standard attributes as well as attributes of XML documents.

[students.btsi.lu/evegi144/WAD/JS/DOM\\_attributes2.html](https://students.btsi.lu/evegi144/WAD/JS/DOM_attributes2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Illustrates access of HTML element attributes using methods</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      const toggleImage = () => {
        const image = document.querySelector('img'); // Get the image element.
        console.log(image.src); // Display the complete URL.
        // Extract the image filename from the complete URL.
        const imageName =

image.getAttribute('src').substring(image.getAttribute('src').lastIndexOf('/')
    + 1);
        // Given that the image is in the same directory as this document,
        // we do not need to specify the complete URL, just the image filename.
        if (imageName === 'camaro256x256.png') {
          image.setAttribute('src', 'ferrari256x256.png');
          image.setAttribute('title', 'https://www.iconfinder.com/icons/67532/'
            + 'car_ferrari_red_small_car_sports_car_icon#size=256');
        }
        else {
          image.setAttribute('src', 'camaro256x256.png');
          image.setAttribute('title', 'https://www.iconfinder.com/icons/67528/'
            + 'camaro_car_sports_car_icon#size=256');
        }
      };

      const toggleTitle = () => {
        const image = document.querySelector('img'); // Get the image element.
        const imageName =

image.getAttribute('src').substring(image.getAttribute('src').lastIndexOf('/')
    + 1);
        // If the image has a title attribute, we remove it.
        if (image.hasAttribute('title')) image.removeAttribute('title');
        // Otherwise, we need to determine the correct title to add.
        else if (imageName === 'camaro256x256.png')
          image.setAttribute('title', 'https://www.iconfinder.com/icons/67528/'
```

```

        + 'camaro_car_sports_car_icon#size=256');
    else
        image.setAttribute('title', 'https://www.iconfinder.com/icons/67532/'
        + 'car_ferrari_red_small_car_sports_car_icon#size=256');
    };
</script>
</head>
<body>
    <button onclick='toggleImage();'>Toggle image</button>
    <button onclick='toggleTitle();'>Toggle image title attribute</button>
    <br>
    <img src=camaro256x256.png alt=Car
title=https://www.iconfinder.com/icons/67528/camaro_car_sports_car_icon#size=256>
</body>
</html>

```

### using dataset attributes

On occasion we might want to add our own attributes to HTML elements in order to store specific information. In order to be HTML5-compliant, we need to prefix our attribute names with **data-**. Our attribute names may not contain capital letters A to Z or semicolons and may not start with **xml** (cf. [developer.mozilla.org/en-US/docs/Web/HTML/Global\\_attributes#attr-data-](https://developer.mozilla.org/en-US/docs/Web/HTML/Global_attributes#attr-data-)). A nice example can be found at [www.w3schools.com/tags/tryit.asp?filename=tryhtml5\\_global\\_data](http://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_global_data). We can access dataset attributes using methods or using the **dataset** property. In the latter case, attribute names are mapped to camel case property names. See also [developer.mozilla.org/en-US/docs/Web/API/HTMLElement.dataset](https://developer.mozilla.org/en-US/docs/Web/API/HTMLElement.dataset).

[students.btsi.lu/evegi144/WAD/JS/DOM\\_attributes3.html](http://students.btsi.lu/evegi144/WAD/JS/DOM_attributes3.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Illustrates access of HTML element dataset attributes</title>
    <!-- Based on
http://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_global_data-->
    <meta charset=utf-8>
    <style>
      li {
        width: 65px;
        cursor: pointer;
      }
    </style>
    <script>
      'use strict';

      const showDetails1 = animal => {
        const animalType = animal.getAttribute("data-animal-type");
        alert(`The ${animal.innerHTML} is a ${animalType}.`);
      };

      const showDetails2 = animal => {
        const animalType = animal.dataset.animalType;
        alert(`The ${animal.innerHTML} is a ${animalType}.`);
      };

    </script>
  </head>
  <body>
    <h1>Species</h1>

    <p>Click on a species to see what type it is:</p>
    <ul>
      <li onclick="showDetails1(this);" id=owl data-animal-type=bird>Owl</li>
      <li onclick="showDetails2(this);" id=salmon data-animal-type=fish>Salmon</li>
      <li onclick="showDetails1(this);" id=tarantula data-animal-
type=spider>Tarantula</li>
    </ul>
  </body>
</html>

```

### as **Attr** objects

From p. 378 of the 6th edition of "JavaScript The Definitive Guide":

The **Node** type defines an **attributes** property. This property is **null** for any nodes that are not **Element** objects. For **Element** objects, **attributes** is a read-only array-like object that represents all the attributes of the element.

The **Attr** object has **name** and **value** properties representing the name and value of the attribute.

[students.btsi.lu/evegi144/WAD/JS/DOM\\_attributes4.html](https://students.btsi.lu/evegi144/WAD/JS/DOM_attributes4.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Illustrates access of HTML element dataset attributes</title>
    <!-- Based on
http://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_global_data-->
    <meta charset=utf-8>
    <script>
      'use strict';

      const doWork = () => {
        const children = document.getElementsByTagName('ul')[0].childNodes;
        for (let i = 0; i < children.length; i++)
          if (children[i].attributes)
            for (let j = 0; j < children[i].attributes.length; j++)
              console.log(`Name: ${children[i].attributes[j].name} value: `
                + children[i].attributes[j].value);
      };

      window.addEventListener('load', doWork);
    </script>
  </head>
  <body>
    <h1>Species</h1>
    <ul>
      <li id=owl data-animal-type=bird>Owl</li>
      <li id=salmon data-animal-type=fish>Salmon</li>
      <li id=tarantula data-animal-type=spider>Tarantula</li>
    </ul>
  </body>
</html>
```

### Manipulating an element's classes

**element.classList** is a highly useful property to manipulate an element's classes. See [developer.mozilla.org/en-US/docs/Web/API/Element/classList](https://developer.mozilla.org/en-US/docs/Web/API/Element/classList) for the details.

### Element content

We can view the content of an element as HTML or plain text.

#### as HTML

The **innerHTML** property of an **Element** object contains the content as an HTML string. **outerHTML** contains the content, including the opening and closing tag of the element. With **insertAdjacentHTML** we can insert HTML before or after the beginning or before or after the end of a given element (cf.

`insertAdjacentHTML()` parses the specified text as HTML or XML and inserts the resulting nodes into the DOM tree at a specified position. **It does not reparse the element it is being used on and thus it does not corrupt the existing elements inside the element.** This, and avoiding the extra step of serialization make it much faster than direct `innerHTML` manipulation.

[students.btsi.lu/evegi144/WAD/JS/contentHTML.html](https://students.btsi.lu/evegi144/WAD/JS/contentHTML.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Element content as HTML</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      const toggleInnerHTML = () => {
        // First we need to know whether the outer element is a <main> or a <div>.
        const el = document.querySelector('main') ? 'main' : 'div';
        if (document.querySelector(el).innerHTML !== '')
          document.querySelector(el).innerHTML = '';
        else document.querySelector(el).innerHTML =
          '<section><h1>Content header</h1>Content</section>';
      };

      const toggleOuterHTML = () => {
        let el, el_replace;
        if (document.querySelector('main')) {
          el = 'main'; // We have a <main>
          el_replace = 'div'; // and want to replace it with a <div>.
        }
        else {
          el = 'div'; // We have a <div>
          el_replace = 'main'; // and want to replace it with a <main>
        }
        document.querySelector(el).outerHTML = `<${el_replace}>` +
          '<section><h1>Content header</h1>Content</section></${el_replace}>';
      };

      const addBeforeBegin = () => {
        document.querySelector('header').insertAdjacentHTML('beforebegin',
          '<h1>Header</h1>');
      };

      const addAfterBegin = () => {
        document.querySelector('header').insertAdjacentHTML('afterbegin',
          '<h1>Header</h1>');
```

```

    };

    const addBeforeEnd = () => {
        document.querySelector('header').insertAdjacentHTML('beforeend',
            '<h1>Header</h1>');
    };

    const addAfterEnd = () => {
        document.querySelector('header').insertAdjacentHTML('afterend',
            '<h1>Header</h1>');
    };

    const init = () => { // Register button event handlers.
        const handlers = [toggleInnerHTML, toggleOuterHTML, addBeforeBegin,
            addAfterBegin, addBeforeEnd, addAfterEnd];
        for (let i = 0; i < handlers.length; i++)
            document.getElementsByTagName('button')[i].addEventListener('click',
                handlers[i]);
    };

    window.addEventListener('load', init);
</script>
</head>
<body>
    <button>Toggle innerHTML</button>
    <button>Toggle outerHTML</button>
    <button>Header add before begin</button>
    <button>Header add after begin</button>
    <button>Header add before end</button>
    <button>Header add after end</button>
    <header>Header content</header>
    <main></main>
</body>
</html>

```

### as plain text

The `textContent` property of an `Element` object contains the content as plain text.

[students.btsi.lu/evegi144/WAD/JS/contentText.html](https://students.btsi.lu/evegi144/WAD/JS/contentText.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Element content as plain text</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      const changeText = () => document.querySelector('main').textContent =
        prompt('Please enter the new text');

      const init = () => // Register button event handler.
        document.getElementsByTagName('button')[0].addEventListener('click',
          changeText);

      window.addEventListener('load', init);
    </script>
  </head>
  <body>
    <button>Change text</button>
    <main>Text</main>
  </body>
</html>

```

## Managing nodes

We create a new element with `createElement` and a new text node with `createTextNode`. There are other creation methods available, as detailed in [developer.mozilla.org/en-US/docs/Web/API/Document](https://developer.mozilla.org/en-US/docs/Web/API/Document).

With `appendChild` we add a node as the last child of the given node. With `insertBefore` we insert the new node (first parameter) before a given node (second parameter). If the second parameter is `null`, the method behaves like `appendChild`.

`removeChild` is invoked on the parent node and given the child node that is to be removed as parameter. `replaceChild` is also invoked on the parent node. It takes the new node as first and the node to be replaced as second parameter.

[students.btsi.lu/evegi144/WAD/JS/manageNodes.html](https://students.btsi.lu/evegi144/WAD/JS/manageNodes.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Node management example</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      const init = () => {
        const button = document.createElement('button');
        button.textContent = 'Click to kill me!';
        // With a normal function, we have 3 options:
        button.addEventListener('click', function(e) {
          // this.parentElement.removeChild(this); // Does not work with arrow
function.
          e.target.parentElement.removeChild(e.target);
          // This should be avoided as button may not exist in the event handler.
          // button.parentElement.removeChild(button);
        });
        document.body.appendChild(button);
      };

      window.addEventListener('load', init);
    </script>
  </head>
  <body>
  </body>
</html>

```

## Document fragments

From [developer.mozilla.org/en-US/docs/Web/API/document.createDocumentFragment](https://developer.mozilla.org/en-US/docs/Web/API/document.createDocumentFragment):

DocumentFragments are DOM Nodes. They are never part of the main DOM tree. The usual use case is to create the document fragment, append elements to the document fragment and then append the document fragment to the DOM tree. In the DOM tree, the document fragment is replaced by all its children.

Since the document fragment is in memory and not part of the main DOM tree, appending children to it does not cause page reflow (computation of element's position and geometry). Consequently, using document fragments often results in better performance.

Cf. [jsperf.com/createdocumentfragmentvscreateelement](https://jsperf.com/createdocumentfragmentvscreateelement).

## Determining the dimensions of elements

[https://developer.mozilla.org/en-US/docs/Web/API/CSS\\_Object\\_Model/Determining\\_the\\_dimensions\\_of\\_elements](https://developer.mozilla.org/en-US/docs/Web/API/CSS_Object_Model/Determining_the_dimensions_of_elements)

## Manipulating CSS

[www.w3.org/wiki/Dynamic\\_style\\_-\\_manipulating\\_CSS\\_with\\_JavaScript](http://www.w3.org/wiki/Dynamic_style_-_manipulating_CSS_with_JavaScript)

[www.w3schools.com/js/js\\_htmlDOM\\_css.asp](http://www.w3schools.com/js/js_htmlDOM_css.asp)

[www.w3schools.com/jsref/dom\\_obj\\_style.asp](http://www.w3schools.com/jsref/dom_obj_style.asp)

[developer.mozilla.org/en-US/docs/Web/API/Window.getComputedStyle](https://developer.mozilla.org/en-US/docs/Web/API/Window.getComputedStyle)

[developer.mozilla.org/en-US/docs/Web/API/CSSStyleDeclaration](https://developer.mozilla.org/en-US/docs/Web/API/CSSStyleDeclaration)

We can easily manipulate CSS via JavaScript, which opens up some interesting applications, for instance moving HTML objects or changing colors dynamically.

In order to access internal or external stylesheets, we can use `document.styleSheets`, which gives us an array with all stylesheets used by the document.

[students.btsi.lu/evegi144/WAD/JS/DOM\\_CSS1.html](http://students.btsi.lu/evegi144/WAD/JS/DOM_CSS1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>DOM CSS manipulation example 1</title>
    <meta charset=UTF-8>
    <style>
      body {
        background-color: black;
      }
    </style>
    <script>
      'use strict';

      const switchToWhite = () => {
        document.styleSheets[0].cssRules[0].style.backgroundColor = "white";
        /* Or we can set the inline style of the body:
        document.body.style.backgroundColor = "white";
        */
      };
    </script>
  </head>
  <body>
    <button onclick=switchToWhite();>Set background color to white</button>
  </body>
</html>
```

In order to access the inline styling of a particular element, we can use the `style` attribute of that

element. For instance, if we want to change the color of the second **p** element in our document, we could write:

```
document.querySelectorAll('p')[1].style.color = "#F0F";
```



Let's look at a more dynamic example, where we move a gorilla:

[students.btsi.lu/evegi144/WAD/JS/DOM\\_CSS2.html](https://students.btsi.lu/evegi144/WAD/JS/DOM_CSS2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>DOM CSS manipulation example 2</title>
    <meta charset=UTF-8>
    <script>
      'use strict';

      let gorilla;

      const init = () => {
        gorilla = document.querySelector('img');
        gorilla.style.cssText = "position: relative; top: 0; left: 0";
      };

      const moveRight = () => {
        const x = parseInt(gorilla.style.left);
        gorilla.style.left = x + 10 + 'px';
      };

      const moveLeft = () => {
        const x = parseInt(gorilla.style.left);
        gorilla.style.left = x - 10 + 'px';
      };

      const moveUp = () => {
        const y = parseInt(gorilla.style.top);
        gorilla.style.top = y - 10 + 'px';
      };

      const moveDown = () => {
        const y = parseInt(gorilla.style.top);
        gorilla.style.top = y + 10 + 'px';
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <button onclick=moveLeft();></button>
    <button onclick=moveRight();></button>
    <button onclick=moveUp();>^</button>
    <button onclick=moveDown();>v</button>
    <img src=gorilla236x256.png>
  </body>
</html>
```

getComputedStyle

[developer.mozilla.org/en-US/docs/Web/API/Window/getComputedStyle](https://developer.mozilla.org/en-US/docs/Web/API/Window/getComputedStyle)

The `Window.getComputedStyle()` method gives the values of all the CSS properties of an element after applying the active stylesheets and resolving any basic computation those values may contain.

This method is particularly useful if we want to query an element's CSS value that we have not set programmatically.

[students.btsi.lu/evegi144/WAD/JS/DOM\\_CSS3.html](https://students.btsi.lu/evegi144/WAD/JS/DOM_CSS3.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>DOM CSS manipulation example 3</title>
    <meta charset=UTF-8>
    <script>
      'use strict';

      addEventListener('load', () =>

alert(getComputedStyle(document.querySelector('body')).getPropertyValue('color')));
    </script>
  </head>
  <body>
  </body>
</html>
```

### Inserting a new style sheet

From [developer.mozilla.org/en-US/docs/Web/API/CSSStyleSheet](https://developer.mozilla.org/en-US/docs/Web/API/CSSStyleSheet):

To create a new stylesheet, insert a `<style>` or `<link>` element into the document.

### Inserting JS dynamically

[students.btsi.lu/evegi144/WAD/JS/insertJS.html](https://students.btsi.lu/evegi144/WAD/JS/insertJS.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Insert JS</title>
  </head>
  <body>
    <script>
      const init = () => {
        const script = document.createElement('script');
        script.innerHTML = 'alert("Test");';
        document.body.appendChild(script);
      };

      addEventListener('load', init);
    </script>
  </body>
</html>
```

## Handling iframes

[www.dyn-web.com/tutorials/iframes](http://www.dyn-web.com/tutorials/iframes)

[stackoverflow.com/questions/6102636/html-code-as-iframe-source-rather-than-a-url](http://stackoverflow.com/questions/6102636/html-code-as-iframe-source-rather-than-a-url)

[javascript.info/tutorial/frames-and-iframes](http://javascript.info/tutorial/frames-and-iframes)

## Observing DOM mutations

[developer.mozilla.org/en-US/docs/Web/API/MutationObserver](https://developer.mozilla.org/en-US/docs/Web/API/MutationObserver)

[hacks.mozilla.org/2012/05/dom-mutationobserver-reacting-to-dom-changes-without-killing-browser-performance](https://hacks.mozilla.org/2012/05/dom-mutationobserver-reacting-to-dom-changes-without-killing-browser-performance)

[students.btsi.lu/evegi144/WAD/JS/mutationObserver1.html](http://students.btsi.lu/evegi144/WAD/JS/mutationObserver1.html)

```

'use strict';

const init = () => {
  // select the target node
  const target = document.querySelector('body');

  // create an observer instance
  const observer = new MutationObserver(mutations => {
    for (const mutation of mutations) console.dir(mutation);
  });

  // configuration of the observer:
  const config = {
    attributes: true, childList: true, characterData: true, subtree: true,
    attributeOldValue: true, characterDataOldValue: true
  };

  // pass in the target node, as well as the observer options
  observer.observe(target, config);

  document.querySelector('button').addEventListener('click', () => {
    observer.disconnect();
  });
};

window.addEventListener('load', init);

```

### 4.3.21. )

**window** is the global object put at our disposal by the browser. It is of central importance and described in detail in [developer.mozilla.org/en-US/docs/Web/API/Window](https://developer.mozilla.org/en-US/docs/Web/API/Window). In [Basic input and output](#) we have already met some useful **window** methods.

#### Timers

We can measure time, accurate to one microsecond, using the **Performance.now** method (cf. [developer.mozilla.org/en-US/docs/Web/API/Performance.now\(\)](https://developer.mozilla.org/en-US/docs/Web/API/Performance.now)).

We can choose between three timer methods.

##### **setTimeout**

**setTimeout** runs a given function (first parameter) after a specified number of milliseconds (second parameter) and returns the ID of the timeout, which can be used with **clearTimeout** if we change our mind and do not want the timer to execute. **setTimeout** is ideal if we just want to execute a function once after a given delay, as in the following example. Note that we need to use an anonymous function if we want to pass parameters to the function that is to be called by the timer:

[students.btsi.lu/evegi144/WAD/JS/setTimeout1.html](https://students.btsi.lu/evegi144/WAD/JS/setTimeout1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>setTimeout example 1</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <script>
        'use strict';

        const displayAlert = msg => alert(msg);

        setTimeout(() => displayAlert('Test'), 2000);
      </script>
    </main>
  </body>
</html>
```

We can, however, also repeat the function call by invoking `setTimeout` within the function that is to be executed repeatedly:

[students.btsi.lu/evegi144/WAD/JS/setTimeout2.html](https://students.btsi.lu/evegi144/WAD/JS/setTimeout2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>setTimeout example 2</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <script>
        'use strict';

        const countDown = x => {
          console.log(`Current counter: ${x}`);
          if (x > 0) setTimeout(() => countDown(x - 1), 1000);
        };

        countDown(10);
      </script>
    </main>
  </body>
</html>
```

## setInterval

`setInterval` is identical to `setTimeout` except that the given function gets invoked repeatedly until `clearInterval` is called with the timer ID.

[students.btsi.lu/evegi144/WAD/JS/setInterval.html](https://students.btsi.lu/evegi144/WAD/JS/setInterval.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>setInterval example</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      let counter = 10, timer; // Initialise.

      const countDown = () => {
        console.log(`Current counter: ${counter--}`); // Display and decrement.
        if (counter < 0) clearInterval(timer); // Stop timer.
      };

      timer = setInterval(countDown, 1000); // Start timer.
    </script>
  </head>
  <body>
    <main>
    </main>
  </body>
</html>
```



This is an example of how we can create a background color animation using DOM CSS

manipulation inside a timer function.

[students.btsi.lu/evegi144/WAD/JS/gradient\\_anim1.html](https://students.btsi.lu/evegi144/WAD/JS/gradient_anim1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Radial gradient animation</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      /*
       We create a radial gradient with an origin and a target.
       We set random origin colors. Target color are created via 255 - origin color.
       The red, green and blue components of the origin are animated upwards or
downwards.
       The RGB components of the target are animated in opposite directions.
      */
      const RGBColorsOrigin = [Math.floor(Math.random() * 256),
Math.floor(Math.random() *
      256),
      Math.floor(Math.random() * 256)],
      RGBColorsTarget = [255 - RGBColorsOrigin[0], 255 - RGBColorsOrigin[1], 255 -
      RGBColorsOrigin[2]];
      const RGBDirections = [Math.random() >= 0.5 ? 1 : -1, Math.random() >= 0.5 ? 1 :
-1,
      Math.random() >= 0.5 ? 1 : -1], RGBDirectionsOrigin = [-RGBDirections[0],
      -RGBDirections[1], -RGBDirections[2]];

      const changeBackground = () => {
        for (let i = 0; i <= 2; i++) {
          if (RGBColorsOrigin[i] >= 255 && RGBDirectionsOrigin[i] === 1 ||
RGBColorsOrigin[i]
          <= 0 && RGBDirectionsOrigin[i] === -1)
            RGBDirectionsOrigin[i] = -RGBDirectionsOrigin[i];
          RGBColorsOrigin[i] += RGBDirectionsOrigin[i] * Math.floor(Math.random() * 2
+ 0.5);
          if (RGBColorsTarget[i] >= 254 && RGBDirections[i] === 1 ||
RGBColorsTarget[i] <= 1 &&
          RGBDirections[i] === -1) RGBDirections[i] = -RGBDirections[i];
          RGBColorsTarget[i] += RGBDirections[i] * Math.floor(Math.random() * 2 +
0.5);
        }
        const originColor = `rgb(${RGBColorsOrigin[0]}, ${RGBColorsOrigin[1]}, ` +
          `${RGBColorsOrigin[2]})`,
          targetColor = `, rgb(${RGBColorsTarget[0]}, ${RGBColorsTarget[1]}, ` +
            `${RGBColorsTarget[2]}) no-repeat fixed`;
        document.querySelector('body').style.background =
          `radial-gradient(${originColor}${targetColor}`;
      };
    </script>
  </head>
</html>
```

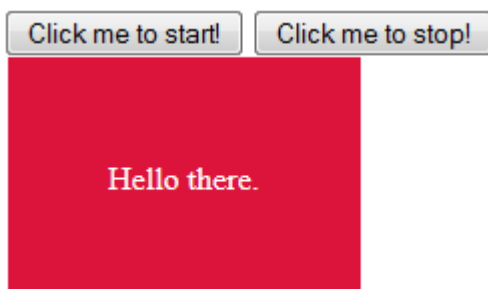
```
    addEventListener('load', () => setInterval(changeBackground, 100));  
</script>  
</head>  
<body>  
  
    </body>  
</html>
```

requestAnimationFrame

You should call this method whenever you're ready to update your animation onscreen. This will request that your animation function be called before the browser performs the next repaint. The number of callbacks is usually 60 times per second, but will generally match the display refresh rate in most web browsers as per W3C recommendation. The callback rate may be reduced to a lower rate when running in background tabs.

[developer.mozilla.org/en-US/docs/Web/API/window.requestAnimationFrame](https://developer.mozilla.org/en-US/docs/Web/API/window.requestAnimationFrame)

[www.w3.org/TR/animation-timing/#requestAnimationFrame](https://www.w3.org/TR/animation-timing/#requestAnimationFrame)



[students.btsi.lu/evegi144/WAD/JS/requestAnimationFrame1.html](https://students.btsi.lu/evegi144/WAD/JS/requestAnimationFrame1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <!-- https://developer.mozilla.org/en-US/docs/Web/API/window.requestAnimationFrame-->
    <title>requestAnimationFrame example</title>
    <meta charset=utf-8>
    <style>
      div {
        position: absolute;
        left: 10px;
        padding: 50px;
        background: crimson;
        color: white
      }
    </style>
  </head>
  <body>
    <script>
      "use strict";

      let requestId = 0, animationStartTime;

      const animate = time => {
        document.getElementById("animated").style.left =
          `${(time - animationStartTime) % 2000 / 4}px`;
        requestId = window.requestAnimationFrame(animate);
      };

      const start = () => {
        if (!requestId) {
          animationStartTime = window.performance.now();
          requestId = window.requestAnimationFrame(animate);
        }
      };

      const stop = () => {
        if (requestId)
          window.cancelAnimationFrame(requestId);
        requestId = 0;
      };
    </script>
    <button onclick="start()">Click me to start!</button>
    <button onclick="stop()">Click me to stop!</button>
    <div id="animated">Hello there.</div>
  </body>
</html>

```

## The location object

This object allows us to redirect the browser to another page. See [www.w3schools.com/jsref/obj\\_location.asp](http://www.w3schools.com/jsref/obj_location.asp) and [developer.mozilla.org/en-US/docs/Web/API/Window.location](https://developer.mozilla.org/en-US/docs/Web/API/Window.location).

## The navigator object

See [www.w3schools.com/jsref/obj\\_navigator.asp](http://www.w3schools.com/jsref/obj_navigator.asp) and [developer.mozilla.org/en-US/docs/Web/API/Navigator](https://developer.mozilla.org/en-US/docs/Web/API/Navigator).

Here is an example that checks once per second whether the browser is online: [students.btsi.lu/evegi144/WAD/JS/onOffline.html](http://students.btsi.lu/evegi144/WAD/JS/onOffline.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>On/offline detector</title>
    <script>
      'use strict';

      const init = () => {
        let onlineStatus = true;
        const checkOnlineStatus = () => {
          console.log(`Currently: ${onlineStatus}`);
          if (navigator.onLine !== onlineStatus) {
            onlineStatus = navigator.onLine;
            alert(`Online status changed to ${onlineStatus}`);
          }
        };

        setInterval(checkOnlineStatus, 1000);
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
  </body>
</html>
```

## The history object

See [www.w3schools.com/jsref/obj\\_history.asp](http://www.w3schools.com/jsref/obj_history.asp) and [developer.mozilla.org/en-US/docs/Web/API/History](https://developer.mozilla.org/en-US/docs/Web/API/History).

## The screen object

See [www.w3schools.com/jsref/obj\\_screen.asp](http://www.w3schools.com/jsref/obj_screen.asp) and [developer.mozilla.org/en-US/docs/Web/API/Window.screen](https://developer.mozilla.org/en-US/docs/Web/API/Window.screen).

### 4.3.22. Strict mode

The strict mode, according to ECMA, "provides enhanced error checking and program security". To enter this mode just put `'use strict'` or `"use strict";` at the top of any script. See [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Strict\\_mode](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Strict_mode) and [www.ecma-international.org/ecma-262/#sec-strict-mode-of-ecmascript](http://www.ecma-international.org/ecma-262/#sec-strict-mode-of-ecmascript).

To find out whether we are currently in strict mode, we can use this one liner from page 167 of the 6th edition of "JavaScript The Definitive Guide":

```
const strict = (function() { return !this; } ());
```

This works because in non-strict mode, the invocation context of a function is the global object, therefore `!this` will be false. In strict mode, the invocation context is `undefined`, thus `!this` will be true.

In Firefox, you can set strict mode to default by changing `javascript.options.strict` to `true` in `about:config`.

### 4.3.23. Dates

<http://www.htmlgoodies.com/html5/javascript/date-parsing-using-javascript-and-regular-expressions.html>

<http://stackoverflow.com/questions/6525538/convert-utc-date-time-to-local-date-time-using-javascript>

### 4.3.24. Modules

From [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/import](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Statements/import):

The import statement is used to import functions, objects, or primitives which are defined in and exported by an external module, script, or the like.

You might have to change a flag in your browser in order to enable this feature (cf. [caniuse.com/#search=export](http://caniuse.com/#search=export) and [jakearchibald.com/2017/es-modules-in-browsers](http://jakearchibald.com/2017/es-modules-in-browsers)).

See [exploringjs.com/es6/ch\\_modules.html](http://exploringjs.com/es6/ch_modules.html) and [developers.google.com/web/fundamentals/primers/modules](http://developers.google.com/web/fundamentals/primers/modules) for a detailed explanation of JS modules.

### 4.3.25. Destructuring

See [exploringjs.com/es6/index.html#toc\\_ch\\_destructuring](http://exploringjs.com/es6/index.html#toc_ch_destructuring).

### 4.3.26. Viewports

[www.quirksmode.org/mobile/viewports.html](http://www.quirksmode.org/mobile/viewports.html)

### 4.3.27. AJAX

From [developer.mozilla.org/en-US/docs/Glossary/AJAX](http://developer.mozilla.org/en-US/docs/Glossary/AJAX):

Asynchronous JavaScript And XML (AJAX) is a term used to describe the programming practice of using HTML, CSS, JavaScript, the Document Object Model (DOM), and the XMLHttpRequest object together to build complex Web pages that update their content without reloading the entire Web page. This makes the application faster and more responsive to user actions.

Study the gentle introduction to AJAX at [www.w3schools.com/xml/ajax\\_intro.asp](http://www.w3schools.com/xml/ajax_intro.asp).

#### XMLHttpRequest

The key object that enables AJAX is `XMLHttpRequest` (cf. [developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest](http://developer.mozilla.org/en-US/docs/Web/API/XMLHttpRequest)), which allows us to take control of HTTP communication with a server, whereas normally this is handled in the background by the browser.

The required steps are the following:

1. Create an `XMLHttpRequest` object.
2. Prepare the request using `open` and register the event handler to handle the response.
3. Send the request to the server using `send`.
4. When the server sends a response, an event gets triggered and our event handler reads the data and takes the required action, for instance update some parts of our web without a page reload.

It's as easy as this:

Click me and something wonderful will happen without page reload!

[students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX1.html](http://students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Very basic AJAX example</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const displayResponse = e =>
        document.querySelector('main').innerHTML = e.target.response;

      const AJAX = () => {
        const req = new XMLHttpRequest(); // Create new request.
        req.open('POST', 'AJAX1.php'); // Specify the method and script to be used.
        // Add the event listener for successful request completion.
        req.addEventListener('load', displayResponse);
        req.send(); // Send the request to the server.
      };
    </script>
  </head>
  <body>
    <main>
      <button onclick=AJAX();>Click me and something wonderful will happen without
page
      reload!
    </button>
  </main>
</body>
</html>

```

[students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX1.php](https://students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX1.php)

```

<?php
  echo '<h1>Hello world!</h1>';
?>

```

`open(method, url, async, user, password)`

For our purposes the relevant methods are `POST` and `GET` (cf. [Forms](#)). The second parameter represents the location of the server script that will receive the request. The third parameter is optional and `true` by default, which specifies an asynchronous request, i.e. our script will not block to wait for the response. Instead, when the server sends a response, an event will be triggered to which our script can react. If this parameter is set to `false`, our script will block and wait for the server response. Parameters four and five are only needed to access password protected resources.

`send(data)`

`send` sends the request to the server. Any data that we want to send is given as argument. The following types can be used: `ArrayBuffer`, `ArrayBufferView`, `Blob`, `Document`, `DOMString` and `FormData`. To

send binary data we should use `Blob` (cf. [developer.mozilla.org/en-US/docs/Web/API/Blob](https://developer.mozilla.org/en-US/docs/Web/API/Blob)) or `ArrayBufferView` (cf. [developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global\\_Objects/TypedArray](https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/TypedArray)) objects.

The `FormData` object (cf. [developer.mozilla.org/en-US/docs/Web/API/FormData](https://developer.mozilla.org/en-US/docs/Web/API/FormData)) allows us to programmatically send form data to the server just like the user can when sending an HTML form. To use it we simply create a new `FormData` object and then add name/value pairs to it using the `append(name, value)` method. The `name` parameter is a string whereas the `value` parameter can be a string or a `Blob` or a `File`. In the latter two cases, a third optional parameter specifies the filename to be reported to the server. For `Blob` objects the default filename is "blob". We can also use `FormData` to submit an HTML form asynchronously (cf. [developer.mozilla.org/en-US/docs/Web/Guide/Using\\_FormData\\_Objects](https://developer.mozilla.org/en-US/docs/Web/Guide/Using_FormData_Objects)).

## `abort`

This method aborts the request if it has already been sent.

## Events

The following events are relevant for the `XMLHttpRequest` object:

<code>loadstart</code>	triggered on request start
<code>progress</code>	triggered periodically during request execution
<code>abort</code>	triggered on request abortion
<code>error</code>	triggered if a request error occurs
<code>load</code>	triggered on successful request completion
<code>timeout</code>	triggered on request timeout
<code>loadend</code>	triggered after <code>load</code> , <code>abort</code> or <code>error</code> have been dispatched

We need to register a handler that takes care of the `load` event. As we have seen in [Events](#), all event handlers automatically receive an `Event` object when invoked. The `target` attribute of the event corresponds to our `XMLHttpRequest` object, given that we registered the event handler on this object. The `status` attribute contains the HTTP response code (cf. [developer.mozilla.org/en-US/docs/Web/HTTP/Response\\_codes](https://developer.mozilla.org/en-US/docs/Web/HTTP/Response_codes)) and `statusText` the status in text form. If this code has the value 200, the request has succeeded and we can use the response data. A specific response header can be queried using `getResponseHeader(header)` or we can retrieve a string with all response headers using `getAllResponseHeaders`. Note that cookie headers are automatically filtered out. The `XMLHttpRequest` object has three properties to contain the response data. `response` contains the response in the format specified by the `responseType` property. `responseText` has the response in text format and `responseXML` as a `Document` object in parsed XML format, which can then be traversed as described in [Traversing the DOM](#), if applicable.

Let's look at a couple more examples:

First name

Last name

Hello Mickey Mouse

Response headers:

```
Connection: Keep-Alive
Content-Length: 19
Content-Type: text/html; charset=UTF-8
Date: Tue, 12 Sep 2017 08:04:48 GMT
Keep-Alive: timeout=5, max=99
Server: Apache
Strict-Transport-Security: max-age=63072000; includeSubdomains; preload
X-Frame-Options: SAMEORIGIN
X-XSS-Protection: 1; mode=block
```

[students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX2.html](http://students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Illustrates the programmatic creation and submission of a form without
      page reload</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      // The event handler reads the data from the server and displays it.
      const displayData = e => {
        console.log(e);
        const p = document.createElement('p'); // Create new paragraph element
        // and set its content to server response data.
        p.innerHTML = e.target.response + '<br>Response headers: ';
        p.innerHTML += '<pre>' + e.target.getAllResponseHeaders() + '</pre>';
        // Append the paragraph to the main element.
        document.querySelector('main').appendChild(p);
      };

      // This event handler gets triggered when the HTML document has finished
      loading.
      const init = () => {
        // Here we create a form, fill it with data and send it to the server.
        let data = new FormData(), req = new XMLHttpRequest();
        // Fill in the form.
        data.append('first_name', 'Mickey');
        data.append('last_name', 'Mouse');
        // Set the event listener to be triggered upon successful request completion.
        req.addEventListener('load', displayData);
        // Open the HTTP connection to the server script using POST method.
        req.open('POST', 'AJAX2.php');
        req.send(data); // Send the form to the server.
```

```
// When the user clicks on submit, we submit the form asynchronously.
document.forms[0].addEventListener('submit', e => {
    e.preventDefault(); // Prevent default form submission
    // Create a new form and a new XMLHttpRequest.
    data = new FormData(document.forms[0]), req = new XMLHttpRequest();
    req.addEventListener('load', displayData);
    // Open the HTTP connection to the server script using POST method.
    req.open('POST', 'AJAX2.php');
    req.send(data); // Send the form to the server.
});
};

window.addEventListener('load', init);
</script>
</head>
<body>
    <main>
        <form method=post>
            <label>First name<input name=first_name required></label><br>
            <label>Last name<input name=last_name required></label><br>
            <button>Log in</button>
        </form>
    </main>
</body>
</html>
```

[students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX2.php](https://students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX2.php)

```
<?php
    if (isset($_POST['first_name']) && isset($_POST['last_name']))
        echo 'Hello ' . $_POST['first_name'] . ' ' . $_POST['last_name'];
?>
```

Data set 3 ▾

First Name	Middle Initials	Last Name	Date of Birth	Street Address	Zip Code	City	State	Gender	Phone Number
Tashina		Oomen	1993-11-29	P.O. Box 22612	29149	Eldora	Alabama	F	261-982-6079
Cory	G.V.	Flips	1980-12-04	47 Little Lagoon Farms	63998	Ideal	Idaho	M	401-513-5610
Danelle	Z.B.T.	Nedermeyer	1963-08-15	943 SW Rest Glen	08378	San Rafael	Oklahoma	F	718-736-5329
Tomasa	R.W.	Ramsey	1976-04-23	289 Dewy Deer Track	49756	Shaniko	New Hampshire	F	812-732-9593
Alejandrina	W.A.M.	Rossum	1953-05-15	542 Little Forest	03260	Amsterdam village	Alaska	F	708-684-3552
Taylor		Carrillo	1992-03-24	240 Silver End	12718	Stateburg	Hawaii	M	781-835-8396
Otha		van der Mheen	1944-12-02	724 Honey Isle	67361	Manchester village	Oregon	M	656-765-3125
Rocky	Z.	Anthonisz	1955-11-13	P.O. Box 84497	20464	Ortonville	Virginia	M	613-754-7417
Ernesto	Q.R.E.	de Weerd	1954-06-29	851 Fallen Well Chase	15155	Sheridan village	Delaware	M	517-990-0863
Kendrick		Guerts	1963-08-21	952 Thunder Rapid Jetty	56774	Ramona	Connecticut	M	934-409-8773

[students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX3.html](https://students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX3.html)

```
<!DOCTYPE html>
<html lang=en>
```

```

<head>
  <title>Illustrates reading comma separated files and displaying them</title>
  <meta charset=utf-8>
  <style>
    select {
      position: fixed;
      top:      0;
      left:     0;
    }

    table {
      border-collapse: collapse;
      position:       relative;
      left:           100px;
    }

    th, td {
      border:      1px solid red;
      padding:     5px;
      text-align:  left;
    }

    aside {
      position:     absolute;
      width:        30px;
      height:       30px;
      background:   repeating-linear-gradient(-45deg, red, red 5px, white 5px,
white 10px);
      border-radius: 15px;
      animation:    asideAnimation 5s infinite alternate;
    }

    @keyframes asideAnimation {
      0% {
        left: 500px;
        top:  0;
      }

      50% {
        left: 250px;
        top: 300px;
      }

      100% {
        left: 0px;
        top: 100px;
      }
    }
  </style>
  <script>
    "use strict";

```

```

const displayData = () => {
  const index = document.querySelector('select').selectedIndex;
  const req = new XMLHttpRequest();
  req.open('POST', `testdata${index + 1}.csv`);
  req.addEventListener('load', e => {
    const oldTable = document.querySelector('table');
    const newTable = document.createElement('table');
    if (oldTable) document.querySelector('main').replaceChild(newTable,
oldTable);
    else document.querySelector('main').appendChild(newTable);
    const lines = e.target.responseText.split('\n'); // Explode rows.
    let line, tableRow, cellTag, cell;
    for (let lineIdx = 0; lineIdx < lines.length; lineIdx++) {
      line = lines[lineIdx].split('|'); // Explode data fields.
      tableRow = document.createElement('tr');
      if (lineIdx === 0) cellTag = 'th'; // head cell
      else cellTag = 'td'; // body cell
      for (let cellIdx = 0; cellIdx < line.length; cellIdx++) {
        cell = document.createElement(cellTag);
        cell.textContent = line[cellIdx];
        tableRow.appendChild(cell);
      }
      newTable.appendChild(tableRow);
    }
    console.log(`Response type: ${e.target.responseType}`);
    console.log('e.target.response:');
    console.log(e.target.response);
    console.log('e.target.responseText:');
    console.log(e.target.responseText);
    console.log('e.target.responseXML:');
    console.log(e.target.responseXML);
  });
  req.send();
};

const init = () => {
  document.querySelector('select').addEventListener('change', displayData);
  displayData(); // Make sure the first file gets displayed at startup.
};

window.addEventListener('load', init);
</script>
</head>
<body>
  <main>
    <select>
      <option>Data set 1</option>
      <option>Data set 2</option>
      <option>Data set 3</option>
    </select>

```

```
</main>
</body>
</html>
```

We can easily use AJAX to upload files to a server. If we want to monitor the upload progress, all we need to do is assign an event handler to the `upload` property of our `XMLHttpRequest`. This handler will automatically receive a `ProgressEvent` (cf. [developer.mozilla.org/en-US/docs/Web/API/ProgressEvent](https://developer.mozilla.org/en-US/docs/Web/API/ProgressEvent)) object as argument. Using the three properties `lengthComputable`, `loaded` and `total` we can monitor the upload progress.

[students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX4.html](https://students.btsi.lu/evegi144/WAD/JS/AJAX/AJAX4.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>File upload to server</title>
    <meta charset=utf-8>
    <style>
      section {
        width: 500px;
        resize: both;
        overflow: auto;
        border: 2px groove darkorange;
        text-align: center;
      }
    </style>
    <script>
      // Adapted from the 2nd edition of "HTML5 for Masterminds" p. 396-398.
      'use strict';

      let dataBox;

      const init = () => {
        dataBox = document.querySelector('section');
        dataBox.addEventListener('dragenter', e => e.preventDefault());
        dataBox.addEventListener('dragover', e => e.preventDefault());
        dataBox.addEventListener('drop', dropped);
      };

      const dropped = e => {
        e.preventDefault();
        const files = e.dataTransfer.files;
        if (files.length) {
          let list = '';
          for (let f = 0; f < files.length; f++) {
            list += '<div>File: ${files[f].name}';
            list += '<br><span><progress value=0 max=100>0%</progress></span>';
            list += '</div>';
          }
        }
      };
    </script>
  </head>
  <body>
    <div id=dataBox></div>
  </body>
</html>
```

```

dataBox.innerHTML = list;
let count = 0;
const upload = () => {
  const myfile = files[count];
  const data = new FormData();
  data.append('file', myfile);
  const url = 'AJAX4.php';
  const request = new XMLHttpRequest();
  const xmlupload = request.upload;
  xmlupload.addEventListener('progress', e => {
    if (e.lengthComputable) {
      let child = count + 1;
      const per = parseInt(e.loaded / e.total * 100);
      const progressBar =
        dataBox.querySelector(`div:nth-child(${child}) > span > progress`);
      progressBar.value = per;
      progressBar.innerHTML = `${per}%`;
    }
  });
  request.addEventListener('load', () => {
    const child = count + 1;
    const elem = dataBox.querySelector(`div:nth-child(${child}) > span`);
    elem.innerHTML = 'done!';
    count++;
    if (count < files.length) upload();
  });
  request.open('POST', url);
  request.send(data);
};
upload();
}
};

window.addEventListener('load', init);
</script>
</head>
<body>
  <main>
    <section>
      <p>Drop files here</p>
    </section>
  </main>
</body>
</html>

```

```
<?PHP
// Only allow authenticated users to upload files to your server!
//move_uploaded_file($_FILES['file']['tmp_name'],
'upload/' . $_FILES['file']['name']);
?>
```

`setRequestHeader(header, value)`

In some cases we need to specify specific HTTP headers to give the server additional information with regards to the data we want to send and/or receive. `setRequestHeader` sets the value of the HTTP request header. If used, **it must be called after open but before send**. If this method is called several times with the same header, the values are merged into one single request header. The official header list can be found at [www.iana.org/assignments/message-headers/message-headers.xml#perm-headers](http://www.iana.org/assignments/message-headers/message-headers.xml#perm-headers) and the official value list for the `Content-Encoding` header, which is the most often used one for our purposes, can be found at [www.iana.org/assignments/media-types/media-types.xhtml](http://www.iana.org/assignments/media-types/media-types.xhtml).

## Cross-origin requests

According to [developer.mozilla.org/en-US/docs/Web/Security/Same-origin\\_policy](http://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy):

The same-origin policy restricts how a document or script loaded from one origin can interact with a resource from another origin.

Cross-Origin Resource Sharing (CORS) is one way to get around these restrictions. The details can be found at [developer.mozilla.org/en-US/docs/Web/HTTP/Access\\_control\\_CORS](http://developer.mozilla.org/en-US/docs/Web/HTTP/Access_control_CORS).

In its simplest form, to give everyone access, we can just add the following at the top of our PHP script:

```
header('Access-Control-Allow-Origin: *');
```

Another possibility is to use JSONP (JSON with padding) as explained in [en.wikipedia.org/wiki/JSONP](http://en.wikipedia.org/wiki/JSONP). For a practical application example, study [WMOTU Invaders](#).

## Fetch

From [developer.mozilla.org/en-US/docs/Web/API/Fetch\\_API/Using\\_Fetch](http://developer.mozilla.org/en-US/docs/Web/API/Fetch_API/Using_Fetch):

The Fetch API provides a JavaScript interface for accessing and manipulating parts of the HTTP pipeline, such as requests and responses. It also provides a global `fetch()` method that provides an easy, logical way to fetch resources asynchronously across the network.

This kind of functionality was previously achieved using `XMLHttpRequest`. Fetch provides a better alternative that can be easily used by other technologies such as Service Workers. Fetch also provides a single logical place to define other HTTP-related concepts such as CORS and extensions to HTTP.

[students.btsi.lu/evegi144/WAD/JS/AJAX/fetch1.html](https://students.btsi.lu/evegi144/WAD/JS/AJAX/fetch1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Fetch demo</title>
    <script>
      'use strict';

      const init = () => {
        fetch('../camaro256x256.png').then(response => {
          if (response.ok)
            response.blob().then(blob =>
              document.querySelector('img').src = URL.createObjectURL(blob));
          else console.log('Network response was not ok. ');
        }).catch(error =>
          console.log(`There has been a problem with your fetch operation:
${error.message}`));

        // Old fashioned alternative
        const req = new XMLHttpRequest();
        req.open('GET', '../ferrari256x256.png');
        // cf. http://stackoverflow.com/questions/20035615/using-raw-image-data-from-
ajax-request-for-data-uri
        req.responseType = 'arraybuffer';
        req.addEventListener('load', e => {
          const blob = new Blob([e.target.response]);
          document.querySelectorAll('img')[1].src = URL.createObjectURL(blob);
        });
        req.send();
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <img>
    <img>
  </body>
</html>

```

Study the following links for a detailed demonstration and explanation of the differences between fetch and XMLHttpRequest:

[jakearchibald.com/2015/thats-so-fetch](http://jakearchibald.com/2015/thats-so-fetch)

[developers.google.com/web/updates/2015/03/introduction-to-fetch](https://developers.google.com/web/updates/2015/03/introduction-to-fetch)

### 4.3.28. JSON

JavaScript Object Notation (JSON) is a data serialization format often used to exchange data, including complex objects, between server and client. Objects are converted into a JSON string, which is sent to/from the server from/to the client.

JSON is based on a subset of JavaScript (cf. [json.org](https://json.org), [www.w3schools.com/json/default.asp](https://www.w3schools.com/json/default.asp) and [en.wikipedia.org/wiki/JSON](https://en.wikipedia.org/wiki/JSON)). Data consists of name/value pairs separated by commas and embedded within `{}`. `[]` are used for arrays.



Key names and strings need to be enclosed in double quotes.

Use [jsonlint.com](https://jsonlint.com) to verify that a given string is valid JSON.

If you use a version of Firefox that is older than version 53, you should turn the JSON viewer on as described in [www.ghacks.net/2017/01/12/firefox-53-json-viewer-on-by-default](https://www.ghacks.net/2017/01/12/firefox-53-json-viewer-on-by-default).

In JavaScript, we use `JSON.stringify` to encode and `JSON.parse` to decode a JSON string.

You can play around with JSON at [www.tutorialspoint.com/online\\_json\\_editor.htm](https://www.tutorialspoint.com/online_json_editor.htm).

Example:

```
const myObj = {firstName: "Donald", lastName: "Duck"};
// Convert JS object to JSON string: '{"firstName":"Donald","lastName":"Duck"}'
console.log(JSON.stringify(myObj));
// -> Convert string back to JS object: {firstName: "Donald", lastName: "Duck"}
JSON.parse(JSON.stringify(myObj));
```

In PHP, we should first tell the browser to expect to receive JSON by sending the header `application/json`. We use `json_encode` to encode a PHP object into a JSON string and `json_decode` to decode a JSON string into a PHP object. Note that the latter can be given a second parameter in order to have returned objects converted into associative arrays.

[students.btsi.lu/evegi144/WAD/JS/JSON/JSON1.html](https://students.btsi.lu/evegi144/WAD/JS/JSON/JSON1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Simple JSON data retrieval via AJAX and PHP</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      let dataJSON;

      const displayData = data => {
        const output = document.querySelector('p');
        console.dir(data);
        const keys = Object.keys(data); // Get the keys of the object.
        for (let i = 0; i < keys.length; i++) { // Display the key/value pairs.
          output.innerHTML += `${keys[i]}: ${data[keys[i]]}<br>`;
        }
      };

      const init = () => {
        // Old approach
        const req = new XMLHttpRequest();
        req.open('POST', 'JSON1.php');
        req.addEventListener('load', e => {
          displayData(JSON.parse(e.target.response));
        });
        req.send();

        // New approach
        fetch('JSON1.php', {
          headers: {
            'Accept': 'application/json',
            'Content-Type': 'application/json'
          },
          method: "POST",
        }).then(response => response.json()).then(data => {
          displayData(data);
        }).catch(error =>
          console.log(`There has been a problem with the fetch operation:
${error.message}`));
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main><p></p></main>
  </body>
</html>

```

```
<?php
header("Content-Type: application/json; charset=UTF-8");
$arr = array("INFOR" => 53, "MATHE" => 45);
echo json_encode($arr); // Send the array as a JSON string to the client browser.
?>
```

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Sending JavaScript array and object to PHP and back</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const displayData = data => {
        const output = document.querySelector('p');
        const keys = Object.keys(data); // Get the keys of the object.
        for (let i = 0; i < keys.length; i++) { // Display the key/value pairs.
          output.innerHTML += `${keys[i]}: ${data[keys[i]]}<br>`;
        }
      };

      const init = () => {
        // Old approach
        const req1 = new XMLHttpRequest(), req2 = new XMLHttpRequest();
        const arr = ['s1', 's2']; // A simple array.
        const obj = {width: 15, name: 'abc'}; // A simple object.
        req1.open('POST', 'JSON2a.php');
        req1.addEventListener('load', e =>
displayData(JSON.parse(e.target.response)));
        req1.send(JSON.stringify(arr)); // Send array as JSON string to the server
script.
        req2.open('POST', 'JSON2b.php');
        req2.addEventListener('load', e =>
displayData(JSON.parse(e.target.response)));
        req2.send(JSON.stringify(obj)); // Send object as JSON string to server
script.

        // New approach
        const headers = {
          'Accept': 'application/json',
          'Content-Type': 'application/json'
        };
        fetch('JSON2a.php', {
          headers: headers,
```

```

        method: "POST",
        body: JSON.stringify(arr)
    }).then(response => response.json()).then(data => displayData(data)).catch(
        error =>
            console.log(`There has been a problem with the fetch operation:
${error.message}`));
    fetch('JSON2b.php', {
        headers: headers,
        method: "POST",
        body: JSON.stringify(obj)
    }).then(response => response.json()).then(data => displayData(data)).catch(
        error =>
            console.log(`There has been a problem with the fetch operation:
${error.message}`));
    };

    addEventListener('load', init);
</script>
</head>
<body>
    <main><p></p></main>
</body>
</html>

```

[students.btsi.lu/evegi144/WAD/JS/JSON/JSON2a.php](https://students.btsi.lu/evegi144/WAD/JS/JSON/JSON2a.php)

```

<?php
header("Content-Type: application/json; charset=UTF-8");
// http://stackoverflow.com/questions/8599595/send-json-data-from-javascript-to-php
$arr = json_decode(file_get_contents('php://input')); // Decode the JSON string.
$arr[0] = 'OK'; // Modify it.
echo json_encode($arr); // Send the array as a JSON string to the client browser.
?>

```

[students.btsi.lu/evegi144/WAD/JS/JSON/JSON2b.php](https://students.btsi.lu/evegi144/WAD/JS/JSON/JSON2b.php)

```

<?php
header("Content-Type: application/json; charset=UTF-8");
// http://stackoverflow.com/questions/8599595/send-json-data-from-javascript-to-php
$obj = json_decode(file_get_contents('php://input')); // Decode the JSON string
$obj->name = 'OK'; // Modify it.
echo json_encode($obj); // Send the object as a JSON string to the client browser.
?>

```

Here is a simple example of reading JSON data from a text file and displaying it in an HTML table:

[students.btsi.lu/evegi144/WAD/JS/JSON/cars.json](https://students.btsi.lu/evegi144/WAD/JS/JSON/cars.json)

```
[
  {
    "brand": "BMW",
    "model": "120d",
    "colour": "black",
    "consumption": 4.5
  },
  {
    "brand": "Audi",
    "model": "A5",
    "colour": "green",
    "consumption": 6.7
  }
]
```

[students.btsi.lu/evegi144/WAD/JS/JSON/JSON3.html](https://students.btsi.lu/evegi144/WAD/JS/JSON/JSON3.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Reading JSON data from a text file</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const displayData = e => {
        const data = JSON.parse(e.target.response);
        const output =
          `<table><tr><th>brand</th><th>model</th><th>colour</th><th>consumption</th>
            </tr><tr><td>${data[0]['brand']}</td><td>${data[0]['model']}</td>
            <td>${data[0]['colour']}</td><td>${data[0]['consumption']}</td></tr>
            <tr><td>${data[1]['brand']}</td><td>${data[1]['model']}</td>
            <td>${data[1]['colour']}</td><td>${data[1]['consumption']}</td></tr></table>`;
        console.dir(data);
        document.querySelector('p').innerHTML = output;
      };

      const init = () => {
        const req = new XMLHttpRequest();
        req.open('POST', 'cars.json');
        req.addEventListener('load', displayData);
        req.send();
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main><p></p>
    </main>
  </body>
</html>

```

Here is an example of exchanging more complex objects between client and server:

[students.btsi.lu/evegi144/WAD/JS/JSON/JSON4.html](https://students.btsi.lu/evegi144/WAD/JS/JSON/JSON4.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Sending complex JS objects to PHP and back</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const displayData = e => {
        const dataJSON = JSON.parse(e.target.response);
        console.dir(dataJSON);
      };

      const init = () => {
        const req = new XMLHttpRequest();
        const obj1 = {elems: [1, 2, 3, 4, 5], name: 'abc'};
        const obj2 = {elems: [6, 7, 8, 9, 10], name: 'def'};
        const arr = [obj1, obj2];
        req.open('POST', 'JSON4.php');
        req.addEventListener('load', displayData);
        console.log(JSON.stringify(arr));
        req.send(JSON.stringify(arr)); // Send array as JSON string to the server
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main><p></p></main>
  </body>
</html>

```

[students.btsi.lu/evegi144/WAD/JS/JSON/JSON4.php](https://students.btsi.lu/evegi144/WAD/JS/JSON/JSON4.php)

```

<?php
  header("Content-Type: application/json; charset=UTF-8");
  // http://stackoverflow.com/questions/8599595/send-json-data-from-javascript-to-php
  $obj = json_decode(file_get_contents('php://input')); // Decode the JSON string.
  $obj[0]->name = 'OK'; // Modify it.
  echo json_encode($obj); // Send the object as a JSON string to the client browser.
?>

```

### 4.3.29. Application Programming Interfaces (API)

[html5-overview.net/current](https://html5-overview.net/current)

[platform.html5.org](https://platform.html5.org)

[html5demos.com](http://html5demos.com)

To find out which HTML5 APIs your browser supports, use [html5test.com](http://html5test.com).

If you want to develop your own JSON API, take a look at [jsonapi.org](http://jsonapi.org).

## File

The File API allows our app to represent and access file objects.

[www.w3.org/TR/FileAPI](http://www.w3.org/TR/FileAPI)

[www.html5rocks.com/en/tutorials/file/dndfiles](http://www.html5rocks.com/en/tutorials/file/dndfiles)

[www.codeproject.com/Articles/668351/HTML-File-API-Capability-and-Compatibility](http://www.codeproject.com/Articles/668351/HTML-File-API-Capability-and-Compatibility)

[students.btsi.lu/evegi144/WAD/JS/file1.html](http://students.btsi.lu/evegi144/WAD/JS/file1.html)

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>File API demo 1</title>
    <script>
      'use strict';

      const init = () => {
        document.querySelector('input').addEventListener('change', handleFile);
      };

      const handleFile = e => {
        const file = e.target.files[0];
        const reader = new FileReader();
        const display = e => document.querySelector('section').innerHTML =
e.target.result;
        reader.addEventListener('load', display);
        reader.readAsBinaryString(file);
        document.querySelector('span').innerHTML = `${file.size} bytes`;
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <header>
      <input type=file>
      File size: <span></span>
    </header>
    <section></section>
  </body>
</html>

```

## Drag and drop

To make any object drag and droppable it needs to have its **position** attribute set to **absolute** or **fixed**. We can then use Peter-Paul Koch's **dragDrop** object (cf. [www.quirksmode.org/js/dragdrop.html](http://www.quirksmode.org/js/dragdrop.html)), which does not use the drag and drop API but implements a solution based on the classic **mousemove** and **mouseup** events. It works very well across browsers. A stripped-down version is used in the following example:

[students.btsi.lu/evegi144/WAD/JS/draganddrop1.html](http://students.btsi.lu/evegi144/WAD/JS/draganddrop1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Drag and drop example 1</title>
    <meta charset=UTF-8>

```

```

<style>
  .draggable {
    position:      fixed;
    cursor:        move;
    background-color: lightgreen;
  }
</style>
<script> // Based on http://www.quirksmode.org/js/dragdrop.html
'use strict';

const dragDrop = {
  initialMouseX: undefined,
  initialMouseY: undefined,
  startX: undefined,
  startY: undefined,
  draggedObject: undefined,
  initElement: element => {
    if (typeof element == 'string') element = document.getElementById(element);
    element.onmousedown = dragDrop.startDragMouse;
    element.className += ' draggable';
  },
  startDragMouse: e => {
    dragDrop.startDrag(e.target);
    var evt = e || window.event;
    dragDrop.initialMouseX = evt.clientX;
    dragDrop.initialMouseY = evt.clientY;
    document.addEventListener('mousemove', dragDrop.dragMouse);
    document.addEventListener('mouseup', dragDrop.releaseElement);
    return false;
  },
  startDrag: obj => {
    if (dragDrop.draggedObject) dragDrop.releaseElement();
    dragDrop.startX = obj.offsetLeft;
    dragDrop.startY = obj.offsetTop;
    dragDrop.draggedObject = obj;
    obj.className += ' dragged';
  },
  dragMouse: e => {
    var evt = e || window.event;
    var dX = evt.clientX - dragDrop.initialMouseX;
    var dY = evt.clientY - dragDrop.initialMouseY;
    dragDrop.setPosition(dX, dY);
    return false;
  },
  setPosition: (dx, dy) => {
    dragDrop.draggedObject.style.left = `${dragDrop.startX + dx}px`;
    dragDrop.draggedObject.style.top = `${dragDrop.startY + dy}px`;
    console.log(`${dx} ${dy}`);
  },
  releaseElement: () => {
    document.removeEventListener('mousemove', dragDrop.dragMouse);
  }
};

```

```

        document.removeEventListener('mouseup', dragDrop.releaseElement);
        dragDrop.draggedObject.className =
            dragDrop.draggedObject.className.replace(/dragged/, '');
        dragDrop.draggedObject = null;
    }
};

const init = () => {
    dragDrop.initElement('art1');
    dragDrop.initElement('art2');
}

addEventListener('load', init);
</script>
</head>
<body>
    <main>
        <article id=art1>
            This is a draggable article
        </article>
        <article id=art2>
            This is another draggable article
        </article>
    </main>
</body>
</html>

```

The following two examples show a Window class to create a draggable and resizable window:



[students.btsi.lu/evegi144/WAD/JS/window1.html](https://students.btsi.lu/evegi144/WAD/JS/window1.html)

```

"use strict";

class Window {
    constructor(id, width, height, top, left, bgColor, resizable) {
        this.elem = document.createElement('aside');
        this.elem.style.position = 'absolute';
        this.elem.style.width = width >= 0 ? width + 'px' : '100px';
        this.elem.style.height = height >= 0 ? height + 'px' : '100px';
        this.elem.style.minWidth = '30px';
    }
}

```

```

this.elem.style.minHeight = '30px';
this.elem.style.border = '2px solid black';
this.elem.style.resize = 'both';
this.elem.style.overflow = 'auto';
this.elem.style.cursor = 'move';
this.elem.id = id;
this.dragging = false;
// cf. http://stackoverflow.com/questions/18942402/unable-to-remove-an-bound-
event-listener
this.mousemoveListener = undefined;
this.initialMouseX = undefined;
this.initialMouseY = undefined;
this.initialElemX = undefined;
this.initialElemY = undefined;
this.startDrag = e => {
  if (this.dragging) return;
  this.dragging = true;
  this.initialMouseX = e.clientX;
  this.initialMouseY = e.clientY;
  this.initialElemX = this.elem.offsetLeft;
  this.initialElemY = this.elem.offsetTop;
  this.elem.addEventListener('mousemove', this.mousemoveListener);
};
this.drag = e => {
  this.elem.style.left = (this.initialElemX + e.clientX - this.initialMouseX) +
'px';
  this.elem.style.top = (this.initialElemY + e.clientY - this.initialMouseY) +
'px';
};
this.stopDrag = e => {
  this.dragging = false;
  this.elem.removeEventListener('mousemove', this.mousemoveListener);
};
const btn = document.createElement('button');
btn.innerHTML = 'X';
btn.addEventListener('click', e => {
  document.body.removeChild(e.target.parentElement)
});
btn.style.position = 'absolute';
btn.style.right = '0';
btn.style.top = '0';
this.elem.appendChild(btn);
this.elem.addEventListener('mousedown', this.startDrag.bind(this));
this.elem.addEventListener('mouseup', this.stopDrag.bind(this));
document.body.appendChild(this.elem);
this.mousemoveListener = this.drag;
}
}

addEventListener('load', () => new Window('w1', 300, 200, 10, 10, 'green', true));

```



[students.btsi.lu/evegi144/WAD/JS/window2.html](https://students.btsi.lu/evegi144/WAD/JS/window2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Window class demo 2</title>
    <meta charset=utf-8>
    <style>
      #w1, #w2, #w3 {
        position:      absolute;
        overflow:      auto;
        width:         300px;
        height:        200px;
        min-width:     30px;
        min-height:    30px;
        background-color: green;
        resize:        both;
        border:        2px solid black;
      }
    </style>
    <script src=window2.js></script>
  </head>
  <body></body>
</html>
```

[students.btsi.lu/evegi144/WAD/JS/window2.js](https://students.btsi.lu/evegi144/WAD/JS/window2.js)

```
"use strict";

class Window {
  constructor(id) {
    this.elem = document.createElement('aside');
    this.elem.id = id;
    this.dragging = false;
    // cf. http://stackoverflow.com/questions/18942402/unable-to-remove-an-bound-
    event-listener
    this.mousemoveListener = undefined;
```

```

this.mouseUpListener = undefined;
this.initialMouseX = undefined;
this.initialMouseY = undefined;
this.initialElemX = undefined;
this.initialElemY = undefined;
this.nav = undefined;
this.startDrag = e => {
  if (this.dragging) return;
  this.dragging = true;
  this.initialMouseX = e.clientX;
  this.initialMouseY = e.clientY;
  this.initialElemX = this.elem.offsetLeft;
  this.initialElemY = this.elem.offsetTop;
  addEventListener('mousemove', this.mousemoveListener);
  addEventListener('mouseup', this.mouseupListener);
};
this.drag = e => {
  let x = this.initialElemX + e.clientX - this.initialMouseX;
  let y = this.initialElemY + e.clientY - this.initialMouseY;
  const elWidth = this.elem.getBoundingClientRect().width;
  const elHeight = this.elem.getBoundingClientRect().height;
  if (x < 0) x = 0;
  else if ((x + elWidth) > innerWidth) x = innerWidth - elWidth;
  if (y < 0) y = 0;
  else if ((y + elHeight) > innerHeight) y = innerHeight - elHeight;
  this.elem.style.left = `${x}px`;
  this.elem.style.top = `${y}px`;
};
this.stopDrag = e => {
  this.dragging = false;
  removeEventListener('mousemove', this.mousemoveListener);
  removeEventListener('mouseup', this.mouseupListener);
};
const nav = document.createElement('nav');
nav.style.backgroundColor = 'lightblue';
nav.style.height = '25px';
nav.style.cursor = 'move';
this.nav = nav;
nav.addEventListener('mousedown', this.startDrag);
const btn = document.createElement('button');
btn.innerHTML = 'X';
btn.addEventListener('click', e =>
  document.body.removeChild(e.target.parentElement.parentElement));
btn.style.position = 'absolute';
btn.style.right = '0';
btn.style.top = '0';
nav.appendChild(btn);
this.elem.appendChild(nav);
document.body.appendChild(this.elem);
this.mousemoveListener = this.drag;
this.mouseupListener = this.stopDrag;

```

```

    }
  }

  const init = () => {
    const win1 = new Window('w1');
    const win2 = new Window('w2');
    const win3 = new Window('w3');
  };

  addEventListener('load', init);

```

If we want to use the HTML5 drag and drop API instead of this object, we need to set the **draggable** attribute to **true**. See [developer.mozilla.org/en-US/docs/DragDrop/Drag\\_and\\_Drop](https://developer.mozilla.org/en-US/docs/DragDrop/Drag_and_Drop) and [developers.whatwg.org/dnd.html#dnd](https://developers.whatwg.org/dnd.html#dnd) for an in-depth explanation of drag and drop.

## Web Workers



This script will use 100% of the processing power of a 4 core CPU.

[students.btsi.lu/evegi144/WAD/JS/WebWorkers/demo1.html](https://students.btsi.lu/evegi144/WAD/JS/WebWorkers/demo1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Web Workers Demo 1</title>
    <meta charset=utf-8>
  </head>
  <body>
    <script>
      'use strict';

      // http://stackoverflow.com/questions/11871452/can-web-workers-utilize-100-of-a-
      // multi-core-cpu
      //window.URL = window.URL || window.webkitURL;
      const blob = new Blob(["while(true){}"], {type: 'text/javascript'});
      const code = window.URL.createObjectURL(blob);
      new Worker(code);
      new Worker(code);
      new Worker(code);
      new Worker(code);
    </script>
  </body>
</html>

```

## Server-Sent Events

This API allows the opening of an HTTP connection for receiving push notifications from a server in the form of DOM events. The specification can be found at [www.w3.org/TR/2009/WD-eventsource-20091029](https://www.w3.org/TR/2009/WD-eventsource-20091029). Good descriptions and examples can be found at:

[www.html5rocks.com/en/tutorials/eventsource/basics](http://www.html5rocks.com/en/tutorials/eventsource/basics)

[html5doctor.com/server-sent-events](http://html5doctor.com/server-sent-events)

[www.sitepoint.com/server-sent-events](http://www.sitepoint.com/server-sent-events)

[developer.mozilla.org/en-US/docs/Server-sent\\_events/Using\\_server-sent\\_events](https://developer.mozilla.org/en-US/docs/Server-sent_events/Using_server-sent_events)

[chimera.labs.oreilly.com/books/1230000000545/ch16.html](http://chimera.labs.oreilly.com/books/1230000000545/ch16.html)

[html.spec.whatwg.org/#server-sent-events](http://html.spec.whatwg.org/#server-sent-events)

For a comparison to AJAX polling and WebSockets, take a look at [stackoverflow.com/questions/11077857/what-are-long-polling-websockets-server-sent-events-sse-and-comet](https://stackoverflow.com/questions/11077857/what-are-long-polling-websockets-server-sent-events-sse-and-comet).

Here is a simple example that sends the current server time to the client every second. Take a look at the networking tab of the browser console. The communication takes place without new HTTP requests, as the existing one is kept alive. This is more efficient than using AJAX polling on the client side, i.e. each client checking with the server every second to see whether any new data has arrived.



When using sessions, we must call `session_write_close`, otherwise the session object will be locked and no other script can use it, given that our SSE-server runs an endless loop.

[students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/SSE\\_client.html](http://students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/SSE_client.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Server-Sent Events Demo</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const init = () => {
        const source = new EventSource('SSE_server.php');
        source.addEventListener('message', e =>
          document.querySelector('main').innerHTML = `${e.data}<br>`);
        source.addEventListener('open', e =>
          document.querySelector('main').innerHTML += 'Connection opened<br>');
        source.addEventListener('error', e =>
          document.querySelector('main').innerHTML += 'Error<br>');
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main>
    </main>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/SSE\\_server.php](https://students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/SSE_server.php)

```

<?php
// From http://stackoverflow.com/questions/9070995/
// html5-server-sent-events-prototyping-ambiguous-error-and-repeated-polling
header('Content-Type: text/event-stream');
header('Cache-Control: no-cache');

function sendMsg($id, $msg) {
    //echo "id: $id" . PHP_EOL;
    echo "data: $msg" . PHP_EOL;
    echo PHP_EOL;
    ob_flush();
    flush();
}

date_default_timezone_set('Europe/Luxembourg');
while (true) {
    $serverTime = time();
    $msg        = 'Server time: ' . date("h:i:s", time());
    sendMsg($serverTime, $msg);
    //session_write_close(); // Only needed if we work with sessions in this script.
    sleep(1);
}
?>

```

Nicolas Detombes has developed a chat app that nicely illustrates how SSE can be used:

[students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/chat\\_client.php](http://students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/chat_client.php)

```

<?php
// Created by Nicolas Detombes
require_once 'chat_database.php';
?>
<!DOCTYPE html>
<html lang='en'>
<head>
    <meta charset='UTF-8'>
    <title>Chat</title>
    <style>
        body {
            font-family: Arial, sans-serif;
        }

        textarea {
            width:      700px;
            height:     200px;
            border:     3px solid #cccccc;
            padding:    5px;
            font-family: Arial, sans-serif;
            resize:     none;
        }
    </style>

```

```

    }

    input {
        border:      3px solid #cccccc;
        padding:      5px;
        font-family: Arial, sans-serif;
    }
</style>
</head>
<body>
    <h2>Chat</h2>
    <textarea disabled></textarea>
    <form method='post'><br>
        <input name='comment' id='comment' required>
        <input type='submit' value='Send'>
    </form>
    <script>
        'use strict';

        function init() {
            var source = new EventSource('chat_load.php');
            source.addEventListener('message', function (e) {
                //get the 20 first chars of the last line in the textarea
                var textarea = document.querySelector('textarea');
                var content = textarea.value;
                var lastLine = content.substr(content.lastIndexOf("[ ID ]"), 20);
                //console.log(e.data.substr(0, 20));
                //console.log(lastLine);
                if (e.data.substr(0, 20) !== lastLine) { //if first 10 chars do not match
(id's)
                    textarea.value += e.data + '\n';
                    //auto-scroll down textarea
                    textarea.scrollTop = textarea.scrollHeight - textarea.clientHeight;
                }
            });

            document.forms[0].addEventListener('submit', function (e) {
                e.preventDefault();
                var data = new FormData(document.forms[0]), req = new XMLHttpRequest();
                req.addEventListener('load', displayData);
                req.open('POST', 'chat_comment.php');
                req.send(data);
                document.getElementById('comment').value = '';
            });
        }

        function displayData(e) {
            document.querySelector('textarea').innerHTML += e.target.response + '\n';
        }

        addEventListener('load', init);

```

```

</script>
<?php
    echo '<script>document.querySelector("textarea").value = "<script>";</script>';
    //load the last 5 posts
    $posts = Database::listPost();
    foreach ($posts as $post) {
        $data = '[ ID ] ' . $post[0] . ' [ TIME ] ' . $post[1] . ' [ CONTENT ] ' .
    $post[2] .
        '\n';
        echo '<script>document.querySelector("textarea").value += "' . $data .
    '"</script>';
    }
    ?>
</body>
</html>

```

[students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/chat\\_comment.php](https://students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/chat_comment.php)

```

<?php
    require_once 'chat_database.php';

    if (isset($_POST['comment'])) {
        //shouldn't be needed since comment input has required set to yes
        Database::post($_POST['comment']);
    }
    ?>

```

[students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/chat\\_database.php](https://students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/chat_database.php)

```

<?php

    require_once 'chat_credentials.php';

    class Database {

        private static $DB_HOST;
        private static $DB_USER;
        private static $DB_PASSWORD;
        private static $DB_NAME;
        private static $DB_TABLE_POST = 'tblPost';

        static function set_credentials($db_host, $db_user, $db_password, $db_name) {
            self::$DB_HOST      = $db_host;
            self::$DB_USER      = $db_user;
            self::$DB_PASSWORD  = $db_password;
            self::$DB_NAME      = $db_name;
        }

        static function connect() {

```

```

        $dbc = @mysqli_connect(self::$DB_HOST, self::$DB_USER,
            self::$DB_PASSWORD, self::$DB_NAME) or
            die('Connect Error (' . mysqli_connect_errno() . ') ' .
mysqli_connect_error());
        mysqli_set_charset($dbc, "utf8");
        return $dbc;
    }

    static function post($dtPost) {
        $dbc = self::connect();
        $query = 'INSERT INTO ' . self::$DB_TABLE_POST . ' (dtPost, fiUser, fiGroup)
VALUES (?,
    1, 1)';
        $stmt = $dbc->prepare($query);
        if (!$stmt)
            trigger_error('Wrong SQL: ' . $query . ' Error: ' . $dbc->error,
E_USER_ERROR);
        $stmt->bind_param('s', $dtPost);
        $stmt->execute();
        $stmt->close();
        $dbc->close();
    }

    static function listPost() {
        //select last 5 records WHERE [id] > (SELECT MAX([id]) - 5 FROM [MyTable]) NOT
ORDER BY
        // idPost ASC limit 5
        $idPost = '';
        $tTimestamp = '';
        $Post = '';
        $dbc = self::connect();
        $query = 'SELECT idPost, dtTimestamp, dtPost FROM ' . self::$DB_TABLE_POST
        .
        ' WHERE idPost > (SELECT MAX(idPost) - 5 FROM ' . self::$DB_TABLE_POST . ')';
        $stmt = $dbc->prepare($query);
        $stmt->execute();
        $stmt->store_result();
        for ($i = 0; $i < $stmt->num_rows; $i++) {
            $stmt->bind_result($idPost, $tTimestamp, $Post);
            $stmt->fetch();
            $result[] = array($idPost, $tTimestamp, $Post);
        }
        $stmt->close();
        $dbc->close();
        return $result;
    }

    static function lastPost() {
        $idPost = '';
        $tTimestamp = '';
        $Post = '';
    }

```

```

        $dbc      = self::connect();
        $query     = 'SELECT idPost, dtTimestamp, dtPost FROM ' . self::$DB_TABLE_POST
        .
        ' WHERE idPost = (SELECT MAX(idPost) FROM ' . self::$DB_TABLE_POST . ')';
        $stmt      = $dbc->prepare($query);
        $stmt->execute();
        $stmt->store_result();
        $stmt->bind_result($idPost, $tTimestamp, $Post);
        $stmt->fetch();
        $result = array($idPost, $tTimestamp, $Post);
        $stmt->close();
        $dbc->close();
        return $result;
    }
}
?>

```

[students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/chat\\_load.php](http://students.btsi.lu/evegi144/WAD/JS/ServerSentEvents/chat_load.php)

```

<?php
    require_once 'chat_database.php';

    header('Content-Type: text/event-stream');
    header('Cache-Control: no-cache');

    function sync($data) {
        echo 'data: [ ID ] ' . $data[0] . ' [ TIME ] ' . $data[1] . ' [ CONTENT ] ' .
        $data[2] .
        PHP_EOL;
        echo PHP_EOL;
        ob_flush();
        flush();
    }

    while (true) {
        sync(Database::lastPost());
        sleep(1);
    }
?>

```

## Canvas

Study the excellent documentation at [developer.mozilla.org/en-US/docs/Web/API/Canvas\\_API/Tutorial](https://developer.mozilla.org/en-US/docs/Web/API/Canvas_API/Tutorial) and [www.w3schools.com/html/html5\\_canvas.asp](http://www.w3schools.com/html/html5_canvas.asp). A good reference can be found at [www.w3schools.com/tags/ref\\_canvas.asp](http://www.w3schools.com/tags/ref_canvas.asp).

Here is a pacman trying to catch the mouse cursor:

[students.btsi.lu/evegi144/WAD/JS/Canvas/pacman.html](http://students.btsi.lu/evegi144/WAD/JS/Canvas/pacman.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Pacman</title>
    <meta charset=UTF-8>
    <style>
      body {
        background-color: black;
        overflow:         hidden;
        cursor:            url("mouseWithCheese64x64.cur"), auto;
      }
    </style>
  </head>
  <body>
    <script>
      'use strict';

      const canvas = document.createElement("canvas");

      const sizeCanvas = () => {
        canvas.width = window.innerWidth;
        canvas.height = window.innerHeight;
      };
      sizeCanvas();
      document.querySelector("body").appendChild(canvas);
      const context = canvas.getContext('2d');
      const pacman = new Image();
      pacman.src = "pacman128x128.png";
      pacman.alt = "pacman128x128.png";

      const TO_RADIANS = Math.PI / 180, TIMEOUT = 20;
      let mouseX = 0, mouseY = 0, currX = 100, currY = 100, timerID = null;

      const drawRotatedImage = (image, x, y, angle) => {
        // Save the current co-ordinate system before we play with it.
        context.save();

        // Move to the middle of where we want to draw our image.
        context.translate(x, y);

        // Rotate around that point, converting our angle from degrees to radians.
        context.rotate(angle * TO_RADIANS);

        // Draw it up and to the left by half the width and height of the image.
        context.drawImage(image, -(image.width / 2), -(image.height / 2));

        // Restore the co-ords to what they were when we began.
        context.restore();
      };
    </script>
  </body>
</html>

```

```

const getMouseXY = e => {
  mouseX = e.pageX;
  mouseY = e.pageY;
};

const animate = () => {
  const dX = currX - mouseX, dY = currY - mouseY;
  const a = Math.floor(dX / 20), b = Math.floor(dY / 20);
  currX -= a;
  currY -= b;
  if ((Math.abs(a) >= 1) || (Math.abs(b) >= 1)) {
    context.clearRect(0, 0, canvas.width, canvas.height);
    drawRotatedImage(pacman, currX, currY, (Math.atan2(dY, dX) - Math.PI) * 180
/ Math.PI);
  }
};

const toggle = () => {
  if (timerID === null) timerID = setInterval("animate()", TIMEOUT);
  else {
    clearInterval(timerID);
    timerID = null;
  }
};

addEventListener('mousemove', getMouseXY);
addEventListener('click', toggle);
addEventListener('resize', sizeCanvas);
timerID = setInterval(animate, TIMEOUT);
</script>
</body>
</html>

```

Here is a skeleton for a pong game:

[students.btsi.lu/evegi144/WAD/JS/Canvas/pong.html](https://students.btsi.lu/evegi144/WAD/JS/Canvas/pong.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Pong</title>
    <meta charset=utf-8>
    <script src=pong.js></script>
  </head>
  <body>
    <canvas width=1000 height=780></canvas>
    <button>Stop</button>
  </body>
</html>

```

```
"use strict";

const BAT_WIDTH = 10, BAT_HEIGHT = 100;
let timer, posLX, posLY, posRX, posRY, posBallX, posBally, ballDX, ballDY, posbatLX,
posbatLY,
    posbatRX, posbatRY, radius, canvas, ctx, running = true;

const init = () => {
    canvas = document.querySelector('canvas');
    ctx = canvas.getContext("2d");
    posLX = 2;
    posLY = 340;
    posRX = 988;
    posRY = 340;
    posBallX = 500;
    posBally = 360;
    ballDX = 3;
    ballDY = 3;
    radius = 20;
    posbatLX = 2;
    posbatLY = 340;
    posbatRX = 988;
    posbatRY = 340;
    document.querySelector('button').addEventListener('click', toggleAnimation);
    requestAnimationFrame(gameLoop);
};

const gameLoop = () => {
    ctx.fillStyle = "#000000";
    ctx.fillRect(0, 0, canvas.width, canvas.height);
    ctx.fillStyle = "#FFFFFF";
    ctx.fillRect(posLX, posLY, 10, 100);
    ctx.fillRect(posRX, posRY, 10, 100);
    ctx.beginPath();
    ctx.arc(posBallX, posBally, radius, 0, 2 * Math.PI);
    ctx.fill();
    // Check for collision with canvas borders.
    if (((posBallX + radius) >= canvas.width) || ((posBallX - radius) <= 0)) ballDX =
    -ballDX;
    if (((posBally + radius) >= canvas.height) || ((posBally - radius) <= 0)) ballDY =
    -ballDY;
    ctx.fillRect(posbatLX, posbatLY, BAT_WIDTH, BAT_HEIGHT);
    ctx.fillRect(posbatRX, posbatRY, BAT_WIDTH, BAT_HEIGHT);
    if ((posBallX - radius) <= (posbatLX + BAT_WIDTH) &&
        (BAT_HEIGHT + posbatLY >= (posBally - radius)) && (posbatLY <= (posBally +
        radius)))
        ballDX = Math.abs(ballDX);
    if ((posBallX + radius) >= posbatRX && (BAT_HEIGHT + posbatRY >= (posBally -
```

```

radius))
    && (posBallY <= (posBallY + radius))) ballDX = -Math.abs(ballDX);
    posBallX += ballDX;
    posBallY += ballDY;
    if (running) requestAnimationFrame(gameLoop);
};

const toggleAnimation = () => {
    running = !running;
    if (running) {
        requestAnimationFrame(gameLoop);
        document.querySelector('button').innerHTML = 'Stop';
    }
    else document.querySelector('button').innerHTML = 'Resume';
};

addEventListener('load', init);

```

And here is a random maze generator class:

[students.btsi.lu/evegi144/WAD/JS/Canvas/maze1.html](https://students.btsi.lu/evegi144/WAD/JS/Canvas/maze1.html)

```

<!DOCTYPE html>
<html lang=en>
<head>
    <title>Canvas 2D maze demo</title>
    <meta charset=utf-8>
    <script>
        'use strict';

        /**
         * Construct a maze cell.
         * @param x {number} horizontal position in the maze (0 <= x < maze.width)
         * @param y {number} vertical position in the maze (0 <= y < maze.height)
         * @param topWall {boolean} does the cell have a top wall?
         * @param rightWall {boolean} does the cell have a right wall?
         * @param bottomWall {boolean} does the cell have a bottom wall?
         * @param leftWall {boolean} does the cell have a left wall?
         * @constructor
         */
        class Cell {
            constructor(x, y, topWall, rightWall, bottomWall, leftWall) {
                this.x = x;
                this.y = y;
                this.topWall = topWall;
                this.rightWall = rightWall;
                this.bottomWall = bottomWall;
                this.leftWall = leftWall;
                this.visited = false;
            }

```

```

}

/**
 * Generate a maze using dept-first search with backtracking.
 * cf.
http://en.wikipedia.org/wiki/Maze\_generation\_algorithm#Recursive\_backtracker
 * @param width {number} number of cells in a row
 * @param height {number} number of cells in a column
 * @param cellSize {number} side length of a cell in pixels
 * @param wallThickness {number} wall thickness in pixels
 * @constructor
 */
class Maze {
  constructor(width, height, cellSize, wallThickness) {
    this.width = width;
    this.height = height;
    this.cellSize = cellSize;
    this.wallThickness = wallThickness;
    this.numCells = width * height;
    this.cells = []; // Arranged in x, y order, i.e. first column, then row
    index.

    this.getUnvisitedNeighbors = (x, y) => {
      const unvisitedNeighbors = [];
      if (x >= 1 && !this.cells[x - 1][y].visited)
        unvisitedNeighbors.push(this.cells[x - 1][y]); // left neighbour
      if (x < (this.width - 1) && !this.cells[x + 1][y].visited)
        unvisitedNeighbors.push(this.cells[x + 1][y]); // right
      if (y >= 1 && !this.cells[x][y - 1].visited)
        unvisitedNeighbors.push(this.cells[x][y - 1]); // top
      if (y < (this.height - 1) && !this.cells[x][y + 1].visited)
        unvisitedNeighbors.push(this.cells[x][y + 1]); // bottom
      return unvisitedNeighbors;
    };

    this.draw2D = () => {
      const canvas = document.querySelector('canvas');
      const ctx = canvas.getContext('2d');
      ctx.fillStyle = 'black';
      ctx.fillRect(0, 0, canvas.width, canvas.height);
      ctx.fillStyle = 'red';
      let cell;
      const cs = this.cellSize, wt = this.wallThickness;
      for (let i = 0; i < this.width; i++)
        for (let j = 0; j < this.height; j++) {
          cell = this.cells[i][j];
          if (cell.topWall) ctx.fillRect(cell.x * cs - wt / 2, cell.y * cs - wt
/ 2, cs +
          wt, wt);
          if (cell.rightWall) ctx.fillRect(cell.x * cs + cs - wt / 2, cell.y *
cs - wt /

```

```

        2, wt, cs + wt);
        if (cell.bottomWall) ctx.fillRect(cell.x * cs - wt / 2, cell.y * cs +
cs - wt
        / 2, cs + wt, wt);
        if (cell.leftWall) ctx.fillRect(cell.x * cs - wt / 2, cell.y * cs - wt
/ 2, wt,
        cs + wt);
    }
    // Draw maze border.
    ctx.lineWidth = 2 * wt;
    ctx.strokeStyle = 'red';
    ctx.strokeRect(0, 0, canvas.width, canvas.height);
    // Draw exit.
    ctx.strokeStyle = 'green';
    ctx.beginPath();
    ctx.moveTo(0, 0);
    ctx.lineTo(cs, 0);
    ctx.stroke();
    // Draw entry.
    ctx.lineWidth = wt;
    ctx.strokeStyle = 'pink';
    ctx.beginPath();
    ctx.moveTo(this.width * cs, (this.height - 1) * cs);
    ctx.lineTo(this.width * cs, this.height * cs);
    ctx.stroke();
};

// Initialize maze with all walls present.
for (let i = 0; i < width; i++) {
    this.cells[i] = [];
    for (let j = 0; j < height; j++)
        this.cells[i][j] = new Cell(i, j, true, true, true, true);
}
// We start at a random place and mark it as visited.
let currentCell = new Cell(Math.floor(Math.random() * width),
    Math.floor(Math.random() * height), true, true, true, true);
currentCell.visited = true;
let numCellsVisited = 1, unvisitedNeighbors, chosenNeighbor;
const stack = [];
// While there are unvisited cells.
while (numCellsVisited <= this.numCells) {
    // If the current cell has any neighbors which have not been visited.
    unvisitedNeighbors = this.getUnvisitedNeighbors(currentCell.x,
currentCell.y);
    if (unvisitedNeighbors.length > 0) {
        // Choose randomly one of the unvisited neighbors.
        chosenNeighbor =
            unvisitedNeighbors[Math.floor(Math.random() *
unvisitedNeighbors.length)];
        // Push the current cell to the stack.
        stack.push(currentCell);
    }
}

```

```

        // Remove the wall between the current cell and the chosen cell.
        if (chosenNeighbor.x < currentCell.x) { // Neighbor is left from current
cell.
            currentCell.leftWall = false;
            chosenNeighbor.rightWall = false;
        }
        else if (chosenNeighbor.x > currentCell.x) {
            // Neighbor is right from current cell.
            currentCell.rightWall = false;
            chosenNeighbor.leftWall = false;
        }
        else if (chosenNeighbor.y < currentCell.y) {
            // Neighbor is above current cell.
            currentCell.topWall = false;
            chosenNeighbor.bottomWall = false;
        }
        else if (chosenNeighbor.y > currentCell.y) {
            // Neighbor is below current cell.
            currentCell.bottomWall = false;
            chosenNeighbor.topWall = false;
        }
        // Make the chosen cell the current cell and mark it as visited.
        currentCell = chosenNeighbor;
        currentCell.visited = true;
        numCellsVisited++;
    }
    // Else if stack is not empty.
    else if (stack.length > 0) {
        // Pop a cell from the stack and make it the current cell.
        currentCell = stack.pop();
    }
    else {
        // Pick a random unvisited cell, make it the current cell and mark it as
visited.
        const unvisitedCells = [];
        for (let i = 0; i < this.width; i++)
            for (let j = 0; j < this.height; j++)
                if (!this.cells[i][j].visited)
unvisitedCells.push(this.cells[i][j]);
        currentCell = unvisitedCells[Math.floor(Math.random() *
unvisitedCells.length)];
        currentCell.visited = true;
        numCellsVisited++;
    }
}
this.cells[0][0].topWall = false; // Open up exit.
this.cells[this.width - 1][this.height - 1].rightWall = false; // Open up
entry.
this.draw2D();
}
}

```

```
        addEventListener('load', () => new Maze(20, 20, 40, 3));
    </script>
</head>
<body>
    <main>
        <canvas width=1000 height=1000></canvas>
    </main>
</body>
</html>
```

## Web Sockets

So far we've used HTTP POST or GET requests to send data to the server, who responded with new HTML. This is a very inefficient and limited approach. If we want real-time two way communication between the server and a potentially large number of clients, we should take advantage of the new [JavaScript WebSocket API](#) that is available in Firefox and Chrome. By establishing a bidirectional communication channel between the server and each client, we can for instance implement real-time chat.

The easiest way to get started can be found at [websocketd.com](https://websocketd.com). The recommended way is to use [WebSockets with Node.js](#).

### Client

On the client side, we need to implement something like the following:

```

const socket = new WebSocket("wss://foxi.ltam.lu:35000");
socket.addEventListener('open', opened);
socket.addEventListener('message', received);
socket.addEventListener('close', closed);
socket.addEventListener('error', error);

const opened = () => {

}

const received = event => {
  console.log("Received: ", event.data);
}

const closed = () => {

}

const error = event => {
  console.error("Error: ", event.data);
}

const send = event => {
  socket.send("Hello world!");
}

```

**opened** will be called once the web socket connection is ready. **received** will be called when a message from the server has been received.

### Server

On the server side, we can use [WebSockets with PHP](#) but [WebSockets with Node.js](#) is highly recommended.

### WebGL

From [developer.mozilla.org/en-US/docs/Web/WebGL](https://developer.mozilla.org/en-US/docs/Web/WebGL):

WebGL (Web Graphics Library) is a JavaScript API for rendering interactive 3D graphics and 2D graphics within any compatible web browser without the use of plug-ins.

Relevant web pages:

[www.khronos.org/registry/webgl/specs/1.0](http://www.khronos.org/registry/webgl/specs/1.0)

[www.khronos.org/webgl/wiki/Main\\_Page](http://www.khronos.org/webgl/wiki/Main_Page)

[developer.cdn.mozilla.net/media/uploads/demos/a/z/azakai/3baf4ad7e600cbda06ec46efec5ec3b8/bananabread\\_1373485124\\_demo\\_package/index.html](https://developer.cdn.mozilla.net/media/uploads/demos/a/z/azakai/3baf4ad7e600cbda06ec46efec5ec3b8/bananabread_1373485124_demo_package/index.html)

[www.spacejack.ca/spacejack](http://www.spacejack.ca/spacejack)

[hexgl.bkcore.com/play](http://hexgl.bkcore.com/play)

[htmlchess.sourceforge.net/demo/example.html](http://htmlchess.sourceforge.net/demo/example.html)

[www.chromeexperiments.com/webgl](http://www.chromeexperiments.com/webgl)

[playwebgl.com](http://playwebgl.com)

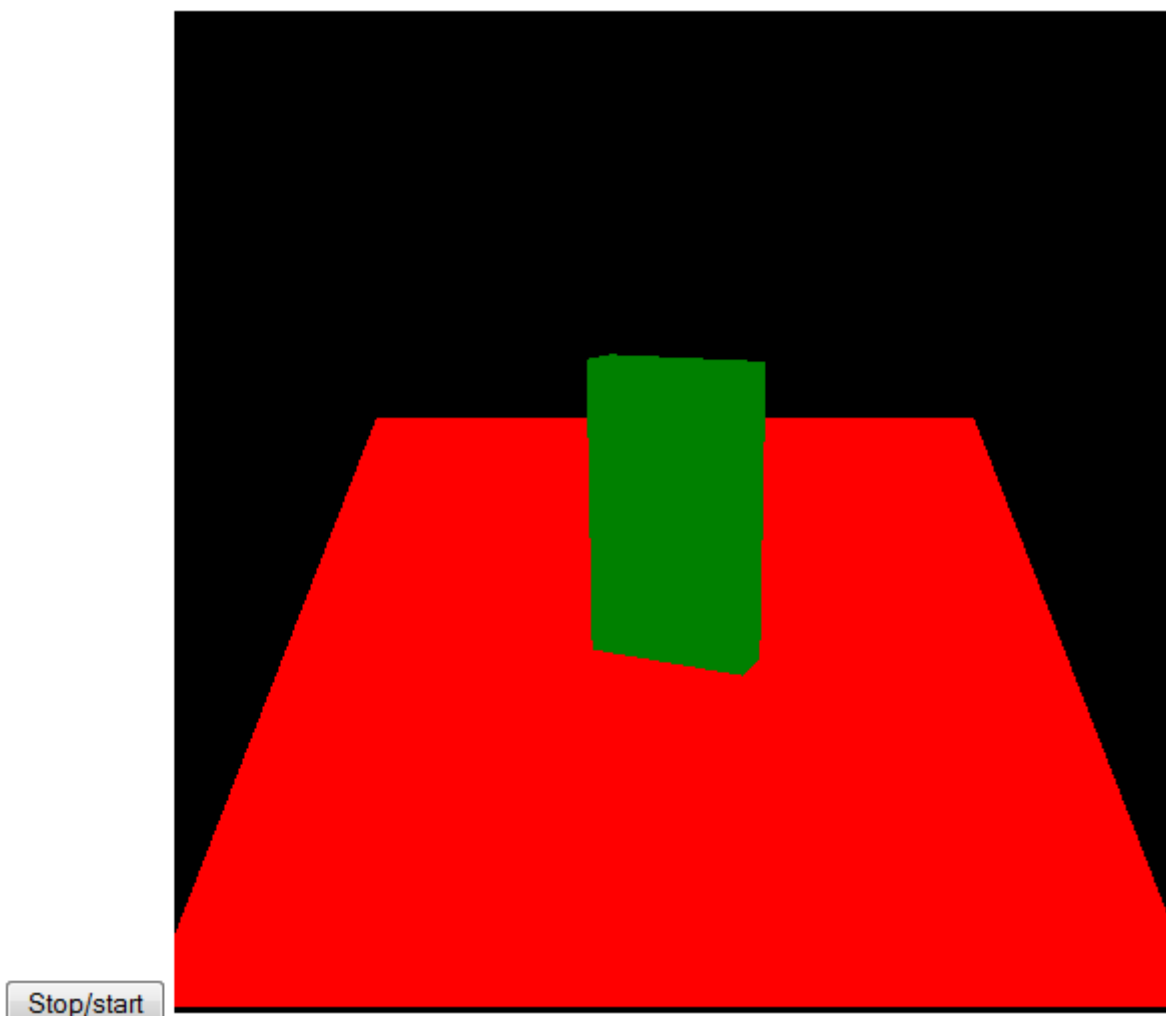
WebGL 2 will be even better, cf. [www.youtube.com/watch?v=2v6iLpY7j5M](http://www.youtube.com/watch?v=2v6iLpY7j5M) and [webglsamples.org/WebGL2Samples/#texture\\_3d](http://webglsamples.org/WebGL2Samples/#texture_3d). The spec can be found at [www.khronos.org/registry/webgl/specs/latest/2.0](http://www.khronos.org/registry/webgl/specs/latest/2.0).

### Three.js

Given the complexity of direct WebGL programming, we'll start by using the **Three.js** JavaScript library, which can be downloaded from [github.com/mrdoob/three.js](https://github.com/mrdoob/three.js) and greatly simplifies the development of 3D web apps.

The best way to get started is to study the documentation at [threejs.org/docs/#manual/introduction/Creating-a-scene](http://threejs.org/docs/#manual/introduction/Creating-a-scene).

Let's start with a very simple example and walk it through step by step:



[students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/test1.html](http://students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/test1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>My first Three.js app</title>
    <meta charset=utf-8>
    <script src=three.js></script>
  </head>
  <body>
    <script>
      "use strict"; // This will be standard in the future and makes debugging easier.
      let scene, camera, renderer, cube, plane, spotlight, animation = true;

      window.addEventListener('load',
        // This function will be executed after the whole document is loaded.
        () => {
          const canvas = document.querySelector('canvas');
          scene = new THREE.Scene();
          camera = new THREE.PerspectiveCamera(50, canvas.width / canvas.height, 0.1,
1000);
          renderer = new THREE.WebGLRenderer();
          renderer.setSize(canvas.width, canvas.height);
          document.body.appendChild(renderer.domElement);
          let geometry = new THREE.BoxGeometry(1, 1.8, 0.3);
          let material = new THREE.MeshBasicMaterial({color: 'green'});
          cube = new THREE.Mesh(geometry, material);
          cube.castShadow = true;
          scene.add(cube);
          geometry = new THREE.PlaneGeometry(5, 5);
          material = new THREE.MeshBasicMaterial({color: 'red'});
          const plane = new THREE.Mesh(geometry, material);
          plane.receiveShadow = true;
          plane.rotation.x = -0.5 * Math.PI;
          plane.position.y = -1;
          scene.add(plane);
          camera.position.y = 4;
          camera.position.z = 5;
          camera.lookAt(scene.position);
          spotlight = new THREE.AmbientLight('blue');
          scene.add(spotlight);
          render();
        });

      // This is our animation loop that will ideally be executed 60 times per second.
      const render = () => {
        if (animation) requestAnimationFrame(render);
        cube.rotation.x += 0.1;
        cube.rotation.y += 0.1;
        renderer.render(scene, camera);
      };
    </script>
  </body>
</html>

```

```
// Stop/restart the animation.  
const toggleAnimation = () => {  
  animation = !animation;  
  if (animation) render();  
};  
</script>  
<canvas width=500 height=500></canvas>  
<button onclick=toggleAnimation();>Stop/start</button>  
</body>  
</html>
```

As explained in the Three.js introductory example, we need a scene, a camera and a renderer to display the scene using the camera. The renderer needs the canvas element, which is where the whole scene will be displayed.

Our script consists of 3 functions:

1. An initialization function that creates the scene, the camera and the renderer. It then adds them to the DOM and starts the rendering. This function is executed only once, after the document has been loaded.
2. The rendering function, which calls itself using `requestAnimationFrame` (cf. [developer.mozilla.org/en-US/docs/Web/API/window.requestAnimationFrame](https://developer.mozilla.org/en-US/docs/Web/API/window.requestAnimationFrame)). Here we perform the animation, but only if the animation is supposed to be running as indicated by the boolean global variable. In this simple example we rotate our cube around the x and y axes.
3. `toggleAnimation` simply toggles a boolean global variable, which indicates whether the animation is currently meant to be running or stopped. In the former case, the render function gets executed.

Here's a slightly more evolved example:



[students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/test2.html](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/test2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>My second Three.js app</title>
    <meta charset=utf-8>
    <style>
      body {
        margin: 0;
        padding: 0;
        overflow: hidden;
      }

      button {
        position: fixed;
        top: 0;
        left: 0;
      }
    </style>
  </head>
  <body>
    <button>Click me!</button>
  </body>
</html>
```

```

#stats {
  position: fixed;
  top: 20px;
  left: 0;
}
</style>
<script src=three.js></script>
<script src=stats.min.js></script>
</head>
<body>
  <script>
    "use strict"; // This will be standard in the future and makes debugging easier.

    let scene, camera, renderer, geometry, material, mesh, animation = true, stats,
        stepX = 0.1, stepY = 0.1, stepZ = 0.1, directionX = 1, directionY = 1,
directionZ = 1;

    const init = () => {
      stats = new Stats();
      document.getElementById('stats').appendChild(stats.domElement);
      scene = new THREE.Scene();
      camera = new THREE.PerspectiveCamera(75, window.innerWidth /
        window.innerHeight, 1, 10000);
      camera.position.x = 0;
      camera.position.y = 70;
      camera.position.z = 30;
      camera.lookAt(scene.position);

      // Add a texture to our sphere.
      const mapUrl = "LTAM256x256.png";
      const map = THREE.ImageUtils.loadTexture(mapUrl);
      geometry = new THREE.SphereGeometry(20, 50, 50);
      material = new THREE.MeshLambertMaterial({color: 0xff00ff, map: map});
      mesh = new THREE.Mesh(geometry, material);
      mesh.position.y = 20;
      mesh.castShadow = true;
      scene.add(mesh);

      const spotLight = new THREE.SpotLight(0xffffffff);
      spotLight.position.set(-40, 100, 50);
      spotLight.castShadow = true;
      scene.add(spotLight);
      renderer = new THREE.WebGLRenderer();
      renderer.setSize(window.innerWidth, window.innerHeight);
      renderer.setClearColor(0x000000);
      renderer.shadowMapEnabled = true;

      // When the window gets resized, we need to remove the old canvas
      // and add a new one with the correct size.
      const childNodes = document.body.childNodes;

```

```

    let n = childNodes.length;
    while (n > 0) {
        if (childNodes[n - 1].tagName && childNodes[n - 1].tagName === 'CANVAS') {
            document.body.removeChild(childNodes[n - 1]);
        }
        n--;
    }
    document.body.appendChild(renderer.domElement);
    render();
};

// This is our animation loop that will ideally be executed 60 times per second.
const render = () => {
    if (animation) requestAnimationFrame(render);
    mesh.rotation.x += 0.01;
    mesh.rotation.y += 0.02;
    mesh.rotation.z += 0.01;
    if (mesh.position.x > 100) directionX = -1;
    else if (mesh.position.x < -100) directionX = 1;
    if (mesh.position.y > 50) directionY = -1;
    else if (mesh.position.y < -50) directionY = 1;
    if (mesh.position.z > 50) directionZ = -1;
    else if (mesh.position.z < -50) directionZ = 1;
    mesh.position.x += stepX * Math.random() * directionX;
    mesh.position.y += stepY * Math.random() * directionY;
    mesh.position.z += stepZ * Math.random() * directionZ;
    stats.update();
    renderer.render(scene, camera);
};

const resize = () => {
    camera.aspect = window.innerWidth / window.innerHeight;
    camera.updateProjectionMatrix();
    renderer.setSize(window.innerWidth, window.innerHeight);
};

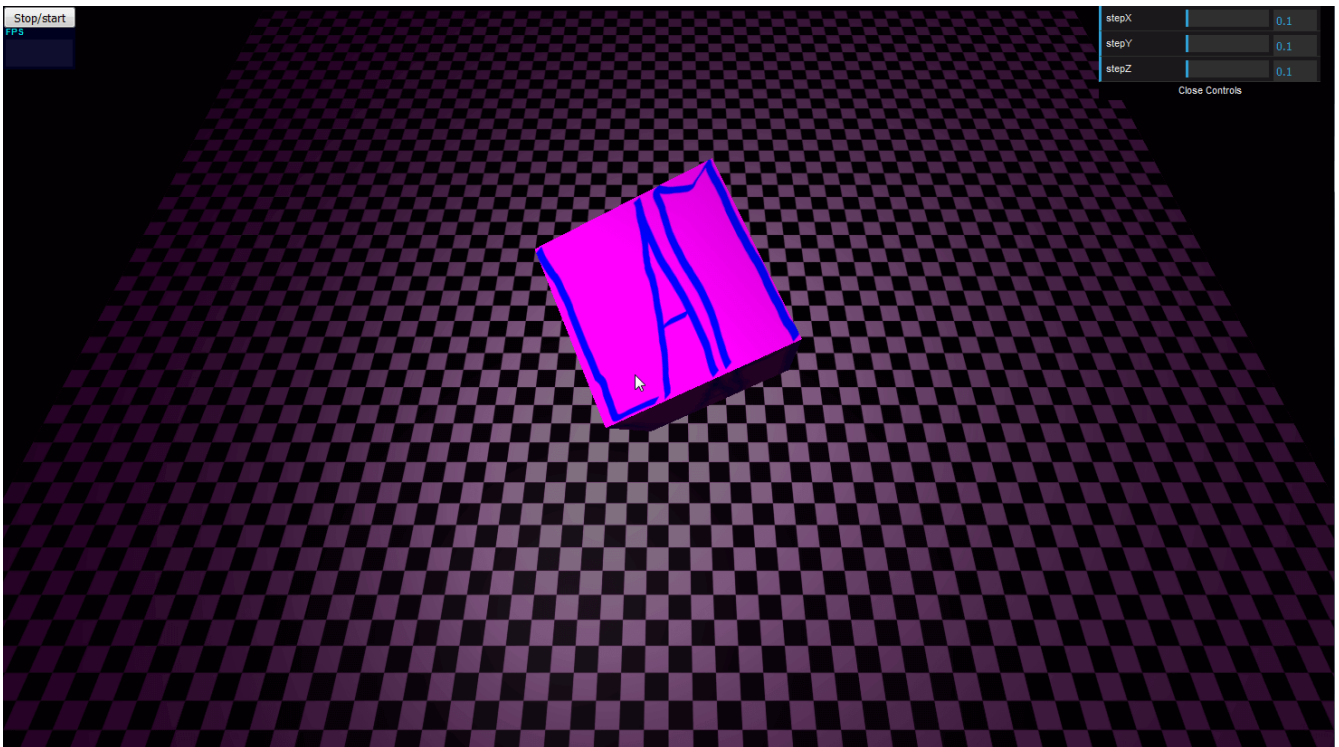
// Stop/restart the animation.
const toggleAnimation = () => {
    animation = !animation;
    if (animation) render();
};

window.addEventListener('load', init);
window.addEventListener('resize', resize);
</script>
<div id=stats></div>
<button onclick=toggleAnimation();>Stop/start</button>
</body>
</html>

```

Instead of declaring a canvas element, we'll let the renderer take care of that. We'll let the scene take up the whole browser window width and height. We can even switch to full screen mode using **F11**. Finally we use Mr. Doob's performance monitor, available from [github.com/mrdoob/stats.js](https://github.com/mrdoob/stats.js).

As a further evolutionary step, we can add user controls (cf. [code.google.com/p/dat-gui/](https://code.google.com/p/dat-gui/)), mouse interaction and a funny background plane:



[students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/3js1.html](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/3js1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Three.js Skeleton</title>
    <meta charset=utf-8>
    <style>
      body {
        margin: 0;
        padding: 0;
        overflow: hidden;
      }

      button {
        position: fixed;
        top: 0;
        left: 0;
      }

      #stats {
        position: fixed;
        top: 20px;
        left: 0;
      }
    </style>
  </head>
  <body>
    <button>Stop/start
    </button>
    <div>
      <div>FPS</div>
      <div>
        <div>stepX</div>
        <div>stepY</div>
        <div>stepZ</div>
        <div>Close Controls</div>
      </div>
    </div>
  </body>
</html>
```

```

}
</style>
<script src=three.js></script>
<script src=OrbitControls.js></script>
<script src=stats.min.js></script>
<script src=dat.gui.min.js></script>
<script>
  "use strict";

  let scene, camera, renderer, geometry, material, mesh, animation = true, stats,
    projector = new THREE.Projector(),
    directionX = 1, directionY = 1, directionZ = 1, controls, cameraControl;

  const init = () => {
    stats = new Stats();
    document.getElementById('stats').appendChild(stats.domElement);
    controls = {
      stepX: 0.1,
      stepY: 0.1,
      stepZ: 0.1
    };

    const gui = new dat.GUI();
    gui.add(controls, 'stepX', 0, 3);
    gui.add(controls, 'stepY', 0, 3);
    gui.add(controls, 'stepZ', 0, 3);
    scene = new THREE.Scene();

    camera = new THREE.PerspectiveCamera(75, window.innerWidth /
      window.innerHeight, 1, 10000);
    camera.position.x = 0;
    camera.position.y = 70;
    camera.position.z = 30;
    camera.lookAt(scene.position);
    cameraControl = new THREE.OrbitControls(camera);
    let mapUrl = "LAM256x256.png";
    let map = THREE.ImageUtils.loadTexture(mapUrl);
    geometry = new THREE.BoxGeometry(20, 20, 20);
    material = new THREE.MeshLambertMaterial({color: 0xff00ff, map: map});
    mesh = new THREE.Mesh(geometry, material);
    mesh.position.y = 20;
    mesh.castShadow = true;
    scene.add(mesh);

    mapUrl = "checker_large.gif";
    map = THREE.ImageUtils.loadTexture(mapUrl);
    map.wrapS = map.wrapT = THREE.RepeatWrapping;
    map.repeat.set(8, 8);
    const color = 0xffffff;
    const ambient = 0x888888;
    // Put in a ground plane to show off the lighting

```

```

geometry = new THREE.PlaneGeometry(200, 200, 50, 50);
const chessboard = new THREE.Mesh(geometry,
  new THREE.MeshPhongMaterial({
    color: color,
    ambient: ambient, map: map, side: THREE.DoubleSide,
    opacity: 0.5, transparent: true
  }));
chessboard.rotation.x = -Math.PI / 2;
chessboard.position.y = -4.02;
scene.add(chessboard);

const spotLight = new THREE.SpotLight(0xffffff);
spotLight.position.set(-40, 100, 50);
spotLight.castShadow = true;
scene.add(spotLight);
const hemisphereLight = new THREE.HemisphereLight(0xdd00dd, 0x00aa00, 0.3);
scene.add(hemisphereLight);
renderer = new THREE.WebGLRenderer();
renderer.setSize(window.innerWidth, window.innerHeight);
renderer.setClearColor(0x000000);
renderer.shadowMapEnabled = true;
/*const childNodes = document.body.childNodes;
let n = childNodes.length;
while (n > 0) {
  if (childNodes[n - 1].tagName && childNodes[n - 1].tagName === 'CANVAS') {
    document.body.removeChild(childNodes[n - 1]);
  }
  n--;
}*/
document.body.appendChild(renderer.domElement);
animate();
};

const animate = () => {
  // note: three.js includes requestAnimationFrame shim
  if (animation) requestAnimationFrame(animate);
  stats.update();
  mesh.rotation.x += 0.01;
  mesh.rotation.y += 0.02;
  mesh.rotation.z += 0.01;
  if (mesh.position.x > 100) directionX = -1;
  else if (mesh.position.x < -100) directionX = 1;
  if (mesh.position.y > 50) directionY = -1;
  else if (mesh.position.y < -50) directionY = 1;
  if (mesh.position.z > 50) directionZ = -1;
  else if (mesh.position.z < -50) directionZ = 1;
  mesh.position.x += Math.random() * controls.stepX * directionX;
  mesh.position.y += Math.random() * controls.stepY * directionY;
  mesh.position.z += Math.random() * controls.stepZ * directionZ;
  cameraControl.update();
  renderer.render(scene, camera);
};

```

```

};

const resize = () => {
  camera.aspect = window.innerWidth / window.innerHeight;
  camera.updateProjectionMatrix();
  renderer.setSize(window.innerWidth, window.innerHeight);
};

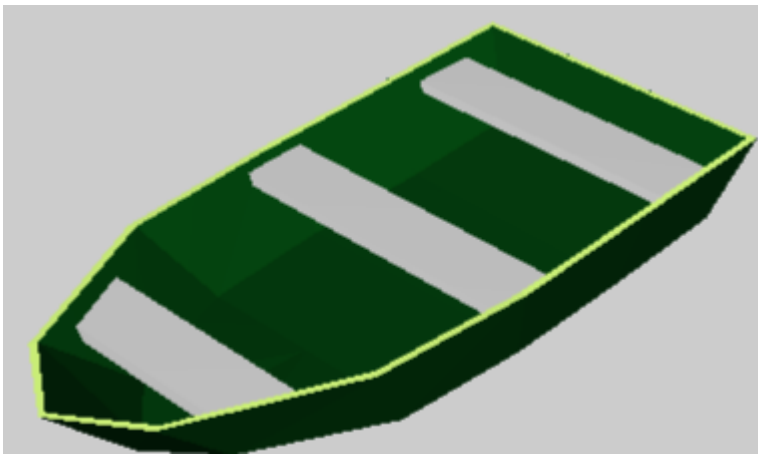
const toggleAnimation = () => {
  animation = !animation;
  if (animation) animate();
};

const onDocumentMouseDown = event => {
  event.preventDefault();
  const vector = new THREE.Vector3((event.clientX / window.innerWidth) * 2 - 1,
    -(event.clientY / window.innerHeight) * 2 + 1, 0.5);
  projector.unprojectVector(vector, camera);
  const raycaster = new THREE.Raycaster(camera.position,
    vector.sub(camera.position).normalize());
  const intersects = raycaster.intersectObjects([mesh]);
  if (intersects.length > 0) {
    const r = Math.floor(Math.random() * 256);
    const g = Math.floor(Math.random() * 256);
    const b = Math.floor(Math.random() * 256);
    intersects[0].object.material.color =
      new THREE.Color("rgb(" + r + ", " + g + ", " + b + ")");
  }
};

window.addEventListener('load', init);
window.addEventListener('resize', resize);
document.addEventListener('mousedown', onDocumentMouseDown);
</script>
</head>
<body>
  <div id=stats></div>
  <button onclick=toggleAnimation();>Stop/start</button>
</body>
</html>

```

## Loading a Collada model



[students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/load\\_collada.html](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/load_collada.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Load Collada model with Three.js</title>
    <meta charset=utf-8>
    <script src=three.js></script>
    <script src=ColladaLoader.js></script>
    <script>
      'use strict';

      const init = () => {
        const scene = new THREE.Scene();
        const camera = new THREE.PerspectiveCamera(45,
          window.innerWidth / window.innerHeight, 0.1, 1000);
        const webGLRenderer = new THREE.WebGLRenderer();
        webGLRenderer.setClearColorHex(0xcccccc, 1.0);
        webGLRenderer.setSize(window.innerWidth, window.innerHeight);
        webGLRenderer.shadowMapEnabled = true;

        // position and point the camera to the center of the scene
        camera.position.x = 15;
        camera.position.y = 15;
        camera.position.z = 15;
        camera.lookAt(new THREE.Vector3(0, 2, 0));

        // add spotlight for the shadows
        const spotLight = new THREE.SpotLight(0xffffff);
        spotLight.position.set(150, 150, 150);
        spotLight.intensity = 2;
        scene.add(spotLight);

        document.querySelector('main').appendChild(webGLRenderer.domElement);

        const loader = new THREE.ColladaLoader();
        loader.options.convertUpAxis = true;
        let mesh;
```

```

    loader.load("Boat2.dae", result => {
      console.dir(result);
      mesh = result.scene;
      mesh.position.set(0, -1, 0);
      //mesh.scale.set(1, 1, 1);
      scene.add(mesh);
    });

    const render = () => {
      requestAnimationFrame(render);
      webGLRenderer.render(scene, camera);
    };

    render();
  };

  addEventListener('load', init);
</script>
</head>
<body>
  <main></main>
</body>
</html>

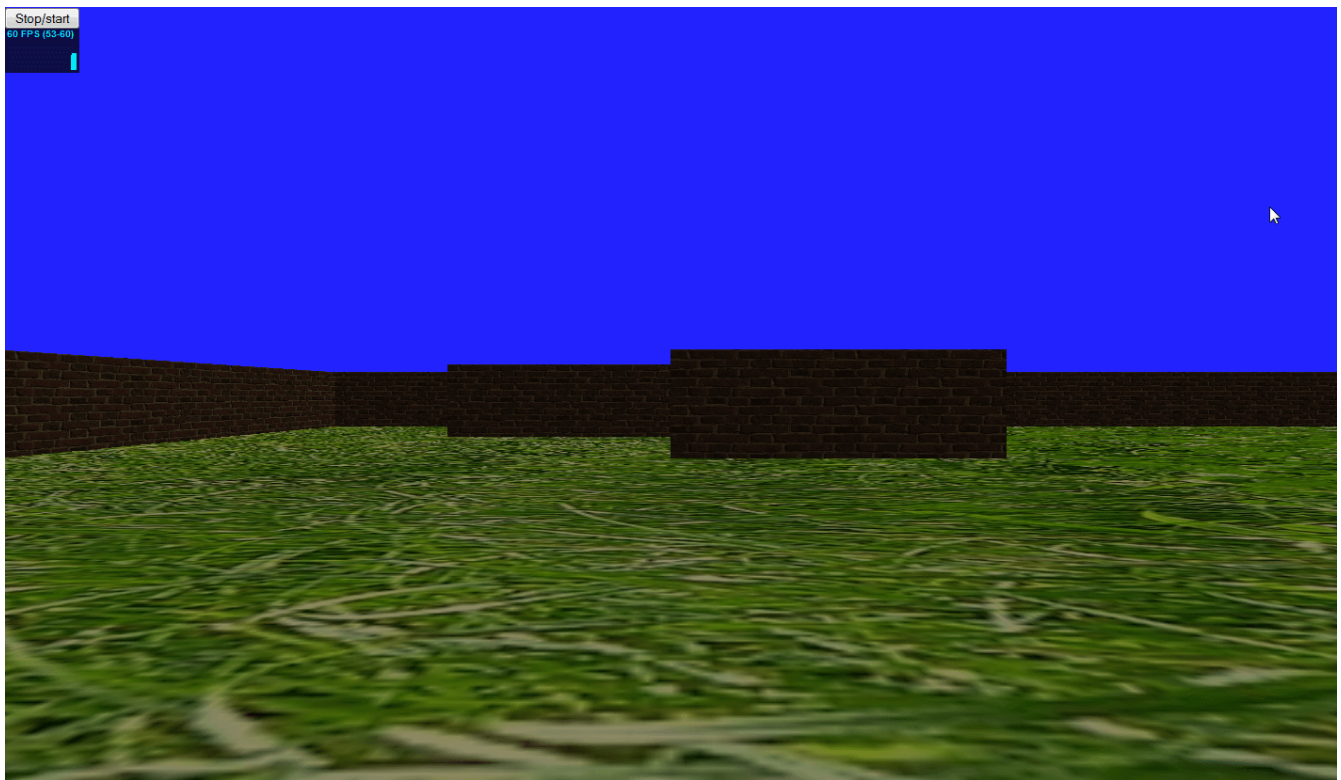
```

## Physijs

[github.com/chandlerprall/Physijs](https://github.com/chandlerprall/Physijs)

## FPS

Let's have some real fun and create a first person shooter (FPS) from scratch.



[students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/FPS/index.html](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/FPS/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>FPS</title>
    <meta charset=utf-8>
    <link rel=stylesheet href=style.css>
    <script src=three.js></script>
    <script src=FirstPersonControls.js></script>
    <script src=stats.min.js></script>
    <script src=dat.gui.min.js></script>
    <script src=main.js></script>
  </head>
  <body>
    <aside id=stats></aside>
    <button onclick="game.toggleAnimation()">Stop/start</button>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/FPS/main.js](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Three.js/FPS/main.js)

```
"use strict";

const game = {};
game.aspectRatio = window.innerWidth / window.innerHeight;
game.scene = new THREE.Scene();
game.camera = new THREE.PerspectiveCamera(60, game.aspectRatio, 1, 10000);
game.cameraControls = new THREE.FirstPersonControls(game.camera);
```

```

game.renderer = new THREE.WebGLRenderer();
game.projector = new THREE.Projector();
game.stats = new Stats();
game.animation = true;
game.clock = new THREE.Clock();
game.map = [ // 1  2  3  4  5  6  7  8  9
  [1, 1, 1, 1, 1, 1, 1, 1, 1], // 0
  [1, 0, 0, 0, 0, 0, 0, 1, 0], // 1
  [1, 1, 0, 0, 2, 0, 0, 0, 1], // 2
  [1, 0, 0, 0, 0, 2, 0, 0, 1], // 3
  [1, 0, 0, 2, 0, 0, 2, 0, 1], // 4
  [1, 0, 0, 0, 2, 0, 0, 0, 1], // 5
  [1, 0, 1, 0, 0, 0, 0, 1, 1], // 6
  [1, 0, 1, 0, 0, 1, 0, 0, 1], // 7
  [1, 0, 1, 0, 1, 0, 0, 0, 1], // 8
  [1, 1, 1, 1, 1, 1, 1, 1, 1] // 9
];
game.mapW = game.map.length;
game.mapH = game.map[0].length;
game.UNITSIZE = 50;
game.WALLHEIGHT = game.UNITSIZE / 3;
game.MOVESPEED = 100;
game.LOOKSPEED = 0.075;
game.init = () => {
  document.getElementById('stats').appendChild(game.stats.domElement);
  game.cameraControls.movementSpeed = game.MOVESPEED;
  game.cameraControls.lookSpeed = game.LOOKSPEED;
  game.cameraControls.lookVertical = false;
  game.cameraControls.noFly = true;
  game.camera.position.y = game.UNITSIZE * .2;
  game.scene.add(game.camera);
  const geometry = new THREE.BoxGeometry(game.UNITSIZE * game.mapW, 1,
    game.UNITSIZE * game.mapW);
  const material = new THREE.MeshLambertMaterial(
    {map: THREE.ImageUtils.loadTexture('images/texture-91829_1920.jpg')}});
  const floor = new THREE.Mesh(geometry, material);
  game.scene.add(floor);
  const cube = new THREE.BoxGeometry(game.UNITSIZE, game.WALLHEIGHT, game.UNITSIZE);
  const materials = [
    new THREE.MeshLambertMaterial({map: THREE.ImageUtils.loadTexture('images/wall-1.jpg')}),
    new THREE.MeshLambertMaterial({map: THREE.ImageUtils.loadTexture('images/wall-2.jpg')}),
    new THREE.MeshLambertMaterial({color: 0xFBEBCE})
  ];
  for (let i = 0; i < game.mapW; i++) {
    for (let j = 0, m = game.map[i].length; j < m; j++) {
      if (game.map[i][j]) {
        const wall = new THREE.Mesh(cube, materials[game.map[i][j] - 1]);
        wall.position.x = (.5 + i - game.mapW / 2) * game.UNITSIZE;
        wall.position.y = game.WALLHEIGHT / 2;
      }
    }
  }
};

```

```

        wall.position.z = (.5 + j - game.mapW / 2) * game.UNITSIZE;
        game.scene.add(wall);
    }
}

const directionalLight1 = new THREE.DirectionalLight(0xF7EFBE, 0.7);
directionalLight1.position.set(.5, 1, .5);
game.scene.add(directionalLight1);
const directionalLight2 = new THREE.DirectionalLight(0xF7EFBE, 0.5);
directionalLight2.position.set(-0.5, -1, -0.5);
game.scene.add(directionalLight2);
game.scene.fog = new THREE.FogExp2(0xa6a1aF, 0.0005);
game.renderer.setSize(window.innerWidth, window.innerHeight);
game.renderer.setClearColor(0x2222ff);
game.renderer.shadowMapEnabled = true;
document.body.appendChild(game.renderer.domElement);
game.animate();
};

game.animate = () => {
    if (game.animation) requestAnimationFrame(game.animate);
    game.stats.update();
    game.cameraControls.update(game.clock.getDelta());
    game.renderer.render(game.scene, game.camera);
};

game.toggleAnimation = () => {
    game.animation = !game.animation;
    if (game.animation) game.animate();
};

game.resize = () => {
    game.camera.aspect = window.innerWidth / window.innerHeight;
    game.camera.updateProjectionMatrix();
    game.renderer.setSize(window.innerWidth, window.innerHeight);
};

window.addEventListener('load', game.init);
window.addEventListener('resize', game.resize);

```

[carvisualizer.plus360degrees.com/threejs](http://carvisualizer.plus360degrees.com/threejs)

[hexgl.bkcore.com](http://hexgl.bkcore.com)

[portableapps.com/apps/graphics\\_pictures/blender\\_portable](http://portableapps.com/apps/graphics_pictures/blender_portable)

[github.com/tparisi/Vizi](https://github.com/tparisi/Vizi)

[www.peter-strohm.de/webgl/index.php](http://www.peter-strohm.de/webgl/index.php)

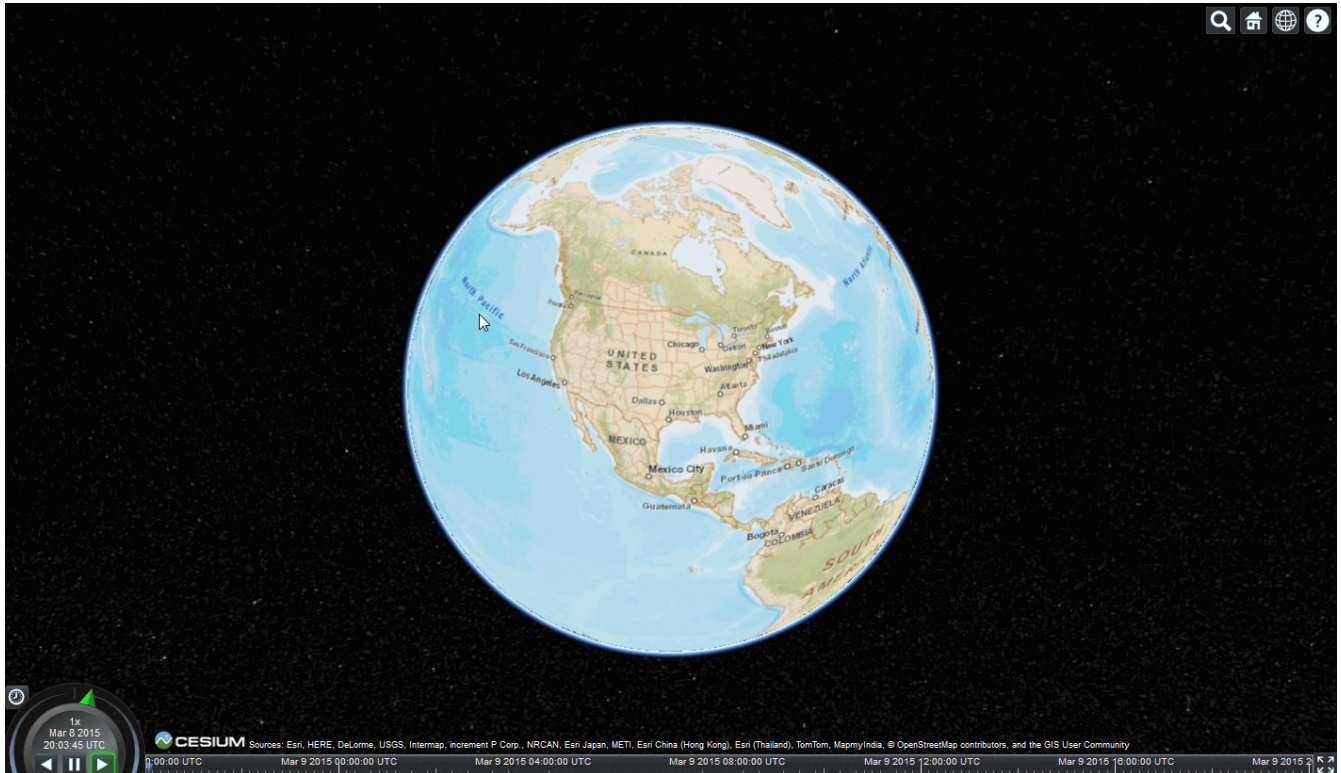
[www.khronos.org/webgl/wiki/User\\_Contributions](http://www.khronos.org/webgl/wiki/User_Contributions)

## GLAM

GLAM (GL And Markup) is a declarative language for 3D web content (cf. [tparisi.github.io/glam](http://tparisi.github.io/glam)).

## Cesium

This is a JavaScript library for creating 3D globes and 2D maps (cf. [cesiumjs.org](http://cesiumjs.org)).



[students.btsi.lu/evegi144/WAD/JS/WebGL/Cesium/cesium1.html](http://students.btsi.lu/evegi144/WAD/JS/WebGL/Cesium/cesium1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>A first Cesium experiment</title>
    <meta charset=utf-8>
    <style>
      @import url(Build/Cesium/Widgets/widgets.css);

      html, body, #cesiumContainer {
        width: 100%;
        height: 100%;
        margin: 0;
        padding: 0;
        overflow: hidden;
      }
    </style>
    <script src=Build/Cesium/Cesium.js></script>
  </head>
  <body>
    <div id=cesiumContainer></div>
    <script>
      'use strict';

      const viewer = new Cesium.Viewer('cesiumContainer', {
        imageryProvider: new Cesium.ArcGisMapServerImageryProvider({
          url:

'https://server.arcgisonline.com/ArcGIS/rest/services/World_Street_Map/MapServer'
        }),
        baseLayerPicker: false
      });
    </script>
  </body>
</html>

```

### Direct WebGL programming

All WebGL drawing happens inside the HTML canvas element (cf. [developer.mozilla.org/en-US/docs/Web/API/HTMLCanvasElement.getContext](https://developer.mozilla.org/en-US/docs/Web/API/HTMLCanvasElement.getContext)). Before using the WebGL API we need a `WebGLRenderingContext` object (cf. [developer.mozilla.org/en-US/docs/Web/API/WebGLRenderingContext](https://developer.mozilla.org/en-US/docs/Web/API/WebGLRenderingContext)), which manages the whole 3D drawing process. To get it we call the `getContext` method of the canvas element and pass `webgl` as parameter to get the 3D context. If we pass `2d` we get a 2D context. If we wanted to ensure compatibility with older browsers, we would have to check `experimental-webgl`, `webkit-3d` and `moz-webgl` as parameters to get a 3D context, but we will assume that the user uses an up to date browser.

WebGL methods correspond to OpenGL methods documented at [www.khronos.org/opengles/sdk/docs/man](http://www.khronos.org/opengles/sdk/docs/man).

Useful `WebGLRenderingContext` methods:

- `createShader(type)` creates an empty shader object of the given type (`VERTEX_SHADER` or `FRAGMENT_SHADER`) and returns its reference.
- `createProgram()` creates an empty program object and returns its reference.
- `clearColor(red, green, blue, alpha)` sets color for drawing area. Values from 0 to 1, alpha === 1 → opaque, alpha === 0 → fully transparent.
- `shaderSource(shader, source)` replaces the source code in a shader object.
- `compileShader(shader)` compiles the shader object.
- `attachShader(program, shader)` attaches a shader to a program.
- `linkProgram(program)` links a program.
- `useProgram(program)` installs the program as part of the current rendering state.
- `clearColor(red, green, blue, alpha)` specifies clear values for the color buffers.
- `clear(buffer)` clears the buffer(s) specified. It takes a single argument. If several buffers are to be cleared it is the bitwise OR of several values from `GL_COLOR_BUFFER_BIT`, `GL_DEPTH_BUFFER_BIT` and `GL_STENCIL_BUFFER_BIT`.
- `drawArrays(mode, first, count)` renders primitives from array data. The first parameter specifies what kind of primitives to render. Choices are `GL_POINTS`, `GL_LINE_STRIP`, `GL_LINE_LOOP`, `GL_LINES`, `GL_TRIANGLE_STRIP`, `GL_TRIANGLE_FAN` and `GL_TRIANGLES`.
- `getAttribLocation(program, name)` returns the location of an attribute variable.
- `vertexAttrib3f(index, v0, v1, c2)` specifies the value of a generic vertex attribute. Similar methods end with `1f`, `2f` or `4f`. From [www.khronos.org/opengles/sdk/docs/man](http://www.khronos.org/opengles/sdk/docs/man):

These commands can be used to specify one, two, three, or all four components of the generic vertex attribute specified by index. A 1 in the name of the command indicates that only one value is passed, and it will be used to modify the first component of the generic vertex attribute. The second and third components will be set to 0, and the fourth component will be set to 1. Similarly, a 2 in the name of the command indicates that values are provided for the first two components, the third component will be set to 0, and the fourth component will be set to 1. A 3 in the name of the command indicates that values are provided for the first three components and the fourth component will be set to 1, whereas a 4 in the name indicates that values are provided for all four components.

- `getUniformLocation(program, name)` returns the location of a uniform variable.
- `uniform[1, 2, 3, 4]f(index, v0 [, v1, v2, v3])` specifies the values of a uniform variable.
- `createBuffer()` creates a buffer object.
- `deleteBuffer(buffer)` deletes a buffer object.

- `bindBuffer(target, buffer)` binds a buffer object telling WebGL what type of data it contains. `target` must be `GL_ARRAY_BUFFER` or `GL_ELEMENT_ARRAY_BUFFER`.
- `bufferData(target, data, usage)` creates and initializes a buffer object's data store. `target` must be `GL_ARRAY_BUFFER` or `GL_ELEMENT_ARRAY_BUFFER`. `usage` is one of `GL_STREAM_DRAW`, `GL_STATIC_DRAW`, or `GL_DYNAMIC_DRAW`.
- `vertexAttribPointer(location, size, type, normalized, stride, offset)`

WebGL uses typed arrays for maximum performance:

Array	Bytes
Int8Array	1
Uint8Array	1
Int16Array	2
Uint16Array	2
Int32Array	4
Uint32Array	4
Float32Array	4
Float64Array	8

To create a typed array, we call the constructor.

Typed arrays have the following methods, properties and constants:

<code>get(index)</code>
<code>set(index, value)</code>
<code>set(array, offset)</code>
<code>length</code>
<code>BYTES_PER_ELEMENT</code>

Let's look at two simple examples:



[students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/webgl1.html](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/webgl1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WebGL Example 1</title>
    <meta charset=utf-8>
    <script>
      "use strict";

      let vertexShaderSource, fragmentShaderSource;

      const handleShader = (e, isFragment) => {
        if (isFragment) fragmentShaderSource = e.target.response;
        else vertexShaderSource = e.target.response;
        if (vertexShaderSource && fragmentShaderSource) runShaders();
      };

      const init = () => {
        const req1 = new XMLHttpRequest(), req2 = new XMLHttpRequest();
        req1.open('POST', 'vertex_shader1.js');
        req1.addEventListener('load', e => {
          handleShader(e, false);
        });
        req1.send();
```

```

    req2.open('GET', 'fragment_shader1.js');
    req2.addEventListener('load', e => {
        handleShader(e, true);
    });
    req2.send();
};

const runShaders = () => {
    const canvas = document.querySelector('canvas');
    let gl;
    try {
        gl = canvas.getContext('webgl');
    } catch (e) {
        alert('Your browser does not seem to support WebGL');
    }
    if (gl) {
        const vertexShader = gl.createShader(gl.VERTEX_SHADER);
        const fragmentShader = gl.createShader(gl.FRAGMENT_SHADER);
        const program = gl.createProgram();
        gl.shaderSource(vertexShader, vertexShaderSource);
        gl.shaderSource(fragmentShader, fragmentShaderSource);
        gl.compileShader(vertexShader);
        gl.compileShader(fragmentShader);
        gl.attachShader(program, vertexShader);
        gl.attachShader(program, fragmentShader);
        gl.linkProgram(program);
        gl.useProgram(program);
        //gl.program = program;
        gl.clearColor(0, 0, 0, 1);
        /*gl.enable(gl.DEPTH_TEST);
        gl.depthFunc(gl.LEQUAL);*/
        gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT);
        gl.drawArrays(gl.POINTS, 0, 1);
    }
};

    addEventListener('load', init);
</script>
</head>
<body>
    <main>
        <canvas width=640 height=480>No canvas support.</canvas>
    </main>
</body>
</html>

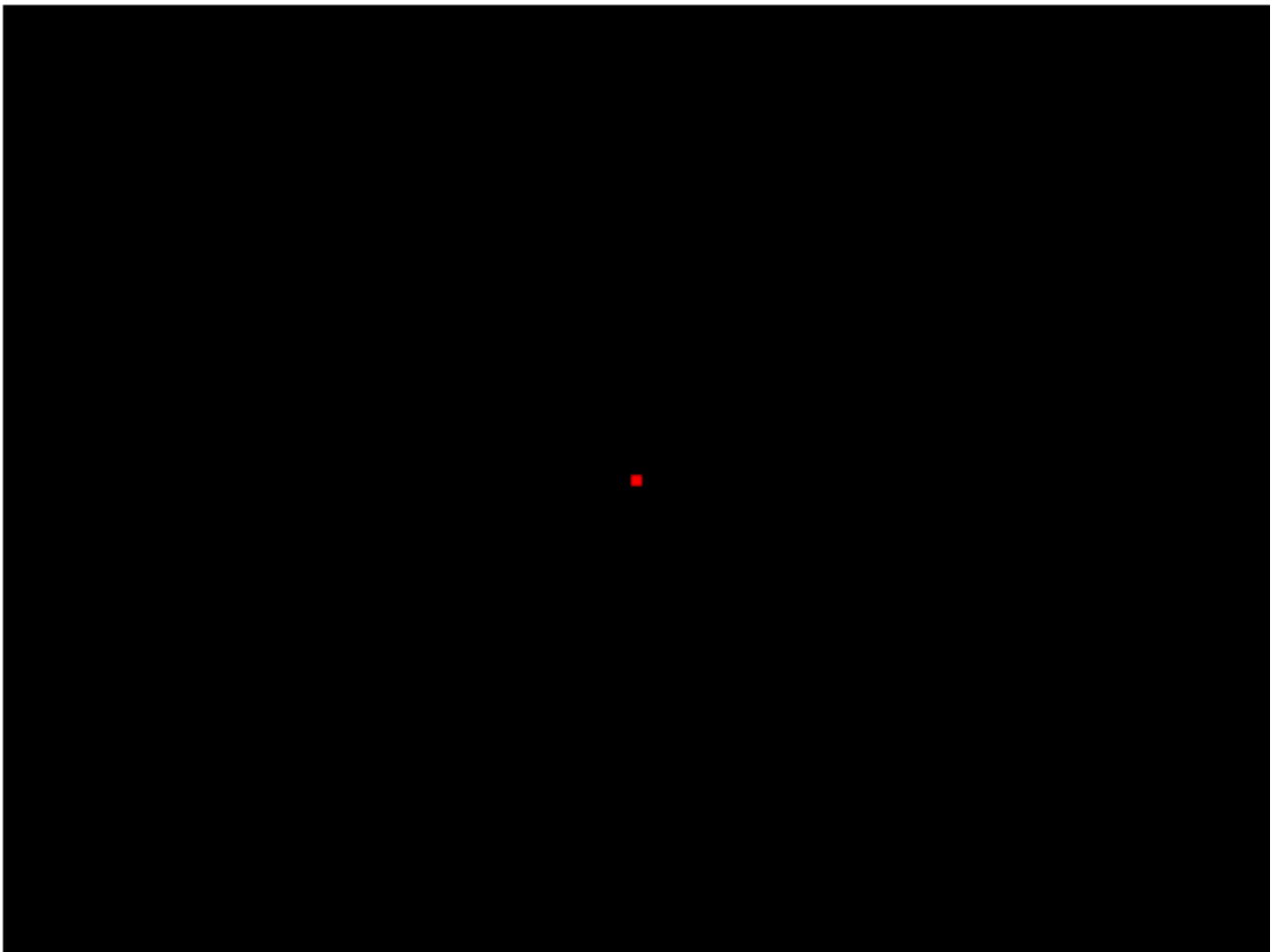
```

[students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/vertex\\_shader1.js](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/vertex_shader1.js)

```
void main() {  
    gl_Position = vec4(0.5, 0.0, 0.0, 1.0);  
    gl_PointSize = 10.0;  
}
```

[students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/fragment\\_shader1.js](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/fragment_shader1.js)

```
void main() {  
    gl_FragColor = vec4(1.0, 0.0, 0.0, 1.0);  
}
```



[students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/webgl2.html](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/webgl2.html)

```
<!DOCTYPE html>  
<html lang=en>  
  <head>  
    <title>WebGL Example 1</title>  
    <meta charset=utf-8>  
    <script>  
      "use strict";  
  
      const VSHADER_FILE = 'vertex_shader2.js', HSHADER_FILE = 'fragment_shader2.js';  
      let vertexShaderSource, fragmentShaderSource;
```

```

const handleShader = (e, isFragment) => {
  if (isFragment) fragmentShaderSource = e.target.response;
  else vertexShaderSource = e.target.response;
  if (vertexShaderSource && fragmentShaderSource) runShaders();
};

const init = () => {
  const req1 = new XMLHttpRequest(), req2 = new XMLHttpRequest();
  req1.open('POST', VSHADER_FILE);
  req1.addEventListener('load', e => {
    handleShader(e, false);
  });
  req1.send();
  req2.open('POST', HSHADER_FILE);
  req2.addEventListener('load', e => {
    handleShader(e, true);
  });
  req2.send();
};

const runShaders = () => {
  const canvas = document.querySelector('canvas');
  let gl;
  try {
    gl = canvas.getContext('webgl');
  } catch (e) {
    alert('Your browser does not seem to support WebGL');
  }
  if (gl) {
    const vertexShader = gl.createShader(gl.VERTEX_SHADER);
    const fragmentShader = gl.createShader(gl.FRAGMENT_SHADER);
    const program = gl.createProgram();
    gl.shaderSource(vertexShader, vertexShaderSource);
    gl.shaderSource(fragmentShader, fragmentShaderSource);
    gl.compileShader(vertexShader);
    gl.compileShader(fragmentShader);
    gl.attachShader(program, vertexShader);
    gl.attachShader(program, fragmentShader);
    gl.linkProgram(program);
    gl.useProgram(program);
    //gl.program = program;
    const a_Position = gl.getAttribLocation(program, 'a_Position');
    gl.vertexAttrib3f(a_Position, 0.0, 0.0, 0.0);
    const a_PointSize = gl.getAttribLocation(program, 'a_PointSize');
    gl.vertexAttrib1f(a_PointSize, 5.0);
    gl.clearColor(0, 0, 0, 1);
    /*gl.enable(gl.DEPTH_TEST);
    gl.depthFunc(gl.LEQUAL);*/
    gl.clear(gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT);
    gl.drawArrays(gl.POINTS, 0, 1);
  }
};

```

```

    }
};

    addEventListener('load', init);
</script>
</head>
<body>
    <main>
        <canvas width=640 height=480>No canvas support.</canvas>
    </main>
</body>
</html>

```

[students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/vertex\\_shader2.js](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/vertex_shader2.js)

```

attribute vec4 a_Position;
attribute float a_PointSize;

void main() {
    gl_Position = a_Position;
    gl_PointSize = a_PointSize;
}

```

[students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/fragment\\_shader2.js](https://students.btsi.lu/evegi144/WAD/JS/WebGL/Direct/fragment_shader2.js)

```

void main() {
    gl_FragColor = vec4(1.0, 0.0, 0.0, 1.0);
}

```

## Blender

According to [www.blender.org](http://www.blender.org):

Blender is a free and open source 3D animation suite. It supports the entirety of the 3D pipeline—modeling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing and game creation.

Blender is an extremely powerful application. Unfortunately, it's user interface is not necessarily the easiest one to master.

## Page Visibility

From [www.w3.org/TR/2011/WD-page-visibility-20110602](http://www.w3.org/TR/2011/WD-page-visibility-20110602):

The Page Visibility specification defines a means for site developers to programmatically determine the current visibility of a document and be notified of visibility changes.

This API is very useful for instance in the case of WMOTU Invaders, where we do not want the aliens to continue moving and shooting in the background whilst we are not playing the game! Take a look at [WMOTU Invaders object-oriented](#) to see how it's done. Further details and examples can be found at [developer.mozilla.org/en-US/docs/Web/API/Page\\_Visibility\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Page_Visibility_API).

## WebAudio

Two excellent Web resources are [www.html5rocks.com/en/tutorials/webaudio/intro](http://www.html5rocks.com/en/tutorials/webaudio/intro) and [webaudioapi.com](http://webaudioapi.com).

Here is a simple example:

[students.btsi.lu/evegi144/WAD/JS/webaudio1.html](http://students.btsi.lu/evegi144/WAD/JS/webaudio1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Web Audio Example</title>
    <meta charset=utf-8>
    <script src=webaudio1.js></script>
  </head>
  <body>
    <main>
      <input type=button value=Play>
    </main>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/JS/webaudio1.js](http://students.btsi.lu/evegi144/WAD/JS/webaudio1.js)

```

"use strict";

let context = null;

const init = () => {
  try {
    window.AudioContext = window.AudioContext || window.webkitAudioContext;
    context = new AudioContext();
  }
  catch (e) {
    window.alert('Web Audio API is not supported in this browser.');
```

```

  }

  let mybuffer = null;
  const button = document.querySelector('input');
  button.addEventListener('click', () => {
    play(mybuffer);
  });

  if (context) {
    const url = 'gunshot.wav';
    const request = new XMLHttpRequest();
    request.open("GET", url, true);
    request.responseType = "arraybuffer";
    request.addEventListener('load', () => {
      context.decodeAudioData(request.response, buffer => {
        mybuffer = buffer;
      });
      button.disabled = false;
    });
    request.send();
  }
};
```

```

const play = buffer => {
  var sourceNode = context.createBufferSource();
  sourceNode.buffer = buffer;
  sourceNode.connect(context.destination);
  sourceNode.start(0);
};
```

```

addEventListener("load", init);
```

## Observers

### MutationObserver

<https://developer.mozilla.org/en-US/docs/Web/API/MutationObserver?redirectlocale=en-US&redirectslug=DOM%2FMutationObserver>

`Object.observe`

[www.html5rocks.com/en/tutorials/es7/observe](http://www.html5rocks.com/en/tutorials/es7/observe)

## Web Storage

Web Storage provides a larger, more secure, and easier-to-use alternative to storing information in cookies. The official specification is at [www.w3.org/TR/webstorage](http://www.w3.org/TR/webstorage) and good introductions with examples can be found at [www.sitepoint.com/an-overview-of-the-web-storage-api](http://www.sitepoint.com/an-overview-of-the-web-storage-api), <http://html5doctor.com/storing-data-the-simple-html5-way-and-a-few-tricks-you-might-not-have-known> and [developer.mozilla.org/en-US/docs/Web/Guide/API/DOM/Storage](http://developer.mozilla.org/en-US/docs/Web/Guide/API/DOM/Storage).

For storage limit info see [developer.mozilla.org/en-US/docs/Web/API/IndexedDB\\_API/Browser\\_storage\\_limits\\_and\\_eviction\\_criteria](http://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API/Browser_storage_limits_and_eviction_criteria).

## IndexedDB

[javascript.info/indexeddb](http://javascript.info/indexeddb)

[medium.com/@AndyHaskell2013/build-a-basic-web-app-with-indexeddb-8ab4f83f8bda](https://medium.com/@AndyHaskell2013/build-a-basic-web-app-with-indexeddb-8ab4f83f8bda)

[developer.mozilla.org/en-US/docs/Web/API/IndexedDB\\_API](http://developer.mozilla.org/en-US/docs/Web/API/IndexedDB_API)

[github.com/jakearchibald/idb](https://github.com/jakearchibald/idb)

[flaviocopes.com/indexeddb](http://flaviocopes.com/indexeddb)

[medium.com/dev-channel/offline-storage-for-progressive-web-apps-70d52695513c](https://medium.com/dev-channel/offline-storage-for-progressive-web-apps-70d52695513c)

[students.btsi.lu/evegi144/WAD/JS/indexeddb1.html](http://students.btsi.lu/evegi144/WAD/JS/indexeddb1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>IndexedDB example 1</title>
    <!--<script src=indexeddb1.js></script>-->
    <script type=module>
      'use strict';

      import {openDB, deleteDB, wrap, unwrap} from 'https://unpkg.com/idb?module';

      const init = async () => {
        // https://github.com/jakearchibald/idb

        if (!indexedDB) {
          console.warn('IndexedDB not supported');
          return;
        }

        const dbName = 'TestDB';
        const storeName = 'store1';
        const version = 1; //versions start at 1
```

```

const db = await openDB(dbName, version, {
  upgrade(db, oldVersion, newVersion, transaction) {
    console.log('upgrade called');
    const store = db.createObjectStore(storeName);
  },
  blocked() {
    console.log('blocked');
  },
  blocking() {
    console.log('blocking');
  }
});

const handleFile = async e => {
  const files = e.target.files;
  console.dir(files);
  const isValidType = type => {
    const validTypes = ['image/jpeg', 'image/jpg', 'image/png', 'image/gif',
      'image/bmp'];
    for (const validType of validTypes) if (type === validType) return true;
    return false;
  };

  const readAndDisplay = async () => {
    let tx = db.transaction(storeName, 'readwrite');
    let store = await tx.objectStore(storeName);
    const pics = await store.getAll();
    await tx.done;
    console.dir(pics);
    for (const pic of pics) {
      let image = new Image();
      let buffer = Uint8Array.from(pic);
      let blob = new Blob([buffer]);
      let img = document.createElement('img');
      img.src = URL.createObjectURL(blob);
      document.querySelector('section').appendChild(img);
    }
  };

  let filesLeft = files.length;
  for (const file of files) {
    if (isValidType(file.type)) {
      console.log('Valid type: ' + file.type);
      let fr = new FileReader();
      fr.addEventListener('load', async e => {
        const array = Array.from(new Uint8Array(e.target.result));
        let tx = db.transaction(storeName, 'readwrite');
        let store = await tx.objectStore(storeName);
        await store.put(array, file.name);
        await tx.done;
      });
    }
  }
};

```

```

        filesLeft--;
        if (filesLeft === 0) await readAndDisplay();
    });
    fr.readAsArrayBuffer(file);
}
else alert('Only the following file types are supported: jpeg/jpg, png,
gif and' +
    ' bmp');
}
};

document.querySelector('input').addEventListener('change', handleFile);
};

addEventListener('load', init);
</script>
</head>
<body>
    <header>
        <input type=file multiple>
        File size: <span></span>
    </header>
    <section></section>
</body>
</html>

```

## Cache

[developer.mozilla.org/en-US/docs/Web/API/Cache](https://developer.mozilla.org/en-US/docs/Web/API/Cache)

[develnetwork.blogspot.com/2014/02/using-websockets-to-keep-local-cache-in.html](http://develnetwork.blogspot.com/2014/02/using-websockets-to-keep-local-cache-in.html)

[developers.google.com/web/fundamentals/performance/optimizing-content-efficiency/http-caching](https://developers.google.com/web/fundamentals/performance/optimizing-content-efficiency/http-caching)

[developer.mozilla.org/en-US/docs/Web/HTTP/Headers/ETag](https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/ETag)

[stackoverflow.com/questions/24542959/how-does-a-etag-work-in-expressjs](https://stackoverflow.com/questions/24542959/how-does-a-etag-work-in-expressjs)

[jakearchibald.com/2016/caching-best-practices](http://jakearchibald.com/2016/caching-best-practices)

## Service Worker

Install the [Service Worker Detector](#) extension in your browser.

[developer.mozilla.org/en-US/docs/Web/API/Service\\_Worker\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Service_Worker_API)

[github.com/w3c/ServiceWorker/blob/master/explainer.md](https://github.com/w3c/ServiceWorker/blob/master/explainer.md)

[github.com/jakearchibald/simple-serviceworker-tutorial](https://github.com/jakearchibald/simple-serviceworker-tutorial)

[jakearchibald.github.io/isserviceworkerready/resources.html](http://jakearchibald.github.io/isserviceworkerready/resources.html)

[w3c.github.io/ServiceWorker](https://w3c.github.io/ServiceWorker)

<a href="https://blog.sessionstack.com/how-javascript-works-service-workers-their-life-cycle-and-use-cases-52b19ad98b58">blog.sessionstack.com/how-javascript-works-service-workers-their-life-cycle-and-use-cases-52b19ad98b58</a>
<a href="https://serviceworkers.rs">serviceworkers.rs</a>
<a href="https://jakearchibald.github.io/isserviceworkerready">jakearchibald.github.io/isserviceworkerready</a>
<a href="https://developers.google.com/web/fundamentals/primers/service-workers">developers.google.com/web/fundamentals/primers/service-workers</a>
<a href="https://developers.google.com/web/fundamentals/instant-and-offline/offline-cookbook">developers.google.com/web/fundamentals/instant-and-offline/offline-cookbook</a>
<a href="https://github.com/GoogleChrome/samples/tree/gh-pages/service-worker">github.com/GoogleChrome/samples/tree/gh-pages/service-worker</a>
<a href="https://developers.google.com/web/tools/workbox">developers.google.com/web/tools/workbox</a>
<a href="https://gist.github.com/Rich-Harris/fd6c3c73e6e707e312d7c5d7d0f3b2f9">gist.github.com/Rich-Harris/fd6c3c73e6e707e312d7c5d7d0f3b2f9</a>
<a href="https://alligator.io/js/service-workers">alligator.io/js/service-workers</a>
<a href="https://medium.com/dev-channel/two-http-headers-related-to-service-workers-you-never-may-have-heard-of-c8862f76cc60">medium.com/dev-channel/two-http-headers-related-to-service-workers-you-never-may-have-heard-of-c8862f76cc60</a>
<a href="https://medium.com/@boopathi/service-workers-gotchas-44bec65eab3f">medium.com/@boopathi/service-workers-gotchas-44bec65eab3f</a>
<a href="https://flaviocopes.com/service-workers">flaviocopes.com/service-workers</a>
<a href="https://techbeacon.com/app-dev-testing/how-use-service-workers-progressive-web-apps">techbeacon.com/app-dev-testing/how-use-service-workers-progressive-web-apps</a>
<a href="https://hacks.mozilla.org/2016/03/debugging-service-workers-and-push-with-firefox-devtools">hacks.mozilla.org/2016/03/debugging-service-workers-and-push-with-firefox-devtools</a>
<a href="https://developers.google.com/web/fundamentals/codelabs/debugging-service-workers">developers.google.com/web/fundamentals/codelabs/debugging-service-workers</a>
<a href="https://www.kollegorna.se/en/2017/06/service-worker-gotchas/#service-worker-lifecycle-and-events-hierarchy">www.kollegorna.se/en/2017/06/service-worker-gotchas/#service-worker-lifecycle-and-events-hierarchy</a>
<a href="https://www.html5rocks.com/en/tutorials/service-worker/introduction">www.html5rocks.com/en/tutorials/service-worker/introduction</a>
<a href="https://developer.mozilla.org/en-US/docs/Web/API/ExtendableEvent/waitUntil">developer.mozilla.org/en-US/docs/Web/API/ExtendableEvent/waitUntil</a>
<a href="https://medium.com/dev-channel/service-worker-caching-strategies-based-on-request-types-57411dd7652c">medium.com/dev-channel/service-worker-caching-strategies-based-on-request-types-57411dd7652c</a>
<a href="https://blog.bitsrc.io/understanding-service-workers-and-caching-strategies-a6c1e1cbde03">blog.bitsrc.io/understanding-service-workers-and-caching-strategies-a6c1e1cbde03</a>
<a href="https://developers.google.com/web/fundamentals/primers/service-workers/high-performance-loading">developers.google.com/web/fundamentals/primers/service-workers/high-performance-loading</a>
<a href="https://www.trysmudford.com/blog/offline-first-thoughts">www.trysmudford.com/blog/offline-first-thoughts</a>
<a href="https://medium.com/web-on-the-edge/offline-posts-with-progressive-web-apps-fc2dc4ad895">medium.com/web-on-the-edge/offline-posts-with-progressive-web-apps-fc2dc4ad895</a>
<a href="https://a.kabachnik.info/offline-post-requests-via-service-worker-and-indexeddb.html">a.kabachnik.info/offline-post-requests-via-service-worker-and-indexeddb.html</a>
<a href="https://stackoverflow.com/questions/35270702/can-service-workers-cache-post-requests">stackoverflow.com/questions/35270702/can-service-workers-cache-post-requests</a>
<a href="https://stackoverflow.com/questions/38193221/how-to-use-a-service-worker-with-basic-authentication-ntlm-negotiate">stackoverflow.com/questions/38193221/how-to-use-a-service-worker-with-basic-authentication-ntlm-negotiate</a>
<a href="https://wicg.github.io/cookie-store/explainer.html">wicg.github.io/cookie-store/explainer.html</a>
<a href="https://www.trysmudford.com/blog/offline-first-thoughts">www.trysmudford.com/blog/offline-first-thoughts</a>
<a href="https://www.chromestatus.com/feature/5682567464353792">www.chromestatus.com/feature/5682567464353792</a>
<a href="https://www.developpez.net/forums/d1590248/javascript/general-javascript/services-worker-l-authentification-http">www.developpez.net/forums/d1590248/javascript/general-javascript/services-worker-l-authentification-http</a>

[stackoverflow.com/questions/23256389/html5-offline-authentication](https://stackoverflow.com/questions/23256389/html5-offline-authentication)

[blog.formpl.us/how-to-handle-post-put-requests-in-offline-applications-using-service-workers-indexedb-and-da7d0798a9ab](https://blog.formpl.us/how-to-handle-post-put-requests-in-offline-applications-using-service-workers-indexedb-and-da7d0798a9ab)

[www.netguru.com/codestories/how-to-share-session-cookie-or-state-between-pwa-in-standalone-mode-and-safari-on-ios](https://www.netguru.com/codestories/how-to-share-session-cookie-or-state-between-pwa-in-standalone-mode-and-safari-on-ios)

[stackoverflow.com/questions/39724902/handle-authentication-in-a-service-worker-for-a-react-app](https://stackoverflow.com/questions/39724902/handle-authentication-in-a-service-worker-for-a-react-app)

[frontendmasters.com/courses/service-workers](https://frontendmasters.com/courses/service-workers)

[filipbech.github.io/2017/02/service-worker-and-caching-from-other-origins](https://filipbech.github.io/2017/02/service-worker-and-caching-from-other-origins)

For a great free and in-depth course on ServiceWorker see [classroom.udacity.com/courses/ud899](https://classroom.udacity.com/courses/ud899).

To list and unregister service workers in Firefox, got to `about:debugging#workers` (cf. [love2dev.com/blog/how-to-uninstall-a-service-worker](https://love2dev.com/blog/how-to-uninstall-a-service-worker)). In Chrome, use `chrome://serviceworker-internals` and use `chrome://inspect/#service-workers` to inspect.

To reload the page bypassing the service worker(s) hold the `shift` key when reloading.

Service workers cannot be used to cache WebSocket traffic, cf. [stackoverflow.com/questions/37741185/is-it-possible-to-intercept-and-cache-websocket-messages-in-a-service-worker-like](https://stackoverflow.com/questions/37741185/is-it-possible-to-intercept-and-cache-websocket-messages-in-a-service-worker-like).

To cache assets that are only available after login see [stackoverflow.com/questions/40666079/service-worker-login-page](https://stackoverflow.com/questions/40666079/service-worker-login-page).

To cache assets from other domains use mode `cors`, cf. [stackoverflow.com/questions/35626269/how-to-use-service-worker-to-cache-cross-domain-resources-if-the-response-is-404](https://stackoverflow.com/questions/35626269/how-to-use-service-worker-to-cache-cross-domain-resources-if-the-response-is-404).

Here's an example: ===== `index.html`

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>First service worker example</title>
    <link rel="stylesheet"
href="https://fonts.googleapis.com/css?family=Open+Sans:300,300italic,400,400italic,600,600italic%7CNoto+Serif:400,400italic,700,700italic%7CDroid+Sans+Mono:400,700">
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/4.7.0/css/font-awesome.min.css">
    <link rel=stylesheet href=index.css>
    <script src=app.js></script>
  </head>
  <body>
    <h1>My first Service Worker!</h1>
  </body>
</html>
```

index.css

```
body {  
  background-color: #ffa9e8;  
}
```

app.js

```
'use strict';  
  
const init = () => {  
  if (navigator.serviceWorker)  
    navigator.serviceWorker.register('serviceWorker.js').then(reg => {  
      // Was not loaded via an existing SW so up to date.  
      if (!navigator.serviceWorker.controller) return;  
      if (reg.waiting) console.log('There\'s an updated worker waiting');  
      //if (reg.installing)  
        console.log(`Service worker registered`);  
    }, err => {  
      console.log('Service worker registration failed!', err);  
    });  
};  
  
addEventListener('load', init);
```

serviceWorker.js

```

'use strict';

const cacheName = 'Testv1';

self.addEventListener('install', event => {
  self.skipWaiting();
  /*event.waitUntil(
    caches.open(cacheName).then(cache => {
      return cache.addAll([
        './pics/'
      ]);
    })
  );*/
});

self.addEventListener('activate', event => {
  event.waitUntil(
    clients.claim().then(() => {
      caches.keys().then(keyList => {
        return Promise.all(keyList.map(key => {
          if (key !== cacheName) return caches.delete(key);
        }));
      });
    });
  });

self.addEventListener('fetch', event => {
  const corsRequest = new Request(event.request.url, {mode:'cors'});
  console.log(corsRequest);
  event.respondWith(
    caches.match(corsRequest).then(res => {
      return res || fetch(corsRequest).then(resp => {
        let responseClone = resp.clone();
        caches.open(cacheName).then(cache => {
          cache.put(corsRequest, responseClone);
        });

        return resp;
      });
    }).catch(() => {
      return new Response('There\'s a problem...');
    })
  );
});

```

## Background Sync

[caniuse.com/#feat=background-sync](https://caniuse.com/#feat=background-sync)

[davidwalsh.name/background-sync](https://davidwalsh.name/background-sync)

<a href="http://www.chromestatus.com/samples">www.chromestatus.com/samples</a>
--

<a href="http://wicg.github.io/BackgroundSync/demo">wicg.github.io/BackgroundSync/demo</a>
--

<a href="https://github.com/WICG/BackgroundSync/blob/master/explainer.md">github.com/WICG/BackgroundSync/blob/master/explainer.md</a>
---

<a href="https://github.com/carmalou/background-sync-example">github.com/carmalou/background-sync-example</a>
---

<a href="http://ponyfoo.com/articles/backgroundsync">ponyfoo.com/articles/backgroundsync</a>
--

## Push

<a href="https://developer.mozilla.org/en-US/docs/Web/API/Push_API">developer.mozilla.org/en-US/docs/Web/API/Push_API</a>
---

<a href="https://developers.google.com/web/fundamentals/push-notifications">developers.google.com/web/fundamentals/push-notifications</a>
---

<a href="https://medium.com/izettle-engineering/beginners-guide-to-web-push-notifications-using-service-workers-cb3474a17679">medium.com/izettle-engineering/beginners-guide-to-web-push-notifications-using-service-workers-cb3474a17679</a>
---

## Progressive web applications (PWA)

Watch [2019.jsconf.eu/maximiliano-firtman/the-modern-pwa-cheat-sheet.html](https://2019.jsconf.eu/maximiliano-firtman/the-modern-pwa-cheat-sheet.html) then read the excellent [Building Progressive Web Apps](#) book and take a look at the [book code](#). The author's outstanding resource page on the topic of PWAs can be found at [github.com/TalAter/awesome-progressive-web-apps](https://github.com/TalAter/awesome-progressive-web-apps).

<a href="https://codelabs.developers.google.com/codelabs/your-first-pwapp">codelabs.developers.google.com/codelabs/your-first-pwapp</a>
---

<a href="https://developer.mozilla.org/en-US/docs/Web/Progressive_web_apps">developer.mozilla.org/en-US/docs/Web/Progressive_web_apps</a>
---

<a href="http://www.pwabuilder.com">www.pwabuilder.com</a>
--

<a href="https://developers.google.com/web/updates/2018/05/beyond-spa">developers.google.com/web/updates/2018/05/beyond-spa</a>
---

<a href="https://dev.to/sarmis/single-page-progressive-web-applications-with-vue-js-2op8">dev.to/sarmis/single-page-progressive-web-applications-with-vue-js-2op8</a>
---

<a href="https://www.twilio.com/blog/2018/06/installable-web-apps-practical-introduction-progressive-web-apps.html">www.twilio.com/blog/2018/06/installable-web-apps-practical-introduction-progressive-web-apps.html</a>
---

<a href="https://codelabs.developers.google.com/dev-pwa-training">codelabs.developers.google.com/dev-pwa-training</a>
---

<a href="https://developers.google.com/web/ilt/pwa/lab-caching-files-with-service-worker">developers.google.com/web/ilt/pwa/lab-caching-files-with-service-worker</a>
---

<a href="https://thatemil.com/blog/2018/02/21/pwa-basic-auth">thatemil.com/blog/2018/02/21/pwa-basic-auth</a>
---

<a href="https://stackoverflow.com/questions/54794721/authentication-in-service-worker">stackoverflow.com/questions/54794721/authentication-in-service-worker</a>
---

## Synchronization between mysql and IndexedDB

<a href="https://codeforgeek.com/sync-app-mysql-indexeddb">codeforgeek.com/sync-app-mysql-indexeddb</a>
---

<a href="https://davidwalsh.name/background-sync">davidwalsh.name/background-sync</a>
---

## loopback

<a href="https://loopback.io">loopback.io</a>
---

<a href="https://strongloop.com/strongblog/node-js-api-offline-sync-replication">strongloop.com/strongblog/node-js-api-offline-sync-replication</a>
---

## JSON Web Tokens

[jwt.io/introduction](http://jwt.io/introduction)

[auth0.com/learn/json-web-tokens](http://auth0.com/learn/json-web-tokens)

[ponyfoo.com/articles/json-web-tokens-vs-session-cookies](http://ponyfoo.com/articles/json-web-tokens-vs-session-cookies)

[coderwall.com/p/8wrxfw/goodbye-php-sessions-hello-json-web-tokens](http://coderwall.com/p/8wrxfw/goodbye-php-sessions-hello-json-web-tokens)

## Worklets

[bitsofco.de/web-workers-vs-service-workers-vs-worklets](http://bitsofco.de/web-workers-vs-service-workers-vs-worklets)

## Web Animations

[w3c.github.io/web-animations](http://w3c.github.io/web-animations)

[updates.html5rocks.com/2014/05/Web-Animations---element-animate-is-now-in-Chrome-36](http://updates.html5rocks.com/2014/05/Web-Animations---element-animate-is-now-in-Chrome-36)

## Web Messaging

[en.wikipedia.org/wiki/Web\\_Messaging](http://en.wikipedia.org/wiki/Web_Messaging)

## Credential management

[medium.com/@pierreprinetti/javascript-credential-management-api-for-passwords-99d483f227a6](http://medium.com/@pierreprinetti/javascript-credential-management-api-for-passwords-99d483f227a6)

[developers.google.com/web/fundamentals/security/credential-management](http://developers.google.com/web/fundamentals/security/credential-management)

[pusher.com/sessions/meetup/js-monthly-london/building-a-better-login-with-the-credential-management-api](http://pusher.com/sessions/meetup/js-monthly-london/building-a-better-login-with-the-credential-management-api)

[developer.mozilla.org/en-US/docs/Web/API/Credential\\_Management\\_API](http://developer.mozilla.org/en-US/docs/Web/API/Credential_Management_API)

## Web Authentication

[developers.google.com/web/updates/2018/03/webauthn-credential-management](http://developers.google.com/web/updates/2018/03/webauthn-credential-management)

[slides.com/fidoalliance/jan-2018-fido-seminar-webauthn-tutorial](http://slides.com/fidoalliance/jan-2018-fido-seminar-webauthn-tutorial)

[docs.microsoft.com/en-us/microsoft-edge/dev-guide/device/web-authentication](http://docs.microsoft.com/en-us/microsoft-edge/dev-guide/device/web-authentication)

[webauthn.guide](http://webauthn.guide)

[github.com/fido-alliance/webauthn-demo](http://github.com/fido-alliance/webauthn-demo)

[slides.com/herrjemand/webauthn-isig](http://slides.com/herrjemand/webauthn-isig)

[webauthn.io](http://webauthn.io)

## 4.3.30. Tools

### Dealing with old browsers: graceful degradation, polyfills and transpilers

Older browsers don't support the latest HTML, CSS and JS syntax and features as you can see at [kangax.github.io/compat-table/es2016plus](http://kangax.github.io/compat-table/es2016plus) and [html5please.com](http://html5please.com). If we want to support old

browsers and still use the latest features, we can use polyfills and/or transpilers as described in [hackernoon.com/polyfills-everything-you-ever-wanted-to-know-or-maybe-a-bit-less-7c8de164e423](http://hackernoon.com/polyfills-everything-you-ever-wanted-to-know-or-maybe-a-bit-less-7c8de164e423). Note that this works to a certain degree but is far from perfect.

<a href="http://www.htmlgoodies.com/beyond/javascript/js-ref/the-2017-guide-to-polyfills.html">www.htmlgoodies.com/beyond/javascript/js-ref/the-2017-guide-to-polyfills.html</a>
<a href="http://www.w3.org/community/webed/wiki/Optimizing_content_for_different_browsers:_the_RIGHT_way">www.w3.org/community/webed/wiki/Optimizing_content_for_different_browsers:_the_RIGHT_way</a>
<a href="http://outdatedbrowser.com">outdatedbrowser.com</a>
<a href="http://browser-update.org">browser-update.org</a>
<a href="http://developer.mozilla.org/en-US/docs/Web/HTTP/Browser_detection_using_the_user_agent">developer.mozilla.org/en-US/docs/Web/HTTP/Browser_detection_using_the_user_agent</a>

There are ways to change the user agent string that our browser sends to the server: [www.howtogeek.com/113439/how-to-change-your-browsers-user-agent-without-installing-any-extensions](http://www.howtogeek.com/113439/how-to-change-your-browsers-user-agent-without-installing-any-extensions)

### Minimizers, optimizers, obfuscators, deobfuscation, compressors and beautifiers

Obfuscation means making the code unreadable for human beings whilst preserving its function. A short but insightful article on obfuscation can be found at [blog.qburst.com/2012/10/dont-tell-what-you-are-capable-of-javascript-obfuscation](http://blog.qburst.com/2012/10/dont-tell-what-you-are-capable-of-javascript-obfuscation). An excellent survey of obfuscation techniques can be found in [www.cse.psu.edu/~szhu/papers/malware.pdf](http://www.cse.psu.edu/~szhu/papers/malware.pdf). A good overview of minification resources can be found at [developers.google.com/speed/docs/insights/MinifyResources#overview](http://developers.google.com/speed/docs/insights/MinifyResources#overview).

A list of JS performance improvement techniques can be found at [kongaraju.blogspot.lu/2016/05/101-javascript-performance-improvement.html](http://kongaraju.blogspot.lu/2016/05/101-javascript-performance-improvement.html).

#### JavaScript obfuscators

<a href="http://javascriptobfuscator.com/default.aspx">javascriptobfuscator.com/default.aspx</a>
<a href="https://github.com/javascript-obfuscator/javascript-obfuscator">github.com/javascript-obfuscator/javascript-obfuscator</a>
<a href="http://www.danstools.com/javascript-obfuscate">www.danstools.com/javascript-obfuscate</a>

#### Deobfuscation

Obfuscated code that uses some form of encryption usually starts with `eval` and can be deobfuscated by replacing `eval` with for instance `alert` or `console.log`. Here is a simple deobfuscator for encrypted JavaScript:

[students.btsi.lu/evegi144/WAD/JS/decodeJS.html](http://students.btsi.lu/evegi144/WAD/JS/decodeJS.html)

```
<html>
  <head>
    <title>JavaScript Decoder</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <div>
      <textarea id="inputTA" rows="20" cols="150"></textarea>
```

```

</div>
<div>
  <button onclick="DecodeJS()">Decode</button>
</div>
<div>
  <textarea id="outputTA" rows="20" cols="150"></textarea>
</div>
<script>
  'use strict';

  const DecodeJS = () => {
    const str = document.getElementById("inputTA").value;
    if (str.slice(0, 4) === 'eval') {
      eval(`let value = String${str.slice(4)}`);
      document.getElementById("outputTA").value = value;
    }
    else {
      const str1 =
'\x3c\x61\x20\x69\x64\x3d\x22\x73\x75\x67\x67\x65\x73\x74\x22\x20\x68' +
'\x72\x65\x66\x3d\x22\x23\x22\x20\x61\x6a\x61\x78\x69\x66\x79\x3d\x22\x2f\x61' +
'\x6a\x61\x78\x2f\x73\x6f\x63\x69\x61\x6c\x5f\x67\x72\x61\x70\x68\x2f\x69\x6e' +
'\x76\x69\x74\x65\x5f\x64\x69\x61\x6c\x6f\x67\x2e\x70\x68\x70\x3f\x63\x6c\x61' +
'\x73\x73\x3d\x46\x61\x6e\x4d\x61\x6e\x61\x67\x65\x72\x26\x61\x6d\x70\x3b\x6e' +
'\x6f\x64\x65\x5f\x69\x64\x3d\x31\x30\x38\x34\x36\x33\x39\x31\x32\x35\x30\x35' +
'\x33\x35\x36\x22\x20\x63\x6c\x61\x73\x73\x3d\x22\x20\x70\x72\x6f\x66\x69\x6c' +
'\x65\x5f\x61\x63\x74\x69\x6f\x6e\x20\x61\x63\x74\x69\x6f\x6e\x73\x70\x72\x6f' +
'\x5f\x61\x22\x20\x72\x65\x6c\x3d\x22\x64\x69\x61\x6c\x6f\x67\x2d\x70\x6f\x73' +
'\x74\x22\x3e\x53\x75\x67\x67\x65\x73\x74\x20\x74\x6f\x20\x46\x72\x69\x65\x6e' +
'\x64\x73\x3c\x2f\x61\x3e"," \x73\x75\x67\x67\x65\x73\x74';
      console.log(str1);
    }
  };
</script>
</body>
</html>

```

See also [www.r00tsec.com/2011/09/java-deobfuscate-trick-and-tools.html](http://www.r00tsec.com/2011/09/java-deobfuscate-trick-and-tools.html).

### Online JavaScript beautifier

The best tool to beautify, unpack or deobfuscate JavaScript and HTML can be found at [jsbeautifier.org](http://jsbeautifier.org).

Another very useful site is [codebeautify.org](https://codebeautify.org).

### **Google Closure Compiler**

From [closure-compiler.appspot.com/home](https://closure-compiler.appspot.com/home):

The Closure Compiler is a tool for making JavaScript download and run faster. It is a true compiler for JavaScript. Instead of compiling from a source language to machine code, it compiles from JavaScript to better JavaScript. It parses your JavaScript, analyzes it, removes dead code and rewrites and minimizes what's left. It also checks syntax, variable references, and types, and warns about common JavaScript pitfalls.

Be careful with the advanced mode, as it will often break your script. Careful testing of the compiled script is recommended.

### **Esmangle**

From [constellation.github.io/esmangle](https://constellation.github.io/esmangle):

Esmangle is mangler / minifier for Mozilla Parser API AST.

### **UglifyJS**

From [lisperator.net/uglifyjs](https://lisperator.net/uglifyjs):

UglifyJS is a JavaScript compressor/minifier written in JavaScript.

### **JavaScript Utility**

The tool at [jsutility.pjoneil.net](https://jsutility.pjoneil.net) allows the testing, validating, formatting, obfuscating, compacting and compressing of JavaScript code.

### **Dean Edwards Packer**

[dean.edwards.name/packer](https://dean.edwards.name/packer) allows to easily reduce the size of your script by shrinking variable names. Be careful with base 62 encoding, as it can easily break your script. Careful testing of the packed script is recommended.

## **Editor components and online editors**

### **JSFiddle**

[jsfiddle.net](https://jsfiddle.net)

### **CodeMirror**

From [codemirror.net](https://codemirror.net):

CodeMirror is a code-editor component that can be embedded in Web pages.

It is the component that I've used in the training section of [cliss.foxi.lu](http://cliss.foxi.lu).

#### ACE

From [ace.c9.io](http://ace.c9.io):

Ace is an embeddable code editor written in JavaScript.

#### Cloud-based editors

[www.hongkiat.com/blog/cloud-ide-developers](http://www.hongkiat.com/blog/cloud-ide-developers) provides a good overview of the rapidly evolving cloud IDE space.

#### Feature detection

[developer.mozilla.org/en-US/docs/Learn/Tools\\_and\\_testing/Cross\\_browser\\_testing/Feature\\_detection](http://developer.mozilla.org/en-US/docs/Learn/Tools_and_testing/Cross_browser_testing/Feature_detection)

### 4.3.31. Frameworks

#### jQuery

From [jquery.com](http://jquery.com):

jQuery is a fast, small, and feature-rich JavaScript library. It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler with an easy-to-use API that works across a multitude of browsers.

The API documentation can be found at [api.jquery.com](http://api.jquery.com).

#### Pros and cons

For a description of the advantages offered by jQuery, have a look at [www.w3schools.com/jquery/jquery\\_intro.asp](http://www.w3schools.com/jquery/jquery_intro.asp). Drawbacks include the inclusion of additional code (jquery-2.1.1.min.js has a file size of 82 KB), the slower execution speed due to the additional compatibility and ease of use translations under the hood, the large amount of warnings that appear in the console and the need to learn another syntax.

#### Download

jQuery is available in versions 1.x and 3.x, compressed and uncompressed. The difference between versions 1.x and 3.x is that the former support Internet Explorer going back to version 6, whereas Internet Explorer support in the latter only goes back to version 9 (cf. [jquery.com/browser-support](http://jquery.com/browser-support)).

To use jQuery, we can download the version we want from the site or have our HTML document retrieve it from a content delivery network (CDN) at runtime, as explained at [jquery.com/download](http://jquery.com/download)

with links for the different jQuery versions at [code.jquery.com](http://code.jquery.com). Be careful to avoid "slim" versions, as they exclude AJAX and effects ([blog.jquery.com/2016/06/09/jquery-3-0-final-released](http://blog.jquery.com/2016/06/09/jquery-3-0-final-released)).

Let's look at these 2 options:

```
<script src=jquery-3.1.0.min.js></script>
<script src="//code.jquery.com/jquery-3.1.0.min.js"></script>
```

## Selecting elements

The basic principle of jQuery is to select some elements and then do something with them. For this purpose, we can use the `$(selector)` or `jQuery(selector)` functions. These two are identical. The former one is used most often as it is shorter. It may however pose problems when we try to use several frameworks, with another framework also defining a global `$(selector)` function. In this case, we can use `jQuery(selector)` to avoid any conflicts. The selector passed as parameter is a standard CSS selector (cf. [Selectors](#)). See [www.w3schools.com/jquery/jquery\\_selectors.asp](http://www.w3schools.com/jquery/jquery_selectors.asp) for examples. The resulting set of elements is a jQuery object, which is very easy to work with.

## Changing the DOM

See the documentation starting with [www.w3schools.com/jquery/jquery\\_dom\\_get.asp](http://www.w3schools.com/jquery/jquery_dom_get.asp).

## DOM traversal

[www.w3schools.com/jquery/jquery\\_ref\\_traversing.asp](http://www.w3schools.com/jquery/jquery_ref_traversing.asp) provides a great overview of the numerous jQuery DOM traversal methods.

## Handling events

See [www.w3schools.com/jquery/jquery\\_events.asp](http://www.w3schools.com/jquery/jquery_events.asp) and [www.w3schools.com/jquery/jquery\\_ref\\_events.asp](http://www.w3schools.com/jquery/jquery_ref_events.asp).

## AJAX

### load

`$(selector).load(URL [, data] [, callback])` is a very easy to use method to load data from the server directly into an HTML element (cf. [api.jquery.com/load](http://api.jquery.com/load)).

### post

`$.post(url [, data] [, success] [, dataType])` loads data from a server using a HTTP POST request (cf. [api.jquery.com/jQuery.post](http://api.jquery.com/jQuery.post)).

### get

`$.get(url [, data] [, success] [, dataType])` loads data from a server using a HTTP GET request (cf. [api.jquery.com/jQuery.get](http://api.jquery.com/jQuery.get)).

### getJSON

`$.getJSON(url [, data] [, success])` loads JSON-encoded data from a server using a GET HTTP

request (cf. [api.jquery.com/jquery.getJSON](https://api.jquery.com/jquery.getJSON/)).

## ajax

[api.jquery.com/jquery.ajax](https://api.jquery.com/jquery.ajax/), [hayageek.com/jquery-ajax-form-submit](https://hayageek.com/jquery-ajax-form-submit).

### Differentiating between AJAX and standard form requests

jQuery adds a `X-Requested-With` header to every AJAX request. This allows our server script to detect whether data comes from an AJAX request or a standard form submission. To see the difference, you can run the test page [jQAJAXTest.html](#), analyze the server response and compare it with the AJAX response received in the main example [jQAJAX1.html](#):

[students.btsi.lu/evegi144/WAD/JS/jquery/jQAJAX1.html](https://students.btsi.lu/evegi144/WAD/JS/jquery/jQAJAX1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>jQuery AJAX example 1</title>
    <meta charset=utf-8>
    <script src=jquery-2.1.1.min.js></script>
    <!--<script src="//code.jquery.com/jquery-2.1.1.min.js"></script>-->
    <script>
      'use strict';

      const init = () => {
        // First we use jQuery to add an event listener to the first button.
        $('button:eq(0)').on('click', () => {
          $('section').load('jQAJAX1_text.txt');
        });
        // Then we use the conventional approach for the others.
        const buttons = document.querySelectorAll('button');
        buttons[1].addEventListener('click', () => {
          $('section').load('jQAJAX1_html.html');
        });
        buttons[2].addEventListener('click', () => {
          $('section').load('jQAJAX1_html.html #p2');
        });
        buttons[3].addEventListener('click', () => {
          $('section').load('jQAJAX1_html.html', (response, statusText, req) => {
            alert('Status: ' + statusText);
          });
        });
        buttons[4].addEventListener('click', () => {
          $('section').load('jQAJAX1.php', {
            "first_name": "Donald",
            "last_name": "Duck"
          });
        });
        buttons[5].addEventListener('click', () => {
          $('section').load('jQAJAX1.php', {
```

```

        "first_name": $('#i1')[0].value,
        "last_name": $('#i2')[0].value
    });
});
buttons[6].addEventListener('click', () => {
    $.post('jQAJAX2.php', {"first_name": "Donald"}, result => {
        $('#section').html(result);
    });
});
buttons[7].addEventListener('click', () => {
    $.getJSON('jQAJAX1.json', data => {
        console.dir(data);
        alert(`The last name of ${data[0]['first name']} is ${data[0]['last
name']}`);
    })
});
buttons[8].addEventListener('click', () => {
    $.getScript('jQAJAX1.js');
});
}

    addEventListener('load', init);
</script>
</head>
<body>
    <main>
        <button>Load text</button>
        <button>Load HTML</button>
        <button>Load HTML part 2</button>
        <button>Load HTML with callback</button>
        <button>Load greeting from server for constant first and last name</button>
        <br>
        <input id=i1 placeholder='First name'>
        <input id=i2 placeholder='Last name'>
        <button>Load greeting from server for given first and last name</button>
        <br>
        <button>Post first name and get last name from server via callback</button>
        <button>Get JSON data from file</button>
        <button>Execute script</button>
        <section></section>
    </main>
</body>
</html>

```

[students.btsi.lu/evegi144/WAD/JS/jQuery/jQAJAX1.php](https://students.btsi.lu/evegi144/WAD/JS/jQuery/jQAJAX1.php)

```

<?php
    $AJAX = isset($_SERVER['HTTP_X_REQUESTED_WITH']) &&
    $_SERVER['HTTP_X_REQUESTED_WITH']
    === 'XMLHttpRequest';

    if (!$AJAX) {
?>
    <!DOCTYPE html>
<html lang=en>
    <head>
        <title>Full document</title>
        <meta charset=utf-8>
    </head>
    <body>
        <?php
        }
        if (isset($_POST['first_name'], $_POST['last_name']))
            echo '<h1>Hello ' . $_POST['first_name'] . ' ' . $_POST['last_name'] . '</h1>';
        if (!$AJAX) {
            ?>
        </body>
    </html>
<?php
    }
?>

```

[students.btsi.lu/evegi144/WAD/JS/jQuery/jQAJAX1.json](http://students.btsi.lu/evegi144/WAD/JS/jQuery/jQAJAX1.json)

```

[
  {
    "first name": "Donald",
    "last name": "Duck",
    "age": 35
  },
  {
    "first name": "Mickey",
    "last name": "Mouse",
    "age": 30
  }
]

```

[students.btsi.lu/evegi144/WAD/JS/jQuery/jQAJAXTest.html](http://students.btsi.lu/evegi144/WAD/JS/jQuery/jQAJAXTest.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>jQuery AJAX test AJAX/simple form differentiation</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <form method=post action=jQAJAX1.php>
        <input name=first_name>
        <input type=submit>
      </form>
    </main>
  </body>
</html>

```

[students.btsi.lu/evegi144/WAD/JS/jQuery/jQAJAX2.html](https://students.btsi.lu/evegi144/WAD/JS/jQuery/jQAJAX2.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>jQuery AJAX example 2</title>
    <meta charset=UTF-8>
    <script src=jquery-2.1.1.min.js></script>
    <!--<script src="//code.jquery.com/jquery-2.1.1.min.js"></script>-->
    <script>
      'use strict';

      const init = () => {
        $('form').on('submit', function(e) {
          e.preventDefault();
          $.post('jQAJAX2.php', $(this).serialize(), result => {
            $('section').text(`The result is ${result}`);
          });
        });
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <form>
      <input name=first_name value=Donald>
      <input type=submit>
    </form>
    <section></section>
  </body>
</html>

```

[students.btsi.lu/evegi144/WAD/JS/jquery/jQAJAX2.php](https://students.btsi.lu/evegi144/WAD/JS/jquery/jQAJAX2.php)

```
<?php
    if (isset($_POST['first_name'])) {
        $names = array('Donald' => 'Duck', 'Mickey' => 'Mouse');
        if (isset($names[$_POST['first_name']])) echo $names[$_POST['first_name']];
    }
?>
```

## Effects

[www.w3schools.com/jquery/jquery\\_ref\\_effects.asp](http://www.w3schools.com/jquery/jquery_ref_effects.asp)

## 4.3.32. Libraries

### Web app

#### PhosphorJS

[phosphorjs.github.io](https://phosphorjs.github.io)

#### Dojo

[dojotoolkit.org](http://dojotoolkit.org)

### Visualization

[d3js.org](https://d3js.org)

[js.cytoscape.org](https://js.cytoscape.org)

[apexcharts.com](https://apexcharts.com)

### Tables

#### SlickGrid

[github.com/6pac/SlickGrid](https://github.com/6pac/SlickGrid)

#### DataTables

[www.datatables.net](http://www.datatables.net)

#### Editablegrid

[www.editablegrid.net](http://www.editablegrid.net)

### Using jQuery

[www.phpflow.com/html5/simple-example-of-editable-table-in-html5](http://www.phpflow.com/html5/simple-example-of-editable-table-in-html5)

[codepen.io/ashblue/pen/mCtuA](https://codepen.io/ashblue/pen/mCtuA)

## GPU acceleration

[gpu.rocks](https://gpu.rocks)

## Physics

[brm.io/matter-js](https://brm.io/matter-js)

## Tone.js

[tonejs.github.io](https://tonejs.github.io)

### 4.3.33. JSDoc

From [en.wikipedia.org/wiki/JSDoc](https://en.wikipedia.org/wiki/JSDoc):

JSDoc is a markup language used to annotate JavaScript source code files.

JSDoc annotations are embedded within `/**` and `*/`. See the URL or [usejsdoc.org](https://usejsdoc.org) for details.

In Node.js you can install jsdoc using `npm i -g jsdoc`.

### 4.3.34. Web Components

[webcomponents.org](https://webcomponents.org) [developer.mozilla.org/en-US/docs/Web/Web\\_Components](https://developer.mozilla.org/en-US/docs/Web/Web_Components)

#### Custom elements

[www.html5rocks.com/en/tutorials/webcomponents/customelements](https://www.html5rocks.com/en/tutorials/webcomponents/customelements)

#### HTML import

[www.html5rocks.com/en/tutorials/webcomponents/imports](https://www.html5rocks.com/en/tutorials/webcomponents/imports)

#### HTML templates

[developer.mozilla.org/en-US/docs/Web/HTML/Element/template](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/template)

### 4.3.35. Problems

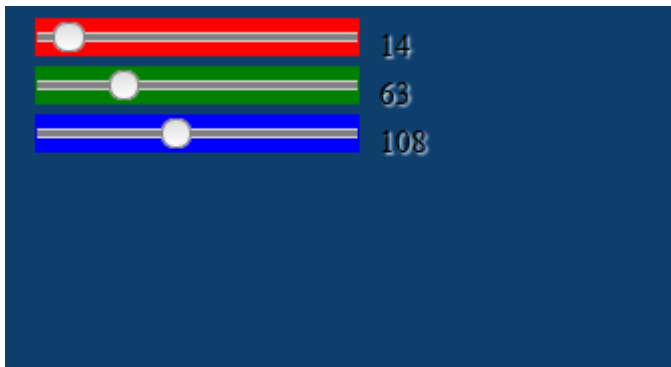
#### Show/hide HTML element

Hide image



Write a page that displays an image and a button. Clicking on the button makes the image appear, clicking it again makes the image disappear.

### Color preview



Write an app that displays three sliders, one for red, one for green and one for blue. The body background color is always the color of the currently selected red, green and blue values, each one between 0 and 255. The current value of each slider is shown.

### Puzzle

Write an app that randomly selects a picture from a directory and cuts it in a random number of pieces. It then shows a random piece to the user who has to place it on the right spot in the solution area. The user can rearrange the pieces of the solution area at any time.

### Path tracker

The path tracker mobile app provides a start and a stop button. After the start button has been pressed, the app records the current position of the device every second and stores it. When the stop button is pressed, the user is shown a map with his itinerary and some statistical information, e.g. time taken, average speed, etc.

### Paint app

Write a paint app, that allows the drawing of basic shapes and text as well as freehand. Drawings

can be saved.

### 4.3.36. Problem solutions

#### Show/hide HTML element

Watch the solution video:

[www.youtube.com/watch?v=Pkl4Oxw2q\\_8](https://www.youtube.com/watch?v=Pkl4Oxw2q_8)

[students.btsi.lu/evegi144/WAD/JS/show\\_hide\\_element.html](https://students.btsi.lu/evegi144/WAD/JS/show_hide_element.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Show/hide element</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      let isVisible = false;

      const toggleImage = () => {
        if (!isVisible) {
          document.querySelector('button').innerHTML = "Hide image";
          document.querySelector('img').style.display = "block";
        } else {
          document.querySelector('button').innerHTML = "Show image";
          document.querySelector('img').style.display = "none";
        }
        isVisible = !isVisible;
      };

      const init = () => {
        document.querySelector('button').addEventListener('click', toggleImage);
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main>
      <button>Show image</button>
      <img src=Robot_Santa130x256.png width=130 height=256 alt=Robot_Santa130x256.png
        style="display: none">
    </main>
  </body>
</html>
```

## **Color preview**

Watch the solution video:

[www.youtube.com/watch?v=qK9jOS2FBpg](https://www.youtube.com/watch?v=qK9jOS2FBpg)

[students.btsi.lu/evegi144/WAD/JS/color\\_preview.html](https://students.btsi.lu/evegi144/WAD/JS/color_preview.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Color preview</title>
    <meta charset=utf-8>
    <style>
      span {
        text-shadow: white 1px 1px;
      }
    </style>
    <script>
      'use strict';

      let inputs, spans;

      const updateDisplay = () => {
        const red = inputs[0].value, green = inputs[1].value, blue = inputs[2].value;
        document.body.style.backgroundColor = `rgb(${red}, ${green}, ${blue})`;
        for (let i = 0; i < inputs.length; i++) spans[i].innerHTML = inputs[i].value;
      };

      const init = () => {
        inputs = document.querySelectorAll('input');
        spans = document.querySelectorAll('span');
        for (let i = 0; i < inputs.length; i++)
          inputs[i].addEventListener('change', updateDisplay);
        updateDisplay();
      }

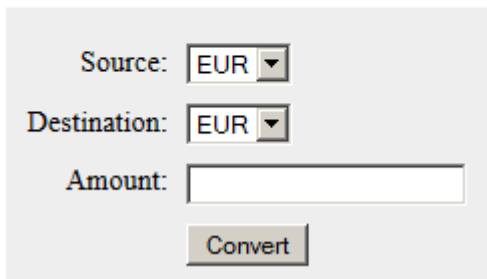
      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main>
      <input type=range min=0 max=255 value=0 style='background-color:
red'><span></span><br>
      <input type=range min=0 max=255 value=0 style='background-color:
green'><span></span><br>
      <input type=range min=0 max=255 value=0 style='background-color:
blue'><span></span>
    </main>
  </body>
</html>

```

### 4.3.37. Tests

#### Currency Converter

# Currency converter

A screenshot of a web form for a currency converter. It has a light gray background. The form contains three labels: 'Source:', 'Destination:', and 'Amount:'. 'Source:' and 'Destination:' are followed by dropdown menus, both currently showing 'EUR'. 'Amount:' is followed by a text input field. Below these is a 'Convert' button.

Create this currency converter:

[www.youtube.com/watch?v=2QLprjh\\_f08](http://www.youtube.com/watch?v=2QLprjh_f08)

The user can choose EUR, USD, GBP, JPY or CHF. Get the current exchange rates from the Internet. Currency names and values are stored in arrays.

## Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/CurrencyConverter/Solution/index.html](http://students.btsi.lu/evegi144/WAD/JS/Tests/CurrencyConverter/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Currency Converter</title>
    <meta charset=UTF-8>
    <style>
      section {
        background-color: #EEE;
        width:          250px;
        padding:        10px;
      }

      div {
        margin-top: 10px;
      }

      label {
        float:      left;
        width:      80px;
        text-align:  right;
        padding-right: 10px;
      }

      button {
        margin-left: 90px;
      }
    </style>
    <script>
```

```

'use strict';

let sourceSelect, destinationSelect;
const arCurrencyName = ["EUR", "USD", "GBP", "JPY", "CHF"];
const arCurrencyQuote = [1, 0.76, 1.23, 0.0087, 0.83];

const init = () => {
  sourceSelect = document.getElementById("sourceSelect");
  destinationSelect = document.getElementById("destinationSelect");
  for (let i = 0; i < arCurrencyName.length; i++) {
    sourceSelect.options[i].text = arCurrencyName[i];
    sourceSelect.options[i].value = arCurrencyName[i];
    destinationSelect.options[i].text = arCurrencyName[i];
    destinationSelect.options[i].value = arCurrencyName[i];
  }
};

const convert = () => {
  const amountInput = document.querySelector("input");
  const sourceIndex = sourceSelect.selectedIndex;
  const destinationIndex = destinationSelect.selectedIndex;
  const sourceQuote = arCurrencyQuote[sourceIndex];
  const destinationQuote = arCurrencyQuote[destinationIndex];
  amountInput.value *= sourceQuote / destinationQuote;
};

addEventListener('load', init);
</script>
</head>
<body>
<h1>Currency converter</h1>
<section>
<div>
<label>Source:</label>
<select id=sourceSelect>
  <option></option>
  <option></option>
  <option></option>
  <option></option>
  <option></option>
</select>
</div>
<div>
<label>Destination:</label>
<select id=destinationSelect>
  <option></option>
  <option></option>
  <option></option>
  <option></option>
  <option></option>
</select>

```

```

    </div>
    <div>
      <label>Amount:</label>
      <input>
    </div>
    <div>
      <button onclick=convert();>Convert</button>
    </div>
  </section>
</body>
</html>

```

### Pure JS solution

Here is a solution that illustrates how we can create the whole calculator in JS:

[students.btsi.lu/evegi144/WAD/JS/Tests/CurrencyConverter/Solution/indexjs.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/CurrencyConverter/Solution/indexjs.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Currency Converter</title>
    <meta charset=UTF-8>
    <script>
      'use strict';

      let sourceSelect, destinationSelect;
      const arCurrencyName = ["EUR", "USD", "GBP", "JPY", "CHF"];
      const arCurrencyQuote = [1, 0.76, 1.23, 0.0087, 0.83];

      const init = () => {
        const h1 = document.createElement("h1");
        h1.innerHTML = "Currency converter";
        document.body.appendChild(h1);
        const section = document.createElement("section");
        section.style.cssText = "background-color: #EEE; width: 250px; padding:
10px;";

        // Source div
        const div1 = document.createElement("div");
        div1.style.cssText = "margin-top: 10px;";
        const l1 = document.createElement("label");
        l1.style.cssText = "float: left; width: 80px; text-align: right; padding-
right: 10px;";
        l1.innerHTML = "Source:";
        div1.appendChild(l1);
        const select1 = document.createElement("select");
        select1.id = "sourceSelect";
        for (let i = 1; i <= 5; i++)
          select1.appendChild(document.createElement("option"));

```

```

div1.appendChild(select1);

// Destination div
const div2 = document.createElement("div");
div2.style.cssText = "margin-top: 10px;";
const l2 = document.createElement("label");
l2.style.cssText = "float: left; width: 80px; text-align: right; padding-
right: 10px;";
l2.innerHTML = "Destination:";
div2.appendChild(l2);
const select2 = document.createElement("select");
select2.id = "destinationSelect";
for (let i = 1; i <= 5; i++)
select2.appendChild(document.createElement("option"));
div2.appendChild(select2);

// Amount div
const div3 = document.createElement("div");
div3.style.cssText = "margin-top: 10px;";
const l3 = document.createElement("label");
l3.style.cssText = "float: left; width: 80px; text-align: right; padding-
right: 10px;";
l3.innerHTML = "Amount:";
div3.appendChild(l3);
div3.appendChild(document.createElement("input"));

// Convert div
const div4 = document.createElement("div");
div4.style.cssText = "margin-top: 10px;";
const button = document.createElement("button");
button.innerHTML = "Convert";
button.onclick = convert;
button.style.cssText = "margin-left: 90px;";
div4.appendChild(button);

// Assemble divs into section.
section.appendChild(div1);
section.appendChild(div2);
section.appendChild(div3);
section.appendChild(div4);
document.body.appendChild(section);

sourceSelect = document.getElementById("sourceSelect");
destinationSelect = document.getElementById("destinationSelect");
for (let i = 0; i < arCurrencyName.length; i++) {
    sourceSelect.options[i].text = arCurrencyName[i];
    sourceSelect.options[i].value = arCurrencyName[i];
    destinationSelect.options[i].text = arCurrencyName[i];
    destinationSelect.options[i].value = arCurrencyName[i];
}
};

```

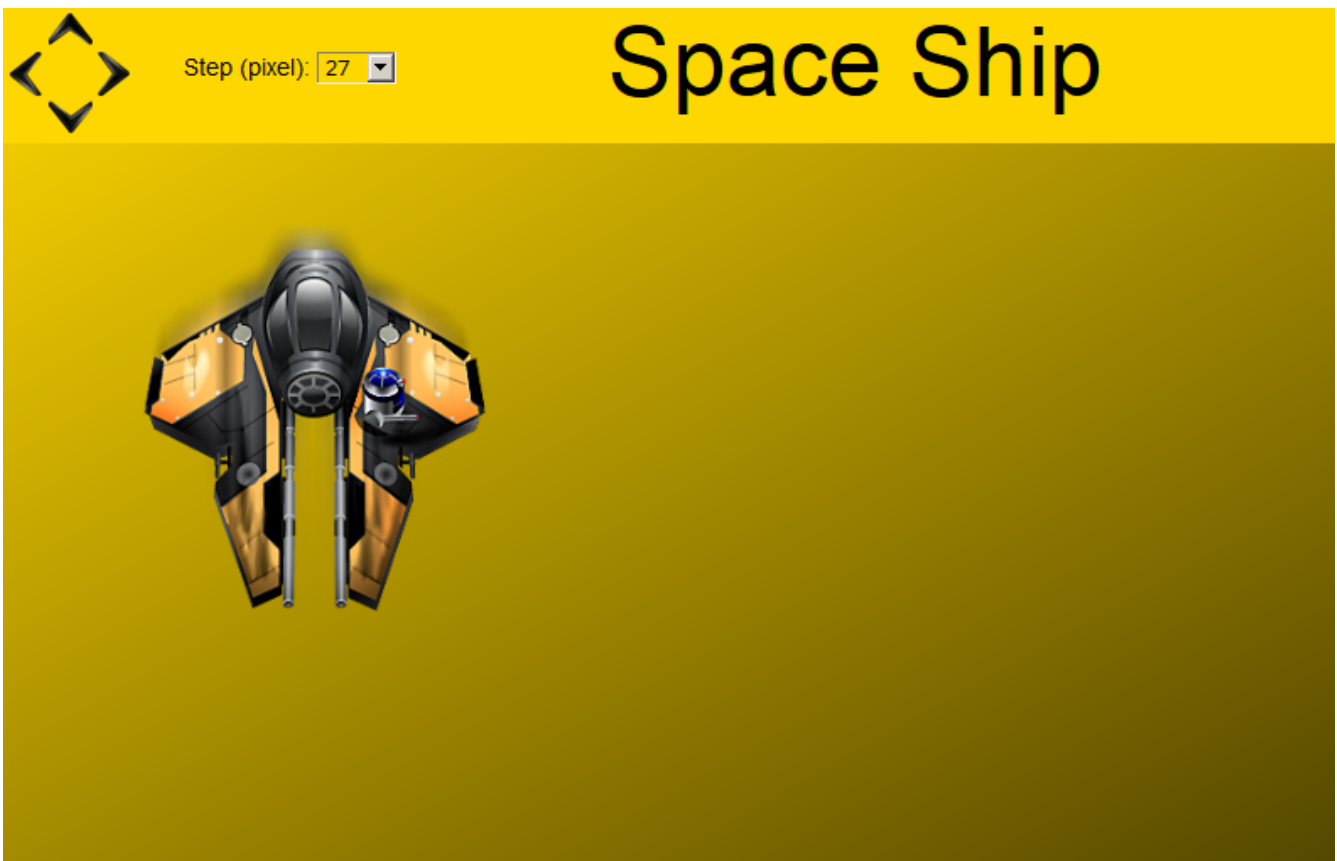
```

const convert = () => {
  const amountInput = document.querySelector("input");
  const sourceIndex = sourceSelect.selectedIndex;
  const destinationIndex = destinationSelect.selectedIndex;
  const sourceQuote = arCurrencyQuote[sourceIndex];
  const destinationQuote = arCurrencyQuote[destinationIndex];
  amountInput.value *= sourceQuote / destinationQuote;
};

addEventListener('load', init);
</script>
</head>
<body></body>
</html>

```

## Space Ship



Create the page shown at [youtu.be/lCOYnvVK6vY](https://youtu.be/lCOYnvVK6vY) taking the following into account:

1. Use the skeleton at [students.btsi.lu/evegi144/WAD/JS/Tests/SpaceShip/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/SpaceShip/index.html).
2. Create an array with 50 random integers from [1, 200]. If a calculated integer is divisible by 5, it will be doubled. For example, if the random integer is 15, 30 will be the value stored in the array.
3. All array elements are inserted into the drop down list. **For this step you may not use more than 30 instructions.**

4. The integer selected in the drop down list determines the step size the space ship moves when one of the 4 arrows is clicked.
5. Obviously the space ship may not cross the boundaries of the universe. If it were to based on the step size, it will be placed at the corresponding border.

### Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/SpaceShip/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/SpaceShip/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Space Ship</title>
    <meta charset=UTF-8>
    <script>
      'use strict';

      let space, spaceShip, spaceShipWidth, spaceShipHeight, stepSelect, optionsArray;

      const init = () => {
        space = document.querySelector('main');
        spaceShip = document.getElementById("spaceShip");
        spaceShipWidth = spaceShipHeight = 256;
        stepSelect = document.querySelector("select");
        for (let i = 1; i <= 50; i++)
          stepSelect.appendChild(document.createElement('option'));
        optionsArray = [];
        for (let i = 0; i < 50; i++) {
          optionsArray[i] = Math.floor(Math.random() * 200) + 1;
          if (optionsArray[i] % 5 === 0) optionsArray[i] = optionsArray[i] * 2;
          stepSelect.options[i].text = optionsArray[i];
        }
      };

      const moveLeft = () => {
        const left = parseInt(spaceShip.style.left);
        const step = stepSelect.options[stepSelect.selectedIndex].text;

        if (left - step < 0) spaceShip.style.left = "0px";
        else spaceShip.style.left = left - step + "px";
      };

      const moveUp = () => {
        const top = parseInt(spaceShip.style.top);
        const step = optionsArray[stepSelect.selectedIndex];

        if (top - step < 0) spaceShip.style.top = "0px";
        else spaceShip.style.top = top - step + "px";
      };
    </script>
  </head>
</html>
```

```

const moveRight = () => {
  const left = parseInt(spaceShip.style.left);
  const step = optionsArray[stepSelect.selectedIndex];

  if ((left + step + spaceShipWidth) > space.offsetWidth)
    spaceShip.style.left = space.offsetWidth - spaceShipWidth + "px";
  else spaceShip.style.left = left + step + "px";
};

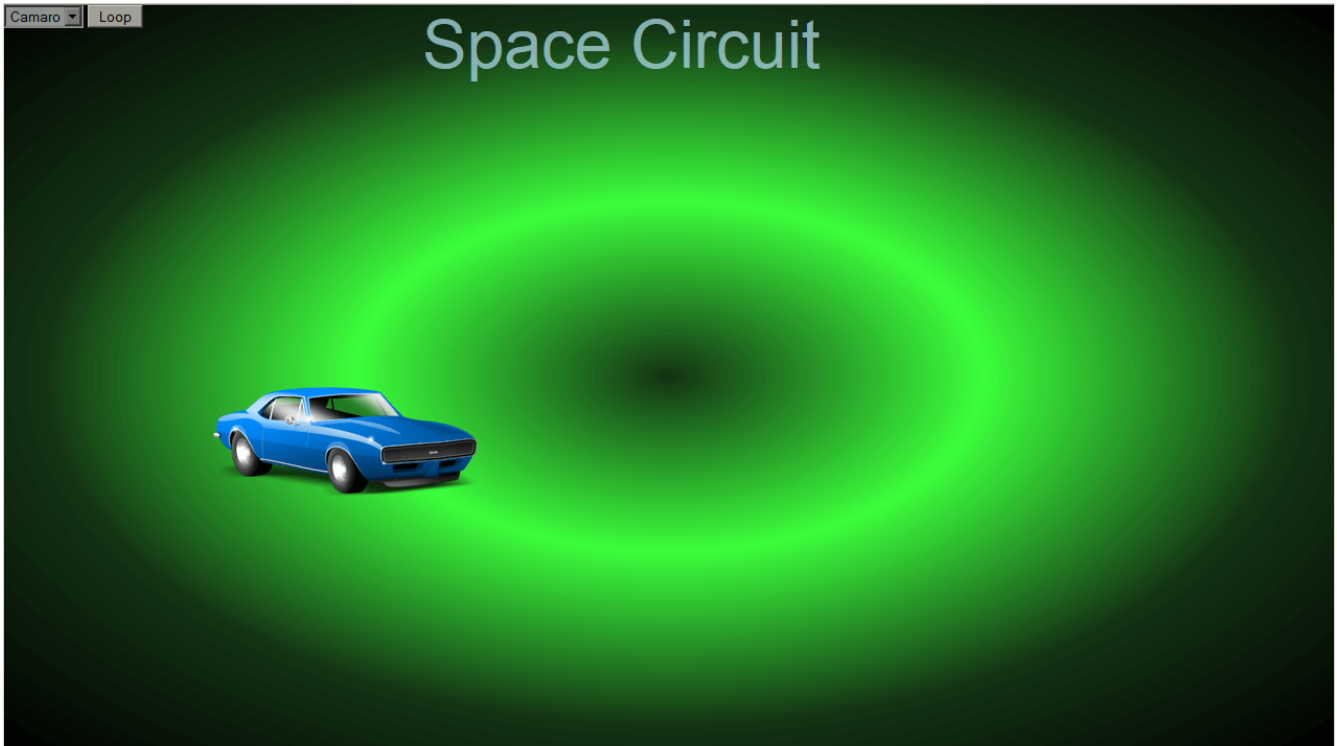
const moveDown = () => {
  const top = parseInt(spaceShip.style.top);
  const step = optionsArray[stepSelect.selectedIndex];

  if (top + step + spaceShipHeight > space.offsetHeight)
    spaceShip.style.top = space.offsetHeight - spaceShipHeight + "px";
  else spaceShip.style.top = top + step + "px";
};

addEventListener('load', init);
</script>
</head>
<body>
  <header>
    style="position: absolute; left: 0; top: 0; width: 100%; height: 100px;
    background-color: gold">
    <div style="position: absolute; left: 0; top: 0; width: 150px; height: 90px;">
      <img src=arrowLeft21x29.png onclick='moveLeft();' alt=arrowLeft21x29.png
        style="position: absolute; top: 30px; left: 5px;">
      <img src=arrowUp29x21.png onclick='moveUp();' alt=arrowUp29x21.png
        style="position: absolute; top: 5px; left: 30px;">
      <img src=arrowRight21x29.png onclick='moveRight();' alt=arrowRight21x29.png
        style="position: absolute; top: 30px; left: 63px;">
      <img src=arrowDown29x21.png onclick='moveDown();' alt=arrowDown29x21.png
        style="position: absolute; top: 63px; left: 30px">
    </div>
    <div style="position: absolute; left: 120px; top: 30px; height: 90px;">
      <label style="font-family: sans-serif;">Step (pixel): </label>
      <select style="background-color: gold;"></select>
    </div>
    <div style="position: absolute; top: 0; left: 400px; font-family: sans-serif;
      font-size: 400%">Space Ship
    </div>
  </header>
  <main style="position: absolute; left: 0; top: 90px; right: 0; bottom: 0;
    background: linear-gradient(to bottom right, gold, black) fixed;">
    <img id=spaceShip src=spaceship256x256.png
      style="position: absolute; left: 0; top: 0;" alt=spaceship256x256.png>
  </main>
</body>
</html>

```

## Space Circuit



Create the page shown at [youtu.be/axrTXBHDbXQ](https://youtu.be/axrTXBHDbXQ) taking the following into account:

1. Use the skeleton at [students.btsi.lu/evegi144/WAD/JS/Tests/SpaceCircuit/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/SpaceCircuit/index.html).
2. Create an empty array `pointXArray` as well as an array `pointYArray`. The latter gets filled with the values at the end of the skeleton.
3. Create function `fillXArray`, which does the following:
  - a. Define a variable `xOffset` with a random integer from [1, 400].
  - b. `pointXArray` is filled with values as follows:
    - i. Positions 0 to 9 get the value `xOffset + pos * 40`, with `pos` representing the position in the array.
    - ii. Positions 10 to 19 get the value `xOffset + 400 - (pos - 10) * 40`.
4. Clicking the button changes its text to "Stop" and calls `fillXArray`. The selected car will then run through the X- and Y-positions stored in the arrays, with a time interval of 100 ms. When the end of the array is reached, positions start again at the beginning.
5. Clicking the button again changes the text back to "Loop" and the car animation stops.
6. Clicking the button again ...

### Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/SpaceCircuit/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/SpaceCircuit/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Space Circuit</title>
```

```

<meta charset=UTF-8>
<script>
  'use strict';

  let car, carWidth = 256, carHeight = 256, loopButton, carSelect, pointXArray =
[];
  const pointYArray = [200, 160, 120, 80, 40, 0, 40, 80, 120, 160, 200, 240, 280,
    320, 360, 400, 360, 320, 280, 240];
  let idx = 0, arrayPoints = 20, timeout = 100, timerID;

  const init = () => {
    loopButton = document.querySelector('button');
    car = document.getElementById("car");
    carSelect = document.querySelector('select');
    selected();
  };

  const fillXArray = () => {
    const xOffset = Math.floor(Math.random() * 400) + 1;
    for (let i = 0; i < arrayPoints / 2; i++) pointXArray[i] = xOffset + i * 40;
    for (let i = arrayPoints / 2; i < arrayPoints; i++)
      pointXArray[i] = xOffset + 400 - (i - 10) * 40;
  };

  const selected = () => {
    car.src = carSelect.options[carSelect.selectedIndex].value;
    car.alt = carSelect.options[carSelect.selectedIndex].value;
  };

  const loop = () => {
    if (idx >= arrayPoints) idx = 0;
    car.style.left = pointXArray[idx] + "px";
    car.style.top = pointYArray[idx++] + "px";
  };

  const startLoop = () => {
    fillXArray();
    timerID = setInterval(loop, timeout);
    loopButton.innerHTML = "Stop";
    loopButton.onclick = stopLoop;
  };

  const stopLoop = () => {
    clearInterval(timerID);
    loopButton.innerHTML = "Loop";
    loopButton.onclick = startLoop;
  };

  addEventListener('load', init);
</script>
</head>

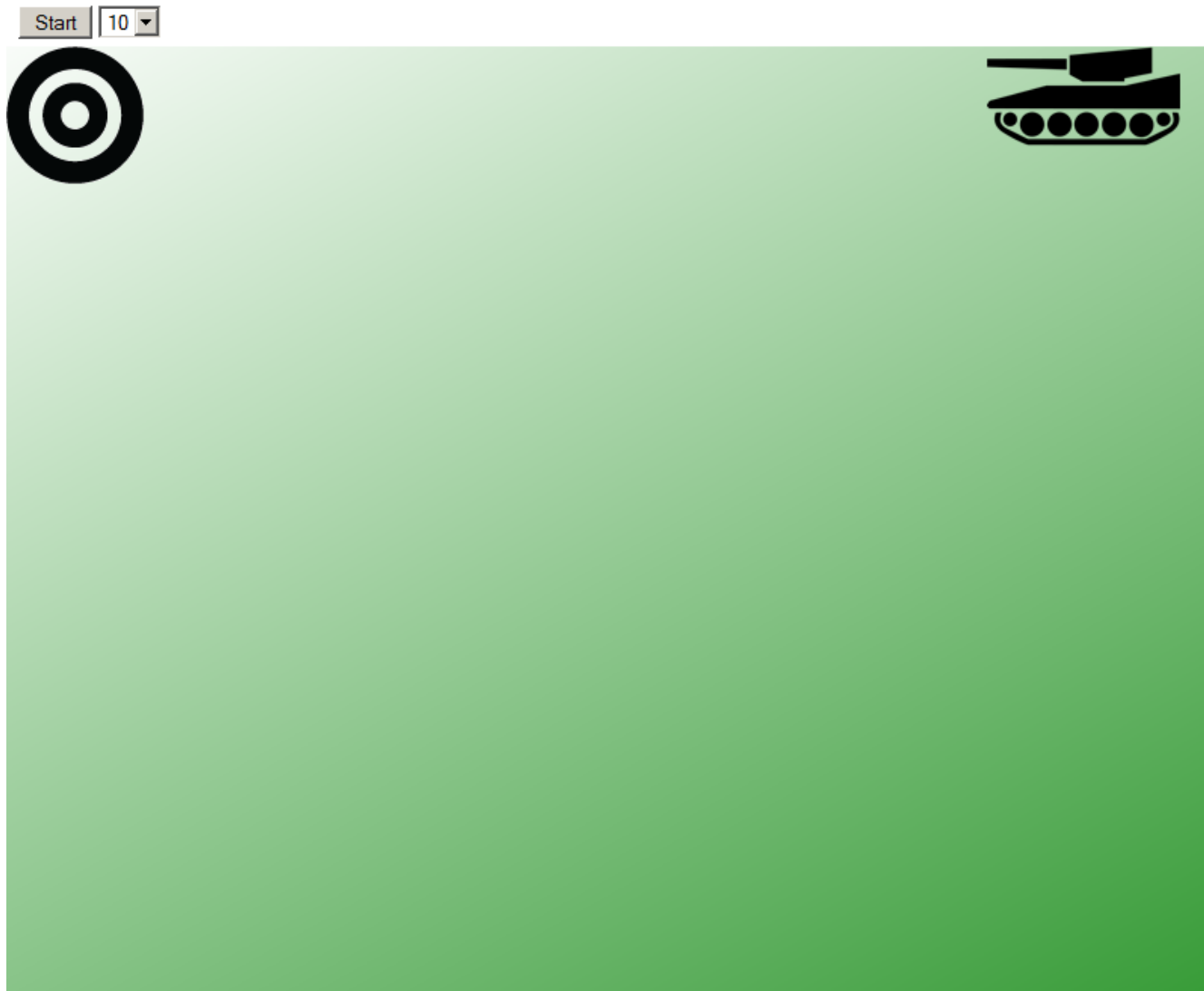
```

```

<body>
  <div style="position: absolute; left: 0; top: 0; width: 100%; height: 100%;
    background: radial-gradient(rgb(20, 50, 20), rgb(60, 255, 60), rgb(20,
50,20),
      black) fixed;">
    <div style="position: absolute; left: 0; top: 0; width: 100%; height: 20px;
      opacity: 0.75;">
      <div style="position: absolute; left: 0; top: 0; height: 90px;">
        <select style="opacity: 0.75;" onchange=selected();>
          <option value=camaro256x256.png>Camaro</option>
          <option value=ferrari256x256.png>Ferrari</option>
          <option value=cabrio256x256.png>Cabrio</option>
        </select>
        <button onclick=startLoop();>Loop</button>
      </div>
      <div style="position: absolute; top: 0; left: 400px; color: lightblue;
        font-family: sans-serif; font-size: 400%">Space Circuit
      </div>
    </div>
    <div style="position: absolute; left: 0; top: 90px; right: 0; bottom: 0;">
      <img id=car src=camaro256x256.png style="position: absolute; left: 200px;
        top: 200px;" alt=camaro256x256.png>
    </div>
  </div>
</body>
</html>

```

## Targeting Practice



## Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/TargetingPractice/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/TargetingPractice/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Targeting Practice</title>
    <meta charset=UTF-8>
    <script>
      'use strict';

      let timerID, timeout = 20, battleFieldHeight = 630;
      let yPosArray = [], arraySize = 100, currIdx = 0, currDirection = 1;
      let button, target, targetHeight = 91, tank, tankHeight = 66, select, tankStep;

      const init = () => {
        button = document.querySelector('button');
        target = document.getElementById("target");
        tank = document.getElementById("tank");
        select = document.querySelector("select");
        tankStep = parseInt(select.options[select.selectedIndex].text);
```

```

    for (let i = 0; i < arraySize; i++)
        yPosArray[i] = Math.floor(Math.random() * (5 * i + 1));
    addEventListener('keydown', keyHandler);
};

const selectChange = () => {
    tankStep = parseInt(select.options[select.selectedIndex].text);
    select.blur();
};

const moveTarget = () => {
    target.style.top = yPosArray[currIdx] + "px";
    if ((currIdx >= arraySize - 1) || (currIdx <= 0 && currDirection === -1))
        currDirection = -currDirection;
    currIdx += currDirection;
};

const start = () => {
    timerID = setInterval("moveTarget()", timeout);
    button.innerHTML = "Stop";
    button.onclick = stop;
};

const stop = () => {
    clearInterval(timerID);
    button.innerHTML = "Start";
    button.onclick = start;
};

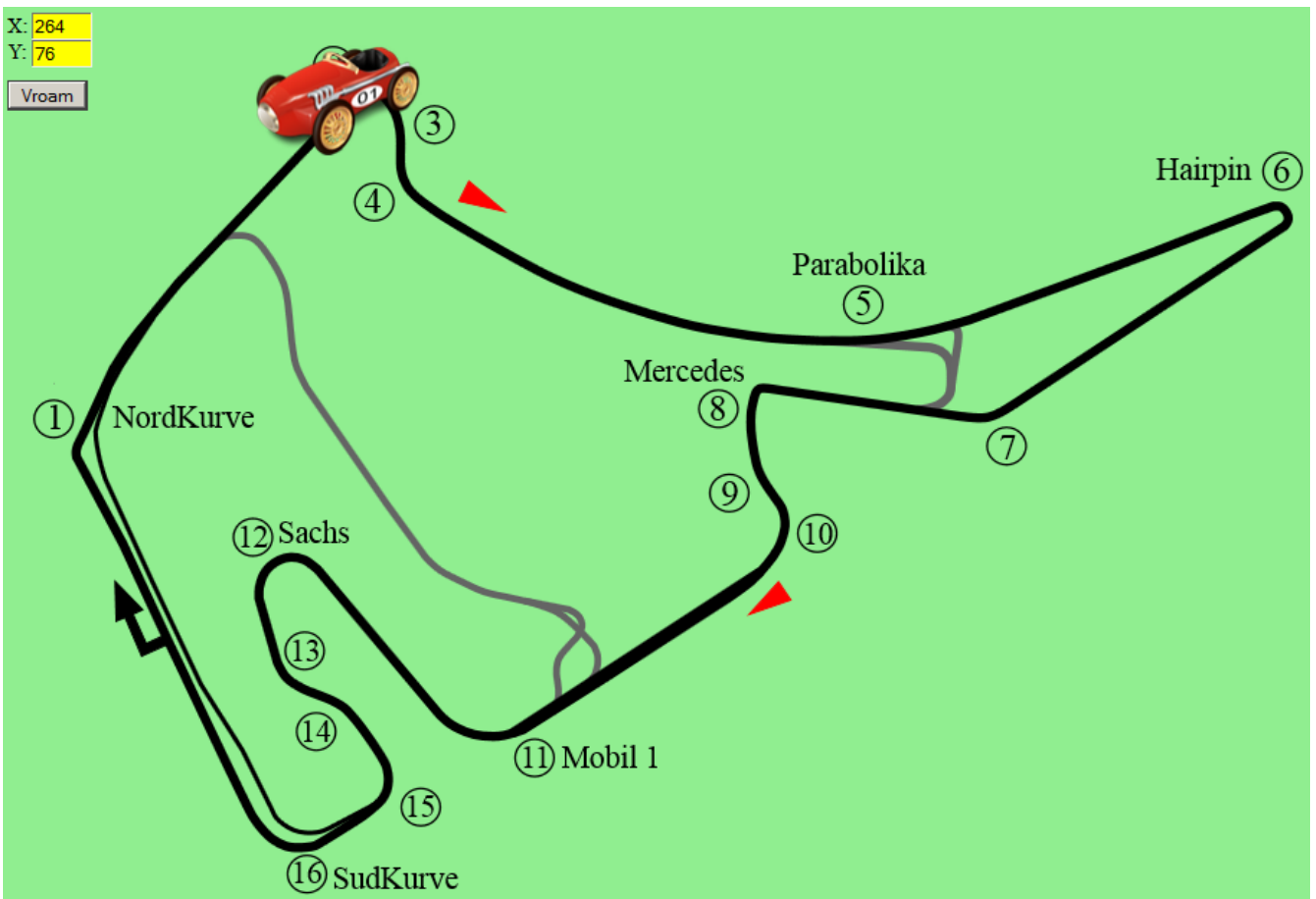
const keyHandler = event => {
    const top = parseInt(tank.style.top);
    if (event.keyCode === 38) // up
        if (top - tankStep < 0) tank.style.top = "0px";
        else tank.style.top = top - tankStep + "px";
    if (event.keyCode === 40) // down
        if (top + tankStep + tankHeight > battleFieldHeight)
            tank.style.top = battleFieldHeight - tankHeight + "px";
        else tank.style.top = top + tankStep + "px";
    if (event.keyCode === 70) // F for fire
        if ((parseInt(target.style.top) <= top) && (parseInt(target.style.top) +
            targetHeight >= top)) alert("Hit!");
};

addEventListener('load', init);
</script>
</head>
<body>
    <button onclick=start();>Start</button>
    <select onchange=selectChange();>
        <option>10</option>
        <option>25</option>

```

```
<option>50</option>
</select>
<div style="background: linear-gradient(to bottom right, white, green) fixed;
    position: absolute; top: 35px; left: 0; right: 0; bottom: 0; width:
800px;
    height: 630px;">
    <img id=tank src=tank128x66.png alt=tank128x66.png width=128 height=66
        style="position: absolute; left: 650px; top: 0;">
    <img id=target src=target91x91.png alt=target91x91.png width=91 height=91
        style="position: absolute; left: 0; top: 0;">
</div>
</body>
</html>
```

## Hockenheim Ring



### Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/HockenheimRing/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/HockenheimRing/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Hockenheim Ring</title>
    <meta charset=UTF-8>
    <style>
```

```

    label {
      float:      left;
      width:      15px;
      text-align: right;
      padding-right: 4px;
    }
  </style>
  <script>
    'use strict';

    let mouseXArray = [], mouseYArray = [], record = false, currCarXIdx = 0,
    currCarYIdx = 0;
    let timerID, button, car, name;

    const init = () => {
      button = document.getElementById("startStopButton");
      car = document.getElementById("car");
      name = prompt("Please enter your name:");
      alert(`Welcome ${name} to the Hockenheim Ring. Today is ` +
        `${(new Date()).toLocaleString()} Enjoy!`);
      addEventListener('keyup', handleKey);
    };

    const handleKey = event => {
      if (event.keyCode === 83) { // S key
        record = !record;
        if (record) document.onmousemove = getMouseXY;
        else document.onmousemove = null;
      }
    };

    const getMouseXY = e => {
      const mouseX = e.pageX;
      const mouseY = e.pageY;
      document.getElementById("mouseX").value = mouseX;
      document.getElementById("mouseY").value = mouseY;
      mouseXArray.push(mouseX);
      mouseYArray.push(mouseY);
    };

    const animate = () => {
      if (currCarXIdx < mouseXArray.length) {
        car.style.left = mouseXArray[currCarXIdx++] - 64 + "px";
        car.style.top = mouseYArray[currCarYIdx++] - 42 + "px";
      }
    };

    const startAnimation = () => {
      timerID = setInterval(animate, 10);
      button.innerHTML = "Stop";
      button.onclick = stopAnimation;
    };
  </script>

```

```

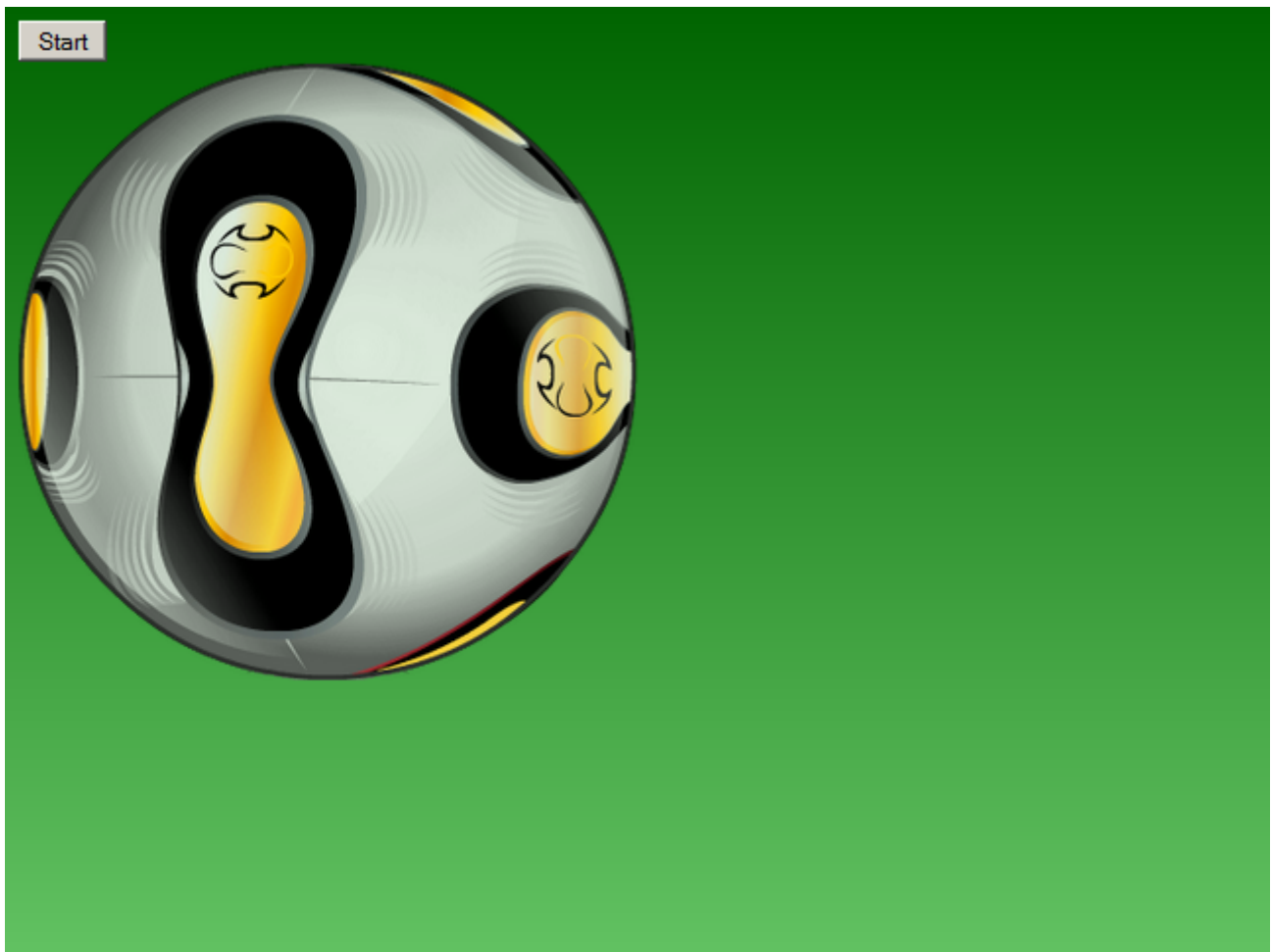
    };

    const stopAnimation = () => {
      clearInterval(timerID);
      button.innerHTML = "Vroam";
      button.onclick = startAnimation;
    };

    addEventListener('load', init);
  </script>
</head>
<body>
  style="background: url('Hockenheim20121052x744.svg') no-repeat;
  background-color: lightgreen;">
  <div>
    <label>X:</label>
    <input id=mouseX value=0 size=4 style="background-color: yellow" readonly>
  </div>
  <div>
    <label>Y:</label>
    <input id=mouseY value=0 size=4 style="background-color: yellow" readonly>
  </div>
  <button id=startStopButton onclick=startAnimation(); style="margin-top:
10px;">Vroam
  </button>
  <img id=car src=vroum128x84.png width=128 height=84 alt=vroum128x84.png
  style="position: absolute">
</body>
</html>

```

## Football Magic



Create a page ([youtu.be/v5zB0ecaCok](https://youtu.be/v5zB0ecaCok)) with a button and a football ([https://foxi.ltam.lu/PROF/evegi144/T2IF2\\_WSERS/WAD/JS/Tests/FootballMagic/football352x352.png](https://foxi.ltam.lu/PROF/evegi144/T2IF2_WSERS/WAD/JS/Tests/FootballMagic/football352x352.png)). The CSS for the body background is `background: linear-gradient(darkgreen, lightgreen) fixed;.`

After the document has loaded, an array with 10 randomly generated football positions (x from [0, 600[ and y from [0, 300[) is created.

When the button is clicked, it changes its text from 'Start' to 'Stop' and the ball jumps to the first position in the array, then after 20 ms to the second, after another 20 ms to the third etc. When the ball has reached the last array position, it moves to the first, then the second and so on.

When the button is clicked again, the animation stops and the button text changes back to 'Start'. Clicking it again resumes the animation from the position where it was stopped and changes the text to 'Stop'.

### Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/FootballMagic/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/FootballMagic/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Football Magic</title>
    <meta charset=utf-8>
```

```

<style>
  body {
    background: linear-gradient(darkgreen, lightgreen) fixed;
  }

  img {
    position: relative;
  }
</style>
<script>
  'use strict';

  let timerHandle, positions = [], currIdx = 0;

  const init = () => {
    for (let i = 0; i < 10; i++)
      positions.push([Math.floor(Math.random() * 600), Math.floor(Math.random() *
300)]);
    document.querySelector('button').addEventListener('click', buttonHandle);
  };

  const move = () => {
    const ball = document.querySelector('img');
    ball.style.left = positions[currIdx][0] + "px";
    ball.style.top = positions[currIdx][1] + "px";
    currIdx++;
    if (currIdx >= positions.length) currIdx = 0;
  };

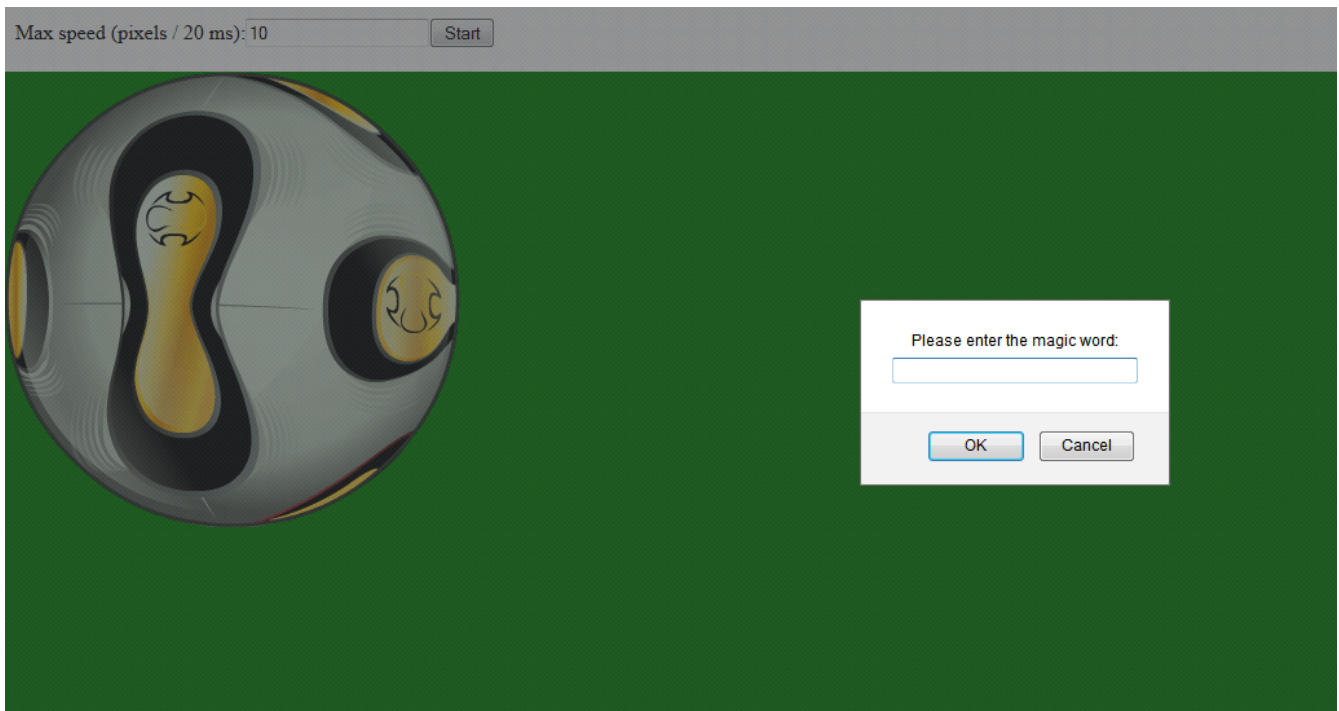
  const buttonHandle = () => {
    if (timerHandle) {
      clearInterval(timerHandle);
      timerHandle = undefined;
      document.querySelector('button').innerHTML = "Start";
    }
    else {
      timerHandle = setInterval(move, 20);
      document.querySelector('button').innerHTML = "Stop";
    }
  };

  addEventListener('load', init);
</script>
</head>
<body>
  <header>
    <button>Start</button>
  </header>
  <main>
    <img src=football352x352.png alt=football352x352>
  </main>

```

```
</body>
</html>
```

## Football Magic v2



Create the web page shown at [youtu.be/mRUPjjmpehM](https://youtu.be/mRUPjjmpehM) taking the following into account:

1. The football field has a green background, starts 50 pixels below the upper border and always uses the full window width as well as the remaining window height (i.e. height - 50). To achieve this you need to style the top, right, bottom and left distances of the football field. You can use the `offsetWidth` and `offsetHeight` attributes of the football field.
2. The header includes a `label`, an `input` as well as two `button` elements, only one of which is visible at any point in time.
3. After the start button has been clicked, the ball moves every 20 ms a certain amount of pixels both horizontally and vertically. The pixel number is calculated as a random number between 1 and the value the user has specified in the `input`. If the ball were to move beyond any of the borders, it will be placed at this border and the corresponding direction (vertical or horizontal) changed.
4. After the page has loaded, the user is asked for the password `CLISS1` using the text "Please enter the magic word.". The password will be asked repeatedly until the user enters the correct one. Only thereafter can the animation be started.
5. Your page must pass the HTML5-validator without errors.

### Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/FootballMagicV2/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/FootballMagicV2/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
```

```

<head>
  <title>Football Magic v2</title>
  <meta charset=UTF-8>
  <script>
    'use strict';

    let football, directionX = 1, directionY = 1, timerID, timeout = 20;
    let footballField, footballWidth = 352, footballHeight = 352;

    const init = () => {
      football = document.querySelector('img');
      footballField = document.querySelector('main');
      document.getElementById("startButton").addEventListener('click', start);
      document.getElementById("stopButton").addEventListener('click', stop);
      checkPassword();
    };

    const checkPassword = () => {
      const text = "Please enter the magic word:";
      let password = prompt(text);
      while (password !== "CLISS1") password = prompt(text);
    };

    const timer = () => {
      const maxStep = document.querySelector('input').value;
      const step = Math.floor(Math.random() * maxStep) + 1;
      const left = parseInt(football.style.left), top =
parseInt(football.style.top);

      if (left + step * directionX < 0) {
        football.style.left = "0";
        directionX = 1;
      }
      else if (left + step * directionX + footballWidth > footballField.offsetWidth)
{
        football.style.left = footballField.offsetWidth - footballWidth + "px";
        directionX = -1;
      }
      else football.style.left = left + step * directionX + "px";
      if (top + step * directionY < 0) {
        football.style.top = "0";
        directionY = 1;
      }
      else if (top + step * directionY + footballHeight >
footballField.offsetHeight) {
        football.style.top = footballField.offsetHeight - footballHeight + "px";
        directionY = -1;
      }
      else football.style.top = top + step * directionY + "px";
    };
  
```

```

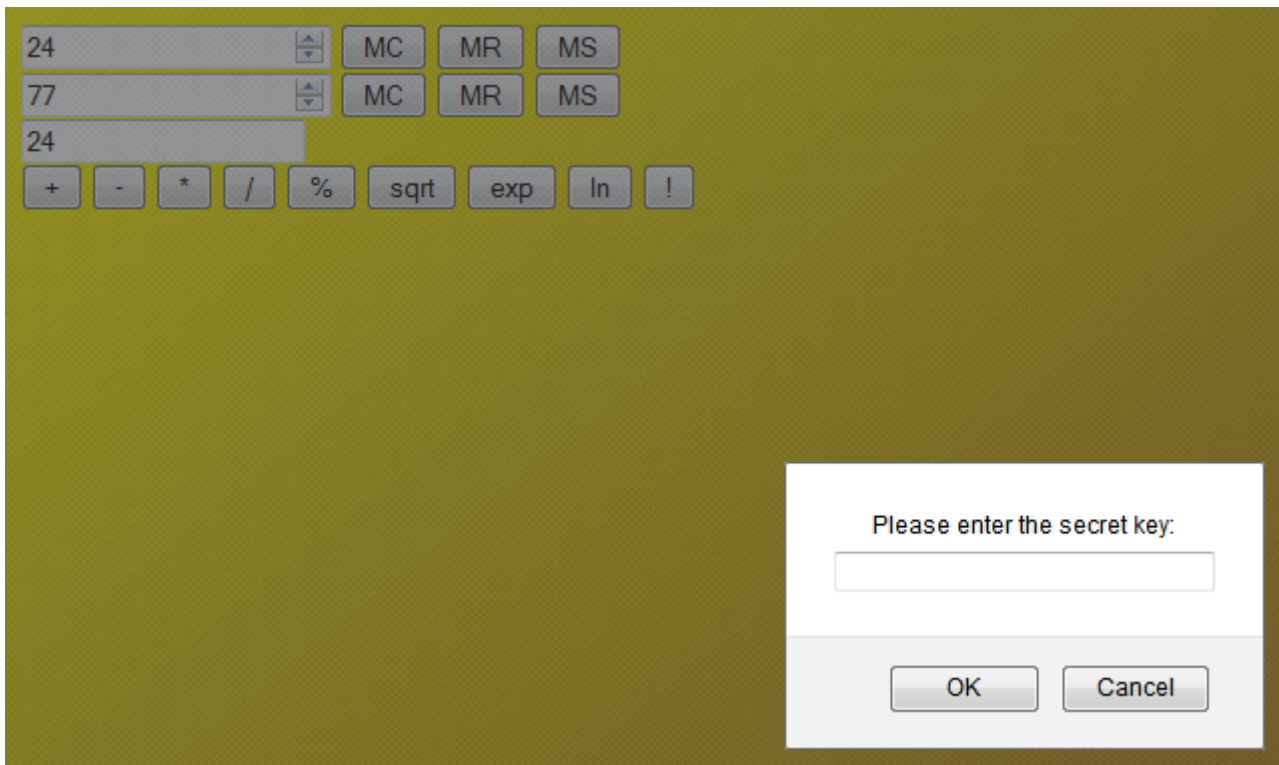
const start = () => {
  document.getElementById("startButton").style.display = "none";
  document.getElementById("stopButton").style.display = "inline";
  timerID = setInterval(timer, timeout);
};

const stop = () => {
  document.getElementById("startButton").style.display = "inline";
  document.getElementById("stopButton").style.display = "none";
  clearInterval(timerID);
};

addEventListener('load', init);
</script>
</head>
<body>
  <header>
    <label for=inp>Max speed (pixels / 20 ms):</label>
    <input id=inp value=10>
    <button id=startButton>Start</button>
    <button id=stopButton style="display: none">Stop</button>
  </header>
  <main style="background-color: green; left: 0; top: 50px; right: 0; bottom: 0;
    position: absolute">
    <img src=football352x352.png alt=football352x352.png width=352 height=352
      style="position: absolute; left: 0; top: 0;">
  </main>
</body>
</html>

```

## Calculator



Create the web page shown at [youtu.be/Fw7PvkvrYss](https://youtu.be/Fw7PvkvrYss) taking the following into account:

1. Use the skeleton at [students.btsi.lu/evegi144/WAD/JS/Tests/Calculator/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/Calculator/index.html).
2. **MC** stands for memory clear, **MR** for memory read and **MS** for memory store. The calculator has two independent memories with initial value 0.
3. The binary operators **+**, **-**, **\*** and **%** work with the two upper inputs fields and write the result into the lower text field. The unary operators uses the upper input field and writes the result there too.
4. The factorial function checks whether the value of the input value is larger than 1000. If so, nothing happens.
5. After the page has loaded, the user is asked for the password **CLISS1** using the text "Please enter the magic word.". The password will be asked up to three times. If after three attempts the user still has not entered the correct password, all document elements will be deleted using `document.body.removeChild(document.querySelector('main'));`.

### Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/Calculator/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/Calculator/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Calculator</title>
    <meta charset=UTF-8>
    <style>
      body {
        background: linear-gradient(to bottom right, yellow, #772222) fixed;
      }
    </style>
  </head>
  <body>
    <div>
      <input type="text" value="24"/>
      <input type="text" value="77"/>
      <input type="text" value="24"/>
      <div>
        <button>MC</button>
        <button>MR</button>
        <button>MS</button>
      </div>
      <div>
        <button>+</button>
        <button>-</button>
        <button>*</button>
        <button>/</button>
        <button>%</button>
        <button>sqrt</button>
        <button>exp</button>
        <button>ln</button>
        <button>!</button>
      </div>
    </div>
  </body>
</html>
```

```
</style>
<script>
  'use strict';

  let memoryA = 0, memoryB = 0, inputA, inputB, result;

  const init = () => {
    inputA = document.getElementById("inputA");
    inputB = document.getElementById("inputB");
    result = document.getElementById("result");
    document.getElementById("b1").onclick = memoryAClear;
    document.getElementById("b2").onclick = memoryARead;
    document.getElementById("b3").onclick = memoryAStore;
    document.getElementById("b4").onclick = memoryBClear;
    document.getElementById("b5").onclick = memoryBRead;
    document.getElementById("b6").onclick = memoryBStore;
    document.getElementById("b7").onclick = add;
    document.getElementById("b8").onclick = subtract;
    document.getElementById("b9").onclick = multiply;
    document.getElementById("b10").onclick = divide;
    document.getElementById("b11").onclick = modulo;
    document.getElementById("b12").onclick = squareRoot;
    document.getElementById("b13").onclick = exponential;
    document.getElementById("b14").onclick = naturalLog;
    document.getElementById("b15").onclick = factorial;
    loginCheck();
  };

  const memoryAClear = () => {
    memoryA = 0;
  };

  const memoryARead = () => {
    inputA.value = memoryA;
  };

  const memoryAStore = () => {
    memoryA = inputA.value;
  };

  const memoryBClear = () => {
    memoryB = 0;
  };

  const memoryBRead = () => {
    inputB.value = memoryB;
  };

  const memoryBStore = () => {
    memoryB = inputB.value;
  };

```

```
const add = () => {
  result.value = Number(inputA.value) + Number(inputB.value);
};

const subtract = () => {
  result.value = inputA.value - inputB.value;
};

const multiply = () => {
  result.value = inputA.value * inputB.value;
};

const divide = () => {
  result.value = inputA.value / inputB.value;
};

const modulo = () => {
  result.value = inputA.value % inputB.value;
};

const squareRoot = () => {
  inputA.value = Math.sqrt(inputA.value);
};

const exponential = () => {
  inputA.value = Math.exp(inputA.value);
};

const naturalLog = () => {
  inputA.value = Math.log(inputA.value);
};

const factorial = () => {
  let fact = 1;
  if (inputA.value >= 1000) return;
  for (let i = 2; i <= inputA.value; i++) fact *= i;
  inputA.value = fact;
};

const loginCheck = () => {
  let password = "", counter = 1, limit = 3;
  while (password !== "CLISS1" && counter <= limit) {
    password = prompt("Please enter the secret key:");
    counter++;
  }
  if (password !== "CLISS1") {
    document.body.removeChild(document.querySelector('main'));
  }
};
```

```
        addEventListener('load', init);
    </script>
</head>
<body>
    <main>
        <input id=inputA type=number>
        <button id=b1>MC</button>
        <button id=b2>MR</button>
        <button id=b3>MS</button>
        <br>
        <input id=inputB type=number>
        <button id=b4>MC</button>
        <button id=b5>MR</button>
        <button id=b6>MS</button>
        <br>
        <input id=result readonly>
        <br>
        <button id=b7>+</button>
        <button id=b8>-</button>
        <button id=b9>*</button>
        <button id=b10>/</button>
        <button id=b11>%</button>
        <button id=b12>sqrt</button>
        <button id=b13>exp</button>
        <button id=b14>ln</button>
        <button id=b15>!</button>
    </main>
</body>
</html>
```

## Space Clock



8:48:49

Create the web page shown at [youtu.be/iNUDJ6zVhYs](https://youtu.be/iNUDJ6zVhYs) taking the following into account:

1. Part 1

- Use the skeleton at [students.btsi.lu/evegi144/WAD/JS/Tests/SpaceClock/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/SpaceClock/index.html).
- Define a two-dimensional array `pixelColorArray`.
- Write function `fillColorArray`, which fills the array with the RGB value (`"rgb(red part, green part, blue part)"`) for each pixel of a 100 x 100 pixel square. The value for each pixel is calculated like this: red and blue part:  $i + j$ , green part:  $255 - (i + j)$ , with  $i$  and  $j$  representing the horizontal and vertical pixel position.
- Write function `drawColorArray`, which draws every pixel with the saved color. Use function `draw`, which is already in the code.
- Execute the two functions and make sure the result corresponds to what you see in the video.

2. Part 2

Create a copy of `findFirstPos` under the name `fastFindFirstPos` and optimize it in terms of number of variables, instructions and iterations used. To verify your success, some sample instructions have already been created that you can use and extend. . Part 3

Below commentary **K3 Debugging** you find 7 lines of buggy JavaScript. Copy and correct them so that

the clock is displayed correctly, as shown in the video, without error messages in the console. All variables need to be declared locally.

### Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/SpaceClock/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/SpaceClock/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Space Clock</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <div style="position: absolute; left: 0; top: 0; width: 100%; height: 100%;
      background: radial-gradient(rgb(20, 50, 20), rgb(60, 255, 60), rgb(20,
50,20),
        black) fixed;">
      <canvas width=150 height=150>Your Browser does not support canvas!</canvas>
      <div id=debugMeDiv style="color: white; font-size: 500%"></div>
    </div>
    <script>
      'use strict';

      // K2 komplexe Verschachtelungen und 2-dimensionale Felder
      const canvas = document.querySelector("canvas"), context =
canvas.getContext("2d");
      const pixelColorArray = [], pixelColorArraySize = 100;
      let numIterations = 0;

      const fillColorArray = () => {
        for (let i = 0; i < pixelColorArraySize; i++) {
          pixelColorArray[i] = [];
          for (let j = 0; j < pixelColorArraySize; j++)
            pixelColorArray[i][j] = `rgb(${i + j},${255 - (i + j)},${i + j})`;
        }
      };

      const draw = (x, y, color) => {
        context.fillStyle = color;
        context.fillRect(x, y, 1, 1);
      };

      const drawColorArray = () => {
        for (let i = 0; i < pixelColorArraySize; i++)
          for (let j = 0; j < pixelColorArraySize; j++) draw(i, j,
pixelColorArray[i][j]);
      };

      fillColorArray();
      drawColorArray();
```

```

// K2 Optimierung von Skripten
const findFirstPos = (arr, x) => { // arr is an array of numbers
  let pos = -1;
  numIterations = 0;
  if (typeof arr === 'undefined') return -1;
  if (typeof arr !== 'undefined')
    for (let i = 0; i < arr.length; i++) {
      if (arr[i] === x && pos === -1) pos = i;
      numIterations++;
    }
  return pos;
};
// Solution
const fastFindFirstPos = (arr, x) => { // arr is an array of numbers
  numIterations = 0;
  if (typeof arr !== 'undefined')
    for (let i = 0; i < arr.length; i++) {
      numIterations++;
      if (arr[i] === x) return i
    }
  return -1;
};
const arr = [1, 2, 3, 3, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 3];
console.log(`Number of iterations before: ${numIterations}`);
console.log(`findFirstPos: ${findFirstPos(arr, 3)}`);
console.log(`Number of iterations after: ${numIterations}`);
numIterations = 0;
console.log(`Number of iterations before: ${numIterations}`);
console.log(`fastFindFirstPos: ${fastFindFirstPos(arr, 3)}`);
console.log(`Number of iterations after: ${numIterations}`);

// K3 Debugging
/*
function debugMe1 i if i < 10 i = 0 + i retourn i
function debugMe2 today == Date() h === today.getHours()
m === today.getMinutes() s === today.getSeconds
m === debugMe1 m s === debugMe1 s
getElementById("debugMeDiv").HTML == h + : + s;
function debugMe3 setInterval debugMe2 1000
debug
*/
// Solution
function debugMe1(i) {
  if (i < 10) i = "0" + i;
  return i;
}

function debugMe2() {
  var today = new Date();
  var h = today.getHours();

```

```

    var m = today.getMinutes();
    var s = today.getSeconds();
    m = debugMe1(m);
    s = debugMe1(s);
    document.getElementById("debugMeDiv").innerHTML = h + ":" + m + ":" + s;
}

function debugMe3() {
    setInterval(debugMe2, 1000);
}
debugMe3();
</script>
</body>
</html>

```

## Dog Race

Go!



Create the web page shown at [youtu.be/\\_WrlTFUyMhA](https://youtu.be/_WrlTFUyMhA) taking the following into account:

1. Use the skeleton at [foxi.ltam.lu/PROF/evegi144/T2IF2\\_WSERS/WAD/JS/Tests/DogRace/index.html](https://foxi.ltam.lu/PROF/evegi144/T2IF2_WSERS/WAD/JS/Tests/DogRace/index.html).
2. Clicking the button changes its label to "Stop!" and starts the dog race. Clicking the button again changes its label back to "Go!" and halts the race. Clicking it again changes the label and resumes the race etc.
3. Every 20 ms each dog is moved a random number of pixels from [0, 5]. The dog that completely disappears from the screen first wins. If the two dogs disappear at the same time, the message "Draw!" is displayed.
4. You can use the attribute `offsetWidth` to determine the width of the element that contains the two dogs.

## Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/DogRace/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/DogRace/Solution/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Dog Race</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      let timerId, img1, img2, button, main;

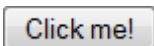
      const init = () => {
        main = document.querySelector('main');
        button = document.querySelector('button');
        button.addEventListener('click', buttonHandler);
        img1 = document.getElementById('img1');
        img2 = document.getElementById('img2');
        img1.style.cssText = "position: absolute; top: 100px; left: 0";
        img2.style.cssText = "position: absolute; top: 250px; left: 0";
      };

      const run = () => {
        const step1 = Math.floor(Math.random() * 6), step2 = Math.floor(Math.random()
* 6);
        const left1 = parseInt(img1.style.left), left2 = parseInt(img2.style.left);
        img1.style.left = left1 + step1 + "px";
        img2.style.left = left2 + step2 + "px";
        const dist1 = main.offsetWidth - (left1 + step1);
        const dist2 = main.offsetWidth - (left2 + step2);
        if (dist1 <= 0 || dist2 <= 0) {
          if (dist1 < dist2) alert('Dog 1 won!');
          else if (dist2 < dist1) alert('Dog 2 won!');
          else alert('Draw!');
          buttonHandler();
          img1.style.left = "0px";
          img2.style.left = "0px";
        }
      };

      const buttonHandler = () => {
        if (timerId) {
          button.innerHTML = 'Go!';
          clearInterval(timerId);
          timerId = undefined;
        }
        else {
          button.innerHTML = 'Stop!';
          timerId = setInterval(run, 20);
        }
      };
    </script>
  </head>
  <body>
    <main>
      <img alt="Dog 1" id="img1" data-bbox="100 100 200 200" data-cs="2" data-kind="parent"/>
      </main>
      <img alt="Dog 2" id="img2" data-bbox="100 250 200 350" data-cs="2" data-kind="parent"/>
      </img>
      <button id="button" data-bbox="100 400 200 450">Go!</button>
    </body>
  </html>
```

```
    }  
    };  
  
    addEventListener('load', init);  
</script>  
</head>  
<body>  
  <header>  
    <button>Go!</button>  
  </header>  
  <main>  
    <img id=img1 src=1422471521_robotic_pet.png width=128 height=128 alt=Dog1>  
    <img id=img2 src=1422471561_robotic_pet.png width=128 height=128 alt=Dog2>  
  </main>  
</body>  
</html>
```

## Crazy Button



Create the web page shown at [youtu.be/z2tDCjzZH3Y](https://youtu.be/z2tDCjzZH3Y) taking the following into account:

1. The button initially is labeled "Click me!".
2. Create an array containing 10 random numbers from [0, 1000].
3. With each click on the button, its label changes to the next random number from the array. When the end of the array has been reached, the labeling restarts with the first array element.
4. The button can be moved in steps of 10 pixels using the cursor keys.

## Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/CrazyButton/Solution/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/CrazyButton/Solution/index.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Crazy Button</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const LEFT = 37, UP = 38, RIGHT = 39, DOWN = 40;
      let button, randomValues = [], buttonValueIndex = 0;

      const buttonHandler = () => {
        button.innerHTML = randomValues[buttonValueIndex++];
        if (buttonValueIndex >= 10) buttonValueIndex = 0;
      };

      const keyHandler = event => {
        if (event.keyCode === LEFT)
          button.style.left = parseInt(button.style.left) - 10 + "px";
        else if (event.keyCode === UP)
          button.style.top = parseInt(button.style.top) - 10 + "px";
        else if (event.keyCode === RIGHT)
          button.style.left = parseInt(button.style.left) + 10 + "px";
        else if (event.keyCode === DOWN)
          button.style.top = parseInt(button.style.top) + 10 + "px";
      };

      const init = () => {
        button = document.querySelector('button');
        button.style.cssText = "position: absolute; left: 0; top: 0;";
        for (var i = 0; i < 10; i++) randomValues.push(Math.floor(Math.random() *
1001));
        button.addEventListener('click', buttonHandler);
        addEventListener('keydown', keyHandler);
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <main>
      <button>Click me!</button>
    </main>
  </body>
</html>

```

## MicroJSON

<input type="text" value="Mickey"/>	<input type="text" value="Mouse"/>	<input type="button" value="Send"/>
-------------------------------------	------------------------------------	-------------------------------------

Fist name: Bill last name: Gates random number: 1041318109

Fist name: Bill last name: Gates random number: 515018495

Fist name: Mickey last name: Mouse random number: 725358901

Fist name: Mickey last name: Mouse random number: 1348121517

Fist name: Mickey last name: Mouse random number: 1632377915

Write a **validated single file** app that does the following ([youtu.be/1nFOE5cTZrU](https://youtu.be/1nFOE5cTZrU)) **without any page reload**:

1. The user can enter a first name and last name. This data is sent to the server using JSON.
2. The server adds a random number to the data received from the client and sends the whole data set (i.e. first name, last name and random number) to the client using JSON.
3. The client displays the three data items.

#### **Solution**

[students.btsi.lu/evegi144/WAD/JS/Tests/MicroJSON/index.php](https://students.btsi.lu/evegi144/WAD/JS/Tests/MicroJSON/index.php)

```

<?php
    $arr = json_decode(file_get_contents('php://input'));
    if ($arr) {
        $arr[] = rand();
        echo json_encode($arr);
        exit;
    }
?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>MicroJSON</title>
        <meta charset=utf-8>
        <script>
            'use strict';

            const displayData = e => {
                const data = JSON.parse(e.target.response);
                const p = document.createElement('p');
                p.innerHTML = `Fist name: ${data[0]} last name: ${data[1]} random number:
${data[2]}`;
                document.body.appendChild(p);
            };

            const init = () => {
                document.forms[0].addEventListener('submit', e => {
                    e.preventDefault();
                    const req = new XMLHttpRequest();
                    const fn = document.forms[0][0].value;
                    const ln = document.forms[0][1].value;
                    req.addEventListener('load', displayData);
                    req.open('POST', 'index.php');
                    req.send(JSON.stringify([fn, ln]));
                });
            };

            addEventListener('load', init);
        </script>
    </head>
    <body>
        <form>
            <input name=first_name placeholder="first name" required autofocus>
            <input name=last_name placeholder="last name" required>
            <button>Send</button>
        </form>
    </body>
</html>

```

Or using jQuery:

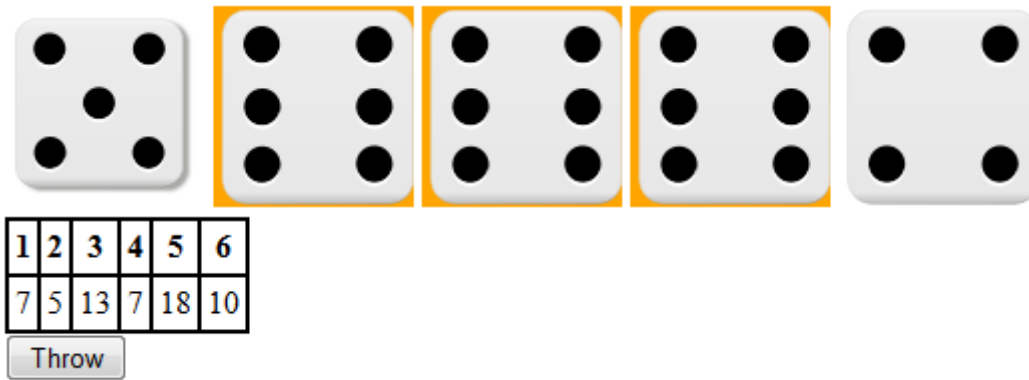
```
<?php
$arr = json_decode(file_get_contents('php://input'));
if ($arr) {
    $arr[] = rand();
    echo json_encode($arr);
    exit;
}
?>
<!DOCTYPE html>
<html lang=en>
<head>
    <title>MicroJSON</title>
    <meta charset=utf-8>
    <script src=//code.jquery.com/jquery-2.1.4.min.js></script>
    <script>
        'use strict';

        const displayData = e => {
            const p = document.createElement('p');
            p.innerHTML = `Fist name: ${e[0]} last name: ${e[1]} random number: ${e[2]}`;
            document.body.appendChild(p);
        };

        const init = () => {
            document.forms[0].addEventListener('submit', e => {
                e.preventDefault();
                const fn = document.forms[0][0].value;
                const ln = document.forms[0][1].value;
                $.ajax({
                    type: "post",
                    dataType: "json",
                    data: JSON.stringify([fn, ln]),
                    success: displayData
                });
            });
        };

        addEventListener('load', init);
    </script>
</head>
<body>
    <form>
        <input name=first_name placeholder="first name" required autofocus>
        <input name=last_name placeholder="last name" required>
        <button>Send</button>
    </form>
</body>
</html>
```

## Dice



Create a dice simulator with 5 dice (cf. [youtu.be/oH6I0o3gFzk](https://youtu.be/oH6I0o3gFzk)).

Upon entering the simulator, the user sees 5 randomly thrown dice as well as a table displaying how often each one of the values 1 to 6 has been thrown so far.

Below the table is a button, which allows to throw the dice again and automatically update the statistics.

When a value appears more than once, all instances are highlighted via CSS.

For the pros: simulate some of the Yahtzee rules ([en.wikipedia.org/wiki/Yahtzee](https://en.wikipedia.org/wiki/Yahtzee)).

### Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/Dice/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/Dice/index.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Dice</title>
    <meta charset=utf-8>
    <style>
      table {
        border-collapse: collapse;
      }
      td, th {
        border: 2px solid black;
        padding: 3px;
      }
    </style>
    <script>
      'use strict';

      const imgNames = ['one.png', 'two.png', 'three.png', 'four.png', 'five.png',
'six.png'];
      const currentThrow = [0, 0, 0, 0, 0], stats = [0, 0, 0, 0, 0, 0];
      const colors = ['red', 'green', 'blue', 'gold', 'pink', 'orange'];
```

```

const throwDice = () => {
  for (let i = 0; i < 5; i++) {
    const num = Math.floor(Math.random() * 6);
    document.getElementById('d' + (i + 1)).src = imgNames[num];
    document.getElementById('d' + (i + 1)).style.backgroundColor = 'white';
    currentThrow[i] = num;
    stats[num]++;
  }
  for (let i = 0; i < 4; i++)
    for (let j = i + 1; j < 5; j++)
      if (currentThrow[i] === currentThrow[j]) {
        const color = colors[currentThrow[i]];
        document.getElementById('d' + (i + 1)).style.backgroundColor = color;
        document.getElementById('d' + (j + 1)).style.backgroundColor = color;
      }
  displayStats();
};

const displayStats = () => {
  const tds = document.querySelectorAll('td');
  for (let i = 0; i < 6; i++) tds[i].innerHTML = stats[i];
};

const init = () => {
  document.querySelector('button').addEventListener('click', throwDice);
  throwDice();
};

addEventListener('load', init);
</script>
</head>
<body>
  <img src=one.png id=d1 alt=d1>
  <img src=one.png id=d2 alt=d2>
  <img src=one.png id=d3 alt=d3>
  <img src=one.png id=d4 alt=d4>
  <img src=one.png id=d5 alt=d5>
  <table>
    <thead>
      <tr>
        <th>1</th>
        <th>2</th>
        <th>3</th>
        <th>4</th>
        <th>5</th>
        <th>6</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td></td>

```

```

        <td></td>
        <td></td>
        <td></td>
        <td></td>
        <td></td>
    </tr>
</tbody>
</table>
<button>Throw</button>
</body>
</html>

```

## Test Stats

Create a web app that displays basic test statistics (cf. [youtu.be/\\_IXETf2fYts](https://youtu.be/_IXETf2fYts)).

The input fields are configured so that they display values outside of [1, 60] as invalid.

Your JavaScript may not produce an error irrespective of the user input.

## Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/TestStats/index.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/TestStats/index.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Test Stats</title>
    <meta charset=utf-8>
    <style>
      table {
        border-collapse: collapse;
      }

      td, th {
        border: 2px solid black;
        padding: 3px;
      }
    </style>
    <script>
      'use strict';

      const calc = () => {
        const inputs = document.querySelectorAll('input');
        let min = 61, max = 0, avg = 0, val;
        for (let i = 0; i < inputs.length; i++) {
          val = parseInt(inputs[i].value);
          if (isNaN(val) || val > 60 || val < 1) {
            alert('Invalid input!');
            return;
          }
        }
      }
    </script>
  </head>
  <body>
    <table>
      <tr>
        <td><input type="text" value="1" />
        <td><input type="text" value="1" />
        <td><input type="text" value="1" />
        <td><input type="text" value="1" />
        <td><input type="text" value="1" />
      </tr>
    </table>
    <button>Throw</button>
  </body>
</html>

```

```

    }
    if (val < min) min = val;
    if (val > max) max = val;
    avg += val;
  }
  avg = Math.round(100 * avg / inputs.length) / 100; // Optional rounding to 2
digits.
  const tds = document.querySelectorAll('td');
  tds[0].innerHTML = min;
  tds[1].innerHTML = max;
  tds[2].innerHTML = avg;
};

// Optional: allow only digits from 0 to 9 as well as cursor left and right, INS
and DEL.
const isNum = evt => {
  const c = evt.keyCode;
  return (c === 8 || c === 9 || c === 46 || c === 37 || c === 39 || c >= 48 && c
<= 57);
};

addEventListener('load', () =>
  document.querySelector('button').addEventListener('click', calc));
</script>
</head>
<body>
  <main>
    <input type=number min=1 max=60 placeholder=grade1 onkeydown="return
isNum(event)">
    <input type=number min=1 max=60 placeholder=grade2 onkeydown="return
isNum(event)">
    <input type=number min=1 max=60 placeholder=grade3 onkeydown="return
isNum(event)">
    <button>Calc stats</button>
    <table>
      <tr>
        <th>Min</th>
        <th>Max</th>
        <th>Avg</th>
      </tr>
      <tr>
        <td></td>
        <td></td>
        <td></td>
      </tr>
    </table>
  </main>
</body>
</html>

```

## Picture Viewer

Create the picture viewer shown at [youtu.be/eZrlapASlIU](https://youtu.be/eZrlapASlIU).

All HTML elements inside the body as well as all CSS styling must be created in JavaScript.

The picture filenames are stored in an array, so that the application can handle any number of images. The pictures can be downloaded from [students.btsi.lu/evegi144/WAD/JS/Tests/PicViewer](https://students.btsi.lu/evegi144/WAD/JS/Tests/PicViewer). The background is a linear gradient to the bottom right from gold to black.

### Solution

[students.btsi.lu/evegi144/WAD/JS/Tests/PicViewer/picviewer.html](https://students.btsi.lu/evegi144/WAD/JS/Tests/PicViewer/picviewer.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Picture Viewer</title>
    <meta charset=utf-8>
    <script>
      'use strict';

      const images = ['camaro256x256.png', 'ferrari256x256.png',
        'football352x352.png'];

      const init = () => {
        const select = document.createElement('select');
        for (let i = 0; i < images.length; i++) {
          const option = document.createElement('option');
          option.innerHTML = images[i];
          select.appendChild(option);
        }
        select.addEventListener('change', displayPic);
        const leftButton = document.createElement('button');
        leftButton.innerHTML = '<';
        leftButton.addEventListener('click', left);
        const rightButton = document.createElement('button');
        rightButton.innerHTML = '>';
        rightButton.addEventListener('click', right);
        document.body.appendChild(select);
        document.body.appendChild(leftButton);
        document.body.appendChild(rightButton);
        document.body.appendChild(document.createElement('br'));
        displayPic();
        document.body.style.background = 'linear-gradient(to bottom right, gold,
black) fixed';
      };

      const displayPic = () => {
        let img = document.querySelector('img');
        if (!img) {
```

```

        img = document.createElement('img');
        document.body.appendChild(img);
    }
    const idx = document.querySelector('select').selectedIndex;
    img.src = images[idx];
};

const left = () => {
    if (document.querySelector('select').selectedIndex > 0)
        document.querySelector('select').selectedIndex--;
    else document.querySelector('select').selectedIndex = images.length - 1;
    displayPic();
};

const right = () => {
    if (document.querySelector('select').selectedIndex < images.length - 1)
        document.querySelector('select').selectedIndex++;
    else document.querySelector('select').selectedIndex = 0;
    displayPic();
};

addEventListener('load', init);
</script>
</head>
<body>
</body>
</html>

```

## 4.4. XML

[www.w3.org/TR/REC-xml](http://www.w3.org/TR/REC-xml) Extensible Markup Language. For a gentle introduction, see [www.w3schools.com/xml/xml\\_what\\_is.asp](http://www.w3schools.com/xml/xml_what_is.asp).

To deal with XML in PHP see [www.w3schools.com/php/php\\_xml\\_parsers.asp](http://www.w3schools.com/php/php_xml_parsers.asp).

XPath ([www.w3.org/TR/xpath](http://www.w3.org/TR/xpath)) is a language for working with XML documents. A good intro to XPath syntax can be found at [www.w3schools.com/xpath/xpath\\_syntax.asp](http://www.w3schools.com/xpath/xpath_syntax.asp).

### 4.4.1. SVG

Using Scalable Vector Graphics (cf. [www.w3schools.com/graphics/svg\\_reference.asp](http://www.w3schools.com/graphics/svg_reference.asp)) we can specify vector graphics using XML. Whereas standard graphics are specified pixel by pixel, usually using an editor software, vector graphics are specified using paths having a start and end point, as well as points, curves and angles in between. The main advantages of SVG over pixel images are as follows:

1. SVG images are pure XML that can be created and edited with any text editor, with open source software such as Inkscape ([www.inkscape.org](http://www.inkscape.org)) or with a good online SVG editor such as [github.com/SVG-Edit/svgedit](https://github.com/SVG-Edit/svgedit).
2. SVG images can be searched, indexed, scripted, and compressed.

3. SVG images are scalable without loss of quality.

The official web site is [www.w3.org/Graphics/SVG](http://www.w3.org/Graphics/SVG) and the specification can be found at [www.w3.org/TR/SVG11](http://www.w3.org/TR/SVG11). Excellent tutorials can be found at [www.tutorialspoint.com/svg](http://www.tutorialspoint.com/svg), [edutechwiki.unige.ch/en/Static\\_SVG\\_tutorial](http://edutechwiki.unige.ch/en/Static_SVG_tutorial), [edutechwiki.unige.ch/en/Using\\_SVG\\_with\\_HTML5\\_tutorial](http://edutechwiki.unige.ch/en/Using_SVG_with_HTML5_tutorial), [www.w3schools.com/graphics/svg\\_intro.asp](http://www.w3schools.com/graphics/svg_intro.asp) and [developer.mozilla.org/en-US/docs/Web/SVG](http://developer.mozilla.org/en-US/docs/Web/SVG).

SVG demos can be found at [www.hongkiat.com/blog/svg-animations](http://www.hongkiat.com/blog/svg-animations), [www.creativebloq.com/design/examples-svg-7112785](http://www.creativebloq.com/design/examples-svg-7112785) and [svg-wow.org](http://svg-wow.org).

## Syntax

SVG is an XML dialect. There are two ways to get SVG into your browser, either via a SVG or an HTML file.

Here are two examples:

[students.btsi.lu/evegi144/WAD/XML/SVG/SVG1.svg](http://students.btsi.lu/evegi144/WAD/XML/SVG/SVG1.svg)

```
<svg viewBox="0 0 500 500" xmlns="http://www.w3.org/2000/svg">
  <defs>
    <pattern id="gridPattern" width="10" height="10"
    patternUnits="userSpaceOnUse">
      <path d="M10,0 H0 V10" fill="none" stroke="black" stroke-width="2.5"/>
    </pattern>
  </defs>
  <rect width="200" height="500" fill="green"/>
  <circle cx="250" cy="25" r="25" fill="purple" stroke="gold" stroke-width="3"/>
  <polyline points="200, 60, 240, 230, 310, 230, 350, 60" fill="lightcyan"
    fill-opacity="0.7" stroke="darkviolet" stroke-width="150" stroke-
linecap="round"
    stroke-opacity="0.3" />
  <rect width="100%" height="100%"
    fill="url(#gridPattern)" fill-opacity="1" stroke="black" stroke-
width="2.5"/>
</svg>
```

[students.btsi.lu/evegi144/WAD/XML/SVG/SVG1.html](http://students.btsi.lu/evegi144/WAD/XML/SVG/SVG1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>My first SVG experiment</title>
    <meta charset=utf-8>
    <style>
      rect:hover {
        fill: gold;
      }
    </style>
  </head>
  <body>
    <img alt="A green rectangle with a purple circle, a light cyan polyline, and a black grid pattern." data-bbox="100 770 489 926"/>
  </body>
</html>
```

```

    span[dir] {
      unicode-bidi: bidi-override;
    }
  </style>
</head>
<body>
  <header>
    <svg xmlns="http://www.w3.org/2000/svg" version="1.1" height="60">
      <!-- add title for accessibility -->
      <title>Applying a gradient background to text in SVG</title> -->
      <!-- Source: http://lea.verou.me/2012/05/text-masking-the-standards-way/ -->
      <defs>
        <linearGradient id="filler">
          <stop stop-color="red" offset="0%"/>
          <stop stop-color="white" offset="50%"/>
          <stop stop-color="blue" offset="100%"/>
        </linearGradient>
      </defs>
      <text x="50%" y="50" font-size="50" fill="url(#filler)">T0IF</text>
    </svg>
  </header>
  <main>
    <section>
      <script>
        'use strict';

        const hello = () => {
          alert('Hello!');
        };
      </script>
      <svg width=1000 height=400>
        <text x=87 y=300 font-size=26 fill=black>Click me
          <!--<animate attributeName=font-size dur=5s values=26;32;20;26
            repeatCount=indefinite></animate>-->
        </text>
        <rect onclick=hello(); x=75 y=276 height=30 width=120 stroke=black
          stroke-width=2 fill=green opacity=.5 rx=10>
          <!--<animate attributeName=width dur=5s values=120;160;90;120
            repeatCount=indefinite></animate>-->
        </rect>
        <path d="M 100 200 200 200 150 100 z" stroke=black stroke-width=2
          fill=url(#g2)></path>
        <linearGradient id=g1>
          <stop offset=0 stop-color=white></stop>
          <stop offset=1 stop-color=black></stop>
        </linearGradient>
        <radialGradient id=g2>
          <stop offset=0 stop-color=white></stop>
          <stop offset=1 stop-color=black></stop>
        </radialGradient>
      </svg>
    </section>
  </main>
</body>
</html>

```

```

<defs>
  <path id=curve d="M 10 100 C 200 30 300 250 350 50"></path>
</defs>
<text font-family=arial font-size=16 fill=black>
  <textPath xlink:href=#curve>Hello, here is some text lying
    along a Bézier curve.
  </textPath>
  <animateMotion dur="2s"
    rotate="auto" fill="freeze" repeatCount="indefinite">
    <mpath xlink:href="#curve"/>
  </animateMotion>
</text>
<line id=water x1=-50 y1=110 x2=100% y2=110 stroke=blue stroke-width=1
  stroke-opacity=0.7></line>
<g id=scene>
  <circle id=sun r=50 cx=30 cy=30 fill=orange stroke=grey
    stroke-width=1></circle>
  <circle id=venusInTransit r=5 cx=15 cy=20 fill=black stroke=grey
    stroke-width=1></circle>
</g>
<use xlink:href=#scene mask=url(#hazeIca) transform=scale(1,-1)
  translate=(30,-210) skewX=(-20) skewY=(5)></use>
<!--<ellipse cx=500 cy=300 rx=30 ry=40 fill=#448 opacity=.75
  stroke=black" stroke-width="3">
  <animate attributeName="rx" dur="5s"
    values="10;70;10" repeatCount="indefinite"/>
  <animate attributeName="ry" dur="5s"
    values="30;60;30" repeatCount="indefinite"/>
</ellipse>-->
</svg>
<p>This is a test text that we are now reversing
  <bdo dir="rtl">This is a test text that we are now reversing</bdo></p>
</section>
<section>
  <svg width=400 height=50>
    <rect width=200 height=50 fill=green></rect>
    <circle cx=250 cy=25 r=25 fill=purple stroke=gold
      stroke-width=3></circle>
  </svg>
</section>
<section>
  <svg width=600 height=500>
    <path d="M 100 350 300 100 500 350 z M 250 320 250 220 350 220 350 320 z"
      fill=#ff8 stroke=black stroke-width=15 fill-rule="evenodd"/>
  </svg>
</section>
<section>
  <svg width=600 height=300>
    <path d="M 0 0 Q 300 0 200 50 100 100" fill=gold stroke=blue
      stroke-width=25></path>
  </svg>

```

```
    </section>
  </main>
</body>
</html>
```

To embed HTML into SVG use `foreignObject` as shown in [stackoverflow.com/questions/4176146/svg-based-text-input-field](https://stackoverflow.com/questions/4176146/svg-based-text-input-field) and [jsfiddle.net/leaverou/qwg3r](https://jsfiddle.net/leaverou/qwg3r).

## Responsive CSS

Study the brilliant [www.creativebloq.com/how-to/10-golden-rules-for-responsive-svgs](http://www.creativebloq.com/how-to/10-golden-rules-for-responsive-svgs) and [developer.mozilla.org/en-US/docs/Web/SVG/Attribute/viewBox](https://developer.mozilla.org/en-US/docs/Web/SVG/Attribute/viewBox) and try the code samples.

[students.btsi.lu/evegi144/WAD/XML/SVG/SVG2.html](http://students.btsi.lu/evegi144/WAD/XML/SVG/SVG2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>SVG2</title>
    <style>
      header svg text {
        font-weight: 900; font-size: 3em; fill: blue;
      }
    </style>
  </head>
  <body>
    <header>
      <svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 500 100">
        <text x="0" y="40">I'm really responsive!</text>
      </svg>
    </header>
  </body>
</html>
```

From [developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5/HTML5\\_Parser](https://developer.mozilla.org/en-US/docs/Web/Guide/HTML/HTML5/HTML5_Parser):

The syntax `<foo/>` opens and immediately closes the `foo` element if it is a MathML or SVG element (i.e. not an HTML element). Attributes are tokenized the same way they are tokenized in HTML, so you can omit quotes in the same situations where you can omit quotes in HTML (i.e. when the attribute value is not the empty string and does not contain whitespace, `"`, `'`, ```, `<`, `=`, or `>`).



The two above features do not combine well due to the reuse of legacy-compatible HTML tokenization. If you omit quotes on the last attribute value, you must have a space before the closing slash. `<circle fill=green />` is OK but `<circle fill=red/>` is not.

#### 4.4.2. RSS

From [www.rssboard.org/rss-specification](http://www.rssboard.org/rss-specification):

RSS is a Web content syndication format. Its name is an acronym for Really Simple Syndication. RSS is a dialect of XML. All RSS files must conform to the XML 1.0 specification, as published on the World Wide Web Consortium (W3C) website.

[www.w3schools.com/webservices/rss\\_intro.asp](http://www.w3schools.com/webservices/rss_intro.asp) provides a good introduction.

Open a RSS feed, for instance [www.ghacks.net/feed](http://www.ghacks.net/feed), in your browser and look at the page source.

At the top level, we have an `rss` element with a mandatory `version` attribute. The latest version is 2.0. By simply checking for the presence of this element, we can easily determine whether it's a RSS feed or not.

The top level element contains a `channel` element. The channel element must contain the following elements:

- `title`
- `link`
- `description`

It can contain a large number of optional elements, which are listed and explained on the specification page shown above. The most important optional element for our purposes is `item`.

[www.w3schools.com/php/php\\_ajax\\_rss\\_reader.asp](http://www.w3schools.com/php/php_ajax_rss_reader.asp) shows a sample RSS reader using PHP and AJAX.

#### 4.4.3. Atom

The official standard specification can be found at [tools.ietf.org/html/rfc4287](http://tools.ietf.org/html/rfc4287).

Open an Atom feed, for instance `<a href="http://www.xn&#8212;&#8203;atenschutzunfile-7nb.de/asu.xml" class="bare" target="_blank">www.xn&#8212;&#8203;atenschutzunfile-7nb.de/asu.xml</a>`, in your browser and look at the page source.

The main element in an atom feed is named `feed`. By simply checking for the presence of this element, we can easily determine whether it's an Atom feed or not.

The top level element contains zero or more `entry` elements. The entry element usually contains at least the following elements:

- **title**
- **link**
- **content**

#### 4.4.4. OPML

Outline Processor Markup Language or OPML is a format for storing outlines in XML 1.0 and to exchange information between outliners and Internet services that can be browsed or controlled through an outliner. The specification can be found at [dev.opml.org/spec2.html](http://dev.opml.org/spec2.html).

#### 4.4.5. MathML

[www.w3.org/Math](http://www.w3.org/Math) The formal specification can be found at [www.w3.org/TR/MathML2](http://www.w3.org/TR/MathML2). The Mozilla Developer Network (MDN) at [developer.mozilla.org/en-US/docs/Web/MathML](http://developer.mozilla.org/en-US/docs/Web/MathML) is a good starting point for MathML, in particular the element reference ([developer.mozilla.org/en-US/docs/Web/MathML/Element](http://developer.mozilla.org/en-US/docs/Web/MathML/Element)). A nice text to MathML converter can be found at [www.mathmlcentral.com/Tools/ToMathML.jsp](http://www.mathmlcentral.com/Tools/ToMathML.jsp).

Here's a very basic MathML example:

[students.btsi.lu/evegi144/WAD/XML/MathML/example1.html](http://students.btsi.lu/evegi144/WAD/XML/MathML/example1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>MathML example</title>
    <meta charset=utf-8>
  </head>
  <body>
    <main>
      <math>
        <mrow>
          <mi>cos</mi>
          <mo>( </mo>
          <msup>
            <mi>x</mi>
            <mn>3</mn>
          </msup>
          <mo>)</mo>
        </mrow>
        <mfrac bevelled="true">
          <mfrac>
            <mi>a</mi>
            <mi>b</mi>
          </mfrac>
          <mfrac>
            <mi>c</mi>
            <mi>d</mi>
          </mfrac>
        </mfrac>
      </math>
    </main>
  </body>
</html>

```

## 4.5. Web Application Programming Interfaces

See [en.wikipedia.org/wiki/Application\\_programming\\_interface#Web\\_APIs](https://en.wikipedia.org/wiki/Application_programming_interface#Web_APIs) for a detailed explanation of the term.

Here is an open-source API for generating random user data: [randomuser.me](https://randomuser.me).

Here is a noncomprehensive list of Web APIs:

[docs.api.tfl.lu](https://docs.api.tfl.lu)

[data.public.lu](https://data.public.lu)

[market.mashape.com/explore](https://market.mashape.com/explore)

[www.programmableweb.com/apis/directory](https://www.programmableweb.com/apis/directory)

[github.com/toddmotto/public-apis](https://github.com/toddmotto/public-apis)

[gearside.com/public-json-feeds](https://gearside.com/public-json-feeds)

[any-api.com](https://any-api.com)

[webresourcesdepot.com/15-free-apis-you-didnt-hear-about-but-will-make-use-of](https://webresourcesdepot.com/15-free-apis-you-didnt-hear-about-but-will-make-use-of)

[shkspr.mobi/blog/2014/04/wanted-simple-apis-without-authentication](https://shkspr.mobi/blog/2014/04/wanted-simple-apis-without-authentication)

[fixer.io](https://fixer.io)

[www.faroo.com/hp/api/api.html](https://www.faroo.com/hp/api/api.html)

[speckyboy.com/18-free-mobile-apis-developers-should-consider](https://speckyboy.com/18-free-mobile-apis-developers-should-consider)

### 4.5.1. CKEditor

A fantastic and well documented web text editor can be found at [ckeditor.com](https://ckeditor.com). The only drawback is that it does not yet work on Android.

A simple usage example: [students.btsi.lu/evegi144/WAD/API/CKEditor/CKEditor1.html](https://students.btsi.lu/evegi144/WAD/API/CKEditor/CKEditor1.html)

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Ajax &mdash; CKEditor Sample</title>
    <script src="//cdn.ckeditor.com/4.4.7/full/ckeditor.js"></script>
    <script>
      'use strict';

      let editor, html = '';

      const createEditor = () => {
        if (editor) return;

        // Create a new editor inside the <div id="editor">, setting its value to html
        editor = CKEDITOR.appendTo('editor', {}, html);
      };

      const removeEditor = () => {
        if (!editor) return;

        // Retrieve the editor contents. In an Ajax application, this data would be
        // sent to the server or used in any other way.
        document.getElementById('editorcontents').innerHTML = html = editor.getData();
        document.getElementById('contents').style.display = '';

        // Destroy the editor.
        editor.destroy();
        editor = null;
      };

    </script>
```

```

</head>
<body>
  <p>
    <input onclick=createEditor(); type="button" value="Create Editor">
    <input onclick=removeEditor(); type="button" value="Remove Editor">
  </p>
  <!-- This div will hold the editor. -->
  <div id="editor">
  </div>
  <div id="contents" style="display: none">
    <p>
      Edited Contents:
    </p>
    <!-- This div will be used to display the editor contents. -->
    <div id="editorcontents">
    </div>
  </div>
</body>
</html>

```

### 4.5.2. Google Charts

You can find detailed documentation and examples at [google-developers.appspot.com/chart](https://google-developers.appspot.com/chart).

Here is a simple example: [students.btsi.lu/evegi144/WAD/API/GoogleCharts/chart1.html](https://students.btsi.lu/evegi144/WAD/API/GoogleCharts/chart1.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>My first Google chart</title>
    <meta charset=utf-8>
    <script src=https://www.google.com/jsapi></script>
    <script>
      'use strict';

      // Callback that creates and populates a data table,
      // instantiates the pie chart, passes in the data and draws it.
      const drawChart = () => {
        // Create the data table.
        const data = new google.visualization.DataTable();
        data.addColumn('string', 'Module');
        data.addColumn('number', 'Hours per week');
        data.addRows([
          ['ALLEM1', 2],
          ['ANGLA1', 2],
          ['ANGTE1', 2],
          ['EDUPH', 2],
          ['MATHE1', 2],
          ['SYSEX1', 6],
          ['CREDO', 5],

```

```

        ['ELINF1', 2],
        ['EDUCI1', 2],
        ['ATINF1', 5]
    ]);

    // Set chart options
    const options = {
        'title': 'T0IF weekly module hours term 1',
        'width': 700,
        'height': 500
    };

    // Instantiate and draw our chart, passing in some options.
    const chart = new
google.visualization.PieChart(document.querySelector('main'));
    chart.draw(data, options);
};

const drawChart2 = () => {
    const data = google.visualization.arrayToDataTable([
        ['Mon', 20, 28, 38, 45],
        ['Tue', 31, 38, 55, 66],
        ['Wed', 50, 55, 77, 80],
        ['Thu', 77, 77, 66, 50],
        ['Fri', 68, 66, 22, 15]
        // Treat first row as data as well.
    ], true);

    const options = {
        legend: 'none'
    };

    const chart = new
google.visualization.CandlestickChart(document.querySelector('main'));
    chart.draw(data, options);
};

// Load the Visualization API and the piechart package.
google.load('visualization', '1.0', {'packages': ['corechart']});

// Set a callback to run when the Google Visualization API is loaded.
google.setOnLoadCallback(drawChart2);
</script>
</head>
<body>
    <main>
    </main>
</body>
</html>

```

### 4.5.3. pdfmake

Generate PDFs in pure JS: [pdfmake.org](http://pdfmake.org)

### 4.5.4. Facebook

Go to [developers.facebook.com](https://developers.facebook.com) and register as a Facebook developer. Then create a FB app id for your application.

### 4.5.5. Yahoo Query Language (YQL)

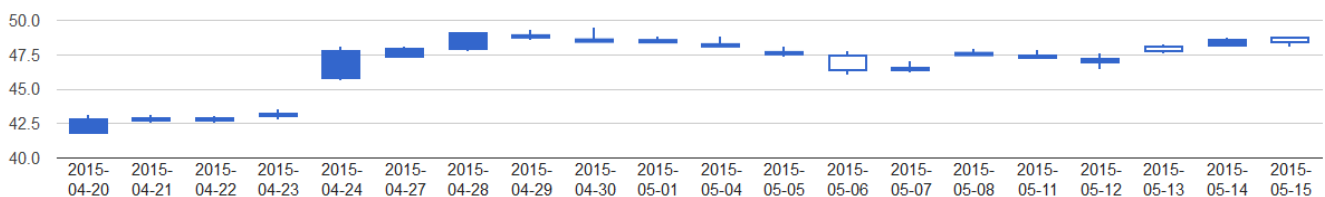
This service allows us to access Internet data with SQL-like commands (cf. [developer.yahoo.com/yql](http://developer.yahoo.com/yql)).

### 4.5.6. Finance

#### Quandl

Quandl ([www.quandl.com](http://www.quandl.com)) is a data platform covering over 10 million datasets from 500 sources accessible via a simple API ([www.quandl.com/help/api](http://www.quandl.com/help/api)).

Here is an example illustrating the retrieval of stock data from Quandl and candlestick charting with Google Charts:



[students.btsi.lu/evegi144/WAD/API/Finance/Quandl/quandl1.html](http://students.btsi.lu/evegi144/WAD/API/Finance/Quandl/quandl1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Using Quandl and Google Charts</title>
    <meta charset=utf-8>
    <script src=https://www.google.com/jsapi></script>
    <script>
      'use strict';

      let result;
      google.load('visualization', '1.0', {'packages': ['corechart']});

      const drawChart = () => {
        /* From Quandl we get open, high, low, close.
         GoogleCharts requires low, open, close, high.
         We therefore need to rearrange the data.
         */
        const dat = result.dataset.data.reverse();
```

```

// We need the oldest data first (left to right).
let low, open, close, high;
for (let i = 0; i < dat.length; i++) {
  low = dat[i][3];
  open = dat[i][1];
  close = dat[i][4];
  high = dat[i][2];
  dat[i].pop();
  dat[i][1] = low;
  dat[i][2] = open;
  dat[i][3] = close;
  dat[i][4] = high;
}
const data = google.visualization.arrayToDataTable(dat.slice(-20), true);

const options = {
  legend: 'none',
  /*height: 600,
  width: 1000,*/
  hAxis: {slantedTextAngle: 90}/*,
  chartArea: {left: 50, top: 20, width: 950, height: 400}*/
};

const chart = new
google.visualization.CandlestickChart(document.querySelector('main'));
chart.draw(data, options);
};

const init = () => {
  const URL = "https://www.quandl.com/api/v3/datasets/GOOG/NASDAQ_MSFT.json";
  const req = new XMLHttpRequest();
  req.open('GET', URL);
  req.addEventListener('load', e => {
    result = JSON.parse(e.target.response);
    drawChart();
  });
  req.send();
};

addEventListener('load', init);
</script>
</head>
<body>
  <main>
  </main>
</body>
</html>

```

# Federal Reserve Bank of St. Louis

FRED is the place for economic research on the US ([research.stlouisfed.org](https://research.stlouisfed.org)). The FRED API ([api.stlouisfed.org/docs/fred](https://api.stlouisfed.org/docs/fred)) allows us programmatic access to the whole DB.

Advance Monthly Sales for Retail and Food Services		Retail Trade: Furniture and Home Furnishings Stores
Date	Value	
1992-01-01	4254	
1992-02-01	4342	
1992-03-01	4330	
1992-04-01	4359	
1992-05-01	4331	
1992-06-01	4401	
1992-07-01	4368	
1992-08-01	4372	
1992-09-01	4345	
1992-10-01	4299	
1992-11-01	4368	
1992-12-01	4428	
1993-01-01	4637	
1993-02-01	4507	
1993-03-01	4463	
1993-04-01	4564	

[students.btsi.lu/evegi144/WAD/API/Finance/FRED/FRED1.html](https://students.btsi.lu/evegi144/WAD/API/Finance/FRED/FRED1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Federal Reserve Bank of St. Louis JSON data retrieval</title>
    <style>
      html, body {
        width: 100%;
        height: 100%;
        margin: 0;
        padding: 0;
        overflow: hidden;
      }

      body {
        display: flex;
        flex-direction: column;
        flex:auto;
      }

      nav {
        background-color: lightgrey;
        padding: 5px;
      }

      main {
        overflow: auto;
        flex: auto;
```

```

}

footer {
  min-height: 18px;
  text-align: center;
  background-color: lightgrey;
  font-size: 50%;
}
table {
  border: 1px solid black;
  border-collapse: collapse;
}

th, td {
  border: 1px solid black;
  padding: 3px;
}
</style>
<script>
  'use strict';

  const AJAXFunctionCall = (functionName, parameter = '', callback) => {
    const req = new XMLHttpRequest();
    const data = new FormData();
    data.append('function', functionName);
    data.append('parameter', parameter);
    req.open('POST', 'FRED_functions.php');
    if (callback) req.addEventListener('load', callback);
    req.send(data);
  };

  const init = () => {
    AJAXFunctionCall('get_releases', '', displayReleases);
    document.querySelector('select').addEventListener('change', releaseChange);
    document.querySelectorAll('select')[1].addEventListener('change',
seriesChange);
  };

  const displayReleases = e => {
    const releases = JSON.parse(e.target.response).releases;
    const select = document.querySelector('select');
    for (let i = 0; i < releases.length; i++) {
      const opt = document.createElement('option');
      opt.value = releases[i].id;
      opt.innerHTML = releases[i].name;
      select.appendChild(opt);
    }
    AJAXFunctionCall('get_series', select.value, displaySeries);
  };

  const displaySeries = e => {

```

```

const series = JSON.parse(e.target.response).series;
const select = document.querySelectorAll('select')[1];
select.innerHTML = '';
for (let i = 0; i < series.length; i++) {
  const opt = document.createElement('option');
  opt.value = series[i].id;
  opt.innerHTML = series[i].title;
  select.appendChild(opt);
}
};

const displayObservations = e => {
  const obs = JSON.parse(e.target.response).observations;
  const table = document.createElement('table');
  let s = '<table><tr><th>Date</th><th>Value</th></tr><tr>';
  for (let i = 0; i < obs.length; i++) {
    s += '<tr><td>${obs[i].date}</td><td>${obs[i].value}</td></tr>';
  }
  table.innerHTML = `${s}</table>`;
  const oldTable = document.querySelector('table');
  if (oldTable) document.querySelector('main').replaceChild(table, oldTable);
  else document.querySelector('main').appendChild(table);
};

const releaseChange = e => {
  AJAXFunctionCall('get_series', document.querySelector('select').value,
    displaySeries);
};

const seriesChange = e => {
  AJAXFunctionCall('get_observations',
document.querySelectorAll('select')[1].value,
    displayObservations);
};

addEventListener('load', init);
</script>
</head>
<body>
  <nav>
    <select></select>
    <select></select>
  </nav>
  <main></main>
  <footer>This product uses the FRED® API but is not endorsed or certified by the
Federal
  Reserve Bank of St. Louis.</footer>
</body>
</html>

```

```
<?php
require_once('key.php');

function get_releases() {
    global $key;
    $URL = 'https://api.stlouisfed.org/fred/releases?api_key=' . $key .
'&file_type=json';
    echo file_get_contents($URL);
}

function get_series($release_id) {
    global $key;
    $URL = 'https://api.stlouisfed.org/fred/release/series?api_key=' . $key .
'&file_type=json&release_id=' . $release_id;
    echo file_get_contents($URL);
}

function get_observations($series_id) {
    global $key;
    $URL = 'https://api.stlouisfed.org/fred/series/observations?api_key=' . $key .
'&file_type=json&series_id=' . $series_id;
    echo file_get_contents($URL);
}

if (isset($_POST['function']))
    if ($_POST['function'] === 'get_releases') get_releases();
    elseif ($_POST['function'] === 'get_series' && isset($_POST['parameter']))
        get_series($_POST['parameter']);
    elseif ($_POST['function'] === 'get_observations' && isset($_POST['parameter']))
        get_observations($_POST['parameter']);
?>
```

## World Bank

## Eurostat

## Yahoo! Finance

This API allows us to download current and historical price and other information, charts and RSS news feeds for financial instruments.

## IEX

IEX is a fair, simple and transparent stock exchange dedicated to investor protection.

Their API can be found at [iextrading.com/developer](https://iextrading.com/developer).

### 4.5.7. Shodan

[developer.shodan.io/api](https://developer.shodan.io/api)

## 4.6. Security



Study [CIA Vault 7!](#)

[www.eff.org](https://www.eff.org)

### 4.6.1. Password hashing

[pthree.org/2016/06/28/lets-talk-password-hashing](https://pthree.org/2016/06/28/lets-talk-password-hashing)

### 4.6.2. Cross-Site Request Forgery (CSRF)

We need to send a token with our form:

```
<?php
$token = password_hash(random_int(1, 999999999), PASSWORD_DEFAULT);
$_SESSION['token'] = $token;
?>

<form method=post>
  <input type=hidden name=token value=<?php echo $token; ?>>
</form>

<?php
  if (isset($_SESSION['token'], $_POST['token']) && $_POST['token'] ==
$_SESSION['token'])
    //valid
?>
```

[www.acunetix.com/websitesecurity/csrf-attacks](https://www.acunetix.com/websitesecurity/csrf-attacks)

[www.owasp.org/index.php/Cross-Site\\_Request\\_Forgery\\_\(CSRF\)\\_Prevention\\_Cheat\\_Sheet](https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF)_Prevention_Cheat_Sheet)

[stackoverflow.com/questions/20504846/why-is-it-common-to-put-csrf-prevention-tokens-in-cookies](https://stackoverflow.com/questions/20504846/why-is-it-common-to-put-csrf-prevention-tokens-in-cookies)

[github.com/pillarjs/understanding-csrf](https://github.com/pillarjs/understanding-csrf)

[stackoverflow.com/questions/20504846/why-is-it-common-to-put-csrf-prevention-tokens-in-cookies](https://stackoverflow.com/questions/20504846/why-is-it-common-to-put-csrf-prevention-tokens-in-cookies)

### 4.6.3. Fiddler

[www.telerik.com/fiddler](https://www.telerik.com/fiddler)

Get SyntaxView	Transformer	Headers	TextView	ImageView	HexView	WebView	Auth	Caching	Cookies
JSON	XML								

The SyntaxView Inspector displays syntax-highlighted HTML, Script, CSS, and XML. If you're a web developer, you'll want this add-on.

[Download and Install SyntaxView now...](#)

#### 4.6.4. Google hacking

[en.wikipedia.org/wiki/Google\\_hacking](https://en.wikipedia.org/wiki/Google_hacking)

[www.exploit-db.com/google-hacking-database](https://www.exploit-db.com/google-hacking-database)

[www.bishopfox.com/resources/tools/google-hacking-diggity](https://www.bishopfox.com/resources/tools/google-hacking-diggity)

#### 4.6.5. HTTPS

[en.wikipedia.org/wiki/HTTPS](https://en.wikipedia.org/wiki/HTTPS)

#### 4.6.6. Email

[emailprivacytester.com](https://emailprivacytester.com)

We can publish our email address on our web site, without getting inundated with spam. Here are some good solutions:

[www.katpatuka.org/pub/doc/anti-spam.html](https://www.katpatuka.org/pub/doc/anti-spam.html)

[www.wbwip.com/wbw/emailencoder.html](https://www.wbwip.com/wbw/emailencoder.html)

[stackoverflow.com/questions/3624667/how-to-spamproof-a-mailto-link](https://stackoverflow.com/questions/3624667/how-to-spamproof-a-mailto-link)

[embed.plnkr.co/96qybOA6IY1EkWPA7X1B](https://embed.plnkr.co/96qybOA6IY1EkWPA7X1B)

[stackoverflow.com/questions/3624667/how-to-spamproof-a-mailto-link](https://stackoverflow.com/questions/3624667/how-to-spamproof-a-mailto-link)

#### 4.6.7. Tracking

[www.ghacks.net/2017/12/09/a-solution-to-etag-tracking-in-firefox](https://www.ghacks.net/2017/12/09/a-solution-to-etag-tracking-in-firefox)

### 4.7. Mobile and desktop application development

[codeburst.io/react-native-vs-nativescript-vs-xamarin-vs-flutter-vs-ionic-vs-cordova-vs-pwas-196810954450](https://codeburst.io/react-native-vs-nativescript-vs-xamarin-vs-flutter-vs-ionic-vs-cordova-vs-pwas-196810954450)

#### 4.7.1. Mobile

Progressive web app development: [developers.google.com/web/fundamentals/codelabs/your-first-pwapp](https://developers.google.com/web/fundamentals/codelabs/your-first-pwapp)

Here is a good article on how to solve the hover problem on mobile devices:

[www.javascriptkit.com/dhtmltutors/sticky-hover-issue-solutions.shtml](http://www.javascriptkit.com/dhtmltutors/sticky-hover-issue-solutions.shtml)

[dzone.com/refcardz/html5-mobile-development](http://dzone.com/refcardz/html5-mobile-development)

[en.wikipedia.org/wiki/Mobile\\_application\\_development](http://en.wikipedia.org/wiki/Mobile_application_development)

[mobiledetect.net](http://mobiledetect.net)

## **Cordova**

[cordova.apache.org](http://cordova.apache.org)

[noeticforce.com/best-hybrid-mobile-app-ui-frameworks-html5-js-css](http://noeticforce.com/best-hybrid-mobile-app-ui-frameworks-html5-js-css)

## **NativeScript**

[www.nativescript.org](http://www.nativescript.org)

## **MIT App Inventor 2**

[ai2.appinventor.mit.edu](http://ai2.appinventor.mit.edu)

## **Titanium**

[www.appcelerator.com](http://www.appcelerator.com)

## **Ubuntu**

```
umask 0022  
npm install cordova -g --user root  
cordova create myApp
```

## **Ionic**

[ionicframework.com](http://ionicframework.com)

[blog.ionic.io/what-is-a-progressive-web-app](http://blog.ionic.io/what-is-a-progressive-web-app)

## **Framework 7**

[framework7.io](http://framework7.io)

## **Sencha**

[www.sencha.com](http://www.sencha.com)

## **Android SDK**

[developer.android.com/sdk/index.html](http://developer.android.com/sdk/index.html)

[www.maketecheasier.com/install-applications-without-the-market](http://www.maketecheasier.com/install-applications-without-the-market)

## 4.7.2. Desktop

### Electron

[electronjs.org](https://electronjs.org)

### NW.js

[nwjs.io](https://nwjs.io)

## 4.8. Other

### 4.8.1. Accessibility

Study [accessibility principles](#) then automatically detect accessibility issues on web pages using [open-indy.github.io/Koa11y](https://open-indy.github.io/Koa11y).

### 4.8.2. Search engine and social network optimization, eCommerce

<a href="https://moz.com/beginners-guide-to-seo">moz.com/beginners-guide-to-seo</a>
<a href="https://www.willcoombe.com/on-page-optimisation">www.willcoombe.com/on-page-optimisation</a>
<a href="https://www.google.com/webmasters/tools/home?hl=en">www.google.com/webmasters/tools/home?hl=en</a>
<a href="https://search.google.com/test/mobile-friendly">search.google.com/test/mobile-friendly</a>
<a href="https://en.wikipedia.org/wiki/Search_engine_optimization">en.wikipedia.org/wiki/Search_engine_optimization</a>
<a href="https://varvy.com">varvy.com</a>
<a href="https://www.quicksprout.com">www.quicksprout.com</a>
<a href="https://www.makemyownwebpage.com/search-engine-optimization">www.makemyownwebpage.com/search-engine-optimization</a>
<a href="https://ogp.me">ogp.me</a>
<a href="https://www.outerboxdesign.com/search-marketing/search-engine-optimization/seo-trends-2018">www.outerboxdesign.com/search-marketing/search-engine-optimization/seo-trends-2018</a>
<a href="https://schema.org">schema.org</a>
<a href="https://searchengineland.com">searchengineland.com</a>
<a href="https://www.robotstxt.org/meta.html">www.robotstxt.org/meta.html</a>
<a href="https://www.sitemaps.org">www.sitemaps.org</a>

Use [Google search console](#).

You should markup your HTML with microdata, see [www.keithcirkel.co.uk/schema-org-the-new-generation-of-seo](https://www.keithcirkel.co.uk/schema-org-the-new-generation-of-seo) and [webknit.co.uk/library/accessibility/webpage-using-schemas-and-microdata](https://webknit.co.uk/library/accessibility/webpage-using-schemas-and-microdata).

### Search engine optimization

Search engines operate as follows (cf. [d2eeipcradle6.cloudfront.net/guides/Moz-The-Beginners-Guide-To-SEO.pdf](https://d2eeipcradle6.cloudfront.net/guides/Moz-The-Beginners-Guide-To-SEO.pdf)):

## 1. Crawling and indexing

Search engines crawl and index links in web pages using robots called "crawlers" or "spiders" to reach the billions of interconnected documents on the web. They then store selected information from them in their databases.

## 2. Providing answers

When a user performs an online search, the engine scours its databases for this information and ranks the results according to their relevance and popularity (cf. [moz.com/search-ranking-factors](https://moz.com/search-ranking-factors)).

### **Know your customer**

A key step in creating a successful eCommerce site is to analyse your market audience and create a list of the top keywords that are relevant for your business.

### **4.8.3. User agent switching**

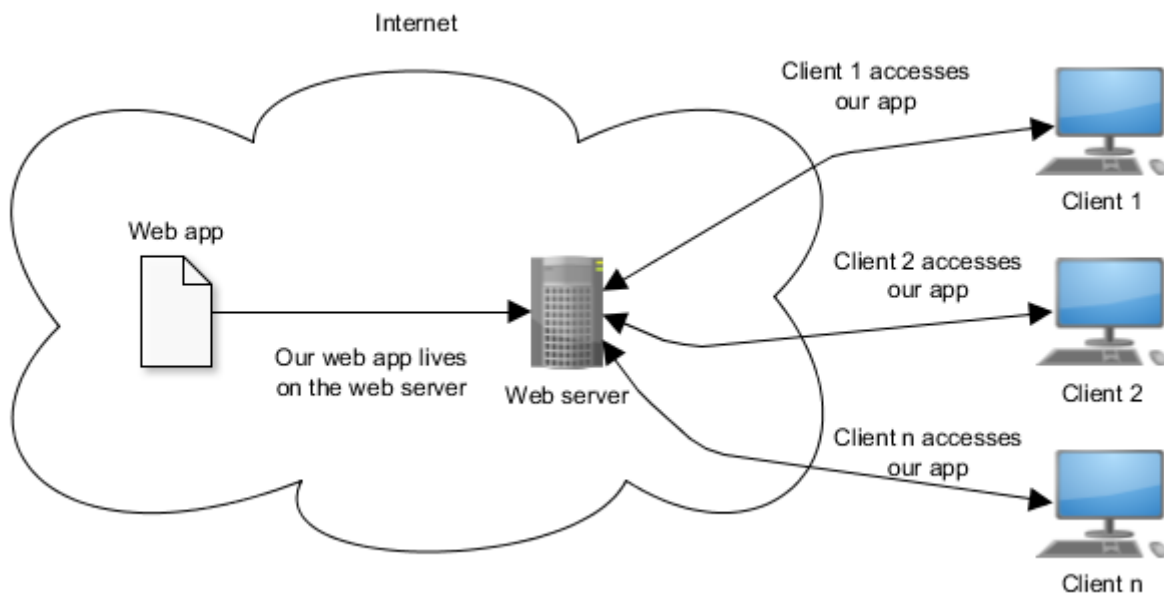
[www.youtube.com/watch?v=xLNe8iGskrw&feature=youtu.be](https://www.youtube.com/watch?v=xLNe8iGskrw&feature=youtu.be)

# Chapter 5. Server side programming

## 5.1. Installing and configuring the tools

### 5.1.1. Introduction

Before we can develop our own web applications, we need access to a web server. The web server is the machine and software that delivers the web page to our users worldwide who are eager to use our apps:



Our web apps will consist of different parts. The content will be structured, or marked up, using HyperText Markup Language (HTML) and styled using Cascading Style Sheets (CSS). The behavior of our app will be programmed using JavaScript. All modern browsers understand HTML, CSS and JavaScript. In order to store data in a database, we use a web server that understands a programming language such as PHP or JavaScript. From PHP or JS it is easy to access a database, such as MySQL, which lives on the server. A relational database can be controlled using Structured Query Language (SQL).



It is essential to distinguish between code that is executed on the client side (HTML, CSS and JavaScript) and code that runs on the server side (for instance PHP, JavaScript or SQL). A browser understands JS but not PHP or SQL. This is not a problem as the server side code gets executed on the server and the results (HTML, CSS and JavaScript) are sent to the client browser for execution.

There are a number of different web servers available. A quick overview can be found at [en.wikipedia.org/wiki/Comparison\\_of\\_web\\_server\\_software](https://en.wikipedia.org/wiki/Comparison_of_web_server_software).

In order to develop your own web apps at home efficiently, you should install your own web server, either directly on your home PC or in a virtual machine. This will increase your understanding of the different parts that are involved in the web app life cycle. In order to execute a PHP script

named, for instance, `index.php` in the main directory of your home web server, you need to run it in your browser using [localhost/index.php](http://localhost/index.php). If you try to run it by double clicking on it in the file manager, your browser will access it using the file protocol, like so: `/C:/Apache24/htdocs/index.php`. Given that your browser does not understand PHP, this will not work. The PHP code needs to be processed by the PHP module. The processing needs to be triggered by the Apache web server using the HTTP protocol, which is the language that Apache speaks. The resulting output should be HTML (possibly including CSS and JavaScript), which is then sent by Apache to your browser.

You can download each component individually from [httpd.apache.org](http://httpd.apache.org), [php.net](http://php.net) and [www.mysql.com](http://www.mysql.com). You then need to install and configure each component, which is a time-consuming and non-trivial task. A quicker and less error-prone approach is to download and install a preconfigured package that contains the Apache web server, PHP, MySQL and other useful tools ready to use. One such package is [XAMPP](http://www.apachefriends.org), which is available for all the main operating systems.

## 5.1.2. Windows

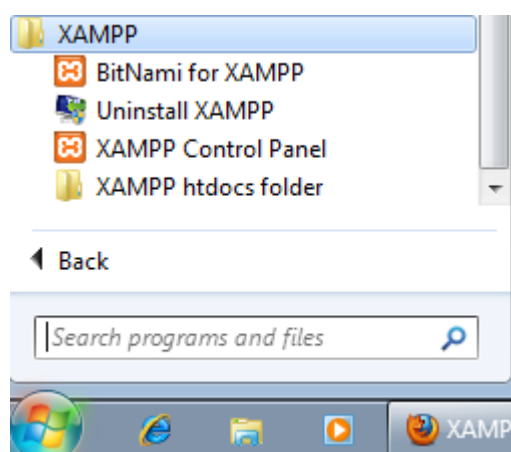
### Apache, PHP and MySQL/MariaDB

First we head over to the [XAMPP home page](http://www.apachefriends.org) and download and install the latest version. If you want to learn more about the idea behind XAMPP, this page is the best starting point. MariaDB is equivalent to MySQL for all intents and purposes. If you are interested in the differences, see [mariadb.com/kb/en/the-mariadb-library/mariadb-vs-mysql-features](http://mariadb.com/kb/en/the-mariadb-library/mariadb-vs-mysql-features).

#### Portable installation

If you want to have a portable web server on your USB stick or external hard drive that you can use anywhere, you can download a portable version of XAMPP. There are also [portable alternatives](#) to XAMPP.

Next, we start the XAMPP control panel:



We make sure that Apache and MySQL are started:

XAMPP Control Panel v3.2.1 [ Compiled: May 7th 2013 ]

## XAMPP Control Panel v3.2.1

Service	Module	PID(s)	Port(s)	Actions
<input type="checkbox"/>	Apache	1708 696	80, 443	Stop Admin Config Logs
<input type="checkbox"/>	MySQL	2596	3306	Stop Admin Config Logs
<input type="checkbox"/>	FileZilla	1860	21, 14147	Stop Admin Config Logs
<input type="checkbox"/>	Mercury			Start Admin Config Logs
<input type="checkbox"/>	Tomcat			Start Admin Config Logs

Config

Netstat

Shell

Explorer

Services

Help

Quit

01:07:41 [main] Initializing Control Panel

01:07:41 [main] Windows Version: Windows 7 Professional SP1 64-bit

01:07:41 [main] XAMPP Version: 1.8.3

01:07:41 [main] Control Panel Version: 3.2.1 [ Compiled: May 7th 2013 ]

01:07:41 [main] You are not running with administrator rights! This will work for most application stuff but whenever you do something with services there will be a security dialogue or things will break! So think about running this application with administrator rights!

01:07:41 [main] XAMPP Installation Directory: "c:\xampp\"

01:07:41 [main] Checking for prerequisites

01:07:41 [main] All prerequisites found

01:07:41 [main] Initializing Modules

01:07:44 [main] Enabling autostart for module "Apache"

01:07:44 [main] Enabling autostart for module "MySQL"

01:07:44 [main] Enabling autostart for module "FileZilla"

01:07:44 [main] Starting Check-Timer

01:07:44 [main] Control Panel Ready

01:07:47 [Apache] Autostart active: starting...

01:07:47 [Apache] Attempting to start Apache app...

01:07:47 [mysql] Autostart active: starting...

01:07:47 [mysql] Attempting to start MySQL app...

01:07:57 [filezilla] Autostart active: starting...

01:07:57 [filezilla] Attempting to start FileZilla app...

01:07:59 [Apache] Status change detected: running

01:07:59 [mysql] Status change detected: running

01:08:00 [filezilla] Status change detected: running

Finally, we verify that the installation was successful. The status page confirms that everything's running fine:

# XAMPP for Windows

**XAMPP**  
**1.8.3**  
[PHP: 5.5.1]

Welcome

**Status**

Security

Documentation

Components

Applications

**Php**

phpinfo()

CD Collection

Biorhythm

Instant Art

Phone Book

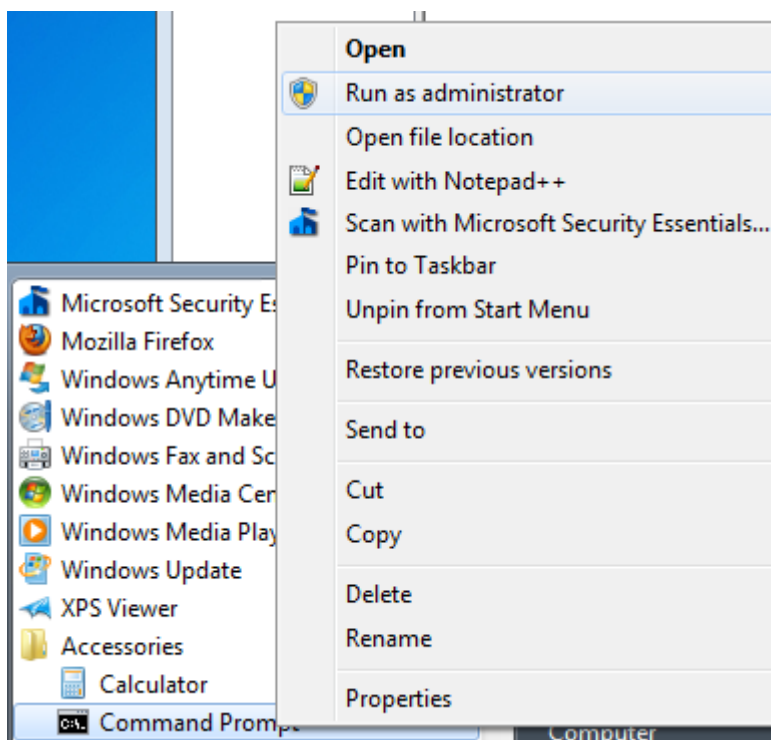
**Perl**

## XAMPP Status

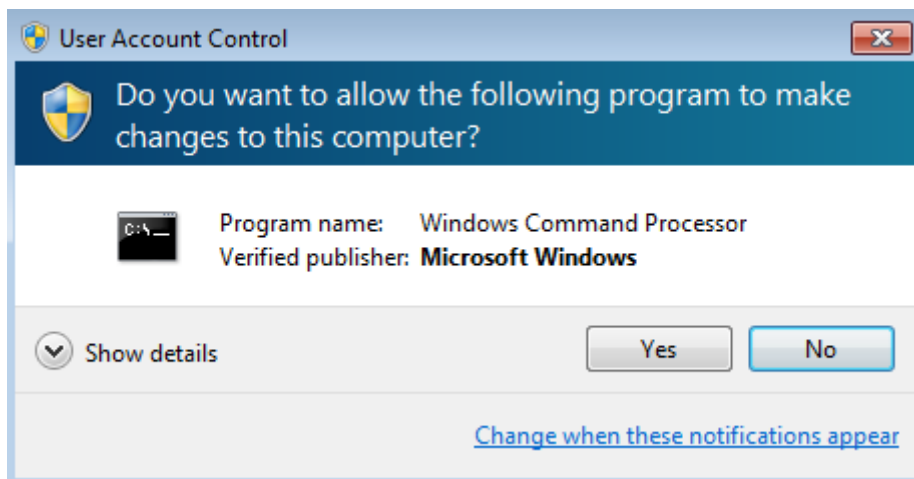
This page offers you one page to view all information about what's running

Component	Status	Hint
MySQL database	ACTIVATED	
PHP	ACTIVATED	
HTTPS (SSL)	ACTIVATED	
Common Gateway Interface (CGI)	ACTIVATED	
Server Side Includes (SSI)	ACTIVATED	
SMTP Service	DEACTIVATED	
FTP Service	ACTIVATED	
Tomcat Service	DEACTIVATED	

For Windows users: in order to avoid having to run the XAMPP control panel after each system restart, we can install Apache and MySQL as a service so that they start automatically. First we need to open an elevated command prompt (i.e. run a command prompt as system administrator):



Depending on our user account control settings (type `uac` in the start menu) Windows will ask for confirmation:



Now we need to execute the following commands:

```
Administrator: Command Prompt
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Windows\system32>cd\xampp\apache

C:\xampp\apache>apache_installservice.bat
Installing Apache2.4 as an Service
Installing the Apache2.4 service
The Apache2.4 service is successfully installed.
Testing httpd.conf....
Errors reported here must be corrected before the service can be started.
Now we Start Apache2.4 :)
The Apache2.4 service is starting.
The Apache2.4 service was started successfully.

Press any key to continue . . .

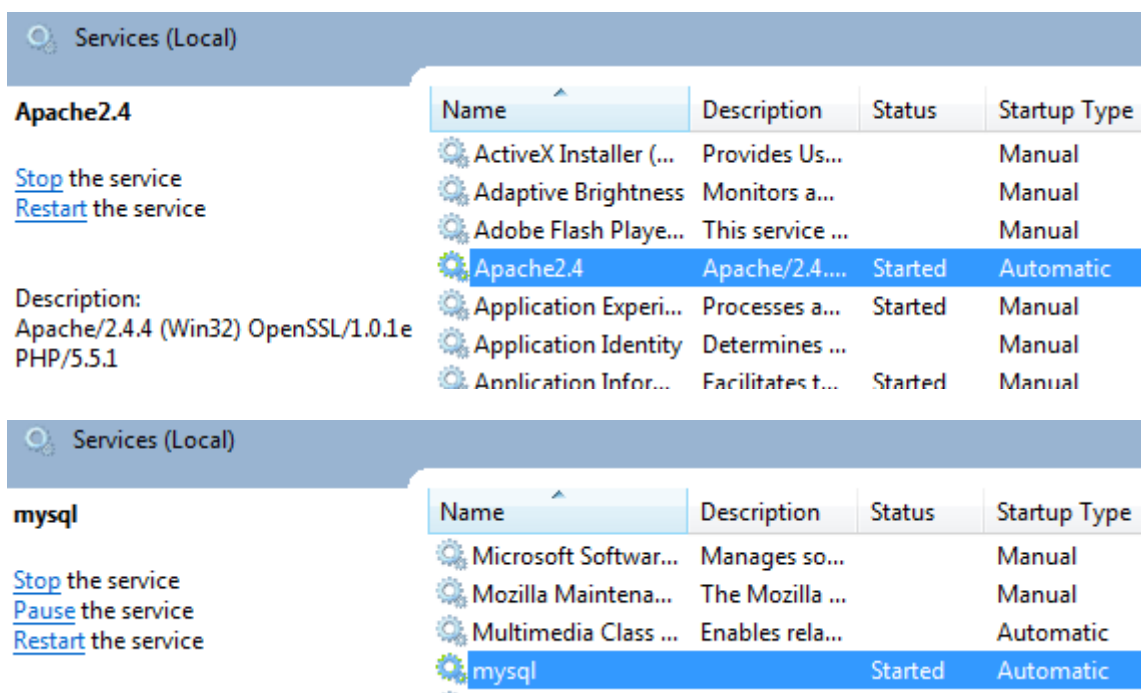
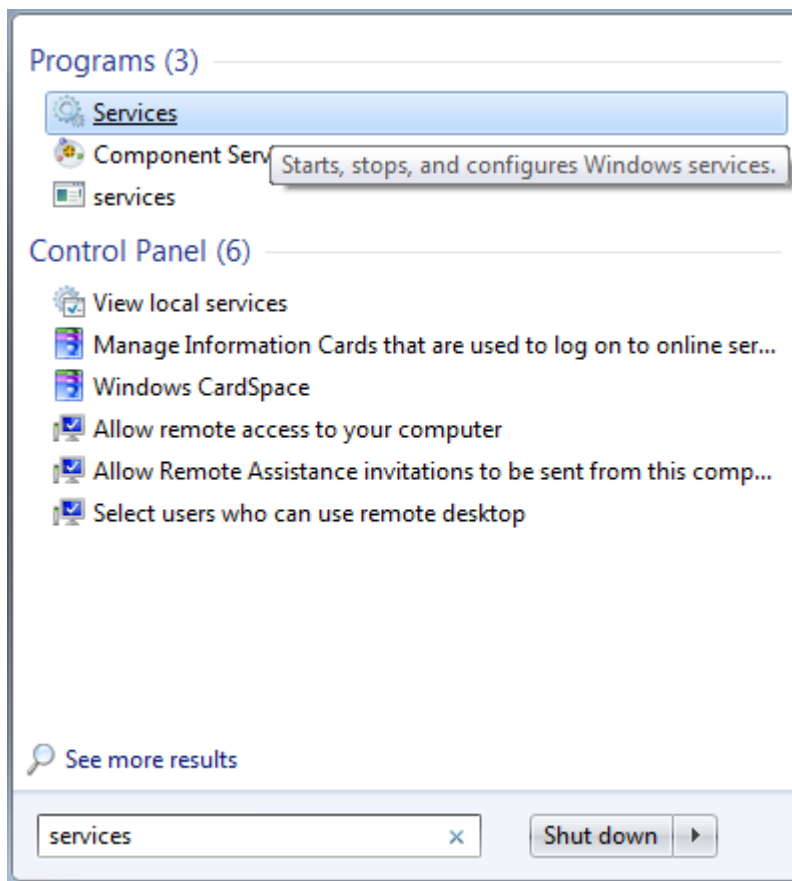
C:\xampp\apache>cd..\mysql

C:\xampp\mysql>mysql_installservice.bat
Installing MySQL as an Service
The system cannot find the file specified.
Service successfully installed.
Try to start the MySQL deamon as service ...
The mysql service is starting..
The mysql service was started successfully.

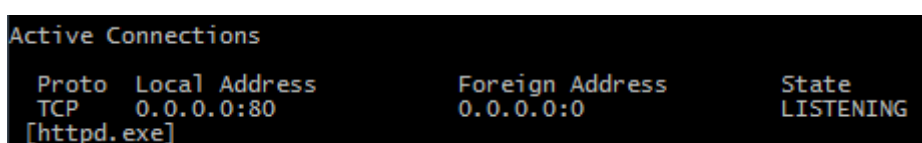
Press any key to continue . . .

C:\xampp\mysql>
```

To make sure that the services have been installed correctly, let's check:



If we get an error message saying that another process is using port 80, we need to check which process this is and kill it. Open an elevated command prompt and run `netstat -anb|more:`



The Apache web server, PHP and MySQL all have their own configuration files.

## Apache

In the folder `xampp\apache\conf` we find the file `httpd.conf`. This is the main Apache configuration file. Open it in a text editor and take a look. For now we won't change it.

In the folder `xampp\php` we find the file `php.ini`. This is the main PHP configuration file. Open it in a text editor and search for the setting `short_open_tag`. Read the explanation carefully and make sure that the setting is set to `off`. Next search for the setting `error_reporting`. Read the explanation carefully and make sure that the setting is set to `E_ALL` (and nothing else). Explain to someone else, why we want this particular setting.

In order to send emails from a Windows system, we could install a mail server. Here we'll limit ourselves to installing a fake sendmail program that sends email via an SMTP server that we specify in `php.ini`. Download the [zip file](#) and unpack it for instance into `C:\sendmail`. In `php.ini` you need to set `SMTP` to point to an SMTP server, e.g. `smtp.restena.lu`. Then you need to specify `smtp_port`. The standard value is 25. Finally, you need to specify `sendmail_path`. If you have installed fake sendmail in `C:\sendmail`, the value would be `"C:\sendmail\sendmail.exe -t"`. You also need to modify `sendmail.ini` by changing `smtp_server` to your SMTP server and `smtp_ssl` to `none`.

In order for the changes to take effect, we need to restart Apache. You can do this from the XAMPP Control Panel or Windows Services (run `services.msc`).

## MySQL

In the folder `mysql\bin` we find a sample config file `my.ini`. We won't change any MySQL config files manually.

At the moment, our MySQL installation is not secured, i.e. user root has no password. In order to avoid unauthorized access to our database, we set a root password as follows (using the command prompt):

```
cd \xampp\mysql\bin
mysql -u root
alter user 'root'@'localhost' identified with mysql_native_password by 'T2IF!secret';
exit
mysql -u root
mysql -u root -p
```

```

C:\> Command Prompt - mysql -u root -p
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.

C:\Users\Asterix>cd\xampp\mysql\bin

C:\xampp\mysql\bin>mysql -u root
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 7
Server version: 5.6.11 MySQL Community Server (GPL)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> set password for 'root'@'localhost'=password('T2IF!secret');
Query OK, 0 rows affected (0.00 sec)

mysql> exit
Bye

C:\xampp\mysql\bin>mysql -u root
ERROR 1045 (28000): Access denied for user 'root'@'localhost' (using password: N
O)

C:\xampp\mysql\bin>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 9
Server version: 5.6.11 MySQL Community Server (GPL)

Copyright (c) 2000, 2013, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>

```

Our database is still not secure, as we need to remove the empty user names, otherwise anyone can connect:

```

C:\ Command Prompt - mysql -u root -p
mysql> select host, user from mysql.user;
+-----+-----+
| host      | user  |
+-----+-----+
| linux     |      |
| linux     | root  |
| localhost |      |
| localhost | pma   |
| localhost | root  |
+-----+-----+
5 rows in set (0.00 sec)

mysql> DROP USER ''@'localhost';
Query OK, 0 rows affected (0.00 sec)

mysql> select host, user from mysql.user;
+-----+-----+
| host      | user  |
+-----+-----+
| linux     |      |
| linux     | root  |
| localhost | pma   |
| localhost | root  |
+-----+-----+
4 rows in set (0.00 sec)

mysql> drop user ''@'linux';
Query OK, 0 rows affected (0.00 sec)

mysql> select host, user from mysql.user;
+-----+-----+
| host      | user  |
+-----+-----+
| linux     | root  |
| localhost | pma   |
| localhost | root  |
+-----+-----+
3 rows in set (0.00 sec)

```

### phpMyAdmin

The main config file is `config.inc.php` in the folder `xampp\phpMyAdmin`. Open it in a text editor and make the changes required to obtain the following settings:

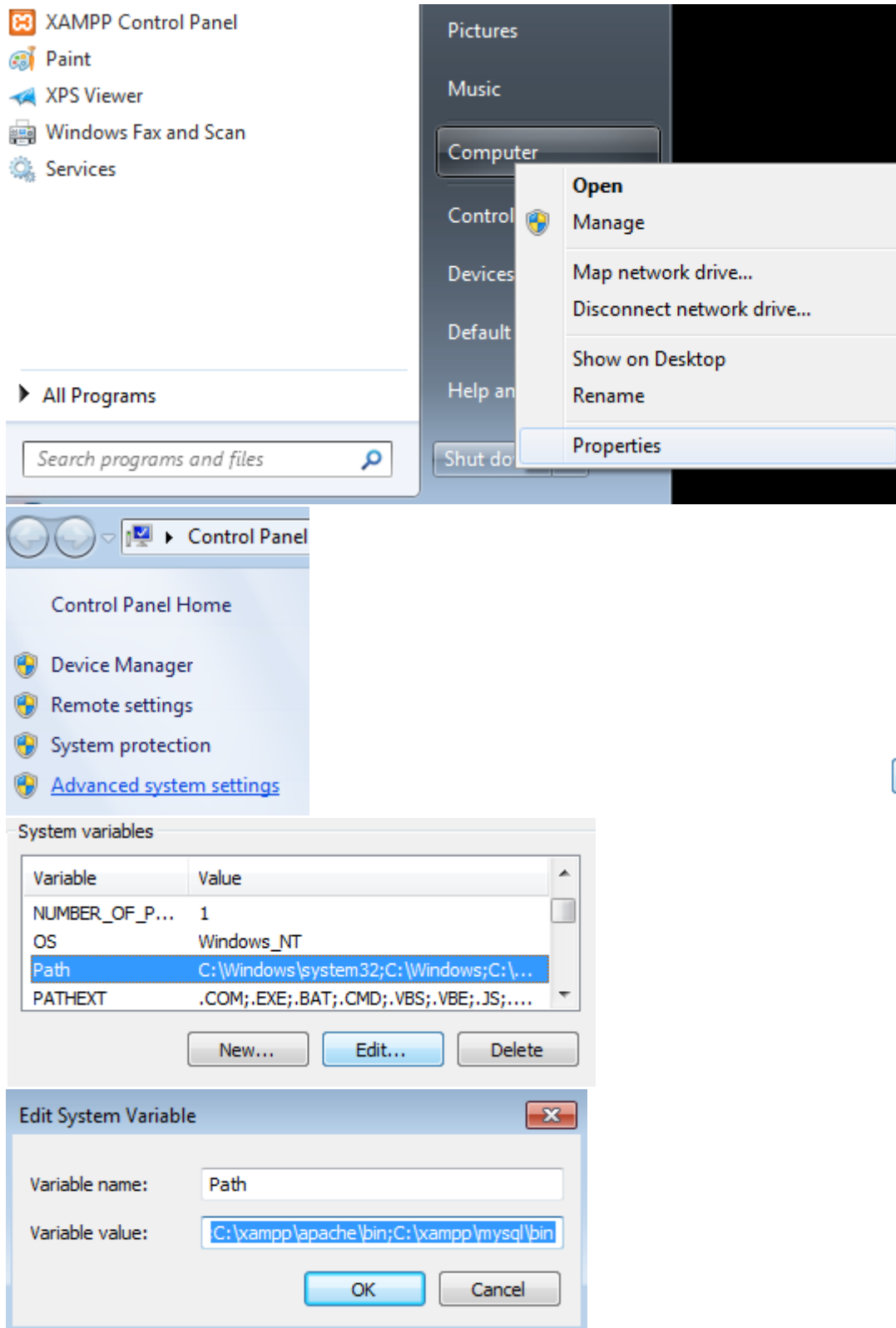
```

/* Authentication type and info */
$config['Servers'][$i]['auth_type'] = 'cookie';
$config['Servers'][$i]['user'] = 'root';
$config['Servers'][$i]['password'] = '';
$config['Servers'][$i]['extension'] = 'mysqli';
$config['Servers'][$i]['AllowNoPassword'] = false;

```

Let's take a look with [localhost/phpmyadmin](http://localhost/phpmyadmin). Enter the MySQL root credentials. As you can see, phpMyAdmin provides a convenient user interface to manage our databases. You might find it more comfortable than the MySQL command line and you can access it from anywhere. If you get a 'The secret passphrase in configuration (blowfish\_secret) is too short.' message you need to increase the size of the `$config['blowfish_secret']` string.

In order to be able to run the main XAMPP tools without having to specify the path or to enter the correct directory, we can add the required paths to the global PATH environment variable:



Now you can run mysql or httpasswd from anywhere using the command prompt.

#### *Adminer*

A fast alternative to phpMyAdmin is [Adminer](#). It is a single file and highly recommended.

#### *DBeaver*

A great portable multi-platform DB tool is [DBeaver](#).

### 5.1.3. Ubuntu

```
apt update
apt upgrade
apt install apache2 apache2-doc mysql-server php libapache2-mod-php php-mysql
phpmyadmin
```

Visit `localhost` with your browser. It should display the "Apache2 Ubuntu Default Page". At `localhost/manual` you can access the HTTP server documentation.

If PhpMyAdmin does not work after you've upgraded the PHP version disable the old PHP version and enable the new one. For instance, after upgrading from PHP 7.1 to 7.2 use (cf. [www.howtoforge.com/community/threads/phpmyadmin-missing-mysqli-extension.78307](http://www.howtoforge.com/community/threads/phpmyadmin-missing-mysqli-extension.78307)):

```
a2dismod php7.1
a2enmod php7.2
```

Check Apache status using `/etc/init.d/apache2 status`, `systemctl status apache2.service` and `journalctl -xe`.

#### Apache multi-processing modules (MPM)

Check which MPM is currently running using `apachectl -V|grep -i mpm`.

To switch, for instance in order to use HTTP2, see [www.linode.com/community/questions/17027/http2-and-apache-prefork-mpm](http://www.linode.com/community/questions/17027/http2-and-apache-prefork-mpm). Note that PHP directives will need to be moved to PHP-FPM user configuration files as described in the linked document.

If you want to install the latest version of MySQL you might need to add the latest repository to your system's package sources list as described in [dev.mysql.com/doc/mysql-apt-repo-quick-guide/en/#apt-repo-fresh-install](http://dev.mysql.com/doc/mysql-apt-repo-quick-guide/en/#apt-repo-fresh-install) and [www.tecmint.com/install-mysql-8-in-ubuntu](http://www.tecmint.com/install-mysql-8-in-ubuntu).

Improve MySQL security by running `mysql_secure_installation`. Remove all anonymous users as well as the test DB and disable remote root login. Then study [www.acunetix.com/websitesecurity/securing-mysql-server-ubuntu-16-04-lts](http://www.acunetix.com/websitesecurity/securing-mysql-server-ubuntu-16-04-lts) and [www.digitalocean.com/community/tutorials/how-to-secure-mysql-and-mariadb-databases-in-a-linux-vps](http://www.digitalocean.com/community/tutorials/how-to-secure-mysql-and-mariadb-databases-in-a-linux-vps).

To disable MYSQL root login without password use `ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY 'ANY_PASSWORD';`.

If you forgot the root password, see [stackoverflow.com/questions/42153059/mysqld-safe-directory-var-run-mysqld-for-unix-socket-file-dont-exists](http://stackoverflow.com/questions/42153059/mysqld-safe-directory-var-run-mysqld-for-unix-socket-file-dont-exists).

To enable caching, see [www.inmotionhosting.com/support/website/htaccess/apache-module-mod-expires](http://www.inmotionhosting.com/support/website/htaccess/apache-module-mod-expires) and [www.digitalocean.com/community/tutorials/how-to-configure-apache-content-caching-on-ubuntu-14-04](http://www.digitalocean.com/community/tutorials/how-to-configure-apache-content-caching-on-ubuntu-14-04).

To see which modules are enabled, use `apache2ctl -M`.

To install Xdebug, see [xdebug.org/docs](http://xdebug.org/docs).

[www.addedbytes.com/articles/for-beginners/url-rewriting-for-beginners](http://www.addedbytes.com/articles/for-beginners/url-rewriting-for-beginners)

## Security

### `mod_evasive`

[blog.rapid7.com/2017/04/09/how-to-configure-modevasive-with-apache-on-ubuntu-linux](http://blog.rapid7.com/2017/04/09/how-to-configure-modevasive-with-apache-on-ubuntu-linux)

[serverfault.com/questions/614707/how-to-protect-a-folder-from-browser-access-but-allow-php-scripts-to-access-it](http://serverfault.com/questions/614707/how-to-protect-a-folder-from-browser-access-but-allow-php-scripts-to-access-it)

[stackoverflow.com/questions/19124109/protect-pdf-docs-from-being-directly-accessed-in-url](http://stackoverflow.com/questions/19124109/protect-pdf-docs-from-being-directly-accessed-in-url)

[stackoverflow.com/questions/2187200/using-php-apache-to-restrict-access-to-static-files-html-css-img-etc](http://stackoverflow.com/questions/2187200/using-php-apache-to-restrict-access-to-static-files-html-css-img-etc)

[stackoverflow.com/questions/3697748/fastest-way-to-serve-a-file-using-php/3731639#3731639](http://stackoverflow.com/questions/3697748/fastest-way-to-serve-a-file-using-php/3731639#3731639)

[www.gravitywell.co.uk/latest/how-to/posts/securing-your-downloads-with-php-and-mod-xsendfile/](http://www.gravitywell.co.uk/latest/how-to/posts/securing-your-downloads-with-php-and-mod-xsendfile/)

[www.h3xed.com/programming/how-to-use-x-sendfile-with-php-apache](http://www.h3xed.com/programming/how-to-use-x-sendfile-with-php-apache)

[php.net/manual/en/features.http-auth.php](http://php.net/manual/en/features.http-auth.php)

[www.maketecheasier.com/securing-apache-ubuntu](http://www.maketecheasier.com/securing-apache-ubuntu)

[httpd.apache.org/docs/2.4/custom-error.html](http://httpd.apache.org/docs/2.4/custom-error.html)

[httpd.apache.org/docs/2.4/misc/security\\_tips.html](http://httpd.apache.org/docs/2.4/misc/security_tips.html)

[help.dreamhost.com/hc/en-us/articles/226327268-The-most-important-steps-to-take-to-make-an-Apache-server-more-secure](http://help.dreamhost.com/hc/en-us/articles/226327268-The-most-important-steps-to-take-to-make-an-Apache-server-more-secure)

## HTTP2

[http2.pro/doc/Apache](http://http2.pro/doc/Apache)

[tools.keycdn.com/http2-test](http://tools.keycdn.com/http2-test)

[daniel.haxx.se/http2](http://daniel.haxx.se/http2)

[icing.github.io/mod\\_h2/howto.html](http://icing.github.io/mod_h2/howto.html)

[httpwg.org/specs/rfc7540.html](http://httpwg.org/specs/rfc7540.html)

[httpd.apache.org/docs/current/howto/http2.html](http://httpd.apache.org/docs/current/howto/http2.html)

[stackoverflow.com/questions/52991676/http-2-not-working-on-rhel-6-apache-2-4-34](http://stackoverflow.com/questions/52991676/http-2-not-working-on-rhel-6-apache-2-4-34)

[helgeklein.com/blog/2018/11/enabling-http-2-in-apache-on-ubuntu-18-04](http://helgeklein.com/blog/2018/11/enabling-http-2-in-apache-on-ubuntu-18-04)

Use [mozilla.github.io/server-side-tls/ssl-config-generator](https://mozilla.github.io/server-side-tls/ssl-config-generator) to get a cipher config that works with HTTP/2.

## Reverse proxy

[httpd.apache.org/docs/2.4/howto/reverse\\_proxy.html](http://httpd.apache.org/docs/2.4/howto/reverse_proxy.html)

Enable `mod_proxy_wstunnel` via `a2enmod proxy_wstunnel`. You might have to enable a few additional modules, see [stackoverflow.com/questions/23931987/apache-proxy-no-protocol-handler-was-valid](https://stackoverflow.com/questions/23931987/apache-proxy-no-protocol-handler-was-valid).

Insert the following into your virtual host config, replacing 9000 with your chosen port number:

```
ProxyRequests Off
ProxyPreserveHost On
SSLProxyEngine on
ProxyPass /wss/ wss://localhost:9000/
ProxyPassReverse /wss/ wss://localhost:9000/
ProxyPass / https://localhost:9000/
ProxyPassReverse / https://localhost:9000/
```

This will however prevent the web server from getting the correct client IP address, as the IP will be the one of the machine that the reverse proxy is running on. See [2bits.com/articles/correct-client-ip-address-reverse-proxy-or-content-delivery-network-cdn.html](https://2bits.com/articles/correct-client-ip-address-reverse-proxy-or-content-delivery-network-cdn.html) for solutions.

Or create a reverse proxy in Node as explained in [blog.wathmal.me/run-apache-with-node-js-reverse-proxy](https://blog.wathmal.me/run-apache-with-node-js-reverse-proxy).

[stackoverflow.com/questions/27526281/websockets-and-apache-proxy-how-to-configure-mod-proxy-wstunnel](https://stackoverflow.com/questions/27526281/websockets-and-apache-proxy-how-to-configure-mod-proxy-wstunnel)

The `!` directive is useful in situations where you don't want to reverse-proxy a subdirectory, see [httpd.apache.org/docs/2.4/mod/mod\\_proxy.html](http://d.apache.org/docs/2.4/mod/mod_proxy.html).

## Caching

To prevent the browser from caching a specific resource, we can append a `?` followed by a random number as described in [www.willmaster.com/library/web-content-prep/preventing-browser-cache.php](http://www.willmaster.com/library/web-content-prep/preventing-browser-cache.php).

For a more general explanation of web caching see [www.mnot.net/cache\\_docs](http://www.mnot.net/cache_docs).

## VirtualBox

If you want to install Ubuntu with Apache in a virtual machine:

1. Create a new VM and add a rule for NAT port forwarding setting the host and guest port to 22. You can leave the IP fields empty.
2. Download Ubuntu Server from [ubuntu.com/#download](https://ubuntu.com/#download) and install it.
3. Enable root login via `sudo passwd root`.
4. Enable the firewall using `ufw enable`. Then add rules to allow SSH, HTTP and HTTPS connections via `ufw allow 22/tcp`, `ufw allow 80/tcp` and `'ufw allow 443/tcp'`.
5. Install the SSH server using `apt install openssh-server`, then follow [linuxconfig.org/allow-ssh-root-login-on-ubuntu-18-04-bionic-beaver-linux](https://linuxconfig.org/allow-ssh-root-login-on-ubuntu-18-04-bionic-beaver-linux).
6. You should now be able to connect via SSH from your host to the server using `ssh`

root@localhost.

## 5.2. PHP

### 5.2.1. Introduction

From [php.net/manual/en/intro-what-is.php](http://php.net/manual/en/intro-what-is.php):

PHP is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML.

A PHP script can either work without output to the browser, or, in the usual case, send HTML, CSS and JavaScript to the browser. In any case, the client never gets to see the underlying PHP code.

What do we need PHP for in the context of web application development?

1. To access a server database. For instance, if we want to build the next Amazon or YouTube, we need a database where we store the data of our gazillion users and products.
2. To access data on the Internet. Remember that for security reasons the browser is severely restricted in terms of accessing content from other sites (same-origin policy (cf. [Cross-origin requests](#))). In PHP, we have no limitations, unless our host provider has put some in place (which many of them do), in which case we switch to a host provider that does not or host our own server.
3. To provide services that can be accessed from anywhere (see [Web Application Programming Interfaces](#)).
4. To run code that we do not want the client to see and/or manipulate.

A good way to start learning PHP is to use [www.phpschool.io](http://www.phpschool.io). The latest version of PHP is version 7. To see what's new read [secure.php.net/manual/en/migration70.new-features.php](http://secure.php.net/manual/en/migration70.new-features.php) and [www.tutorialspoint.com/php7](http://www.tutorialspoint.com/php7).

A PHP script has a file name ending with `.php`.

### 5.2.2. Hello world

[students.btsi.lu/evegi144/WAD/PHP/helloworld1.php](http://students.btsi.lu/evegi144/WAD/PHP/helloworld1.php)

```
<?php
    echo 'Hello world';
?>
```

`echo` is a PHP construct that outputs all parameters. This script is parsed by the Apache PHP module and sends the string `Hello world` to the browser. Try it out for yourself now! This is of course not a valid HTML5 page (run the validator), but still, the browser will display the text.

There's also a short form of echo:

[students.btsi.lu/evegi144/WAD/PHP/helloworld2.php](https://students.btsi.lu/evegi144/WAD/PHP/helloworld2.php)

```
<?= 'Hello world'?>
```

If we want to create a valid HTML5 page, we can mix HTML and PHP:

[students.btsi.lu/evegi144/WAD/PHP/helloworldvalid.php](https://students.btsi.lu/evegi144/WAD/PHP/helloworldvalid.php)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Hello World</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <?php
      echo 'Hello world!';
    ?>
  </body>
</html>
```

Take a look at the source in your browser. What do you see? What happened to the PHP part? How do you explain your observation?

From now on, the HTML part will not be shown, unless it is relevant.

### 5.2.3. Logging

In your home directory on Foxi you have a directory named `.log` where the PHP interpreter saves errors generated during the execution of your PHP scripts in a file called `error.log`. It's a good idea to monitor this file. To do this, log in to Foxi with Putty and run the command `tail -f /www/<class>/<username>/.log/error.log`.

Use the PHP function `error_log` to log an error message to the log file.

It's a good idea to set the error level in your script to `E_ALL` (cf. [php.net/manual/en/errorfunc.constants.php](https://php.net/manual/en/errorfunc.constants.php)), irrespective of the default setting on the server. This can be done using the `error_reporting` function.

Let's run an erroneous script and see what happens to our log file:

[students.btsi.lu/evegi144/WAD/PHP/error1.php](https://students.btsi.lu/evegi144/WAD/PHP/error1.php)

```
<?php
  eccho 'abc';
?>
```

## 5.2.4. Variables and data types

[students.btsi.lu/evegi144/WAD/PHP/variables1.php](https://students.btsi.lu/evegi144/WAD/PHP/variables1.php)

```
<?php
$name = 'Asterix';
echo "My name is $name<br>";
echo 'My name is $name<br>';
unset($name);
echo "My name is $name<br>";
?>
```

This is a simple example of a variable declaration and usage. A variable is used to temporarily store data under a given name.



Variable names in PHP are case-sensitive and always start with **\$** followed by a letter or an underscore, then any combination of letters, numbers and the underscore character (cf. [php.net/manual/en/language.variables.basics.php](https://php.net/manual/en/language.variables.basics.php)).

**unset** is used to destroy a variable.

Strings are embedded in `"` or `'` in PHP. There is however a significant difference between the two. Can you see it in the example above?



PHP will not perform variable substitution inside single-quoted strings and won't even replace most escape characters (except `\`). In double-quoted strings, PHP will replace variables with their current value.

Take a look at the output of the following script. What happened?

[students.btsi.lu/evegi144/WAD/PHP/variables2.php](https://students.btsi.lu/evegi144/WAD/PHP/variables2.php)

```
<?php
$fruit = 'apple';
echo "I've bought 5 $fruits.<br>";
echo "I've bought 5 ${fruit}s.<br>";
echo "I've bought 5 {$fruit}s.";
?>
```

By adding **s** we have changed the variable name and the PHP interpreter now looks for a variable named **\$fruits**, which does not exist (line 3). We can use braces to tell PHP where the variable name ends (cf. lines 4 and 5).



PHP has nine **data types**: **string**, **integer**, **float**, **boolean**, **array**, **object**, **callable**, **resource** and **NULL**. PHP considers **0**, **0.0**, **"**, **"0"** and **NULL** to be equivalent to **FALSE** and everything else to be **TRUE**.

From [php.net/manual/en/language.types.type-juggling.php](http://php.net/manual/en/language.types.type-juggling.php):

A variable's type is determined by the context in which the variable is used. That is to say, if a string value is assigned to variable `$var`, `$var` becomes a string. If an integer value is then assigned to `$var`, it becomes an integer.

With `settype` we can change the type of a variable explicitly.

To check the type and value of an expression, use the `var_dump` function. To get a human-readable representation of a type for debugging, use the `gettype` function. To check for a certain type use `is_int`, `is_array` etc. functions.

## String concatenation

Strings can be concatenated using the `.` operator.

[students.btsi.lu/evegi144/WAD/PHP/strings1.php](http://students.btsi.lu/evegi144/WAD/PHP/strings1.php)

```
<?php
    $phrase = "The glass is half full";
    echo 'Quote of the day: ' . $phrase . '<br>';

    // Alternatively, we can escape the ".
    echo "Quote of the day: \"$phrase\"";
?>
```

### 5.2.5. Heredoc

To deal with long strings, we can use `heredoc` syntax. It allows us to define our own string delimiter so that we do not have to use quotes. The string delimiter, `EOT` in our example, needs to be in the first column on a line by itself. Spaces or tabs are not allowed.

[students.btsi.lu/evegi144/WAD/PHP/heredoc1.php](http://students.btsi.lu/evegi144/WAD/PHP/heredoc1.php)

```
<?php
$string = "Hello world!";
echo <<<EOT
    <!DOCTYPE html>
    <html lang=en>
        <head>
            <title>$string</title>
            <meta charset=UTF-8>
        </head>
        <body>
            $string
        </body>
    </html>
EOT;
?>
```

However, the same can be achieved without the heredoc hassle given that in PHP strings can span multiple lines. Here is the same example without heredoc:

[students.btsi.lu/evegi144/WAD/PHP/longecho1.php](https://students.btsi.lu/evegi144/WAD/PHP/longecho1.php)

```
<?php
$string = "Hello world!";
echo "<!DOCTYPE html>
    <html lang=en>
        <head>
            <title>$string</title>
            <meta charset=UTF-8>
        </head>
        <body>
            $string
        </body>
    </html>";
?>
```

### 5.2.6. Constants

Constants are like variables, except that once they are defined, they cannot be undefined or changed. Their name does not start with a **\$**. A constant is case-sensitive by default. By convention, constant identifiers are always uppercase. They are defined using **const** or the **define** function:

[students.btsi.lu/evegi144/WAD/PHP/constants1.php](https://students.btsi.lu/evegi144/WAD/PHP/constants1.php)

```
<?php
const SECONDSPERDAY = 86400;
define("DAYOFTHEWEEK", 'Sunday');
echo SecondsPerDay . '<br>';
echo SECONDSPERDAY . '<br>';
echo DAYOFTHEWEEK;
?>
```

The `define()` function can be given a third parameter to turn off case-sensitivity:

[students.btsi.lu/evegi144/WAD/PHP/constants2.php](https://students.btsi.lu/evegi144/WAD/PHP/constants2.php)

```
<?php
define("SECONDSPERDAY", 86400, true);
echo SecondsPerDay;
?>
```

Constants are automatically global across the entire script, unlike variables.

## 5.2.7. Comments

`#` or `//` tell PHP to ignore the rest of the line. For multi line comments, use `/* */`.

[students.btsi.lu/evegi144/WAD/PHP/comments1.php](https://students.btsi.lu/evegi144/WAD/PHP/comments1.php)

```
<?php
echo 'Comments are useful'; // This is a comment.
# This is a comment too.
/* This is a comment
that stretches over
several lines.
*/
?>
```

If we want to enable automatic documentation generation, we can use [PHPDoc](#).

## 5.2.8. Operators

Detailed information and examples related to operators can be found at [php.net/manual/en/language.operators.php](https://php.net/manual/en/language.operators.php) and [www.w3schools.com/php/php\\_operators.asp](https://www.w3schools.com/php/php_operators.asp).



Pay particular attention to the equal operator.

A single `=` is used to assign the right value to the left operand.

`==` and `===` are comparison operators. The former may perform an implicit type conversion whereas the latter will not.

```
<?php
var_dump($x); // $x is not defined
echo '<br>';
$x = 24; // Assignment -> variable $x now has the value 24
var_dump($x);
echo '<br>';
if ("2" === 2) echo '"2" === 2'; // Check type and value without implicit
conversion.
else echo '"2" !== 2';
echo '<br>';
if ("2" == 2) echo '"2" == 2'; // Check value, if necessary after implicit type
conversion.
else echo '"2" != 2';
?>
```

When combining several operators, it is helpful to know their [precedence](#).

### @ operator

If we put this operator in front of an expression in PHP, any error messages that might be generated by that expression will be ignored (cf. [php.net/manual/en/language.operators.errorcontrol.php](http://php.net/manual/en/language.operators.errorcontrol.php)).

## 5.2.9. Conditional statements

### if else

We can take decisions based on conditions.

[students.btsi.lu/evegi144/WAD/PHP/conditionalstatement1.php](https://students.btsi.lu/evegi144/WAD/PHP/conditionalstatement1.php)

```
<?php
$password = "abc";
if ($password === "secret") echo "You may proceed!<br>";
else echo "No entry!<br>";

$mark = rand(-100, 100); // Generate a random number from [-100, 100].
echo "Mark: $mark<br>";
if ($mark < 1 || $mark > 60) echo "Something went wrong!"; // || -> logical or
else if ($mark === 60) echo "Excellent!!";
else if ($mark >= 50) echo "Very good!";
else if ($mark >= 40) echo "Good";
else if ($mark >= 30) echo "Pass";
else if ($mark >= 20) echo "Fail";
else if ($mark >= 10) echo "Bad";
else echo "Very bad!";
?>
```

## Ternary operator

Like JavaScript, PHP provides a shortcut version of `if else`:

[students.btsi.lu/evegi144/WAD/PHP/conditionalstatement2.php](https://students.btsi.lu/evegi144/WAD/PHP/conditionalstatement2.php)

```
<?php
    echo (time() % 2 === 0 ? "Even" : "Uneven") . " number of seconds.";

    // The previous line is functionally equivalent to the following:
    if (time() % 2 === 0) echo "Even number of seconds.";
    else echo "Uneven number of seconds.";
?>
```

## 5.2.10. Loops

### for

The for loop consists of three parts: the declaration, the condition and the action. The declaration sets the variable `$i` to 1. The condition checks whether `$i` is less than 10. As long as the condition is true, we increment `$i` by 1 after every iteration.

[students.btsi.lu/evegi144/WAD/PHP/for1.php](https://students.btsi.lu/evegi144/WAD/PHP/for1.php)

```
<?php
    for ($i = 1; $i < 10; $i++) echo "Number $i<br>";
    echo "Variable $i now has the value $i.";
?>
```

### while



We need to verify that a loop terminates, i.e. that there will be a time, where the condition won't be true anymore. Otherwise, we end up with an infinite loop, meaning our program will never terminate!

[students.btsi.lu/evegi144/WAD/PHP/while1.php](https://students.btsi.lu/evegi144/WAD/PHP/while1.php)

```
<?php
    $num1 = rand(1, 100); # Generate a random number from [1, 100].
    $num2 = rand(1, 100); # Generate a random number from [1, 100].
    while ($num1 < 90 && $num2 > 10) { # Loop as long as the condition is true.
        echo "Life goes on<br>";
        $num1 = rand(1, 100); # Generate a random number from [1, 100].
        $num2 = rand(1, 100); # Generate a random number from [1, 100].
    }
?>
```

## break and continue

**break** exits the loop immediately. **continue** skips the rest of the current iteration and goes on to the next.

[students.btsi.lu/evegi144/WAD/PHP/breakandcontinue1.php](https://students.btsi.lu/evegi144/WAD/PHP/breakandcontinue1.php)

```
<?php
for ($i = 1; $i < 10; $i++) {
    if ($i % 2 === 0) continue;
    if ($i > 8) break;
    echo "$i ";
}
?>
```

## Nested loops

[students.btsi.lu/evegi144/WAD/PHP/nestedloop1.php](https://students.btsi.lu/evegi144/WAD/PHP/nestedloop1.php)

```
<?php
for ($i = 1; $i < 10; $i++) {
    for ($j = 1; $j < 10; $j++) echo "$i ";
    echo "<br>";
}
?>
```

### 5.2.11. Arrays

An **array** in PHP is a container. You can use it to store values and/or variables inside, which are called **elements**. Each element has:

1. a **key**, which can be an integer or a string.
2. a **value**, which can be any data type or a variable.

We can create an array using the **array** function or using the **[]** operator. We can also add elements to an array using **[]**.

The function **print\_r** is very helpful to visualize the contents of an array. To optimize the readability of the output, we can set the function's second parameter to **TRUE** and embed it in a **pre** tag.

[students.btsi.lu/evegi144/WAD/PHP/arrays1.php](https://students.btsi.lu/evegi144/WAD/PHP/arrays1.php)

```
<?php
    $car_brands = ["Audi", "BMW", "Renault", "Nissan"]; // Create an array with 4
elements.
    echo '<pre>' . print_r($car_brands, true) . '</pre>';
    $car_brands[] = "Hyundai";
    echo '<pre>' . print_r($car_brands, true) . '</pre>';
?>
```

The function `count` (or `sizeof`, which is just an alias of `count`) tells us the number of elements in an array.

`in_array` returns `TRUE` if a given element is in the array, `FALSE` otherwise. Note that if we set the third parameter to `TRUE`, it will check both the type and value of the element.

The elements of an array variable are variables too, so we can delete them using `unset`. However, this will create an empty space in our array, although the length will be correct. To completely remove an array element, we can use `array_splice`.

If we want to assign the elements of an array to individual variables, the `list` command comes in handy.

PHP standard operators can be used with arrays as explained at [secure.php.net/manual/en/language.operators.array.php](https://secure.php.net/manual/en/language.operators.array.php).

PHP offers a number of useful functions to change the internal pointer of an array using `reset`, `current`, `each`, `end`, `next` and `prev`. See [php.net/manual/en/function.reset.php](https://php.net/manual/en/function.reset.php) for usage examples.

To get the full list of PHP array functions, see [php.net/manual/en/ref.array.php](https://php.net/manual/en/ref.array.php).

Study the examples at [php.net/manual/en/language.types.array.php](https://php.net/manual/en/language.types.array.php).

Take your time to study the following examples:

[students.btsi.lu/evegi144/WAD/PHP/arrays2.php](https://students.btsi.lu/evegi144/WAD/PHP/arrays2.php)

```
<?php
    $car_brands = ["Audi", "BMW", "Renault", "Nissan"]; // Create an array with 4
elements.
    echo "Number of array elements: " . count($car_brands) . "<br>";
    echo '<pre>' . print_r($car_brands, true) . '</pre>';

    if (in_array("Audi", $car_brands)) echo 'Audi is part of our brands.';
    else echo 'Audi is not part of our brands.';
    echo "<br>";

    if (in_array("Mercedes", $car_brands)) echo 'Mercedes is part of our brands.';
    else echo 'Mercedes is not part of our brands.';
    echo "<br>";

    $car_brands[] = "Hyundai";
```

```

echo "Number of array elements: " . count($car_brands) . "<br>";
echo '<pre>' . print_r($car_brands, true) . '</pre>';

unset($car_brands[3]); // Delete the fourth element in the array.
echo "Number of array elements: " . count($car_brands) . "<br>";
echo '<pre>' . print_r($car_brands, true) . '</pre>';

$car_brands[3] = "Nissan";
echo $car_brands[3] . ' ** ' . $car_brands[4] . '<br>';
echo '<pre>' . print_r($car_brands, true) . '</pre>';

array_splice($car_brands, 3, 1);
echo "Number of array elements: " . count($car_brands) . "<br>";
echo '<pre>' . print_r($car_brands, true) . '</pre>';

$car_brands[] = "VW";
echo "Number of array elements: " . count($car_brands) . "<br>";
echo '<pre>' . print_r($car_brands, true) . '</pre>';
echo "The third brand is $car_brands[2]<br>";

$schools[] = "LAM";
echo '<pre>' . print_r($schools, true) . '</pre>';

$numbers1 = array(1, 2, 3, 4);
if (in_array('2', $numbers1, TRUE)) echo "'2' is in " . '$numbers1';
else echo "'2' is not in " . '$numbers1';
echo '<br>';
if (in_array('2', $numbers1)) echo "'2' is in " . '$numbers1';
else echo "'2' is not in " . '$numbers1';
echo '<br>';

$numbers2 = [2, 4, 8, 3, 1, 5, 6, 7];
echo '<pre>' . print_r($numbers1 + $numbers2, true) . '</pre>';

list($n1, $n2, $n3, $n4) = $numbers1;
echo "<br>The result of the magic formula is " . ($n1 + $n2 * $n3 / $n4);
?>

```

## Associative arrays

In addition to specifying array values, we can also specify our own keys as well as mix own keys with automatically generated ones:

[students.btsi.lu/evegi144/WAD/PHP/arrays3.php](https://students.btsi.lu/evegi144/WAD/PHP/arrays3.php)

```
<?php
    $car_brands = ["a" => "Audi", "b" => "BMW", "n" => "Nissan", "r" => "Renault"];
    echo '<pre>' . print_r($car_brands, true) . '</pre>';
    echo "I like {$car_brands['b']}.";
    $car_brands['v'] = 'VW';
    echo '<pre>' . print_r($car_brands, true) . '</pre>';
    $car_brands[] = 'Tesla';
    echo '<pre>' . print_r($car_brands, true) . '</pre>';
?>
```

## Looping through arrays with **foreach**

**foreach** allows us to iterate through an array or object (we'll get to objects later on):

[students.btsi.lu/evegi144/WAD/PHP/foreach1.php](https://students.btsi.lu/evegi144/WAD/PHP/foreach1.php)

```
<?php
    $car_brands = ["a" => "Audi", "b" => "BMW", "n" => "Nissan", "r" => "Renault"];
    foreach ($car_brands as $brand) echo "$brand<br>"; // Ignore the keys and use only
values.
    foreach ($car_brands as $key => $val) echo "$key = $val<br>"; // Here we use both.
    #We can also use foreach with a standard array.
    $arr = [random_int(1, 100), random_int(1, 100), random_int(1, 100), random_int(1,
100)];
    $sum = 0;
    foreach ($arr as $elem) $sum += $elem;
    echo "The sum of all array elements is $sum.<br>";
    foreach ($arr as $key => $val) echo "$key = $val<br>";
?>
```

Of course, we can use a standard for loop to do exactly the same:

[students.btsi.lu/evegi144/WAD/PHP/forarray1.php](https://students.btsi.lu/evegi144/WAD/PHP/forarray1.php)

```

<?php
$car_brands = ["a" => "Audi", "b" => "BMW", "n" => "Nissan", "r" => "Renault"];
for ($i = 0; $i < count($car_brands); $i++)
    echo "{$car_brands[array_keys($car_brands)[$i]]}<br>";
for ($i = 0; $i < count($car_brands); $i++) {
    $key = array_keys($car_brands)[$i];
    echo "$key = {$car_brands[$key]}<br>";
}

$arr = [random_int(1, 100), random_int(1, 100), random_int(1, 100), random_int(1,
100)];
$sum = 0;
for ($i = 0; $i < count($arr); $i++) $sum += $arr[$i];
echo "The sum of all array elements is $sum.<br>";
for ($i = 0; $i < count($arr); $i++) {
    $key = array_keys($arr)[$i];
    echo "$key = {$arr[$key]}<br>";
}
?>

```

## Multidimensional arrays

Array elements can be arrays, so we can create multidimensional arrays.

[students.btsi.lu/evegi144/WAD/PHP/arrays4.php](https://students.btsi.lu/evegi144/WAD/PHP/arrays4.php)

```

<?php
// Two-dimensional array
$matrix = [[1, 2, 3, 4, 5], [6, 7, 8, 9, 10]];
echo '<pre>' . print_r($matrix, true) . '</pre>';
echo '<table border=1>';
foreach ($matrix as $row) {
    echo '<tr>';
    foreach ($row as $elem) echo "<td>$elem</td>";
    echo '</tr>';
}
echo '</table>';

// Three-dimensional mixed associative/non-associative array
$grades = ["WSERS1" => ["Asterix" => [56, 55, 57], "Obelix" => [34, 24, 41]],
           "WSERS2" => ["Asterix" => [58, 59, 60], "Obelix" => [39, 43, 48]]];
echo '<pre>' . print_r($grades, true) . '</pre>';
echo '<pre>' . print_r($grades['WSERS1'], true) . '</pre>';
echo '<pre>' . print_r($grades['WSERS1']['Asterix'], true) . '</pre>';
echo '<pre>' . print_r($grades['WSERS1']['Asterix'][0], true) . '</pre>';
foreach ($grades as $subject => $students) {
    echo "Subject: $subject<br>";
    echo '<pre>' . print_r($students, true) . '</pre>';
    foreach ($students as $student => $student_grades) {
        echo "$student's grades:";
        foreach ($student_grades as $grade) echo " $grade";
        echo "<br>";
    }
}
?>

```

## 5.2.12. Functions

**Functions** are a key element of effective code reuse and maintainable software development. You may not redefine PHP's built-in functions.

### Type declarations

From [secure.php.net/manual/en/functions.arguments.php](https://secure.php.net/manual/en/functions.arguments.php):

Type declarations allow functions to require that parameters are of a certain type at call time. If the given value is of the incorrect type, then an error is generated. To specify a type declaration, the type name should be added before the parameter name. The declaration can be made to accept NULL values if the default value of the parameter is set to NULL.

Type declarations are not compulsory, but they help to clarify the purpose of your functions and to avoid mistakes.

```
<?php
function factorial(int $n): void {
    $res = 1;
    for ($i = $n; $i > 1; $i--) $res *= $i;
    echo "$n! -> $res<br>";
}

try {
    factorial(5);
    factorial(-5);
    factorial(50);
    factorial('asd');
} catch (Error $e) {
    echo 'Caught error: ', $e->getMessage();
}

?>
```



1. A function can return a value using the **return** keyword.
2. The **return** statement terminates the function meaning any following instructions will never get executed.

```
<?php
function factorial(int $n): float {
    $res = 1;
    for ($i = $n; $i > 1; $i--) $res *= $i;
    return $res;
}

try {
    echo "5! -> " . factorial(5) . "<br>";
    echo "-5! -> " . factorial(-5) . "<br>";
    echo "50! -> " . factorial(50);
} catch (Error $e) {
    echo 'Caught error: ', $e->getMessage();
}

?>
```

A function can of course also return HTML. Let's create a function that returns an HTML table filled with the data from a one- or two-dimensional array:

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Function example</title>
    <meta charset=UTF-8>
    <style>
      table {
        border: 1px solid black;
        border-collapse: collapse;

        td {
          border: 1px solid black;
          border-collapse: collapse;
          padding: 5px;
        }
      }
    </style>
  </head>
  <body>
    <?php
      // Can handle 1- and 2-dimensional arrays.
      function get_table(array $data): string {
        $output = '<table>';
        foreach ($data as $row) {
          $output .= "<tr>";
          if (is_array($row))
            foreach ($row as $cell) $output .= "<td>$cell</td>";
          else $output .= "<td>$row</td>";
          $output .= "</tr>";
        }
        $output .= '</table>';
        return $output;
      }

      try {
        echo 'Table 1<br>';
        echo get_table([[1, 2, 3], [1, 4, 'a']]);
        echo 'Table 2<br>';
        echo get_table([1, 2, 3, [1, 4, 'a']]);
      } catch (Error $e) {
        echo 'Caught error: ', $e->getMessage(), "\n";
      }
    ?>
  </body>
</html>

```

## Variable scope

Cf. [secure.php.net/manual/en/language.variables.scope.php](https://secure.php.net/manual/en/language.variables.scope.php).

## global

Variables declared outside of functions and classes are global, i.e. they are available everywhere in the script except within functions and classes. Inside functions we can use the `global` keyword to access global variables.

Examples:

[students.btsi.lu/evegi144/WAD/PHP/variablescope1.php](https://students.btsi.lu/evegi144/WAD/PHP/variablescope1.php)

```
<?php
$name = "Asterix";

function test(): void {
    echo $name;
}

echo $name . "<br>";
test(); // This produces an error, as $name is global and not visible inside
function test.
?>
```

[students.btsi.lu/evegi144/WAD/PHP/variablescope2.php](https://students.btsi.lu/evegi144/WAD/PHP/variablescope2.php)

```
<?php
$name = "Asterix";

function test(): void {
    global $name;
    $name .= " and Obelix<br>";
    echo $name;
}

echo $name . "<br>";
test(); // This is OK.
echo $name . "<br>";
?>
```

## static

A static variable exists only in a local function scope, but it does not lose its value when program execution leaves this scope.

Example:

[students.btsi.lu/evegi144/WAD/PHP/variablescope3.php](https://students.btsi.lu/evegi144/WAD/PHP/variablescope3.php)

```
<?php
function test(): void {
    static $x = 0;
    echo $x++ . '<br>';
}

test();
test();
test();
?>
```

## Default values for function parameters

We can specify default values for parameters in case the caller does not pass them:

[students.btsi.lu/evegi144/WAD/PHP/functions4.php](https://students.btsi.lu/evegi144/WAD/PHP/functions4.php)

```
<?php
function createParagraph(string $text, bool $strong = false): string {
    $s1 = $strong ? "<strong>" : "";
    $s2 = $strong ? "</strong>" : "";
    return "<p>$s1$text$s2</p>";
}

try {
    echo createParagraph("This is normal text.");
    echo createParagraph("This is important text.", true);
} catch(Error $e) {
    echo 'Caught error: ', $e->getMessage(), "\n";
}
?>
```



When using default arguments, any defaults should be on the right side of any non-default arguments.

## Passing values by reference

If we want a function to be able to change the value of a parameter, so that the changed value is available to the caller, we pass the parameter by reference by prepending a **&** to the parameter name in the function prototype:

[students.btsi.lu/evegi144/WAD/PHP/functions5.php](https://students.btsi.lu/evegi144/WAD/PHP/functions5.php)

```

<?php
function doubleArray(array &$arr) {
    for ($i = 0; $i < count($arr); $i++) $arr[$i] *= 2;
}

try {
    $arr = array(1, 2, 3);
    echo '<pre>' . print_r($arr, true) . '</pre>';
    doubleArray($arr);
    echo '<pre>' . print_r($arr, true) . '</pre>';
} catch(Error $e) {
    echo 'Caught error: ', $e->getMessage(), "\n";
}
?>

```

## Using a variable number of parameters

Since PHP 5.6 (cf. [secure.php.net/manual/en/functions.arguments.functions.variable-arg-list](https://secure.php.net/manual/en/functions.arguments.functions.variable-arg-list)):

argument lists may include the `...` token to denote that the function accepts a variable number of arguments. The arguments will be passed into the given variable as an array

We can also write functions that accept a variable number of parameters using `func_num_args`, `func_get_args` or `func_get_arg`:

[students.btsi.lu/evegi144/WAD/PHP/functions6.php](https://students.btsi.lu/evegi144/WAD/PHP/functions6.php)

```

<?php
function test1(...$args) {
    echo "Number of arguments: " . $num = count($args) . "<br>";
    for ($i = 0; $i < $num; $i++) echo "Argument $i: " . $args[$i] . "<br>";
    echo '<pre>' . print_r($args, true) . '</pre>';
}

function test2() {
    echo "Number of arguments: " . $num = func_num_args() . "<br>";
    for ($i = 0; $i < $num; $i++) echo "Argument $i: " . func_get_arg($i) . "<br>";
    echo '<pre>' . print_r(func_get_args(), true) . '</pre>';
}

test1();
test1(1, "abc", 5.67);
test2();
test2(1, "abc", 5.67);
?>

```

## Anonymous functions

[Anonymous functions](#), also known as closures, can be used for callbacks and other purposes.

[students.btsi.lu/evegi144/WAD/PHP/closure1.php](https://students.btsi.lu/evegi144/WAD/PHP/closure1.php)

```
<?php
$numbers = [7, 5, 1, 3, 4, 8, 6, 2];
usort($numbers, function ($x, $y) {
    if ($x < $y) return -1;
    else return 1;
});
echo '<pre>' . print_r($numbers, true) . '</pre>';
?>
```

## Recursive functions

The factorial problem can be solved very elegantly using a recursive function call:

[students.btsi.lu/evegi144/WAD/PHP/functions3.php](https://students.btsi.lu/evegi144/WAD/PHP/functions3.php)

```
<?php
function factorial(int $n): float {
    if ($n <= 0) return 1;
    return $n * factorial($n - 1);
}

try {
    echo "5! -> " . factorial(5) . "<br>";
    echo "-5! -> " . factorial(-5) . "<br>";
    echo "50! -> " . factorial(50) . "<br>";
} catch(Error $e) {
    echo 'Caught error: ', $e->getMessage(), "\n";
}
?>
```

### 5.2.13. Including external scripts

For major projects PHP files can become very large and unwieldy. In these cases, splitting large files into separate entities and including them in the main script can render our project much more manageable. Let's look at an example:

[students.btsi.lu/evegi144/WAD/PHP/include1.php](https://students.btsi.lu/evegi144/WAD/PHP/include1.php)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Include demo 1</title>
  </head>
  <body>
    <?php
      include 'includeheader.html';
      include 'includemain.html';
      include 'includefooter.html';
    ?>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/PHP/includeheader.html](https://students.btsi.lu/evegi144/WAD/PHP/includeheader.html)

```
<header>
  <h1>Include demo 1</h1>
</header>
```

[students.btsi.lu/evegi144/WAD/PHP/includemain.html](https://students.btsi.lu/evegi144/WAD/PHP/includemain.html)

[students.btsi.lu/evegi144/WAD/PHP/includefooter.html](https://students.btsi.lu/evegi144/WAD/PHP/includefooter.html)

```
<footer>
  &copy; 2017 LAM T2IF1
</footer>
```

PHP provides four keywords to include external scripts:

<code>include</code>	<code>include_once</code>	<code>require</code>	<code>require_once</code>
----------------------	---------------------------	----------------------	---------------------------

[students.btsi.lu/evegi144/WAD/PHP/include2.php](https://students.btsi.lu/evegi144/WAD/PHP/include2.php)

```
<?php
    echo "Let's include...<br>";
    include 'includeme.php';
    include 'includeme.php';
    echo "Let's include_once...<br>";
    include_once 'includeme.php';
    echo "Let's require...<br>";
    require 'includeme.php';
    require 'includeme.php';
    echo "Let's require_once...<br>";
    require_once 'includeme.php';
?>
```

[students.btsi.lu/evegi144/WAD/PHP/includeme.php](http://students.btsi.lu/evegi144/WAD/PHP/includeme.php)

```
<?php
    echo "I've been included!<br>";
?>
```

What is the difference between **include** and **require**? Let's take a look:

[students.btsi.lu/evegi144/WAD/PHP/include3.php](http://students.btsi.lu/evegi144/WAD/PHP/include3.php)

```
<?php
    echo 'Check 1<br>';
    include_once 'include3mega.php';
    echo 'Check 2<br>';
    require_once 'include3mega.php';
    echo 'Check 3';
?>
```

## 5.2.14. Superglobals

There are nine [superglobal arrays](#) that we can use anywhere in our scripts.

**`$_GET`**

This array contains all variables sent via a HTTP GET request, either directly through the URL or via an HTML form (cf. [GET](#)). Look at the invoking URL in the following example!

This can be useful, for instance, to create a simple page navigation:

[students.btsi.lu/evegi144/WAD/PHP/get2.php](http://students.btsi.lu/evegi144/WAD/PHP/get2.php)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>GET hyperlink navigation example</title>
  </head>
  <body>
    <h1>GET hyperlink navigation example</h1>
    <?php
      echo '<pre>' . print_r($_GET, true) . '</pre>';
      if (isset($_GET['login'])) echo 'Please log in <a href=get2.php>Home</a>';
      elseif (isset($_GET['register'])) echo 'Please register <a
href=get2.php>Home</a>';
      else echo '<a href=?login>Login</a> <a href=?register>Register</a>';
    ?>
  </body>
</html>

```

## \$\_POST

This array contains all variables sent via a HTTP POST request, either through an HTML form or directly through an HTTP request (cf. [POST](#)).

## \$\_FILES

Contains all variables sent via a HTTP POST file upload.



Allowing anyone to upload files to a server represents a major security risk, as the uploaded file can be executed by everyone who has access to the directory. Only allow file upload for authenticated users and only for those file types that you want.

[students.btsi.lu/evegi144/WAD/PHP/upload.php](https://students.btsi.lu/evegi144/WAD/PHP/upload.php)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>File Upload</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <form action=upload1.php method=post enctype=multipart/form-data>
      <label for=file>Filename:</label>
      <input type=file name=file id=file><br>
      <input type=submit value=Upload>
    </form>
  </body>
</html>

```

```
<?php
//exit; For security reasons, script may not run further.
echo '<pre>' . print_r($_FILES, true) . '</pre>';
if (isset($_FILES["file"]) && $_FILES["file"]["size"] > 0) {
    if ($_FILES["file"]["error"] > 0) {
        echo "Error: " . $_FILES["file"]["error"] . "<br>";
    }
    else {
        move_uploaded_file($_FILES["file"]["tmp_name"], $_FILES["file"]["name"]);
        echo <<<EOT
            <p>
                Type:  {$_FILES["file"]["type"]}<br>
                Size:  {$_FILES["file"]["size"]} / 1024 kB<br>
                Temporarily stored in: {$_FILES["file"]["tmp_name"]}<br>
                Moved to: {$_FILES["file"]["name"]}
            </p>
            <button onclick=location.href='upload.php'>Upload another file</button>
EOT;
    }
}
else header('Location: https://' . $_SERVER['HTTP_HOST'] .
dirname($_SERVER['PHP_SELF']) .
'/upload.php');
?>
```

For an excellent more in-depth example, study [w3schools.com/php/php\\_file\\_upload.asp](https://www.w3schools.com/php/php_file_upload.asp).

### **`$_COOKIE`**

Contains all variables sent via HTTP cookies. See [Cookies and sessions](#).

### **`$_REQUEST`**

Contains all variables sent via HTTP GET, HTTP POST and HTTP cookies unless there are variables with the same name, in which case some are overwritten (cf. [php.net/manual/en/ini.core.php#ini.request-order](https://www.php.net/manual/en/ini.core.php#ini.request-order)). It is preferable to use `$_GET`, `$_POST` or `$_COOKIE` directly instead of `$_REQUEST`.

### **`$_SESSION`**

Contains all variables stored in a user's session (server-side data store). See [Cookies and sessions](#).

### **`$_SERVER`**

Contains all variables set by the web server or other sources that relate directly to the execution of the script.

```
<?php
    echo '<pre>' . print_r($_SERVER, true) . '</pre>';
?>
```

### **`$_ENV`**

Contains all variables passed to the current script via the environment, e.g. when the script has been called from a shell instead of via the web server. [students.btsi.lu/evegi144/WAD/PHP/ENV.php](https://students.btsi.lu/evegi144/WAD/PHP/ENV.php)

```
<?php
    echo '<pre>' . print_r($_ENV, true) . '</pre>';
?>
```

### **`$GLOBALS`**

[students.btsi.lu/evegi144/WAD/PHP/GLOBALS.php](https://students.btsi.lu/evegi144/WAD/PHP/GLOBALS.php)

```
<?php
    $x = 27;
    echo '<pre>' . print_r($GLOBALS, true) . '</pre>';
?>
```

## **5.2.15. Forms**

HTML forms are the main mechanism used to transfer data from the client to the server. The [HTTP](#) protocol provides two methods to send form data to the server: POST and GET.

### **POST**

The POST method transmits form data within the HTTP request. It is the recommended method to transmit data to the server:

[students.btsi.lu/evegi144/WAD/PHP/forms1.html](https://students.btsi.lu/evegi144/WAD/PHP/forms1.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Forms1</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <form method=post action=forms1.php>
      <label for=name>Please enter your name:</label>
      <input id=name name=x1>
      <input type=submit value=Submit>
    </form>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/PHP/forms1.php](https://students.btsi.lu/evegi144/WAD/PHP/forms1.php)

```
<?php
  if (isset($_POST['x1']) && strlen($_POST['x1'])) echo "Hello {$_POST['x1']}";
  else echo "Alarm! Unknown intruder!";
?>
```

## GET

The GET method transmits data within the URL, i.e. visible to the world! Take a look at the URL after submitting the form:

[students.btsi.lu/evegi144/WAD/PHP/forms2.html](https://students.btsi.lu/evegi144/WAD/PHP/forms2.html)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Forms2</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <a href=forms2.php?name=Otto>Click</a>
    <form action=forms2.php>
      <label for=name>Please enter your name:</label>
      <input id=name name=name>
      <input type=submit value=Submit>
    </form>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/PHP/forms2.php](https://students.btsi.lu/evegi144/WAD/PHP/forms2.php)

```
<?php
    if (isset($_GET['name']) && strlen($_GET['name'])) echo "Hello {$_GET['name']}";
    else echo "Alarm! Unknown intruder!";
?>
```

This is about as insecure as it gets. It should only be used to request information from a server, for instance from Yahoo: [students.btsi.lu/evegi144/WAD/PHP/get1.php](http://students.btsi.lu/evegi144/WAD/PHP/get1.php)

```
<?php
    header('Location: http://finance.yahoo.com/q?s=EURUSD=X');
?>
```

## Form validation

Study [w3schools.com/php/php\\_form\\_validation.asp](http://w3schools.com/php/php_form_validation.asp). We should not simply accept any input the user submits using a form. In order to avoid an attack on our application and to prevent invalid data from entering our database, we need to perform an in-depth check. To check whether a value has been entered in a text box, use `strlen` (cf. `strlen`) instead of `empty`. If the user enters 0, `empty` will return true:

[students.btsi.lu/evegi144/WAD/PHP/formvalidation1.php](http://students.btsi.lu/evegi144/WAD/PHP/formvalidation1.php)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Form Validation</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <form method=post>
      Number of dogs:
      <input type=text name=dogs>
      <input type=submit value=Submit>
    </form>
    <?php
      if (isset($_POST['dogs'])) {
        if (empty($_POST['dogs'])) echo "Dog number empty!<br>";
        if (!strlen($_POST['dogs'])) echo "Missing dog number!";
      }
    ?>
  </body>
</html>
```

## Allow only alphanumeric input

In order to allow only alphanumeric characters to be submitted in a form element we can use `ctype_alnum`:

```
if (!ctype_alnum($username)) die('Invalid characters in Username');
```

### Number validation

If we want to accept decimal, hexadecimal, binary, octal and exponential notation numbers, `is_numeric` is the way to go. If we only want integers larger than or equal to 0, we use `ctype_digit`. To accept negative and positive integers, we can convert the string to an integer using `intval` and then back to a string and check whether we get the original string. The same procedure works for decimals, where we convert the string to a float using `floatval`.

[students.btsi.lu/evegi144/WAD/PHP/numbervalidation1.php](https://students.btsi.lu/evegi144/WAD/PHP/numbervalidation1.php)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Number Validation</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <form method=post>
      <label>Number of dogs:</label>
      <input name=dogs required>
      <input type=submit value=Submit>
    </form>
    <?php
      if (isset($_POST['dogs'])) {
        $x = $_POST['dogs'];
        if (!is_numeric($x)) echo "You did not enter a valid numeric string.<br>";
        if (!ctype_digit($x))
          echo "You did not enter an integer larger than or equal to zero.<br>";
        if (strval(intval($x)) !== $x) echo "You did not enter an integer.<br>";
        if (strval(floatval($x)) !== $x) echo "You did not enter a float.<br>";
      }
    ?>
  </body>
</html>
```

### Email validation

We can use a simplistic regular expression (cf. "Head First PHP & MySQL" first edition p. 600):

[students.btsi.lu/evegi144/WAD/PHP/emailvalidation1.php](https://students.btsi.lu/evegi144/WAD/PHP/emailvalidation1.php)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Email Validation</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <form method=post>
      <label>E-mail:</label>
      <input type=email name=email required>
      <input type=submit value=Submit>
    </form>
    <?php
      if (isset($_POST['email'])) {
        if (preg_match('/^[a-zA-Z0-9][a-zA-Z0-9\._\-\&!?=#]*@/ ', $_POST['email']))
          echo "Valid email";
        else echo "Invalid email";
      }
    ?>
  </body>
</html>

```

The safest way of handling email validation is probably to use Carl Henderson's `is_valid_email_address` function followed by a call to `checkdnsrr`.

### Drop-down lists

[students.btsi.lu/evegi144/WAD/PHP/formvalidation2.php](https://students.btsi.lu/evegi144/WAD/PHP/formvalidation2.php)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Form Validation</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <form method=post>
      <select name=selection>
        <option value=1>Option 1</option>
        <option value=2>Option 2</option>
        <option value=3>Option 3</option>
        <option value=4>Option 4</option>
      </select>
      <button>Submit</button>
    </form>
    <?php
      if (isset($_POST['selection']))
        echo 'Option number ' . $_POST['selection'] . ' selected';
    ?>
  </body>
</html>
```

If we allow multiple selections, we need to add `[]` to the name of the selection so that the values are available as an array in PHP.

[students.btsi.lu/evegi144/WAD/PHP/formvalidation3.php](https://students.btsi.lu/evegi144/WAD/PHP/formvalidation3.php)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Form Validation</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <form method=post>
      <select multiple name=selection[]>
        <option value=1>Option 1</option>
        <option value=2>Option 2</option>
        <option value=3>Option 3</option>
        <option value=4>Option 4</option>
      </select>
      <button>Submit</button>
    </form>
    <?php
      if (isset($_POST['selection']))
        echo '<pre>' . print_r($_POST['selection'], true) . '</pre>';
    ?>
  </body>
</html>
```

### Multiple choice forms

[students.btsi.lu/evegi144/WAD/PHP/formvalidation4.php](https://students.btsi.lu/evegi144/WAD/PHP/formvalidation4.php)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Form Validation</title>
    <meta charset=UTF-8>
    <style>
      form {
        width:      150px;
        margin-left: auto;
        margin-right: auto;
      }

      input {
        margin: 3px;
      }
    </style>
  </head>
  <body>
    <form method=post>
      <input name=name placeholder=Name required>
      <input type=email name=email placeholder=E-mail required><br>
      <input type=checkbox name=colours[] value=white checked>White<br>
      <input type=checkbox name=colours[] value=black>Black<br>
      <input type=checkbox name=colours[] value=red>Red
      <input type=submit name=submit>
    </form>
    <?php
      if (isset($_POST['name'])) echo $_POST['name'];
      if (isset($_POST['colours'])) {
        echo "<br><hr>";
        echo '<pre>' . print_r($_POST['colours'], true) . '</pre>';
      }
    ?>
  </body>
</html>

```

### Preventing form hijacking

What runs on the client side is out of our control. Thus we cannot prevent a malicious user from trying to hijack our application in order to inflict damage. What we can do is to make our scripts as safe as possible in order to deal with attacks. How can our forms be attacked? Let's take the previous example of the multiple choice form. The way we have implemented it so far, an attacker can simply write and run a script such as the following:

[students.btsi.lu/evegi144/WAD/PHP/formvalidation5.php](http://students.btsi.lu/evegi144/WAD/PHP/formvalidation5.php)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Form Validation</title>
    <meta charset=UTF-8>
    <style>
      form {
        width:      150px;
        margin-left: auto;
        margin-right: auto;
      }

      input {
        margin: 3px;
      }
    </style>
  </head>
  <body>
    <?php $terror = '<script>alert("Your app has been infiltrated. '
      . 'Wire $10m to my account, or else...")</script>' ?>
    <form method=post action=formvalidation4.php>
      <input name=name placeholder=Name required>
      <input type=email name=email placeholder=E-mail required><br>
      <input type=checkbox name=colours[] value='<?php echo $terror; ?>'
        checked>White<br>
      <input type=checkbox name=colours[] value='<?php echo $terror; ?>'>Black<br>
      <input type=checkbox name=colours[] value='<?php echo $terror; ?>'>Red
      <input type=submit name=submit>
    </form>
  </body>
</html>

```

This is a harmless example, but a real attacker would obviously submit code in the value field that would try to hijack our server and/or database or perform some other mischief.

### Using arrays to submit large volumes of data to the server

If we want to send large quantities of data to the server via a form, arrays can be very useful:

```

Array
(
    [ids] => Array
        (
            [0] => 1
            [1] => 2
            [2] => 6
        )

    [brands] => Array
        (
            [0] => LTAM
            [1] => LTAMmm
            [2] => LTAMa
        )

    [models] => Array
        (
            [0] => Mega12
            [1] => Mega2
            [2] => Mega3
        )

    [prices] => Array
        (
            [0] => 123.45
            [1] => 234.56
            [2] => 345.67
        )

    [submit] =>
)

```

Id	Brand	Model	Price
<input type="text" value="1"/>	<input type="text" value="LTAM"/>	<input type="text" value="Mega12"/>	<input type="text" value="123.45"/>
<input type="text" value="2"/>	<input type="text" value="LTAMmm"/>	<input type="text" value="Mega2"/>	<input type="text" value="234.56"/>
<input type="text" value="6"/>	<input type="text" value="LTAMa"/>	<input type="text" value="Mega3"/>	<input type="text" value="345.67"/>

[students.btsi.lu/evegi144/WAD/PHP/forms\\_arrays1.php](https://students.btsi.lu/evegi144/WAD/PHP/forms_arrays1.php)

```

<!DOCTYPE html>
<html lang=en>
<head>
    <title>Submit tabular data to the server with arrays</title>
    <meta charset=utf-8>
</head>
<body>
    <main>
        <?php
            echo '<pre>' . print_r($_POST, true) . '</pre>';
            if (isset($_POST['submit'])) {
                $ids = $_POST['ids'];
                $brands = $_POST['brands'];
                $models = $_POST['models'];
                $prices = $_POST['prices'];
            }
        </?php>
    </main>
</body>

```

```

else {
    $ids = [1, 2, 3];
    $brands = ['LAM', 'LAM', 'LAM'];
    $models = ['Mega1', 'Mega2', 'Mega3'];
    $prices = [123.45, 234.56, 345.67];
}

?>
<form method=post>
  <table>
    <tr>
      <th>Id</th>
      <th>Brand</th>
      <th>Model</th>
      <th>Price</th>
    </tr>
    <tr>
      <td><input name=ids[] value=<?php echo $ids[0]; ?>></td>
      <td><input name=brands[] value=<?php echo $brands[0]; ?>></td>
      <td><input name=models[] value=<?php echo $models[0]; ?>></td>
      <td><input name=prices[] value=<?php echo $prices[0]; ?>></td>
    </tr>
    <tr>
      <td><input name=ids[] value=<?php echo $ids[1]; ?>></td>
      <td><input name=brands[] value=<?php echo $brands[1]; ?>></td>
      <td><input name=models[] value=<?php echo $models[1]; ?>></td>
      <td><input name=prices[] value=<?php echo $prices[1]; ?>></td>
    </tr>
    <tr>
      <td><input name=ids[] value=<?php echo $ids[2]; ?>></td>
      <td><input name=brands[] value=<?php echo $brands[2]; ?>></td>
      <td><input name=models[] value=<?php echo $models[2]; ?>></td>
      <td><input name=prices[] value=<?php echo $prices[2]; ?>></td>
    </tr>
  </table>
  <button name=submit>Submit</button>
</form>
</main>
</body>
</html>

```

## Using form fields to transfer data between pages

The safer way to transfer data between pages is to use [sessions](#), as session data is stored on the server and therefore cannot be manipulated by the user. An easy, though not recommended, way to transfer data between pages is using input fields. In addition to simple data structures such as numbers or text, we can also transfer multidimensional arrays.

A simple application example is the creation of a list:

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Simple list</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <header>
      <h1>Simple list</h1>
    </header>
    <main>
      <?php
        // To understand what happens, open your browser console, switch to the
network tab
        // and look at the response to each POST request.
        $list = []; // Start with an empty list.

        if (isset($_POST['submit'])) { // Has the form been submitted?
          if (isset($_POST['list'])) $list = $_POST['list']; // Store previous list
items.
          $list[] = $_POST['item']; // Add the new item.
        }
      ?>
      <form method=post>
        <?php
          $count = 0; // We need this as index for the list array in the input.
          foreach ($list as $item) { // Loop through the item list.
            // Add item as input so that it gets resubmitted with the form and not
lost.
            echo "<input name=list[$count] value=$item><br>";
            $count++;
          }
        ?>
        Number of items: <?php echo $count; ?>
        <br><br>
        <label for=item>Item:</label>
        <input id=item name=item required>
        <input type=submit name=submit value=Add>
      </form>
    </main>
  </body>
</html>
```

Let's say we want a page that allows us to create the name list for our sports club. Using input fields, we could do this as follows:

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Tennis club member list</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <header>
      <h1>Tennis club member list</h1>
    </header>
    <main>
      <?php
        // To understand what happens, open your browser console, switch to the
network tab
        // and look at the response to each POST request.
        $names = [];

        if (isset($_POST['submit'])) {
          if (isset($_POST['existing_names'])) $names = $_POST['existing_names'];
          $names[] = [$_POST['first_name'], $_POST['last_name']];
        }
      ?>
      <form method=post>
        <?php
          $count = 0;
          foreach ($names as $name) {
            echo "<input name=existing_names[$count][0] value=$name[0]>";
            echo "<input name=existing_names[$count][1] value=$name[1]><br>";
            $count++;
          }
        ?>
        Number of members: <?php echo $count; ?>
        <br>
        First name: <input name=first_name required>
        Last name: <input name=last_name required>
        <input type=submit name=submit value=Add>
      </form>
    </main>
  </body>
</html>

```

If we want to transfer information between pages without the user seeing it (unless he/she uses the console or some other tool), we can hide the input fields.

## 5.2.16. Useful PHP functions

### isset

Determine if a variable is set and not **NULL**. A good list of examples can be found by clicking on the section title.

## strlen

Get the length of a string:

[students.btsi.lu/evegi144/WAD/PHP/strlen1.php](https://students.btsi.lu/evegi144/WAD/PHP/strlen1.php)

```
<?php
    $test_string = "WSERS rocks";
    echo "Length of string '$test_string': " . strlen($test_string);
?>
```

## exit or die

These two perform exactly the same job, i.e. output a message and terminate the current script. Both are not really functions but language constructs.

## header

Sends a raw HTTP header, often used to redirect the browser to another page. The function must be called before any actual output is sent, either by normal HTML tags, blank lines in a file, or from PHP. Alternatively, you can use `ob_start` and `ob_flush` to activate and use output buffering. In this case it does not matter where in your script you use `header`.



The client is not obliged to respect the location header. Therefore we need to make sure our script stops after the location header has been sent (cf. [thedailywtf.com/articles/WellIntentioned-Destruction](https://thedailywtf.com/articles/WellIntentioned-Destruction)).

[students.btsi.lu/evegi144/WAD/PHP/header1.php](https://students.btsi.lu/evegi144/WAD/PHP/header1.php)

```
<?php
    header('Location: header2.php');
    exit;
?>
```

[students.btsi.lu/evegi144/WAD/PHP/header2.php](https://students.btsi.lu/evegi144/WAD/PHP/header2.php)

```
<?php
    echo "This is header2.php.";
?>
```

## mt\_rand

Generates a random integer. Minimum and maximum can be specified.

## eval

Evaluates a given string as PHP code.

Example:

[students.btsi.lu/evegi144/WAD/PHP/eval1.php](https://students.btsi.lu/evegi144/WAD/PHP/eval1.php)

```
<?php
    $x = 7;
    $y = 19;
    $op = '+';
    $str = $x . $op . $y;
    echo eval('return ' . $str . ';');
?>
```

### dirname

Given a string containing the path of a file or directory, this function will return the parent directory's path.

[students.btsi.lu/evegi144/WAD/PHP/dirname1.php](https://students.btsi.lu/evegi144/WAD/PHP/dirname1.php)

```
<?php
    echo dirname($_SERVER['PHP_SELF']);
?>
```

### number\_format

Excellent description and examples at the usual place.

### strpos

### implode

### explode

### filter\_var

This function filters a variable with a specified filter.

[students.btsi.lu/evegi144/WAD/PHP/filtervar1.php](https://students.btsi.lu/evegi144/WAD/PHP/filtervar1.php)

```
<?php
    $email = 'gilles.everling@education.lu';
    var_dump(filter_var($email, FILTER_VALIDATE_EMAIL));
    $res = checkdnsrr(preg_replace('/^[a-zA-Z0-9][a-zA-Z0-9\.\_\&!?=#]*@/ ', '',
    $email));
    if ($res) echo "<br>OK";
    else echo "<br>Not OK";
?>
```

`preg_match`

`sprintf`

`printf`

Study this example: [w3schools.com/php/showphp.asp?filename=demo\\_func\\_string\\_printf4](http://w3schools.com/php/showphp.asp?filename=demo_func_string_printf4)

`list`

`ini_set`

`rawurlencode`

Encode given string according to [www.faqs.org/rfcs/rfc3986.html](http://www.faqs.org/rfcs/rfc3986.html).

`htmlspecialchars`

Convert special characters to HTML entities, cf. [secure.php.net/manual/en/faq.html.php](http://secure.php.net/manual/en/faq.html.php).

`htmlentities`

`html_entity_decode`

`serialize` **and** `unserialize`

In order to store arrays or objects in a database without loss of type and structure we can use the `serialize` function. We can use `unserialize` to restore the saved structure. In the database it is important to use a binary column type such as a `BLOB` in MySQL to store the object.

`getdate`

`getdate` returns an associative array containing the date information of the timestamp, or the current local time if no timestamp is given.

`date`

`date(format)` gives us a string with the current date formatted according to the given format string. We should call `date_default_timezone_set` to set the correct timezone before calling `date`. The list of supported timezones can be found at [php.net/manual/en/timezones.php](http://php.net/manual/en/timezones.php).

Example: [students.btsi.lu/evegi144/WAD/PHP/date1.php](http://students.btsi.lu/evegi144/WAD/PHP/date1.php)

```
<?php
    date_default_timezone_set('Europe/Luxembourg');
    echo 'Current date: ' . date('d-m-Y H:i:s');
?>
```

`nl2br`

`nl2br` inserts HTML line breaks before all newlines in a string. This function is particularly useful for chat boxes, see [Message board](#) for a usage example.

## Executing system commands

PHP offers several options to execute system commands:

1. `exec`
2. `system`
3. `shell_exec`
4. Backtick operator, cf. [php.net/manual/en/language.operators.execution.php](http://php.net/manual/en/language.operators.execution.php)
5. `passthru`
6. `proc_open`

For user input you should apply `escapeshellarg` before using any of the above functions.

## Password hashing

From [php.net/manual/en/faq.passwords.php](http://php.net/manual/en/faq.passwords.php):

Hashing algorithms such as MD5, SHA1 and SHA256 are designed to be very fast and efficient. With modern techniques and computer equipment, it has become trivial to "brute force" the output of these algorithms, in order to determine the original input.

Therefore, it is important to use the `password_hash` and `password_verify` functions, like so:

[students.btsi.lu/evegi144/WAD/PHP/password\\_hash1.php](http://students.btsi.lu/evegi144/WAD/PHP/password_hash1.php)

```
<?php
    $pw = '1!@23sa#GdTERWsd';
    $pw_hash = password_hash($pw, PASSWORD_DEFAULT);
    echo $pw_hash . '<br>';
    if (password_verify($pw, $pw_hash)) echo 'Password verified';
    else echo 'Password verification failed.';
?>
```

### 5.2.17. Regular expressions

Regular expressions can be used to replace text in, test for a pattern in or extract a substring from a string. They are very flexible and powerful and can be used in many programming languages, including JavaScript and PHP.

[regexper.com](http://regexper.com)

[www.regextester.com](http://www.regextester.com)

In PHP, they are implemented using the [PCRE extension](http://php.net/manual/en/book.pcre.php).

See [www.noupe.com/development/php-regular-expressions.html](http://www.noupe.com/development/php-regular-expressions.html) and [www.rexegg.com/regex-](http://www.rexegg.com/regex-)

[cookbook.html](#) for tutorials.

Remember that the pattern must be enclosed by delimiters. A delimiter can be any non-alphanumeric, non-backslash, non-whitespace character.

### 5.2.18. Cookies and sessions

HTTP is a stateless protocol, thus any data is forgotten when the page has been sent to the client and the connection is closed. To store client-specific data, we can use cookies, sessions, databases, files, etc. Here we'll take a closer look at cookies and sessions.

#### Cookies

Cookies are bits of information that are stored locally in files by the client browser (cf. [php.net/manual/en/function.setcookie.php](#) and [w3schools.com/php/func\\_http\\_setcookie.asp](#)).

From php.net:

`setcookie()` defines a cookie to be sent along with the rest of the HTTP headers. Like other headers, **cookies must be sent before any output from your script** (this is a protocol restriction). This requires that you place calls to this function prior to any output, including `<html>` and `<head>` tags as well as any whitespace.

The next time the user points his browser to the same page, assuming that the cookie's lifetime has not expired and the user has not deleted the cookie, which is easy to do using the browser options, the cookie will be automatically sent to the server. Thus, cookies are a mechanism to store data persistently on the client side, i.e. the information does not disappear on page reload or when the client browser is shut down.

As a simple example, let's store a user name in the client's browser for the duration of 3 days (3 days \* 24 hours \* 60 minutes \* 60 seconds → 259200 seconds) that will only be sent for files that are located in the `cookies` folder or subfolders thereof:

[students.btsi.lu/evegi144/WAD/PHP/cookies1.php](#)

```
<?php
    // Store the cookie named 'user' with value 'Asterix' for 3 days in the client's
    browser.
    setcookie('user10', 'Asterix', time() + 259200,dirname($_SERVER['PHP_SELF']) .
    '/cookies', '',
        true, true);
    header('Location: cookies/cookies2.php');
?>
```

[students.btsi.lu/evegi144/WAD/PHP/cookies/cookies2.php](#)

```
<?php
    print_r($_COOKIE);
?>
```

Now when we visit this page again, we can see that the cookie is indeed sent to the server:

Network	Storage	Web Audio
Size		Headers
0 KB	– 21 ms	Filter cookies
		Request cookies
		user: "Asterix"

To delete a cookie before its expiration time, we can set its value to empty and its expiration time in the past:

[students.btsi.lu/evegi144/WAD/PHP/cookies3.php](https://students.btsi.lu/evegi144/WAD/PHP/cookies3.php)

```
<?php
    print $_COOKIE['user10'];
    setcookie('user10', '', 1, dirname($_SERVER['PHP_SELF']) . '/cookies', '',
        true, true);
    header('Location: cookies/cookies2.php');
?>
```

Changes made to cookies are only visible the next time our script gets executed. Therefore, this won't work as expected:

[students.btsi.lu/evegi144/WAD/PHP/cookies4.php](https://students.btsi.lu/evegi144/WAD/PHP/cookies4.php)

```
<?php
    setcookie('my_cookie', 'HYPER_COOKIE', time() + 10);
    print $_COOKIE['my_cookie'];
?>
```

If we want to use them to store confidential data, we must use strong encryption. If we want to store the user's password in encrypted form, so that nobody, including ourselves, can get the unencrypted password, we can proceed as follows:

[students.btsi.lu/evegi144/WAD/PHP/cookies5.php](https://students.btsi.lu/evegi144/WAD/PHP/cookies5.php)

```
<?php
    setcookie('password', password_hash('Asterix', PASSWORD_DEFAULT), time() + 10);
    header('Location: cookies2.php');
?>
```

If we want to store encrypted information that we can decrypt, we can use [php.net/manual/en/function.mcrypt-encrypt.php](http://php.net/manual/en/function.mcrypt-encrypt.php).



You should not use spaces or other special characters in file or path names. If you absolutely have to, then use `rawurlencode`.

## Sessions

If we do not want our data to be accessible outside of our server, we can use sessions. Sessions serve to maintain visitor-specific state between page requests. Here the description from [secure.php.net/manual/en/intro.session.php](http://secure.php.net/manual/en/intro.session.php):

A visitor accessing your web site is assigned a unique id, the so-called session id. This is either stored in a cookie on the user side or is propagated in the URL. The session support allows you to store data between requests in the `$_SESSION` superglobal array. When a visitor accesses your site, PHP will check automatically (if `session.auto_start` is set to 1) or on your request (explicitly through `session_start()` or implicitly through `session_register()`) whether a specific session id has been sent with the request. If this is the case, the prior saved environment is recreated.



To use cookie-based sessions, `session_start()` must be called BEFORE sending anything to the browser.

Let's look at a very basic, though insecure, example:

[students.btsi.lu/evegi144/WAD/PHP/session1.php](http://students.btsi.lu/evegi144/WAD/PHP/session1.php)

```
<?php
    session_start();
    $_SESSION['first name'] = 'Donald';
    $_SESSION['last name'] = 'Duck';
    header('Location: session1a.php');
?>
```

[students.btsi.lu/evegi144/WAD/PHP/session1a.php](http://students.btsi.lu/evegi144/WAD/PHP/session1a.php)

```
<?php
    session_start();
    echo '<pre>' . print_r($_SESSION, true) . '</pre>';
    var_dump(session_get_cookie_params());
?>
```

The bouncer (cf. `bouncer.php`) used in the [WMOTU Address Book](#) illustrates how to implement safe session handling. There are a number of runtime configuration settings that we can use to control

session handling (cf. [php.net/manual/en/session.configuration.php](http://php.net/manual/en/session.configuration.php)):

```
<?php
// Protect from session fixation via session adoption.
ini_set('session.use_strict_mode', true);
# Only send session id cookie over SSL.
ini_set('session.cookie_secure', true);
# Session IDs may only be passed via cookies, not appended to URL.
ini_set('session.use_only_cookies', true);
ini_set('session.cookie_httponly', true);
ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
if (!isset($_SERVER['HTTPS'])) {# If SSL is not active, activate it.
    header('Location: https://' . $_SERVER['HTTP_HOST'] . $_SERVER['PHP_SELF']);
    exit;
}
if (!isset($_SESSION)) session_start(); // Start session.
# After 30 seconds we'll generate a new session ID to prevent a session
# fixation attack (cf. PHP cookbook p. 338).
if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
    session_regenerate_id();
    $_SESSION['generated'] = time();
}
if (!isset($_SESSION['user_id'])) {// No user logged in -> go to the login page.
    header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php');
    exit;
}
?>
```

The main page `index.php` (cf. `index.php`) saves the user id in `$_SESSION['user_id']` if the user has provided a valid login. As long as the user is not logged in, the bouncer will always refer back to the login and sign up page. **The bouncer is required in every file that executes any instructions by simple invocation in order to prevent any unauthorized access to our application.** Our database (cf. `database.php`) class is one of the few exceptions, as the invocation of `database.php` by the client does essentially nothing.

To prevent a so-called session fixation attack (cf. [en.wikipedia.org/wiki/Session\\_fixation](http://en.wikipedia.org/wiki/Session_fixation)) where a malicious user uses a session id to impersonate another user, we use the `session_regenerate_id` (cf. [php.net/manual/en/function.session-regenerate-id.php](http://php.net/manual/en/function.session-regenerate-id.php)) function, which will replace the current session id with a new one, but keep the current session information.

A session terminates when the user closes the browser. If we want the session to survive the closing of the browser, we need to change `session.cookie_lifetime` (cf. [secure.php.net/manual/en/session.configuration.php#ini.session.cookie-lifetime](http://secure.php.net/manual/en/session.configuration.php#ini.session.cookie-lifetime)). When the user logs out, the session needs to terminate immediately. We achieve this as follows ([php.net/manual/en/function.session-destroy.php](http://php.net/manual/en/function.session-destroy.php)):

[students.btsi.lu/evegi144/WAD/PHP/session1kill.php](http://students.btsi.lu/evegi144/WAD/PHP/session1kill.php)

```
<?php
    if (!isset($_SESSION)) session_start(); # Start session if not done already.
    $_SESSION = []; # Empty session array.
    # If session cookie exists, kill it.
    if (session_id() != "" || isset($_COOKIE[session_name()])) {
        $params = session_get_cookie_params();
        setcookie(session_name(), '', 1, $params["path"],
            $params["domain"], $params["secure"], $params["httponly"]
        );
    }
    session_destroy(); # Kill session.
    echo '<pre>' . print_r($_SESSION, true) . '</pre>';
?>
```



To kill the session cookie in the browser, we must specify the same path that we used to create the session when we specified `cookie_path`, i.e. usually `rawurlencode(dirname($_SERVER['PHP_SELF']))`. Make sure to verify that the session cookie really gets deleted in the client browser.



Session security is of paramount importance. If not handled correctly, your users and your app will be exposed. Study [www.phparch.com/2018/01/php-sessions-in-depth](http://www.phparch.com/2018/01/php-sessions-in-depth) and [secure.php.net/manual/en/features.session.security.management.php](http://secure.php.net/manual/en/features.session.security.management.php) for excellent descriptions of the issues and solutions.



You can store objects in sessions if you remember to **include the class definition before calling `session_start` in each file** where you want to use an object retrieved from a session. See [ir2.php.net/manual/ro/oop4.serialization.php](http://ir2.php.net/manual/ro/oop4.serialization.php).

## 5.2.19. Files

### Read file contents as a string

`file_get_contents` (cf. [php.net/manual/en/function.file-get-contents.php](http://php.net/manual/en/function.file-get-contents.php)) returns the content of the file as a string, including new line characters (`\n`) where appropriate. If we don't want the new lines included, we can use the `FILE_IGNORE_NEW_LINES` option. We might also want to skip empty lines using `FILE_SKIP_EMPTY_LINES`. If the file couldn't be opened, the function returns `FALSE`.

[students.btsi.lu/evegi144/WAD/PHP/files1.php](http://students.btsi.lu/evegi144/WAD/PHP/files1.php)

```
<?php
    $file_name = 'files1.txt';
    $file_string = file_get_contents($file_name, FILE_IGNORE_NEW_LINES |
        FILE_SKIP_EMPTY_LINES);
    if ($file_string) echo $file_string;
    else echo "Could not open $file_name";
?>
```

## Read file contents as an array

`file` (cf. [php.net/manual/en/function.file.php](http://php.net/manual/en/function.file.php)) returns the content of the file as an array. Each element of the array corresponds to a line in the file, with the newline still attached, unless we use the ignore option. If the file couldn't be opened, the function returns `FALSE`.

[students.btsi.lu/evegi144/WAD/PHP/files2.php](http://students.btsi.lu/evegi144/WAD/PHP/files2.php)

```
<?php
$file_name = 'files1.txt';
$file_array = file($file_name, FILE_IGNORE_NEW_LINES | FILE_SKIP_EMPTY_LINES);
if ($file_array) echo '<pre>' . print_r($file_array, TRUE) . '</pre>';
else echo "Could not open $file_name";
?>
```

## Write to a file

Before writing to a new or existing file, we need to make sure the web server has write access to the directory (use `chmod` to change if necessary) and then get a handle to the file via `fopen` (cf. [php.net/manual/en/function.fopen.php](http://php.net/manual/en/function.fopen.php)). Be careful to select the right mode.

Then we can write to the file using `fwrite` (cf. [php.net/manual/en/function.fwrite.php](http://php.net/manual/en/function.fwrite.php)).

[students.btsi.lu/evegi144/WAD/PHP/filewrite1.php](http://students.btsi.lu/evegi144/WAD/PHP/filewrite1.php)

```
<?php
$file_name = 'files2.txt';
// Remember to give the web server write access to the directory.
$file = fopen($file_name, 'a+');
if ($file) fwrite($file, 'This is a test string. ');
else echo '<pre>' . print_r(error_get_last(), TRUE) . '</pre>';
?>
```

## Delete a file

To delete a file, we use the `unlink` function (cf. [php.net/manual/en/function.unlink.php](http://php.net/manual/en/function.unlink.php)).

## Create a zip file

Use the excellent code from [ens.ro/2012/04/06/create-zip-archive-from-directory-in-php](http://ens.ro/2012/04/06/create-zip-archive-from-directory-in-php). Remember to make sure that the web server has write access to the directory that you want to save the zip file in.

## 5.2.20. Classes and objects

Object-oriented programming allows us to align our programs much better with the real world. Instead of having variables and numerous functions seemingly independent of these variables but manipulating them, we can define things or objects that exist in the real world and model their

characteristics and behavior, thereby linking data and functions.

Cf. [php.net/manual/en/language.oop5.basic.php](http://php.net/manual/en/language.oop5.basic.php) and [php.net/manual/en/ref.classobj.php](http://php.net/manual/en/ref.classobj.php).

## Defining a class

[students.btsi.lu/evegi144/WAD/PHP/class1.php](http://students.btsi.lu/evegi144/WAD/PHP/class1.php)

```
<?php

class Car {
    private $color;

    public function get_color() {
        return $this->color;
    }

    public function set_color($color) {
        $this->color = $color;
    }
}

?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>First class example</title>
        <meta charset=utf-8>
    </head>
    <body>
        <main>
            <?php
                $my_car = new Car();
                $my_car->set_color('black');
                echo $my_car->get_color();
                echo '<br>';
                var_dump($my_car);
                echo '<br>';
                print_r($my_car);
            ?>
        </main>
    </body>
</html>
```

## Static properties and functions

[students.btsi.lu/evegi144/WAD/PHP/class2.php](http://students.btsi.lu/evegi144/WAD/PHP/class2.php)

```

<?php

class Database {
    # This is a private static property, so we can use it without creating an object.
    private static $DB_HOST;

    public static function set_host($db_host) {
        self::$DB_HOST = $db_host;
        echo 'Host set to ' . self::$DB_HOST . '.';
    }
}

?>

<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Second class example</title>
        <meta charset=utf-8>
    </head>
    <body>
        <main>
            <?php
                Database::set_host('foxi.ltam.lu');
            ?>
        </main>
    </body>
</html>

```

## Objects as parameters

[php.net/manual/en/language.oop5.references.php](http://php.net/manual/en/language.oop5.references.php)

### 5.2.21. Exceptions

[php.net/manual/en/language.exceptions.php](http://php.net/manual/en/language.exceptions.php)

### 5.2.22. Generators

[php.net/manual/en/language.generators.overview.php](http://php.net/manual/en/language.generators.overview.php)

### 5.2.23. PDF generation

[ourcodeworld.com/articles/read/226/top-5-best-open-source-pdf-generation-libraries-for-php](http://ourcodeworld.com/articles/read/226/top-5-best-open-source-pdf-generation-libraries-for-php)  
[blog.chapagain.com.np/php-easily-create-pdf-on-the-fly](http://blog.chapagain.com.np/php-easily-create-pdf-on-the-fly)

## TCPDF

Generate PDFs with PHP: [www.tcpdf.org](http://www.tcpdf.org). This package is normally already installed as php-tcpdf.

## FPDF

[www.fpdf.org](http://www.fpdf.org)

## dompdf

Convert HTML to PDF using [github.com/dompdf/dompdf](https://github.com/dompdf/dompdf).

### 5.2.24. WebSockets

Currently the Apache web server does not come with a web socket module. We could write our own web socket server in PHP, but why reinvent the wheel when there are good open source libraries available. We'll use [github.com/Devristo/phpws](https://github.com/Devristo/phpws) and adapt it to our needs. Download the zip file and extract it on the web server. If you are interested in the implementation details, take a look at the files. For our purposes, we only need to modify the `demo.php` file, particularly the `onMessage` function. Make a copy of the file and save it as `server.php`.

Here is a sample implementation of `onMessage`, which simply sends the message received to all other clients:

```
public function onMessage(IWebSocketConnection $user, IWebSocketMessage
$msg) {
    $thisuser = $user->getId();
    $msg      = trim($msg->getData());
    $msgback  = WebSocketMessage::create($msg);

    foreach ($this->server->getConnections() as $user)
        if ($user->getId() !== $thisuser) $user->sendMessage($msgback);
}
```

There are other libraries around, such as [github.com/walkor/phpsocket.io](https://github.com/walkor/phpsocket.io) or [Ratchet](https://github.com/Ratchetphp/Ratchet/issues/489) (see [github.com/Ratchetphp/Ratchet/issues/489](https://github.com/Ratchetphp/Ratchet/issues/489) on how to use it with HTTPS).

But, as mentioned before, in order to really work with WebSockets you should consider using Node.js as it is much better suited for this purpose.

### 5.2.25. Web scraping

The following sources can give you a quick overview of what web scraping is and how it can be done:

[en.wikipedia.org/wiki/Web\\_scraping](https://en.wikipedia.org/wiki/Web_scraping)

[stackoverflow.com/questions/3577641/how-do-you-parse-and-process-html-xml-in-php](https://stackoverflow.com/questions/3577641/how-do-you-parse-and-process-html-xml-in-php)

We will look at two approaches, one using PHP and one using JS.

#### using PHP

[students.btsi.lu/evegi144/WAD/PHP/web\\_scrape1.php](https://students.btsi.lu/evegi144/WAD/PHP/web_scrape1.php)

```

<?php
    $URL = 'https://www.dne.lu';
    $html = file_get_contents($URL);
    $dom = new DOMDocument();
    $dom->loadHTML($html);
    echo '<pre>' . print_r($dom->childNodes, true) . '</pre>';
    foreach ($dom->childNodes as $child) var_dump($child);
    //echo $dom->saveHTML(); // In case we manipulate the DOM, we can then save the DOM
as HTML

                                // and send it to the browser.
    /*$meteo_header = $dom->getElementById('meteo-header');
    echo '<pre>' . print_r($meteo_header, true) . '</pre>';*/
    /*$out = "<script>'use strict'; const meteoHeader = ' " . $meteo_header .
    "'</script>";
    echo $out;
    error_log($out);*/
?>

```

## using JS

[students.btsi.lu/evegi144/WAD/PHP/web\\_scrape2.html](https://students.btsi.lu/evegi144/WAD/PHP/web_scrape2.html)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Web scraper</title>
    <script>
      'use strict';

      const scrape = e => {
        const html = e.target.response;
        if (html == 'Invalid URL') console.log(html);
        else {
          const doc = new DOMParser().parseFromString(html, "text/html");
          console.dir(html);
          console.dir(doc.querySelector('.meteo-short'));
        }
      };

      const URL = 'http://www.rtl.lu';
      const req = new XMLHttpRequest();
      req.open('POST', 'web_scrape2.php');
      req.addEventListener('load', scrape);
      req.send(URL);
    </script>
  </head>
  <body>
  </body>
</html>

```

[students.btsi.lu/evegi144/WAD/PHP/web\\_scrape2.php](http://students.btsi.lu/evegi144/WAD/PHP/web_scrape2.php)

```

<?php
  $URL = file_get_contents('php://input');
  if (filter_var($URL, FILTER_VALIDATE_URL)) echo file_get_contents($URL);
  else echo 'Invalid URL';
?>

```

## 5.2.26. Frameworks

[socialcompare.com/en/comparison/php-frameworks-comparison](http://socialcompare.com/en/comparison/php-frameworks-comparison)

[www.codeigniter.com](http://www.codeigniter.com)

## 5.2.27. Tests

**National Elections 2013**

- déi Lénk ☐
- adr ☐
- KPL ☐
- DP ☐
- Piraten ☐
- déi gréng ☐
- LSAP ☐
- CSV ☐
- PID ☐

Party	#	%
déi Lénk	5	2
adr	1	0
KPL	2	1
DP	1	0
Piraten	17	8
déi gréng	139	65
LSAP	3	1
CSV	23	11
PID	22	10
Total	213	100

You have been tasked to develop an online election system ([students.btsi.lu/evegi144/WAD/PHP/Tests/NationalElections2013](http://students.btsi.lu/evegi144/WAD/PHP/Tests/NationalElections2013)) for the Luxembourg national elections 2013. Use [students.btsi.lu/evegi144/WAD/PHP/Tests/NationalElections2013/index\\_skeleton.zip](http://students.btsi.lu/evegi144/WAD/PHP/Tests/NationalElections2013/index_skeleton.zip) as skeleton.

Create a folder **Tests/NationalElections2013** in your main Foxi folder and save your solution under the name **index.php**. Make sure that the group **www-data** has write access to the **NationalElections2013** folder.

The array **\$parties** stores the name of each party and **\$polls** contains the number of votes for each party.

Write the function **get\_total\_number\_of\_votes** that returns the total number of votes. The script generates an HTML table with the votes and percentages for each party. Use **number\_format** to format the percentage numbers.

## Solution

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Elections</title>
    <meta charset=UTF-8>
    <style>
      label {
        float:      left;
        clear:      left;
        width:       60px;
        text-align:  right;
        padding-right: 10px;
        margin-top:  5px;
        display:     inline-block;
      }

      input {
        margin-top: 9px;
      }

      table {
        margin-top: 10px;
        border:     1px solid black;
        border-spacing: 0;
        background-color: green;
      }

      table th, td {
        border: 4px groove green;
        padding: 4px;
      }

      table th {
        color: gold;
        text-align: left;
      }

      table tr:nth-of-type(even) {
        color: lightgreen;
      }

      thead {
        background-color: darkgreen;
      }

      tfoot {
        background-color: orange;
      }
    </style>
```

```
</head>
<body>
  <form>
    <div>
      <label for=1>déi Lénk</label>
      <input type=radio id=1 name=vote <?php if (isset($_GET['vote'])) &&
$_GET['vote']==1)
        echo "checked";?> value=1>
    </div>
    <div>
      <label for=2>adr</label>
      <input type=radio id=2 name=vote <?php if (isset($_GET['vote'])) &&
$_GET['vote']==2)
        echo "checked";?> value=2>
    </div>
    <div>
      <label for=3>KPL</label>
      <input type=radio id=3 name=vote <?php if (isset($_GET['vote'])) &&
$_GET['vote']==3)
        echo "checked";?> value=3>
    </div>
    <div>
      <label for=4>DP</label>
      <input type=radio id=4 name=vote <?php if (isset($_GET['vote'])) &&
$_GET['vote']==4)
        echo "checked";?> value=4>
    </div>
    <div>
      <label for=5>Piraten</label>
      <input type=radio id=5 name=vote <?php if (isset($_GET['vote'])) &&
$_GET['vote']==5)
        echo "checked";?> value=5>
    </div>
    <div>
      <label for=6>déi gréng</label>
      <input type=radio id=6 name=vote <?php if (isset($_GET['vote'])) &&
$_GET['vote']==6)
        echo "checked";?> value=6>
    </div>
    <div>
      <label for=7>LSAP</label>
      <input type=radio id=7 name=vote <?php if (isset($_GET['vote'])) &&
$_GET['vote']==7)
        echo "checked";?> value=7>
    </div>
    <div>
      <label for=8>CSV</label>
      <input type=radio id=8 name=vote <?php if (isset($_GET['vote'])) &&
$_GET['vote']==8)
        echo "checked";?> value=8>
    </div>
```

```

<div>
    <label for=9>PID</label>
    <input type=radio id=9 name=vote <?php if (isset($_GET['vote']) &&
$_GET['vote']==9)
        echo "checked";?> value=9>
</div>
<input type=submit value=Vote>
</form>
<?php
function get_polls() {
    $filename = "poll_result.txt";
    $contents = @file($filename);
    if (!$contents) $array = array(0, 0, 0, 0, 0, 0, 0, 0, 0, 0);
    else $array = explode('x', $contents[0]);
    if (isset($_GET['vote'])) {
        $array[$_GET['vote'] - 1]++;
        $fp = fopen($filename, "w");
        fputs($fp, implode('x', $array));
        fclose($fp);
    }
    return $array;
}

function get_total_number_of_votes() {
    $sum = 0;
    global $polls;
    //foreach ($polls as $poll) $sum += $poll;
    for ($i = 0; $i < count($polls); $i++) $sum += $polls[$i];
    return $sum;
}

$parties = array('déi Lénk', 'adr', 'KPL', 'DP', 'Piraten', 'déi gréng', 'LSAP',
'CSV',
    'PID');
$polls = get_polls();
$total = get_total_number_of_votes();
echo "<table><thead><tr><th>Party</th><th>#</th><th>%</th></tr></thead>";
for ($i = 0; $i < count($parties); $i++)
    echo "<tr><td>$parties[$i]</td><td>$polls[$i]</td><td>"
        . ($total > 0 ? 100 * number_format($polls[$i] / $total, 2) : "0") .
        "</td></tr>";
echo
"<tfoot><tr><td>Total</td><td>$total</td><td>100</td></tr></tfoot></table>";
?>
</body>
</html>

```

The skeleton provides the HTML frame with embedded CSS and the form. The CSS part hints already at the elements that we'll need to generate the poll result table. Note the rule `table tr:nth-of-type(even)` on line 38, which creates the color alternation between even and odd lines.

The form illustrates how we can implement mutually exclusive radio buttons and remember the last selection. For this to work, all radio inputs need to have the same `name`, but different `value` attributes. We use PHP to check for each radio input whether it was checked when the form was submitted. If it was, we'll check it again. The only modification required in the first 104 lines of the skeleton is in line 89, where we need to add ``type=submit` to turn the input into a submit button. Otherwise the user has no way to send his vote to the server.

It is important to realize that the `get_polls` function should not be called more than once, given that it increases the number of votes of the chosen party during each call.

The function `get_total_number_of_votes` is superfluous, as PHP provides the function [php.net/manual/en/function.array-sum.php](http://php.net/manual/en/function.array-sum.php)[``array_sum^`]. We'll still develop it, for pedagogical reasons. In fact we've done this already in a previous exercise. The problem statement did not specify whether the function should take a parameter or not, although the skeleton suggests in line 114 that the function should take no parameters. In this case, we need to work with the global variable `$polls`. As we've seen in [Variable scope](#), we need to use the keyword `global` to make a global variable visible inside a function. Alternatively, we can pass `$polls` as parameter to our function.

Now we can generate the HTML table.

## WMOTU Speed Calc

User name:  Password:

---

## WMOTU Speed Calc

From our offices serving 5 continents we deliver unmatched calculation speed at insane prices.

### Enter your text and see the magic

Number of words:

### Contact us

Registered users: user admin

### Convert an integer to binary

Binary equivalent:

### Calculate the volume of a room

width:  length:  height:

Volume:

### Calculate your power cost

Average operating time:

☐ daily  
☐ weekly  
☐ monthly

Cost:

Develop the web site ([www.youtube.com/watch?v=Dm-mPRNqYQc](https://www.youtube.com/watch?v=Dm-mPRNqYQc)) for WMOTU Speed Calc. Create a folder **Tests/WMOTUSpeedCalc** in your main Foxi folder and save your solution under the name **index.php**. Implement the following:

1. The whole app consists of a single script **index.php**.
2. Login with cookie-based sessions. User data (user name, password) are stored statically in an array. You need 2 users with name/password **user/user** and **admin/admin**. The session cookie is only sent via SSL. Session IDs are regenerated every 20 seconds.
3. Logout.

4. The page contains a short description of the company.
5. Everyone can enter a text into a form. After submission the number of words will be displayed. The form provides a check box. If it is selected, the first letter of every word in the text will be capitalized. Example: "WSERS1 is great" → "WSERS1 Is Great".
6. Everyone can send a contact request. Form submission will send an email to your teacher. The subject and from fields must have sensible values.
7. For a logged in user the screen background color is green. For an admin user, the background color is gold. For everyone else the background color is the default value, i.e. white.
8. The administrator sees a list of all registered users.
9. A logged in user can perform the following calculations:
  - a. Convert an integer to binary.
  - b. Calculate the volume of a room given its width, length and height.
  - c. Calculate the annual cost of the user's power consumption based on the average operation time, which can be provided on a daily, weekly or monthly basis.

### Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/WMOTUSpeedCalc](https://students.btsi.lu/evegi144/WAD/PHP/Tests/WMOTUSpeedCalc)

```
<?php
# SESSION HANDLING
# We need to handle the session related stuff before we send anything (HTML, CSS,
# JavaScript) to the browser, as the session cookie is sent in the HTTP header,
# which needs to be sent before the HTTP body containing the web page content.
ini_set('session.cookie_secure', true); # Only send session id cookie over SSL.
ini_set('session.use_only_cookies', true); # Pass session IDs only via cookies, not
URL.
ini_set('session.cookie_path', dirname(htmlspecialchars($_SERVER['PHP_SELF'])));
if (!isset($_SERVER['HTTPS'])) # If SSL is not active, activate it.
    header('Location: https://' . $_SERVER['HTTP_HOST'] .
        htmlspecialchars($_SERVER['PHP_SELF']));
if (!isset($_SESSION)) session_start(); # If we are not already in a session, start
one.
# After 20 seconds we'll generate a new session ID to prevent a session
# fixation attack (cf. PHP cookbook p. 338).
if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 20)) {
    session_regenerate_id();
    $_SESSION['generated'] = time();
}

# GLOBAL VARIABLE DECLARATION
$users = array(array('user', 'user'), array('admin', 'admin')); # Static user array.
$text = $word_count = $number = $volume = $cost = ''; # Global vars for form
processing.

# LOGOUT PROCESSING
```

```

# If the user has clicked the logout button, we delete the session array, expire the
# session cookie, destroy the session and reload the page.
if (isset($_POST['logout'])) {
    if (!isset($_SESSION)) session_start();
    $_SESSION = array();
    if (session_id() != "" || isset($_COOKIE[session_name()]))
setcookie(session_name(),
    '', time() - 2592000, '/');
    session_destroy();
    header('Location: https://' . $_SERVER['HTTP_HOST'] .
dirname($_SERVER['PHP_SELF']));
}

# LOGIN PROCESSING
elseif (isset($_POST['username']) && isset($_POST['password'])) {
    if ($_POST['username'] === $users[0][0] && $_POST['password'] === $users[0][1])
        $_SESSION['user_id'] = 0;
    elseif ($_POST['username'] === $users[1][0] && $_POST['password'] ===
$users[1][1])
        $_SESSION['user_id'] = 1;
}

# TEXT FORM PROCESSING
elseif (isset($_POST['textBox'])) {
    $text      = $_POST['textBox'];
    $word_count = str_word_count($text);
    if (isset($_POST['checkBox'])) $text = ucwords($text);
}

# CONTACT FORM PROCESSING
elseif (isset($_POST['contactBox'])) {
    $to      = 'gilles.everling@education.lu';
    $subject = 'Contact request';
    $message = $_POST['contactBox'];
    $headers = 'From: T2IF2@ltam.lu';
    mail($to, $subject, $message, $headers);
}

# INTEGER TO BINARY CONVERSION FORM PROCESSING
elseif (isset($_POST['number'])) $number = sprintf('%b', $_POST['number']);

# ROOM VOLUME FORM PROCESSING
elseif (isset($_POST['width']) && isset($_POST['length']) &&
isset($_POST['height']))
    $volume = $_POST['width'] * $_POST['length'] * $_POST['height'];

# ANNUAL POWER COST FORM PROCESSING
elseif (isset($_POST['operatingTime']) && isset($_POST['duration'])) {
    if ($_POST['duration'] === 'day') $cost = $_POST['operatingTime'] * 365;
    elseif ($_POST['duration'] === 'week') $cost = $_POST['operatingTime'] * 52;
    else $cost = $_POST['operatingTime'] * 12;
}

```

```

}
?>
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU Speed Calc</title>
    <meta charset=UTF-8>
  </head>
  <body>
    <main>
      <!-- LOGIN INDEPENDENT STUFF -->
      <!-- LOGIN/LOGOUT FORM -->
      <form method=post>
        User name:
        <input type=text name=username required autofocus>
        Password:
        <input type=password name=password required>
        <input type=submit value=Login>
      </form>

      <!-- LOGOUT FORM -->
      <form method=post>
        <input type=submit name=logout value=Logout>
      </form>

      <!-- COMPANY DESCRIPTION -->
      <hr>
      <h1>WMOTU Speed Calc</h1>
      <p>
        From our offices serving 5 continents we deliver unmatched calculation speed
at
        insane prices.
      </p>

      <!-- TEXT FORM -->
      <form method=post>
        <h2>Enter your text and see the magic</h2>
        <textarea name=textBox required><?php echo $text; ?></textarea>
        <input type=checkbox name=checkBox>
        <input type=submit value=Send>
      </form>
      Number of words: <?php echo $word_count; ?><br>

      <!-- CONTACT FORM -->
      <form method=post>
        <h2>Contact us</h2>
        <textarea name=contactBox required></textarea>
        <input type=submit value=Send>
      </form>

      <!-- LOGIN DEPENDENT STUFF -->

```

```

<?php
    if (isset($_SESSION['user_id'])) { # If a user is logged in.
        if ($_SESSION['user_id'] == 0) $color = 'green'; # Normal user -> green bg.
        else { # Admin user gets a list of all users and golden bg.
            echo '<p>Registered users: ';
            for ($i = 0; $i < count($users); $i++) echo " {$users[$i][0]} ";
            echo '</p>';
            $color = 'gold';
        }
        echo "<script>document.getElementsByTagName('body')[0].style." .
            "backgroundColor = '$color'</script>"; # Set bg color based on login.
        echo <<<EOT
    <!-- INTEGER TO BINARY CONVERSION FORM -->
    <h2>Convert an integer to binary</h2>
    <form method=post>
        <input type=number name=number required>
        <input type=submit value="Convert to binary">
    </form>
    Binary equivalent: $number

    <!-- ROOM VOLUME CALCULATION FORM -->
    <h2>Calculate the volume of a room</h2>
    <form method=post>
        width: <input type=number name=width required>
        length: <input type=number name=length required>
        height: <input type=number name=height required>
        <input type=submit value="Calculate volume">
    </form>
    Volume: $volume

    <!-- ANNUAL POWER COST FORM -->
    <h2>Calculate your power cost</h2>
    <form method=post>
        Average operating time: <input type=number name=operatingTime required><br>
        <input type="radio" name=duration value=day>daily<br>
        <input type="radio" name=duration value=week>weekly<br>
        <input type="radio" name=duration value=month>monthly
        <input type=submit value="Calculate power cost">
    </form>
    Cost: $cost
EOT;
    }
?>
</main>
</body>
</html>

```

# Shopping list

Desktop	599.99 €
Mobile	299.99 €
Web cam	19.99 €
<b>Total</b>	<b>919.97 €</b>

Name	Price €
<input type="text"/>	<input type="text"/>
<input type="button" value="Add"/>	

Create the web page shown at [youtu.be/aWo78bwl0s8](https://youtu.be/aWo78bwl0s8).

## Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/WMOTUShop](https://students.btsi.lu/evegi144/WAD/PHP/Tests/WMOTUShop)

```
<?php
    $items = []; // All items will be stored in this array.

    if (isset($_POST['BUTTON_add'])) { // If the form has been submitted.
        if (isset($_POST['DATA_items'])) $items = $_POST['DATA_items']; // Existing items.
        if ($_POST['DATA_name'] !== '') { // Add new item.
            $items[] = ['name' => $_POST['DATA_name'],
                'price' => str_replace(',', '.', $_POST['DATA_price'])]; // Convert , to .
        }
    }
?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <meta charset=utf-8>
        <title>WMOTU Shop</title>
        <style>
            div.left {
                display: inline-block;
                width: 250px;
            }

            div.right {
                display: inline-block;
                width: 80px;
                text-align: right;
            }

            input[type=text] {
                width: 100%;
            }
        </style>
    </head>
    <body>
        <table>
            <thead>
                <tr>
                    <th>Name</th>
                    <th>Price €</th>
                </tr>
            </thead>
            <tbody>
                <tr>
                    <td><input type="text"></td>
                    <td><input type="text"></td>
                </tr>
                <tr>
                    <td><input type="button" value="Add"></td>
                    <td></td>
                </tr>
            </tbody>
        </table>
    </body>
</html>
```

```

</style>
</head>
<body>
  <h1>Shopping list</h1>
  <form method=POST>
    <?php // Add the existing items as hidden inputs so they'll be included in the
form.
      $total = 0;
      $i = 0;
      foreach ($items as $item) {
        echo "<input type=hidden name=DATA_items[$i][name] value=" . $item['name'] .
">";
        echo "<input type=hidden name=DATA_items[$i][price] value=" . $item['price']
. ">";
        echo '<div class=left>' . $item['name'] . '</div>';
        echo '<div class=right>' . number_format($item['price'], 2) . '
&euro;</div><br>';
        $total += $item['price'];
        $i++;
      }
    ?>
    <div class=left style="border-top: 1px #000 solid; font-weight:
bold;">Total</div>
    <div class=right style="border-top: 1px #000 solid; font-weight: bold;">
      <?php echo number_format($total, 2) ?> &euro;</div>
    <br><br><br>
    <div class=left>Name</div>
    <div class=right>Price &euro;</div>
    <br>
    <div class=left><input name=DATA_name autofocus></div>
    <div class=right><input name=DATA_price></div>
    <br>
    <button name=BUTTON_add>Add</button>
  </form>
</body>
</html>

```

## Quiz and Shopping Center

[Shop](#)

## Quiz

You've answered 2 questions correctly out of 3.

8\*18



[youtu.be/dunVlaxdZfI](https://youtu.be/dunVlaxdZfI)

Create a folder **Tests/QaSC** in your main Foxi folder and save your solution under the name **index.php**.

The initial page shows a navigation bar with two links, one to the quiz and one to the shop. The main part shows a welcome message.

The quiz link calls the same script. It's navigation bar shows only the link to the shop. The main part shows a heading, an empty line and then the problem to be solved, which consists of 2 random integers between 0 and 20 and an operator, which is either **+**, **-** or **\***. You may find the **eval** function (cf. **eval**) helpful. Beneath is an input and a submit button. After the user has submitted an answer, the text "You've answered " followed by the number of correctly answered questions followed by "question" followed by an "s" if several questions have been answered correctly followed by " correctly out of " followed by the total number of questions answered is displayed.

The shop link calls the same script. It's navigation bar shows only the link to the quiz. The main part shows a heading, then the listing of all the items added so far with their name and price and the total price. Beneath are 2 labels and 2 inputs for the name and price of a new item to be added to the shopping list and the submit button.

### Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/QuizAndShoppingCenter](https://students.btsi.lu/evegi144/WAD/PHP/Tests/QuizAndShoppingCenter)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=utf-8>
    <title>Quiz and Shopping Center</title>
    <style>
      div.left {
        display: inline-block;
        width: 250px;
      }

      div.right {
        display: inline-block;
        width: 80px;
        text-align: right;
      }

      input[type=text] {
        width: 100%;
      }
    </style>
  </head>
  <body>
    <nav>
      <?php
        if (!isset($_REQUEST['quiz'])) echo '<a href=?quiz>Quiz</a> ';
        if (!isset($_REQUEST['shop'])) echo '<a href=?shop>Shop</a> ';
```

```

?>
</nav>
<main>
  <?php
    if (!isset($_REQUEST['quiz']) && !isset($_REQUEST['shop'])) echo '<h1>Welcome
to the Quiz and Shopping Center</h1>';
    elseif (isset($_REQUEST['quiz'])) {
      echo '<h1>Quiz</h1>';
      $a = rand(0, 20);
      $b = rand(0, 20);
      $ops = ['+', '-', '*'];
      $op = $ops[rand(0, 2)];
      $str = $a . $op . $b;
      eval('$correct_answer = ' . $str . ';');
      $correct_count = 0;
      $total_count = 0;
      if (isset($_POST['answer'])) {
        $correct_count = $_POST['correct_count'];
        $total_count = $_POST['total_count'] + 1;
        if ($_POST['answer'] === $_POST['correct_answer']) $correct_count++;
        echo "You've answered $correct_count question" .
          (($correct_count < 2) ? '' : 's') . " correctly out of
          $total_count.";
      }
      echo '<br>' . $str;
    }
    ?>
    <form method=POST>
      <?php
        echo "<input type=hidden name=correct_answer value=$correct_answer>";
        echo "<input type=hidden name=correct_count value=$correct_count>";
        echo "<input type=hidden name=total_count value=$total_count>";
      ?>
      <input type=number name=answer autofocus>
      <input type=submit name=quiz value=Check>
    </form>
  <?php
  }
  if (isset($_REQUEST['shop'])) {
    $items = []; // All items will be stored in this array.
    if (isset($_POST['shop'])) { // If the form has been submitted.
      if (isset($_POST['items'])) {$items = $_POST['items']; // Existing items.
        error_log(print_r($_POST['items'], TRUE));}
      if ($_POST['name'] !== '') // Add new item.
        $items[] = ['name' => $_POST['name'],
                    'price' => str_replace(',', '.', $_POST['price'])]; //
Convert , to .
    }
    ?>
    <h1>Shopping list</h1>
    <form method=POST>
      <?php // Add existing items as hidden inputs so they'll be included in the

```

```

form.
    $total = 0;
    $i      = 0;
    foreach ($items as $item) {
        echo "<input type=hidden name=items[$i][name] value=" . $item['name']
    . ">";
        echo "<input type=hidden name=items[$i][price] value=" .
$item['price'] . ">";
        echo '<div class=left>' . $item['name'] . '</div>';
        echo '<div class=right>' . number_format($item['price'], 2) . '
&euro;</div>
        <br>';
        $total += $item['price'];
        $i++;
    }
    ?>
    <div class=left style="border-top: 1px #000 solid; font-weight: bold;">
    Total
    </div>
    <div class=right
        style="border-top: 1px #000 solid; font-weight: bold;">
        <?php echo number_format($total, 2) ?> &euro;</div>
    <br><br><br>
    <div class=left>Name</div>
    <div class=right>Price &euro;</div>
    <br>
    <div class=left><input name=name autofocus></div>
    <div class=right><input name=price></div>
    <br>
    <button name=shop>Add</button>
    </form>
    <?php
    }
    ?>
    </main>
    </body>
    </html>

```

## Roll The Dice

Results	Statistics
3	1 has been thrown 3 times, i.e. 11.54%
2	2 has been thrown 4 times, i.e. 15.38%
5	3 has been thrown 4 times, i.e. 15.38%
6	4 has been thrown 3 times, i.e. 11.54%
4	5 has been thrown 7 times, i.e. 26.92%
6	6 has been thrown 5 times, i.e. 19.23%
Throw	

Develop the following app using only HTML, CSS and PHP: [youtu.be/Vbwny2QUi4Y](https://youtu.be/Vbwny2QUi4Y)

## Solution

This solution uses hidden fields just to illustrate their usage. A better approach would be to use sessions.

[students.btsi.lu/evegi144/WAD/PHP/Tests/RollTheDice](https://students.btsi.lu/evegi144/WAD/PHP/Tests/RollTheDice)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Roll The Dice</title>
    <meta charset=utf-8>
    <style>
      #div1 {
        display: inline-block;
        width: 150px;
      }

      #div2 {
        display: inline-block;
        width: 350px;
      }

      select {
        width: 100%;
      }
    </style>
  </head>
  <body>
    <main>
      <form method=POST>
        <?php
          $throws = [];
          $stats = [0, 0, 0, 0, 0, 0];
          if (isset($_POST['throw'])) {
            if (isset($_POST['throws'])) $throws = explode(',', $_POST['throws']);
            $rnd = rand(1, 6);
            $throws[] = $rnd;
            foreach ($throws as $throw) $stats[$throw - 1]++;
            echo "<input type=hidden name=throws value=" . implode(',', $throws) .
">";
          }
        ?>
        <div id=div1>
          Results<br>
          <select size=6>
            <?php
              foreach ($throws as $throw) echo "<option>$throw</option>";
            ?>
          </select>
        </div>
```

```

<div id=div2>
  Statistics<br>
  <select size=6>
    <?php
      $sum = array_sum($stats);
      for ($i = 0; $i < count($stats); $i++)
        echo "<option>" . ($i + 1) . " has been thrown {$stats[$i]}
            times" . ($sum > 0 ? sprintf(', i.e. %1.2f', 100 * $stats[$i] /
array_sum
            ($stats)) . ' %' : '') . "</option > "; // Avoid division by zero!
    ?>
  </select>
</div>
<br>
<button name=throw>Throw</button>
</form>
</main>
</body>
</html>

```

## Message board

---

01-02-2015 16:06:20: Another  
one  
01-02-2015 16:06:09: Multiline message  
continues  
here  
...  
01-02-2015 16:05:51: Single line message

Develop a simple message board, as shown in [youtu.be/1KoDEvFF7q0](https://youtu.be/1KoDEvFF7q0), using only HTML, CSS and PHP. You may use sessions.

Messages are stored in a 2-dimensional array, the first element of each message being the timestamp and the second the message text.

Define and use the two functions `displayMessage($msg)` and `displayMessages()`. The former displays a given message with its timestamp. The latter displays all messages, with the newest at the top.

Use colors `lavender`, `lightskyblue` and `magenta`.

## Solution

Watch the solution video at [youtu.be/hAsZs4amUkA](https://youtu.be/hAsZs4amUkA).

[students.btsi.lu/evegi144/WAD/PHP/Tests/MessageBoard](https://students.btsi.lu/evegi144/WAD/PHP/Tests/MessageBoard)

```

<?php
    session_start(); // We need to start a session in order to be able to use $_SESSION.
    date_default_timezone_set('Europe/Luxembourg'); // Set default timezone.

    /* If the messages array does not yet exist or if the clear button has been pressed,
       we need to initialize the messages array with an empty array. */
    if (!isset($_SESSION['messages']) || isset($_POST['clear'])) $_SESSION['messages'] =
[];
    if (isset($_POST['msg'])) // If a message text has been sent, store it with date and
time.
        $_SESSION['messages'][] = [date('d-m-Y H:i:s'), $_POST['msg']];

    /**
     * Display a single message consisting of timestamp and text.
     * @param $msg message array consisting of timestamp and message text
     */
    function displayMessage($msg)
    {
        echo "<div><span>$msg[0]: </span>" . nl2br($msg[1]) . "</div>";
    }

    /**
     * Display all messages, newest first.
     */
    function displayMessages()
    {
        for ($i = count($_SESSION['messages']) - 1; $i >= 0; $i--)
            displayMessage($_SESSION['messages'][$i]);
    }

?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Message Board</title>
        <meta charset=utf-8>
        <style>
            div:nth-of-type(odd) {
                background-color: lavender;
            }

            div:nth-of-type(even) {
                background-color: lightskyblue;
            }

            span {
                color: magenta;
            }
        </style>
    </head>

```

```

<body>
  <main>
    <form method=post>
      <textarea name=msg autofocus required></textarea>
      <button>Send</button>
    </form>
    <hr>
    <form method=post>
      <button name=clear>Clear messages</button>
    </form>
    <?php
      displayMessages(); // Display all messages.
    ?>
  </main>
</body>
</html>

```

## Book Shop

Quantity	Book Title	Price
3	JavaScript - The Definitive Guide	\$149.97
1	HTML5 for Masterminds	\$39.95
2	PHP and MySQL Web Development	\$119.98
1	PHP in a Nutshell	\$29.95
0	Advanced PHP Programming	\$0
Number of articles: 7 Total: \$339.85		

Develop the **single file** app exactly as shown at [youtu.be/Er7XZmGWNkg](https://youtu.be/Er7XZmGWNkg) using only HTML, CSS and PHP and taking the following into account:

1. Use the skeleton at `\url{https://foxi.ltam.lu/PROF/evegi144/T2IF2_WSERS/WAD/PHP5/Tests/BookShop}`.
2. The book titles, authors and prices are all stored in a single multidimensional array.
3. Write a function `get_total`, which takes a single parameter from which it computes and returns the total cost of all items currently in the shopping cart.
4. Write a function `get_table`, which takes a single parameter from which it generates and returns the HTML table with the contents of the shopping cart.

## Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/BookShop/Solution](https://students.btsi.lu/evegi144/WAD/PHP/Tests/BookShop/Solution)

```

<?php
    session_start();
    $books = [['JavaScript - The Definitive Guide', 'David Flanagan', 49.99],
        ['HTML5 for Masterminds', 'J.D. Gauchat', 39.95],
        ['PHP and MySQL Web Development', 'Luke Welling and Laura Thomson', 59.99],
        ['PHP in a Nutshell', 'Paul Hudson', 29.95],
        ['Advanced PHP Programming', 'George Schlossnagle', 49.99]];
    if (!isset($_SESSION['cart_items']) || isset($_POST['empty']))
        $_SESSION['cart_items'] = [0, 0, 0, 0, 0];
    if (isset($_POST['selected_book']))
        $_SESSION['cart_items'][$_POST['selected_book']]++;

    function get_total($cart)
    {
        global $books;
        $total = 0;
        for ($i = 0; $i < count($cart); $i++) $total += $books[$i][2] * $cart[$i];
        return $total;
    }

    function get_table($cart)
    {
        global $books;
        $table = '<table>';
        for ($i = 0; $i < count($cart); $i++) {
            $table .= "<tr><td>$cart[$i]</td><td>" . $books[$i][0] . "</td><td>$" .
                $books[$i][2] *
                $cart[$i] .
                "</td></tr>";
        }
        $table .= "<tr><th colspan=3>Number of articles: " . array_sum($cart) . " Total:
    $" .
            get_total($cart) . '</th></tr>';
        $table .= '</table>';
        return $table;
    }

?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Book Shop</title>
        <meta charset=utf-8>
        <style>
            body {
                background: linear-gradient(darkgreen, lightgreen) fixed;
            }

            #div1 {
                float: left;
            }

```

```

#div2 {
    display: inline-block;
    width: 350px;
}

select {
    width: 100%;
    vertical-align: top;
    background-color: khaki;
}

table {
    border: 2px solid blue;
}

td, th {
    border: 1px inset blue;
    padding: 5px;
}

button {
    background-color: lightblue;
}

h1 {
    text-shadow: 2px 2px 1px gold;
}
</style>
</head>
<body>
<header>
<h3>
<a
<?php
    if (isset($_GET['credits'])) echo 'href=?book_shop>Book Shop';
    else echo 'href=?credits>Credits';
?>
</a>
</h3>
<hr>
</header>
<main>
<?php
    if (isset($_GET['credits'])) {
        ?>
        <h1>Web Server Side Scripting is the greatest thing on Earth</h1>
    } else {
        ?>
        <div id=div1>

```

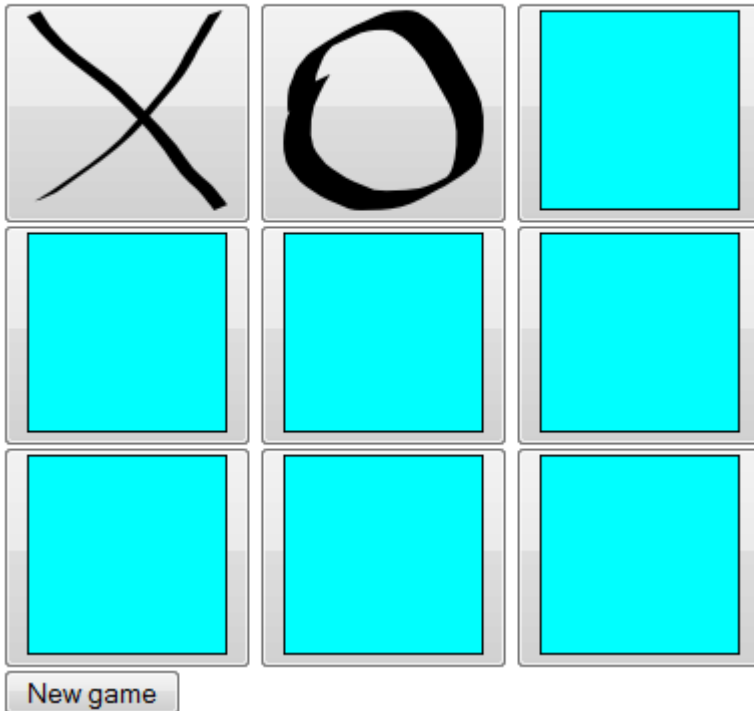
```

        <form method=post>
            <select name=selected_book>
                <?php
                    for ($i = 0; $i < count($books); $i++) {
                        echo "<option value=" . $i . ">" . $books[$i][0] . " - $" .
                            $books[$i][2] . "</option>";
                    }
                ?>
            </select>
            <br>
            <button>Add</button>
        </form>
        <form method=post>
            <button name=empty>Empty cart</button>
        </form>
    </div>
    <div id=div2>
        <?php
            echo get_table($_SESSION['cart_items']);
        ?>
    </div>
<?php
}
?>
</main>
</body>
</html>

```

## TicTacToe

# Player 1, your turn



Develop the **single file** app exactly as shown at [youtu.be/pZaja-g-o2U](https://youtu.be/pZaja-g-o2U) using only HTML, CSS and PHP and taking the following into account:

1. Use the skeleton at [foxi.ltam.lu/PROF/evegi144/T2IF2\\_WSERS/WAD/PHP5/Tests/TicTacToe](https://foxi.ltam.lu/PROF/evegi144/T2IF2_WSERS/WAD/PHP5/Tests/TicTacToe).
2. The play field is stored in a simple array.
3. Each button contains one of the three images ([image1](#), [image2](#) and [image3](#)), but these `img` tags are generated by a function `get_img`, which takes the play field position as parameter (i.e. an integer from [0, 8]) and returns the complete `img` tag.
4. The two players play in alternation. The current player is indicated in the header.
5. If a player clicks an occupied field, nothing happens.
6. Pressing the "New game" button clears the play field and asks player 1 to make a move.

## Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/TicTacToe/Solution](https://students.btsi.lu/evegi144/WAD/PHP/Tests/TicTacToe/Solution)

```
<?php
session_start();
define('NONE', 0);
define('PLAYER_ONE', 1);
define('PLAYER_TWO', 2);
if (!isset($_SESSION['field']) || isset($_POST['new_game'])) {
    $_SESSION['field'] = [NONE, NONE, NONE, NONE, NONE, NONE, NONE, NONE, NONE];
    $_SESSION['player1'] = true;
    $_SESSION['h1'] = "Player 1, your turn";
} elseif (isset($_POST['b'])) {
```

```

    if ($_SESSION['field'][$_POST['b'] - 1] === NONE) {
        $_SESSION['field'][$_POST['b'] - 1] = $_SESSION['player1'] ? PLAYER_ONE :
PLAYER_TWO;
        $_SESSION['player1'] = !$_SESSION['player1'];
        $_SESSION['h1'] = $_SESSION['player1'] ? "Player 1, your turn" : "Player 2, your
turn";
    }
}

function get_img($i) {
    if ($_SESSION['field'][$i - 1] === NONE)
        return "<img src=empty100x100.png width=100 height=100 alt=empty>";
    elseif ($_SESSION['field'][$i - 1] === PLAYER_ONE)
        return "<img src=X100x100.png width=100 height=100 alt=X>";
    else return "<img src=O100x100.png width=100 height=100 alt=O>";
}

?>
<!DOCTYPE html>
<html lang=en>
<head>
    <title>TicTacToe</title>
    <meta charset=utf-8>
    <style>
        section {
            display: table;

            article {
                display: table-row;

                input {
                    display: table-cell;
                }
            }
        }
    </style>
</head>
<body>
    <main>
        <h1><?php echo $_SESSION['h1']; ?></h1>
        <form method=post>
            <section>
                <article>
                    <button name=b value=1><?php echo get_img(1); ?></button>
                    <button name=b value=2><?php echo get_img(2); ?></button>
                    <button name=b value=3><?php echo get_img(3); ?></button>
                </article>
                <article>
                    <button name=b value=4><?php echo get_img(4); ?></button>
                    <button name=b value=5><?php echo get_img(5); ?></button>
                    <button name=b value=6><?php echo get_img(6); ?></button>
                </article>
            </section>
        </form>
    </main>
</body>
</html>

```

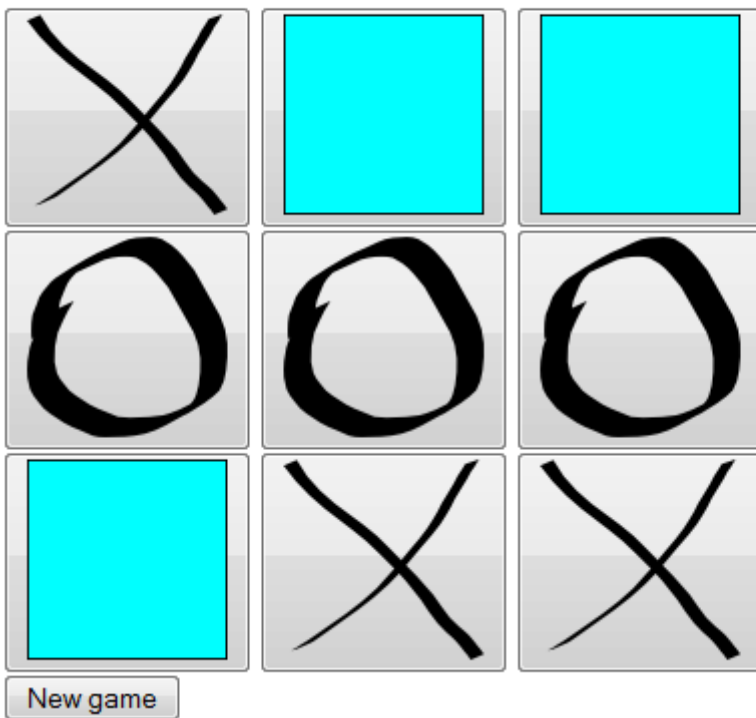
```

</article>
<article>
  <button name=b value=7><?php echo get_img(7); ?></button>
  <button name=b value=8><?php echo get_img(8); ?></button>
  <button name=b value=9><?php echo get_img(9); ?></button>
</article>
</section>
<button name=new_game>New game</button>
</form>
</main>
</body>
</html>

```

## TicTacToe Pro

**Congratulations player 2, you won!**



Enhance your TicTacToe app as shown at [youtu.be/fvKEeKhjCDI](https://youtu.be/fvKEeKhjCDI).

## Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/TicTacToe/SolutionPro](https://students.btsi.lu/evegi144/WAD/PHP/Tests/TicTacToe/SolutionPro)

```

<?php
  session_start();
  define('NONE', 0);
  define('PLAYER_ONE', 1);
  define('PLAYER_TWO', 2);
  define('DRAW', 3);
  if (!isset($_SESSION['field']) || isset($_POST['new_game'])) {

```

```

$_SESSION['field'] = [NONE, NONE, NONE, NONE, NONE, NONE, NONE, NONE, NONE];
$_SESSION['player1'] = true;
$_SESSION['num_empty_fields'] = 9;
$_SESSION['status'] = NONE;
$_SESSION['h1'] = "Player 1, your turn";
} elseif (isset($_POST['b']) && isset($_SESSION['status']) && $_SESSION['status']
=== NONE) {
    if ($_SESSION['field'][$_POST['b'] - 1] === NONE) {
        $_SESSION['field'][$_POST['b'] - 1] = $_SESSION['player1'] ? PLAYER_ONE :
PLAYER_TWO;
        $_SESSION['player1'] = !$_SESSION['player1'];
        $_SESSION['num_empty_fields']--;
    }
    $_SESSION['status'] = real_game_over();
    if ($_SESSION['status'] === NONE)
        $_SESSION['h1'] = $_SESSION['player1'] ? "Player 1, your turn" : "Player 2, your
turn";
    elseif ($_SESSION['status'] === PLAYER_ONE)
        $_SESSION['h1'] = "Congratulations player 1, you won!";
    elseif ($_SESSION['status'] === PLAYER_TWO)
        $_SESSION['h1'] = "Congratulations player 2, you won!";
    else $_SESSION['h1'] = "Draw";
}

function game_over() {
    $count = 0;
    foreach ($_SESSION['field'] as $field)
        if ($field !== NONE) $count++;
    if ($count === 9) return true;
    else return false;
}

function real_game_over() {
    $winners = [[0, 1, 2], [3, 4, 5], [6, 7, 8], [0, 3, 6], [1, 4, 7], [2, 5, 8], [0,
4, 8],
    [2, 4, 6]];
    foreach ($winners as $winner) {
        if ($_SESSION['field'][$winner[0]] === PLAYER_ONE &&
$_SESSION['field'][$winner[1]] ===
        PLAYER_ONE && $_SESSION['field'][$winner[2]] === PLAYER_ONE
        ) return PLAYER_ONE;
        elseif ($_SESSION['field'][$winner[0]] === PLAYER_TWO &&
$_SESSION['field'][$winner[1]]
        === PLAYER_TWO && $_SESSION['field'][$winner[2]] === PLAYER_TWO
        ) return PLAYER_TWO;
    }
    if (game_over()) return DRAW;
    else return NONE;
}

function get_img($i) {

```

```

        if ($_SESSION['field'][$i - 1] === NONE)
            return "<img src=empty100x100.png width=100 height=100 alt=empty>";
        elseif ($_SESSION['field'][$i - 1] === PLAYER_ONE)
            return "<img src=X100x100.png width=100 height=100 alt=X>";
        else return "<img src=O100x100.png width=100 height=100 alt=O>";
    }

?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>TicTacToe Pro</title>
        <meta charset=utf-8>
        <style>
            section {
                display: table;
            }

            article {
                display: table-row;
            }

            input {
                display: table-cell;
            }
        </style>
    </head>
    <body>
        <main>
            <h1><?php echo $_SESSION['h1']; ?></h1>
            <form method=post>
                <section>
                    <article>
                        <button name=b value=1><?php echo get_img(1); ?></button>
                        <button name=b value=2><?php echo get_img(2); ?></button>
                        <button name=b value=3><?php echo get_img(3); ?></button>
                    </article>
                    <article>
                        <button name=b value=4><?php echo get_img(4); ?></button>
                        <button name=b value=5><?php echo get_img(5); ?></button>
                        <button name=b value=6><?php echo get_img(6); ?></button>
                    </article>
                    <article>
                        <button name=b value=7><?php echo get_img(7); ?></button>
                        <button name=b value=8><?php echo get_img(8); ?></button>
                        <button name=b value=9><?php echo get_img(9); ?></button>
                    </article>
                </section>
                <button name=new_game>New game</button>
            </form>
        </main>

```

```
</body>
</html>
```

## WSERS1 Shop

### Home

Article	Price	Quantity
Desktop	599.99	8
Mobile	299.99	6
USB cam	19.99	7

Develop the **single file** app exactly as shown at [youtu.be/zmsARb06Ths](https://youtu.be/zmsARb06Ths) using only HTML, CSS and PHP and taking the following into account:

1. The header shows a link to the shop if the home page is displayed, otherwise a link to the home page.
2. The home page allows the user to enter his/her name, which is then displayed embedded in a welcome message.
3. The shop requires the following:
  - a. A two-dimensional array, in which you store a number of items as arrays. Each item has a name and a price.
  - b. Function `get_table`, which takes a two-dimensional item array as parameter and returns a string containing the HTML table. This table includes the names and prices from the array as well as a random quantity calculated for each article from [1, 10].

### Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/WSERS1Shop/Solution](https://students.btsi.lu/evegi144/WAD/PHP/Tests/WSERS1Shop/Solution)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WSERS1 Shop</title>
    <meta charset=utf-8>
    <style>
      table, th, td {
        border: 1px solid black;
        border-collapse: collapse;
        padding: 5px;
      }
    </style>
  </head>
  <body>
    <header>
      <?php
        if (!isset($_GET['shop'])) echo '<a href=?shop>Shop</a>';
        else echo '<a href=?home>Home</a>';
      ?>
    </header>
    <main>
      <?php
        if (!isset($_GET['shop']))
          if (!isset($_POST['name'])) {
            echo "<form method=post>
              <input placeholder=Name name=name>
              <button>Submit</button>
            </form>";
          } else echo '<h1>Welcome, ' . $_POST['name'] . ', to the WSERS1 shop</h1>';
        else {
          $articles = [['Desktop', 599.99], ['Mobile', 299.99], ['USB cam', 19.99]];

          function get_table($arr) {
            $table =
'<table><tr><th>Article</th><th>Price</th><th>Quantity</th></tr>';
            foreach ($arr as $article)
              $table .= "<tr><td>{$article[0]}</td><td>{$article[1]}</td><td>" .
rand(1, 10)
              . "</td></tr>";
            return $table . '</table>';
          }

          echo get_table($articles);
        }
      ?>
    </main>
  </body>
</html>

```

[Home](#)

First name	Last name	City
Bill	Gates	Microsoft
Linus	Torvalds	Linux
Donald	Duck	Entenhausen
Mickey	Mouse	Disneyland

The sum between 12 and 222 is 24687

Develop the **single file** app exactly as shown at [youtu.be/VWQTI6QXCk0](https://youtu.be/VWQTI6QXCk0) using only HTML, CSS and PHP and taking the following into account:

1. The header shows a link to the address book if the latter is not displayed.
2. The header shows a link to the calculator if the latter is not displayed.
3. The header shows a link to the home content if the latter is not displayed.
4. The address book and the calculator can be displayed at the same time, in which case the address book is always displayed first.
5. The date in the home content is shown using Luxembourg as timezone.
6. The address table is generated from a two-dimensional array.
7. The calculator calculates the sum of all integers from [a, b] using a function that receives a and b as parameters and returns the sum.

## Operating Systems

### Welcome admin

Windows	7	8	8.1	10
Linux	Ubuntu	openSUSE	Mint	

Current date: 21-03-2015 09:49:08

Write a **single** PHP file that performs exactly as shown at [youtu.be/Dn5er8tfSKQ](https://youtu.be/Dn5er8tfSKQ) using only HTML, CSS and PHP.

Your HTML should be valid. The border color is greenyellow.

The script recognizes two user/password combinations: dummy1/d1pw and admin/admin. Any other combination returns to the login display.

Your PHP script defines and uses a function `get_table`, which takes a two-dimensional associative array as parameter. It returns the HTML table as a string.

For the admin user, the current date and time is shown below the table.

## Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/OperatingSystems](https://students.btsi.lu/evegi144/WAD/PHP/Tests/OperatingSystems)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>Operating Systems</title>
    <meta charset=utf-8>
    <style>
      table, th, td {
        border:      2px greenyellow double;
        text-align: center;
      }
    </style>
  </head>
  <body>
    <main>
      <?php
        $user = false;
        if (isset($_POST['user_name'], $_POST['password'])) {
          if ($_POST['user_name'] === 'dummy1' && $_POST['password'] === 'd1pw')
            $user = 'dummy1';
          elseif ($_POST['user_name'] === 'admin' && $_POST['password'] === 'admin')
            $user = 'admin';
        }
        if ($user) {
          function get_table($OSs) {
            $output = "<table>";
            foreach ($OSs as $OS => $versions) {
              $output .= "<tr><th>$OS</th>";
              foreach ($versions as $version) $output .= "<td>$version</td>";
              $output .= '</tr>';
            }
            return $output . '</table>';
          }

          $OSs = ['Windows' => ['7', '8', '8.1', '10'],
                  'Linux'   => ['Ubuntu', 'openSUSE', 'Mint']];
          echo "<h1>Welcome $user</h1>" . get_table($OSs);
          if ($user === 'admin') {
            date_default_timezone_set('Europe/Luxembourg');
            echo '<br>Current date: ' . date('d-m-Y H:i:s');
          }
        }
        else {
          ?>
          <form method=post>
            <input placeholder='user name' name=user_name required>
            <input type=password placeholder=password name=password required>
            <button>Log in</button>
```

```
</form>
<?php
}
?>
</main>
</body>
</html>
```

## Quiz

### Question

The capital of China is:

Write a **single** PHP file that performs exactly as shown at [youtu.be/eMXcLq6mdHM](https://youtu.be/eMXcLq6mdHM) using only HTML, CSS and PHP.

Your HTML should be valid.

Your PHP script defines and uses the following:

1. 

```
$questions = [['The capital of China is: ', 0], ['The value of 5^2 is:', 4]];
```

The first element is the question, the second the index of the correct answer.

2. 

```
$answers = [['Beijing', 'Nanjing', 'Luoyang', "Chang'an"], ['19', '23', '45', '67', '25']];
```

These are all answers shown in the dropdown list.

3. Function `checkAnswer($idx, $answer)`, which returns true if the correct answer for the question with index \$idx is \$answer, otherwise false.

### Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/Quiz](https://students.btsi.lu/evegi144/WAD/PHP/Tests/Quiz)

```
<!DOCTYPE html>
<html lang="en">
  <head>
    <title>Quiz</title>
    <meta charset="UTF-8">
  </head>
  <body>
    <?php
```

```

$questions = [['The capital of China is: ', 0], ['The value of 5^2 is:', 4]];
$answers = [['Beijing', 'Nanjing', 'Luoyang', "Chang'an"],
['19', '23', '45', '67', '25']];
if (isset($_POST['questions_answered'], $_POST['correct_answers'],
$_POST['answer'])) {
    $questions_answered = $_POST['questions_answered'];
    $correct_answers = $_POST['correct_answers'];
    $answer = $_POST['answer'];

    function checkAnswer($idx, $answer) {
        global $answers, $questions;
        if ($answers[$idx][$questions[$idx][1]] === $answer) return true;
        return false;
    }

    if (checkAnswer($questions_answered, $answer)) $correct_answers++;
} else {
    $questions_answered = -1;
    $correct_answers = 0;
}

$questions_answered++;
if ($questions_answered === 2)
    echo <<<EOT
        <h3>You scored $correct_answers out of $questions_answered.</h3>
        <form>
            <button>New game</button>
        </form>
EOT;
else {
    echo <<<EOT
        <h3>Question </h3>
        <form method=post>
            <input type=hidden name=questions_answered value="$questions_answered">
            <input type=hidden name=correct_answers value="$correct_answers">
            {$questions[$questions_answered][0]}
            <select name=answer>
EOT;
        for ($i = 0; $i < count($answers[$questions_answered]); $i++)
            echo '<option value="' . $answers[$questions_answered][$i] . '">' .
                "{$answers[$questions_answered][$i]}</option>";
        echo "</select><button>Send</button></form>";
    }
    ?>
</body>
</html>

```

## Login

Create the app exactly as shown in [youtu.be/lkEaFVX9NEo](https://youtu.be/lkEaFVX9NEo).

1. Create a file `functions.php`. In this file:
  - a. Write the PHP function `check_login`, which takes two parameters, the user name and the password entered by the user (valid user name/password combinations are a1/a1 and admin/admin). In this function:
    - i. If the two parameters are not defined (for instance, the function was called with only one or without any parameters), return 0;
    - ii. Define the array `$valid_logins`, which contains two arrays, the first one with the values 'a1', 'a1' and 1, the second one with the values 'admin', 'admin' and 2.
    - iii. Run through `$valid_logins` and check whether the given user name and password correspond to the first two elements of one of the arrays. If this is the case, return the third value of the matching array (i.e. 1 or 2).
    - iv. If no matching login was found, return 0.
  - b. Write the PHP function `generate_welcome`, which takes one parameter, a number. In this function:
    - i. If the parameter is not defined or its value is less than 1 or greater than 2 return "error".
    - ii. Define the associative array `$outputs`, which contains 2 elements: the first one has key 1 and value "Welcome user a1", the second one has key 2 and as value the HTML code to generate a heading with background color gold and the text "Welcome master of the universe!".
    - iii. Select the output corresponding to the parameter value from `$outputs` and return it.
2. Create the file `index.php`:
  - a. The title is "Test 1: Login".
  - b. The generated HTML document must obviously be valid HTML5.
  - c. Include the functions file.
  - d. Create, but not necessarily display, a form with two inputs, one for the user name and one for the password, as well as a login button.
  - e. Check whether the login button has been pressed and the values in the two inputs correspond to a valid login (you must use function `check_login` for this purpose).
  - f. If the check is positive, send the result of `generate_welcome` called with the result from `check_login` to the browser. Then send a new line with a link to log out.
  - g. If the previous check is not positive, display the login form created in point 4.

## Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/Login](https://students.btsi.lu/evegi144/WAD/PHP/Tests/Login)

`functions.php`

```

<?php
function check_login($un, $pw) {
    if (!isset($un, $pw)) return 0;
    $valid_logins = [['a1', 'a1', 1], ['admin', 'admin', 2]];
    foreach ($valid_logins as $valid_login)
        if ($un === $valid_login[0] && $pw === $valid_login[1]) return $valid_login[2];
    return 0;
}

function generate_welcome($num) {
    if (!isset($num) || $num < 1 || $num > 2) return 'error';
    $outputs = [1 => "Welcome user a1",
        2 => "<h1 style='background-color: gold'>Welcome master of the
universe!</h1>"];
    return $outputs[$num];
}

?>

```

## index.php

```

<!DOCTYPE html>
<html lang=en>
<head>
    <meta charset=UTF-8>
    <title>Login</title>
</head>
<body>
    <?php
        require_once 'functions.php';
        if (isset($_POST['login']) && check_login($_POST['un'], $_POST['pw']) > 0) {
            echo generate_welcome(check_login($_POST['un'], $_POST['pw']));
            echo '<br><a href="">Logout</a>';
        } else echo "<form method=post><input name=un placeholder='user name' required>"
        .
            "<input type=password name=pw placeholder=password required>" .
            "<button name=login>Login</button></form>";
    ?>
</body>
</html>

```

## index.php

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Gradiator</title>
  </head>
  <body>
    <form method=post>
      <input type=radio name=action value=radial checked>Radial gradient<br>
      <input type=radio name=action value=linear>Linear gradient<br>
      Number of stops: <input type=range name=stops min=2 max=10 value=6 step=1><br>
      <button name=submit>Submit</button>
    </form>
    <?php
      require_once 'function.php';
      if (isset($_POST['submit'])) echo get_gradiant_JS($_POST['action'],
$_POST['stops']);
    ?>
  </body>
</html>

```

## Computer Shop

Create the single file app exactly as shown in [youtu.be/dtIWHlG8joU](https://youtu.be/dtIWHlG8joU).

1. The title is "Computer Shop".
2. The generated HTML document must obviously be valid HTML5.
3. If the user has selected something from the drop down list, the text "Congratulations, you bought " followed by the the article name followed by a "." followed, in the next line, by the "Return" button are displayed.
4. Otherwise, the form with the selection drop down list is displayed. To do this:
  - a. Create one 2-dimensional array with the three keys "PC", "Laptop" and "Tablet" and the associated values "Mega PC1" and "Mega PC2" for the first key, "Mega Lap1" and "Mega Lap2" for the second as well as "Mega Tab1" and "Mega Tab2" for the third.
  - b. The keys are used to generate the option groups. The values are used as the option text.
  - c. Do not forget the "Buy" button.

## Solution

[students.btsi.lu/evegi144/WAD/PHP/Tests/ComputerShop](https://students.btsi.lu/evegi144/WAD/PHP/Tests/ComputerShop)

```

<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8">
    <title>Computer Shop</title>
  </head>
  <body>
    <?php
      if (isset($_POST['selection'])) {
        echo "Congratulations, you bought {$_POST['selection']}. ";
        echo "<br><a href=>Return</a>";
      } else {
        echo "<form method=post><select name=selection>";
        $opts = ['PC' => ['Mega PC1', 'Mega PC2'],
                  'Laptop' => ['Mega Lap1', 'Mega Lap2'],
                  'Tablet' => ['Mega Tab1', 'Mega Tab2']];
        foreach ($opts as $opt_key => $opt_values) {
          echo "<optgroup label=$opt_key>";
          foreach ($opt_values as $opt) echo "<option>$opt</option>";
          echo "</optgroup>";
        }
        echo "</select><button>Buy</button></form>";
      }
    ?>
  </body>
</html>

```

## Simple Calculator

In this test, you will create the single file app exactly as shown in [www.youtube.com/watch?v=zTiS1Rnh9is](https://www.youtube.com/watch?v=zTiS1Rnh9is).

1. The title is "Simple Calculator".
2. The generated HTML document must obviously be valid HTML5 and your PHP script may not generate any messages in your PHP log file.
3. Create the following PHP array:

```

$arr = ['bg-color' => ['black', 'green', 'blue', 'red'],
        'color' => ['white', 'brown', 'cyan', 'maroon']];

```

4. Set the document background and font colors by selecting randomly from the corresponding arrays in `$arr`.
5. Create the PHP function `calc($x, $y, $op)`, which does the following:
  - a. Check if `$x` and `$y` are numbers and if `$op` is one of '+', '-', '\*' or '/'. If not, return the empty string.
  - b. Return the result of `$x $op $y` (use the `eval` function and check out the example in the book).

Example, if  $x$  is 3,  $y$  is 4 and  $op$  is '-', the function returns -1.

6. The two inputs in the form allow only numbers to be entered and require input.
7. If the form has been submitted, the text 'The result of ' followed by  $x$   $op$   $y$  followed by ' is ' followed by the result from the calc function for the submitted parameters is displayed above the form.

### **Solution**

[students.btsi.lu/evegi144/WAD/PHP/Tests/SimpleCalculator](https://students.btsi.lu/evegi144/WAD/PHP/Tests/SimpleCalculator)

```

<?php
    $arr = ['bg-color' => ['black', 'green', 'blue', 'red'],
            'color'     => ['white', 'brown', 'cyan', 'maroon']];
?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <meta charset=UTF-8>
        <title>Simple Calculator</title>
        <style>
            body {
                background-color: <?php echo $arr['bg-color'][rand(0, 3)]?>;
                color: <?php echo $arr['color'][rand(0, 3)]?>;
            }
        </style>
    </head>
    <body>
        <?php
            //echo '<script>document.body.style.backgroundColor="' . $arr['bg-
color'][rand(0, 3)] .
            //  "'>document.body.style.color="' . $arr['color'][rand(0, 3)] . "'</script>';

            function calc($x, $y, $op) {
                if (!is_numeric($x) || !is_numeric($y) || !in_array($op, ['+', '-', '*',
'/' ]))
                    return '';
                return eval('return ' . $x . $op . $y . ';');
            }

            if (isset($_POST['x'])) echo "The result of " . $_POST['x'] . " " . $_POST['op']
.
            " " . $_POST['y'] . " is " . calc($_POST['x'], $_POST['y'], $_POST['op']);
?>
<form method=post>
    x: <input type=number name=x title=x required>
    <select name=op title=operator>
        <option value=++>+</option>
        <option value=->-</option>
        <option value=*>*</option>
        <option value=/'>/'</option>
    </select>
    y: <input type=number name=y title=y required>
    <button>Calc</button>
</form>
</body>
</html>

```

## 5.3. MySQL

### 5.3.1. When and where do we need a database?

Before we take a closer look at MySQL we should understand when and where we need a database. Study [opentextbc.ca/dbdesign01/chapter/chapter-1-before-the-advent-of-database-systems](https://opentextbc.ca/dbdesign01/chapter/chapter-1-before-the-advent-of-database-systems) and [opentextbc.ca/dbdesign01/chapter/chapter-3-characteristics-and-benefits-of-a-database](https://opentextbc.ca/dbdesign01/chapter/chapter-3-characteristics-and-benefits-of-a-database).

### 5.3.2. Structured data

By structured data we mean data that has a defined format and maximum size. Traditionally these have been for instance numbers, dates or strings. Increasingly we see machine-generated structured data as described in [www.dummies.com/programming/big-data/engineering/structured-data-in-a-big-data-environment](https://www.dummies.com/programming/big-data/engineering/structured-data-in-a-big-data-environment).

#### Personal data

From [gdpr-info.eu/art-4-gdpr](https://gdpr-info.eu/art-4-gdpr):

'Personal data' means any information relating to an identified or identifiable natural person ('data subject'); an identifiable natural person is one who can be identified, directly or indirectly, in particular by reference to an identifier such as a name, an identification number, location data, an online identifier or to one or more factors specific to the physical, physiological, genetic, mental, economic, cultural or social identity of that natural person;

### 5.3.3. Legal base for storing data

The main legal base for storing data in the EU is the [General Data Protection Regulation \(GDPR\)](#) which is explained by [Wikipedia](#). The [European Commission](#) provides another great resource. In Luxembourg, the [Commission Nationale pour la Protection des Données \(CNPD\)](#) provides detailed information.

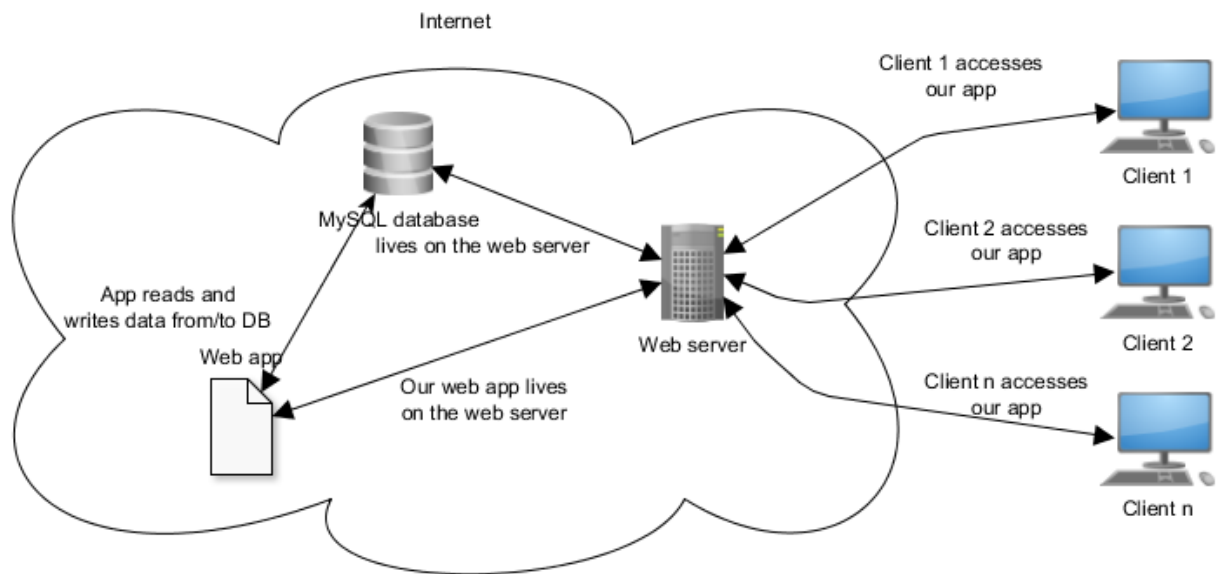


You are obliged to inform your visitors about their privacy rights. To do this, link them to [www.knowyourprivacyrights.org](https://www.knowyourprivacyrights.org).

### 5.3.4. Introduction to MySQL

According to [www.mysql.com/products](https://www.mysql.com/products), MySQL is the world's most popular open source database. We will use it to store our web app's data.

If we add a database to the picture we discussed in section [Installing and configuring the tools](#), we get the following:



MySQL is a program that runs on the server and specializes in storing, retrieving and manipulating large amounts of data in a structured and highly efficient way.

SQL stands for Structured Query Language. It is the standard language that has been used for decades to manage databases.



Two very good online SQL learning resources can be found at [www.tutorialspoint.com/sql/sql-overview.htm](http://www.tutorialspoint.com/sql/sql-overview.htm) and [sqlzoo.net](http://sqlzoo.net).



Remember to use some of the excellent cheat sheets on the web, for instance [www.cheatography.com/guslong/cheat-sheets/essential-mysql](http://www.cheatography.com/guslong/cheat-sheets/essential-mysql) or [www.cheatography.com/davechild/cheat-sheets/mysql](http://www.cheatography.com/davechild/cheat-sheets/mysql).

Data is organized in databases. Each database consists of one or more tables. A table contains data for a specific purpose, for instance addresses. A table is organized in rows and columns. Each column represents a specific data field, e.g. last name. Rows contain data. If we have 10 addresses stored in our address table, there'll be 10 rows, each row containing the address of one person. For each row we can access a specific field, e.g. street.

Here is a sample table for the top 10 clubs in the German Bundesliga (not up to date!):

Rank	Club	Points
1	Bayern München	23
2	Dortmund	22
3	Leverkusen	22
4	Berlin	15
5	Schalke 04	14
6	M'Gladbach	13
7	Hannover 96	13

Rank	Club	Points
8	Bremen	12
9	Stuttgart	11
10	Hoffenheim	10

### 5.3.5. Data types

For each column, we need to specify the type of data we want to store. For instance, in the sample table above, the first and third columns need to be numbers, whereas the second is used for text. A detailed overview of the data types available in MySQL can be found at [dev.mysql.com/doc/refman/5.7/en/data-types.html](https://dev.mysql.com/doc/refman/5.7/en/data-types.html).

The main ones that we'll use are listed in the following table:

Data type	Meaning	Example
<b>VARCHAR</b>	string up to specified length (max. 65535)	<b>VARCHAR(50)</b>
<b>INT</b>	integer	<b>INT</b>
<b>DECIMAL</b>	decimal with given precision	<b>DECIMAL(5, 2)</b>
<b>TIMESTAMP</b>	date and time	<b>TIMESTAMP</b>
<b>DATETIME</b>	date and time	<b>DATETIME</b>
<b>DATE</b>	date	<b>DATE</b>
<b>TEXT</b>	string up to different sizes	<b>TEXT</b>
<b>BLOB</b>	binary large object	<b>BLOB</b>
<b>ENUM</b>	string from a given list	<b>ENUM('Donald Duck', 'Asterix', 'Mickey Mouse')</b>
<b>SET</b>	string object with zero or more values from a list of permitted values	<b>SET('BMW', 'Nissan')</b>
<b>BOOL</b>	boolean	<b>BOOL</b>
<b>JSON</b>	JSON	<b>JSON</b>

From [dev.mysql.com/doc/refman/5.7/en/string-literals.html](https://dev.mysql.com/doc/refman/5.7/en/string-literals.html):

A string is a sequence of bytes or characters, enclosed within either single quote (') or double quote (") characters. Examples:

```
'a string'
```

```
"another string"
```

Quoted strings placed next to each other are concatenated to a single string. The following lines are equivalent:

```
'a string'
```

```
'a' ' ' 'string'
```

`TIMESTAMP` and `DATETIME` values are in the format '`YYYY-MM-DD HH:MM:SS`', for instance '`2014-08-11 15:04:00`'. The supported range is '`1970-01-01 00:00:01.000000`' to '`2038-01-19 03:14:07.999999`' for `TIMESTAMP` and '`1000-01-01 00:00:00`' to '`9999-12-31 23:59:59`' for `DATETIME`. For a `TIMESTAMP` and (starting from MySQL 5.6.5) `DATETIME` column we can specify `DEFAULT CURRENT_TIMESTAMP`, in which case the current timestamp is automatically set for inserted rows that specify no value for the column. We can also set `ON UPDATE CURRENT_TIMESTAMP` to have the date and time automatically updated when the value of any other column in the row is changed (cf. [dev.mysql.com/doc/refman/5.7/en/timestamp-initialization.html](http://dev.mysql.com/doc/refman/5.7/en/timestamp-initialization.html)).

When inserting `SET` values into a table, there should be no spaces between values, i.e. use '`val1,val2,val3`' instead of '`val1, val2, val3`'.

### 5.3.6. Case sensitivity

SQL keywords, function names, column and index names are not case sensitive. Database, table and view names however are represented using directories and files in the underlying filesystem. Whether they are case sensitive or not depends therefore on the operating system.

### 5.3.7. Comments

MySQL supports three comment styles which are very well illustrated at [dev.mysql.com/doc/refman/5.7/en/comments.html](http://dev.mysql.com/doc/refman/5.7/en/comments.html).

[students.btsi.lu/evegi144/WAD/MySQL/comments1.sql](http://students.btsi.lu/evegi144/WAD/MySQL/comments1.sql)

```
# This comment goes until the end of the line.
-- So does this comment.
SELECT dtFirstName, dtLastName /* inline comment */ FROM tblT1;
/* Multiline
comment
*/
```

### 5.3.8. Naming conventions

Item	Example(s)
Database	dbShop
Table	tblUser, tblLoggedInUser
Primary key	idUser
Data field	dtName
Foreign key	fiUser referencing tblUser.idUser

Table names start with the prefix **tbl** and are always singular, i.e. **tblUsers** should be avoided.

### 5.3.9. Character sets and collations

From [dev.mysql.com/doc/refman/5.7/en/charset-general.html](https://dev.mysql.com/doc/refman/5.7/en/charset-general.html):

A character set is a set of symbols and encodings. A collation is a set of rules for comparing characters in a character set. Let's make the distinction clear with an example of an imaginary character set.

Suppose that we have an alphabet with four letters: A, B, a, b. We give each letter a number: A = 0, B = 1, a = 2, b = 3. The letter A is a symbol, the number 0 is the encoding for A, and the combination of all four letters and their encodings is a character set.

Suppose that we want to compare two string values, A and B. The simplest way to do this is to look at the encodings: 0 for A and 1 for B. Because 0 is less than 1, we say A is less than B. What we've just done is apply a collation to our character set. The collation is a set of rules (only one rule in this case): "compare the encodings." We call this simplest of all possible collations a binary collation.

But what if we want to say that the lowercase and uppercase letters are equivalent? Then we would have at least two rules: (1) treat the lowercase letters a and b as equivalent to A and B; (2) then compare the encodings. We call this a case-insensitive collation. It is a little more complex than a binary collation.

In real life, most character sets have many characters: not just A and B but whole alphabets, sometimes multiple alphabets or eastern writing systems with thousands of characters, along with many special symbols and punctuation marks. Also in real life, most collations have many rules, not just for whether to distinguish lettercase, but also for whether to distinguish accents (an "accent" is a mark attached to a character as in German Ö), and for multiple-character mappings (such as the rule that Ö = OE in one of the two German collations).

For our purposes we always use `utf8mb4` as character set and `utf8mb4_bin` as collation type (cf. [www.eversql.com/mysql-utf8-vs-utf8mb4-whats-the-difference-between-utf8-and-utf8mb4](http://www.eversql.com/mysql-utf8-vs-utf8mb4-whats-the-difference-between-utf8-and-utf8mb4)).



There are default settings for character sets and collations at four levels: server, database, table, and column. Usually we do not specify the character set and collation at column level, which means that the table settings will be used.

### 5.3.10. DB Diagram design tools

[www.holistics.io/blog/top-5-free-database-diagram-design-tools](http://www.holistics.io/blog/top-5-free-database-diagram-design-tools)

<a href="http://www.dbdesigner.net">www.dbdesigner.net</a>
<a href="http://dbdiffo.com">dbdiffo.com</a>
<a href="http://www.oracle.com/database/technologies/appdev/datamodeler.html">www.oracle.com/database/technologies/appdev/datamodeler.html</a>

### 5.3.11. Creating and dropping databases and tables

The detailed MySQL syntax to create and drop a database and a table can be found here:

- [dev.mysql.com/doc/refman/8.0/en/create-user.html](http://dev.mysql.com/doc/refman/8.0/en/create-user.html)
- [dev.mysql.com/doc/refman/8.0/en/create-database.html](http://dev.mysql.com/doc/refman/8.0/en/create-database.html)
- [dev.mysql.com/doc/refman/8.0/en/drop-database.html](http://dev.mysql.com/doc/refman/8.0/en/drop-database.html)
- [dev.mysql.com/doc/refman/8.0/en/create-table.html](http://dev.mysql.com/doc/refman/8.0/en/create-table.html)
- [dev.mysql.com/doc/refman/8.0/en/drop-table.html](http://dev.mysql.com/doc/refman/8.0/en/drop-table.html)

Here is a simple application example:

[students.btsi.lu/evegi144/WAD/MySQL/createDB1.sql](http://students.btsi.lu/evegi144/WAD/MySQL/createDB1.sql)

```

# Drop database evegi144_dbDemo1, if it already exists.
DROP DATABASE IF EXISTS evegi144_dbDemo1;

# Create new DB named evegi144_dbDemo1.
CREATE DATABASE evegi144_dbDemo1
  # Avoid problems with é, ö etc.
  DEFAULT CHARSET utf8mb4
  # Use case sensitivity.
  DEFAULT COLLATE utf8mb4_bin;

# If we want to create a user
CREATE USER evegi144_dbDemo1@localhost IDENTIFIED BY 'xxx';
# If you want to change the password validation configuration, see
# https://stackoverflow.com/questions/43094726/your-password-does-not-satisfy-the-
current-policy-requirements

# If we want to give another user access to the DB.
GRANT ALL PRIVILEGES ON evegi144_dbDemo1.* TO 'evegi144'@'localhost';
FLUSH PRIVILEGES;

# First select the DB that we want to use.
USE evegi144_dbDemo1;

# Drop table tblT1 if it already exists.
DROP TABLE IF EXISTS tblT1;

# Create table tblT1 with 8 columns.
CREATE TABLE tblT1 (
  idT1          INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
  dtFirstName   VARCHAR(20) NOT NULL,
  dtLastName    VARCHAR(20) NOT NULL,
  dtBirthDate   DATE,
  dtCreationTimestamp TIMESTAMP          DEFAULT CURRENT_TIMESTAMP,
  dtAverage     DECIMAL(4, 2),
  dtNumberOfGrades INT UNSIGNED,
  dtModules     SET('HTSTA', 'CLISS1', 'CLISS2', 'WSERS1', 'WSERS2', 'WEBAP1',
                    'WEBAP2')
)
ENGINE = INNODB # We use INNODB as the DB engine.
DEFAULT CHARSET utf8mb4 # Same as above
DEFAULT COLLATE utf8mb4_bin;

# Insert a row of data into the table.
INSERT INTO tblT1 (dtFirstName, dtLastName, dtBirthDate, dtAverage, dtNumberOfGrades,
                  dtModules)
VALUES
  ('Mickey', 'Mouse', '1927-07-03', 23.45, 15, 'HTSTA,CLISS1,CLISS2'),
  ('Donald', 'Duck', '1949-02-04', 33.55, 12, 'WSERS1,WSERS2');

```

We use the INNODB storage engine. From [dev.mysql.com/doc/refman/5.7/en/storage-engines.html](https://dev.mysql.com/doc/refman/5.7/en/storage-engines.html):

Storage engines are MySQL components that handle the SQL operations for different table types. InnoDB is the most general-purpose storage engine, and Oracle recommends using it for tables except for specialized use cases.

We set the default character set to be UTF8 in order to avoid any problems with storing é, ö etc. We also set collation to `utf8_bin` in order to have case sensitivity. If we skip this, the DB will treat the strings `home`, `Home` and `HOME` as identical.



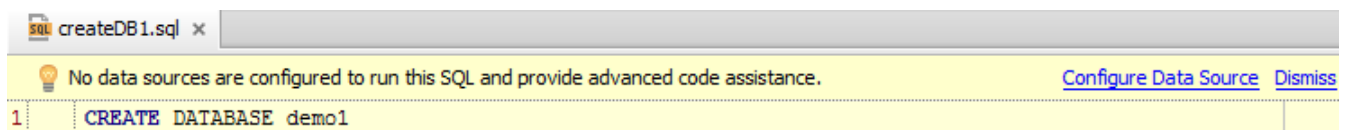
In a production environment, you should create one or several specific DB users for your app that have only the minimum privileges required. Given that you cannot do this on Foxi, we won't go into detail here, but take a look at [dev.mysql.com/doc/refman/5.7/en/create-user.html](https://dev.mysql.com/doc/refman/5.7/en/create-user.html) and [dev.mysql.com/doc/refman/5.7/en/grant.html](https://dev.mysql.com/doc/refman/5.7/en/grant.html).

There are several ways to create and work with a database.

### Via PhpStorm

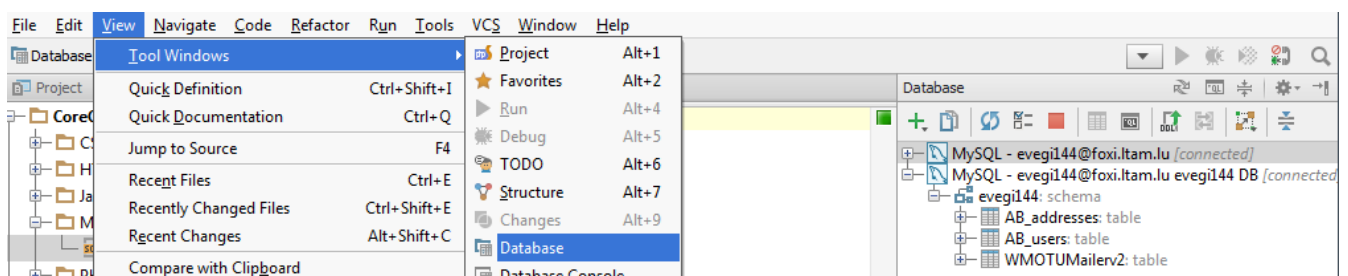
The recommended approach is to write a SQL file that creates the DB using SQL statements. This allows us to easily review the database structure that we have created, make modifications and recreate the database.

If we use PhpStorm, we can configure a data source when we create a new file with extension `.sql`:

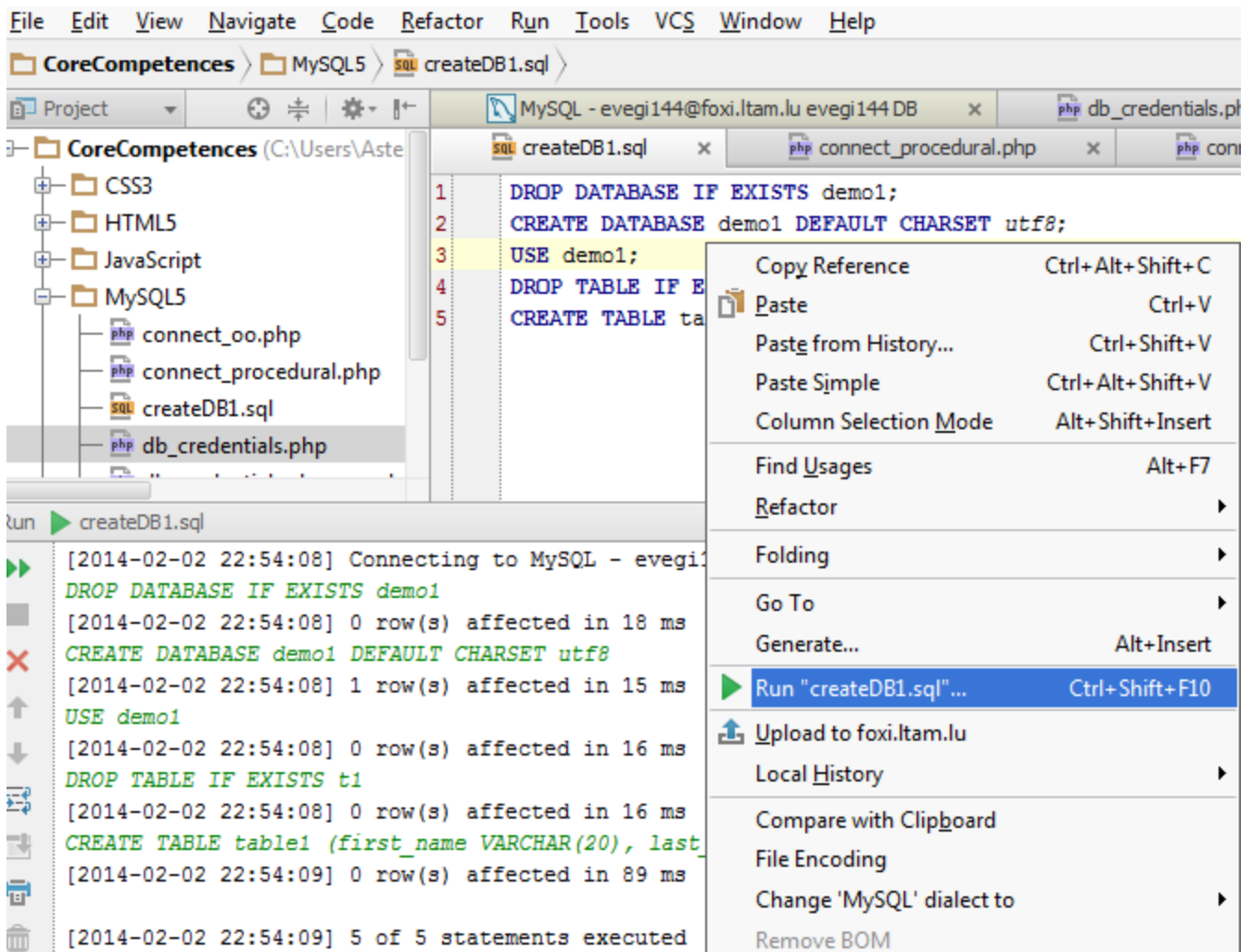


To configure the PhpStorm DB connection, see [PhpStorm DB setup](#).

If you cannot see the database window, you can open it like this:

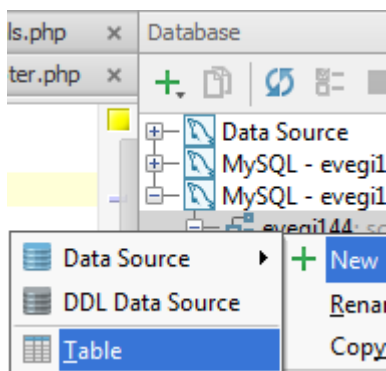


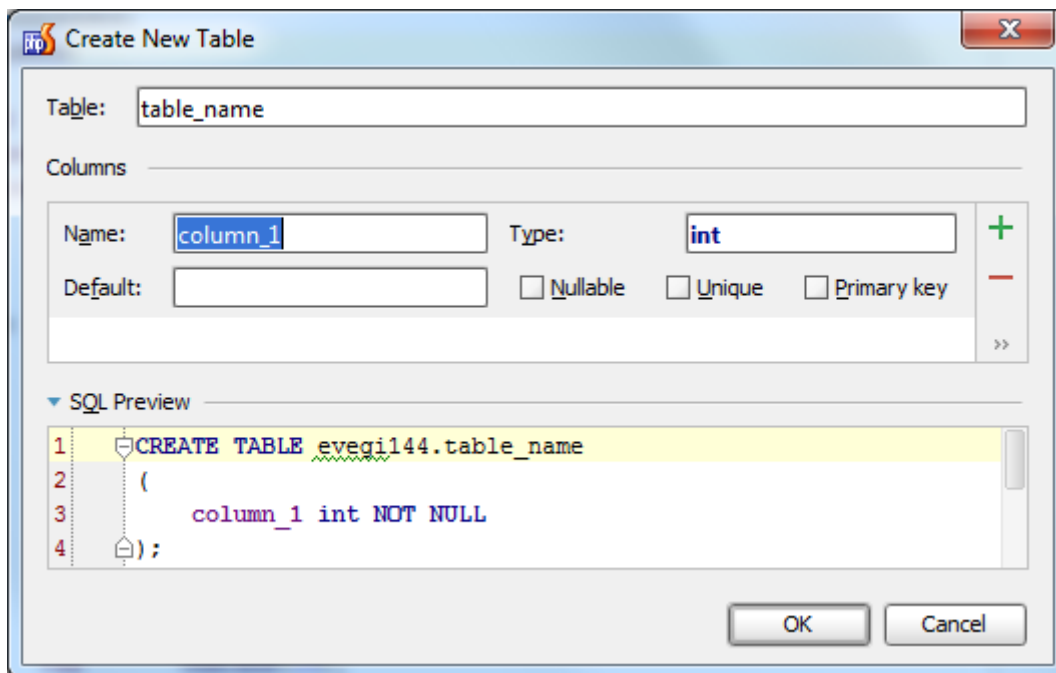
We can run our SQL script to create the DB and tables:



Don't forget to click  to update the DB display.

We can also add a table using the PhpStorm wizard:





### Via the MySQL command line

If you have the option to log into your web server (as you do with Foxi) you can create and manage your database via the MySQL command line. Before you do so, it is important to set the system variables with respect to the character set and collation correctly, otherwise you are asking for trouble. Here's why:

```
mysql> SHOW VARIABLES LIKE 'character_set%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| character_set_client | latin1 |
| character_set_connection | latin1 |
| character_set_database | latin1 |
| character_set_filesystem | binary |
| character_set_results | latin1 |
| character_set_server | latin1 |
| character_set_system | utf8 |
| character_sets_dir | /usr/share/mysql/charsets/ |
+-----+-----+
8 rows in set (0.00 sec)

mysql> SHOW VARIABLES LIKE 'collation%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| collation_connection | latin1_swedish_ci |
| collation_database | latin1_swedish_ci |
| collation_server | latin1_swedish_ci |
+-----+-----+
3 rows in set (0.00 sec)
```

As system administrator you could set these settings in the MySQL configuration file, like so (cf. [stackoverflow.com/questions/3513773/change-mysql-default-character-set-to-utf-8](https://stackoverflow.com/questions/3513773/change-mysql-default-character-set-to-utf-8)):

```
[mysqld]
collation-server = utf8mb4_bin
init-connect='SET NAMES utf8mb4'
character-set-server = utf8mb4
```

But often you won't have admin rights on the host machine, so you'll need to run the following script after logging in to MySQL and before creating or manipulating any structures or data (cf. [dev.mysql.com/doc/refman/5.7/en/server-system-variables.html](https://dev.mysql.com/doc/refman/5.7/en/server-system-variables.html) and [dev.mysql.com/doc/refman/5.7/en/using-system-variables.html](https://dev.mysql.com/doc/refman/5.7/en/using-system-variables.html)):

[students.btsi.lu/evegi144/WAD/MySQL/mysql\\_conf.sql](https://students.btsi.lu/evegi144/WAD/MySQL/mysql_conf.sql)

```
SET character_set_client=utf8mb4;
SET character_set_connection=utf8mb4;
SET character_set_database=utf8mb4;
SET character_set_results=utf8mb4;
SET character_set_server=utf8mb4;
SET collation_connection=utf8mb4_bin;
SET collation_database=utf8mb4_bin;
SET collation_server=utf8mb4_bin;
```

```
mysql> source mysql_conf.sql
Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

Query OK, 0 rows affected (0.00 sec)

mysql> SHOW VARIABLES LIKE 'character_set%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| character_set_client | utf8 |
| character_set_connection | utf8 |
| character_set_database | utf8 |
| character_set_filesystem | binary |
| character_set_results | utf8 |
| character_set_server | utf8 |
| character_set_system | utf8 |
| character_sets_dir | /usr/share/mysql/charsets/ |
+-----+-----+
8 rows in set (0.00 sec)

mysql> SHOW VARIABLES LIKE 'collation%';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| collation_connection | utf8_bin |
| collation_database | utf8_bin |
| collation_server | utf8_bin |
+-----+-----+
3 rows in set (0.00 sec)
```

Now you can work on your structures and data. Here is an example:

```
evegi144@web:~$ mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 2479
Server version: 5.1.49-3 (Debian)

Copyright (c) 2000, 2010, Oracle and/or its affiliates. All rights reserved.
This software comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to modify and redistribute it under the GPL v2 license

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE demo1;
Query OK, 1 row affected (0.00 sec)

mysql> USE demo1;
Database changed
mysql> CREATE TABLE t1(first_name VARCHAR(30), last_name VARCHAR(30));
Query OK, 0 rows affected (0.10 sec)

mysql> DESC t1;
+-----+-----+-----+-----+-----+-----+
| Field      | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| first_name | varchar(30)   | YES  |     | NULL    |       |
| last_name  | varchar(30)   | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

If you have created a SQL file you can then simply execute it:

```
evegi144@web:~$ cd T2IF2_WSERS/Tutorial/CoreCompetences/MySQL5/
evegi144@web:~/T2IF2_WSERS/Tutorial/CoreCompetences/MySQL5$ mysql -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 751
Server version: 5.1.49-3 (Debian)

Copyright (c) 2000, 2010, Oracle and/or its affiliates. All rights reserved.
This software comes with ABSOLUTELY NO WARRANTY. This is free software,
and you are welcome to modify and redistribute it under the GPL v2 license

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> source createDB1.sql
Query OK, 0 rows affected (0.00 sec)

Query OK, 1 row affected (0.00 sec)
```

Note that before you can work with a database, you need to select it as the active database using the **use** command.

**Via phpMyAdmin**

127.0.0.1

Databases

SQL

Status

Processes

Privileges

## Databases

Create new database

Collation



Create

127.0.0.1

Databases

SQL

Status

Processes

Privileges

Export

Import

## Importing into the current server

### File to Import:

File may be compressed (gzip, bzip2, zip) or uncompressed.

A compressed file's name must end in `.[format].[compression]`. Example: `.sql.zip`

Browse your computer:  No file selected. (Max: 10MiB)

Character set of the file:

### Partial Import:

☒ Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. *(This might be break transactions.)*

Number of rows to skip, starting from the first row:

### Format:

### Format-Specific Options:

SQL compatibility mode:

☒ Do not use `AUTO_INCREMENT` for zero values

Go

127.0.0.1

Databases SQL Status Processes Privileges Export Import

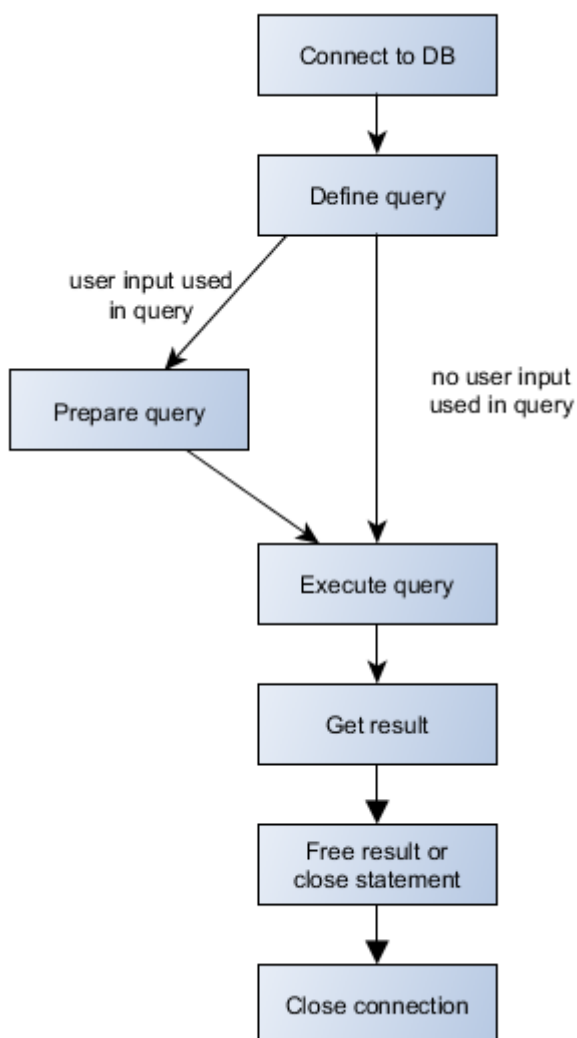
✓ Import has been successfully finished, 2 queries executed. (createDB1.sql)

```

DROP DATABASE IF EXISTS demo1; # MySQL returned an empty result set (i.e. zero rows).
CREATE DATABASE demo1 DEFAULT CHARSET utf8; # 1 row affected.

```

### 5.3.12. Work flow in PHP



### 5.3.13. Connecting to and disconnecting from a database via PHP

In [Installing and configuring the tools](#) we saw how to install and configure MySQL. Now we'll see how we can connect to an existing database (DB) via PHP (cf. [php.net/manual/en/mysqli.construct.php](http://php.net/manual/en/mysqli.construct.php)). We use the MySQL Improved Extension (mysqli) to access a MySQL DB. A great summary of the MySQLi extension can be found at [php.net/manual/en/mysqli.summary.php](http://php.net/manual/en/mysqli.summary.php).

First we define the constants that we need to connect to the database and put them into a file that we can include:

[students.btsi.lu/evegi144/WAD/MySQL/db\\_credentials\\_dummy.php](https://students.btsi.lu/evegi144/WAD/MySQL/db_credentials_dummy.php)

```
<?php
const DB_HOST = 'localhost';
const DB_USER = 'your_user_name';
const DB_PW   = 'your_password';
const DB_NAME = 'your_database';
?>
```

Note that this is safe, given that PHP code gets executed on the server and in this case only an empty document is sent to the browser as you can easily verify by clicking the link.

Now we will connect to our DB using PHP.

[students.btsi.lu/evegi144/WAD/MySQL/connect\\_oo.php](https://students.btsi.lu/evegi144/WAD/MySQL/connect_oo.php)

```
<?php
require_once 'db_credentials.php'; // Import connection details.
$dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME); // Connect to DB.
// Stop script with error message if connection fails. Don't do this in production
version!
if ($dbc->connect_error) die("Database connection failed: $dbc->connect_error");
$dbc->set_charset("utf8mb4");
echo '<pre>' . print_r($dbc, true) . '</pre>';
echo '<strong>mysqli class methods:</strong>';
echo '<pre>' . print_r(get_class_methods('mysqli'), true) . '</pre>';
$dbc->close(); // Close the DB connection.
?>
```

`$dbc` is a `mysqli` object, containing all the data fields and methods we need to work with our DB (cf. [www.php.net/manual/en/class.mysqli.php](http://www.php.net/manual/en/class.mysqli.php)). It is essential to become familiar with a number of these fields and methods, which we'll use in many of our DB operations. In particular:

- `connect_errno` contains the error code for the last connect call.
- `connect_error` contains a string description of the last connect error.
- `errno` contains the error code for the last function call.
- `error` contains a string description of the last error.
- `close` closes a previously opened DB connection.
- `prepare` prepares an SQL statement for execution. The parameter markers must be bound to application variables using the `bind_param` method of the `stmt` class before executing the statement.

There are four possible data types:

Character	Data type
i	integer
d	double
s	string
b	blob (cf. <a href="http://php.net/manual/en/mysqli-stmt.send-long-data.php">php.net/manual/en/mysqli-stmt.send-long-data.php</a> )

The resulting data can be bound to specific variables and then retrieved using `bind_result` and `fetch` or retrieved using the result object with `get_result` and then using one of the fetch functions described below.

We'll look at practical examples in the following sections.

- `query` performs a query on the DB. Returns FALSE on failure. For successful SELECT, SHOW, DESCRIBE or EXPLAIN queries it returns a `mysqli_result` object, for other successful queries TRUE.
- `escape_string` escapes special characters in a string for use in an SQL statement, taking into account the current charset of the connection. However, `_` and `%` are not escaped, although they have special meaning in LIKE clauses. It is therefore safer to use prepared statements, which will be our preferred approach in the following sections.
- `set_charset` sets the default client character set. We use `utf8`.

### 5.3.14. SQL injection

We need to distinguish between queries that use data provided by the user and those that use internal data. In the first case we need to sanitize user input or use prepared statements to avoid hijacking of our database, which is a major threat to our web app (cf. [en.wikipedia.org/wiki/SQL\\_injection](http://en.wikipedia.org/wiki/SQL_injection)).

Why? Let's look at an example provided by Robin Nixon in his book "PHP, MySQL, JavaScript, & CSS" on page 249:

Suppose you have a simple piece of code to verify a user, and it looks like this:

```
$user = $_POST['user'];  
$password = $_POST['password'];  
$query = "SELECT * FROM tblUser WHERE dtUser='$user' AND dtPassword='$password'";
```

At first glance, you might think this code is perfectly fine. But what if someone enters the following for `$user` (and doesn't enter anything for `$password`)?

```
admin' #
```

Let's look at the string that would be sent to MySQL:

```
SELECT * FROM tblUser WHERE dtUser='admin' #' AND dtPassword=''
```

Do you see the problem? In MySQL, the `#` symbol represents the start of a comment. Therefore, the user will be logged in as `admin` (assuming there is a user `admin`), without having to enter a password.

What about the case in which your application code removes a user from the database? The code might look something like this:

```
$user = $_POST['user'];  
$password = $_POST['password'];  
$query = "DELETE FROM tblUser WHERE dtUser='$user' AND dtPassword='$password'";
```

Again, this looks quite normal at first glance, but what if someone entered the following for `$user`?

```
anything' OR 1=1 #
```

MySQL would interpret this as the following:

```
DELETE FROM tblUser WHERE dtUser='anything' OR 1=1 #' AND dtPassword=''
```

Ouch - that SQL query will always be true, and therefore you've lost all your users table data!

Let's look at a working example.

[students.btsi.lu/evegi144/WAD/MySQL/createHijackDB1.sql](https://students.btsi.lu/evegi144/WAD/MySQL/createHijackDB1.sql)

```
# First select the DB that we want to use.
USE evegi144_dbDemo1;

# Drop table tblHijack if it already exists.
DROP TABLE IF EXISTS tblHijack1;

# Create table tblHijack1.
CREATE TABLE tblHijack1 (
    idHijack1 INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
    dtUserName VARCHAR(20),
    dtPassword VARCHAR(20)
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;

# Insert a row of data into the table.
INSERT INTO tblHijack1 (dtUserName, dtPassword) VALUES
('Mickey', '123'), ('Donald', 'abc');
```

[students.btsi.lu/evegi144/WAD/MySQL/hijack.php](https://students.btsi.lu/evegi144/WAD/MySQL/hijack.php)

```
<?php
if (!isset($_POST['user'], $_POST['pw'])) {
    echo '<form method=post><input placeholder=user name=user required>
        <input placeholder=password name=pw required>
        <button name=login>Login</button>
        <button name=delete>Delete</button></form>';
} else {
    require_once 'db_credentials.php';
    $dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME) or
    die('Connect Error (' . $dbc->connect_errno . ') ' . $dbc->connect_error);
    $dbc->set_charset('utf8');

    $query = "INSERT INTO tblHijack1 (dtUserName, dtPassword) VALUES
        ('Mickey', '123'), ('Donald', 'abc')";
    echo $query . '<br>';
    $result = $dbc->query($query) or die($dbc->errno . ':' . $dbc->error);

    $query = "SELECT * FROM tblHijack1";
    echo $query . '<br>';
    $result = $dbc->query($query) or die($dbc->errno . ':' . $dbc->error);
    $rows = $result->fetch_all();
    echo '<pre>' . print_r($rows, true) . '</pre>';
}
```

```

if (isset($_POST['login'])) {
    // User name and password provided by user
    $user_name = $_POST['user']; // "Donald#"
    $password = $_POST['pw']; // ''
    $query = "SELECT * FROM tblHijack1 WHERE dtUserName = '$user_name' AND
dtPassword
            = '$password'";
    echo $query . '<br>';
    $result = $dbc->query($query) or die($dbc->errno . ':' . $dbc->error);
    echo 'Number of records returned: ' . $result->num_rows . '<br>';
    if ($result->num_rows >= 1) echo "You're logged in!";
} elseif (isset($_POST['delete'])) {
    $user_name = $_POST['user']; // "" or 1=1#"
    $password = $_POST['pw']; // ''
    $query = "DELETE FROM tblHijack1 WHERE dtUserName = '$user_name' AND
dtPassword = "
            . "'$password'";
    echo $query . '<br>';
    $dbc->query($query) or die($dbc->errno . ':' . $dbc->error);

    $query = "SELECT * FROM tblHijack1";
    echo $query . '<br>';
    $result = $dbc->query($query) or die($dbc->errno . ':' . $dbc->error);
    $rows = $result->fetch_all();
    echo '<pre>' . print_r($rows, true) . '</pre>';
}

$dbc->close(); # Close the DB connection.
}
?>

```

See [stackoverflow.com/questions/5741187/sql-injection-that-gets-around-mysql-real-escape-string](https://stackoverflow.com/questions/5741187/sql-injection-that-gets-around-mysql-real-escape-string) for further injection examples.

Check out [www.hacksplaining.com/exercises/sql-injection](http://www.hacksplaining.com/exercises/sql-injection) for an excellent tutorial.



To sanitize user input, we can call the `escape_string` function mentioned above, like this:

```
$user = $dbc->escape_string($_POST['user']);
```



This approach suffers from several major security problems:

1. This does not escape `_` and `%`.
2. Numeric user provided data is not enclosed in quotation marks, in which case escaping won't prevent SQL injection.
3. Storing escaped data in the DB, reading it back and then storing it again will remove the escaping.

If we also want to remove HTML and PHP tags and slashes, we can write a function that we call for every input:

[students.btsi.lu/evegi144/WAD/MySQL/sanitize\\_string.php](https://students.btsi.lu/evegi144/WAD/MySQL/sanitize_string.php)

```
<?php
function sanitize_string($dbc, $string) {
    $string = strip_tags(trim($string)); # Remove HTML and PHP tags from string.
    /* If slashes have been added automatically, remove them.
       Starting from PHP 5.4 this is not necessary anymore
       (cf. http://php.net/manual/en/function.get-magic-quotes-gpc.php)
    */
    if (get_magic_quotes_gpc()) $string = stripslashes($string);
    $result = $dbc->escape_string($string); # Escape special characters.
    return $result;
}
?>
```

To avoid the problem of questionable user input we could force the user to enter only alphanumeric characters using `ctype_alnum`:

```
if (!ctype_alnum($username)) die('Invalid characters in Username');
```

The preferred way is to use prepared statements (cf. [en.wikipedia.org/wiki/Prepared\\_statement](https://en.wikipedia.org/wiki/Prepared_statement)):

Prepared statements are resilient against SQL injection, because parameter values, which are transmitted later using a different protocol, need not be correctly escaped.

Prepared statements separate query structure from data. Given that user data has no influence on query structure, this is the safest approach. A good article on the subject can be found at [websitebeaver.com/prepared-statements-in-php-mysqli-to-prevent-sql-injection](https://websitebeaver.com/prepared-statements-in-php-mysqli-to-prevent-sql-injection).

### 5.3.15. Retrieving data

In SQL we retrieve data from our DB using the `SELECT` statement. A straightforward explanation can be found at [www.w3schools.com/sql/sql\\_select.asp](http://www.w3schools.com/sql/sql_select.asp). The gory details are at [dev.mysql.com/doc/refman/5.7/en/select.html](https://dev.mysql.com/doc/refman/5.7/en/select.html). We can specify columns for which we want to retrieve data or use `*` to indicate that we want data for all columns.

We can retrieve data for all rows or only for those rows that fulfill conditions. The conditions are specified using the **WHERE** clause. We can even use regular expressions, which are explained in detail at [dev.mysql.com/doc/refman/5.7/en/regexp.html](http://dev.mysql.com/doc/refman/5.7/en/regexp.html).

Let's look at a few SQL query examples:

[students.btsi.lu/evegi144/WAD/MySQL/select1.sql](http://students.btsi.lu/evegi144/WAD/MySQL/select1.sql)

```
SELECT * FROM evegi144_dbDemo1.tblT1; # Select all columns from the tblT1 table of the
demo1 DB.
USE evegi144_dbDemo1; # Select the demo1 DB as default.
SELECT dtLastName FROM tblT1; # Select the last name from table tblT1 of the currently
used DB.
# Get all first names where last name is Gates or Torvalds
SELECT dtFirstName FROM tblT1 WHERE dtLastName = 'Gates' OR dtLastName = 'Torvalds';
# Get all first names where last name is ate.
SELECT dtFirstName FROM tblT1 WHERE dtLastName LIKE 'ate';
# Get all first names where last name contains the word ate.
SELECT dtFirstName FROM tblT1 WHERE dtLastName LIKE '%ate%';
# Get all first names where last name starts with G and ends with s.
SELECT dtFirstName FROM tblT1 WHERE dtLastName REGEXP '^G.*s$';
# Get all data ordered by last name in descending order.
SELECT * FROM tblT1 ORDER BY dtLastName DESC;
```

How do we execute SQL queries in PHP? We can use the object-oriented or the procedural approach. In the former we work with objects, such as a DB connection or result set and use a property or call a method (function) of this object using `→`, for instance `$result→num_rows` or `$dbc→query($query)`. The procedural approach requires that we call the corresponding `mysqli` function and provide the variable to be used. Thus we need to write `mysqli_num_rows($result)` or `mysqli_query($dbc, $query)`.

### With internal data

With internal data we do not need to take special precautions to avoid DB hijacking although the safest approach would be to always use prepared statements, given that we cannot be sure that the data was not processed in some unexpected way.

[students.btsi.lu/evegi144/WAD/MySQL/select1\\_oo.php](http://students.btsi.lu/evegi144/WAD/MySQL/select1_oo.php)

```

<?php
    require_once 'db_credentials.php'; # Include the DB connection credentials.
    /* Create a new DB connection. We need to store the connection as we will need it
    for all DB
        operations. If something goes wrong, $dbc will be null and we display the
        connection error number as well as the error text and terminate the script. */
    $dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME);
    if ($dbc->connect_error) die("Database connection failed: $dbc->connect_error");
    $dbc->set_charset('utf8'); # Specify the character set to be used for our queries.
    $query = 'SELECT * FROM tblT1'; # Our query

    /* Execute the SQL query. To do this we need to provide the query.
        If something goes wrong, we display the error message and terminate the script.
        If the query executed correctly, we get a result object which we store in
    $result. */
    $result = $dbc->query($query); # Store the result object.
    if (!$result) die("Wrong SQL: $query Error: $dbc->error");

    # Take a look at the result object.
    echo '<pre>' . print_r($result, true) . '</pre>';

    # Display the number of records returned by the query.
    echo '<br>Number of rows: ' . $result->num_rows;

    /* Retrieve each record one by one. fetch_assoc returns the next record until we
        have reached the last one. We append each record to the $all_rows array. */
    while ($row = $result->fetch_assoc()) $rows[] = $row;

    # Let's take a look at the result array.
    echo '<pre>' . print_r($rows, true) . '</pre>';

    /* Since PHP 5.4 the mysqli result object supports iteration (cf.
        https://secure.php.net/manual/en/class.mysqli-result.php) which makes our life
    easier: */
    foreach ($result as $row) echo '<pre>' . print_r($row, true) . '</pre>';

    if ($result->data_seek(0)) // Set the result pointer back to the beginning.
        while ($row = $result->fetch_array())
            echo '<pre>' . print_r($row, true) . '</pre>';

    echo '<pre>' . print_r($dbc, true) . '</pre>';
    $result->free(); # Free the memory used by the result object.
    $dbc->close(); # Close the DB connection.
?>

```

First we establish a DB connection and set the character set. Then we define our query and execute it using the **query** method. This method returns **FALSE** if something went wrong, otherwise **TRUE** or the result object of the query, depending on the type of query. The **result** object has several interesting properties and methods.

For our purposes, the following property and methods are particularly useful:

Property/method	Description
<a href="#">num_rows</a>	Number of rows returned by query.
<a href="#">fetch_row</a>	Fetch current result row as a numeric array.
<a href="#">fetch_assoc</a>	Fetch current result row as an associative array.
<a href="#">fetch_array</a>	Fetch current result row as an associative, a numeric array, or both.
<a href="#">fetch_all</a>	Fetch all result rows as an associative array, a numeric array, or both.
<a href="#">fetch_object</a>	Fetch current result row as a PHP object.



Since PHP 5.4 the mysqli [result](#) object supports iteration which makes our life easier (see code above).



The result set has an internal pointer representing the current row. Every call to a fetch method increases this internal pointer. Thus, every fetch will return the next row until the end of the result set has been reached. If we want to retrieve a different row, we can use the [data\\_seek](#) method to move the pointer.

To finish the job we free the memory of the result object using its [free](#) method and we close the DB connection.

[fetch\\_array](#) returns an array that corresponds to the fetched row or NULL if there are no more rows for the result set.

Example:

[students.btsi.lu/evegi144/WAD/MySQL/select2\\_oo.php](http://students.btsi.lu/evegi144/WAD/MySQL/select2_oo.php)

```
<?php
require_once 'db_credentials.php';
$dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
    $dbc->connect_errno . ') ' . $dbc->connect_error);
$dbc->set_charset('utf8');

$query = 'SELECT * FROM tblT1';
$result = $dbc->query($query);
if (!$result) die("Wrong SQL: $query Error: $dbc->error");
while ($row = $result->fetch_array(MYSQLI_ASSOC)) $rows[] = $row;
echo '<pre>' . print_r($rows, true) . '</pre>';
$result->free();
$dbc->close();
?>
```

[fetch\\_all](#) returns an array of associative or numeric arrays holding result rows.

[students.btsi.lu/evegi144/WAD/MySQL/select3\\_oo.php](http://students.btsi.lu/evegi144/WAD/MySQL/select3_oo.php)

```
<?php
require_once 'db_credentials.php';
$dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
    $dbc->connect_errno . ') ' . $dbc->connect_error);
$dbc->set_charset('utf8');

$query = 'SELECT * FROM tblT1';
$result = $dbc->query($query);
if (!$result) die("Wrong SQL: $query Error: $dbc->error");
$rows = $result->fetch_all();
echo '<pre>' . print_r($rows, true) . '</pre>';
$result->free();
$dbc->close();
?>
```

## With user provided data

If some of the data used to query the DB is user provided, we either need to sanitize it ourselves or use prepared statements.

### Using `escape_string`

All we have to do is run the user provided inputs through `escape_string`, like so:

[students.btsi.lu/evegi144/WAD/MySQL/select4\\_escape\\_oo.php](https://students.btsi.lu/evegi144/WAD/MySQL/select4_escape_oo.php)

```
<?php
require_once 'db_credentials.php'; # Include constants for host, user, pw and DB.
# Create a new DB connection.
$dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
    $dbc->connect_errno . ') ' . $dbc->connect_error);
$dbc->set_charset('utf8'); # Specify the character set to be used for our queries.

$last_name = "Duck";
$last_name = $dbc->escape_string($last_name);
$query = "SELECT dtFirstName, dtLastName FROM tblT1 WHERE dtLastName =
'$last_name'"; # query
$result = $dbc->query($query) or die("Wrong SQL: $query Error: $dbc->error");
echo 'Number of rows ' . $result->num_rows . '<br>'; # Display the number of rows.
# Loop through the result object and display each first name.
while ($row = $result->fetch_array()) printf("%s %s<br>", $row[0], $row[1]);
$dbc->close(); # Close the DB connection.
?>
```

Keep in mind that escaping is not fully secure.

### Using prepared statements

[students.btsi.lu/evegi144/WAD/MySQL/select4\\_oo.php](https://students.btsi.lu/evegi144/WAD/MySQL/select4_oo.php)

```

<?php
    require_once 'db_credentials.php';
    $dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
        $dbc->connect_errno . ') ' . $dbc->connect_error);
    $dbc->set_charset('utf8');

    $query = 'SELECT dtFirstName, dtLastName FROM tblT1 WHERE dtLastName = ?'; # our
query
    $stmt = $dbc->prepare($query); # Prepare query and store statement object in $stmt.
    if (!$stmt) die('Wrong SQL: ' . $query . ' Error: ' . $dbc->error); # If query
preparation failed, display error and stop script.
    echo '<pre>' . print_r($stmt, true) . '</pre>';

    $last_name = 'Gates'; # Save the last name that we want to use in our query in a
variable.
    # Bind the parameter to our query as a string.
    $stmt->bind_param('s', $last_name) or die('Bind failure: ' . $stmt->error);
    $stmt->execute() or die($stmt->error);
    $result = $stmt->get_result();
    if ($result) {
        echo 'Result object properties:<br>';
        echo '<pre>' . print_r($result, true) . '</pre>';
        echo 'Result object methods:<br>';
        //echo '<pre>' . print_r(get_class_vars(get_class($result)), true) . '</pre>';
        echo '<pre>' . print_r(get_class_methods(get_class($result)), true) . '</pre>';
        echo 'Number of rows: ' . $result->num_rows;
        echo '<pre>' . print_r($result->fetch_all(), true) . '</pre>';
    }
    $stmt->close();
    $dbc->close();
?>

```

**prepare** returns a prepared statement object or **FALSE** if something went wrong. See [www.php.net/manual/en/class.mysqli-stmt.php](http://www.php.net/manual/en/class.mysqli-stmt.php).

**store\_result** returns a buffered result object or **FALSE** if something went wrong. To see what this object looks like, go to [php.net/manual/en/class.mysqli-result.php](http://php.net/manual/en/class.mysqli-result.php). Once we have a result object, we can for instance figure out how many rows have been retrieved.

With **bind\_result** we can bind columns in the result set to variables. To free the resources used for our query, we close the prepared statement and the DB connection.

**get\_result** returns a result set from a prepared statement.

[students.btsi.lu/evegi144/WAD/MySQL/select5\\_oo.php](http://students.btsi.lu/evegi144/WAD/MySQL/select5_oo.php)

```

<?php
    require_once 'db_credentials.php';
    $dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
        $dbc->connect_errno . ') ' . $dbc->connect_error);
    $dbc->set_charset('utf8');

    $query = 'SELECT dtFirstName, dtLastName FROM tblT1 WHERE dtLastName = ?'; # our
query
    $stmt = $dbc->prepare($query); # Prepare query and store statement object in $stmt.
    if (!$stmt) die('Wrong SQL: ' . $query . ' Error: ' . $dbc->error); # If query
preparation failed, display error and stop script.
    echo '<pre>' . print_r($stmt, true) . '</pre>';

    $last_name = 'Gates'; # Save the last name that we want to use in our query in a
variable.
    # Bind the parameter to our query as a string.
    $stmt->bind_param('s', $last_name) or die('Bind failure: ' . $stmt->error);
    $stmt->execute() or die($stmt->error);
    $result = $stmt->get_result();
    if ($result)
        while ($row = $result->fetch_array(MYSQLI_NUM))
            printf("%s %s<br>", $row[0], $row[1]);
    $stmt->close();
    $dbc->close();
?>

```

[students.btsi.lu/evegi144/WAD/MySQL/select6\\_oo.php](http://students.btsi.lu/evegi144/WAD/MySQL/select6_oo.php)

```

<?php
    require_once 'db_credentials.php'; # Include constants for host, user, pw and DB.
    # Create a new DB connection.
    $dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
        $dbc->connect_errno . ') ' . $dbc->connect_error);
    $dbc->set_charset('utf8'); # Specify the character set to be used for our queries.

    $query = 'SELECT dtFirstName, dtLastName FROM tblT1 WHERE dtLastName = ?'; # our
query
    $stmt = $dbc->prepare($query); # Prepare query and store statement object in $stmt.
    # If query preparation failed, display error and stop script.
    if (!$stmt) die("Wrong SQL: $query Error: $dbc->error");
    echo '<pre>' . print_r($stmt, true) . '</pre>';

    $last_name = 'Gates'; # Save the last name that we want to use in our query in a
variable.
    # Bind the parameter to our query as a string.
    $stmt->bind_param('s', $last_name) or die('Bind failure: ' . $stmt->error);
    $stmt->execute() or die($stmt->error); # Execute query and stop if failure.
    $stmt->store_result() or die($stmt->error); # Store the result object.
    echo 'Number of rows ' . $stmt->num_rows . '<br>'; # Display the number of rows.
    $stmt->bind_result($result[0], $result[1]) or die($stmt->error); # Bind the query
result to variables.
    # Loop through the result object and display each first name.
    while ($stmt->fetch()) printf("%s %s<br>", $result[0], $result[1]);
    $stmt->close();
    $dbc->close(); # Close the DB connection.
?>

```

### 5.3.16. Inserting data

In SQL we insert data into our DB using the **INSERT** statement. A straightforward explanation can be found at [www.w3schools.com/sql/sql\\_insert.asp](http://www.w3schools.com/sql/sql_insert.asp). The gory details are at [dev.mysql.com/doc/refman/5.7/en/insert.html](http://dev.mysql.com/doc/refman/5.7/en/insert.html).

#### From a CSV file

[www.mysqltutorial.org/import-csv-file-mysql-table](http://www.mysqltutorial.org/import-csv-file-mysql-table)

[medium.com/@AviGoom/how-to-import-a-csv-file-into-a-mysql-database-ef8860878a68](https://medium.com/@AviGoom/how-to-import-a-csv-file-into-a-mysql-database-ef8860878a68)

[dev.mysql.com/doc/refman/8.0/en/load-data.html](http://dev.mysql.com/doc/refman/8.0/en/load-data.html)

#### Preparing strings for database insertion

First we should in most cases remove any whitespace from the beginning and end of any input provided by the user using the **trim** function. The following characters can cause problems when inserting data into a database, as they may be interpreted as control characters by the database: **"**, **'**, **\**, **NULL**. We need to tell MySQL that we mean these characters literally and not as control characters. To do this, we add a backslash in front of them.

Once our insertion statement has been executed, we can easily get the id of the inserted record (cf. [php.net/manual/en/mysqli.insert-id.php](http://php.net/manual/en/mysqli.insert-id.php)):

The `mysqli_insert_id` function returns the ID generated by a query on a table with a column having the `AUTO_INCREMENT` attribute. If the last query wasn't an `INSERT` or `UPDATE` statement or if the modified table does not have a column with the `AUTO_INCREMENT` attribute, this function will return zero.

In the case of prepared statements, we use the `insert_id` field of the statement (cf. examples below).

### With internal data

[students.btsi.lu/evegi144/WAD/MySQL/insert1\\_oo.php](http://students.btsi.lu/evegi144/WAD/MySQL/insert1_oo.php)

```
<?php
exit; // to avoid misuse
require_once 'db_credentials.php';
$dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME) or
die('Connect Error (' . $dbc->connect_errno . ') ' . $dbc->connect_error);
$dbc->set_charset('utf8');

$first_name = $dbc->escape_string('Bill');
$last_name  = $dbc->escape_string('Gates');
$query      = "INSERT INTO tblT1 (dtFirstName, dtLastName)
              VALUES ('$first_name', '$last_name')";
$dbc->query($query) or die('Error inserting into DB: ' . $dbc->error);
echo "Id of inserted record: " . $dbc->insert_id . '<br>';

$query = 'SELECT dtFirstName, dtLastName FROM tblT1 WHERE idT1=' . $dbc->insert_id;
echo $query . '<br>';
$result = $dbc->query($query) or die('Error inserting into DB: ' . $dbc->error);
$row = $result->fetch_assoc();
print_r($row);
$dbc->close(); # Close the DB connection.
?>
```

### With user provided data

The simplest approach is to use `escape_string` on all user provided data, as illustrated above. The safer approach is to use prepared statements.

For illustration of how vulnerable insert queries can be to SQL injection, take a look at [amolnaik4.blogspot.lu/2012/02/sql-injection-in-insert-query.html](http://amolnaik4.blogspot.lu/2012/02/sql-injection-in-insert-query.html).

[students.btsi.lu/evegi144/WAD/MySQL/insert2\\_oo.php](http://students.btsi.lu/evegi144/WAD/MySQL/insert2_oo.php)

```

<?php
    //exit; // to avoid misuse
    require_once 'db_credentials.php';
    $dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME);
    if ($dbc->connect_error) die('Database connection failed: ' . $dbc->connect_error);
    $dbc->set_charset('utf8');

    $first_name = 'Bill';
    $last_name  = 'Gates';
    $query      = 'INSERT INTO tblT1 (dtFirstName, dtLastName) VALUES (?, ?)';
    $stmt = $dbc->prepare($query);
    if (!$stmt) die("Wrong SQL: $query Error: $dbc->error");
    $stmt->bind_param('ss', $first_name, $last_name) or die('Bind failure: ' . $stmt->error);
    $stmt->execute() or die('Error inserting into DB: ' . $stmt->error);
    echo "Id of inserted record: " . $stmt->insert_id . '<br>';

    $first_name = 'Linus';
    $last_name  = 'Torvalds';
    $stmt->bind_param('ss', $first_name, $last_name) or die('Bind failure: ' . $stmt->error);
    $stmt->execute() or die('Error inserting into DB: ' . $stmt->error);
    echo "Id of inserted record: " . $stmt->insert_id;

    $stmt->close();
    $dbc->close();
?>

```

## Storing objects, arrays and other complex data types

If we want to store more complex PHP data types in a MySQL table, we can use the **JSON** data type (cf. [JSON](#)) or alternatively the **serialize** and **unserialize** functions. Serialized data should be stored in a MySQL **BLOB**.

### 5.3.17. Deleting data

In SQL we delete data from our DB using the **DELETE** statement. A straightforward explanation can be found at [www.w3schools.com/sql/sql\\_delete.asp](http://www.w3schools.com/sql/sql_delete.asp). The gory details are at [dev.mysql.com/doc/refman/5.7/en/delete.html](http://dev.mysql.com/doc/refman/5.7/en/delete.html).

Deleting all the data in a table is very straightforward in MySQL:

[students.btsi.lu/evegi144/WAD/MySQL/delete1.sql](http://students.btsi.lu/evegi144/WAD/MySQL/delete1.sql)

```

# Delete all data in table tblT1 of DB demo1. The table structure remains.
DELETE FROM demo1.tblT1;

```

We can filter which data gets deleted using the **WHERE** clause, just as for **SELECT** queries:

[students.btsi.lu/evegi144/WAD/MySQL/delete2.sql](https://students.btsi.lu/evegi144/WAD/MySQL/delete2.sql)

```
DELETE FROM demo1.tb1T1 WHERE dtFirstName = 'Bill';
```

### With internal data

[students.btsi.lu/evegi144/WAD/MySQL/delete1\\_oo.php](https://students.btsi.lu/evegi144/WAD/MySQL/delete1_oo.php)

```
<?php
require_once 'db_credentials.php';
$dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME);
if ($dbc->connect_error) die("Database connection failed: $dbc->connect_error");
$dbc->set_charset('utf8');

$query = 'DELETE FROM tb1T1 WHERE dtLastName = "Gates"';
$result = $dbc->query($query);
if (!$result) die("Wrong SQL: $query Error: $dbc->error");
$dbc->close(); // Close DB.
?>
```

### With user provided data

The simplest approach is to use `escape_string` on all user provided data, as illustrated above. The preferred approach is to use prepared statements.

[students.btsi.lu/evegi144/WAD/MySQL/delete2\\_oo.php](https://students.btsi.lu/evegi144/WAD/MySQL/delete2_oo.php)

```
<?php
require_once 'db_credentials.php';
$dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME);
if ($dbc->connect_error) die("Database connection failed: $dbc->connect_error");
$dbc->set_charset('utf8');

$query = 'DELETE FROM tb1T1 WHERE dtLastName = ?';
$stmt = $dbc->prepare($query);
if (!$stmt) die("Wrong SQL: $query Error: $dbc->error");
$last_name = 'Gates';
$stmt->bind_param('s', $last_name) or die('Bind failure: ' . $stmt->error);
$stmt->execute() or die($stmt->error);
$stmt->close(); // Close prepared statement.
$dbc->close(); // Close DB connection.
?>
```

## 5.3.18. Updating data

In SQL we update data in our DB using the `UPDATE` statement. A straightforward explanation can be found at [www.w3schools.com/sql/sql\\_update.asp](http://www.w3schools.com/sql/sql_update.asp). The gory details are at [dev.mysql.com/doc/](http://dev.mysql.com/doc/)

[refman/5.7/en/update.html](http://refman/5.7/en/update.html).

A simple example:

[students.btsi.lu/evegi144/WAD/MySQL/update1.sql](http://students.btsi.lu/evegi144/WAD/MySQL/update1.sql)

```
UPDATE demo1.tbLT1 SET dtFirstName = 'James', dtLastName = 'Bond' WHERE dtLastName = 'Gates';
```

### With internal data

[students.btsi.lu/evegi144/WAD/MySQL/update1\\_oo.php](http://students.btsi.lu/evegi144/WAD/MySQL/update1_oo.php)

```
<?php
require_once 'db_credentials.php';
$dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME);
if ($dbc->connect_error) die('Database connection failed: ' . $dbc->connect_error);
$dbc->set_charset('utf8');

$query = 'UPDATE tblT1 SET dtFirstName = "James", dtLastName = "Bond" WHERE ' .
        'dtLastName = "Gates"';
$result = $dbc->query($query);
if (!$result) die("Wrong SQL: $query Error: $dbc->error");
$dbc->close();
?>
```

### With user provided data

The simplest approach is to use `escape_string` on all user provided data, as illustrated above. The preferred approach is to use prepared statements.

[students.btsi.lu/evegi144/WAD/MySQL/update2\\_oo.php](http://students.btsi.lu/evegi144/WAD/MySQL/update2_oo.php)

```

<?php
    require_once 'db_credentials.php';
    $dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME);
    if ($dbc->connect_error) die('Database connection failed: ' . $dbc->connect_error);
    $dbc->set_charset('utf8');

    $query = 'UPDATE tblT1 SET dtLastName = ? WHERE dtLastName = ?';
    $stmt = $dbc->prepare($query);
    if (!$stmt) die("Wrong SQL: $query Error: $dbc->error");
    $new_last_name = 'Duck';
    $old_last_name = 'Bond';
    $stmt->bind_param('ss', $new_last_name, $old_last_name) or
        die('Bind failure: ' . $stmt->error);
    $stmt->execute() or die($stmt->error);
    $stmt->close();
    $dbc->close();
?>

```

### 5.3.19. Altering tables

In SQL we can change the structure of a table using the **ALTER TABLE** statement. A straightforward explanation can be found at [www.w3schools.com/sql/sql\\_alter.asp](http://www.w3schools.com/sql/sql_alter.asp). The gory details are at [dev.mysql.com/doc/refman/5.7/en/alter-table.html](http://dev.mysql.com/doc/refman/5.7/en/alter-table.html).

A simple example:

[students.btsi.lu/evegi144/WAD/MySQL/alter\\_table1\\_.sql](http://students.btsi.lu/evegi144/WAD/MySQL/alter_table1_.sql)

```

ALTER TABLE demo1.tblT1 ADD dtBirthDate date;

```

### 5.3.20. DB normalization

Normalization is the process of organizing our DB efficiently in order to avoid anomalies that will prevent our application from working correctly (cf. [en.wikipedia.org/wiki/Database\\_normalization](http://en.wikipedia.org/wiki/Database_normalization)).

Study the following excellent explanations:

1. [www.tutorialspoint.com/sql/sql-rdbms-concepts.htm](http://www.tutorialspoint.com/sql/sql-rdbms-concepts.htm)
2. [www.tutorialspoint.com/sql/first-normal-form.htm](http://www.tutorialspoint.com/sql/first-normal-form.htm)
3. [www.tutorialspoint.com/sql/second-normal-form.htm](http://www.tutorialspoint.com/sql/second-normal-form.htm)
4. [www.tutorialspoint.com/sql/third-normal-form.htm](http://www.tutorialspoint.com/sql/third-normal-form.htm)

### 5.3.21. Creating a safe registration and login

## Rudimentary version

We need the following:

1. A DB table to store our users.
2. A PHP class that provides login and registration functionality for our user table.
3. A login and registration script.

### User table

As recommended at [php.net/manual/en/password.constants.php](http://php.net/manual/en/password.constants.php), we use a `VARCHAR(255)` to store the password hash.

[students.btsi.lu/evegi144/WAD/MySQL/loginDB1.sql](http://students.btsi.lu/evegi144/WAD/MySQL/loginDB1.sql)

```
DROP TABLE IF EXISTS tblUser;

CREATE TABLE tblUser (
  idUser          INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
  dtUserName      VARCHAR(32) NOT NULL UNIQUE,
  dtPasswordHash  VARCHAR(255) NOT NULL
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;
```

### DB class

[students.btsi.lu/evegi144/WAD/MySQL/loginDB1.php](http://students.btsi.lu/evegi144/WAD/MySQL/loginDB1.php)

```
<?php
require_once 'login1_db_credentials.php';

class Database {
  private static $DB_HOST;
  private static $DB_USER;
  private static $DB_PASSWORD;
  private static $DB_NAME;
  private static $DB_USER_TABLE = 'tblUser';

  static function set_credentials($db_host, $db_user, $db_password, $db_name) {
    self::$DB_HOST      = $db_host;
    self::$DB_USER      = $db_user;
    self::$DB_PASSWORD  = $db_password;
    self::$DB_NAME      = $db_name;
  }

  static function connect() {
    $dbc = new mysqli(self::$DB_HOST, self::$DB_USER, self::$DB_PASSWORD,
```

```

self::$DB_NAME);
    if ($dbc->connect_error) trigger_error('Database connection failed: ' .
        $dbc->connect_error, E_USER_ERROR);
    $dbc->set_charset("utf8");
    return $dbc;
}

# Returns user id or FALSE.
static function login($user_name, $password) {
    $dbc = self::connect();
    $query = 'SELECT idUser, dtPasswordHash FROM ' . self::$DB_USER_TABLE .
        ' WHERE dtUserName = ?';
    $stmt = $dbc->prepare($query) or trigger_error('Wrong SQL: ' . $query .
        ' Error: ' . $dbc->error, E_USER_ERROR);
    $stmt->bind_param('s', $user_name);
    if ($stmt->execute()) {
        $result = $stmt->get_result();
        if ($result && $result->num_rows === 1) {
            $row = $result->fetch_assoc();
            if ($row && password_verify($password, $row['dtPasswordHash'])) return
$row['idUser'];
        }
    } else trigger_error('Wrong SQL: ' . $query .
        ' Error: ' . $dbc->error, E_USER_ERROR);
    return false;
}

# Returns FALSE if user could not be created, otherwise user id.
static function create_user($user_name, $password) {
    $dbc = self::connect();
    $query = 'INSERT INTO ' . self::$DB_USER_TABLE . ' (dtUserName, dtPasswordHash)'
        .
        ' VALUES(?, ?)';
    $stmt = $dbc->prepare($query) or trigger_error('Wrong SQL: ' . $query . ' Error:
    ' .
        $dbc->error, E_USER_ERROR);
    $pw_hash = password_hash($password, PASSWORD_DEFAULT);
    $stmt->bind_param('ss', $user_name, $pw_hash);
    if ($stmt->execute()) return $stmt->insert_id;
    else return false;
}
}
?>

```

## Login and registration script

[students.btsi.lu/evegi144/WAD/MySQL/login1.php](http://students.btsi.lu/evegi144/WAD/MySQL/login1.php)

```

<?php
// Protect from session fixation via session adoption.

```

```

ini_set('session.use_strict_mode', true);
# Only send session id cookie over SSL.
ini_set('session.cookie_secure', true);
# Session IDs may only be passed via cookies, not appended to URL.
ini_set('session.use_only_cookies', true);
ini_set('session.cookie_httponly', true);
// Set the path for the cookie to the current directory in order to prevent it from
// being available to scripts in other directories.
ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
if (!isset($_SERVER['HTTPS'])) { // If SSL is not active, activate it.
    header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']));
    exit;
}
// If no session is started yet, we'll start one.
if (!isset($_SESSION)) session_start();
// After 30 seconds we'll generate a new session ID to prevent a session fixation
// attack (cf. PHP cookbook p. 338).
if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
    session_regenerate_id();
    $_SESSION['generated'] = time();
}
// Include the database class needed to access the database.
require_once 'loginDB1.php';
// If a user is already logged in, let him through to the main page.
if (isset($_SESSION['user_id'])) {
    header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/logged_in1.php');
    exit;
}
// Else, if the user has submitted his login details, we need to check them.
elseif (isset($_POST['login'])) {
    if (isset($_POST['username'], $_POST['password'])) {
        $result = Database::login($_POST['username'], $_POST['password']);
        // If a user with this login exists, we load the main page.
        if ($result) {
            $_SESSION['user_id'] = $result;
            $_SESSION['user_name'] = $_POST['username'];
            header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
                dirname($_SERVER['PHP_SELF']) . '/logged_in1.php');
            exit;
        }
    }
}
// Else, if the user has signed up for a new account, we need to check if
// such a user already exists.
elseif (isset($_POST['register'])) {
    if (isset($_POST['username'], $_POST['pw1'], $_POST['pw2']) && $_POST['pw1'] ===
        $_POST['pw2'])
    )
        if (Database::create_user($_POST['username'], $_POST['pw1']))

```

```

        echo "<script>alert('Registration succeeded, please log in!');</script>";
        else echo "<script>alert('Registration failed!');</script>";
    }
?>

<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Login and registration</title>
        <meta charset=utf-8>
    </head>
    <body>
        <main>
            <form method=post>
                <input name=username placeholder="User name" required>
                <input name=password type=password placeholder=Password required>
                <button name=login>Log in</button>
            </form>
            <form method=post>
                <input name=username placeholder="User name" required>
                <input name=pw1 type=password placeholder=Password required>
                <input name=pw2 type=password placeholder="Repeat password" required>
                <button name=register>Register</button>
            </form>
        </main>
    </body>
</html>

```

[students.btsi.lu/evegi144/WAD/MySQL/logged\\_in1.php](https://students.btsi.lu/evegi144/WAD/MySQL/logged_in1.php)

```

<?php
    // Protect from session fixation via session adoption.
    ini_set('session.use_strict_mode', true);
    # Only send session id cookie over SSL.
    ini_set('session.cookie_secure', true);
    # Session IDs may only be passed via cookies, not appended to URL.
    ini_set('session.use_only_cookies', true);
    ini_set('session.cookie_httponly', true);
    // Set the path for the cookie to the current directory in order to prevent it from
    // being available to scripts in other directories.
    ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
    if (!isset($_SERVER['HTTPS'])) // If SSL is not active, activate it.
        header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
            dirname($_SERVER['PHP_SELF']));
    // If no session is started yet, we'll start one.
    if (!isset($_SESSION)) session_start();
    // After 30 seconds we'll generate a new session ID to prevent a session fixation
    // attack (cf. PHP cookbook p. 338).
    if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
        session_regenerate_id();
    }

```

```

    $_SESSION['generated'] = time();
}
// If no user is logged in, send him to the login and registration page.
if (isset($_POST['logout'])) {
    $_SESSION = [];
    if (session_id() != "" || isset($_COOKIE[session_name()]))
setcookie(session_name(),
    '', 1, rawurlencode(dirname($_SERVER['PHP_SELF'])));
    session_destroy();
}
if (!isset($_SESSION['user_id']))
    header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/login1.php');
?>

<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Login and registration</title>
        <meta charset=utf-8>
    </head>
    <body>
        <main>
            <h1>Welcome <?php if (isset($_SESSION['user_name'])) echo
$_SESSION['user_name']; ?></h1>
            <form method=post>
                <button name=logout>Log out</button>
            </form>
        </main>
    </body>
</html>

```

### A more sophisticated approach

[stackoverflow.com/questions/244882/what-is-the-best-way-to-implement-remember-me-for-a-website](https://stackoverflow.com/questions/244882/what-is-the-best-way-to-implement-remember-me-for-a-website)

[paragonie.com/blog/2015/04/secure-authentication-php-with-long-term-persistence#title.2](https://paragonie.com/blog/2015/04/secure-authentication-php-with-long-term-persistence#title.2)

### 5.3.22. Procedural use of mysqli

[students.btsi.lu/evegi144/WAD/MySQL/connect\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/connect_procedural.php)

```
<?php
    require_once 'db_credentials.php'; // Import connection details.
    // Connect to DB. Stop script with error message if connection fails. Don't do this
    in
    // production version!
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or
        die('Connect Error (' . mysqli_connect_errno() . ') ' . mysqli_connect_error());
    mysqli_set_charset($dbc, "utf8"); // Set the default character set.
    echo '<pre>' . print_r($dbc, true) . '</pre>';
    echo '<strong>mysqli class methods:</strong>';
    echo '<pre>' . print_r(get_class_methods('mysqli'), true) . '</pre>';
    mysqli_close($dbc); // Close the DB connection.
?>
```

[students.btsi.lu/evegi144/WAD/MySQL/select1\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/select1_procedural.php)

```

<?php
    require_once 'db_credentials.php'; // Include the DB connection credentials.

    // Connect to the DB. We need to store the connection as we will need it for all DB
    // operations. If something goes wrong, $dbc will be null and we display the
    // connection error number as well as the error text and terminate the script.
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
        mysqli_connect_errno() . ') ' . mysqli_connect_error());

    // Don't forget to set the character set, otherwise you are looking for trouble!
    mysqli_set_charset($dbc, 'utf8');

    $query = 'SELECT * FROM tblT1'; // Store our SQL query in a variable.

    // Execute the SQL query. To do this we need to provide the DB connection and the
    query.
    // If something goes wrong, we display the error message and terminate the script.
    // If the query executed correctly, we get a result object which we store in
    $result.
    $result = mysqli_query($dbc, $query) or die("Wrong SQL: $query Error: " .
        mysqli_error($dbc));

    // This is just to take a look at the result object.
    echo '<pre>' . print_r($result, true) . '</pre>';

    // Display the number of records returned by the query.
    echo '<br>Number of rows: ' . mysqli_num_rows($result);

    // Retrieve each record one by one. mysqli_fetch_assoc returns the next record until
    we
    // have reached the last one. We append each record to the $all_rows array.
    while ($row = mysqli_fetch_assoc($result)) $all_rows[] = $row;

    // Let's take a look at the result array.
    echo '<pre>' . print_r($all_rows, true) . '</pre>';

    // Since PHP 5.4 the mysqli result object supports iteration (cf.
    // https://secure.php.net/manual/en/class.mysqli-result.php) which makes our life
    easier.
    // Note however, that $row will only be an associative, not a numerically indexed
    array.
    foreach ($result as $row) echo '<pre>' . print_r($row, true) . '</pre>';

    if (mysqli_data_seek($result, 0)) // Set the result pointer back to the beginning.
        while ($row = mysqli_fetch_array($result))
            echo '<pre>' . print_r($row, true) . '</pre>';

    mysqli_free_result($result); # Free the memory used by the result object.
    mysqli_close($dbc); # Close the Db connection.
?>

```

[students.btsi.lu/evegi144/WAD/MySQL/select2\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/select2_procedural.php)

```
<?php
require_once 'db_credentials.php';
$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
    mysqli_connect_errno() . ') ' . mysqli_connect_error());
mysqli_set_charset($dbc, 'utf8');

$query = 'SELECT * FROM tblT1';
$result = mysqli_query($dbc, $query);
if (!$result) die("Wrong SQL: $query Error: " . mysqli_error($dbc));
while ($row = mysqli_fetch_array($result, MYSQLI_ASSOC)) $rows[] = $row;
echo '<pre>' . print_r($rows, true) . '</pre>';
mysqli_free_result($result);
mysqli_close($dbc);
?>
```

[students.btsi.lu/evegi144/WAD/MySQL/select3\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/select3_procedural.php)

```
<?php
require_once 'db_credentials.php';
$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
    mysqli_connect_errno() . ') ' . mysqli_connect_error());
mysqli_set_charset($dbc, 'utf8');

$query = 'SELECT * FROM tblT1';
$result = mysqli_query($dbc, $query);
if (!$result) die("Wrong SQL: $query Error: " . mysqli_error($dbc));
$rows = mysqli_fetch_all($result);
echo '<pre>' . print_r($rows, true) . '</pre>';
mysqli_free_result($result);
mysqli_close($dbc);
?>
```

[students.btsi.lu/evegi144/WAD/MySQL/select4\\_escape\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/select4_escape_procedural.php)

```

<?php
require_once 'db_credentials.php';
$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
    mysqli_connect_errno() . ') ' . mysqli_connect_error());
mysqli_set_charset($dbc, 'utf8');

$last_name = "Duck";
$last_name = mysqli_escape_string($dbc, $last_name);
$query = "SELECT dtFirstName, dtLastName FROM tblT1 WHERE dtLastName =
'$last_name'";
$result = mysqli_query($dbc, $query) or die('Wrong SQL: ' . $query . ' Error: ' .
    mysqli_error($dbc));
echo 'Number of rows ' . mysqli_num_rows($result) . '<br>'; # Display the number of
rows.
# Loop through the result object and display each first name.
while ($row = mysqli_fetch_array($result)) printf("%s %s<br>", $row[0], $row[1]);
mysqli_close($dbc);
?>

```

[students.btsi.lu/evegi144/WAD/MySQL/select4\\_procedural.php](http://students.btsi.lu/evegi144/WAD/MySQL/select4_procedural.php)

```

<?php
require_once 'db_credentials.php';
$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
    mysqli_connect_errno() . ') ' . mysqli_connect_error());
mysqli_set_charset($dbc, 'utf8');

$query = 'SELECT dtFirstName, dtLastName FROM tblT1 WHERE dtLastName = ?'; # our
query
$stmt = mysqli_prepare($dbc, $query);
if (!$stmt) die('Wrong SQL: ' . $query . ' Error: ' . mysqli_error($dbc));
echo '<pre>' . print_r($stmt, true) . '</pre>';

$last_name = 'Gates'; # Save the last name that we want to use in our query in a
variable.
# Bind the parameter to our query as a string.
mysqli_stmt_bind_param($stmt, 's', $last_name) or die('Bind failure: ' .
    mysqli_error($dbc));
mysqli_stmt_execute($stmt) or die('Error reading from DB.');
```

[students.btsi.lu/evegi144/WAD/MySQL/select5\\_procedural.php](http://students.btsi.lu/evegi144/WAD/MySQL/select5_procedural.php)

```

<?php
    require_once 'db_credentials.php';
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
        mysqli_connect_errno() . ') ' . mysqli_connect_error());
    mysqli_set_charset($dbc, 'utf8');

    $query = 'SELECT dtFirstName, dtLastName FROM tblT1 WHERE dtLastName = ?'; # our
query
    $stmt = mysqli_prepare($dbc, $query);
    if (!$stmt) die('Wrong SQL: ' . $query . ' Error: ' . mysqli_error($dbc));
    echo '<pre>' . print_r($stmt, true) . '</pre>';

    $last_name = 'Gates'; # Save the last name that we want to use in our query in a
variable.
    # Bind the parameter to our query as a string.
    mysqli_stmt_bind_param($stmt, 's', $last_name) or die('Bind failure: ' .
        mysqli_error($dbc));
    mysqli_stmt_execute($stmt) or die('Error reading from DB. ');
    $result = mysqli_stmt_get_result($stmt);
    while ($row = mysqli_fetch_array($result, MYSQLI_NUM))
        printf("%s %s<br>", $row[0], $row[1]);
    mysqli_stmt_close($stmt);
    mysqli_close($dbc);
?>

```

[students.btsi.lu/evegi144/WAD/MySQL/select6\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/select6_procedural.php)

```

<?php
    require_once 'db_credentials.php';
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or die('Connect Error (' .
        mysqli_connect_errno() . ') ' . mysqli_connect_error());
    mysqli_set_charset($dbc, 'utf8');

    $query = 'SELECT dtFirstName, dtLastName FROM tblT1 WHERE dtLastName = ?';
    $stmt = mysqli_prepare($dbc, $query);
    if (!$stmt) die('Wrong SQL: ' . $query . ' Error: ' . mysqli_error($dbc));
    echo '<pre>' . print_r($stmt, true) . '</pre>';

    $last_name = 'Gates';
    mysqli_stmt_bind_param($stmt, 's', $last_name) or die('Bind failure: ' .
        mysqli_error($dbc));
    mysqli_stmt_execute($stmt) or die('Error reading from DB.');
```

// \$result = mysqli\_stmt\_get\_result(\$stmt);

```

    $result_set = mysqli_stmt_store_result($stmt) or die('Error storing result.');
```

echo 'Number of rows ' . mysqli\_stmt\_num\_rows(\$stmt) . '<br>'; # Display the number of rows

```

    mysqli_stmt_bind_result($stmt, $row[0], $row[1]);
    while (mysqli_stmt_fetch($stmt)) printf("%s %s<br>", $row[0], $row[1]);
    mysqli_stmt_close($stmt);
    mysqli_close($dbc);
?>
```

[students.btsi.lu/evegi144/WAD/MySQL/insert1\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/insert1_procedural.php)

```

<?php
    exit; // to avoid misuse
    require_once 'db_credentials.php';
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or
        die('Connect Error (' . mysqli_connect_errno() . ') ' . mysqli_connect_error());
    mysqli_set_charset($dbc, 'utf8');

    $first_name = 'Bill';
    $last_name = 'Gates';
    $query = "INSERT INTO tblT1 (dtFirstName, dtLastName)
        VALUES ('$first_name', '$last_name')";
    mysqli_query($dbc, $query) or die('Error inserting into DB: ' . mysqli_error($dbc));
    echo "Id of inserted record: " . mysqli_insert_id($dbc);
    mysqli_close($dbc); # Close the DB connection.
?>
```

[students.btsi.lu/evegi144/WAD/MySQL/insert2\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/insert2_procedural.php)

```

<?php
    exit; // to avoid misuse
    require_once 'db_credentials.php';
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or
        die('Connect Error (' . mysqli_connect_errno() . ') ' . mysqli_connect_error());
    mysqli_set_charset($dbc, 'utf8');

    $first_name = 'Bill';
    $last_name = 'Gates';
    $query = 'INSERT INTO tblT1 (dtFirstName, dtLastName) VALUES (?, ?)';
    $stmt = mysqli_prepare($dbc, $query);
    if (!$stmt) die("Wrong SQL: $query Error: mysqli_error($dbc)");
    mysqli_stmt_bind_param($stmt, 'ss', $first_name, $last_name) or
        die('Bind failure: ' . mysqli_error($dbc));
    mysqli_stmt_execute($stmt) or die('Error inserting into DB. ');
    echo "Id of inserted record: " . mysqli_stmt_insert_id($stmt);

    $first_name = 'Linus';
    $last_name = 'Torvalds';
    mysqli_stmt_bind_param($stmt, 'ss', $first_name, $last_name);
    mysqli_stmt_execute($stmt) or die('Error inserting into DB. ');
    echo "Id of inserted record: " . mysqli_stmt_insert_id($stmt);

    mysqli_stmt_close($stmt);
    mysqli_close($dbc);
?>

```

[students.btsi.lu/evegi144/WAD/MySQL/delete1\\_procedural.php](http://students.btsi.lu/evegi144/WAD/MySQL/delete1_procedural.php)

```

<?php
    require_once 'db_credentials.php';
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or
        die('Connect Error (' . mysqli_connect_errno() . ') ' . mysqli_connect_error());
    mysqli_set_charset($dbc, 'utf8');

    $query = 'DELETE FROM tblT1 WHERE dtLastName = "Gates"';
    $result = mysqli_query($dbc, $query);
    if (!$result) die("Wrong SQL: $query Error: mysqli_error($dbc)");
    mysqli_close($dbc);
?>

```

[students.btsi.lu/evegi144/WAD/MySQL/delete2\\_procedural.php](http://students.btsi.lu/evegi144/WAD/MySQL/delete2_procedural.php)

```

<?php
    require_once 'db_credentials.php';
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or
        die('Connect Error (' . mysqli_connect_errno() . ') ' . mysqli_connect_error());
    mysqli_set_charset($dbc, 'utf8');

    $query = 'DELETE FROM tblT1 WHERE dtLastName = ?';
    $stmt = mysqli_prepare($dbc, $query);
    if (!$stmt) die("Wrong SQL: $query Error: mysqli_error($dbc)");
    $last_name = 'Gates';
    mysqli_stmt_bind_param($stmt, 's', $last_name);
    mysqli_stmt_execute($stmt) or die('Error deleting from DB.');
```

```

    mysqli_stmt_close($stmt);
    mysqli_close($dbc);
?>
```

[students.btsi.lu/evegi144/WAD/MySQL/update1\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/update1_procedural.php)

```

<?php
    require_once 'db_credentials.php';
    $dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or
        die('Connect Error (' . mysqli_connect_errno() . ') ' . mysqli_connect_error());
    mysqli_set_charset($dbc, 'utf8');

    $query = 'UPDATE tblT1 SET dtFirstName = "James", dtLastName = "Bond" WHERE ' .
        'dtLastName = "Gates"';
    $result = mysqli_query($dbc, $query);
    if (!$result) die("Wrong SQL: $query Error: " . mysqli_error($dbc));
    mysqli_close($dbc);
?>
```

[students.btsi.lu/evegi144/WAD/MySQL/update2\\_procedural.php](https://students.btsi.lu/evegi144/WAD/MySQL/update2_procedural.php)

```

<?php
require_once 'db_credentials.php';
$dbc = mysqli_connect(DB_HOST, DB_USER, DB_PW, DB_NAME) or
    die('Connect Error (' . mysqli_connect_errno() . ') ' . mysqli_connect_error());
mysqli_set_charset($dbc, 'utf8');

$query = 'UPDATE tblT1 SET dtLastName = ? WHERE dtLastName = ?';
$stmt = mysqli_prepare($dbc, $query);
if (!$stmt) die("Wrong SQL: $query Error: " . mysqli_error($dbc));
$new_last_name = 'Duck';
$old_last_name = 'Bond';
mysqli_stmt_bind_param($stmt, 'ss', $new_last_name, $old_last_name) or
    die('Bind failure: ' . mysqli_error($dbc));
mysqli_stmt_execute($stmt) or die('Error deleting from DB.');
```

```

mysqli_stmt_close($stmt);
mysqli_close($dbc);
?>

```

### 5.3.23. PHP Data Objects (PDO)

From [php.net/manual/en/book.pdo.php](http://php.net/manual/en/book.pdo.php):

The PHP Data Objects (PDO) extension defines a lightweight, consistent interface for accessing databases in PHP. PDO provides a data-access abstraction layer, which means that, regardless of which database you're using, you use the same functions to issue queries and fetch data.

The details of the PDO class can be found at [php.net/manual/en/class.pdo.php](http://php.net/manual/en/class.pdo.php). A good tutorial is at [websitebeaver.com/php-pdo-prepared-statements-to-prevent-sql-injection](http://websitebeaver.com/php-pdo-prepared-statements-to-prevent-sql-injection).

#### Connection

[students.btsi.lu/evegi144/WAD/MySQL/PDO1.php](http://students.btsi.lu/evegi144/WAD/MySQL/PDO1.php)

```

<?php
require_once 'db_credentials.php';
try {
    $dbh = new PDO('mysql:host=' . DB_HOST . ';dbname=' . DB_NAME, DB_USER, DB_PW);
    foreach ($dbh->query('SELECT * FROM tblT1') as $row) {
        echo '<pre>' . print_r($row, true) . '</pre>';
    }
} catch (PDOException $e) {
    print "Error!: " . $e->getMessage() . "<br>";
}
?>

```



It is essential to catch the exception thrown by the PDO constructor otherwise your user name and password are at risk as explained in [php.net/manual/en/pdo.connections.php](http://php.net/manual/en/pdo.connections.php).

## Statement execution

To execute an SQL statement, we can choose between `exec` and `query` or use a prepared statement.

`query` executes an SQL statement in a single function call, returning the result set (if any) returned by the statement as a PDOStatement object.

`exec` also executes an SQL statement in a single function call but returns the number of rows affected by the statement. It does not return results from a SELECT statement.



For `query` and `exec` you need to escape the data inside the query using `quote` if there's any chance it comes from outside, e.g. a user.

For a statement that you need to issue multiple times, prepare a PDOStatement object with `prepare` and issue the statement with `execute`.

From [php.net/manual/en/pdo.quote.php](http://php.net/manual/en/pdo.quote.php):



You are strongly recommended to use `prepare` to prepare SQL statements with bound parameters instead of using `quote` to interpolate user input into an SQL statement. Prepared statements with bound parameters are not only more portable, more convenient, immune to SQL injection, but are often much faster to execute than interpolated queries, as both the server and client side can cache a compiled form of the query.

Here's a simple example: [students.btsi.lu/evegi144/WAD/MySQL/PDO2.php](http://students.btsi.lu/evegi144/WAD/MySQL/PDO2.php)

```
<?php
require_once 'db_credentials.php';
try {
    $dbh = new PDO('mysql:host=' . DB_HOST . ';dbname=' . DB_NAME, DB_USER, DB_PW);
    // https://stackoverflow.com/questions/5741187/sql-injection-that-gets-around-
mysql-real-escape-string
    $dbh->setAttribute(PDO::ATTR_EMULATE_PREPARES, false);
    $sql = 'SELECT dtFirstName, dtLastName FROM tblT1 WHERE dtNumberOfGrades >= ?';
    $sth = $dbh->prepare($sql);
    $sth->execute([12]);
    foreach ($sth->fetchAll() as $row) {
        echo '<pre>' . print_r($row, true) . '</pre>';
    }
} catch (PDOException $e) {
    print "Error!: " . $e->getMessage() . "<br>";
}
?>
```

The predefined PDO constants provided by the MySQL driver can be found at [php.net/manual/en/ref.pdo-mysql.php](http://php.net/manual/en/ref.pdo-mysql.php).

### 5.3.24. Security

[sqlmap.org](http://sqlmap.org)

### 5.3.25. Tests

#### WMOTU Shop DB

Article	Description	Price	
Desktop	Desktop computer	599	<a href="#">delete</a> <a href="#">edit</a>
Mobile	smartphone	299	<a href="#">delete</a> <a href="#">edit</a>

Name:

Description:

Price:

WMOTU Shop DB provides the following functionality (cf. [youtu.be/fpzwKXdBnpc](https://youtu.be/fpzwKXdBnpc)):

1. The user sees all the articles in the database. For each article, the name, description and price are displayed.
2. The user can add a new and delete or edit an existing article.
3. The database is created using a MySQL script, which is located in a protected place, so that the user cannot access it.
4. The user cannot hijack our DB with malignant input.
5. Special characters such as ' and " are handled correctly.

## Solution

The solution can be found at [students.btsi.lu/evegi144/WAD/MySQL/Tests/WMOTUShopDB](https://students.btsi.lu/evegi144/WAD/MySQL/Tests/WMOTUShopDB). To protect the `protected` directory, we proceed as described in [Security](#).

### createDB.sql

```
USE evegi144;

DROP TABLE IF EXISTS WMOTUShopArticle;

CREATE TABLE WMOTUShopArticle (
  idArticle      INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
  dtName         VARCHAR(40) NOT NULL,
  dtDescription  VARCHAR(40) NOT NULL,
  dtPrice        DEC          NOT NULL
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;

INSERT INTO WMOTUShopArticle (dtName, dtDescription, dtPrice)
VALUES ('Car', 'World\'s fastest car', 50000), ('TV', 'Experience 4K resolution',
1000);
```

### database.php

```
<?php
require_once 'db_credentials.php';

class Database {
  private static $DB_HOST;
  private static $DB_USER;
  private static $DB_PASSWORD;
  private static $DB_NAME;
  private static $DB_TABLE = 'WMOTUShopArticle';

  static function set_credentials($db_host, $db_user, $db_password, $db_name) {
    self::$DB_HOST      = $db_host;
    self::$DB_USER      = $db_user;
    self::$DB_PASSWORD  = $db_password;
    self::$DB_NAME      = $db_name;
  }

  static private function connect() {
    $dbc = new mysqli(self::$DB_HOST, self::$DB_USER, self::$DB_PASSWORD,
      self::$DB_NAME);
    if ($dbc->connect_error) trigger_error('Database connection failed: ' .
      $dbc->connect_error, E_USER_ERROR);
  }
}
```

```

        $dbc->set_charset("utf8");
        return $dbc;
    }

    static function get_articles() {
        $dbc    = self::connect();
        $result = FALSE;
        $query   = 'SELECT * FROM ' . self::$DB_TABLE;
        $res      = $dbc->query($query);
        if (!$res) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        while ($dat = $res->fetch_assoc()) $result[] = $dat;
        $dbc->close();
        return $result;
    }

    static function get_article($id) {
        $dbc    = self::connect();
        $query   = 'SELECT * FROM ' . self::$DB_TABLE . " WHERE idArticle = $id";
        $res      = $dbc->query($query);
        if (!$res) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        $result = $res->fetch_assoc();
        $dbc->close();
        return $result;
    }

    static function delete_article($id) {
        $dbc    = self::connect();
        $query   = 'DELETE FROM ' . self::$DB_TABLE . " WHERE idArticle = $id";
        $result = $dbc->query($query);
        if (!$result) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        $dbc->close();
    }

    static function update_article($id, $name, $description, $price) {
        $dbc    = self::connect();
        $query   = 'UPDATE ' . self::$DB_TABLE . ' SET dtName = ?, dtDescription = ?, ' .
            "dtPrice = ? WHERE idArticle = $id";
        $stmt    = $dbc->prepare($query);
        if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        $stmt->bind_param('ssi', $name, $description, $price);
        $result = $stmt->execute();
        $dbc->close();
        return $result;
    }

    static function add_article($name, $description, $price) {
        $dbc    = self::connect();

```

```

$query = 'INSERT INTO ' . self::$DB_TABLE .
        ' (dtName, dtDescription, dtPrice) VALUES (?, ?, ?)';
$stmt = $dbc->prepare($query);
if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
$stmt->bind_param('ssi', $name, $description, $price);
$result = $stmt->execute();
$dbc->close();
return $result;
}
}
?>

```

## index.php

```

<?php
require_once 'protected/database.php';
if (isset($_GET['op']) && isset($_GET['id'])) {
    if ($_GET['op'] === 'del') Database::delete_article($_GET['id']);
    elseif ($_GET['op'] === 'edit') $edit_id = $_GET['id'];
}
elseif (isset($_POST['update']))
    Database::update_article($_POST['id'], $_POST['name'],
        $_POST['description'], $_POST['price']);
elseif (isset($_POST['add']))
    Database::add_article($_POST['name'], $_POST['description'], $_POST['price']);
?>
<!DOCTYPE html>
<html lang=en>
<head>
    <title>WMOTU Shop DB</title>
    <meta charset=UTF-8>
</head>
<body>
    <main>
        <?php
        if (isset($edit_id)) {
            $article = Database::get_article($edit_id);
            ?>
            <form method=post action=index.php>
                <input type=hidden name=id value=<?php echo $edit_id ?>>
                Name: <input name=name autofocus value='<?php echo
                htmlspecialchars($article['name'], ENT_QUOTES, 'UTF-8'); ?>'><br>
                Description: <input name=description value='<?php echo
                htmlspecialchars($article['description'], ENT_QUOTES, 'UTF-8'); ?>'><br>
                Price: <input type=number name=price value='<?php echo
                htmlspecialchars($article['price'], ENT_QUOTES, 'UTF-8'); ?>'>
                <input type=submit value=Update name=update>
            </form>
        <?php

```

```

    }
    else {
        $articles = Database::get_articles();
        $output = '<table><thead><tr><th>Article</th><th>Description</th>' .
            '<th>Price</th><th></th><th></th></tr></thead>';
        if ($articles)
            foreach ($articles as $article) {
                $output .= '<tr><td>' . $article['dtName'] . '</td><td>' .
                    $article['dtDescription'] . '</td><td>' . $article['dtPrice'] .
                    '</td><td><a href="index.php?op=del&id=' . $article['idArticle'] .
                    '">delete</a></td><td><a href="index.php?op=edit&id=' .
                    $article['idArticle'] . '">edit</a></td></tr>';
            }
        echo $output . '</table>';
    }
    ?>
    <form method=post action=index.php>
        Name: <input name=name autofocus><br>
        Description: <input name=description><br>
        Price: <input type=number name=price><br>
        <input type=submit value=Add name=add>
    </form>
    <?php
    }
    ?>
    </main>
</body>
</html>

```

## Perfume Shop

Log out

## Our products

Article	Price	
Eau de SYSEX	€ 99.99	<input type="button" value="buy"/>
Eau de HTSTA	€ 129.99	<input type="button" value="buy"/>
Eau de CLISS	€ 179.99	<input type="button" value="buy"/>
Eau de WSERS	€ 299.99	<input type="button" value="buy"/>
Eau de WEBAP	€ 499.99	<input type="button" value="buy"/>

## Your shopping cart

Article	Quantity	Total price	
Eau de SYSEX	6	€ 599.94	<input type="button" value="remove"/>
Eau de HTSTA	1	€ 129.99	<input type="button" value="remove"/>
Eau de CLISS	3	€ 539.97	<input type="button" value="remove"/>
Eau de WSERS	2	€ 599.98	<input type="button" value="remove"/>
Eau de WEBAP	4	€ 1999.96	<input type="button" value="remove"/>
Total		€ 3869.84	

Develop a perfume shop with the following features:

1. A user DB table to store the user name and encrypted password. There are 2 users (dummy1, dummy1) and (dummy2, dummy2).
2. A perfume DB table to store the perfumes available for sale with the following names and prices (€):
  - Eau de SYSEX 99.99
  - Eau de HTSTA 129.99
  - Eau de CLISS 179.99
  - Eau de WSERS 299.99
  - Eau de WEBAP 499.99
3. A purchase DB table, storing for each purchase a link to the user, a link to the perfume and the quantity.

4. A login (no sign up).
5. Every user has his own shopping cart, which displays the articles bought so far and gives the total price. For each article there may not be more than one entry in the shopping cart, e.g. if the user buys 1 Eau de WSERS and later on buys another 3, then the shopping cart will show 4 Eau de WSERS at €299.99 each for a total of €1199.96.
6. The user can add and remove individual perfumes from his shopping cart. For instance, using the previous example, he can remove 2 Eau de WSERS, which leaves 2 for a total of €599.98.
7. Each user has a logout button.
8. Obviously, if the user logs out and then logs in again, the shopping cart still shows all purchases made.

### **Solution**

The solution can be found at [students.btsi.lu/evegi144/WAD/MySQL/Tests/PerfumeShop](https://students.btsi.lu/evegi144/WAD/MySQL/Tests/PerfumeShop).

createDB.sql

```

DROP TABLE IF EXISTS tblPerfumeShopPurchase;
DROP TABLE IF EXISTS tblPerfumeShopPerfume;
DROP TABLE IF EXISTS tblPerfumeShopUser;

CREATE TABLE tblPerfumeShopUser (
    idUser      INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    dtName      VARCHAR(32)  NOT NULL UNIQUE,
    dtPassword  VARCHAR(255) NOT NULL
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;

CREATE TABLE tblPerfumeShopPerfume (
    idPerfume INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    dtName     VARCHAR(32)  NOT NULL UNIQUE,
    dtPrice    DECIMAL(5, 2) NOT NULL
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;

INSERT INTO tblPerfumeShopPerfume (dtName, dtPrice) VALUES ('Eau de SYSEX', 99.99),
('Eau de HTSTA', 129.99), ('Eau de CLISS', 179.99), ('Eau de WSERS', 299.99),
('Eau de WEBAP', 499.99);

CREATE TABLE tblPerfumeShopPurchase (
    idPurchase INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    fiUser      INT UNSIGNED NOT NULL,
    fiPerfume   INT UNSIGNED NOT NULL,
    FOREIGN KEY (fiUser) REFERENCES tblPerfumeShopUser (idUser)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    FOREIGN KEY (fiPerfume) REFERENCES tblPerfumeShopPerfume (idPerfume)
        ON DELETE CASCADE
        ON UPDATE CASCADE,
    dtQuantity INT UNSIGNED NOT NULL
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;

```

## database.php

```

<?php
require_once 'db_credentials.php';

class Database {
    private static $DB_HOST;

```

```

private static $DB_USER;
private static $DB_PASSWORD;
private static $DB_NAME;

static function set_credentials($db_host, $db_user, $db_password, $db_name) {
    self::$DB_HOST      = $db_host;
    self::$DB_USER      = $db_user;
    self::$DB_PASSWORD  = $db_password;
    self::$DB_NAME      = $db_name;
}

static function connect() {
    $dbc = @new mysqli(self::$DB_HOST, self::$DB_USER, self::$DB_PASSWORD,
self::$DB_NAME);
    if ($dbc->connect_error) error_log('Database connection failed: ' . $dbc-
>connect_error);
    $dbc->set_charset("utf8");
    return $dbc;
}

# Returns user id or FALSE.
static function login($user_name, $password) {
    $dbc = self::connect();
    $query = 'SELECT idUser, dtPassword FROM tblPerfumeShopUser WHERE dtName = "' .
        $dbc->real_escape_string(($user_name)) . '"';
    $result = $dbc->query($query) or trigger_error('Wrong SQL: ' . $query . ' Error:
' .
        $dbc->error, E_USER_ERROR);
    $res = false;
    if ($result && $result->num_rows === 1) {
        $arr = $result->fetch_assoc();
        if (password_verify($password, $arr['dtPassword'])) $res = $arr['idUser'];
        $result->free();
    }
    $dbc->close();
    return $res;
}

# Returns FALSE if user could not be created, otherwise user id.
static function create_user($user_name, $password) {
    $dbc = self::connect();
    $query = 'INSERT INTO tblPerfumeShopUser(dtName, dtPassword) VALUES("' .
        $dbc->real_escape_string($user_name) . '", "' .
        password_hash($password, PASSWORD_DEFAULT) . '")';
    $result = $dbc->query($query) or trigger_error('Wrong SQL: ' . $query . ' Error:
' .
        $dbc->error, E_USER_ERROR);
    $result->free();
    $dbc->close();
    return $result;
}

```

```

# Inserts a purchase, returns nothing.
static function purchase($user_id, $perfume_id) {
    $dbc = self::connect();
    $query = "INSERT INTO tblPerfumeShopPurchase (fiUser, fiPerfume, dtQuantity)
VALUES
    ($user_id, $perfume_id, 1)";
    $result = $dbc->query($query) or die('Error inserting into DB.' . $dbc->error);
}

# Removes a purchase, returns nothing.
static function remove($user_id, $perfume_id) {
    $dbc = self::connect();
    $query = "SELECT idPurchase FROM tblPerfumeShopPurchase WHERE fiUser = $user_id
AND
    fiPerfume = $perfume_id";
    $result = $dbc->query($query) or die('Error inserting into DB.' . $dbc->error);
    if ($result->num_rows > 0) {
        $purchase_id = $result->fetch_row()[0];
        $query = "DELETE FROM tblPerfumeShopPurchase WHERE idPurchase =
$purchase_id";
        $result = $dbc->query($query) or die('Error inserting into DB.' . $dbc->error);
    }
    $dbc->close();
}

# Returns all perfumes.
static function get_perfumes() {
    $dbc = self::connect();
    $query = "SELECT * FROM tblPerfumeShopPerfume";
    $result = $dbc->query($query) or die('Error inserting into DB.' . $dbc->error);
    $perfumes = [];
    while ($perfume = $result->fetch_assoc()) $perfumes[] = $perfume;
    $result->free();
    $dbc->close();
    return $perfumes;
}

# Returns purchases for given user or empty array.
static function get_purchases($user_id) {
    $dbc = self::connect();
    $query = "SELECT idPerfume, dtPrice, dtName, SUM(dtQuantity) as dtQuantity FROM
    tblPerfumeShopPerfume, tblPerfumeShopPurchase WHERE fiUser = $user_id AND
    fiPerfume = idPerfume GROUP BY idPerfume";
    $result = $dbc->query($query) or die('Error inserting into DB.' . $dbc->error);
    $purchases = [];
    while ($purchase = $result->fetch_assoc()) $purchases[] = $purchase;
    $result->free();
    $dbc->close();
    return $purchases;
}

```

```

    }
}

/*Database::create_user('dummy1', 'dummy1');
Database::create_user('dummy2', 'dummy2');*/
?>

```

## index.php

```

<?php
    ini_set('session.cookie_secure', true);
    ini_set('session.use_only_cookies', true);
    ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
    if (!isset($_SERVER['HTTPS']))
        header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
dirname($_SERVER['PHP_SELF']));
    if (!isset($_SESSION)) session_start();
    if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
        session_regenerate_id();
        $_SESSION['generated'] = time();
    }
    if (isset($_SESSION['user_id'])) header('Location: ' . 'https://' .
$_SERVER['HTTP_HOST'] .
    dirname($_SERVER['PHP_SELF']) . '/main.php');
    elseif (isset($_POST['username'], $_POST['password'])) {
        require_once 'database.php';
        $result = Database::login($_POST['username'], $_POST['password']);
        if ($result) {
            $_SESSION['user_id'] = $result;
            $_SESSION['user_name'] = $_POST['username'];
            header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
                dirname($_SERVER['PHP_SELF']) . '/main.php');
        }
    }
?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Perfume Shop</title>
        <meta charset=utf-8>
    </head>
    <body>
        <form method=post>
            <input name=username placeholder="User name" required autofocus>
            <input type=password name=password placeholder=Password required>
            <button>Log in</button>
        </form>
    </body>
</html>

```

## logout.php

```
<?php
    if (!isset($_SESSION)) session_start();
    $_SESSION = [];
    if (session_id() != "" || isset($_COOKIE[session_name()])) setcookie(session_name(),
        '', 1, '/');
    session_destroy();
    header('Location: https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php');
?>
```

## main.php

```
<?php
    ini_set('session.cookie_secure', true);
    ini_set('session.use_only_cookies', true);
    ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
    if (!isset($_SERVER['HTTPS']))
        header('Location: https://' . $_SERVER['HTTP_HOST'] .
            dirname($_SERVER['PHP_SELF']));
    if (!isset($_SESSION)) session_start();
    if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
        session_regenerate_id();
        $_SESSION['generated'] = time();
    }
    if (!isset($_SESSION['user_id'])) header('Location: ' . 'https://' .
$_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php');
    if (!isset($_SESSION['quantities'])) $_SESSION['quantities'] = [0, 0, 0, 0, 0];
    require_once 'database.php';
    $perfumes = Database::get_perfumes();
    if (count($_POST) > 0)
        if (isset($_POST["a"])) Database::purchase($_SESSION['user_id'], $_POST['a']);
        elseif (isset($_POST["r"])) Database::remove($_SESSION['user_id'], $_POST['r']);
?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Perfume Shop</title>
        <meta charset=utf-8>
        <style>
            table {
                border-collapse: collapse;
            }

            th, td {
                border: 3px solid black;
                padding: 5px;
            }
        </style>
    </head>
    <body>
```

```

    }
</style>
</head>
<body>
    <button onclick="location='logout.php'">Log out</button>
    <?php
        $out = '<h1>Our products</h1><form
method=post><table><tr><td>Article</td><td>Price</td>'
        . '<td></td></tr>';
        foreach ($perfumes as $perfume)
            $out .= "<tr><td>{$perfume['dtName']}</td><td>€{$perfume['dtPrice']}</td>
            <td><button name=a value={$perfume['idPerfume']}>buy</button></td>";
        $out .= '</table></form>';
        echo $out;
        $purchases = Database::get_purchases($_SESSION['user_id']);
        if (count($purchases) > 0) {
            $total = 0;
            $out = '<h1>Your shopping cart</h1><form
method=post><table><tr><td>Article</td>'
            . '<td>Quantity</td><td>Total price</td><td></td></tr>';
            foreach ($purchases as $purchase) {
                $total += $purchase['dtQuantity'] * $purchase['dtPrice'];
                $out .=
                "<tr><td>{$purchase['dtName']}</td><td>{$purchase['dtQuantity']}</td><td>€" .
                $purchase['dtQuantity'] * $purchase['dtPrice'] . "</td><td><button
                name=r value={$purchase['idPerfume']}>remove</button></td></tr>";
            }
            $out .= "<tr><td colspan=2>Total</td><td colspan=2>€$total</td></tr></table></form>";
            echo $out;
        }
    ?>
</body>
</html>

```

## MicroQuack

Create a simple Internet message board with the following features:

- A user needs to register first with a user name and password. Users are stored in a user table, which you create via a SQL script. Your app needs to be protected against SQL injection attempts. You may not store plaintext passwords in your DB.
- A logged-in user sees a list of all messages together with their timestamp and author. Messages are stored in a message table, which you create via a SQL script.
- A logged-in user can write a new message or edit or delete one of his previous messages.

## Solution

The solution can be found at [students.btsi.lu/evegi144/WAD/MySQL/Tests/MicroQuack](https://students.btsi.lu/evegi144/WAD/MySQL/Tests/MicroQuack).

## createDB.sql

```
USE evegi144;

DROP TABLE IF EXISTS tblMessage;
DROP TABLE IF EXISTS tblUser;
CREATE TABLE tblUser (
    idUser      INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
    dtUserName  VARCHAR(50)  NOT NULL UNIQUE,
    dtPassword  VARCHAR(255) NOT NULL
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;

CREATE TABLE tblMessage (
    idMessage   INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
    dtMessage   TEXT          NOT NULL,
    dtTimestamp TIMESTAMP      DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,
    fiUser      INT UNSIGNED NOT NULL,
    FOREIGN KEY (fiUser) REFERENCES tblUser (idUser)
    ON DELETE CASCADE
    ON UPDATE CASCADE
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;
```

## main.php

```
<?php
    ini_set('session.cookie_secure', true);
    ini_set('session.use_only_cookies', true);
    ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
    if (!isset($_SERVER['HTTPS']))
        header('Location: https://' . $_SERVER['HTTP_HOST'] .
dirname($_SERVER['PHP_SELF']));
    if (!isset($_SESSION)) session_start();
    if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
        session_regenerate_id();
        $_SESSION['generated'] = time();
    }
    require_once 'db_credentials.php';
    $dbc = new mysqli(DB_HOST, DB_USER, DB_PW, DB_NAME) or
        die("Connect error: $dbc->connect_error");
    $dbc->set_charset('utf8');
?>
<!DOCTYPE html>
```

```

<html lang=en>
<head>
  <meta charset=UTF-8>
  <title>MicroQuack</title>
</head>
<body>
  <?php
    if (!isset($_SESSION['user_id'])) {
      if (isset($_POST['login'])) {
        $un = $dbc->real_escape_string($_POST['user']);
        $query = "SELECT * FROM tblUser WHERE dtUserName = '$un'";
        $result = $dbc->query($query) or die('SQL error: ' . $dbc->error);
        if ($result->num_rows != 1) echo 'Invalid login!';
        else {
          $row = $result->fetch_assoc();
          if (password_verify($_POST['password'], $row['dtPassword'])) {
            $_SESSION['user_id'] = $row['idUser'];
            header('Location: index.php');
          } else echo 'Wrong password!';
        }
      } elseif (isset($_POST['register'])) {
        $un = $dbc->real_escape_string($_POST['user']);
        $query = "SELECT * FROM tblUser WHERE dtUserName = '$un'";
        $result = $dbc->query($query) or die('SQL error: ' . $dbc->error);
        if ($result->num_rows > 0) echo 'User name is already taken!';
        $pw = password_hash($_POST['password'], PASSWORD_DEFAULT);
        $query = "INSERT INTO tblUser(dtUserName, dtPassword) VALUES('$un', '$pw')";
        $dbc->query($query) or die('SQL error: ' . $dbc->error);
      }
      echo <<<EOT
        <form method=post>
          <input placeholder=user name=user required>
          <input type=password placeholder=password name=password required>
          <button name=login>Login</button>
          <button name=register>Register</button>
        </form>
      EOT;
    } elseif (isset($_POST['logout'])) {
      $_SESSION = [];
      if (session_id() != "" || isset($_COOKIE[session_name()]))
        setcookie(session_name(), '', 1,
rawurlencode(dirname($_SERVER['PHP_SELF'])));
      session_destroy();
      header('Location: index.php');
    } elseif (isset($_POST['save'])) {
      $msg = $dbc->real_escape_string($_POST['message']);
      $query = "INSERT INTO tblMessage(dtMessage, fiUser) VALUES('$msg',
        {$_SESSION['user_id']})";
      $dbc->query($query) or die('SQL error: ' . $dbc->error);
      header('Location: index.php');
    } elseif (isset($_POST['delete'])) {

```

```

$id = $_POST['delete'];
$query = "DELETE FROM tblMessage WHERE idMessage=$id";
$dbc->query($query) or die('SQL error: ' . $dbc->error);
header('Location: index.php');
} elseif (isset($_POST['edit'])) {
    $id = $_POST['edit'];
    $query = "SELECT dtMessage FROM tblMessage WHERE idMessage=$id";
    $result = $dbc->query($query) or die('SQL error: ' . $dbc->error);
    $msg = $result->fetch_assoc();
    echo <<<EOT
        <form method=post>
            <textarea name=message required>{$msg['dtMessage']}</textarea>
            <button name=update value=$id>Update</button>
        </form>
EOT;
} elseif (isset($_POST['update'])) {
    $id = $_POST['update'];
    $msg = $dbc->real_escape_string($_POST['message']);
    $query = "UPDATE tblMessage SET dtMessage='$msg' WHERE idMessage=$id";
    $dbc->query($query) or die('SQL error: ' . $dbc->error);
    header('Location: index.php');
} else {
    echo <<<EOT
        <form method=post>
            <button name=logout>Logout</button>
        </form>
EOT;

    $query = 'SELECT * FROM tblMessage';
    $result = $dbc->query($query) or die('SQL error: ' . $dbc->error);
    if ($result->num_rows > 0) {
        $msg_table =
'<table><tr><th>Id</th><th>Message</th><th>User</th><th>Timestamp</th>';
        $msg_table .= '<th></th><th></th></tr>';
        while ($msg = $result->fetch_assoc()) {
            $query = "SELECT dtUserName FROM tblUser WHERE idUser = {$msg['fiUser']}";
            $res = $dbc->query($query) or die('SQL error: ' . $dbc->error);
            $user = $res->fetch_assoc();
            //$messages[] = $msg;
            $msg_table .= "<tr><td>{$msg['idMessage']}</td>";
            $msg_table .= "<td><textarea
readonly>{$msg['dtMessage']}</textarea></td>";
            $msg_table .= "<td>{$user['dtUserName']}</td>";
            $msg_table .= "<td>{$msg['dtTimestamp']}</td>";
            $msg_table .= '<td>';
            if ($SESSION['user_id'] === $msg['fiUser'])
                $msg_table .= '<form method=post><button name=delete value=' .
                    $msg['idMessage'] . '>delete</button></form>';
            $msg_table .= '</td><td>';
            if ($SESSION['user_id'] === $msg['fiUser'])
                $msg_table .= '<form method=post><button name=edit value=' .
                    $msg['idMessage'] . '>edit</button></form>';

```

```

        $msg_table .= '</td></tr>';
    }
    echo $msg_table . '</table>';
}
echo <<<EOT
<form method=post>
    <textarea name=message required></textarea>
    <button name=save>Save</button>
</form>
EOT;
    }
    ?>
</body>
</html>

```

## 5.4. Node.js

From [nodejs.org](https://nodejs.org):

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine. Node.js uses an event-driven, non-blocking I/O model that makes it lightweight and efficient. Node.js' package ecosystem, npm, is the largest ecosystem of open source libraries in the world.

You can play with Node right in your browser using [npm.runkit.com](https://npm.runkit.com).

Note that even though Node is single threaded, you can still have race conditions as is the case with JavaScript too, see [medium.com/@slavik57/async-race-conditions-in-javascript-526f6ed80665](https://medium.com/@slavik57/async-race-conditions-in-javascript-526f6ed80665).

### 5.4.1. Installation

Installation on [Ubuntu](#) is easy.

On [Windows](#) you need an [environment variable](#). [Configuring your Windows development environment](#). For Raspberry Pi see [linux.tips/tutorials/how-to-install-latest-version-of-node-js-on-raspberry-pi-3](https://linux.tips/tutorials/how-to-install-latest-version-of-node-js-on-raspberry-pi-3).

Verify that the `NODE_PATH` environment variable is set, otherwise Node might not find modules:

```
find /etc -type f -exec grep -F NODE_PATH {} +
```

If it is not set, create the file `nodejs.sh` in `/etc/profile.d` with this content:

```
NODE_PATH=/usr/lib/nodejs:/usr/lib/node_modules:/usr/share/javascript
export NODE_PATH
```

If required uncomment the respective line in the `nodejs.sh` file. Then you need to login again in

order for the variable to be set.

## 5.4.2. NPM

**NPM** is the Node package manager. Study [docs.npmjs.com/cli/install](https://docs.npmjs.com/cli/install) to get started.

Make sure that `/usr/lib/node_modules` and its subfolders are readable and accessible by everyone. In order to achieve this, run `umask 022` before running npm installation.

The current NPM version installs modules as belonging to user `root` but older versions used user `nobody` (cf. [stackoverflow.com/questions/44633419/no-access-permission-error-with-npm-global-install-on-docker-image](https://stackoverflow.com/questions/44633419/no-access-permission-error-with-npm-global-install-on-docker-image)).

To install a package globally, use `-g`.

To upgrade NPM, run:

```
npm i -g npm@latest
```

To see a list of all outdated packages:

```
npm outdated -g ---depth=0
```

To update all global packages the documentation claims you can use:

```
npm up -g
```

However, this does not work. See [gist.github.com/othiym23/4ac31155da23962afd0e](https://gist.github.com/othiym23/4ac31155da23962afd0e) for a solution.

To determine where global modules are installed:

```
npm root -g
```

To determine the version of a globally installed module:

```
npm list -g <module>
```

To uninstall npm use `npm uninstall npm -g` (cf. [askubuntu.com/questions/873729/how-to-uninstall-corrupted-npm](https://askubuntu.com/questions/873729/how-to-uninstall-corrupted-npm)).

If you get `gyp WARN EACCES user "root" does not have permission to access the dev dir` or something similar when trying to install a module, check [github.com/GoogleChrome/puppeteer/issues/375](https://github.com/GoogleChrome/puppeteer/issues/375). This may help:

```
npm i -g <module> --unsafe-perm=true
```

### 5.4.3. Node Version Manager (NVM)

[github.com/nvm-sh/nvm](https://github.com/nvm-sh/nvm)

[www.sitepoint.com/quick-tip-multiple-versions-node-nvm](https://www.sitepoint.com/quick-tip-multiple-versions-node-nvm)

### 5.4.4. Introduction

The easiest way to get started with Node.js might be to follow a tutorial, such as [www.youtube.com/watch?v=TLB\\_eWDSMt4](https://www.youtube.com/watch?v=TLB_eWDSMt4), [The Art of Node](#), [stream-handbook](#), [NodeSchool](#), [ultra fast apps using Node.js](#) or [learnyounode](#) and play around on [RequireBin](#).

Next you should study the documentation at [Node.js](#) and some of the free books such as [Programming JavaScript Applications](#) or others to be found at [JSbooks](#). If you prefer German, have a look at [NodeCode](#).

You can also take an online course and/or certification, see for example [digitaldefynd.com/best-nodejs-courses-class-certification-online](https://digitaldefynd.com/best-nodejs-courses-class-certification-online).

Avoid running Node apps as root: [syskall.com/dont-run-node-dot-js-as-root](https://syskall.com/dont-run-node-dot-js-as-root).

For debugging you can dump an object at unlimited depth using `console.log(util.inspect(result, false, null));`.

From [expressjs.com/en/advanced/best-practice-performance.html#use-a-reverse-proxy](https://expressjs.com/en/advanced/best-practice-performance.html#use-a-reverse-proxy):

Synchronous functions and methods tie up the executing process until they return. A single call to a synchronous function might return in a few microseconds or milliseconds, however in high-traffic websites, these calls add up and reduce the performance of the app. Avoid their use in production.

Although Node and many modules provide synchronous and asynchronous versions of their functions, always use the asynchronous version in production. The only time when a synchronous function can be justified is upon initial startup.

A fantastic site to find open source Node projects is [bestofjs.org](https://bestofjs.org).

### Streams

Streams are a key feature of Node.js and very well explained in [medium.freecodecamp.org/node-js-streams-everything-you-need-to-know-c9141306be93](https://medium.freecodecamp.org/node-js-streams-everything-you-need-to-know-c9141306be93).

Here is an example of a transform stream that reads an HTML file and replaces the string `{replace}`

with **Godzilla**.

**streams1.js**

```
'use strict';

const fs = require('fs');
const rs = fs.createReadStream('streamtest.html');
const ws = fs.createWriteStream('streamtestout.html');
const { Transform } = require('stream');

const transform = new Transform({
  transform(chunk, encoding, callback) {
    this.push(chunk.toString().replace(/{\replace}/g, 'Godzilla'));
    callback();
  }
});

rs.on('error', err => res.end('Could not read file!'));
rs.pipe(transform).pipe(ws);
```

**streamtest.html**

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Hello {\replace}</title>
  </head>
  <body>
    <h1>Hello {\replace}</h1>
  </body>
</html>
```

**streamtestout.html**

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Hello Godzilla</title>
  </head>
  <body>
    <h1>Hello Godzilla</h1>
  </body>
</html>
```

## Hello world web server

The following is a very basic example of an HTTP web server that reads an HTML file from the server and sends it as a response to each client request.



Remember that ports 0-1024 are designated as [well-known ports](#) and reserved for privileged services. So please choose a port number larger than 1024 otherwise you'll need root permission to listen on that port.

helloworld\_web\_server.js

```

'use strict';

// curl -k https://localhost:9000
const http = require('http'); // include HTTP module
const fs = require('fs'); // include File System module
const port = 9000; // Port that our server will be listening on. It needs to be opened
on the
// firewall.

const server = http.createServer((req, res) => {
  const rs = fs.createReadStream('helloworld.html'); // Read the page to be sent to
the
  // browser.
  rs.on('error', err => res.end('Could not read file!'));
  rs.pipe(res, {
    end: false
  }); // Send it as part of the server response.
  rs.on('end', () => { // Send some additional info to the client.
    res.write('That\'s all folks...');
    res.end(); // Do not forget to end the response.
  });
  /*res.writeHead(200);
  console.log(req.headers);
  res.end('hello world\n');*/
});

server.on('listening', () => {
  console.log(`Listening on port ${port}...`);
});

server.on('connection', () => {
  console.log('New connection...');
});

server.on('close', () => {
  console.log('Connection closed...');
});

server.listen(port, err => {
});

```

Here is the same example using HTTPS. Note that it needs to be run with root permissions, preferably using `sudo`, to access the private key.

```

'use strict';

// curl -k https://localhost:9000
const https = require('https'); // include HTTPS module
const fs = require('fs'); // include File System module
const port = 9000; // Port that our server will be listening on. It needs to be opened
on the firewall.
const options = { // Paths to our private key and certificate.
  key: fs.readFileSync('/etc/letsencrypt/live/students.btsi.lu/privkey.pem'),
  cert: fs.readFileSync('/etc/letsencrypt/live/students.btsi.lu/cert.pem')
};

const server = https.createServer(options, (req, res) => {
  const rs = fs.createReadStream('helloworld.html'); // Read the page to be sent to
the
  rs.on('error', err => res.end('Could not read file!'));
  // browser.
  rs.pipe(res, {
    end: false
  }); // Send it as part of the server response.
  rs.on('end', () => { // Send some additional info to the client.
    res.write('That\'s all folks...');
    res.end(); // Do not forget to end the response.
  });
  /*res.writeHead(200);
  console.log(req.headers);
  res.end('hello world\n');*/
});

server.on('listening', () => {
  console.log(`Listening on port ${port}...`);
});

server.on('connection', () => {
  console.log('New connection...');
});

server.on('close', () => {
  console.log('Connection closed...');
});

server.listen(port, err => { //http://syskall.com/dont-run-node-dot-js-as-root/
  let uid = parseInt(process.env.SUDO_UID);
  if (uid) process.setuid(uid); // Set our server's uid to that user.
  console.log('Server\'s UID is now ' + process.getuid());
}
);

```

## AJAX and JSON

Let's see how we can exchange data via JSON between client and server:

HTTPServer1.js

```
'use strict';

const fs = require('fs');
const http = require('http'); // Include the HTTP module
const server = http.createServer(); // Create our HTTP server which is an event emitter.
const port = 9000; // The port number that our HTTP server will be listening on.

// Add an event listener for the request event.
server.on('request', (req, res) => {
  const {method, url, headers} = req; // All headers are represented in lower-case only.
  console.log(method);
  console.log(url);
  console.dir(headers);
  console.log(req.headers['content-type']);
  if (req.method === 'POST') {
    if (req.headers['content-type'] === 'application/json') {
      let data = '';
      req.on('error', err => {
        console.error(err.stack);
      }).on('data', chunk => {
        data += chunk;
      }).on('end', () => {
        if (data) {
          try {
            data = JSON.parse(data);
          } catch (e) {
            console.error(e);
          }
        }
        console.dir(data);
        res.on('error', err => {
          console.error(err);
        });
        let login = false;
        if (data.un === 'a1' && data.pw === 'a1') login = true;
        res.json(JSON.stringify({login: login}));
      });
    }
  } else {
    console.dir(req.headers);
    res.on('error', err => {
      console.error(err);
    });
  }
});
```

```
    let fileName = 'HTTPClient1.html';
    if (req.url === '/HTTPClient1.css') fileName = 'HTTPClient1.css';
    const rs = fs.createReadStream(fileName);
    rs.on('error', err => res.end('Could not read file!'));
    rs.pipe(res);
  }
});

server.listen(port); // Let the server listen for HTTP requests.
server.on('error', err => {
  console.error('Something went wrong...');
});
```

HTTPClient1.html

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>HTTP Client 1</title>
    <link href=node/HTTPClient1.css rel=stylesheet>
    <script>
      'use strict';

      const init = () => {
        const button = document.querySelector('button');
        button.addEventListener('click', () => {
          const inputs = document.querySelectorAll('input');
          if (inputs[0].value && inputs[1].value) {
            fetch('', {
              headers: {
                'Accept': 'application/json',
                'Content-Type': 'application/json'
              },
              method: "POST",
              credentials: 'same-origin', //
https://github.com/expressjs/session/issues/374
              body: JSON.stringify({un: inputs[0].value, pw: inputs[1].value})
            }).then(response => response.json()).then(data => {
              console.dir(data);
            }).catch(error => {
              console.log(`There has been a problem with the fetch operation:
${error.message}`);
            });
          }
        });
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <input name=user placeholder=username required>
    <input type=password name=pw placeholder=password required>
    <button>Login</button>
  </body>
</html>

```

## Error handling

A must read is located at [expressjs.com/en/advanced/best-practice-performance.html#use-try-catch](https://expressjs.com/en/advanced/best-practice-performance.html#use-try-catch).

For stream error handling see [stackoverflow.com/questions/21771220/error-handling-with-node-js-streams](https://stackoverflow.com/questions/21771220/error-handling-with-node-js-streams).

## Requiring modules

[medium.freecodecamp.org/requiring-modules-in-node-js-everything-you-need-to-know-e7fbd119be8](https://medium.freecodecamp.org/requiring-modules-in-node-js-everything-you-need-to-know-e7fbd119be8)

`__dirname`

[www.hostingadvice.com/how-to/nodejs\\_\\_dirname](http://www.hostingadvice.com/how-to/nodejs__dirname)

## 5.4.5. WebSockets

[WebSocketAPI](#)

[WebSocket protocol](#)

[WebSocket protocol name registry](#)

[Overview of the main WebSocket packages for Node](#)

At [javascript.info/websocket](http://javascript.info/websocket) is a great introductory article.

[WebSocket monitor](#) is a very useful Firefox extension to monitor WebSockets but not yet updated for current Firefox versions.

[samsaffron.com/archive/2015/12/29/websockets-caution-required](http://samsaffron.com/archive/2015/12/29/websockets-caution-required)

[www.terlici.com/2015/11/26/realtime-node-expressjs-with-websockets.html](http://www.terlici.com/2015/11/26/realtime-node-expressjs-with-websockets.html)

[banksco.de/p/state-of-realtime-web-2016.html](http://banksco.de/p/state-of-realtime-web-2016.html)

[www.dmuth.org/node/1463/extracting-session-ids-websocket-requests-expressjs](http://www.dmuth.org/node/1463/extracting-session-ids-websocket-requests-expressjs)

[deepstream.io/blog/realtime-framework-overview](http://deepstream.io/blog/realtime-framework-overview)

[blog.sessionstack.com/how-javascript-works-deep-dive-into-websockets-and-http-2-with-sse-how-to-pick-the-right-path-584e6b8e3bf7](http://blog.sessionstack.com/how-javascript-works-deep-dive-into-websockets-and-http-2-with-sse-how-to-pick-the-right-path-584e6b8e3bf7)

[hackernoon.com/implementing-a-websocket-server-with-node-js-d9b78ec5ffa8](http://hackernoon.com/implementing-a-websocket-server-with-node-js-d9b78ec5ffa8) provides a deep dive into the details of WebSockets and how to implement them yourself in Node.

[blog.securityevaluators.com/websockets-not-bound-by-cors-does-this-mean-2e7819374acc](http://blog.securityevaluators.com/websockets-not-bound-by-cors-does-this-mean-2e7819374acc)

[www.freecodecamp.org/news/how-to-secure-your-websocket-connections-d0be0996c556](http://www.freecodecamp.org/news/how-to-secure-your-websocket-connections-d0be0996c556)

## ws

From [www.npmjs.com/package/ws](http://www.npmjs.com/package/ws):

ws is a simple to use, blazing fast, and thoroughly tested WebSocket client and server implementation.

In order to use WebSocket over TLS we need to create a HTTPS-Server first and use it to create the WSS server. This is so because the HTTPS server takes care of the connection encryption, which takes place before the upgrade request to the WebSocket protocol is received from the client (cf. [www.giacomovacca.com/2015/02/websockets-over-nodejs-from-plain-to.html](http://www.giacomovacca.com/2015/02/websockets-over-nodejs-from-plain-to.html)). The example at [github.com/websockets/ws/blob/master/examples/ssl.js](https://github.com/websockets/ws/blob/master/examples/ssl.js) shows how it can be done.

To have ws reconnect automatically, take a look at [github.com/joewalnes/reconnecting-websocket](https://github.com/joewalnes/reconnecting-websocket).

Let's take a look at a basic chat app:

wsServer1.js

```
'use strict';

const DEBUG = true;
const port = 9000;
const WebSocketServer = require('ws').Server;
const wss = new WebSocketServer({port: port});

wss.on('error', err => {
  console.dir(err);
});

wss.on('connection', (socket, req) => {
  console.log('Client connected...');

  socket.on('error', err => {
    console.dir(err);
  });

  socket.on('message', data => {
    if (DEBUG) console.log(data);
    socket.send(`You said: ${data}<br>`);
  });

  socket.on('close', () => {
    console.log('Socket closed');
  });
});

wss.on('listening', () => {
  console.log('Listening...');
});
```

wsClient1.html

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>ws Client 1</title>
    <script>
      'use strict';

      const init = () => {
        const DEBUG = true;
        const input = document.querySelector('input');
        const button = document.querySelector('button');
        const section = document.querySelector('section');
        const wssURL = 'wss://students.btsi.lu/node/wss';
        const wss = new WebSocket(wssURL);
        wss.addEventListener('open', () => {
          if (DEBUG) console.log('WebSocket connection opened.');
```

```

          button.addEventListener('click', () => {
            wss.send(input.value);
          });
        });

        wss.addEventListener('message', e => {
          section.innerHTML += e.data;
        });

        wss.addEventListener('close', () => {
          if (DEBUG) console.log('Disconnected...');
        });

        wss.addEventListener('error', () => {
          if (DEBUG) console.log('Error...');
        });
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <input placeholder=message>
    <button>Send</button><br>
    <section></section>
  </body>
</html>

```

Here's a WS server that understands the command `getRandomNums` with a given number and will reply with as many random numbers as requested. Note that the communication between client and server is handled using JSON:



```

'use strict';

const DEBUG = true;
const port = 9000;
const WebSocketServer = require('ws').Server;
const wss = new WebSocketServer({port: port});

wss.on('error', err => {
  console.dir(err);
});

wss.on('connection', (socket, req) => {
  console.log('Client connected...');

  // Generic send function taking a message string and an options object.
  const send = (msg, options) => {
    const obj = {};
    Object.assign(obj, {msg: msg}, options);
    socket.send(JSON.stringify(obj));
    if (DEBUG) console.log('WS sent:');
    if (DEBUG) console.log(obj);
  };

  socket.on('error', err => {
    console.dir(err);
  });

  socket.on('message', data => {
    try { // Remember that JSON.parse will fail if data is not valid JSON.
      data = JSON.parse(data);
      if (DEBUG) console.dir(data);
      if (data && data.command && data.command === 'getRandomNums' && data.num) {
        const nums = [];
        for (let i = 0; i < data.num; i++) nums.push(Math.random());
        send('randomNums', {nums: nums});
      }
    } catch(error) {
      console.error(error);
    }
  });
});

socket.on('close', () => {
  console.log('Socket closed');
});

});

wss.on('listening', () => {
  console.log('Listening...');
});

```

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>ws Client 2</title>
    <script>
      'use strict';

      const DEBUG = true;
      const wssURL = 'wss://students.btsi.lu/node/wss';
      const wss = new WebSocket(wssURL);
      const messageCallbacks = [];

      // Generic function to register a callback for a given message.
      const registerMessageCB = (msg, cb) => {
        messageCallbacks.push({msg: msg, cb: cb});
      };

      const displayNums = nums => {
        // Cf. https://exploringjs.com/es6/ch\_for-of.html#sec\_pitfall-for-of-iterability
        for (const num of Array.from(nums)) console.log(num);
      };

      const execute = (command, options) => {
        let obj = {};
        // Build an object with the given fields.
        Object.assign(obj, {command: command}, options);
        wss.send(JSON.stringify(obj));
        if (DEBUG) {
          console.log('WS sent:');
          console.dir(obj);
        }
      };

      wss.addEventListener('open', () => {
        if (DEBUG) console.log('WebSocket connection opened.');
```

// Tell the server to send us 5 random numbers asynchronously.

```

        execute('getRandomNums', {num: 5});
      });

      wss.addEventListener('message', e => {
        try { // Remember that JSON.parse will fail if data is not valid JSON.
          const data = JSON.parse(e.data);
          if (DEBUG) {
            console.log('WS client received the following message:');
            console.dir(data);
          }
          if (!data.msg) return;

```

```

        // Let's see if we have one or more callbacks for this message.
        for (const cb of messageCallbacks) if (cb.msg === data.msg) cb.cb(data);
    } catch(error) {
        console.error(error);
    }
});

wss.addListener('close', () => {
    if (DEBUG) console.log('Disconnected...');
});

wss.addListener('error', () => {
    if (DEBUG) console.log('Error...');
});

// If the server sends a 'randomNums' msg we want 'displayNums' to be called.
registerMessageCB('randomNums', displayNums);
</script>
</head>
<body></body>
</html>

```

Now let's look at WS communication using promises. The server remains unchanged. On the client side however, we need to promisify our execute function. To do this we need to store the resolve function of the promise as the resolve function will only be called after the answer from the server has been received (see [stackoverflow.com/questions/26150232/resolve-javascript-promise-outside-function-scope](https://stackoverflow.com/questions/26150232/resolve-javascript-promise-outside-function-scope)):

**wsClient3.html**

```

<!DOCTYPE html>
<html lang=en>
<head>
    <meta charset=UTF-8>
    <title>ws Client 3</title>
    <script>
        'use strict';

        const DEBUG = true;
        const wssURL = 'wss://students.btsi.lu/node/wss';
        const wss = new WebSocket(wssURL);
        const promises = [];

        wss.addListener('open', () => {
            if (DEBUG) console.log('WebSocket connection opened.');

```

```

// Send a command with options to the server and wait for a given reply msg.
const execute = (command, options, msg) => {
  return new Promise((resolve, reject) => {
    promises.push({msg: msg, resolve: resolve});
    if (DEBUG) console.log('Promise added');
    let obj = {};
    Object.assign(obj, {command: command}, options);
    wss.send(JSON.stringify(obj)); // Send command to server.
    if (DEBUG) {
      console.log('WS sent:');
      console.dir(obj);
    }
  });
};

wss.addEventListener('message', e => {
  try { // Remember that JSON.parse will fail if data is not valid JSON.
    const data = JSON.parse(e.data);
    if (DEBUG) {
      console.log('WS client received the following message:');
      console.dir(data);
    }
    if (!data.msg) return;
    // Let's see if we have one or more promises for this message.
    for (const [index, promise] of promises.entries()) if (promise.msg ===
data.msg) {
      promise.resolve(data); // Fulfil the promise.
      promises.splice(index, 1); // Delete the promise from the waiting list.
    }
  } catch (error) {
    console.error(error);
  }
});

wss.addEventListener('close', () => {
  if (DEBUG) console.log('Disconnected...');
});

wss.addEventListener('error', () => {
  if (DEBUG) console.log('Error...');
});

// Tell the server to send us 5 random numbers and wait for the result.
execute('getRandomNums', {num: 5}, 'randomNums').then(data => {
  displayNums(data.nums);
});
if (DEBUG) console.log('Done!');
});
</script>
</head>
<body></body>

```

```
</html>
```

We can also use `async` and `await`, like so:

`wsClient4.html`

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>ws Client 4</title>
    <script>
      'use strict';

      const DEBUG = true;
      const wssURL = 'wss://students.btsi.lu/node/wss';
      const wss = new WebSocket(wssURL);
      const promises = [];

      wss.addEventListener('open', () => {
        if (DEBUG) console.log('WebSocket connection opened.');
```

```

    for (const [index, promise] of promises.entries()) if (promise.msg ===
data.msg) {
        promise.resolve(data); // Fulfil the promise.
        promises.splice(index, 1); // Delete the promise from the waiting list.
    }
    } catch (error) {
        console.error(error);
    }
});

wss.addListener('close', () => {
    if (DEBUG) console.log('Disconnected...');
});

wss.addListener('error', () => {
    if (DEBUG) console.log('Error...');
});

// Tell the server to send us 5 random numbers and wait for the result.
(async () => {
    const obj = await execute('getRandomNums', {num: 5}, 'randomNums');
    displayNums(obj.nums);
})();
if (DEBUG) console.log('Done!');
})();
</script>
</head>
<body></body>
</html>

```

### Displaying images received from the server

Displaying images from the server can be done as follows, assuming that **data** contains the binary data obtained from the server:

```

const image = new Image();
const buffer = Uint8Array.from(data);
const blob = new Blob([buffer]);
const URL = window.URL || window.webkitURL;
// https://developer.mozilla.org/en-US/docs/Web/API/URL/createObjectURL
image.src = URL.createObjectURL(blob);
document.querySelector('body').appendChild(image);

```

### Sending binary data embedded in JSON to the server

Blob, ArrayBuffer, File etc. are not part of JSON. So the tricky part is to read the image file as ArrayBuffer and convert it into an array that can then be stringified. On the server side we then need to turn the array into a buffer which we can write to disc.

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>ws Image Client 1</title>
    <script>
      'use strict';

      const init = () => {
        const DEBUG = true;
        const input = document.querySelector('input');
        const wssURL = 'wss://students.btsi.lu/node/wss';
        const wss = new WebSocket(wssURL);
        wss.addEventListener('open', e => {
          if (DEBUG) console.log('WebSocket connection opened.');
          input.addEventListener('change', e => {
            const files = e.target.files;
            if (DEBUG) console.dir(files);
            const isValidType = type => {
              const validTypes = ['image/jpeg', 'image/jpg', 'image/png', 'image/gif',
                'image/bmp'];
              for (const validType of validTypes) if (type === validType) return true;
              return false;
            };
            for (const file of files) {
              if (isValidType(file.type)) {
                if (DEBUG) console.log('Valid type: ' + file.type);
                let fr = new FileReader();
                fr.onload = e => {
                  const array = Array.from(new Uint8Array(e.target.result));
                  wss.send(JSON.stringify({filename: file.name, data: array}));
                };
                fr.readAsArrayBuffer(file);
              }
              else alert('Only the following file types are supported: jpeg/jpg, png,
gif and' +
                ' bmp');
            }
          });
        });

        wss.addEventListener('message', e => {
          section.innerHTML += e.data;
        });

        wss.addEventListener('close', () => {
          if (DEBUG) console.log('Disconnected...');
        });

        wss.addEventListener('error', () => {
          if (DEBUG) console.log('Error...');
        });
      };
    </script>
  </head>
  <body>
    <div>
      <input type="file"/>
    </div>
  </body>
</html>

```

```
    });  
  };  
  
  addEventListener('load', init);  
</script>  
</head>  
<body>  
  <input type=file multiple>  
</body>  
</html>
```

```

'use strict';

const DEBUG = true;
const port = 9000;
const WebSocketServer = require('ws').Server;
const wss = new WebSocketServer({port: port});
const fs = require('fs');

wss.on('error', err => {
  console.dir(err);
});

wss.on('connection', (socket, req) => {
  if (DEBUG) console.log('Client connected...');

  // Generic send function taking a message string and an options object.
  const send = (msg, options) => {
    const obj = {};
    Object.assign(obj, {msg: msg}, options);
    socket.send(JSON.stringify(obj));
    if (DEBUG) console.log('WS sent:');
    if (DEBUG) console.log(obj);
  };

  socket.on('error', err => {
    console.dir(err);
  });

  socket.on('message', async data => {
    try { // Remember that JSON.parse will fail if data is not valid JSON.
      data = JSON.parse(data);
      if (DEBUG) console.dir(data);
      fs.writeFile(data.filename, Buffer.from(data.data), err => {
        if (err) console.dir(err);
      });
    } catch(error) {
      console.error(error);
    }
  });

  socket.on('close', () => {
    if (DEBUG) console.log('Socket closed');
  });
});

wss.on('listening', () => {
  if (DEBUG) console.log('Listening...');
});

```

## uws

uws ([www.npmjs.com/package/uws](http://www.npmjs.com/package/uws)) offers the same functionality as ws but is much faster. See [hackernoon.com/%C2%B5ws-as-your-next-websocket-library-d34209686357](http://hackernoon.com/%C2%B5ws-as-your-next-websocket-library-d34209686357) for an illustration of the difference it can make.

## Socket.io

[github.com/socketio/socket.io](https://github.com/socketio/socket.io) or [www.npmjs.com/package/socket.io](http://www.npmjs.com/package/socket.io) enables real-time bidirectional event-based communication. It works on every platform, browser or device, focusing equally on reliability and speed.

[www.tutorialspoint.com/socket.io/socket.io\\_overview.htm](http://www.tutorialspoint.com/socket.io/socket.io_overview.htm)

File upload: [github.com/vote539/socketio-file-upload](https://github.com/vote539/socketio-file-upload)



1. In order for the client to access the `socket.io.js` file, you need to specify the correct path to it. If you installed Socket.io locally using npm it will be `/socket.io/socket.io.js`.
2. If you want to access the client as a normal HTML file without having to specify the socket port, you need to specify the domain and port in the Socket.io constructor call:

```
<script src=socket.io.js></script>
<script>
  const socket = io('https://everling.lu:9000');
</script>
```

3. To turn debugging on on the client enter `localStorage.debug = '*';` in the console. To turn it off, set the value to `''`. To log only certain debug messages, just specify what you want, e.g. `localStorage.debug = 'engine.io-client:polling, engine.io-client:socket';` (cf. "Socket.IO Cookbook").
4. To turn on debugging on the server either start the server using `DEBUG=* node server` or set the `DEBUG` environment variable. using `export DEBUG=*` and start the server. To stop server logging, set the variable to `null`.
5. Using socket.io behind a reverse proxy is a nightmare, see [xpo6.com/socket-io-via-apache-reverse-proxy](http://xpo6.com/socket-io-via-apache-reverse-proxy).

```
'use strict';

const io = require('socket.io');
const https = require('https'); // include HTTPS module
const fs = require('fs'); // include File System module
const port = 9000; // Port that our server will be listening on. It needs to be opened
on
// the firewall.
const options = { // Paths to our private key and certificate.
  key: fs.readFileSync('/etc/letsencrypt/live/students.btsi.lu/privkey.pem'),
  cert: fs.readFileSync('/etc/letsencrypt/live/students.btsi.lu/cert.pem')
};
const server = https.createServer(options, (req, res) => {
  res.writeHead(200);
  console.log(req.headers);
  res.end('hello world\n');
});

server.on('listening', () => {
  console.log(`Listening on port ${port}...`);
});

server.on('connection', () => {
  console.log('New connection...');
});

server.on('close', () => {
  console.log('Connection closed...');
});

server.listen(port);
const socket_server = io(server);
socket_server.on('connection', socket => {
  socket.emit('news', {hello: 'world'});
  socket.on('my other event', data => {
    console.log(data);
  });
});
})
```

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Socket.io client 1</title>
    <script src=socket.io.js></script> <!-- File needs to be in the same directory as
this
    file. Alternatively, use src=https://everling.lu:9000/socket.io/socket.io.js if
Socket.io
    was installed locally in which case the Socket.io server will serve it
automatically at
    this link. -->
    <script>
      const socket = io('https://students.btsi.lu/node');
      socket.on('news', data => {
        console.log(data);
        socket.emit('my other event', {my: 'data'});
      });
    </script>
  </head>
  <body>
  </body>
</html>

```

#### Authentication/authorization

[www.christian-schneider.net/CrossSiteWebSocketHijacking.html](http://www.christian-schneider.net/CrossSiteWebSocketHijacking.html)

[security.stackexchange.com/questions/76816/preventing-csrf-attacks-against-websocket-communications](https://security.stackexchange.com/questions/76816/preventing-csrf-attacks-against-websocket-communications)

#### WebRTC

[www.webrtc-experiment.com](http://www.webrtc-experiment.com)

[github.com/rtc-io/rtc](https://github.com/rtc-io/rtc)

[bitbucket.org/webrtc/codelab](https://bitbucket.org/webrtc/codelab)

[www.html5rocks.com/en/tutorials/webrtc/basics](http://www.html5rocks.com/en/tutorials/webrtc/basics)

[www.ghacks.net/2018/03/29/how-to-check-if-sites-use-webrtc](http://www.ghacks.net/2018/03/29/how-to-check-if-sites-use-webrtc)

#### 5.4.6. Express.js

**Express** is a minimal and flexible Node.js web application framework that provides a robust set of features for web and mobile applications.

#### Form handling

## HTTPFormClient1.html

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>HTTP Form Client 1</title>
    <link href=node/HTTPClient1.css rel=stylesheet>
  </head>
  <body>
    <form method=post>
      <input name=user placeholder=username required>
      <input type=password name=pw placeholder=password required>
      <button>Login</button>
    </form>
  </body>
</html>
```

## HTTPFormServer1.js

```
'use strict';

const fs = require('fs');
const express = require('express');
const app = express();
const port = 9000; // The port number that our HTTP server will be listening on.

app.use(express.urlencoded({extended: true}));
app.post('/', (req, res) => {
  console.dir(req.body);
  const {method, url, headers} = req; // All headers are represented in lower-case only.
  console.log(method);
  console.log(url);
  console.log(req.headers['content-type']);
  if (req.headers['content-type'] === 'application/x-www-form-urlencoded') {
    if (req.body) {
      console.dir(req.body);
      let login = false;
      if (req.body.user && req.body.pw) {
        if (req.body.user === 'a1' && req.body.pw === 'a1') login = true;
        res.send(login ? '<h1>logged in</h1>' : '<h1>invalid credentials</h1>');
      } else res.send('<h1>Something went wrong!</h1>');
    } else
      res.send('<h1>Empty request body...</h1>');
  } else
    res.send('<h1>Nothing to see here...</h1>');
});

app.get('/HTTPClient1.css', (req, res) => {
```

```

    console.log(req.url);
    res.on('error', err => {
        console.error(err);
    });
    try {
        const rs = fs.createReadStream('HTTPClient1.css');
        rs.on('error', err => res.end('Could not read file!'));
        rs.pipe(res);
    } catch (e) {
        res.send('<h1>Could not read file!</h1>');
    }
});

app.get('/', (req, res) => {
    console.log(req.url);
    res.on('error', err => {
        console.error(err);
    });
    try {
        const rs = fs.createReadStream('HTTPFormClient1.html');
        rs.on('error', err => res.end('Could not read file!'));
        rs.pipe(res);
    } catch (e) {
        res.send('<h1>Could not read file!</h1>');
    }
});

app.listen(port); // Let the server listen for HTTP requests.

```

We can make our life easier by creating a folder where we put all the files that we want to be publicly accessible and then create a static route in Express as explained in [expressjs.com/en/starter/static-files.html](https://expressjs.com/en/starter/static-files.html).

HTTPFormClient2.html

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>HTTP Form Client 1</title>
    <link href=node/HTTPFormClient2.css rel=stylesheet>
  </head>
  <body>
    <form method=post>
      <input name=user placeholder=username required>
      <input type=password name=pw placeholder=password required>
      <button>Login</button>
    </form>
  </body>
</html>
```

#### HTTPFormClient2.css

```
body {
  background-color: orange;
}
```

#### HTTPFormServer2.js

```

'use strict';

const fs = require('fs');
const express = require('express');
const app = express();
const port = 9000; // The port number that our HTTP server will be listening on.

app.use(express.static(__dirname + '/public'));
app.use(express.urlencoded({extended: true}));
app.post('/', (req, res) => {
  console.dir(req.body);
  const {method, url, headers} = req; // All headers are represented in lower-case only.
  console.log(method);
  console.log(url);
  console.log(req.headers['content-type']);
  if (req.headers['content-type'] === 'application/x-www-form-urlencoded') {
    if (req.body) {
      console.dir(req.body);
      let login = false;
      if (req.body.user && req.body.pw) {
        if (req.body.user === 'a1' && req.body.pw === 'a1') login = true;
        res.end(login ? '<h1>logged in</h1>' : '<h1>invalid credentials</h1>');
      } else res.end('<h1>Something went wrong!</h1>')
    } else
      res.end('<h1>Empty request body...</h1>');
  } else
    res.end('<h1>Nothing to see here...</h1>');
});

app.get('/', (req, res) => {
  console.log(req.url);
  res.on('error', err => {
    console.error(err);
  });
  try {
    const rs = fs.createReadStream('HTTPFormClient2.html');
    rs.on('error', err => res.end('Could not read file!'));
    rs.pipe(res);
  } catch (e) {
    res.end('<h1>Could not read file!</h1>');
  }
});

app.listen(port); // Let the server listen for HTTP requests.

```

Here is a slightly more elaborated example where we give the user 3 login attempts after which his IP address will be blocked for 5 minutes:

## HTTPFormServer3.js

```
'use strict';

const fs = require('fs');
const express = require('express');
const app = express();
const port = 9000; // The port number that our HTTP server will be listening on.
const blackListedIPs = {}, failedAttemptIPs = {};

const blacklist = (req, res, next) => {
  const IP = req.headers['x-forwarded-for'];
  if (blackListedIPs[IP]) {
    if (blackListedIPs[IP] < Date.now()) {
      delete blackListedIPs[IP];
      delete failedAttemptIPs[IP];
    } else {
      console.log(`${IP} is blacklisted`);
      res.end('<h1>You have been blacklisted!</h1>');
    }
  }
  next();
};

app.use(express.static(__dirname + '/public'));
app.use(express.urlencoded({extended: true}));
app.use(blacklist);
app.post('/', (req, res) => {
  console.dir(req.headers['content-type']);
  if (req.headers['content-type'] === 'application/x-www-form-urlencoded') {
    if (req.body) {
      if (req.body.user && req.body.pw) {
        if (req.body.user === 'a1' && req.body.pw === 'a1')
          res.end('<h1>logged in</h1>');
        else {
          const IP = req.headers['x-forwarded-for'];
          if (failedAttemptIPs[IP]) failedAttemptIPs[IP]++;
          else failedAttemptIPs[IP] = 1;
          if (failedAttemptIPs[IP] === 3) blackListedIPs[IP] = Date.now() + 300000;
          res.end('<h1>Invalid credentials.<br>You have ${3- failedAttemptIPs[IP]}
attempts left</h1>');
        }
      } else res.end('<h1>Something went wrong!</h1>')
    } else
      res.end('<h1>Empty request body...</h1>');
  } else
    res.end('<h1>Nothing to see here...</h1>');
});

app.get('/', (req, res) => {
  console.log(req.url);
```

```
res.on('error', err => {
  console.error(err);
});
try {
  const rs = fs.createReadStream('HTTPFormClient2.html');
  rs.on('error', err => res.end('Could not read file!'));
  rs.pipe(res);
} catch (e) {
  res.end('<h1>Could not read file!</h1>');
}
});

app.listen(port); // Let the server listen for HTTP requests.
```

Here is a more elegant solution to the same problem:

[BlackList.js](#)

```

'use strict';

class BlackList {
  constructor() {
    const failedAttempts = {};
    const blackList = {};

    this.failedAttempt = ip => {
      if (typeof ip !== 'string' && !(ip instanceof String)) ip = toString(ip);
      if (!failedAttempts[ip]) {
        failedAttempts[ip] = 1;
        return 2;
      } else {
        failedAttempts[ip]++;
        if (failedAttempts[ip] === 3) {
          blackList[ip] = Date.now() + 300000;
          return 0;
        }
        return 1;
      }
    };

    this.isBlackListed = ip => {
      if (typeof ip !== 'string' && !(ip instanceof String)) ip = toString(ip);
      if (blackList[ip])
        if (blackList[ip] >= Date.now()) return true;
      else {
        delete failedAttempts[ip];
        delete blackList[ip];
      }
      return false;
    };
  }
}

module.exports = new BlackList();

```

### HTTPFormServer4.js

```

'use strict';

const fs = require('fs');
const express = require('express');
const blackList = require('./BlackList');
const app = express();
const port = 9000; // The port number that our HTTP server will be listening on.

const blacklist = (req, res, next) => {
  const ip = req.headers['x-forwarded-for'];

```

```

    if (blackList.isBlackListed(ip)) {
      console.log(`${ip} is blacklisted`);
      res.send('<h1>You have been blacklisted!</h1>');
    }
    next();
  };

  app.use(express.static(__dirname + '/public'));
  app.use(express.urlencoded({extended: true}));
  app.use(blacklist);
  app.post('/', (req, res) => {
    console.dir(req.headers['content-type']);
    if (req.headers['content-type'] === 'application/x-www-form-urlencoded') {
      if (req.body) {
        console.dir(req.body);
        if (req.body.user && req.body.pw) {
          if (req.body.user === 'a1' && req.body.pw === 'a1')
            res.send('<h1>logged in</h1>');
          else {
            const failures = blackList.failedAttempt(req.headers['x-forwarded-for']);
            res.end('<h1>Invalid credentials.<br>You have ${failures} attempts
left</h1>');
          }
        } else res.send('<h1>Something went wrong!</h1>')
      } else
        res.send('<h1>Empty request body...</h1>');
    } else
      res.send('<h1>Nothing to see here...</h1>');
  });

  app.get('/', (req, res) => {
    console.log(req.url);
    res.on('error', err => {
      console.error(err);
    });
    try {
      const rs = fs.createReadStream('HTTPFormClient2.html');
      rs.on('error', err => res.end('Could not read file!'));
      rs.pipe(res);
    } catch (e) {
      res.send('<h1>Could not read file!</h1>');
    }
  });

  app.listen(port); // Let the server listen for HTTP requests.

```

The AJAX and JSON example from above can be implemented and enhanced using Express like so:

## HTTPServer2.js

```
'use strict';

const fs = require('fs');
const express = require('express');
const app = express();
const port = 9000; // The port number that our HTTP server will be listening on.

const logger = (req, res, next) => {
  const d = new Date();
  let s = `${d.getDate()} ${d.getMonth()} ${d.getFullYear()} ${d.getHours()}`;
  s += ` ${d.getMinutes()} ${d.getSeconds()}`;
  console.log(s);
  next();
};

app.use(logger);
app.post('/', (req, res) => {
  console.dir(req);
  const {method, url, headers} = req; // All headers are represented in lower-case only.
  console.log(method);
  console.log(url);
  console.dir(headers);
  console.log(req.headers['content-type']);
  if (req.headers['content-type'] === 'application/json') {
    let data = '';
    req.on('error', err => {
      console.error(err.stack);
    }).on('data', chunk => {
      data += chunk;
    }).on('end', () => {
      if (data) {
        try {
          data = JSON.parse(data);
        } catch (e) {
          console.error(e);
        }
      }
      console.dir(data);
      res.on('error', err => {
        console.error(err);
      });
      let login = false;
      if (data.un === 'a1' && data.pw === 'a1') login = true;
      res.json(JSON.stringify({login: login}));
    });
  }
});

/*app.get('/*', (req, res) => {
```

```

    console.log(req.url);
    res.on('error', err => {
      console.error(err);
    });
    try {
      if (req.url === '/HTTPClient1.css') {
        const rs = fs.createReadStream('HTTPClient1.css');
        rs.on('error', err => res.end('Could not read file!'));
        rs.pipe(res);
      } else {
        const rs = fs.createReadStream('HTTPClient1.html');
        rs.on('error', err => res.end('Could not read file!'));
        rs.pipe(res);
      }
    } catch (e) {
      res.send('<h1>Could not read file!</h1>');
    }
  });*/

app.get('/HTTPClient1.css', (req, res) => {
  console.log(req.url);
  res.on('error', err => {
    console.error(err);
  });
  try {
    const rs = fs.createReadStream('HTTPClient1.css');
    rs.on('error', err => res.end('Could not read file!'));
    rs.pipe(res);
  } catch (e) {
    res.send('<h1>Could not read file!</h1>');
  }
});

app.get('/', (req, res) => {
  console.log(req.url);
  res.on('error', err => {
    console.error(err);
  });
  try {
    const rs = fs.createReadStream('HTTPClient1.html');
    rs.on('error', err => res.end('Could not read file!'));
    rs.pipe(res);
  } catch (e) {
    res.send('<h1>Could not read file!</h1>');
  }
});

app.listen(port); // Let the server listen for HTTP requests.

```

If our server is running behind a reverse proxy, we can use `app.set('trust proxy', true)`; to get

the client IP address (cf. [stackoverflow.com/questions/46295635/how-to-get-ip-address-in-node-js-express](https://stackoverflow.com/questions/46295635/how-to-get-ip-address-in-node-js-express) and [expressjs.com/en/guide/behind-proxies.html](https://expressjs.com/en/guide/behind-proxies.html)).

### 5.4.7. Database handling

#### MySQL

We can use the `mysql2` package to access a MySQL/MariaDB database. See `DB.js` for a concrete application example.

[codeforgeek.com/2015/01/nodejs-mysql-tutorial](https://codeforgeek.com/2015/01/nodejs-mysql-tutorial)

[www.sitepoint.com/using-node-mysql-javascript-client](https://www.sitepoint.com/using-node-mysql-javascript-client)

Here is a simple example:

`createNodeDBTable1.sql`

```
USE evegi144;

DROP TABLE IF EXISTS tblNodeUser;

CREATE TABLE tblNodeUser
(
    idNodeUser      INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
    dtUserName      VARCHAR(32) NOT NULL UNIQUE,
    dtPasswordHash  VARCHAR(255) NOT NULL
)
ENGINE = INNODB
DEFAULT CHARSET utf8mb4
DEFAULT COLLATE utf8mb4_bin;
```

`mysql_options.js`

```
'use strict';

const DBConnectOpts = {
  host: 'localhost',
  port: 3306,
  charset: 'utf8_bin',
  user: 'xxx',
  password: 'xxx',
  database: 'evegi144'
};

exports.DBConnectOpts = DBConnectOpts;
```

DB1.js

```
'use strict';

class DB {
  constructor() {
    const mysql = require('mysql2');
    const DBConnectOpts = require(__dirname + '/mysql_options').DBConnectOpts;
    const conn = mysql.createConnection(DBConnectOpts);

    this.destroy = () => {
      if (conn) conn.destroy();
    };

    this.getUserData = (userName, cb) => {
      conn.execute('SELECT * FROM tblNodeUser WHERE dtUserName=?', [`${userName}`],
        (err, rows, fields) => {
          if (cb) cb(err, rows);
        });
    };

    this.registerUser = (userName, passwordHash, cb) => {
      conn.execute('INSERT INTO tblNodeUser(dtUserName, dtPasswordHash) VALUES (?,
?)',
        [`${userName}`, `${passwordHash}`], (err, rows, fields) => {
          if (cb) cb(err, rows);
        });
    };
  }
}

module.exports = new DB();
```

mysql1.js

```

'use strict';

const bcrypt = require('bcryptjs');
const DB = require('./DB1');
const userName = 'a1';
const password = 'a1';

DB.getUserData(userName, (err, rows) => {
  if (err) throw err;
  if (rows.length === 1) {
    bcrypt.compare(password, rows[0].dtPasswordHash).then(res => {
      if (res) console.log('You are now logged in.');
```

```

      else console.log('Invalid credentials');
      process.exit();
    });
  } else { // Must be 0 as > 1 not possible due to uniqueness in DB.
    bcrypt.hash(password, 10).then(hash => {
      DB.registerUser(userName, hash, (err, rows) => {
        if (err) throw err;
        console.log('New user inserted');
        process.exit();
      });
    });
  }
});

```

The same example using `mysql2/promise` and `async/await`:

`DB2.js`

```

'use strict';

class DB {
  constructor() {
    const mysql = require('mysql2/promise');
    const DBConnectOpts = require(__dirname + '/mysql_options').DBConnectOpts;
    let conn = undefined;

    const connect = async () => {
      if (!conn) conn = await mysql.createConnection(DBConnectOpts);
    };

    this.destroy = async () => {
      return conn.destroy();
    };

    this.getUserData = async userName => {
      if (!conn) await connect();
      return conn.execute('SELECT * FROM tblNodeUser WHERE dtUserName=?',
        [`${userName}`]);
    };

    this.registerUser = async (userName, passwordHash,) => {
      if (!conn) await connect();
      return conn.execute('INSERT INTO tblNodeUser(dtUserName, dtPasswordHash) VALUES
        (?, ?)',
        [`${userName}`, `${passwordHash}`]);
    };
  }
}

module.exports = new DB();

```

mysql2.js

```

'use strict';

const bcrypt = require('bcryptjs');
const DB = require('./DB2');
const userName = 'a1';
const password = 'a1';

(async () => {
  try {
    const data = await DB.getUserData(userName);
    const rows = data[0];
    let loggedin = false;
    if (rows.length === 1) {
      loggedin = await bcrypt.compare(password, rows[0].dtPasswordHash);
      if (loggedin) console.log('You are now logged in.');
```

```

      else console.log('Invalid credentials');
```

```

    } else { // Must be 0 as > 1 not possible due to uniqueness in DB.
```

```

      const hash = await bcrypt.hash(password, 10);
```

```

      await DB.registerUser(userName, hash);
```

```

      console.log('New user inserted');
```

```

    }
```

```

    console.log(`Final loggedin value: ${loggedin}`);
```

```

  } catch (err) {
```

```

    console.log(err);
```

```

  }
```

```

  process.exit();
```

```

})();
```

## CouchDB

[couchdb.apache.org](http://couchdb.apache.org)

## PostgreSQL

[PostgreSQL on Ubuntu](#)

[PostgreSQL docs](#)

[Secure PostgreSQL](#)

[PostgreSQL and NodeJS](#)

[blog.risingstack.com/node-js-database-tutorial](http://blog.risingstack.com/node-js-database-tutorial)

Edit `/etc/postgresql/9.6/main/pg_hba.conf` to look like this:

```
# Database administrative login by Unix domain socket
local  all                postgres                    peer

# TYPE  DATABASE      USER      ADDRESS              METHOD

# "local" is for Unix domain socket connections only
local  all                all                md5 ①
# IPv4 local connections:
host   all                all             127.0.0.1/32        md5
# IPv6 local connections:
host   all                all             ::1/128             md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
#local  replication    postgres                    peer
#host   replication    postgres                    md5
#host   replication    postgres                    md5
host   all                all             0.0.0.0/0            reject ②
```

① Changed from **peer** to **md5**.

② Added for security so that connections from other hosts get rejected.

Unquoted names are case-insensitive and explicitly lower-cased. Quoted names are case-sensitive.

## psql

### psql

```
su - postgres
psql dbname user
psql -d dbname -U user
\du #list users
\l #list databases
\q exit
\password user #set user password
CREATE ROLE <user> LOGIN CREATEDB PASSWORD <pw>
```

## pgAdmin

### Download server deployment

## MassiveJS

**MassiveJS** is a PostgreSQL-specific data access tool.

## MongoDB

**MongoDB**, a NoSQL DB, “is an open-source document database that provides high performance, high availability, and automatic scaling”. MongoDB is fully programmable in JS. The [MongoDB manual](#) should be your starting place.

## Installation

Follow the official [instructions](#). On Ubuntu 16.10, the following needs to be done ([source](#)):

1. Comment out the `Limit` and `Task` instructions in the `Service` section of `/lib/systemd/system/mongod.service`.
2. `systemctl enable mongod.service`
3. `systemctl status mongod` or `systemctl list-unit-files --type=service`
4. `systemctl daemon-reload` may or may not be needed.

## Configuration

[Disable transparent huge pages](#). Edit `/etc/mongod.conf` and set `bindIp: 0.0.0.0`. In the MongoDB shell:

```
use admin
db.createUser(
  {
    user: "myUserAdmin",
    pwd: "abc123",
    roles: [{role: "userAdminAnyDatabase", db: "admin"}]
  }
)
```

Then set

```
security:
  authorization: enabled
```

in `/etc/mongod.conf`

After restarting mongod you need to connect as `mongo -u <user> -p --authenticationDatabase admin`.

## Usage

`use <db>` Switches to a DB or creates it if it does not yet exist.

## Redis

Start Redis command line interface using `redis-cli`. List all keys via `keys` . Get the type of a key via `type <key>`. Depending on the type use `get <key>` or `smembers <key>` or `hgetall <key>`. Delete keys using pattern: `redis-cli --scan --pattern 'Product:*' | xargs redis-cli DEL`. Delete all keys in all databases using `flushall`.



Redis security is extremely weak: [www.digitalocean.com/community/tutorials/how-to-secure-your-redis-installation-on-ubuntu-14-04](http://www.digitalocean.com/community/tutorials/how-to-secure-your-redis-installation-on-ubuntu-14-04)

[redis.io](https://redis.io)

[redis.io/topics/persistence](https://redis.io/topics/persistence)

[redis.js.org](https://redis.js.org)

[www.cheatography.com/tasjaevan/cheat-sheets/redis](https://www.cheatography.com/tasjaevan/cheat-sheets/redis)

[lzone.de/cheat-sheet/Redis](https://lzone.de/cheat-sheet/Redis)

[developer.mozilla.org/en-US/docs/Mozilla/Redis\\_Tips](https://developer.mozilla.org/en-US/docs/Mozilla/Redis_Tips)

[matt.sh/thinking-in-redis-part-one](https://matt.sh/thinking-in-redis-part-one)

## 5.4.8. Sessions

### With HTTP(s)

Using `express-session` and `express-mysql-session` we can add sessions to our earlier AJAX and JSON example:

`mysql_options.js`

```
'use strict';

const mysqlSessionOpts = {
  host: 'localhost', // Host name for database connection.
  port: 3306, // Port number for database connection.
  user: 'xxx', // Database user.
  password: 'xxx', // Password for the above database user.
  database: 'xxx', // Database name.
  checkExpirationInterval: 90000, // How frequently expired sessions will be cleared;
  milliseconds.
  expiration: 86400000, // The maximum age of a valid session; 24 hours in
  milliseconds.
  createDatabaseTable: true
};

const DBConnectOpts = {
  host: 'localhost',
  port: 3306,
  charset: 'utf8_bin',
  user: 'xxx',
  password: 'xxx',
  database: 'xxx'
};

exports.mysqlSessionOpts = mysqlSessionOpts;
exports.DBConnectOpts = DBConnectOpts;
```

```

'use strict';

const fs = require('fs');
const express = require('express');
const app = express();
const session = require('express-session');
const MySQLStore = require('express-mysql-session')(session);
const port = 9000; // The port number that our HTTP server will be listening on.

const mysqlSessionOpts = require('./mysql_options').mysqlSessionOpts;
const sessionStore = new MySQLStore(mysqlSessionOpts);
const sessionOpts = {
  cookie: {
    path: '/', secure: false, httpOnly: true, sameSite: true /*,
    maxAge: new Date(Date.now() + 300000)*/ /* 5 min */
  },
  secret: 'my-secret',
  resave: false,
  saveUninitialized: true,
  name: 'test1.sid',
  key: 'session_cookie_name',
  store: sessionStore,
  proxy: true
};
const sess = session(sessionOpts);

const logger = (req, res, next) => {
  const d = new Date();
  let s = `${d.getDate()} ${d.getMonth()} ${d.getFullYear()} ${d.getHours()}`;
  s += ` ${d.getMinutes()} ${d.getSeconds()}`;
  console.log(s);
  console.dir(req.session);
  next();
};

app.set('trust proxy', true); // https://expressjs.com/en/guide/behind-proxies.html
app.use(sess);
app.use(logger);
app.post('*', (req, res) => {
  const {method, url, headers} = req; // All headers are represented in lower-case
  only.
  console.log(req.headers['content-type']);
  if (req.headers['content-type'] === 'application/json') {
    let data = '';
    req.on('error', err => {
      console.error(err.stack);
    }).on('data', chunk => {
      data += chunk;
    }).on('end', () => {

```

```

    if (data) {
      try {
        data = JSON.parse(data);
      } catch (e) {
        console.error(e);
      }
    }
    console.dir(data);
    res.on('error', err => {
      console.error(err);
    });
    let login = false;
    if (data.un === 'a1' && data.pw === 'a1') login = true;
    req.session.loggedin = login;
    //req.session.save();
    console.dir(req.session);
    res.end(JSON.stringify({login: login}));
  });
}
});
app.get('/', (req, res) => {
  res.on('error', err => {
    console.error(err);
  });
  try {
    let filenameToServe = 'session1.html';
    if (req.session && req.session.loggedin) filenameToServe =
'session1_loggedin.html';
    const rs = fs.createReadStream(filenameToServe);
    rs.on('error', err => res.end('Could not read file!'));
    rs.pipe(res);
  } catch (e) {
    res.end('<h1>Could not read file!</h1>');
  }
});

app.listen(port); // Let the server listen for HTTP requests.

```

session1.html

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Session Client 1</title>
    <script>
      'use strict';

      const init = () => {
        const button = document.querySelector('button');
        button.addEventListener('click', () => {
          const inputs = document.querySelectorAll('input');
          if (inputs[0].value && inputs[1].value) {
            fetch('', {
              headers: {
                'Accept': 'application/json',
                'Content-Type': 'application/json'
              },
              method: "POST",
              credentials: 'same-origin', //
https://github.com/expressjs/session/issues/374
              body: JSON.stringify({un: inputs[0].value, pw: inputs[1].value})
            }).then(response => response.json()).then(data => {
              window.location = '/node';
            }).catch(error => {
              console.log(`There has been a problem with the fetch operation:
${error.message}`);
            });
          }
        });
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <input name=user placeholder=username required>
    <input type=password name=pw placeholder=password required>
    <button>Login</button>
  </body>
</html>

```

session1\_loggedin.html

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Session Client 1 loggedin</title>
    <script>
      'use strict';

      const init = () => {
        const button = document.querySelector('button');
        button.addEventListener('click', () => {
          fetch('', {
            headers: {
              'Accept': 'application/json',
              'Content-Type': 'application/json'
            },
            method: "POST",
            credentials: 'same-origin', //
https://github.com/expressjs/session/issues/374
            body: JSON.stringify({logout: 'logout'})
          }).then(response => response.json()).then(data => {
            window.location = '/node';
          }).catch(error => {
            console.log(`There has been a problem with the fetch operation:
${error.message}`);
          });
        });
      };
      addEventListener('load', init);
    </script>
  </head>
  <body>
    <button>Logout</button>
  </body>
</html>

```

## With HTTP(s) and WebSocket

We can also use sessions for both HTTP and WebSocket communication. Please note that we need to parse the session for every WebSocket message we receive:

HTTPWSServer1.js

```

'use strict';

const fs = require('fs');
const http = require('http');
const WebSocketServer = require('ws').Server;
const express = require('express');

```

```

const app = express();
const session = require('express-session');
const MySQLStore = require('express-mysql-session')(session);
const port = 9000; // The port number that our HTTP server will be listening on.

const mysqlSessionOpts = require(__dirname + '/mysql_options').mysqlSessionOpts;
const sessionStore = new MySQLStore(mysqlSessionOpts);
const sessionOpts = {
  cookie: {
    path: '/', domain: 'students.btsi.lu', secure: false, httpOnly: true, sameSite:
true,
    expires: new Date(Date.now() + 86400000)/*,
    maxAge: new Date(Date.now() + 300000)*/ /* 5 min */
  },
  secret: 'my-secret',
  resave: false,
  saveUninitialized: true,
  name: 'test1.sid',
  key: 'session_cookie_name',
  store: sessionStore,
  proxy: true
};
const sess = session(sessionOpts);

app.set('trust proxy', true); // https://expressjs.com/en/guide/behind-proxies.html
app.use(sess);
app.post('/', (req, res) => {
  if (req.headers['content-type'] === 'application/json') {
    let data = '';
    req.on('error', err => {
      console.error(err.stack);
    }).on('data', chunk => {
      data += chunk;
    }).on('end', () => {
      if (data) {
        try {
          data = JSON.parse(data);
        } catch (e) {
          console.error(e);
        }
      }
      console.dir(data);
      res.on('error', err => {
        console.error(err);
      });
      let login = false;
      if (data.un === 'a1' && data.pw === 'a1') login = true;
      req.session.loggedin = login;
      res.json(JSON.stringify({login: login}));
      console.log('*** POST ***');
      console.log(req.sessionID);
    }
  }
});

```

```

        console.dir(req.session);
        console.log();
    });
}
});
app.get('/', (req, res) => {
    res.on('error', err => {
        console.error(err);
    });
    try {
        if (req.session.a) req.session.a++;
        else req.session.a = 1;
        console.log('*** GET ***');
        console.log(req.sessionID);
        console.dir(req.session);
        console.log();
        const rs = fs.createReadStream('HTTPWSCClient1.html');
        rs.on('error', err => res.end('Could not read file!'));
        rs.pipe(res);
    } catch (e) {
        res.send('<h1>Could not read file!</h1>');
    }
});

const server = http.createServer(app);
const wss = new WebSocketServer({server});

wss.on('error', err => {
    console.dir(err);
});

wss.on('connection', (socket, req) => {
    console.log('WS client connected...');

    socket.on('error', err => {
        console.dir(err);
    });

    socket.on('message', async data => {
        // Initialize the WS session from the HTTPS session.
        const initSession = async () => {
            return new Promise((resolve, reject) => {
                sess(req, {}, () => {
                    resolve();
                    console.log('WebSocket session is parsed.');
                });
            });
        };

        await initSession(); // Make sure the WebSocket session is current.
        req.session.test = Math.random();
    });
});

```

```

    req.session.save(); // We need to call this in WebSocket connections.
    console.log(data);
    console.log('*** WS ***');
    console.log(req.sessionID);
    console.dir(req.session);
    console.log();
    socket.send(`You said: ${data}<br>`);
  });

  socket.on('close', () => {
    console.log('Socket closed');
  });

}
);

server.listen(port); // Let the server listen for HTTP requests.

```

#### HTTPWSCliet1.html

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>HTTP Client 1</title>
    <script>
      'use strict';

      const init = () => {
        const wssURL = 'wss://students.btsi.lu/node/wss';
        const buttons = document.querySelectorAll('button');
        const inputs = document.querySelectorAll('input');
        const section = document.querySelector('section');
        buttons[0].addEventListener('click', () => {
          if (inputs[0].value && inputs[1].value) {
            fetch('', {
              headers: {
                'Accept': 'application/json',
                'Content-Type': 'application/json'
              },
              method: "POST",
              credentials: 'same-origin', //
https://github.com/expressjs/session/issues/374
              body: JSON.stringify({un: inputs[0].value, pw: inputs[1].value})
            }).then(response => response.json()).then(data => {
              console.dir(data);
            }).catch(error => {
              console.log(`There has been a problem with the fetch operation:
${error.message}`);
            });
          }
        });
      };
    </script>
  </head>
  <body>
    <div>
      <input type="text" value="username" />
      <input type="password" value="password" />
      <button value="connect" />
    </div>
  </body>
</html>

```

```

    /*const req1 = new XMLHttpRequest();
    req1.open('POST', '');
    req1.addEventListener('load', e => {
        console.dir(JSON.parse(e.target.response));
    });
    req1.send(JSON.stringify({un: inputs[0].value, pw: inputs[1].value}));*/
}
});
const wss = new WebSocket(wssURL);
wss.addEventListener('open', () => {
    console.log('WebSocket connection opened.');
    buttons[1].addEventListener('click', () => {
        wss.send(inputs[2].value);
    });
});

wss.addEventListener('message', e => {
    section.innerHTML += e.data;
});

wss.addEventListener('close', () => {
    console.log('Disconnected...');
});

wss.addEventListener('error', () => {
    console.log('Error...');
});
};

    addEventListener('load', init);
</script>
</head>
<body>
    <h1>HTTP</h1>
    <input name=user placeholder=username required>
    <input type=password name=pw placeholder=password required>
    <button>Login</button>
    <h1>WS</h1>
    <input name=msg placeholder=message required>
    <button>Send</button>
    <section></section>
</body>
</html>

```

[www.npmjs.com/package/express-session](http://www.npmjs.com/package/express-session)

[www.npmjs.com/package/express-mysql-session](http://www.npmjs.com/package/express-mysql-session)

[lockmedown.com/securing-node-js-managing-sessions-express-js](http://lockmedown.com/securing-node-js-managing-sessions-express-js)

[www.packtpub.com/books/content/using-socketio-and-express-together](http://www.packtpub.com/books/content/using-socketio-and-express-together)

<a href="https://github.com/expressjs/session/issues/208">github.com/expressjs/session/issues/208</a>
<a href="https://botleg.com/stories/using-your-sessions-with-socketio">botleg.com/stories/using-your-sessions-with-socketio</a>
<a href="https://mono.software/2014/08/25/Sharing-sessions-between-SocketIO-and-Express-using-Redis">mono.software/2014/08/25/Sharing-sessions-between-SocketIO-and-Express-using-Redis</a>
<a href="https://github.com/jfromaniello/passport.socketio">github.com/jfromaniello/passport.socketio</a>
<a href="https://www.npmjs.com/package/socket.io-mysql-session">www.npmjs.com/package/socket.io-mysql-session</a>
<a href="https://socket.io-express-session">socket.io-express-session</a>
<a href="https://www.npmjs.com/package/express-socket.io-session">www.npmjs.com/package/express-socket.io-session</a>
<a href="https://stackoverflow.com/questions/23494016/socket-io-and-express-4-sessions">stackoverflow.com/questions/23494016/socket-io-and-express-4-sessions</a>
<a href="https://stackoverflow.com/questions/24290699/socket-io-1-0-5-how-to-save-session-variables/24380110#24380110">stackoverflow.com/questions/24290699/socket-io-1-0-5-how-to-save-session-variables/24380110#24380110</a>
<a href="https://www.codementor.io/tips/0217388244/sharing-passport-js-sessions-with-both-express-and-socket-io">www.codementor.io/tips/0217388244/sharing-passport-js-sessions-with-both-express-and-socket-io</a>
<a href="https://www.packtpub.com/mapt/book/Web%20Development/9781783984480/2/ch02lvl1sec26/Using+Socket.IO+and+Express+together">www.packtpub.com/mapt/book/Web%20Development/9781783984480/2/ch02lvl1sec26/Using+Socket.IO+and+Express+together</a>
<a href="https://stackoverflow.com/questions/4753957/socket-io-authentication">stackoverflow.com/questions/4753957/socket-io-authentication</a>
<a href="https://www.safaribooksonline.com/library/view/node-up-and/9781449332235/ch07s02.html">www.safaribooksonline.com/library/view/node-up-and/9781449332235/ch07s02.html</a>
<a href="https://mono.software/2014/08/25/Sharing-sessions-between-SocketIO-and-Express-using-Redis">mono.software/2014/08/25/Sharing-sessions-between-SocketIO-and-Express-using-Redis</a>
<a href="https://www.packtpub.com/mapt/book/Web%20Development/9781785284922/7/ch07lvl1sec29/Authentication">www.packtpub.com/mapt/book/Web%20Development/9781785284922/7/ch07lvl1sec29/Authentication</a>
<a href="https://security.stackexchange.com/questions/115695/when-should-server-side-sessions-be-used-instead-of-client-side-sessions">security.stackexchange.com/questions/115695/when-should-server-side-sessions-be-used-instead-of-client-side-sessions</a>
<a href="http://phillbarber.blogspot.lu/2014/02/client-side-vs-server-side-session.html">phillbarber.blogspot.lu/2014/02/client-side-vs-server-side-session.html</a>
<a href="https://stackoverflow.com/questions/9071969/using-express-and-node-how-to-maintain-a-session-across-subdomains-hostheaders/14627464#14627464">stackoverflow.com/questions/9071969/using-express-and-node-how-to-maintain-a-session-across-subdomains-hostheaders/14627464#14627464</a>
<a href="https://lockmedown.com/securing-node-js-managing-sessions-express-js">lockmedown.com/securing-node-js-managing-sessions-express-js</a>
<a href="https://www.duringthedrive.com/2017/11/25/sesh-express-sessions-cookies-and-json-web-tokens">www.duringthedrive.com/2017/11/25/sesh-express-sessions-cookies-and-json-web-tokens</a>

### 5.4.9. Registration and login

reglogin.html

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Reglogin Client 1</title>
    <script>
      'use strict';

      const init = () => {
        const buttons = document.querySelectorAll('button');
        buttons[0].addEventListener('click', () => {
```

```

const inputs = document.querySelectorAll('input');
if (inputs[0].value && inputs[1].value) {
  fetch('', {
    headers: {
      'Accept': 'application/json',
      'Content-Type': 'application/json'
    },
    method: "POST",
    credentials: 'same-origin', //
https://github.com/expressjs/session/issues/374
    body: JSON.stringify({type: 'login', un: inputs[0].value, pw:
inputs[1].value})
  }).then(response => response.json()).then(data => {
    console.dir(data);
    window.location = '/node';
  }).catch(error => {
    console.log(`There has been a problem with the fetch operation:
${error.message}`);
  });
}
});
buttons[1].addEventListener('click', () => {
  const inputs = document.querySelectorAll('input');
  if (inputs[0].value && inputs[1].value) {
    fetch('', {
      headers: {
        'Accept': 'application/json',
        'Content-Type': 'application/json'
      },
      method: "POST",
      credentials: 'same-origin', //
https://github.com/expressjs/session/issues/374
      body: JSON.stringify({type: 'register', un: inputs[0].value, pw:
inputs[1].value})
    }).then(response => response.json()).then(data => {
      console.dir(data);
      window.location = '/node';
    }).catch(error => {
      console.log(`There has been a problem with the fetch operation:
${error.message}`);
    });
  }
});
};

  addEventListener('load', init);
</script>
</head>
<body>
  <input name=user placeholder=username required>
  <input type=password name=pw placeholder=password required>

```

```
<button>Login</button>
<button>Register</button>
</body>
</html>
```

reglogin\_loggedin.html

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <meta charset=UTF-8>
    <title>Reglogin Client 1 logged in</title>
    <script>
      'use strict';

      const init = () => {
        const button = document.querySelector('button');
        button.addEventListener('click', () => {
          const inputs = document.querySelectorAll('input');
          fetch('', {
            headers: {
              'Accept': 'application/json',
              'Content-Type': 'application/json'
            },
            method: "POST",
            credentials: 'same-origin', //
https://github.com/expressjs/session/issues/374
            body: JSON.stringify({type: 'logout'})
          }).then(response => response.json()).then(data => {
            console.dir(data);
            window.location = '/node';
          }).catch(error => {
            console.log(`There has been a problem with the fetch operation:
${error.message}`);
          });
        });
      };

      addEventListener('load', init);
    </script>
  </head>
  <body>
    <button>Logout</button>
  </body>
</html>
```

DB2.js

```

'use strict';

class DB {
  constructor() {
    const mysql = require('mysql2/promise');
    const DBConnectOpts = require(__dirname + '/mysql_options').DBConnectOpts;
    let conn = undefined;

    const connect = async () => {
      if (!conn) conn = await mysql.createConnection(DBConnectOpts);
    };

    this.destroy = async () => {
      return conn.destroy();
    };

    this.getUserData = async userName => {
      if (!conn) await connect();
      return conn.execute('SELECT * FROM tblNodeUser WHERE dtUserName=?',
        [`${userName}`]);
    };

    this.registerUser = async (userName, passwordHash,) => {
      if (!conn) await connect();
      return conn.execute('INSERT INTO tblNodeUser(dtUserName, dtPasswordHash) VALUES
        (?, ?)',
        [`${userName}`, `${passwordHash}`]);
    };
  }
}

module.exports = new DB();

```

### reglogin.js

```

'use strict';

const fs = require('fs');
const express = require('express');
const app = express();
const bcrypt = require('bcryptjs');
const DB = require('./DB2');
const session = require('express-session');
const MySQLStore = require('express-mysql-session')(session);
const port = 9000; // The port number that our HTTP server will be listening on.

const mysqlSessionOpts = require('./mysql_options').mysqlSessionOpts;
const sessionStore = new MySQLStore(mysqlSessionOpts);
const sessionOpts = {

```

```

cookie: {
  path: '/', secure: false, httpOnly: true, sameSite: true /*,
    maxAge: new Date(Date.now() + 300000)*/ /* 5 min */
},
secret: 'my-secret',
resave: false,
saveUninitialized: true,
name: 'test1.sid',
key: 'session_cookie_name',
store: sessionStore,
proxy: true
};

const sess = session(sessionOpts);

app.set('trust proxy', true); // https://expressjs.com/en/guide/behind-proxies.html
app.use(sess);
app.post('*', (req, res) => {
  if (req.headers['content-type'] === 'application/json') {
    let data = '';
    req.on('error', err => {
      console.error(err.stack);
    }).on('data', chunk => {
      data += chunk;
    }).on('end', async () => {
      if (data) {
        try {
          data = JSON.parse(data);
          console.dir(data);
          res.on('error', err => {
            console.error(err);
          });
          if (data.type) {
            req.session.loggedin = false;
            if (data.type === 'logout') {
              res.end(JSON.stringify({login: false}));
            } else if (data.type === 'login') {
              if (data.un && data.pw) {
                const userData = await DB.getUserData(data.un);
                const rows = userData[0];
                let loggedin = false;
                if (rows.length === 1) {
                  loggedin = await bcrypt.compare(data.pw, rows[0].dtPasswordHash);
                  req.session.loggedin = loggedin;
                } else console.log('Invalid credentials');
              }
            } else if (data.type === 'register') {
              if (data.un && data.pw) {
                const userData = await DB.getUserData(data.un);
                const rows = userData[0];
                if (rows.length < 1) {
                  const hash = await bcrypt.hash(data.pw, 10);

```

```

        await DB.registerUser(data.un, hash);
        console.log('New user inserted');
      }
    }
  }
  res.end(JSON.stringify({login: req.session.loggedin}));
}
} catch (e) {
  console.error(e);
}
}
});
}
});
app.get('/', (req, res) => {
  res.on('error', err => {
    console.error(err);
  });
  try {
    let filenameToServe = 'reglogin.html';
    if (req.session && req.session.loggedin) filenameToServe =
'reglogin_loggedin.html';
    const rs = fs.createReadStream(filenameToServe);
    rs.on('error', err => res.end('Could not read file!'));
    rs.pipe(res);
  } catch (e) {
    res.end('<h1>Could not read file!</h1>');
  }
});

app.listen(port); // Let the server listen for HTTP requests.

```

## 5.4.10. Security



Study

[nodesource.com/blog/nine-security-tips-to-keep-express-from-getting-](https://nodesource.com/blog/nine-security-tips-to-keep-express-from-getting-pwned)

[pwned.](https://nodesource.com/blog/nine-security-tips-to-keep-express-from-getting-pwned)

[github.com/animir/node-rate-limiter-flexible](https://github.com/animir/node-rate-limiter-flexible)

### Content Security Policy (CSP) and HTTP headers

[developer.mozilla.org/en-US/docs/Web/HTTP/CSP](https://developer.mozilla.org/en-US/docs/Web/HTTP/CSP)

[2019.jsconf.eu/stefan-judis/http-headers-for-the-responsible-developer.html](https://2019.jsconf.eu/stefan-judis/http-headers-for-the-responsible-developer.html)

[www.youtube.com/watch?v=TNlcoYLIGFk](https://www.youtube.com/watch?v=TNlcoYLIGFk)

[schepp.github.io/HTTP-headers](https://schepp.github.io/HTTP-headers)

## 5.4.11. Performance

### Cache control

[csswizardry.com/2019/03/cache-control-for-civilians](http://csswizardry.com/2019/03/cache-control-for-civilians)

### Diffeential serving

[calendar.perfplanet.com/2018/doing-differential-serving-in-2019](http://calendar.perfplanet.com/2018/doing-differential-serving-in-2019)

## 5.4.12. Other useful APIs and packages

### Server Sent Events

[www.terlici.com/2015/12/04/realtime-node-expressjs-with-sse.html](http://www.terlici.com/2015/12/04/realtime-node-expressjs-with-sse.html)

[www.npmjs.com/package/sse-nodejs](http://www.npmjs.com/package/sse-nodejs)

[www.npmjs.com/package/sse-node](http://www.npmjs.com/package/sse-node)

[www.npmjs.com/package/sse](http://www.npmjs.com/package/sse)

### Connect

[github.com/senchalabs/connect#middleware](https://github.com/senchalabs/connect#middleware)

### Authentication

[medium.com/@evangow/server-authentication-basics-express-sessions-passport-and-curl-359b7456003d](https://medium.com/@evangow/server-authentication-basics-express-sessions-passport-and-curl-359b7456003d)

[hackernoon.com/your-node-js-authentication-tutorial-is-wrong-f1a3bf831a46](https://hackernoon.com/your-node-js-authentication-tutorial-is-wrong-f1a3bf831a46)

### HTTP2

[webapplog.com/http2-server-push-node-express](http://webapplog.com/http2-server-push-node-express)

[medium.com/@sibu.it13/an-example-of-server-push-with-http-2-in-node-js-22757256f0b3](https://medium.com/@sibu.it13/an-example-of-server-push-with-http-2-in-node-js-22757256f0b3)

### PDF

#### Display

[PDF.js](#)

#### Generation

[PDFKit](#)

[pdfmake.org](http://pdfmake.org)

## Technical analysis

[github.com/anandanand84/technicalindicators](https://github.com/anandanand84/technicalindicators)

[github.com/andredumas/techan.js](https://github.com/andredumas/techan.js)

[github.com/oransel/node-talib](https://github.com/oransel/node-talib)

[github.com/TulipCharts/tulipnode](https://github.com/TulipCharts/tulipnode)

## Email

[www.sitepoint.com/sending-email-using-node-js](http://www.sitepoint.com/sending-email-using-node-js)

[emailjs.org](http://emailjs.org)

## SMTP

[nodemailer.com](http://nodemailer.com)

[www.mailgun.com](http://www.mailgun.com)

## IMAP

[github.com/mscdex/node-imap](https://github.com/mscdex/node-imap)

[www.npmjs.com/package/imap-simple](http://www.npmjs.com/package/imap-simple)

## Dealing with attachments:

[msdn.microsoft.com/en-us/library/office/aa563350\(v=exchg.140\).aspx](http://msdn.microsoft.com/en-us/library/office/aa563350(v=exchg.140).aspx)

[stackoverflow.com/questions/2342546/how-to-add-attachment-in-rfc822-format-mail-using-google-api](http://stackoverflow.com/questions/2342546/how-to-add-attachment-in-rfc822-format-mail-using-google-api)

[stackoverflow.com/questions/47338180/imap-append-not-working-with-attachments](http://stackoverflow.com/questions/47338180/imap-append-not-working-with-attachments)

[stackoverflow.com/questions/16088612/drafting-imap-emails-with-an-attachment-using-php](http://stackoverflow.com/questions/16088612/drafting-imap-emails-with-an-attachment-using-php)

## Working example

```

'use strict';

const nodemailer = require('nodemailer');
const imaps = require('imap-simple');
// Convert LF to CRLF to avoid bare newlines error, cf.
https://cr.yp.to/docs/smtp1f.html
const eol = require('eol');
const smtpConfig = {
  host: 'smtp.restena.lu',
  port: 587,
  secure: false, // upgrade later with STARTTLS
  auth: {
    user: 'xxxx',
    pass: 'xxxx'
  }
};
const imapConfig = {
  imap: {
    user: 'xxxx',
    password: 'xxxx',
    host: 'mail.restena.lu',
    port: 993,
    tls: true,
    authTimeout: 3000
  }
};
const transporter = nodemailer.createTransport(smtpConfig);
const message = {
  sender: 'gilles.everling@education.lu',
  from: 'gilles.everling@education.lu',
  /*replyTo: 'gilles.everling@education.lu',*/
  to: 'gilles.everling@education.lu',
  subject: 'Test',
  html: '<h1>Test</h1>'
};
imaps.connect(imapConfig).then(connection => {
  const message = eol.crlf(`Content-Type: text/html
To: gilles.everling@education.lu
From: gilles.everling@education.lu
Subject: Test

<h1>Test</h1>
`);
  connection.append(message, {mailbox: 'Inbox.Sent'});
  process.exit();
});

```

## Deployment

[pm2.keymetrics.io/docs/usage/process-management](https://pm2.keymetrics.io/docs/usage/process-management)

[certsimple.com/blog/deploy-node-on-linux](https://certsimple.com/blog/deploy-node-on-linux)

[www.npmjs.com/package/express-minify](https://www.npmjs.com/package/express-minify)

## Streams

Streams are not yet fully implemented in most browsers, cf. [caniuse.com/#feat=streams](https://caniuse.com/#feat=streams).

[developer.mozilla.org/en-US/docs/Web/API/Streams\\_API](https://developer.mozilla.org/en-US/docs/Web/API/Streams_API)

[deanhume.com/experimenting-with-the-streams-api](https://deanhume.com/experimenting-with-the-streams-api)

[jakearchibald.com/2016/streams-ftw](https://jakearchibald.com/2016/streams-ftw)

## Web and image optimization

[images.guide](https://images.guide)

[www.webpagetest.org](https://www.webpagetest.org)

[www.npmjs.com/package/compress-images](https://www.npmjs.com/package/compress-images)

[www.npmjs.com/package/sharp](https://www.npmjs.com/package/sharp)

## Others

[www.npmjs.com/package/jsfiddle-downloader](https://www.npmjs.com/package/jsfiddle-downloader)

download fiddles from jsFiddle.net

[github.com/molnarg/node-http2](https://github.com/molnarg/node-http2)

[www.clinicjs.org](https://www.clinicjs.org)

## 5.4.13. apess.lu

`mysql_options.js`

```

'use strict';

const mysqlSessionOpts = {
  host: 'localhost',// Host name for database connection.
  port: 3306,// Port number for database connection.
  user: 'xxx',// Database user.
  password: 'xxx',// Password for the above database user.
  database: 'xxx',// Database name.
  checkExpirationInterval: 90000,// How frequently expired sessions will be cleared;
  milliseconds.
  expiration: 86400000,// The maximum age of a valid session; 24 hours in
  milliseconds.
  createDatabaseTable: true
};

const DBConnectOpts = {
  host: 'localhost',
  port: 3306,
  charset: 'utf8_bin',
  user: 'xxx',
  password: 'xxx',
  database: 'xxx'
};

exports.mysqlSessionOpts = mysqlSessionOpts;
exports.DBConnectOpts = DBConnectOpts;

```

## DB.js

```

'use strict';

class DB {
  constructor() {
    const mysql = require('mysql2');
    const asyncMysql = require('mysql2/promise');
    const DBConnectOpts = require('./mysql_options').DBConnectOpts;
    const conn = mysql.createConnection(DBConnectOpts);
    let asyncConn = undefined;

    this.destroy = () => {
      if (conn) conn.destroy();
    };

    this.updatePassword = (email, hash, cb) => {
      conn.execute(`UPDATE tblMember SET dtPassword=? WHERE dtEmail=?`, [`${hash}`,
`${email}`],
      (err, rows, fields) => {
        if (cb) cb(err, rows);
      });
    };
  }
}

```

```

};

this.test = (exceptionEmails, cb) => {
  console.log('Exceptions: ' + exceptionEmails);
  conn.execute(`SELECT COUNT(*) FROM tblMember WHERE dtEmail NOT IN (?)`,
    [`${exceptionEmails}`],
    (err, rows, fields) => {
      if (cb) cb(err, rows);
    });
};

this.deletePasswords = (exceptionEmails, cb) => {
  conn.execute(`UPDATE tblMember SET dtPassword='' WHERE dtEmail NOT IN (?)`,
    [`${exceptionEmails}`],
    (err, rows, fields) => {
      if (cb) cb(err, rows);
    });
};

this.updateMember = (email, newMemberData, cb) => {
  conn.execute(`UPDATE tblMember SET
    dtFirstName=?, dtLastName=?, dtEmail=?, dtStreet=?, dtCity=?, dtCountry=?,
    dtPostalCode=?, dtSchool=?, dtFunction=?, dtSubject=?, dtBIC=?, dtIBAN=?,
    dtAP_V=?,
    dtGetEmail=?, dtLastUpdateTime=now() WHERE dtEmail=?`,
    [`${newMemberData.dtFirstName}`,
      `${newMemberData.dtLastName}`,
      `${newMemberData.dtEmail}`,
      `${newMemberData.dtStreet}`,
      `${newMemberData.dtCity}`,
      `${newMemberData.dtCountry}`,
      `${newMemberData.dtPostalCode}`,
      `${newMemberData.dtSchool}`,
      `${newMemberData.dtFunction}`,
      `${newMemberData.dtSubject}`,
      `${newMemberData.dtBIC}`,
      `${newMemberData.dtIBAN}`,
      `${newMemberData.dtAP_V}`,
      `${newMemberData.dtGetEmail}`,
      `${email}`],
    (err, rows, fields) => {
      if (!err) {
        conn.execute(`INSERT INTO tblMemberUpdateLog (dtFirstName, dtLastName,
dtEmail,
          dtStreet, dtCity, dtCountry, dtPostalCode, dtSchool, dtFunction,
dtSubject, dtBIC,
          dtIBAN, dtAP_V, dtGetEmail, fiMemberId) VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?,
?, ?, ?,
          ?, ?, ?)`,
          [`${newMemberData.dtFirstName}`,
            `${newMemberData.dtLastName}`,

```

```

        `${newMemberData.dtEmail}`,
        `${newMemberData.dtStreet}`,
        `${newMemberData.dtCity}`,
        `${newMemberData.dtCountry}`,
        `${newMemberData.dtPostalCode}`,
        `${newMemberData.dtSchool}`,
        `${newMemberData.dtFunction}`,
        `${newMemberData.dtSubject}`,
        `${newMemberData.dtBIC}`,
        `${newMemberData.dtIBAN}`,
        `${newMemberData.dtAP_V}`,
        `${newMemberData.dtGetEmail}`,
        `${newMemberData.idMember}`],
    (err, rows, fields) => {
        if (cb) cb(err, rows);
    });
    } else if (cb) cb(err, rows);
});
};

this.loginUser = (email, cb) => {
    conn.execute('SELECT * FROM tblMember WHERE dtEmail=?', [`${email}`],
        (err, rows, fields) => {
            if (cb) cb(err, rows);
        });
};

this.updateLoginTimestamp = (email, cb) => {
    conn.execute('UPDATE tblMember SET dtLastLoginTime=now() WHERE dtEmail=?',
    [`${email}`],
        (err, rows, fields) => {
            if (cb) cb(err, rows);
        });
};

this.getMembers = cb => {
    conn.execute('SELECT * FROM tblMember WHERE idMember NOT IN
        (SELECT fiMember FROM tblMembershipDate)', (err, rows, fields) => {
            if (cb) cb(err, rows);
        });
};

this.getPayingMembers = (cb, lastPaidSchoolYear) => {
    conn.execute('SELECT * FROM tblMember WHERE /*(dtAdhesion < '2017-07-1' OR
dtAdhesion IS
    NULL) AND*/ (dtLastPaidSchoolYear IS NULL OR dtLastPaidSchoolYear <>
    ${lastPaidSchoolYear}) AND idMember NOT IN
    (SELECT fiMember FROM tblMembershipDate)', (err, rows, fields) => {
        if (cb) cb(err, rows);
    });
};
};

```

```

    this.getSEPAErrorMembers = cb => {
      conn.execute(`SELECT * FROM tblMember as m, tblSEPAError20182019 as e WHERE
m.dtRef=e.fiRef`,
        (err, rows, fields) => {
          if (cb) cb(err, rows);
        });
    };

    this.insertIPUA = (IP, userAgent) => {
      if (IP && userAgent)
        conn.execute(`INSERT INTO tblIP (dtIP, dtUserAgent) VALUES(?, ?)\`, [IP,
userAgent],
          (err, rows, fields) => {
            if (err) console.log(err);
          });
    };

    this.insertSEPAError = (ID, errorCode, cb) => {
      conn.execute(`INSERT INTO tblSEPAError20182019 (dtErrorCode, fiRef) VALUES(?,
?)\`, [errorCode, ID],
        (err, rows, fields) => {
          if (cb) cb(err, rows);
        });
    };

    const connect = async () => {
      if (!asyncConn) asyncConn = await asyncMysql.createConnection(DBConnectOpts);
    };

    this.getHighestdtRef = async () => {
      if (!asyncConn) await connect();
      return asyncConn.execute(`SELECT MAX(dtRef) /*AS maxRef*/ FROM tblMember`);
    };

    this.getMemberIdsWithoutdtRef = async () => {
      if (!asyncConn) await connect();
      return asyncConn.execute(`SELECT idMember FROM tblMember WHERE dtRef IS NULL`);
    };

    this.updateHighestdtRef = async (id, dtRef) => {
      if (!asyncConn) await connect();
      return asyncConn.execute(`UPDATE tblMember SET dtRef=? WHERE idMember=?`,
['${dtRef}', `${id}']);
    };
  }
}

module.exports = new DB();

```

```

module.exports = class WebSocketServer {
  constructor(server, session) {
    const DEBUG = false;
    const WebSocketServer = require('ws').Server;
    const wss = new WebSocketServer({
      /* verifyClient: (info, done) => {
        if (DEBUG) console.log('Parsing session from request...');
        session(info.req, {}, () => {
          if (DEBUG) console.log('Session is parsed!');
          if (DEBUG) console.dir(info.req.session);
          // We can reject the connection by returning false to done(). For example,
          // reject here if user is unknown.
          done(true);
        });
      },*/
      server
    });

    wss.on('connection', (ws, req) => {
      //console.log(req.connection.remoteAddress);
      //console.dir(req);
      const fs = require('fs');
      const util = require('util');
      const crypto = require('crypto');
      const bcrypt = require('bcryptjs');
      const path = require('path');
      const DB = require(__dirname + '/DB');
      ws.mySend = (msg, options = {}) => {
        const obj = {};
        Object.assign(obj, {msg: msg}, options);
        ws.send(JSON.stringify(obj));
      };

      //ws.upgradeReq = req; // https://github.com/websockets/ws/pull/1099
      if (DEBUG) {
        console.log('***** socket request headers start');
        console.dir(req.headers);
        console.log('***** socket request headers end');
      }
      session(req, {}, () => {
        ws.session = req.session;
        if (DEBUG) console.log('Session created');
        if (req.headers.host === 'apess.lu' || req.headers.host === 'www.apess.lu' ||
          req.headers.referer === 'https://apess.lu/' ||
          req.headers.referer === 'https://www.apess.lu/') {
          if (DEBUG) console.log('A user connected');
          crypto.randomBytes(5, (err, buffer) => {
            if (err) throw err;
            ws.token = buffer.toString('hex');
          });
        }
      });
    });
  }
};

```

```

ws.mySend('token', {
  token: ws.token
});
if (DEBUG) console.log('Token sent');
if (req.session.memberData) {
  const result = {
    success: true,
    error: '',
    memberData: req.session.memberData
  };
  ws.mySend('loggedin', result);
  if (DEBUG) console.log('loggedin sent');
}
});

}
});

ws.on('error', error => {
  if (error) console.err(error);
});

ws.on('close', () => {
  //sessionStore.close();
  if (DEBUG) console.log('User disconnected');
  ws.token = null;
});

ws.on('message', data => {
  data = JSON.parse(data);
  if (data && data.token && data.token === ws.token) {
    switch (data.command) {
      /*case 'register':
        if (data.email && data.password && /^[6,50]$/.test(data.password)) {
          bcrypt.hash(data.password, 10).then(hash => {
            DB.updatePassword(data.email, hash, (err, rows) => {
              if (err) throw err;
            });
            //fs.mkdir(`users/${data.user}`, 0o770, err => {
            //  if (err) throw err;
            //});
            ws.mySend('registered', {
              success: true,
            });
          }).catch(err => {
            if (DEBUG) console.log(`bcrypt error: ${err}`);
          });
        } else {
          if (DEBUG) console.log('invalid registration data');
          ws.mySend('registered', {
            success: false,
          });
        }
      */
    }
  }
}

```

```

    }
    break;*/
case 'login':
    if (data.email && data.password) {
        DB.loginUser(data.email, (err, rows) => {
            if (err) throw err;
            const result = {
                success: true,
                error: '',
                memberData: undefined
            };
            if (rows.length === 1) {
                bcrypt.compare(data.password, rows[0].dtPassword).then(res =>
{
                    if (res) {
                        req.session.memberData = rows[0];
                        //req.session.memberData.IP = req.headers['x-forwarded-
for'];

                        result.memberData = req.session.memberData;
                        req.session.save();
                        if (DEBUG) console.log('----- Login
session start');

                        if (DEBUG) console.dir(req.session);
                        if (DEBUG) console.log('----- Login
session end');

                        DB.updateLoginTimestamp(data.email, (err, rows) => {
                            if (err) throw err;
                        });
                    } else {
                        if (DEBUG) console.log('Wrong password');
                        result.success = false;
                        result.error = 'Invalid login!';
                    }
                    ws.mySend('loggedin', result);
                }).catch(err => {
                    if (DEBUG) console.log(`bcrypt error: ${err}`);
                });
            } else {
                if (DEBUG) console.log('Invalid data received: ');
                if (DEBUG) console.log(data.email + ' ' + data.password);
                result.success = false;
                result.error = 'Invalid login!';
                ws.mySend('loggedin', result);
            }
        });
    }
    break;
case 'logout':
    if (DEBUG) console.log('logout received...');
    delete req.session.memberData;

```

```

    req.session.save();
    /*
    // https://github.com/expressjs/session/issues/425
    const tempSession = req.session;
    req.session.regenerate(err => {
      if (err && DEBUG) console.log('Session could not be regenerated: ' +
err);

      Object.assign(req.session, tempSession);
    });*/
    /*req.session.regenerate(err => {
      if (err && DEBUG) console.log('Session could not be regenerated: ' +
err);

    });*/
    ws.mySend('loggedout');
    break;
  case 'updateMember':
    if (data.email && data.email === req.session.memberData.dtEmail &&
data.newMemberData &&
      /^.{1,50}$/ .test(data.newMemberData.dtFirstName) &&
      /^.{1,50}$/ .test(data.newMemberData.dtLastName) &&
      /^.{6,50}$/ .test(data.newMemberData.dtEmail) &&
      /^.{1,50}$/ .test(data.newMemberData.dtStreet) &&
      /^.{1,50}$/ .test(data.newMemberData.dtCity) &&
      /^[A-Z]-$/ .test(data.newMemberData.dtCountry) &&
      /^[0-9]{4,15}$/ .test(data.newMemberData.dtPostalCode) &&
      /^$|^.{1,50}$/ .test(data.newMemberData.dtSchool) &&
      /^$|^.{1,50}$/ .test(data.newMemberData.dtFunction) &&
      /^$|^.{1,50}$/ .test(data.newMemberData.dtSubject) &&
      /^$|^[A-Za-z]{2}[0-9 ]{18,22}$/ .test(data.newMemberData.dtIBAN) &&
      /^.{1,3}$/ .test(data.newMemberData.dtAP_V) &&
      /^[0-1}$/ .test(data.newMemberData.dtGetEmail) &&
      (!data.newPW || /^.{6,50}$/ .test(data.newPW))
    ) {
      DB.updateMember(data.email, data.newMemberData, (err, rows) => {
        if (data.newPW) {
          bcrypt.hash(data.newPW, 10).then(hash => {
            DB.updatePassword(data.newMemberData.dtEmail, hash, (err,
rows) => {
              if (err) throw err;
            });
          }).catch(err => {
            if (DEBUG) console.log(`bcrypt error: ${err}`);
          });
        }
        const result = {
          success: err ? false : true,
          error: err
        };
        ws.mySend('updatedMember', result);
        if (err) console.dir(err);
        else {

```

```

        req.session.memberData = data.newMemberData; // Update session
member
        // data
        req.session.save();
    }
}
);
} else {
    const result = {
        success: false,
        error: 'Invalid data'
    };
    ws.mySend('updatedMember', result);
}
break;
case 'getFile':
    if (req.session.memberData && fs.existsSync(data.path)) {
        if (DEBUG) console.log(`getFile path: ${data.path}`);
        fs.readFile(data.path, (err, data) => {
            if (DEBUG && err) console.err(err);
            ws.mySend('file', {
                data: data
            });
        })
    } else if (DEBUG) console.log(`Cannot find ${data.path}`);
    break;
case 'test':
    if (DEBUG) console.log(data.msg);
    ws.mySend('test', {
        msg: data.msg
    });
    break;
case 'getMembers':
    if (req.session.memberData)
        DB.getMembers((err, rows) => {
            if (err && DEBUG) console.log(err);
            else ws.mySend('members', {rows: rows});
        });
    break;
case 'getAssets':
    const assets = [];
    for (const asset of data.assets) {
        if (fs.existsSync(asset.file))
            assets.push({
                asset: asset.file, data: fs.readFileSync(asset.file,
                    asset.encoding === 'binary' ? '' : 'utf8')
            });
    }
    ws.mySend('assets', {assets: assets});
    break;

```

```

        case 'reloadMemberData':
            if (req.session.memberData)
                DB.loginUser(req.session.memberData.dtEmail, (err, rows) => {
                    if (err) throw err;
                    const result = {
                        success: true,
                        error: '',
                        memberData: undefined
                    };
                    if (rows.length === 1) {
                        req.session.memberData = rows[0];
                        result.memberData = req.session.memberData;
                        req.session.save();
                    } else {
                        result.success = false;
                        result.error = 'Invalid user!';
                    }
                    ws.mySend('reloadedMemberData', result);
                });
            break;
        }
    }
    else if (DEBUG) console.log('register: Intruder alarm: no token sent!');
}
);
});

wss.on('error', error => {
    if (error) console.err(error);
});
}
};

```

### server.js

```

'use strict';

const DEBUG = false;
const fs = require('fs');
const spdy = require('spdy');
const express = require('express');
const app = express();
const minify = require('express-minify');
const uglifyEs = require('uglify-es');
const helmet = require('helmet');
const DB = require(__dirname + '/DB');
const session = require('express-session');
const MySQLStore = require('express-mysql-session')(session);
const WebSocketServer = require(__dirname + '/WebSocketServer');
const port = 9002; // Port that our server will be listening on. It needs to be opened

```

```

on
// the firewall.
const options = { // Paths to our private key and certificate.
  key: fs.readFileSync('/etc/letsencrypt/live/apess.lu/privkey.pem'),
  cert: fs.readFileSync('/etc/letsencrypt/live/apess.lu/cert.pem')
};
const mysqlSessionOpts = require(__dirname + '/mysql_options').mysqlSessionOpts;
const sessionStore = new MySQLStore(mysqlSessionOpts);
const sessionOpts = {
  cookie: {
    path: '/', domain: 'apess.lu', secure: true, httpOnly: true/*, expires: new
      Date(Date.now() + 86400000)*//* 24
    hours,
    maxAge: new Date(Date.now() + 300000)*/ /* 5 min */
  },
  secret: 'my-secret',
  resave: false,
  saveUninitialized: true,
  name: 'test1.sid',
  key: 'session_cookie_name',
  store: sessionStore,
  secure: true,
  proxy: true,
  sameSite: true
};

process.setuid(1000); // Switch to webdev user as running with root is too dangerous
http://syskall.com/dont-run-node-dot-js-as-root.
app.use(helmet());
const sess = session(sessionOpts);
// https://stackoverflow.com/questions/46295635/how-to-get-ip-address-in-node-js-express
app.disable('x-powered-by'); // https://stackoverflow.com/questions/10717685/how-to-remove-x-powered-by-in-expressjs
app.set('trust proxy', true); // https://expressjs.com/en/guide/behind-proxies.html
app.use(sess);
app.use(minify({
  uglifyJsModule: uglifyEs
}));
app.use(express.static(__dirname + '/assets/public'));
// https://stackoverflow.com/questions/21170253/cannot-use-basic-authentication-while-serving-static-files-using-express/21170931
const isLoggedIn = (req, res, next) => {
  if (DEBUG) console.log('isLoggedIn1');
  if (DEBUG) console.dir(req.session);
  if (req.session.memberData) { // && req.headers['x-forwarded-for'] ===
req.session.memberData.IP)
    if (DEBUG) console.log('isLoggedIn2');
    if (DEBUG) console.dir(req.session);
    return next();
  }
}

```

```

    res.end();
  };
  app.use('/members', [isLoggedIn, express.static(__dirname + '/assets/members')]);
  app.get('/*', (req, res) => {
    /*console.dir(req.ip);
    console.dir(req.ips);*/
    if (req.headers.host === 'apess.lu' || req.headers.host === 'www.apess.lu' ||
        req.headers.referer === 'https://apess.lu/' ||
        req.headers.referer === 'https://www.apess.lu/') {
      if (DEBUG) console.log(req.url);
      const validPublicFileNames = ['/', '/index.html', '/index.css', '/index.babel.js',
        '/polyfill.min.js',
        '/core.min.js', '/reconnecting-websocket.min.js', '/WebSocketClient.babel.js'];
      const pubIdx = validPublicFileNames.indexOf(req.url);
      if (pubIdx > -1 && pubIdx < 2) {
        DB.insertIPUA(req.headers['x-forwarded-for'], req.headers['user-agent']);
        res.sendFile(__dirname + '/index.html');
      }
      else if (pubIdx >= 2 && pubIdx < validPublicFileNames.length) {
        res.sendFile(__dirname + validPublicFileNames[pubIdx]);
        if (DEBUG) console.log(__dirname + validPublicFileNames[pubIdx]);
      }
      else res.end();
      if (DEBUG && req.cookie) console.log(req.cookie);
      if (DEBUG) console.log(req.sessionID);
      if (DEBUG) console.dir(req.session);
      if (req.session) {
        //req.session.userName = 'Bill';
        if (DEBUG) console.dir(req.session);
      } else console.log('app.get NO SESSION!!!');
    } else {
      res.end();
      console.log('Invalid host!');
    }
  });

  const server = spdy.createServer(options, app);

  server.on('listening', () => {
    if (DEBUG) console.log(`Listening on port ${port}...`);
  });

  server.on('connection', () => {
    if (DEBUG) console.log('New connection...');
  });

  server.on('close', () => {
    sessionStore.close();
    if (DEBUG) console.log('Connection closed...');
  });

```

```

server.listen(port, err => {
  if (err) console.log('Server error');
});

const WSS = new WebSocketServer(server, sess);

```

### WebSocketClient.js

```

// This is the WebSocket client controller class, which handles all WebSocket
communication
// but does not touch the GUI.
class WebSocketClient {
  constructor(cb) { // Callback should only be called after server has sent token to
client.
    const DEBUG = false;
    const wsURL = 'wss://apess.lu/wss';
    const ws = new ReconnectingWebSocket(wsURL);
    const messageCallbacks = [];
    ws.token = null;
    ws.addEventListener('open', () => {
      if (DEBUG) console.log('WebSocket connection opened.');
```

```

    });

    ws.addEventListener('message', e => {
      const data = JSON.parse(e.data);
      if (!data.msg) return;
      if (data.msg === 'token') {
        ws.token = data.token;
        cb();
      }
      else for (const cb of messageCallbacks)
        if (cb.msg === data.msg) {
          cb.cb(data);
          break;
        }
    });

    ws.addEventListener('close', () => {
      if (DEBUG) console.log('Disconnected...');
```

```

    });

    ws.addEventListener('error', () => {
      if (DEBUG) console.log('Error...');
```

```

    });

    const execute = (command, options = {}) => {
      let obj = {};
      Object.assign(obj, {command: command}, options);
      ws.send(JSON.stringify(obj));
    };

```

```

this.registerMessageCB = (msg, cb) => {
  messageCallbacks.push({msg: msg, cb: cb});
};

this.deleteAllCBsForMessage = msg => {
  const filter = cb => {
    return cb.msg === msg;
  };
  let idx = messageCallbacks.findIndex(filter);
  while (idx >= 0) {
    messageCallbacks.splice(idx, 1);
    idx = messageCallbacks.findIndex(filter);
  }
};

this.login = (email, pw) => {
  execute('login', {
    token: ws.token,
    email: email,
    password: pw
  });
};

this.register = (email, pw) => {
  if (/^.{6,50}$/.test(pw)) {
    execute('register', {
      token: ws.token,
      email: email,
      password: pw
    });
  } else alert('Invalid registration data!');
};

this.logout = () => {
  execute('logout', {
    token: ws.token
  });
};

this.updateMember = (email, newMemberData, newPW) => {
  execute('updateMember', {
    token: ws.token,
    email: email,
    newMemberData: newMemberData,
    newPW: newPW
  });
};

this.getFile = path => {
  //console.log(`Path: ${path}`);

```

```

        execute('getFile', {
            token: ws.token,
            path: path
        });
    };

    this.test = () => {
        execute('test', {token: ws.token, msg: 'Hello world!'});
    };

    this.getMembers = () => {
        execute('getMembers', {token: ws.token});
    };

    this.getAssets = assets => {
        if (Array.isArray(assets)) execute('getAssets', {token: ws.token, assets:
assets});
    };

    this.reloadMemberData = () => {
        execute('reloadMemberData', {
            token: ws.token
        });
    };
}
}

```

## index.js

```

'use strict';

{
    const init = () => {
        const DEBUG = false;
        const assetPaths = [
            /*{file: 'accueil.html', encoding: 'utf8'},
            {file: 'accueil.css', encoding: 'utf8'},*/
            {file: 'presentatioun.html', encoding: 'utf8'},
            {file: 'presentatioun.css', encoding: 'utf8'},
            {file: 'communiqueen.html', encoding: 'utf8'},
            {file: 'communiqueen.css', encoding: 'utf8'},
            {file: 'editiounen.html', encoding: 'utf8'},
            {file: 'editiounen.css', encoding: 'utf8'},
            {file: 'dossieren.html', encoding: 'utf8'},
            {file: 'dossieren.css', encoding: 'utf8'},
            {file: 'legislatioun.html', encoding: 'utf8'},
            {file: 'legislatioun.css', encoding: 'utf8'},
            {file: 'ressourcen.html', encoding: 'utf8'},
            {file: 'ressourcen.css', encoding: 'utf8'},
            {file: 'interna.html', encoding: 'utf8'},

```

```

    {file: 'interna.css', encoding: 'utf8'},
    {file: 'interna.js', encoding: 'utf8'},
    {file: 'login.html', encoding: 'utf8'},
    {file: 'login.css', encoding: 'utf8'},
    {file: 'profil.html', encoding: 'utf8'},
    {file: 'profil.css', encoding: 'utf8'},
    {file: 'admin.html', encoding: 'utf8'},
    {file: 'admin.css', encoding: 'utf8'}
    /*, {file: 'Apess Statuts 2009.pdf', encoding: 'binary'}*/
  ];
  const assets = [];
  let wsClient = undefined;
  let loggedin = false;
  let memberData = undefined;

  // http://dev.mobify.com/blog/beginners-guide-to-perceived-performance
  document.addEventListener("touchstart", () => {}, true);

  const getAsset = path => {
    for (const asset of assets) if (asset.asset === path) return asset.data;
    return false;
  };

  const toggleNavButton = id => {
    const buttons = document.querySelectorAll('header > nav > ul > li > button');
    for (const button of buttons) button.style.backgroundColor = '';
    const button = document.querySelector(`header > nav > ul > li >
button#${id}`);
    button.style.backgroundColor = 'lightgreen';
  };

  const switchMainDisplay = subject => {
    toggleNavButton(subject);
    const sheets = document.querySelectorAll('style');
    for (const sheet of sheets) document.head.removeChild(sheet);
    const html = getAsset(`${subject}.html`);
    const css = getAsset(`${subject}.css`);
    const js = getAsset(`${subject}.js`);
    document.querySelector('main').innerHTML = '';
    if (html && css) {
      document.querySelector('main').innerHTML = html;
      if (js) {
        const script = document.createElement('script');
        script.innerHTML = js;
        document.body.querySelector('main').appendChild(script);
      }
      const style = document.createElement('style');
      style.id = `${subject}Style`;
      style.innerHTML = css;
      document.head.appendChild(style);
    }
  }

```

```

};

const displayLogin = () => {
  const html = getAsset('login.html');
  const css = getAsset('login.css');
  if (html && css) {
    const loginStyle = document.createElement('style');
    loginStyle.id = 'loginStyle';
    loginStyle.innerHTML = css;
    document.head.appendChild(loginStyle);
    const oldBodyHTML = document.body.innerHTML;
    document.body.innerHTML = html;
    const buttons = document.querySelectorAll('button');
    const loginButton = buttons[0], cancelButton = buttons[1];
    loginButton.addEventListener('click', () => {
      wsClient.login(document.getElementsByName('email')[0].value,
        document.getElementsByName('password')[0].value);
      document.body.innerHTML = oldBodyHTML;
      document.head.removeChild(loginStyle);
      //document.querySelector('button').addEventListener('click',
displayLogin);
      displayMain();
    });
    cancelButton.addEventListener('click', () => {
      document.body.innerHTML = oldBodyHTML;
      document.head.removeChild(loginStyle);
      displayMain();
    });
    const unInput = document.querySelector('input[name=email]');
    if (unInput) {
      unInput.focus();
      unInput.addEventListener('keyup', e => {
        if (e.code === 'Enter') {
          const input = document.querySelector('input[name=password]');
          if (input) input.focus();
        }
      });
    }
    const pwInput = document.querySelector('input[name=password]');
    if (pwInput)
      pwInput.addEventListener('keyup', e => {
        if (e.code === 'Enter' &&
document.querySelector('input[name=email]:valid')) {
          const button = document.querySelector('button[name=login]');
          if (button) button.click();
        }
      });
  }
};

const displayPresentation = () => {

```

```

    switchMainDisplay('presentatioun');
};

const displayCommuniqueen = () => {
    switchMainDisplay('communiqueen');
    /*      PDFJS.getDocument('Communique_22.10.17.pdf').then(pdf => {
        pdf.getPage(1).then(page => {
            const scale = 1.5;
            const viewport = page.getViewport(scale);
            const canvas = document.querySelector('canvas');
            const context = canvas.getContext('2d');
            canvas.height = viewport.height;
            canvas.width = viewport.width;

            const renderContext = {
                canvasContext: context,
                viewport: viewport
            };
            page.render(renderContext);
        });
    });*/
};

const displayEditiounen = () => {
    switchMainDisplay('editiounen');
};

const displayDossieren = () => {
    switchMainDisplay('dossieren');
};

const displayLegislatioun = () => {
    switchMainDisplay('legislatioun');
    /*PDFJS.getDocument('ApessStatuts.pdf').then(pdf => {
        pdf.getPage(1).then(page => {
            const scale = 1.5;
            const viewport = page.getViewport(scale);
            const canvas = document.querySelector('canvas');
            const context = canvas.getContext('2d');
            canvas.height = viewport.height;
            canvas.width = viewport.width;

            const renderContext = {
                canvasContext: context,
                viewport: viewport
            };
            page.render(renderContext);
        });
    });*/
};

```

```

const displayRessourcen = () => {
  switchMainDisplay('ressourcen');
};

const foldMenuButton = () => {
  document.querySelector('#menuButton').classList.remove('active');
  document.querySelector('nav').classList.remove('active');
};

const toggleMenuButton = () => {
  document.querySelector('#menuButton').classList.toggle('active');
  document.querySelector('nav').classList.toggle('active');
};

const displayMain = () => {
  /*const getPic = path => {
    ws.mySend('getFile', {
      token: ws.token,
      path: path,
    });
  };*/
  /*
  else
    if (data.command === 'pic') {
      const buffer = Uint8Array.from(data.data.data);
      const blob = new Blob([buffer]);
      const image = new Image();
      const URL = window.URL || window.webkitURL;
      image.src = URL.createObjectURL(blob);
      document.querySelector('body').appendChild(image);
      URL.revokeObjectURL(image.src);
    }
  */
  /*document.head.appendChild(mainStyle);
  let HTML = '<img src=APESS_logo.jpg><h1>APESS</h1>`;
  document.body.innerHTML = HTML;*/
  const listeners = [/*displayAccueil,*/ displayPresentatioun,
displayCommuniqueen,
  displayEditiounen, displayDossieren, displayLegislatioun, displayRessourcen,
  displayLogin];
  const buttons = document.querySelectorAll('header nav > ul > li > button');
  if (buttons.length === listeners.length)
    for (let i = 0; i < listeners.length; i++)
      buttons[i].onclick = function () {
        foldMenuButton();
        listeners[i]();
      };
  document.querySelector('#menuButton').addEventListener('click',
toggleMenuButton);
};

```

```

const displayInterna = () => {
  switchMainDisplay('interna');
};

const displayProfil = () => {
  foldMenuButton();
  toggleNavButton('profil');
  const sheets = document.querySelectorAll('style');
  for (const sheet of sheets) document.head.removeChild(sheet);
  const html = getAsset('profil.html');
  const css = getAsset('profil.css');
  document.querySelector('main').innerHTML = '';
  if (html && css) {
    document.querySelector('main').innerHTML = html;
    const style = document.createElement('style');
    style.id = `profilStyle`;
    style.innerHTML = css;
    document.head.appendChild(style);
    const firstNameInput = document.querySelector('input[name=firstName]');
    firstNameInput.value = memberData.dtFirstName;
    const lastNameInput = document.querySelector('input[name=lastName]');
    lastNameInput.value = memberData.dtLastName;
    const emailInput = document.querySelector('input[name=email]');
    emailInput.value = memberData.dtEmail;
    const streetInput = document.querySelector('input[name=street]');
    streetInput.value = memberData.dtStreet;
    const cityInput = document.querySelector('input[name=city]');
    cityInput.value = memberData.dtCity;
    const countryInput = document.querySelector('input[name=country]');
    countryInput.value = memberData.dtCountry;
    const postalCodeInput = document.querySelector('input[name=postalCode]');
    postalCodeInput.value = memberData.dtPostalCode;
    const schoolInput = document.querySelector('input[name=school]');
    schoolInput.value = memberData.dtSchool;
    const functionInput = document.querySelector('input[name=function]');
    functionInput.value = memberData.dtFunction;
    const subjectInput = document.querySelector('input[name=subject]');
    subjectInput.value = memberData.dtSubject;
    const adhesionInput = document.querySelector('input[name=adhesion]');
    let val = '';
    if (memberData.dtAdhesion) {
      const d1 = new Date(memberData.dtAdhesion);
      val = d1.getDate() + '.' + (d1.getMonth() + 1) + '.' + d1.getFullYear();
    }
    adhesionInput.value = val;
    const lastPaymentDateInput =
document.querySelector('input[name=lastPaymentDate]');
    val = '';
    if (memberData.dtLastPaymentDate) {
      const d2 = new Date(memberData.dtLastPaymentDate);
      val = d2.getDate() + '.' + (d2.getMonth() + 1) + '.' + d2.getFullYear();
    }
  }
};

```

```

    }
    lastPaymentDateInput.value = val;
    const BICInput = document.querySelector('input[name=BIC]');
    BICInput.value = memberData.dtBIC;
    const IBANInput = document.querySelector('input[name=IBAN]');
    IBANInput.value = memberData.dtIBAN;
    const AP_VInput = document.querySelectorAll('input[name=AP_V]');
    if (memberData.dtAP_V === 'V') AP_VInput[1].checked = true;
    else AP_VInput[0].checked = true;
    //AP_VInput.value = memberData.dtAP_V;
    const getEmailInput = document.querySelector('input[name=getEmail]');
    getEmailInput.checked = memberData.dtGetEmail;
    const pw1Input = document.querySelector('input[name=pw1]');
    const pw2Input = document.querySelector('input[name=pw2]');
    document.querySelector('button[name=save]').onclick = () => {
        // Check values client-side.
        if (firstNameInput.checkValidity() && lastNameInput.checkValidity()
            && emailInput.checkValidity() && streetInput.checkValidity()
            && cityInput.checkValidity() && countryInput.checkValidity()
            && postalCodeInput.checkValidity() && schoolInput.checkValidity()
            && functionInput.checkValidity() && subjectInput.checkValidity()
            && BICInput.checkValidity()
            && IBANInput.checkValidity() /*&& AP_VInput.checkValidity()
            && getEmailInput.checkValidity() */ && pw1Input.checkValidity()
            && pw2Input.checkValidity()) {
            const newMemberData = memberData;
            newMemberData.dtFirstName = firstNameInput.value;
            newMemberData.dtLastName = lastNameInput.value;
            /*newMemberData.dtEmail = emailInput.value;*/
            newMemberData.dtStreet = streetInput.value;
            newMemberData.dtCity = cityInput.value;
            newMemberData.dtCountry = countryInput.value;
            newMemberData.dtPostalCode = postalCodeInput.value;
            newMemberData.dtSchool = schoolInput.value;
            newMemberData.dtFunction = functionInput.value;
            newMemberData.dtSubject = subjectInput.value;
            newMemberData.dtBIC = BICInput.value;
            newMemberData.dtIBAN = IBANInput.value;
            if (AP_VInput[0].checked) newMemberData.dtAP_V = 'APS';
            else newMemberData.dtAP_V = 'V';
            newMemberData.dtGetEmail = getEmailInput.checked ? 1 : 0;
            let newPW = '';
            if (pw1Input.value || pw2Input.value)
                if (pw1Input.value && pw2Input.value && pw1Input.value ===
pw2Input.value)
                    newPW = pw1Input.value;
                else {
                    alert("D'Passwierder stëmmen net iwwereneen");
                    return;
                }
            }
            wsClient.updateMember(memberData.dtEmail, newMemberData, newPW);
        }
    }

```

```

        } else alert('Iwwerpréift w.e.g. Är Daten.');
```

```

        // If OK send them to the server, then check them server-side.
    };
}
};

const displayLoggedInScreen = () => {
    //wsClient.getMembers();
    /*let li = document.querySelector('#ressourcen').parentNode;
    li.parentNode.removeChild(li);*/
    let li = document.querySelector('#login').parentNode;
    li.parentNode.removeChild(li);
    li = document.createElement('li');
    let button = document.createElement('button');
    button.id = 'interna';
    button.innerText = 'Interna';
    li.appendChild(button);
    document.querySelector('ul').appendChild(li);
    li = document.createElement('li');
    button = document.createElement('button');
    button.id = 'profil';
    button.innerText = 'Profil';
    button.addEventListener('click', () => {
        wsClient.reloadMemberData();
    });
    li.appendChild(button);
    document.querySelector('ul').appendChild(li);
    li = document.createElement('li');
    button = document.createElement('button');
    button.id = 'logout';
    button.innerText = 'Ausloggen';
    button.addEventListener('click', e => {
        wsClient.logout();
    });
    li.appendChild(button);
    document.querySelector('ul').appendChild(li);

    const listeners = [/*displayAccueil,*/ displayPresentation,
displayCommuniqueen,
        displayEditionen, displayDossieren, displayLegislation, displayRessourcen,
        displayInterna];
    const buttons = document.querySelectorAll('header nav > ul > li > button');
    for (let i = 0; i < listeners.length; i++)
        buttons[i].onclick = () => {
            foldMenuButton();
            listeners[i]();
        };
    document.querySelector('#menuButton').addEventListener('click',
toggleMenuButton);
};

```

```

const displayMembers = members => {
  if (!members || !Array.isArray(members)) return;
  const keys = Object.keys(members[0]);
  const main = document.querySelector('main');
  const table = document.createElement('table');
  const caption = document.createElement('caption');
  caption.innerText = 'Members';
  table.appendChild(caption);
  const thead = document.createElement('thead');
  let tr = document.createElement('tr');
  for (const key of keys) {
    let th = document.createElement('th');
    th.innerText = key;
    tr.appendChild(th);
  }
  thead.appendChild(tr);
  table.appendChild(thead);
  const tbody = document.createElement('tbody');
  for (const member of members) {
    let tr = document.createElement('tr');
    for (const key of keys) {
      let td = document.createElement('td');
      td.innerText = member[key];
      tr.appendChild(td);
    }
    tbody.appendChild(tr);
  }
  table.appendChild(tbody);
  const css = getAsset('admin.css');
  if (css) {
    const tableStyle = document.createElement('style');
    tableStyle.id = 'tableStyle';
    tableStyle.innerHTML = css;
    document.head.appendChild(tableStyle);
  }
  main.innerHTML = '';
  main.appendChild(table);
};

wsClient = new WebSocketClient(() => {
  wsClient.registerMessageCB('loggedin', data => {
    if (data.success) {
      loggedin = true;
      memberData = data.memberData;
      displayLoggedInScreen();
      switchMainDisplay('presentation');
    } else {
      window.alert('Är Emailadress an/oder Äert Passwuert si net korrekt!');
    }
  });
  wsClient.registerMessageCB('loggedout', data => {

```

```

    window.location = window.location; //reload
  });
  wsClient.registerMessageCB('registered', data => {
    alert('Registered!');
  });
  wsClient.registerMessageCB('test', data => {
    if (DEBUG) console.log(data.msg)
  });
  wsClient.registerMessageCB('file', data => {
    console.dir(data.data.data);
    const showFile = blob => {
      // It is necessary to create a new blob object with mime-type explicitly
      // otherwise only Chrome works like it should
      let newBlob = new Blob([blob], {type: "application/pdf"})

      // IE doesn't allow using a blob object directly as link href
      // instead it is necessary to use msSaveOrOpenBlob
      if (window.navigator && window.navigator.msSaveOrOpenBlob) {
        window.navigator.msSaveOrOpenBlob(newBlob);
        return;
      }

      // For other browsers:
      // Create a link pointing to the ObjectURL containing the blob.
      const data = window.URL.createObjectURL(newBlob);
      let link = document.createElement('a');
      link.href = data;
      link.download = "file.pdf";
      link.click();
      console.dir(link);
      setTimeout(() => {
        // For Firefox it is necessary to delay revoking the ObjectURL
        window.URL.revokeObjectURL(data);
      }, 100);
    };
    showFile(data.data.data);
  });
  wsClient.registerMessageCB('members', data => {
    displayMembers(data.rows);
  });
  wsClient.registerMessageCB('image', data => {
    const buffer = Uint8Array.from(data.data.data);
    //https://stackoverflow.com/questions/21002750/send-pdf-file-using-
    websocket-node-js
    const blob = new Blob([buffer], {type: 'image/jpeg'});
    const image = new Image();
    const URL = window.URL || window.webkitURL;
    image.src = URL.createObjectURL(blob);
    document.querySelector('body').appendChild(image);
    URL.revokeObjectURL(image.src);
  });

```

```

});
wsClient.registerMessageCB('assets', data => {
  for (const arr of data.assets) assets.push(arr);
  if (DEBUG) console.dir(assets);
  if (!loggedin) displayMain();
  switchMainDisplay('presentatioun');
});
wsClient.registerMessageCB('updatedMember', function (result) {
  if (result.error) {
    if (DEBUG) console.log(result.error);
    alert('Är Daten konnten net gespäichert ginn.');
```

}

```

  } else alert('Är Daten goufen gespäichert.');
```

}

```

});
wsClient.registerMessageCB('reloadedMemberData', result => {
  if (result.error) {
    if (DEBUG) console.log(result.error);
  } else {
    memberData = result.memberData;
    displayProfil();
  }
});
wsClient.getAssets(assetPaths);
});
}
;

window.addEventListener('load', init);
}

```

## index.html

```

<!DOCTYPE html>
<html lang=lu>
  <head>
    <meta charset=UTF-8>
    <meta name=viewport content="width=device-width, initial-scale=1">
    <title>APESS</title>
    <link rel=canonical href=https://apess.lu>
    <link rel="apple-touch-icon" sizes="57x57" href="favicon/apple-icon-57x57.png">
    <link rel="apple-touch-icon" sizes="60x60" href="favicon/apple-icon-60x60.png">
    <link rel="apple-touch-icon" sizes="72x72" href="favicon/apple-icon-72x72.png">
    <link rel="apple-touch-icon" sizes="76x76" href="favicon/apple-icon-76x76.png">
    <link rel="apple-touch-icon" sizes="114x114" href="favicon/apple-icon-
114x114.png">
    <link rel="apple-touch-icon" sizes="120x120" href="favicon/apple-icon-
120x120.png">
    <link rel="apple-touch-icon" sizes="144x144" href="favicon/apple-icon-
144x144.png">
    <link rel="apple-touch-icon" sizes="152x152" href="favicon/apple-icon-
152x152.png">

```

```

<link rel="apple-touch-icon" sizes="180x180" href="favicon/apple-icon-
180x180.png">
<link rel="icon" type="image/png" sizes="192x192" href="favicon/android-icon-
192x192.png">
<link rel="icon" type="image/png" sizes="32x32" href="favicon/favicon-32x32.png">
<link rel="icon" type="image/png" sizes="96x96" href="favicon/favicon-96x96.png">
<link rel="icon" type="image/png" sizes="16x16" href="favicon/favicon-16x16.png">
<link rel="manifest" href="favicon/manifest.json">
<meta name="msapplication-TileColor" content="#ffffff">
<meta name="msapplication-TileImage" content="ms-icon-144x144.png">
<meta name="theme-color" content="#ffffff">
<!--<link href=https://fonts.googleapis.com/css?family=Shadows+Into+Light
rel=stylesheet>-->
<!--<link href=https://fonts.googleapis.com/css?family=Pacifico rel=stylesheet>-->
<link href=index.css rel=stylesheet>
<!--
<script>
//https://stackoverflow.com/questions/24861073/detect-if-any-kind-of-ie-msie
var isIE = !!navigator.userAgent.match(/Trident/g) ||
!!navigator.userAgent.match(/MSIE/g);
if (isIE) {
alert("Op Grond vum Asaz vu modernster Webtechnologie funktionéiert dëse Site
net" +
" optimal mam Microsoft Internet Explorer deem e groussen Deel vun deenen" +
" Technologien net ënnerstëtzt. Mam Firefox, Chrome, Safari an anere Browseren
dierft" +
" et keng Problemer ginn.");
}
</script>-->
<!--<script src=polyfill.min.js></script>-->
<script src=core.min.js></script>
<script src=reconnecting-websocket.min.js></script>
<script src=WebSocketClient.babel.js></script>
<!--<script src=pdf.min.js></script>-->
<script src=index.babel.js></script>
</head>
<body>
<noscript><h1>Dëse Site funktionéiert net ouni JavaScript!</h1></noscript>
<header>
<!--<img src=APESS_logo.jpg
alt="Association des Professeurs de l'Enseignement Secondaire et Supérieur"
title="Association des Professeurs de l'Enseignement Secondaire et
Supérieur"
width=171 height=79>-->
<section id=header>
<div>APESS</div>
<div>Association des Professeurs de l'Enseignement Secondaire et Supérieur du
Grand-Duché de Luxembourg, association sans but lucratif</div>
<div><a
href="#109;&#x61;&#x69;&#108;&#116;&#111;&#58;&#x61;&#x70;&#101;&#115;&#115;&#x40;&#1
01;&#x64;&#117;&#99;&#x61;&#x74;&#x69;&#x66;&#110;&#x2e;&#x6c;&#117;">&#97;&#112;&#x65

```

```

;#x73;#x115;#x40;#x65;#x64;#x117;#x63;#x61;#x74;#x69;#x6f;#x110;#x2e;#x108;#x117;</a>
</div>
</section>
<section id=menu><button id=menuButton></button><span>APESS</span></section>
<nav>
  <ul>
    <!--<li>
      <button id=accueil>Accueil</button>
    </li>-->
    <li>
      <button id=presentation>Presentation</button>
    </li>
    <li>
      <button id=communiquen>Communiqu  n a Press</button>
    </li>
    <li>
      <button id=editionen>Editionen</button>
    </li>
    <li>
      <button id=dossieren>Dossieren</button>
    </li>
    <li>
      <button id=legislation>Legislation</button>
    </li>
    <li>
      <button id=ressourcen>Ressourcen</button>
    </li>
    <li>
      <button id=login>Aloggen</button>
    </li>
  </ul>
</nav>
</header>
<main>
</main>
</body>
</html>

```

index.css

```

html { /* http://www.paulirish.com/2012/box-sizing-border-box-ftw */
  box-sizing: border-box;
}

*, *:before, *:after {
  box-sizing: inherit;
}

::-moz-focus-inner {

```

```
border: 0; /* Remove dotted line around clicked button */
}

/* .email {
  unicode-bidi: bidi-override;
  direction: rtl;
}*/

html, body {
  width: 100%;
  height: 100%;
  margin: 0;
}

body {
  display: flex;
  flex-flow: column;
  overflow: hidden;
  hyphens: auto;
  font-family: Helvetica, sans-serif;
  /* font-family: 'Pacifico', cursive;*/
}

main {
  margin: 5px;
  padding: 5px;
  /*padding-right: 15px;*/
  overflow: auto;
  text-align: justify;
}

img {
  /*max-width: 100%;
  height: auto;*/
  /*max-height: 100%;
  width: 100%;
  height: auto;*/
  margin: auto;
}

h1 {
  text-align: left;
}

header > nav > ul {
  list-style: none;
  padding: 0;
  margin: 0;
  display: flex;
  flex: auto;
  flex-wrap: wrap;
```

```

}

header > nav > ul > li {
  flex:      auto;
  text-align: center;
  padding:   2px;
}

input, button {
  font-family: inherit;
}

header > nav > ul > li > button {
  width:      100%;
  border:     none;
  background-color: rgb(240, 240, 240);
  padding:    2px;
  font-size:  1.3em;
  box-shadow: 4px 4px 4px #888888;
  font-family: inherit;
}

header > nav > ul > li > button:focus, header nav > ul > li > button:active {
  border: none;
}

header > nav > ul > li > button:hover {
  background-color: grey;
}

@media all and (min-width: 571px) and (min-height: 401px) {
  #header {
    display: flex;
    /*font-family: 'Pacifico', cursive;*/
  }

  #header > div {
    flex:      auto;
    padding-right: 10px;
    padding-left: 10px;
  }

  #header > div {
    text-align: center;
    margin:     auto;
  }

  #header > div:first-child {
    font-size:  2em;
    color:      rgb(255, 50, 50);
    text-shadow: 5px 5px 5px grey;
  }

```

```

}

header > nav {
  width: 100%;
}

header {
  /*min-height: 80px;*/
}

#menu {
  display: none;
}
}

/*@media all and (max-width: 700px) {
  header > img {
    display: none;
  }
}*/

@media all and (max-width: 570px), (max-height: 400px) {
  #header {
    display: none;
  }

  #menu {
    /*font-family: 'Pacifico', cursive;*/
    /*font-size: 2em;*/
  }

  #menu > span {
    display: inline-block;
    width: calc(100vw - 160px);
    text-align: center;
    color: rgb(255, 50, 50);
    text-shadow: 5px 5px 5px grey;
    font-size: 1.5em;
  }

  #menuButton {
    /*display: block;*/
    padding: 0 0.2em 0.2em 0.2em;
    background-color: rgb(240, 240, 240);
    text-decoration: none;
    color: #333;
    cursor: pointer;
    font-size: inherit;
    width: 10em;
    height: 3em;
  }
}

```

```

#menuButton.active {
  background-color: #333;
  color:          #fff;
}

header {
  max-height: 100%;
}

header > nav {
  display: none;
}

header > nav.active {
  /* https://stackoverflow.com/questions/27992881/max-height-x-doesnt-work-on-chrome
*/
  /* vh -> https://css-tricks.com/viewport-sized-typography/ */
  max-height: calc(100vh - 48px); /* 48px is height of menu section */
  display:    block;
  overflow:   auto;
}

header > nav.active > ul {
  display:    block;
  max-height: calc(100vh - 27.2px);
}

header > nav.active > ul > li {
  display: block;
}
}

/*@media all and (max-height: 404.2px) {
  body {
    display: block;
  }
  header {
    overflow: hidden;
  }

  header > nav.active {
    flex: auto;
    overflow: auto;
    min-height: 368px;
    height: 368px;
  }

  header > nav, header > nav > ul {
    overflow: auto;
  }
}

```

```

main {
    display: none;
}
}*/

```

## dossieren.html

```

<article><h1>Ofschafe vun de Proffecomitéen</h1>
D'<a href=http://legilux.public.lu/eli/etat/leg/loi/2004/06/25/n9/jo
target=_blank>Gesetz
    vum 25.6.04</a> iwwert d'Organisatioun vun de Lycéeën an technesche Lycéeën gouf am
<a href=http://legilux.public.lu/eli/etat/leg/loi/2016/12/15/n1/jo
target=_blank>Gesetz
    vum 15.12.16</a> geännert (cf. Säit 5 Artikelen 9 an 10 resp. Säit 7 Artikel 33 am
ale
Gesetz).

```

```

Dat Gesetz ass säit dem 25.12.16 a Kraaft, legal gëtt et also keng Proffecomitéen méi.
</article>
<article>
    <a href=http://www.men.public.lu/fr/actualites/articles/communiqués-conference-
presse/2017/10/20-manuels-gratuits
        target=_blank><h1>Gratis Schoulbicher fir d'Lycéesschüler vun der Rentrée 2018-
2019 un</a>
</article>

```

## dossieren.css

```

main > article > h1 {
    text-align: center;
}

/*main > article > a {
    text-decoration: none;
}*/

main > article {
    background-color: rgb(240, 240, 240);
    padding: 5px;
    margin: 0 10px 20px;
    box-shadow: 10px 10px 10px black;
}

main > article > img {
    display: block;
}

```

```
<section>
  <header>
    <h1>Cahiers pédagogiques</h1>
  </header>
</div>
<article>
  <img src=editiounen/cp/CP1.png width=230 height=155
    alt='Cahier pédagogique 1 - Homo Faber'>
  <h1>Cahier pédagogique 1 - Homo Faber</h1>
  6e édition revue et corrigée, 136 p.<br>
  ISBN 978-2-87979-146-3<br>
  prix : 11 €<br>
  emballage pour les librairies : cartons à 32 exemplaires
</article>
<article>
  <img src=editiounen/cp/CP7.jpg width=87 height=123
    alt='Cahier pédagogique 7 - Allemand VIIe Ve'>
  <h1>Cahier pédagogique 7 - Allemand VII<sup>e</sup> V<sup>e</sup></h1>
  Exercices (238 p.) et corrigés (142 p.)<br>
  ISBN 978-2-87979-070-1<br>
  prix : 33 €<br>
  emballage pour les librairies : cartons à 18 exemplaires
</article>
<article>
  <img src=editiounen/cp/CP8.png width=230 height=155
    alt='Cahier pédagogique 8 - Mathématiques Ve'>
  <h1>Cahier pédagogique 8 - Mathématiques V<sup>e</sup></h1>
  ISBN 2-87979-017-4<br>
  prix : 15 €
</article>
<article>
  <img src=editiounen/cp/CP12.jpg width=88 height=123
    alt='Cahier pédagogique 12 - Philosophie'>
  <h1>Cahier pédagogique 12 - Philosophie</h1>
  ISBN 978-2-87979-072-5<br>
  prix : 19 €<br>
  emballage pour les librairies : cartons à 32 exemplaires
</article>
<article>
  <img src=editiounen/cp/CP248.png width=230 height=155
    alt='Cahier pédagogique 248 - Mathématiques VIIe Ve'>
  <h1>Cahier pédagogique 248 - Mathématiques VII<sup>e</sup> V<sup>e</sup></h1>
  Exercices (128 p.) et corrigés (216 p.)<br>
  ISBN 978-2-87979-248-4<br>
  prix : 33 €<br>
  emballage pour les librairies : cartons à 19 exemplaires
</article>
<article>
  <img src=editiounen/cp/CP356.png width=230 height=155
```

```

    alt='Cahier pédagogique 356 - Grammaire française VIIe-IVe'>
    <h1>Cahier pédagogique 356 - Grammaire française
VII<sup>e</sup>-IV<sup>e</sup></h1>
    Exercices (320 p.) et corrigés (104 p.)<br>
    ISBN 2-87979-018-2<br>
    prix : 33 €<br>
    emballage pour les librairies : cartons à 16 exemplaires
  </article>
</div>
</section>
<section>
  <header>
    <h1>Ausbléck - Récré</h1>
    <h2>Annuaire culturel des professeurs luxembourgeois</h2>
  </header>
  <div>
    <article>
      <img src=editiounen/recre/Aus30.png width=211 height=299 alt='Ausbléck-Récré
30'>
      <h1>Ausbléck-Récré 30</h1>
      209 p., 2019<br>
      ISBN 978-2-87979-130-2<br>
      <a href=editiounen/recre/ExtraitAus30.pdf target=_blank>Table des matières et
préface</a>
    </article>
    <article>
      <img src=editiounen/recre/Aus29.jpg width=211 height=300 alt='Ausbléck-Récré
29'>
      <h1>Ausbléck-Récré 29</h1>
      260 p., 2016<br>
      ISBN 978-2-87979-129-6<br>
      <a href=editiounen/recre/ExtraitAus29.pdf target=_blank>Table des matières et
préface</a>
    </article>
    <article>
      <img src=editiounen/recre/Aus28.jpg width=212 height=300 alt='Ausbléck-Récré
28'>
      <h1>Ausbléck-Récré 28</h1>
      326 p., 2014<br>
      ISBN 978-2-87979-128-9<br>
      <a href=editiounen/recre/ExtraitAus28.pdf target=_blank>Table des matières et
préface</a>
    </article>
    <article>
      <img src=editiounen/recre/Aus27.jpg width=214 height=300 alt='Ausbléck-Récré
27'>
      <h1>Ausbléck-Récré 27</h1>
      344 p., 2013<br>
      ISBN 978-2-87979-127-2<br>
      <a href=editiounen/recre/ExtraitAus27.pdf target=_blank>Table des matières et
préface</a>

```

```

</article>
<article>
  <img src=editiounen/recre/Aus26.jpg width=211 height=300 alt='Ausbléck-Récré
26'>
  <h1>Ausbléck-Récré 26</h1>
  274 p., 2011<br>
  ISBN 978-2-87979-126-5<br>
  <a href=editiounen/recre/Aus26intro.pdf target=_blank>Table des matières et
  avant-propos</a>
</article>
<article>
  <img src=editiounen/recre/recres.png width=230 height=155 alt='Récré 1 ò 25'>
  <h1>Récré 1 ò 25</h1>
  1986-2010<br>
  Tous les numéros sont encore disponibles.
</article>
</div>
</section>
<section>
  <header>
    <h1>Collection</h1>
  </header>
  <div>
    <article>
      <img src=editiounen/collection/LaubUndNadel.jpg width=230 height=155
      alt=LaubUndNadel.jpg>
      <h1>Laub und Nadel</h1>
      Roland Harsch, 2012 (nouvelle édition), Collection 11<br>
      ISBN 978-2-87979-211-8
    </article>
    <article>
      <img src=editiounen/collection/DuTourDeFrantzAuTourDeGaul.jpg width=230
height=155
      alt=DuTourDeFrantzAuTourDeGaul.jpg>
      <h1>Du Tour de Frantz au Tour de Gaul</h1>
      François Guillaume, 2006 (nouvelle édition), Collection 14 (ancienne 13)<br>
      ISBN 2-87979-202-9
    </article>
    <article>
      <img src=editiounen/collection/PARODIESUndDAS.jpg width=230 height=155
      alt=PARODIESUndDAS.jpg>
      <h1>PARODIES und DAS</h1>
      Roland Harsch, 2004, Collection 14<br>
      ISBN 2-87979-214-2
    </article>
    <article>
      <img src=editiounen/collection/LettresÀSophie.jpg width=230 height=155
      alt=LettresÀSophie.jpg>
      <h1>Lettres à Sophie</h1>
      Claude Conter, 2002, Collection 12<br>
      ISBN 2-87979-212-6
    </article>
  </div>
</section>

```

```

</article>
<article>
  <img src=editiounen/collection/Chroniques1961-1997.jpg width=230 height=155
    alt=Chroniques1961-1997.jpg>
  <h1>Chroniques 1961-1997</h1>
  Nic Klecker, 1998, Collection 10<br>
  ISBN 2-87979-210-X
</article>
<article>
  <img src=editiounen/collection/SatirenUndGlossenII.jpg width=230 height=155
    alt=SatirenUndGlossenII.jpg>
  <h1>Satiren und Glossen II</h1>
  Henry Gelhausen, 1997, Collection 9<br>
  ISBN 2-87979-209-6
</article>
<article>
  <img src=editiounen/collection/Kalendarium.jpg width=230 height=155
alt=Kalendarium.jpg>
  <h1>Kalendarium</h1>
  Roland Harsch, 1997, Collection 8<br>
  ISBN 2-87979-208-8
</article>
<article>
  <img src=editiounen/collection/MusikalischeFederspiele.jpg width=230 height=155
    alt=MusikalischeFederspiele.jpg>
  <h1>Musikalische Federspiele</h1>
  Roland Harsch, Carlo Schmitz, 1996, Collection 7<br>
  ISBN 2-87979-207-X
</article>
<article>
  <img src=editiounen/collection/DerBürgerImStaatII.jpg width=230 height=155
    alt=DerBürgerImStaatII.jpg>
  <h1>Der Bürger im Staat II</h1>
  Marcel Engel, 1995, Collection 6<br>
  ISBN 2-87979-206-1
</article>
<article>
  <img src=editiounen/collection/SatirenUndGlossen.jpg width=230 height=155
    alt=SatirenUndGlossen.jpg>
  <h1>Satiren und Glossen</h1>
  Henry Gelhausen, 1994, Collection 5<br>
  ISBN 2-87979-205-0
</article>
<article>
  <img src=editiounen/collection/LesFleursOntFroid.jpg width=230 height=155
    alt=LesFleursOntFroid.jpg>
  <h1>Les fleurs ont froid</h1>
  Rosemarie Kieffer, 1993, Collection 4<br>
  ISBN 2-87979-204-5
</article>
<article>

```

```

<img src=editiounen/collection/Chroniques1961-1997.jpg width=230 height=155
    alt=Chroniques1961-1997.jpg>
<h1>Chroniques de J.-M- Durand</h1>
Léon Thyès, 1991, Collection 3<br>
ISBN 2-87979-203-7
</article>
<article>
    <img src=editiounen/collection/DieStreuweise.jpg width=230 height=155
        alt=DieStreuweise.jpg>
    <h1>Die Streuweise</h1>
    Fernand Hoffmann, 1988, Collection 2<br>
    ISBN 2-87979-202-9
</article>
<article>
    <img src=editiounen/collection/DerBürgerImStaat.jpg width=230 height=155
        alt=DerBürgerImStaat.jpg>
    <h1>Der Bürger im Staat</h1>
    Marcel Engel, 1987, Collection 1<br>
    ISBN 2-87979-201-0
</article>
</div>
</section>
<footer>
    <h1>Kontakt</h1>
    <h2>Administrateur</h2>
    Pascal Zeihen
    Tel. 691 273 769
    <h2>Comité de rédaction</h2>
    Franck Colotte (President), Roland Harsch, Edouard Kayser, Paul Kintziger, Georges
    Milmeister
    <h1>Bestellungen</h1>
    <h2>Librairien (an alphabetescher Reiefolleg)</h2>
    <ul>
        <li><a href=http://www.ernster.com target=_blank>Ernster</a> Bartreng a
        Lëtzebuerg</li>
        <li><a href=http://www.libo.lu target=_blank>LIBO</a> Lëtzebuerg, Dikrech,
        Ettelbréck,
            Wolz a Gréiwemaacher
        </li>
        <li>Librairie A.B.C. À la Bouquinerie du Centre, 40 avenue de la Gare, L-4130
        Esch/Alzette,
            Tel. 53 05 82
        </li>
        <li>Librairie Alinéa, 5 rue Beaumont, L-1219 Luxembourg, Tel. 22 67 87</li>
        <li>Librairie des Lycées, 30 av. Victor Hugo, L-1750 Luxembourg, Tel. 22 79
        83</li>
        <li><a href=http://www.diderich.lu target=_blank>Librairie Diderich</a>
        Esch/Alzette</li>
        <li>Librairie um Fieldgen, 3 rue Glesener, L-1631 Luxembourg, Tel. 48 88 93</li>
        <li>Librairie Zimmer, 30-32 Grand-Rue, L-9240 Diekirch, Tel. 80 95 59</li>
        <li><a href=http://www.samkats.com target=_blank>Samkats</a> Iechternach</li>
    </ul>

```

```

</ul>
<h2>Direkt Bestellungen</h2>
Eis Memberen profitéieren vu reduzéierte Präiser a bezuele keng Versandkäschen.
<a href=mailto:editions@apess.lu?subject=Commande target=_blank>Kontaktéiert
eis</a>.
</footer>

```

## editiounen.css

```

main {
  /*overflow: hidden;*/
}

main > section > div {
  display:          grid;
  grid-template-columns: 1fr 1fr 1fr 1fr 1fr;
  /*grid-template-rows: 1fr 5fr 5fr;*/
  /*align-items:      stretch;*/
  /*overflow:          auto;
  max-height:         100%; calc(100vh - 150px);*/
}

main > nav > ul {
  list-style: none;
  padding: 0;
  margin: 0;
  display: flex;
  flex: auto;
  flex-wrap: wrap;
}

main > nav > ul > li {
  flex: auto;
  text-align: center;
  padding: 2px;
}

main > nav > ul > li > button {
  width: 100%;
  border: none;
  background-color: rgb(200, 200, 200);
  padding: 2px;
  font-size: 1.3em;
  box-shadow: 4px 4px 4px #888888;
  font-family: inherit;
}

main > nav > ul > li > button:focus, header nav > ul > li > button:active {
  border: none;
}

```

```
main > nav > ul > li > button:hover {
  background-color: grey;
}

main > section > header {
  grid-column: span 5;
  grid-row: span 1;
  margin: 0;
}

main > section > div > article {
  background-color: rgb(240, 240, 240);
  padding: 5px;
  margin: 0 10px 20px;
  box-shadow: 10px 10px 10px black;
  display: flex;
  flex-flow: column;
  text-align: center;
  float: left;
}

main > section > div > article > img {
  margin: auto;
  /*max-height: 100%;
  max-width: 100%;*/
  width: initial;
  height: initial;
}

main > section > div > article > h1 {
  text-align: center;
}

@media all and (max-width: 679px) {
  main > section > div {
    grid-template-columns: 1fr;
  }

  main > section > header {
    grid-column: span 1;
  }
}

@media all and (min-width: 680px) {
  main > section > div {
    grid-template-columns: 1fr 1fr;
  }

  main > section > header {
    grid-column: span 2;
  }
}
```

```

}
}

@media all and (min-width: 1006px) {
  main > section > div {
    grid-template-columns: 1fr 1fr 1fr;
  }

  main > section > header {
    grid-column: span 3;
  }
}

@media all and (min-width: 1338px) {
  main > section > div {
    grid-template-columns: 1fr 1fr 1fr 1fr;
  }

  main > section > header {
    grid-column: span 4;
  }
}

@media all and (min-width: 1662px) {
  main > section > div {
    grid-template-columns: 1fr 1fr 1fr 1fr 1fr;
  }

  main > section > header {
    grid-column: span 5;
  }
}

@media all and (min-width: 571px) and (min-height: 401px) {
  main > nav {
    width: 100%;
  }

  #editiounenMenuButton {
    display: none;
  }
}

@media all and (max-width: 570px), (max-height: 400px) {
  #editiounenMenuButton {
    padding: 0 0.2em 0.2em 0.2em;
    background-color: rgb(240, 240, 240);
    text-decoration: none;
    color: #333;
    cursor: pointer;
    font-size: inherit;
  }
}

```

```

width:          10em;
height:         3em;
}

#editiounenMenuButton.active {
  background-color: #333;
  color:          #fff;
}

main > nav {
  display: none;
}

main > nav.active {
  max-height: calc(100% - 48px);
  display:    block;
  overflow:   auto;
}

main > nav.active > ul {
  display:    block;
  max-height: calc(100% - 27.2px);
}

main > nav.active > ul > li {
  display: block;
}
}

```

### interna.js

```

document.getElementById('delegéiert').addEventListener('click', () => {
  const table = document.querySelector('#delegéiert table');
  if (!table) return;
  if (table.style.display && table.style.display !== 'none') table.style.display =
'none';
  else table.style.display = 'table';
});

```

### interna.html

```

<aside>
  Wann Dir eis wëllt e vertrauleche Mail schécken a wëllt sécher sinn, datt keen
  aneren e
  liese kann, da benotzt w.e.g. d'Mailadress <a
href="#x6d;x61;x69;x6c;x74;x111;x3a;x61;x112;x65;x73;x73;x64;x
70;x72;x6f;x116;x111;x110;x6d;x61;x69;x6c;x2e;x99;x6f;x109;">x6
1;x70;x65;x73;x115;x64;x70;x72;x6f;x74;x6f;x110;x109;x61;x69;x
#108;x46;x99;x6f;x109;</a>.

```

Dir dierft Äre Mail

dann awer net vun enger education.lu Adress schécken mä vun enger, déi net ënner d'Kontroll vum MENJE fält. Am séchersten geet dat, wann Dir Iech och e Gratiskont

op

[https://protonmail.com](https://protonmail.com target=_blank) erstellt, dann ass d'komplett Kommunikatioun verschlüsselt an och protonmail selwer kënnen Är Mailen net liesen.

</aside>

<article><h1>Instruction ministérielle ES 2019-4 du 9 octobre 2019

concernant l'organisation des épreuves orales en classe de première</h1></article>

</a>

<article><h1>Instruction ministérielle du 11 juillet 2019

concernant l'organisation scolaire des lycées</h1></article>

</a>

<section>

<article class=noflex><h1>Legalitéit vun der Publikatioun vun de Memberen vun der Commission

d'experts

chargés d'examiner les sujets ou questions proposés aux différentes épreuves de l'examen de

fin d'études secondaires de l'enseignement secondaire classique, sessions de l'année 2019

3.6.19</h1>

No enger Ufro vun engem Member hu mir der Commission nationale pour la protection des

données (CNPD) folgend Fro gestallt:

"Pouvez-vous nous confirmer, que la publication des noms des membres de la commission

d'experts chargés d'examiner les sujets ou questions proposés aux différentes épreuves de

l'examen de fin d'études secondaires de l'enseignement secondaire classique, sessions de

l'année 2019 par arrêté ministériel

(<a href=http://legilux.public.lu/eli/etat/adm/amin/2019/03/30/b1231/jo

target=\_blank>http://legilux.public.lu/eli/etat/adm/amin/2019/03/30/b1231/jo</a>) n'enfreint pas la loi relative à la protection des données ?"

<br><br>

Mir kruten déi heiten Äntwert:<br>

"Après analyse, nous avons pu constater que les articles 3

du règlement grand-ducal modifié du 31 juillet 2006 portant organisation de l'examen de fin

d'études secondaires classiques et le règlement grand-ducal modifié du 31 juillet 2006

portant organisation de l'examen de fin d'études secondaires générales prévoient que

« l'examen a lieu devant des commissions nommées chaque année par le ministre ».

<br><br>

Dans ce contexte, il nous semble que la publication des noms et prénoms des membres de ces

commissions peut être considérée comme nécessaire dans le cadre de la mission d'intérêt

public poursuivie par le Ministre de l'Éducation nationale, de l'Enfance et de la Jeunesse,

conformément à ces dispositions. Une telle publication répond par ailleurs à une finalité

de transparence, alors que tout arrêté ministériel doit être publié au Journal officiel du

Grand-Duché de Luxembourg.

<br><br>

Par conséquent, la publication des noms et prénoms des membres de la commission d'experts

chargés d'examiner les sujets ou questions proposés aux différentes épreuves de l'examen de

fin d'études secondaires de l'enseignement secondaire classique, sessions de l'année 2019,

nous paraît répondre à la condition de licéité de l'article 6, paragraphe (1), lettre (e),

du règlement général sur la protection des données (UE) 2016/679 (ou « RGPD »). Nous ne

voyons par ailleurs a priori pas en quoi la publication des noms et prénoms des membres de

ces commissions enfreindrait d'autres dispositions du RGPD."

</article>

</section>

<section>

<article><h1>AGO vum 14.3.19</h1>

<h2><a href="members/pdf/APESS - Rapport de l'AGO du 14 mars 2019.pdf" target=\_blank>

Rapport

</a></h2>

<h2><a href='members/pdf/Assemblée générale ordinaire 2019.pdf' target=\_blank>

Presentatioun

</a></h2>

<h2><a href='members/pdf/APESS Rapport financier 2018.pdf' target=\_blank>

Finanzrapport 2018

</a></h2>

</article>

</section>

<section>

<article><h1>Kommunikatioun zum Bewäertungssystem 22.11.18</h1>

<h2><a href='members/pdf/Avis juridique Bewäertungssystem 23.11.18.pdf' target=\_blank>

Avis juridique

</a></h2>

</article>

</section>

```
<a href='members/pdf/2018-07-11 IM organisation scolaire VF_signée.pdf'
  target=_blank>
  <article><h1>Instruction ministérielle du 11 juillet 2018
    concernant l'organisation scolaire des lycées</h1></article>
</a>
<a>
  <article>
    <h1>Delegéiert</h1>
    <table>
      <tr>
        <th>Numm</th>
        <th>Schoul</th>
      </tr>
      <tr>
        <td>Marco BREYER</td>
        <td>AL</td>
      </tr>
      <tr>
        <td>Gilles EVERLING</td>
        <td>LAM</td>
      </tr>
      <tr>
        <td>Myriam WAGNER ép. ENGEL</td>
        <td>LAML</td>
      </tr>
      <tr>
        <td>Pascal ZEIHEN</td>
        <td>LCD</td>
      </tr>
      <tr>
        <td>Jacques DELLERÉ</td>
        <td>LCE</td>
      </tr>
      <tr>
        <td>Jean-Louis WEIS</td>
        <td>LGL</td>
      </tr>
      <tr>
        <td>Jean-Marc CIMA</td>
        <td>LJBM</td>
      </tr>
      <tr>
        <td>Emmanuel BOCK</td>
        <td>LN</td>
      </tr>
      <tr>
        <td>Fabien HENGEN</td>
        <td>LTB</td>
      </tr>
      <tr>
        <td>Lelio LOEWEN</td>
```

```

        <td>LTC</td>
    </tr>
    <tr>
        <td>Paulo DE SOUSA</td>
        <td>LTMA</td>
    </tr>
    <tr>
        <td>Guy RHEIN</td>
        <td>MLG</td>
    </tr>
</table>
</article>
</a>
<a href='members/pdf/Traitements_Fonctionnaires_Enseignement_version_2.1.20180703.pdf'
    target=_blank>
    <article><h1>Carrières et rémunérations applicables aux fonctionnaires (-stagiaires)
    engagés depuis la réforme du 1er octobre 2015</h1></article>
</a>
<a href='members/pdf/FICHE_OPERA_PLUS_FR-APESS.pdf' target=_blank>
    <article><h1>Raiffeisen Opera Plus Offer fir Memberen</h1></article>
</a>
<section>
    <article><h1>AGO an AG vum 17.5.18</h1>
        <h2><a href='members/pdf/Rapport_AG_17.5.18.pdf' target=_blank>
            Rapport
        </a></h2>
        <h2><a href='members/pdf/Assemblée_générale_ordinaire_2018.pdf' target=_blank>
            Presentatioun
        </a></h2>
        <h2><a href='members/pdf/APESS_Rapport_financier_2017.pdf' target=_blank>
            Finanzrapport 2017
        </a></h2>
        <h2><a href='members/pdf/APESS_Rapport_financier_2016.pdf' target=_blank>
            Finanzrapport 2016
        </a></h2>
        <h2>
            <a href='members/pdf/Homologatioun.pdf' target=_blank>
                Homologatioun vum der Statutenännerung</a>
        </h2>
    </article>
</section>

```

interna.css

```

/*table {
    border-collapse: collapse;
    box-shadow:      2px 2px 2px #888888;
}

table, tr, th, td {

```

```

border: 4px groove black;
}

th, td {
padding: 5px;
}
*/

main > aside {
background: linear-gradient(to bottom right, #550000, red);
padding: 5px;
margin: 0 10px 20px;
box-shadow: 10px 10px 10px black;
color: gold;
}

main > aside > a {
color: lightblue;
}

main article > h1, main article h2 {
text-align: center;
}

main article h2 {
margin: 5px;
}

main a {
text-decoration: none;
}

main article {
background: linear-gradient(to bottom right, lightgray, gray);
padding: 5px;
margin: 0 10px 20px;
box-shadow: 10px 10px 10px black;
display: flex;
flex-flow: column;
}

main article.noflex {
display: block;
}

main article:hover {
background: linear-gradient(to bottom right, darkslategray, lightgray);
}

main article > img {
display: block;
}

```

```

max-width: 100%;
height: auto;
/*max-width: initial;
max-height: initial;*/
}

main article {
display: flex;
align-items: center;
}

main article table, main article table tr, main article table th, main article table
td {
padding: 5px;
color: lightyellow;
text-shadow: 2px 2px 2px black;
}

/*@media all and (min-width: 571px) and (min-height: 401px) {
article {
transition: 5s;
}
}*/

```

## legislatioun.html

```

<!--https://stackoverflow.com/questions/291813/recommended-way-to-embed-pdf-in-
html?rq=1-->
<!--<h1>Statuten, Gesetzer, Reglementer an Instruktiounen</h1>-->
<section>
<a href='pdf/Statuts approuvés le 17.5.18.pdf' target=_blank>
<article>
<div>APESS-Statuten</div>
<img
src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
</a>
<a
href=http://www.mj.public.lu/legislation/asbl_fondations/2009_Loi_21_avril_1928.pdf
target=_blank class=law>
<article>
<div>A.s.b.l.-Gesetz vum 21.4.1928</div>
<img src=biller/if_law_298810.svg
alt=if_law_298810.svg width=32
height=32></article>
</a>
<a href=http://legilux.public.lu/eli/etat/leg/loi/2002/08/02/n2/jo
target=_blank class=law>
<article>
<div>Dateschutzgesetz vum 2.8.2</div>

```

```

        <img src=biller/if_law_298810.svg
            alt=if_law_298810.svg width=32
            height=32></article>
    </a>
    <a href=http://legilux.public.lu/eli/etat/leg/code/education_nationale
target=_blank>
        <article>
            <div>Code de l'éducation nationale
            </div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
            height=32></article>
    </a>
    <a href=http://legilux.public.lu/eli/etat/leg/code/fonction_publique target=_blank>
        <article>
            <div>Code de la fonction publique
            </div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
            height=32></article>
    </a>
    <a href=http://legilux.public.lu/eli/etat/leg/code/travail target=_blank>
        <article>
            <div>Code du travail</div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
            height=32></article>
    </a>
    <a href=https://budget.public.lu/lb/budget2015/zukunftspak.html target=_blank>
        <article>
            <div>Zukunftspak</div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
            height=32></article>
    </a>
    <a href=pdf/Recueil-de-legislation-stagiaires-2017.pdf target=_blank>
        <article>
            <div>Recueil de législation stagiaires 2017</div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg
            width=32
            height=32></article>
    </a>
    <a href=http://www.men.public.lu/fr/legislation/education-nationale/enseignement-
secondaire/index.html
        target=_blank>
        <article>
            <div>Legislatiounskollektioun fir de Secondaire vum MENJE</div>
            <img
                src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
                height=32></article>
    </a>
    <a href=http://legilux.public.lu/eli/etat/leg/loi/1980/06/10/n1/jo
        target=_blank class=law>
        <article>
            <div>Gesetz vum 10.6.7 iwwert d'Tâche vum Enseignant</div>

```

```

    <img
      src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
      height=32></article>
  </a>
  <a href=http://legilux.public.lu/eli/etat/leg/rgd/2007/07/24/n1/jo
    target=_blank class=RGD>
    <article>
      <div>RGD vum 24.7.7 iwwert d'Tâche vum Enseignant</div>
      <img
        src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
        height=32></article>
    </a>
    <a href=http://legilux.public.lu/eli/etat/leg/loi/2017/08/29/a789/jo
      target=_blank class=law>
      <article>
        <div>Gesetz vum 29.8.17 iwwert de Secondaire</div>
        <img
          src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
          height=32></article>
      </a>
      <a href=http://legilux.public.lu/eli/etat/leg/loi/2008/12/19/n19/jo
        target=_blank class=law>
        <article>
          <div>Gesetz vum 19.12.08 iwwert d'Reform vun der Berufsausbildung</div>
          <img
            src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
            height=32></article>
        </a>
        <a href=http://legilux.public.lu/eli/etat/leg/rgd/2017/08/21/a803/jo
          target=_blank class=RGD>
          <article>
            <div>RGD vum 21.8.17 iwwert d'Evaluatioun an d'Promotioun am Secondaire</div>
            <img
              src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
              height=32></article>
          </a>
          <a href=http://legilux.public.lu/eli/etat/leg/rgd/2018/07/20/a735/jo
            target=_blank class=RGD>
            <article>
              <div>RGD vum 20.7.18 iwwert d'Evaluatioun an d'Promotioun am Secondaire</div>
              <img
                src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
                height=32></article>
            </a>
            <a href='pdf/IM 2018_03_23.pdf' target=_blank class=IM>
            <article>
              <div>Instruction ministérielle du 23 mars 2018 portant sur les mesures
                transitoires pour la décision de promotion à la fin de l'année scolaire
                2017/2018 dans le cadre de la réforme des classes inférieures et supérieures
                de
                l'enseignement secondaire général

```

```

    </div>
    <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
        height=32></article>
</a>
<a href=http://legilux.public.lu/eli/etat/leg/rgd/2016/08/31/n1/jo
    target=_blank class=RGD>
    <article>
        <div>RGD vum 31.8.16 iwwert d'Evaluatioun an d'Promotioun an der
Beruffsausbildung</div>
        <img src=biller/if_law_298810.svg
            alt=if_law_298810.svg width=32
            height=32></article>
</a>
<a href=http://legilux.public.lu/eli/etat/leg/rgd/2016/09/06/n1/jo
    target=_blank class=RGD>
    <article>
        <div>RGD vum 6.9.16 betreffend d'Chargés d'éducation, Examen an d'Tâche vun den
            Enseignanten
        </div>
        <img src=biller/if_law_298810.svg
            alt=if_law_298810.svg width=32
            height=32></article>
</a>
<a
    href='pdf/2018-06-22 IM Formation continue.pdf'
    target=_blank class=IM>
    <article>
        <div>Instruction ministérielle du 22 juin 2018 concernant la formation continue
            des enseignants fonctionnaires et employés de l'enseignement secondaire
classique et
            de l'enseignement secondaire général
        </div>
        <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
            height=32></article>
</a>
<a
    href='pdf/10 07 14 Instruction organisation scolaire.pdf' target=_blank class=IM>
    <article>
        <div>Instruction ministérielle du 17 juillet 2014 concernant l'organisation
scolaire
            des lycées et lycées techniques
        </div>
        <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
            height=32></article>
</a>
<a
    href=http://www.men.public.lu/fr/secondaire/personnel-ecoles/recrutement-
enseignant-fonctionnaire
    target=_blank>
    <article>
        <div>Recrutement des enseignants-fonctionnaires</div>

```

```

        <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
            height=32></article>
    </a>
    <a
        href=https://ssl.education.lu/ifen/stage-f-es target=_blank>
        <article>
            <div>Stage des fonctionnaires de l'enseignement secondaire, de la formation
d'adultes et
                du Centre de logopédie
            </div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
                height=32></article>
        </a>
    <a
        href=biller/PRP.png target=_blank>
        <article>
            <div>Postes à responsabilités particulières</div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
                height=32></article>
        </a>
    <a
        href=http://legilux.public.lu/eli/etat/leg/loi/2016/07/23/n22/jo target=_blank
class=law>
        <article>
            <div>Gesetz vum 23.7.16 iwwert d'Chargés d'éducation (cf.
                <object><a
                    href=http://www.men.public.lu/fr/actualites/articles/communiqués-conference-
presse/2016/06/30-charges/index.html
                    target=_blank>Artikel MENJE</a>)
                </object>
            </div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
                height=32></article>
        </a>
    <a
        href="pdf/Applicabilité loi 15.12.2017 aux agents de l'Etat - Version finale.pdf"
        target=_blank>
        <article>
            <div>Applicabilité de la loi du 15 décembre 2017 aux agents de l'Etat
            </div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
                height=32></article>
        </a>
    <a
        href=http://www.men.public.lu/fr/actualites/articles/communiqués-conference-
presse/2016/03/17-CAR
        target=_blank>
        <article>
            <div>Aménagements raisonnables</div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32 height=32>
        </article>

```

```

</a>
<a
  href=http://www.men.public.lu/fr/actualites/articles/communiqués-conference-
presse/2015/12/10-medicaments-ecole
  target=_blank>
  <article>
    <div>Projet d'accueil individualisé (PAI)
      <object>(cf. <a
        href=http://www.guichet.public.lu/citoyens/fr/famille/parents/assistance-
enfance/projet-accueil-individualise/pai-lettre-medecine-scolaire.pdf
        target=_blank>Aktualiséierung vun der Instruction de service PAI vum
15.10.15</a>)
      </object>
    </div>
    <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32 height=32>
  </article>
</a>
<a
  href=http://legilux.public.lu/eli/etat/leg/rgd/2018/05/09/a425/jo
  target=_blank class=RGD>
  <article>
    <div>RGD vum 9.5.18 iwwert d'Disziplin an de Lycéeën</div>
    <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
  </a>
  <a
    href=http://legilux.public.lu/eli/etat/leg/loi/2018/05/09/a373/jo target=_blank
class=law>
    <article>
      <div>Gesetz vum 9.5.18 dat de Beamtestatut verännert</div>
      <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
    </a>
    <a
      href=https://fonction-publique.public.lu/fr/plus/actualites/articles-
actualites/2018/09/miseplacecetnouvelleorganisationtempstravail.html
      target=_blank class=law>
      <article>
        <div>Mise en place d'un compte épargne-temps et nouvelle organisation du temps
de
          travail
          <object>(
            <a
              href=http://chd.lu/wps/portal/public/Accueil/TravailALaChambre/Recherche/RoleDesAffair
es?action=doDocpaDetails&backto=/wps/portal/public/Accueil/Actualite&id=7171
              target=_blank>Projet de loi</a>
            </div>
            <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
          </a>
          <a
            href=http://legilux.public.lu/eli/etat/leg/rgd/2018/07/20/a677/jo

```

```

target=_blank class=RGD>
<article>
  <div>RGD vum 20.7.18 iwwert de Fonctionnement an d'Missiounen vum Collège des
    directeurs
  </div>
  <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
</a>
<a
href=http://legilux.public.lu/eli/etat/leg/rgd/2018/08/31/a815/jo
target=_blank class=RGD>
<article>
  <div>RGD vum 31.8.18 iwwert d'Grilles horaires am Secondaire classique</div>
  <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
</a>
<a
href=http://legilux.public.lu/eli/etat/leg/rgd/2018/08/31/a816/jo
target=_blank class=RGD>
<article>
  <div>RGD vum 31.8.18 iwwert d'Grilles horaires am Secondaire général</div>
  <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
</a>
<a
href=http://legilux.public.lu/eli/etat/leg/rgd/2018/08/31/a817/jo
target=_blank class=RGD>
<article>
  <div>RGD vum 31.8.18 iwwert d'Grilles horaires an der Berufsausbildung</div>
  <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
</a>
<a
href=http://data.legilux.public.lu/eli/etat/leg/memorial/2013/108/fr/pdf
target=_blank class=RGD>
<article>
  <div>RGD vum 21.6.13 iwwert d'Reduktioun vun den Indemnitéiten vun de
Kommissiounen
  </div>
  <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
</a>
<a
href=http://legilux.public.lu/eli/etat/leg/rgd/2018/10/19/a985/jo
target=_blank class=RGD>
<article>
  <div>RGD vum 19.10.18 iwwert d'Indemnitéiten vun de Kommissiounen
  </div>
  <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
</a>

```

```

<a
  href=http://legilux.public.lu/eli/etat/leg/loi/2019/08/01/a563/jo
  target=_blank class=law>
<article>
  <div>Gesetz vum 1.8.19 iwwert de Stage an d'Formation continue
    <object>(cf. <a
href=https://chd.lu/wps/portal/public/Accueil/TravailALaChambre/Recherche/RoleDesAffai
res?action=doDocpaDetails&id=7440
      target=_blank>kompletten Dossier vum Projet de loi 7440</a>)
    </object>
  </div>
  <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32 height=32>
</article>
</a>
<a
  href=http://legilux.public.lu/eli/etat/leg/rgd/2019/08/15/a571/jo
  target=_blank class=RGD>
<article>
  <div>RGD vum 15.8.19 iwwert d'Evaluatioun an d'Promotioun an der
    Beruffsausbildung</div>
  <img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
height=32></article>
</a>
<a href='http://www.ja.etat.lu/35001-40000/37203.pdf' target=_blank class=judgement>
<article>
  <div>Geriichtsuerteel vum 12.10.16 dat den RGD vum 25.8.15 iwwer
d'Ofschlossexamen,
    d'Tâche vum Enseignant an d'Indemnitéiten annuléiert</div>
  <img
    src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
    height=32></article>
</a>
<a href='http://www.ja.etat.lu/35001-40000/37490.pdf' target=_blank class=judgement>
<article>
  <div>Geriichtsuerteel vum 24.1.17 dat den RGD vum 19.10.15 iwwer d'Tâche vum
    Enseignant annuléiert</div>
  <img
    src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
    height=32></article>
</a>
<a
  href=https://justice.public.lu/fr/actualites/2019/02/arret-cour-administrative-
tache-des-professeurs-zukunftspak.html
  target=_blank class=judgement>
<article>
  <div>Pressecommuniqué vum Geriicht zum Uerteel vum 12.2.19 zur Formation
continue an
    dem Coefficient correcteur
  <object>(cf. <a
    href=http://www.ja.etat.lu/40001-45000/40638CA.pdf

```

```

        target=_blank>Geriichtsuerteel vum 12.2.19</a>, <a
        href=http://www.ja.etat.lu/40001-45000/40638C.pdf
        target=_blank>Geriichtsuerteel vum 19.6.18</a>, <a
        href=http://www.ja.etat.lu/35001-40000/38823.pdf
        target=_blank>Geriichtsuerteel vum 8.12.17</a> an <a
        href=http://legilux.public.lu//eli/etat/leg/acc/2018/12/07/a1127/jo
        target=_blank>Unuerdnung vum Verfassungsgeriicht vum 7.12.18</a>)
    </object>
</div>
<img src=biller/if_law_298810.svg alt=if_law_298810.svg width=32 height=32>
</article>
</a>
<a href='http://www.ja.etat.lu/35001-40000/39953.pdf' target=_blank class=judgement>
<article>
    <div>Geriichtsuerteel vum 5.4.19 iwwert de Refus vum MENJE fir e Recalcul vum
    Traitement an den Indemnitéiten vir ze huelen op Grond vun der Annulation vum
    RGD vum 25.8.15
    </div>
    <img
    src=biller/if_law_298810.svg alt=if_law_298810.svg width=32
    height=32></article>
</a>
</section>
<!--<h1>Geriichtsuerteeler</h1>-->
<section>
    <article>
        <p>Benotzte Faarwen: <span style='color: white; background-color:
red'>Gesetzer</span>,
        <span style='color: white; background-color: orange'>Reglementer</span>,
        <span style='color: white; background-color: green'>ministeriell
Instruktiounen</span>,
        <span style='color: white; background-color: blue'>Geriichtsuerteeler</span>,
        <span style='color: white; background-color: black'>Aneres</span>.</p>
        Déi éischt Plaz fir e spezifeschen Gesetzestext oder Reglement ze fannen ass de
        <a href=http://legilux.public.lu/search/A target=_blank>Legilux</a>. Hei e puer
Hiweiser
        wéi ee schnell eppes fanne kann wann een ongeféier weess wat ee sicht:
        <ul>
            <li>Wann een den Titel vun engem Gesetz oder RGD kennt kann een deen an der
            Sichkëscht aginn.
            </li>
            <li>Wann een no engem RGD sicht an och weess a wéi engem Joer an eventuell Mount
en
            ënnerschriwwen ginn ass, da kann een deen heite Link benotzen:
            <a href=http://legilux.public.lu/eli/etat/leg/rgd/2007/07 target=_blank>
            http://legilux.public.lu/eli/etat/leg/rgd/2007/07</a> woubäi
            <pre>2007</pre>
            duerch d'Joer an
            <pre>07</pre>
            duerch de Mount z'ersetzen sinn. Wann ee just
            d'Joer kennt da léisst een den

```

```

    <pre>/07</pre>
    Deel ewech. Wann een souguer den Dag kennt da setzt een deen
    dobäi, z.B. <a href=http://legilux.public.lu/eli/etat/leg/rgd/2007/07/24
        target=_blank>
        http://legilux.public.lu/eli/etat/leg/rgd/2007/07/24</a>.
    </li>
    <li>Wann een no engem Gesetz sicht ass d'Approche genau déi selwecht, et ersetzt
een
        just
        <pre>rgd</pre>
        duerch
        <pre>loi</pre>
        , z.B.
        <a href=http://legilux.public.lu/eli/etat/leg/loi/2017/08/29 target=_blank>
        http://legilux.public.lu/eli/etat/leg/loi/2017/08/29</a>.
    </li>
</ul>
    Geriichtsuerteeler fënnt een um Site vun der <a
    href=http://www.justice.public.lu/fr/jurisprudence/juridictions-
administratives/index.php
    target=_blank>Justiz</a>.
    Gesetzesprojeten, Motiounen etc. fënnt een um Site vun der
    <a
href=http://chd.lu/wps/portal/public/Accueil/TravailALaChambre/Recherche/RoleDesAffair
es
        target=_blank>Chamber</a>. Bei der Sich no engem Affekot kann de
    <a href=https://www.barreau.lu/advancedsearch target=_blank>Barreau</a>
    ganz hëllefräich sinn.
</article>
</section>

```

## legislatioun.css

```

main {
    overflow: auto;
}

/* https://developer.mozilla.org/en-US/docs/Web/CSS/@supports */
@supports not (-ms-high-contrast: none) {
    /* not IE10+ CSS */
    main > section > a {
        text-decoration: none;
        background: linear-gradient(to bottom right, lightgray, gray);
        box-shadow: 10px 10px 10px black;
        padding: 5px;
        margin: 0 10px 20px;
    }

    main > section > a:hover {
        background: linear-gradient(to bottom right, darkslategray, lightgray);
    }
}

```

```

}

main > section > a.law {
  box-shadow:      10px 10px 10px red;
}

main > section > a.RGD {
  box-shadow:      10px 10px 10px orange;
}

main > section > a.IM {
  box-shadow:      10px 10px 10px green;
}

main > section > a.judgement {
  box-shadow:      10px 10px 10px blue;
}

main > section > a > article {
  display:    flex;
  min-height: 50px;
}
}

/* https://stackoverflow.com/questions/43528940/how-to-detect-ie-and-edge-browsers-in-
css */
@media screen and (-ms-high-contrast: none), (-ms-high-contrast: active) {
  /* IE10+ CSS */
  main > section > a {
    text-decoration: none;
  }

  main > section > a > article {
    background: linear-gradient(to bottom right, lightgray, gray);
    box-shadow: 10px 10px 10px black;
    padding:    5px;
    margin:     0 10px 20px;
    display:    flex;
    min-height: 50px;
  }

  main > section > a > article:hover {
    background: linear-gradient(to bottom right, darkslategray, lightgray);
  }
}

main > section > a > article > div {
  flex: auto;
}

main > section > a > article > img {

```

```

min-width:      32px;
max-width:      32px;
justify-content: flex-end;
margin-left:    10px;
}

main > section:first-of-type {
  display:      grid;
  grid-template-columns: 1fr 1fr 1fr 1fr 1fr;
  grid-template-rows:   1fr 1fr;
  align-items:      stretch;
}

main > section pre {
  display: inline;
}

/*main > section:nth-child(2) > article a {
  text-decoration: none;
}*/

main > section:nth-of-type(2) > article {
  background-color: rgb(240, 240, 240);
  padding:         5px;
  margin:          0 10px 20px;
  box-shadow:      10px 10px 10px black;
}

@media all and (max-width: 679px) {
  main > section:first-child {
    grid-template-columns: 1fr;
  }
}

@media all and (min-width: 680px) {
  main > section:first-child {
    grid-template-columns: 1fr 1fr;
  }
}

@media all and (min-width: 1006px) {
  main > section:first-child {
    grid-template-columns: 1fr 1fr 1fr;
  }
}

@media all and (min-width: 1338px) {
  main > section:first-child {
    grid-template-columns: 1fr 1fr 1fr 1fr;
  }
}

```

```
@media all and (min-width: 1662px) {
  main > section:first-child {
    grid-template-columns: 1fr 1fr 1fr 1fr 1fr;
  }
}
```

## login.html

```
<input type=email name=email placeholder=Emailadress required autofocus>
<input pattern={6,50} type=password name=password placeholder=Passwuert required>
<button name=login>Aloggen</button>
<button name=cancel>Annuléieren</button>
```

## login.css

```
body > input {
  text-align: center;
}

body > input, body > button {
  font-size: 2em;
}

input:not([type=file]):valid {
  background-color: lawngreen;
}

input:not([type=file]):invalid {
  background-color: red;
}
```

## member.html

```
<p>Dat éischt Schouljoer ass d'Memberschaft fir nei Memberen gratis. Duerno kascht et
60
€/Joer.
Doranner abegraff
sinn och d'Servicer wéi Rechtsschutz an Haftpflicht. Wann Dir wëllt Member
ginn da schéckt w.e.g. en Email un
<a
```

```
href="#109;&#x61;&#x69;&#108;&#116;&#111;&#58;&#x61;&#x70;&#101;&#115;&#115;&#x40;&#1
01;&#x64;&#117;&#99;&#x61;&#x74;&#x69;&#x6f;&#110;&#x2e;&#x6c;&#117;&#x20subject=APES%20M
ember%20ginn">&#97;&#112;&#x65;&#x73;&#115;&#x40;&#x65;&#x64;&#117;&#x63;&#x61;&#x74;&
#x69;&#x6f;&#110;&#x2e;&#108;&#117;</a>.</p>
```

```
p {
}
```

```
<h1>Association des Professeurs de I'Enseignement Secondaire et Supérieur du
  Grand-Duché de Luxembourg, association sans but lucratif</h1>
<h2>Adress</h2>
389, Route d'Arlon<br>
L-8011 Strassen

<h2>Exekutivcomité</h2>
<!--
<img srcset="fotoen/Exekutivcomite533x400.jpg 533w,
             fotoen/Exekutivcomite1000x750.jpg 1000w,
             fotoen/Exekutivcomite1440x1080.jpg 1440w,
             fotoen/Exekutivcomite2000x1500.jpg 2000w,
             fotoen/Exekutivcomite3000x2250.jpg 3000w,
             fotoen/Exekutivcomite3902x2927.jpg 3902w"
      sizes="(max-width: 1000px) 533px,
             (max-width: 1500px) 1000px,
             (max-width: 2000px) 1440px,
             (max-width: 2500px) 2000px,
             (max-width: 3500px) 3000px,
             3902px"
      src=fotoen/Exekutivcomite3902x2927.jpg alt=fotoen/Exekutivcomité3902x2927.jpg>
<br>Vu lénks no riets:-->
<table>
  <tr>
    <th>President</th>
    <td>Gilles Everling</td>
    <td>Professeur-ingénieur</td>
    <td>Lycée des Arts et Métiers</td>
    <td>Informatik</td>
  </tr>
  <tr>
    <th>Vizepresident</th>
    <td>Patrick Beil</td>
    <td>Professeur-ingénieur</td>
    <td>Lycée technique du Centre</td>
    <td>Mechanik</td>
  </tr>
  <tr>
    <th>Sekretär</th>
    <td>Jean-Marc Cima</td>
    <td>Professeur-ingénieur</td>
    <td>Lycée Josy Barthel Mamer</td>
    <td>Génie civil</td>
```

```

</tr>
<tr>
  <th>Tresorier</th>
  <td>Pascal Zeihen</td>
  <td>Professeur</td>
  <td>Lycée classique de Diekirch</td>
  <td>Mathematik</td>
</tr>

```

```

<!-- <tr>
  <th>President</th>
  <td>André Berns</td>
  <td>Professeur</td>
  <td>Lycée de Garçons de
    Luxembourg
  </td>
  <td>Mathematik</td>
</tr>-->

```

```

</table>

```

## Kontakt</h2>

```

<a

```

```

href="#109;&#x61;&#x69;&#108;&#116;&#111;&#58;&#x61;&#x70;&#101;&#115;&#115;&#x40;&#1
01;&#x64;&#117;&#99;&#x61;&#x74;&#x69;&#x6f;&#110;&#x2e;&#x6c;&#117;">&#97;&#112;&#x65
;&#x73;&#115;&#x40;&#x65;&#x64;&#117;&#x63;&#x61;&#x74;&#x69;&#x6f;&#110;&#x2e;&#108;&#
117;</a>

```

## Member ginn</h2>

D'Memberschaft kascht 60 €/Schouljoer. Doranner abegraff sinn och déi ënnen  
beschriwwen

Servicer. Wann Dir wëllt Member ginn da schéckt w.e.g. en Email un

```

<a

```

```

href="#109;&#x61;&#x69;&#108;&#116;&#111;&#58;&#x61;&#x70;&#101;&#115;&#115;&#x40;&#1
01;&#x64;&#117;&#99;&#x61;&#x74;&#x69;&#x6f;&#110;&#x2e;&#x6c;&#117;?subject=APESS%20M
ember%20ginn">&#97;&#112;&#x65;&#x73;&#115;&#x40;&#x65;&#x64;&#117;&#x63;&#x61;&#x74;&#
x69;&#x6f;&#110;&#x2e;&#108;&#117;</a>.

```

## Kontosnummer</h2>

CCPLLULL LU53 1111 0220 7859 0000

# Servicer</h1>

## Rechtsschutz- an Haftpflichtversécherung bei der Lalux</h2>

All APESS-Member huet eng Rechtsschutz- an Haftpflichtversécherung bei der Lalux.

Déi verséichert Zommen sinn:

```

<ul>

```

```

  <li>Rechtsschutz: 4.800,00 €</li>

```

```

  <li>Haftpflicht: Dommages corporels et immatériels consécutifs: 11.143.195,52 €</li>

```

```

  <li>Haftpflicht: Dommages matériels et immatériels consécutifs: 718.200,00 €</li>

```

```

</ul>

```



```

rue du Village'>
  <label>Stad:</label><input pattern=. {1,50} name=city required
                                title='1 bis 50 Zeechen' placeholder='1 bis 50 Zeechen'>
  <label>Land:</label><input pattern=[A-Z]- name=country title='Z.B. L-, D-, F- oder
B-'
                                required placeholder='Z.B. L-, D-, F- oder B-'>
  <label>Postleetzuel:</label><input pattern=[0-9]{4,15} name=postalCode required
                                title='4 bis 15 Zifferen' placeholder='4 bis 15
Zifferen'>
  <label>Schoul:</label><input placeholder='(RET -> pension., DIV -> divers/décharg.)'
                                pattern=. {1,50} name=school
                                title='(RET -> pension., DIV -> divers/décharg.)'>
  <label>Fach:</label><input pattern=. {1,50} name=subject
                                title='1 bis 50 Zeechen, z.B. FRA oder MAT'
                                placeholder='1 bis 50 Zeechen, z.B. FRA oder MAT'>
  <label>Funktoun:</label><input pattern=. {1,50} name=function
                                title='1 bis 50 Zeechen, z.B. Professeur'
                                placeholder='1 bis 50 Zeechen, z.B. Professeur'>
  <label>Member säit:</label><input readonly name=adhesion title='Member säit'>
  <label>Lescht Cotisatioun:</label><input readonly name=lastPaymentDate>
  <label>BIC:</label><input pattern=[A-Z]{8} name=BIC title='8 grouss Buschtawen'
                                placeholder='8 grouss Buschtawen'>
  <label>IBAN:</label><input pattern='[A-Za-z]{2}[0-9 ]{18,22}' name=IBAN
                                title='2 Buschtawen 18 Zifferen an evtl. Espacen'
                                placeholder='2 Buschtawen 18 Zifferen an evtl. Espacen'>
  <label>Cotisatiounsmodus:</label>
  <div id=AP_V><label>Domiciliatioun</label><input type=radio name=AP_V
value='Domiciliatioun'>
    <label>Virement</label><input type=radio name=AP_V value='Virement'></div>
  <!--<input
                                placeholder='V -> Virement, APS -> Ordre permanent' pattern=. {1,3} name=AP_V
                                title='1-3 characters'>-->
  <label>Emaile kréien:</label><input type=checkbox name=getEmail>
  <label id=pw>Falls Dir wëllt Äert Passwuert änneren da gitt dat neit Passwuert hei
zweemol
    an, anerefalls loosst déi Felder eidel. Äert Passwuert gëtt verschlësselt
    gespäichert a ka vu kengem gesi ginn, och net vum Administrateur vum Site.</label>
  <input pattern=. {6,50} type=password name=pw1 placeholder='Neit Passwuert'
                                title='6-50 Zeechen'>
  <input pattern=. {6,50} type=password name=pw2 placeholder='Neit Passwuert
widerhuelen'
                                title='6-50 Zeechen'>
  <button name=save>All Ännerunge späicheren</button>
  <!-- <a href='members/pdf/Rapport Entrevue APESS-MENJE 16_05_18.pdf' target=_blank>
    Rapport Entrevue APESS-MENJE 16_05_18
  </a>-->
</section>

```



```
main {
  overflow: auto;
}

main > section {
  max-width: 400px;
}

main > section > label {
  text-align: right;
  display: inline-block;
  width: 150px;
  padding-right: 10px;
}

main > section > input {
  width: 250px;
}

main > section > input:not([type=file]):valid {
  background-color: lawngreen;
}

main > section > input:not([type=file]):invalid {
  background-color: red;
}

main > section > input[type=checkbox], section > input[type=radio] {
  width: auto;
}

main > section > #AP_V {
  display: inline-block;
  width: 245px;
}

main > section > input[type=password] {
  width: 197px;
}

main > section > label#pw {
  text-align: justify;
  display: block;
  width: auto;
  padding: 0;
}

main > section > button[name=save] {
  width: 100%;
}
```

```

<table>
  <tr>
    <td><a href=http://www.men.public.lu/fr/actualites/publications/index.html
      target=_blank>MENJE</a></td>
    <td><a
      href=http://www.men.public.lu/fr/actualites/publications/themes-
transversaux/edi/index.html
      target=_blank>Infomagazin fir Elteren</a></td>
  </tr>
  <tr>
    <td colspan=2><a href=http://www.etat.public.lu target=_blank>Ëffentlech
Siten</a></td>
  </tr>
  <tr>
    <td
      colspan=2><a href=http://www.fonction-publique.public.lu target=_blank>Portal
vun
      der Fonction publique</a></td>
  </tr>
  <tr>
    <td colspan=2><a href=https://chfep.lu target=_blank>Chambre des fonctionnaires et
employés
publics</a></td>
  </tr>
  <tr>
    <td
      colspan=2><a href=http://www.gouvernement.lu target=_blank>Portal vun der
Regierung</a></td>
  </tr>
  <tr>
    <td
      colspan=2><a href=http://www.cnel.lu target=_blank>CNEL - Conférence Nationale
des
      Élèves du Luxembourg</a></td>
  </tr>
  <tr>
    <td
      colspan=2><a href=https://acel.lu target=_blank>ACEL - Association des Cercles
d'Étudiants Luxembourgeois</a></td>
  </tr>
  <tr>
    <td
      colspan=2><a href=https://fapel.lu target=_blank>FAPEL - Fédération des
Associations de Parents d'Élèves du Luxembourg</a></td>
  </tr>
  <tr>
    <td colspan=2><a href=http://leopold-loewenheim.uni.lu/apul target=_blank>APUL -
The
      Association of Professors of the University of Luxembourg</a></td>

```

```

</tr>
<tr>
  <td colspan=2><a href=https://ei-ie.org target=_blank>Education
International</a></td>
</tr>
<tr>
  <td colspan=2><a href=https://dysfocus.lu target=_blank>DysFocus</a></td>
</tr>
<tr>
  <td colspan=2>
    <a
      href=http://www.men.public.lu/fr/actualites/articles/communiques-conference-
presse/2018/02/08-observatoire/index.html
      target=_blank>Observatoire national de la qualité scolaire et Conseil national
des
      programmes</a></td>
</tr>
<tr>
  <td colspan=2>
    <a
      href=http://www.lifelong-learning.lu
      target=_blank>D'Portal fir liewenslaangt Léieren</a></td>
</tr>
<tr>
  <td colspan=2>
    <h2>Nationale Bildungsbericht 2018</h2>
    <ul>
      <li><a
        href=http://www.men.public.lu/fr/actualites/publications/themes-
transversaux/statistiques-analyses/bildungsbericht/2018
        target=_blank>Bericht</a></li>
      <li><a href=https://www.bildungsbericht.lu target=_blank>Erklärungen an
Auteuren</a>
      </li>
    </ul>
  </td>
</tr>
<tr>
  <td colspan=2>
    <h2>Elektromagnéitesch Felder (WiFi etc.)</h2>
    <ul>
      <li><a
        href=https://ehtrust.org/science/research-on-wireless-health-effects
        target=_blank>Peer Reviewed Scientific Research on Wireless
Radiation</a></li>
      <li><a href=http://emf.foxi.lu target=_blank>Fuerschung an Impakt op eis
Gesondheet</a>
      </li>
      <li><a
        href=http://www.men.public.lu/fr/actualites/articles/questions-
parlementaires/2017/10/27-qp-3313/hansen.pdf

```

```

        target=_blank>QP 3313 21.9.17 : Usage du Wi-Fi à l'école</a></li>
    </ul>
</td>
</tr>
<tr>
    <td colspan=2>
        <a
            href=http://www.men.public.lu/fr/actualites/articles/questions-
parlementaires/2017/10/09-qp-3237/hansen.pdf
            target=_blank>QP 3237 23.8.17 : Réforme de la formation
professionnelle</a></td>
    </tr>
<tr>
    <td colspan=2>
        <a href=https://www.100komma7.lu/article/aktualiteit/d-zukunft-vun-der-aarbecht
            target=_blank>12.11.17 100komma7 Zukunft vun der Aarbecht</a></td>
    </tr>
<tr>
    <td colspan=2>
        <a href=http://tele.rtl.lu/emissiounen/kloertext/1097227.html
            target=_blank>19.11.17 RTL Kloertext Aarbecht: Wéi, wéini a wou (muer)
            schaffen?</a></td>
    </tr>
<tr>
    <td colspan=2>
        <a href=http://www.webcorp.org.uk
            target=_blank>WebCorp fir rar an/oder nei Aspekter vun der englescher Sprooch
no ze
            sichen.</a></td>
    </tr>
<tr>
    <td colspan=2>
        <h2>D'Lëtzebuerger Sprooch</h2>
        <ul>
            <li><a href=https://spellchecker.lu/online-checker target=_blank>Online-
Checker</a>
            </li>
            <li><a href=http://www.lod.lu target=_blank>Lëtzebuerger Online
Dictionnaire</a></li>
            <li><a href=http://dict.luxdico.com target=_blank>Luxdico.com</a></li>
            <li><a href=https://www.lexilogos.com/english/luxembourgish_dictionary.htm
                target=_blank>Lexilogos</a></li>
            <li><a href=http://engelmann.uni.lu:8080/portal/WBB2009/LWB//wbgui_py
                target=_blank>Luxemburger Wörterbuch</a></li>
            <li><a href=https://portal.education.lu/schreiwen
target=_blank>Schreiwen</a></li>
            <li><a href=http://www.quattropole.org/en/e_learning target=_blank>Learn
            Luxembourgish!</a></li>
            <li><a href=http://www.cp11.lu target=_blank>Conseil fir d'Lëtzebuerger
Sprooch</a>
            </li>

```

```

        <li><a
            href=http://www.men.public.lu/fr/actualites/grands-dossiers/systeme-
            educatif/letzebuerger-sprooch
            target=_blank>Promotioun vun der Lëtzebuerger Sprooch</a></li>
        <li><a href=http://tele.rtl.lu/emissionen/kloertext/1149479.html
            target=_blank>14.3.18 RTL Kloertext Wéi vill Lëtzebuergesch brauch onst
Land?</a>
        </li>
    </ul>
</td>
</tr>
<!--<tr>
    <td><a href=http://legilux.public.lu/editorial/codes target=_blank>Code de la
fonction
        publique</td>
</tr>
<tr>
    <td><a href=http://legilux.public.lu/eli/etat/leg/loi/2017/08/29/a789/jo
        target=_blank>Loi du 29 août 2017 portant sur l'enseignement
secondaire</a></td>
</tr>
<tr>
    <td><a href=http://legilux.public.lu/eli/etat/leg/rgd/2016/08/31/n1/jo
target=_blank>
        Règlement grand-ducal du 31 août 2016 portant sur l'évaluation et la promotion
des élèves
        de la formation professionnelle</a>
    </td>
</tr>-->
</table>

```

## ressourcen.css

```

table {
    border-collapse: collapse;
    box-shadow: 2px 2px 2px #888888;
}

table, tr, th, td {
    border: 4px groove black;
}

th, td {
    padding: 5px;
}

```

# Chapter 6. Game development

[Mozilla](#) host a great resource for all things related to game development.

Here is a list of some games running in the browser:

1. [github.com/kripken/BananaBread](https://github.com/kripken/BananaBread)
2. [hexgl.bkcore.com](https://hexgl.bkcore.com)
3. [files.unity3d.com/jonas/AngryBots](https://files.unity3d.com/jonas/AngryBots)
4. [work.goodboydigital.com/runpixierun](https://work.goodboydigital.com/runpixierun)
5. [flashvhtml.com/html](https://flashvhtml.com/html)
6. [robostorm.io](https://robostorm.io)
7. [playcanv.as/p/44MRmJRU](https://playcanv.as/p/44MRmJRU)

Game engines	<a href="https://html5gameengine.com">html5gameengine.com</a>
HTML5 Game Development	<a href="https://www.html5gamedevelopment.com">www.html5gamedevelopment.com</a>

## 6.1. JS game engines

<a href="https://phaser.io">phaser.io</a>
<a href="https://www.babylonjs.com">www.babylonjs.com</a>
<a href="https://playcanvas.com">playcanvas.com</a>
<a href="https://www.melonjs.org">www.melonjs.org</a>
<a href="https://screeps.com">screeps.com</a>
<a href="https://cocos2d-x.org">cocos2d-x.org</a>

### 6.1.1. Phaser

<a href="https://yorkcs.com/2019/02/06/build-a-space-shooter-with-phaser-3">yorkcs.com/2019/02/06/build-a-space-shooter-with-phaser-3</a>
<a href="https://snowbillr.github.io/blog/2018-07-03-buttons-in-phaser-3">snowbillr.github.io/blog/2018-07-03-buttons-in-phaser-3</a>
<a href="https://www.codeandweb.com/free-sprite-sheet-packer">www.codeandweb.com/free-sprite-sheet-packer</a>

## 6.2. Game assets

<a href="https://opengameart.org">opengameart.org</a>
<a href="https://itch.io">itch.io</a>
<a href="https://bayat.itch.io/platform-game-assets">bayat.itch.io/platform-game-assets</a>
<a href="https://www.vgmusic.com">www.vgmusic.com</a>

## 6.3. Game promotion monetization

[developer.mozilla.org/en-US/docs/Games/Publishing\\_games/Game\\_promotion](https://developer.mozilla.org/en-US/docs/Games/Publishing_games/Game_promotion)

[developer.mozilla.org/en-US/docs/Games/Publishing\\_games/Game\\_monetization](https://developer.mozilla.org/en-US/docs/Games/Publishing_games/Game_monetization)

[gameanalytics.com/blog/42-ways-to-monetize-your-mobile-game.html](https://gameanalytics.com/blog/42-ways-to-monetize-your-mobile-game.html)

[gamedev.stackexchange.com/questions/4968/what-revenue-models-exist-for-online-games](https://gamedev.stackexchange.com/questions/4968/what-revenue-models-exist-for-online-games)

[en.wikipedia.org/wiki/Video\\_game\\_monetization](https://en.wikipedia.org/wiki/Video_game_monetization)

[websitesetup.org/33-ways-to-monetize-website](https://websitesetup.org/33-ways-to-monetize-website)

# Chapter 7. Machine learning

According to [en.wikipedia.org/wiki/Artificial\\_intelligence](https://en.wikipedia.org/wiki/Artificial_intelligence):

Artificial intelligence (AI, also machine intelligence, MI) is intelligence demonstrated by machines, in contrast to the natural intelligence (NI) displayed by humans and other animals. In computer science AI research is defined as the study of "intelligent agents": any device that perceives its environment and takes actions that maximize its chance of successfully achieving its goals. Colloquially, the term "artificial intelligence" is applied when a machine mimics "cognitive" functions that humans associate with other human minds, such as "learning" and "problem solving".

According to [en.wikipedia.org/wiki/Machine\\_learning](https://en.wikipedia.org/wiki/Machine_learning):

Machine learning is a field of computer science that often uses statistical techniques to give computers the ability to "learn" (i.e., progressively improve performance on a specific task) with data, without being explicitly programmed.

[www.stateof.ai](http://www.stateof.ai)

[developers.google.com/machine-learning](https://developers.google.com/machine-learning)

[github.com/jctillman/js-ml-workshop](https://github.com/jctillman/js-ml-workshop)

[burakkanber.com/tag/ml-in-js](https://burakkanber.com/tag/ml-in-js)

[tutorialzine.com/2017/04/10-machine-learning-examples-in-javascript](https://tutorialzine.com/2017/04/10-machine-learning-examples-in-javascript)

[bri.im/learn](https://bri.im/learn)

[www.robinwieruch.de/machine-learning-javascript-web-developers](https://www.robinwieruch.de/machine-learning-javascript-web-developers)

To solve the mathematical imprecision problem in JS (try  $0.1 + 0.2$  in the console and see what you get) we can use [decimal.js](https://github.com/mikolajdobrzanski/decimal.js). For details on the problem in different programming languages see [0.30000000000000004.com](https://0.30000000000000004.com).

Machine learning (ML) requires a reasonable understanding of derivatives and gradient descent. See [www.khanacademy.org/math/ap-calculus-ab/ab-derivative-intro](https://www.khanacademy.org/math/ap-calculus-ab/ab-derivative-intro) and [www.khanacademy.org/math/multivariable-calculus/multivariable-derivatives](https://www.khanacademy.org/math/multivariable-calculus/multivariable-derivatives) for a very intuitive explanation.

A basic understanding of statistics is helpful in getting a deeper understanding of how the algorithms work. See [www.openintro.org/stat/textbook.php](https://www.openintro.org/stat/textbook.php).

## 7.1. The right model to solve the problem

The following table is based on "Mastering Machine Learning with R" from Cory Lesmeister:

Text data	Word frequency
	Word cloud
	Topic models
	Lexical analysis
Univariate time series data	Auto-regressive integrated moving average (ARIMA)
	Exponential smoothing
	Linear regression
	ARCH
	GARCH
Multivariate time series data	Dynamic linear regression
	Vector autoregression (VAR)
	Granger causality
	Vector error correction model (VECM)
	Cointegration
Making recommendations	User-based collaborative filtering (UBCF)
	Item-based collaborative filtering (IBCF)
	Singular value decomposition (SVD)
	Principal components analysis (PCA)
Looking for associations	Apriori
	Equivalence class clustering with bottom-up lattice traversal (ECLAT)
Predicting a quantity	Linear regression
	Least absolute shrinkage and selection operator (LASSO)
	Ridge regression
	ElasticNet
	PCA regression
	Generalized additive models (GAM)
	Partial least squares (PLS)

Categorize unlabeled data that is not text or time series	Clustering
	Hierarchical
	K-means
	Partition around medoids
	Self-organizing map (SOM)
	Fuzzy clustering
Categorize labeled data that is not text or time series	Classification
	Logistic regression
	Linear discriminant analysis (LDA)
	K-nearest neighbors (KNN)
	Support vector machines
	Neural networks/deep learning
	Decision trees
	Random forest
	Gradient boosting
	Naïve Bayes
	Survival analysis

## 7.2. Applications

[www.seeitmarket.com/machine-learning-meets-investment-portfolio-management-18003](http://www.seeitmarket.com/machine-learning-meets-investment-portfolio-management-18003)  
[modeldepot.github.io/tfjs-yolo-tiny-demo](http://modeldepot.github.io/tfjs-yolo-tiny-demo)  
[hackernoon.com/tensorflow-js-real-time-object-detection-in-10-lines-of-code-baf15dfb95b2](http://hackernoon.com/tensorflow-js-real-time-object-detection-in-10-lines-of-code-baf15dfb95b2)  
[experiments.withgoogle.com/collection/creatability](http://experiments.withgoogle.com/collection/creatability)

## 7.3. Tools

### 7.3.1. Python

[dev.to/underdogio/python-for-javascript-developers](http://dev.to/underdogio/python-for-javascript-developers)  
[book.pythontips.com/en/latest/object\\_introspection.html](http://book.pythontips.com/en/latest/object_introspection.html)  
[docs.python.org/3/reference/datamodel.html](http://docs.python.org/3/reference/datamodel.html)

To inspect an object in Python, see [stackoverflow.com/questions/1006169/how-do-i-look-inside-a-python-object](http://stackoverflow.com/questions/1006169/how-do-i-look-inside-a-python-object) and [stackoverflow.com/questions/192109/is-there-a-built-in-function-to-print-all-the](http://stackoverflow.com/questions/192109/is-there-a-built-in-function-to-print-all-the)

[current-properties-and-values-of-a/192184](#).

### 7.3.2. R

From [www.r-project.org](http://www.r-project.org):

R is a free software environment for statistical computing and graphics

Use [r-script](#) or [OpenCPU](#) to use R from JS (cf. [stackoverflow.com/questions/17665565/is-there-a-way-to-run-r-code-from-javascript](https://stackoverflow.com/questions/17665565/is-there-a-way-to-run-r-code-from-javascript)).

## 7.4. Regression

### 7.4.1. Linear regression

Linear regression can be used if we have a set of input and output data and suspect that there is a linear relationship between them, i.e. the output value can be calculated from the input value(s) by applying a linear function. If we have only one input value, also called explanatory variable, per output value we talk about univariate linear regression. If we have several explanatory variables we use multivariate linear regression.

#### Univariate linear regression

Let's say we have a series of systolic blood pressure measurements from people of different ages and we'd like to be able to estimate the systolic blood pressure for a person of any age using a linear function of the form  $y = ax + b$ . This function is called our hypothesis. We need to determine  $a$  and  $b$  so that our hypothesis matches our given data set as closely as possible, which will give us some confidence that it will produce a sensible result for a new data point. This amounts to finding a line that passes through our data points so as to minimise the average distance to each data point.

To determine the best values for  $a$  and  $b$  we will proceed as follows:

1. Start with some random values for  $a$  and  $b$ .
2. Calculate the average squared distance of our hypothesis from the correct data using the formula  $\frac{1}{2m} \sum_{i=1}^m (h(x^{(i)}) - y^{(i)})^2$  where  $m$  is the number of data points,  $x^{(i)}$  the  $i$ th input and  $y^{(i)}$  the  $i$ th output value and  $h$  our hypothesis function. This formula is known as the cost or error function and our goal is to minimize it.
3. Calculate the derivative of  $h$  wrt to  $a$  as well as the derivative of  $h$  wrt to  $b$ .
4. Deduct the derivative times a factor, called the learning rate, from  $a$  and  $b$  respectively. This method is called [gradient descent](#) and moves our parameters closer to their optimum values. How to we determine the learning rate? If we set it too low the algorithm will take a long time to converge to a minimum as it is taking only very small steps. If we set the rate too high we may never reach the minimum as we take too big steps. We can use our cost function to check whether our learning rate is too high. Normally the cost should go down with each iteration as it converges to the minimum. If it goes up that's an indication that our learning rate is too high and needs to be reduced. In the case of univariate regression we can use the inverse of the second derivative of  $h$ , i.e.  $\frac{1}{f''(x)}$  (cf. [developers.google.com/machine-learning/](https://developers.google.com/machine-learning/)

[crash-course/reducing-loss/learning-rate](#)).

5. Repeat steps 2 to 4 above until a set number of iterations has been executed or the cost decreases by less than a given amount for an iteration.

Let's calculate the derivatives required for point 3 above. If you don't remember derivatives rules see [www.mathsisfun.com/calculus/derivatives-rules.html](http://www.mathsisfun.com/calculus/derivatives-rules.html) for a quick refresher.

[students.btsi.lu/evegi144/WAD/JS/AI/Linear%20regression/index1.html](http://students.btsi.lu/evegi144/WAD/JS/AI/Linear%20regression/index1.html)

```
'use strict';
// Based on https://www.robinwieruch.de/linear-regression-gradient-descent-javascript
// Systolic blood pressure by age.
// Source: http://people.sc.fsu.edu/~jburkardt/datasets/regression/x03.txt
const X = [39, 47, 45, 47, 65, 46, 67, 42, 67, 56, 64, 56, 59, 34, 42, 48, 45, 17, 20,
19,
    36, 50, 39, 21, 44, 53, 63, 29, 25, 69],
    Y = [144, 220, 138, 145, 162, 142, 170, 124, 158, 154, 162, 150, 140, 110, 128, 130,
    135, 114, 116, 124, 136, 142, 120, 120, 160, 158, 144, 130, 125, 175];
const LEARNING_RATE = 0.0003;
let a = 0; // a in y = ax + b
let b = 0; // b in y = ax + b

// The hypothesis function is our current best fit estimate for the linear regression
function.
const hypothesis = x => a * x + b;

// In linear regression the cost function calculates the sum of the squared
differences.
const cost = () => {
    let sum = 0;
    for (let i = 0; i < X.length; i++) sum += Math.pow(hypothesis(X[i]) - Y[i], 2);
    return sum / (2 * X.length);
};

const learn = () => {
    let aSum = 0;
    let bSum = 0;

    for (let i = 0; i < X.length; i++) {
        aSum += (hypothesis(X[i]) - Y[i]) * X[i];
        bSum += hypothesis(X[i]) - Y[i];
    }

    a = a - (LEARNING_RATE / X.length) * aSum;
    b = b - (LEARNING_RATE / X.length) * bSum;
};

const plot = () => {
    const MINX = Math.min(...X), MAXX = Math.max(...X), MINY = Math.min(...Y),
        MAXY = Math.max(...Y), WIDTH = 1000, HEIGHT = 770;
```

```

let html = `<svg width=${WIDTH} height=${HEIGHT} style='background-color:
lightblue'>`;
for (let i = 0; i <= 70; i += 5)
  html += `<text x=${50 + 10 * i} y=${HEIGHT - 15}>${i}</text>
    <line x1=${50 + 10 * i} y1=${HEIGHT - 50} x2=${50 + 10 * i} y2=${HEIGHT -
3 * 230 - 50} stroke=black stroke-width=1></line>`;
for (let i = 0; i <= 230; i += 10)
  html += `<text x=10 y=${HEIGHT - 3 * i - 50}>${i}</text>
    <line x1=50 y1=${HEIGHT - 3 * i - 50} x2=${50 + 10 * 70} y2=${HEIGHT - 3 *
i - 50} stroke=black stroke-width=1></line>`;
for (let i = 0; i < X.length; i++)
  html += `<circle cx=${50 + 10 * X[i]} cy=${HEIGHT - 3 * Y[i] - 50} r=10
stroke=black stroke-width=3 fill=yellow></circle>`;
const X1 = MINX, Y1 = hypothesis(X1), X2 = MAXX, Y2 = hypothesis(X2);
html += `<line x1=${50 + 10 * X1} y1=${HEIGHT - 3 * Y1 - 50} x2=${50 + 10 * X2}
y2=${HEIGHT - 3 * Y2 - 50} stroke=black stroke-width=5></line>`;
html += `<text x=${WIDTH - 230} y=32 font-size=26 fill=black>Learn</text>
  <rect x=${WIDTH - 240} y=10 height=30 width=80
    stroke=black stroke-width=2 fill=green opacity=.5 rx=10></rect>
  <foreignObject x=${WIDTH - 150} y=12 width=70 height=50>
    <div xmlns=http://www.w3.org/1999/xhtml><input value=1></div>
  </foreignObject>
  <text x=${WIDTH - 75} y=27>iteration(s)</text>
  <text x=${WIDTH - 240} y=70>Hypothesis:  $y = \{a.toFixed(2)\} * x + \{b.toFixed(2)\}$ </text>
  <text x=${WIDTH - 240} y=100>Cost:  $\{cost().toFixed(2)\}$ </text>
</svg>`;
document.querySelector('body').innerHTML = html;
document.querySelector('rect').addEventListener('click', iterate);
};

const iterate = () => {
  let iterations = document.querySelector('input').value;
  if (!iterations) iterations = 1;
  for (let i = 1; i <= iterations; i++) learn();
  plot();
};

addEventListener('load', plot);

```

We can simplify our life and use an external library to do the work:

```
'use strict';

const ml = require('ml-regression');
const csv = require('csvtojson');

// Data from http://mathbits.com/MathBits/TISection/Statistics2/linearREAL.htm
let inputs = [20, 16, 19.8, 18.4, 17.1, 15.5, 14.7, 15.7, 15.4, 16.3, 15, 17.2, 16,
17, 14.4];
let outputs = [88.6, 71.6, 93.3, 84.3, 80.6, 75.2, 69.7, 71.6, 69.4, 83.3, 79.6, 82.6,
80.6, 83.5, 76.3];

let regression = new ml.SLR(inputs, outputs);
console.log(regression.toString(3)); // === 'f(x) = 3.41 * x + 22.8';
```

## Vectorization

[www.robinwieruch.de/linear-regression-gradient-descent-vectorization-javascript](http://www.robinwieruch.de/linear-regression-gradient-descent-vectorization-javascript)

In our linear regression example each learning iteration implies the execution of a loop over the  $m$  data points which is very time consuming. We can simplify this by using matrix multiplication. Given the optimisations implemented in good math libraries (cf. [softwareengineering.stackexchange.com/questions/312445/why-does-expressing-calculations-as-matrix-multiplications-make-them-faster](https://softwareengineering.stackexchange.com/questions/312445/why-does-expressing-calculations-as-matrix-multiplications-make-them-faster)) this can lead to significant speed improvements.

## Multivariate linear regression

### 7.4.2. Polynomial regression

[www.robinwieruch.de/polynomial-regression-model-selection-javascript](http://www.robinwieruch.de/polynomial-regression-model-selection-javascript)

## 7.5. Neural networks

[www.3blue1brown.com](http://www.3blue1brown.com)

[www.youtube.com/watch?v=aircAruvnKk](https://www.youtube.com/watch?v=aircAruvnKk)

[www.youtube.com/watch?v=IHZwWFHWa-w](https://www.youtube.com/watch?v=IHZwWFHWa-w)

[wagenkaartje.github.io/neataptic](https://wagenkaartje.github.io/neataptic)

[github.com/ssusnic/Machine-Learning-Flappy-Bird](https://github.com/ssusnic/Machine-Learning-Flappy-Bird)

[WebDNN](#) is the fastest DNN execution framework for the browser.

## 7.6. Reinforcement learning

### 7.6.1. Introduction

According to [en.wikipedia.org/wiki/Reinforcement\\_learning](https://en.wikipedia.org/wiki/Reinforcement_learning):

Reinforcement learning (RL) is an area of machine learning inspired by behaviourist psychology, concerned with how software agents ought to take actions in an environment so as to maximize some notion of cumulative reward. ... Reinforcement learning differs from standard supervised learning in that correct input/output pairs need not be presented, and sub-optimal actions need not be explicitly corrected. Instead the focus is on performance, which involves finding a balance between exploration (of uncharted territory) and exploitation (of current knowledge).

RL is also called neuro-dynamic programming (NDP) or approximate dynamic programming (ADP).

Based on [incompleteideas.net/book/the-book-2nd.html](https://incompleteideas.net/book/the-book-2nd.html), a reinforcement learning system consists of four main elements:

1. A *policy* defines the learning agent's way of behaving at a given time.
2. A *reward signal* defines the immediate goal.
3. A *value function* specifies what is good in the long run. The *value* of a state is the total amount of reward an agent can expect to accumulate over the future, starting from that state.
4. A *model* (optional) mimics the behavior of the environment and is used for planning.

These elements enable our learning agent to have explicit goals, sense aspects of its environment and choose actions to influence its environment. The agent's sole goal is to maximize the total reward it receives over the long run.

We assume that the system we wish to control is stochastic, i.e. random. We also assume that the problem we try to solve is a [Markov Decision Process \(MDP\)](#):

A Markov decision process is a discrete time stochastic control process. At each time step, the process is in some state  $s$ , and the decision maker may choose any action  $a$  that is available in state  $s$ . The process responds at the next time step by randomly moving into a new state  $s'$ , and giving the decision maker a corresponding reward  $R_a(s,s')$ .

The probability that the process moves into its new state  $s$  is influenced by the chosen action. Specifically, it is given by the state transition function  $P_a(s,s')$ . Thus, the next state  $s'$  depends on the current state  $s$  and the decision maker's action  $a$ . But given  $s$  and  $a$ , it is conditionally independent of all previous states and actions; in other words, the state transitions of an MDP satisfy the Markov property.

The action  $\pi(s)$  taken in state  $s$  under deterministic policy  $\pi$  is:  $\pi(s) = \{ \arg \max_a \{ \sum_{s'} P_a(s,s') (R_a(s,s') + \gamma V(s')) \} \}$



An excellent deep reinforcement learning course can be found at [simoninithomas.github.io/Deep\\_reinforcement\\_learning\\_Course](https://simoninithomas.github.io/Deep_reinforcement_learning_Course).

## 7.6.2. Q-learning

One approach to learning the value function is the Q-learning algorithm. Given a state and action it calculates the expected discounted cumulative reward:

$$Q^{\pi}(s_t, a_t) = E[R_{t+1} + \gamma R_{t+2} + \gamma^2 R_{t+3} + \dots | s_t, a_t]$$

[web.mst.edu/~gosavia/tutorial.pdf](http://web.mst.edu/~gosavia/tutorial.pdf)

[sites.ualberta.ca/~szepesva/RLBook.html](http://sites.ualberta.ca/~szepesva/RLBook.html)

[janhuenermann.com/projects/learning-to-drive](http://janhuenermann.com/projects/learning-to-drive)

[mnemstudio.org/path-finding-q-learning.htm](http://mnemstudio.org/path-finding-q-learning.htm)

[www.robinwieruch.de/machine-learning-javascript-web-developers](http://www.robinwieruch.de/machine-learning-javascript-web-developers)

[arxiv.org/abs/1804.04577](http://arxiv.org/abs/1804.04577)

[skymind.ai/wiki/deep-reinforcement-learning](http://skymind.ai/wiki/deep-reinforcement-learning)

[towardsdatascience.com/curiosity-driven-learning-made-easy-part-i-d3e5a2263359](http://towardsdatascience.com/curiosity-driven-learning-made-easy-part-i-d3e5a2263359)

[www.metacar-project.com](http://www.metacar-project.com)

[researchers.lille.inria.fr/~lazaric](http://researchers.lille.inria.fr/~lazaric)

[students.btsi.lu/evegi144/WAD/JS/AI/Reinforcement%20learning/rooms1.html](http://students.btsi.lu/evegi144/WAD/JS/AI/Reinforcement%20learning/rooms1.html)

```
// Corrected version of http://mnemstudio.org/ai/path/q_learning_js_ex1.txt
'use strict';

const qSize = 6;
const gamma = 0.8;
const iterations = 1000;
let initialStates = [1, 3, 5, 2, 4, 0];

/*
Each row represents actions for a state, i.e. R[0][1] = -1 -> we cannot go from
room/state
0 to 1. R[2][3] = 0 means we can go from room 2 to 3 but have not yet reached the
goal
(room 5). R[1][5] = 1000 -> we can go from room 1 directly to 5 and reach our goal.
*/
let R = [[-1, -1, -1, -1, 0, -1],
  [-1, -1, -1, 0, -1, 100],
  [-1, -1, -1, 0, -1, -1],
  [-1, 0, 0, -1, 0, -1],
  [0, -1, -1, 0, -1, 100],
  [-1, 0, -1, -1, 0, 100]];

let Q = new Array(qSize);
for (let a = 0; a < qSize; a++) Q[a] = new Array(qSize);
let currentState = 0;
```

```

const initialize = () => {
  for (let i = 0; i <= (qSize - 1); i++)
    for (let j = 0; j <= (qSize - 1); j++) Q[i][j] = 0;
};

const maximum = (state, returnIndexOnly) => {
  // if returnIndexOnly = true, a Q matrix index is returned.
  // if returnIndexOnly = false, a Q matrix element is returned.

  let winner = 0;
  let foundNewWinner = false;
  let done = false;

  winner = 0;

  do {
    foundNewWinner = false;
    for (let m = 0; m < qSize; m++) {
      if ((m < winner) || (m > winner)) //Avoid self-comparison.
        if (Q[state][m] > Q[state][winner]) {
          winner = m;
          foundNewWinner = true;
        }
    }
  }

  if (!foundNewWinner) done = true;
} while (!done);

if (returnIndexOnly) return winner;
else return Q[state][winner];
};

const reward = action => {
  return Math.round(R[currentState][action] + (gamma * maximum(action, false)));
};

const episode = initialState => {
  currentState = initialState;

  //Travel from state to state until goal state is reached.
  do chooseAnAction(); while (currentState !== 5);

  //When currentState = 5, run through the set once more for convergence.
  for (let i = 0; i < qSize; i++) chooseAnAction();
};

const chooseAnAction = () => {
  let possibleAction = 0;

  //Randomly choose a possible action connected to the current state.
  possibleAction = getRandomAction(qSize, 0);

```

```

    if (R[currentState][possibleAction] >= 0) {
      Q[currentState][possibleAction] = reward(possibleAction);
      currentState = possibleAction;
    }
  };

const getRandomAction = (upperBound, lowerBound) => {
  let action = 0;
  let choiceIsValid = false;
  let range = upperBound - lowerBound;

  //Randomly choose a possible action connected to the current state.
  do {
    //Get a random value between 0 and 6.
    action = lowerBound + Math.round(range * Math.random());

    if (R[currentState][action] > -1) choiceIsValid = true;
  } while (!choiceIsValid);

  return action;
};

const init = () => {
  let initialState = 0;
  initialize();
  for (let i = 0; i < iterations; i++)
    for (let j = 0; j < qSize; j++) episode(initialStates[j]);
  let str = '';
  for (let i = 0; i < qSize; i++) {
    for (let j = 0; j < qSize; j++) str += Q[i][j] + ' ';
    str += '<br>';
  }
  document.body.innerHTML = 'Q matrix values:<br>' + str +
    '<br>Shortest routes from initial states:<br>';
  // Now let's display the shortest path from each starting node.
  for (let state = 0; state < qSize; state++) {
    let str = state + ' ';
    let maxState = state;
    do {
      maxState = maximum(maxState, true);
      str += maxState + ' ';
    } while (maxState < 5);
    document.body.innerHTML += str + '<br>';
  }
};

addEventListener('load', init);

```

### 7.6.3. OpenAI Gym

[github.com/openai/gym](https://github.com/openai/gym)

[github.com/udacity/deep-learning/blob/master/reinforcement/Q-learning-cart.ipynb](https://github.com/udacity/deep-learning/blob/master/reinforcement/Q-learning-cart.ipynb)

[www.oreilly.com/learning/introduction-to-reinforcement-learning-and-openai-gym](http://www.oreilly.com/learning/introduction-to-reinforcement-learning-and-openai-gym)

[simoninithomas.github.io/Deep\\_reinforcement\\_learning\\_Course](https://simoninithomas.github.io/Deep_reinforcement_learning_Course)

[github.com/simoninithomas/Deep\\_reinforcement\\_learning\\_Course/blob/master/Deep%20Q%20Learning/Doom/Deep%20Q%20learning%20with%20Doom.ipynb](https://github.com/simoninithomas/Deep_reinforcement_learning_Course/blob/master/Deep%20Q%20Learning/Doom/Deep%20Q%20learning%20with%20Doom.ipynb)

[www.freecodecamp.org/news/an-introduction-to-deep-q-learning-lets-play-doom-54d02d8017d8](http://www.freecodecamp.org/news/an-introduction-to-deep-q-learning-lets-play-doom-54d02d8017d8)

#### Installation

You need to have Python and pip installed.

```
apt install python3 python3-pip
apt install -y python-numpy python-dev cmake zlib1g-dev libjpeg-dev xvfb ffmpeg xorg-dev
python-opengl libboost-all-dev libsdl2-dev swig
pip3 install 'gym[all]'
```

Create a test file:

```
import gym
env = gym.make('MsPacman-v0')
state = env.reset()
reward, info, done = None, None, None
maxReward = 0
while done != True:
    state, reward, done, info = env.step(env.action_space.sample())
    env.render()
    if reward > maxReward:
        maxReward = reward
        print(reward)
# https://github.com/openai/gym/issues/893
env.close()
```

Run the file using `python3 <file>` and study [www.oreilly.com/learning/introduction-to-reinforcement-learning-and-openai-gym](http://www.oreilly.com/learning/introduction-to-reinforcement-learning-and-openai-gym).

To use Gym Retro we'll also need [Docker](#) installed. Run `pip3 install gym-retro` (cf. [github.com/openai/retro](https://github.com/openai/retro)).

Create a test file:

```
import retro
env = retro.make(game='Airstriker-Genesis', state='Level1')
env.reset()
for _ in range(1000):
    env.render()
    env.step(env.action_space.sample()) # take a random action
# https://github.com/openai/gym/issues/893
env.close()
```

To run the gym remotely use `ssh -X username@hostname` (cf. [stackoverflow.com/questions/40195740/how-to-run-openai-gym-render-over-a-server](https://stackoverflow.com/questions/40195740/how-to-run-openai-gym-render-over-a-server)).

To get a list of all the environments installed on your system, see [github.com/openai/gym/blob/master/examples/scripts/list\\_envs](https://github.com/openai/gym/blob/master/examples/scripts/list_envs).

#### 7.6.4. ViZDoom

Doom-based AI Research Platform for Reinforcement Learning from Raw Visual Information.

[github.com/mwydmuch/ViZDoom](https://github.com/mwydmuch/ViZDoom)

[github.com/mwydmuch/ViZDoom/issues/365](https://github.com/mwydmuch/ViZDoom/issues/365)

To get [www.freecodecamp.org/news/an-introduction-to-deep-q-learning-lets-play-doom-54d02d8017d8](https://www.freecodecamp.org/news/an-introduction-to-deep-q-learning-lets-play-doom-54d02d8017d8) to work you need to copy [github.com/mwydmuch/ViZDoom/blob/master/scenarios/basic.wad](https://github.com/mwydmuch/ViZDoom/blob/master/scenarios/basic.wad) to your project directory.

#### 7.6.5. Arcade Learning Environment (ALE)

[github.com/mgbellemare/Arcade-Learning-Environment](https://github.com/mgbellemare/Arcade-Learning-Environment)

#### 7.6.6. Tabular solution methods

If the state and action spaces are small enough for the value functions to be represented as arrays RL can often find the optimal value function and the optimal policy.

### 7.7. Tensorflow

[www.tensorflow.org/js](https://www.tensorflow.org/js)

[github.com/tensorflow/tfjs-examples](https://github.com/tensorflow/tfjs-examples)

[playground.tensorflow.org](https://playground.tensorflow.org)

[medium.com/emergent-future/simple-reinforcement-learning-with-tensorflow-part-0-q-learning-with-tables-and-neural-networks-d195264329d0](https://medium.com/emergent-future/simple-reinforcement-learning-with-tensorflow-part-0-q-learning-with-tables-and-neural-networks-d195264329d0)

[ml5js.org](https://ml5js.org)

[github.com/BeTomorrow/ReImproveJS](https://github.com/BeTomorrow/ReImproveJS)

[www.youtube.com/watch?v=SV-cgdohtTA](https://www.youtube.com/watch?v=SV-cgdohtTA)

[2019.jsconf.eu/nick-kreeger/tensorflowjs-bringing-machine-learning-to-the-web-and-beyond.html](https://2019.jsconf.eu/nick-kreeger/tensorflowjs-bringing-machine-learning-to-the-web-and-beyond.html)

[www.analyticsvidhya.com/blog/2019/06/build-machine-learning-model-in-your-browser-tensorflow-js-deeplearn-js](https://www.analyticsvidhya.com/blog/2019/06/build-machine-learning-model-in-your-browser-tensorflow-js-deeplearn-js)

[blog.apptension.com/2018/06/27/tensorflow-js-machine-learning-and-flappy-bird-frontend-artificial-intelligence](https://blog.apptension.com/2018/06/27/tensorflow-js-machine-learning-and-flappy-bird-frontend-artificial-intelligence)

[becominghuman.ai/machine-learning-in-the-browser-using-tensorflow-js-3e453ef2c68c](https://becominghuman.ai/machine-learning-in-the-browser-using-tensorflow-js-3e453ef2c68c)

### 7.7.1. Tensors

A tensor is a structure of numbers, e.g. a number, a vector or a matrix. A tensor is defined by its rank, which is its number of dimensions. For instance a number (or scalar) has rank 0, a vector rank 1, a m by n matrix rank 2 and a m by n by p matrix rank 3. Study [medium.freecodecamp.org/a-quick-introduction-to-tensorflow-js-a046e2c3f1f2](https://medium.freecodecamp.org/a-quick-introduction-to-tensorflow-js-a046e2c3f1f2) for a quick introduction to tensors in TensorFlow.js.

[students.btsi.lu/evegi144/WAD/JS/AI/Reinforcement%20learning/tensorflow0.html](https://students.btsi.lu/evegi144/WAD/JS/AI/Reinforcement%20learning/tensorflow0.html)

```
'use strict';

const init = () => {
  let t1 = tf.tensor(4.5);
  t1.print();
  let t2 = tf.tensor([2, 5, 99]);
  t2.print();

  // The second parameter specifies the shape, i.e. 4 rows and 1 column.
  let xs = tf.tensor2d([1, 3, 5, 7], [4, 1]);
  let ys = tf.tensor2d([1, 3, 5, 7], [1, 4]);
  let zs = tf.tensor2d([1, 3, 5, 7], [2, 2]);
  xs.print();
  ys.print();
  zs.print();

  t1 = t1.add(t1);
  t1.print();
  xs.mul(ys).print();
  ys.mul(xs).print();
};

addEventListener('load', init);
```

If we don't need a tensor anymore we can call its **dispose** method to free memory:

```
const x = tf.tensor([1,2,3]);
x.dispose();
```

Having to do this for each tensor would quickly become unwieldy therefore TF.js offers a **tidy** operator that works as follows:

[students.btsi.lu/evegi144/WAD/JS/AI/Reinforcement%20learning/tensorflow1.html](https://students.btsi.lu/evegi144/WAD/JS/AI/Reinforcement%20learning/tensorflow1.html)

```
'use strict';

const init = () => {
  /*const xs = tf.tensor3d([
    [0, 0, 0],
    [1, 1, 1],
    [2, 2, 2],
    [4, 4, 4],
    [7, 7, 7],
    [10, 10, 10],
    [0, 0, 0],
    [0, 0, 0],
    [0, 0, 0],
    [0, 0, 0],
  ]);*/

  /*const model = tf.sequential();
  model.add(tf.layers.dense({units: 32, inputShape: [50]}));
  model.add(tf.layers.dense({units: 4}));
  console.log(JSON.stringify(model.outputs[0].shape));*/

  // y = 2 ^ 2 + 1
  const y = tf.tidy(() => {
    // a, b, and one will be cleaned up when the tidy ends.
    const one = tf.scalar(1);
    const a = tf.scalar(2);
    const b = a.square();

    console.log('numTensors (in tidy): ' + tf.memory().numTensors);

    // The value returned inside the tidy function will return
    // through the tidy, in this case to the variable y.
    return b.add(one);
  });

  console.log('numTensors (outside tidy): ' + tf.memory().numTensors);
  y.print();
};

addEventListener('load', init);
```

## 7.7.2. Layers

[students.btsi.lu/evegi144/WAD/JS/AI/Reinforcement%20learning/tensorflow2.html](https://students.btsi.lu/evegi144/WAD/JS/AI/Reinforcement%20learning/tensorflow2.html)

```
'use strict';

const init = async () => {
  // Notice there is no 'import' statement. 'tf' is available on the index-page
  // because of the script tag above.

  // Define a model for linear regression.
  const model = tf.sequential();
  model.add(tf.layers.dense({units: 1, inputShape: [1]}));

  // Prepare the model for training: Specify the loss and the optimizer.
  model.compile({loss: 'meanSquaredError', optimizer: 'sgd'});

  // Generate some synthetic data for training.
  // The second parameter specifies the shape, i.e. 4 rows and 1 column.
  const xs = tf.tensor2d([1, 2, 3, 4], [4, 1]);
  const ys = tf.tensor2d([1, 3, 5, 7], [4, 1]);

  // Train the model using the data.
  await model.fit(xs, ys);
  // Use the model to do inference on a data point the model hasn't seen before:
  // Open the browser devtools to see the output
  model.predict(tf.tensor2d([5], [1, 1])).print();
};

addEventListener('load', init);
```

Here's a more elaborate example from [www.tensorflow.org/js/tutorials/training/linear\\_regression](https://www.tensorflow.org/js/tutorials/training/linear_regression):

[students.btsi.lu/evegi144/WAD/JS/AI/Tensorflow/Car mpg prediction/index.html](https://students.btsi.lu/evegi144/WAD/JS/AI/Tensorflow/Car%20mpg%20prediction/index.html)<sup>[^]</sup>

```
'use strict';

/**
 * Get the car data reduced to just the variables we are interested
 * and cleaned of missing data.
 */
const getData = async () => {
  const carsDataReq = await fetch('https://storage.googleapis.com/tfjs-
tutorials/carsData.json');
  const carsData = await carsDataReq.json();
  const cleaned = carsData.map(car => ({
    mpg: car.Miles_per_Gallon,
    horsepower: car.Horsepower,
  })))
```

```

.filter(car => (car.mpg != null && car.horsepower != null));

return cleaned;
};

/**
 * Convert the input data to tensors that we can use for machine
 * learning. We will also do the important best practices of _shuffling_
 * the data and _normalizing_ the data
 * MPG on the y-axis.
 */
const convertToTensor = data => {
  // Wrapping these calculations in a tidy will dispose any
  // intermediate tensors.

  return tf.tidy(() => {
    // Step 1. Shuffle the data
    tf.util.shuffle(data);

    // Step 2. Convert data to Tensor
    const inputs = data.map(d => d.horsepower)
    const labels = data.map(d => d.mpg);

    const inputTensor = tf.tensor2d(inputs, [inputs.length, 1]);
    const labelTensor = tf.tensor2d(labels, [labels.length, 1]);

    //Step 3. Normalize the data to the range 0 - 1 using min-max scaling
    const inputMax = inputTensor.max();
    const inputMin = inputTensor.min();
    const labelMax = labelTensor.max();
    const labelMin = labelTensor.min();

    const normalizedInputs = inputTensor.sub(inputMin).div(inputMax.sub(inputMin));
    const normalizedLabels = labelTensor.sub(labelMin).div(labelMax.sub(labelMin));

    return {
      inputs: normalizedInputs,
      labels: normalizedLabels,
      // Return the min/max bounds so we can use them later.
      inputMax,
      inputMin,
      labelMax,
      labelMin,
    }
  });
};

const createModel = () => {
  // Create a sequential model
  const model = tf.sequential();

```

```

// Add a single hidden layer
model.add(tf.layers.dense({inputShape: [1], units: 50, useBias: true}));
model.add(tf.layers.dense({units: 50, activation: 'relu'}));
model.add(tf.layers.dense({units: 50, activation: 'relu'}));

// Add an output layer
model.add(tf.layers.dense({units: 1, useBias: true}));

return model;
};

const trainModel = async (model, inputs, labels) => {
  // Prepare the model for training.
  model.compile({
    optimizer: tf.train.adam(),
    loss: tf.losses.meanSquaredError,
    metrics: ['mse'],
  });

  const batchSize = 32;
  const epochs = 50;

  return await model.fit(inputs, labels, {
    batchSize,
    epochs,
    shuffle: true,
    callbacks: tfvis.show.fitCallbacks(
      { name: 'Training Performance' },
      ['loss', 'mse'],
      { height: 200, callbacks: ['onEpochEnd'] }
    )
  });
};

const testModel = (model, inputData, normalizationData) => {
  const {inputMax, inputMin, labelMin, labelMax} = normalizationData;

  // Generate predictions for a uniform range of numbers between 0 and 1;
  // We un-normalize the data by doing the inverse of the min-max scaling
  // that we did earlier.
  const [xs, preds] = tf.tidy(() => {

    const xs = tf.linspace(0, 1, 100);
    const preds = model.predict(xs.reshape([100, 1]));

    const unNormXs = xs
      .mul(inputMax.sub(inputMin))
      .add(inputMin);

    const unNormPreds = preds
      .mul(labelMax.sub(labelMin))

```

```

        .add(labelMin);

    // Un-normalize the data
    return [unNormXs.dataSync(), unNormPreds.dataSync()];
});

const predictedPoints = Array.from(xs).map((val, i) => {
    return {x: val, y: preds[i]}
});

const originalPoints = inputData.map(d => ({
    x: d.horsepower, y: d.mpg,
}));

tfvis.render.scatterplot(
    {name: 'Model Predictions vs Original Data'},
    {values: [originalPoints, predictedPoints], series: ['original', 'predicted']},
    {
        xLabel: 'Horsepower',
        yLabel: 'MPG',
        height: 300
    }
);
};

const run = async () => {
    // Load and plot the original input data that we are going to train on.
    const data = await getData();
    const values = data.map(d => ({
        x: d.horsepower,
        y: d.mpg,
    }));

    /*tfvis.render.scatterplot(
        {name: 'Horsepower v MPG'},
        {values},
        {
            xLabel: 'Horsepower',
            yLabel: 'MPG',
            height: 300
        }
    );*/

    // Create the model
    const model = createModel();
    tfvis.show.modelSummary({name: 'Model Summary'}, model);
    // Convert the data to a form we can use for training.
    const tensorData = convertToTensor(data);
    const {inputs, labels} = tensorData;

```

```
// Train the model
await trainModel(model, inputs, labels);
console.log('Done Training');

    // Make some predictions using the model and compare them to the
// original data
testModel(model, data, tensorData);
};

document.addEventListener('DOMContentLoaded', run);
```

# Chapter 8. Exercises

## 8.1. HTML5

### 8.1.1. Exercise 1

Write an HTML5 document with the title **My first HTML5 page** that includes the empty files **ex1.css** (CSS3) and **ex1.js** (JavaScript), which you create. Your page displays the text **I'm learning HTML5!** and passes HTML5 and CSS3 validation without errors.

### 8.1.2. Exercise 2

Create an HTML5 document, which produces the following output (without the border):

*HTML5 is a markup language used for structuring and presenting content.*  
JavaScript allows us to make a decision based on a condition, like this:  
`if (x >= 0) console.log('x is positive'); else console.log('x is negative');`  
Here *x* is a variable. If we give it a value of 23, the script output will be *x is positive*.

Make sure to choose tags that are semantically correct, i.e. convey the right meaning. Don't forget to validate your page.

### 8.1.3. Exercise 3

Create an HTML5 document with a header and an ordered list that uses the three ordered list attributes.

### 8.1.4. Exercise 4

Create an HTML5 document with a description list that describes the competences of the SYSEX1 module (cf. [portal.education.lu/programmes/Programme-Formation-professionnelle](http://portal.education.lu/programmes/Programme-Formation-professionnelle)).

### 8.1.5. Exercise 5

Create two HTML5 documents that look like this:

**HTML Exercise 5**  
**A hyperlink adventure**  
[Page2 LTAM](#)

**This is page 2 of the exercise.**

The first page is named **ex5.html**, the second **ex5p2.html**. The first link opens the second page in the same window/tab. The second link opens the page [www.ltam.lu](http://www.ltam.lu), also in the same window/tab. The first page includes the following external style sheet, which you should save under the name

ex5.css:

```
ul {  
  list-style: none;  
  padding: 0;  
  margin: 0;  
}
```

```
li { display: inline; }
```

### 8.1.6. Exercise 6

Create an HTML5 document that uses an image, that you have created, as navigation bar. The user can click on different parts of the image, which will take him to other pages. The main part of the document can be empty. Here is a sample solution:

[www.youtube.com/watch?v=-H2x\\_tGaxSM](http://www.youtube.com/watch?v=-H2x_tGaxSM)

## 8.2. JavaScript

### 8.2.1. Exercise 1

In a group execute the "marching orders" found in the book at [csunplugged.org/books](http://csunplugged.org/books).

### 8.2.2. Exercise 2

Complete all 15 levels of [lightbot.lu](http://lightbot.lu).

### 8.2.3. Exercise 3

Create a valid HTML5 document that includes an external script. The script defines a function that displays a message box with a text of your choice. Your `<main>` element includes an image which, if clicked, displays the message box.

### 8.2.4. Exercise 4

Create a valid HTML5 document with an embedded script that uses all the basic input and output functions seen in [Basic input and output](#). Take the opportunity to experiment to your heart's content.

### 8.2.5. Exercise 5

Write JavaScript that declares a variable, gives it a value and writes its value to the console.

### 8.2.6. Exercise 6

Create a web page that declares four variables with the values `'Text'`, `123`, `23.45` and `false` respectively. It then generates an alert that displays the type of each one of the four variables.

### 8.2.7. Exercise 7

Write a script that defines two string variables. The first one contains your first name, the second one your last name. Define a third variable that contains the first variable followed by a space followed by the second variable. Verify the result in the console.

### 8.2.8. Exercise 8

Write a script that defines two variables with the values 5 and 7.233453543 respectively. It then writes the sum with 2 decimals to the console.

### 8.2.9. Exercise 9

Without using a computer, write down the answers to the following questions, then check your solutions using the console:

[students.btsi.lu/evegi144/WAD/JS/exop1.js](https://students.btsi.lu/evegi144/WAD/JS/exop1.js)

```
"use strict";
// Let's declare a few variables first.
var x, y, s1 = 'Hello', car = {weight: 1500, color: 'black'};
// Q1: What is the value of x after this statement?
x = Math.round(10 / 3);
// Q2: What is the value of y after this statement?
y = 10 % 3;
// Q3: What is the value of x after this statement?
x += y;
// Q4: What value does this statement give?
x === y;
// Q5: And this one?
x % x;
// Q6: What value does this statement give?
111 + 111;
// Q7: And this one?
111 + '111';
// Q8: And this one?
'111' + 111;
// Q9: And this one?
x & 0;
// Q10: And this one?
x | 1;
// Q11: And this one?
x << 2;
// Q12: And this one?
typeof x;
// Q13: And this one?
typeof s1;
// Q14: And this one?
+s1;
// Q15: What is the value of car after this statement?
```

```

delete car.weight;
// Q16: What value does this statement give?
x === '5';
// Q17: And this one?
x == '5';
// Q18: And this one?
x === '4';
// Q19: And this one?
x == '4';
// Q20: And this one?
x === 4;
// Q21: And this one?
y !== '1';
// Q22: And this one?
y >= 1;
// Q23: And this one?
(x > 1) && (y < 27);
// Q24: And this one?
!(x < 1) || (y > 27);
// Q25: And this one?
(x < 1) ? 'yes' : 'no';

```

### 8.2.10. Exercise 10

Write a valid HTML5 document that implements the following items:

1. The user is asked to enter two integers, which are saved in variables `x` and `y`.
2. Variable `r` contains the remainder of `x` divided by `y`.
3. Variable `bool` contains `true` if `x` is bigger than `y`, otherwise `false`.
4. Variable `s1` contains the text 'Hello', variable `s2` the text 'guest'.
5. Variable `s` contains the text 'Hello guest!', which is created from `s1` and `s2` in a single statement.
6. A window displays the values of `r`, `bool` and `s`.

### 8.2.11. Exercise 11

Create a valid HTML5 document that asks the user a question and then informs him whether his answer was correct or not.

### 8.2.12. Exercise 12

Create a valid HTML5 document that asks for a grade between 1 and 60 and then tells the user whether it is very bad (<10), bad (<20), insufficient (<30), sufficient (<40), good (<50), very good (<60) or excellent (60).

### 8.2.13. Exercise 13

Create a valid HTML5 document that displays the current day of the week in text form using `switch`.

### 8.2.14. Exercise 14

Write a script that calculates and displays the factorial of a positive integer stored in the variable `x`. Remember, `!` is the mathematical symbol for factorial,  $x! = x * (x-1)!$  and  $1! = 1$ .

### 8.2.15. Exercise 15

The user is asked for a number. If the number is not positive, nothing happens. Otherwise, a count down, starting at the number and counting down to 0, is displayed using a message box.

### 8.2.16. Exercise 16

Write a script that calculates the greatest common divisor (gcd, cf. [en.wikipedia.org/wiki/Greatest\\_common\\_divisor](https://en.wikipedia.org/wiki/Greatest_common_divisor)) of two given positive integers stored in variables `a` and `b`. A simple method to do this is to realize that the gcd cannot be bigger than the smallest of the two numbers. So that's your starting point. Now you can find the gcd by testing whether the smallest of `a` and `b` divides both. If not, you subtract 1 from that value and test again. Your loop must terminate, because 1 is a divisor of every number.

### 8.2.17. Exercise 17

Write a function `sumUpTo` that takes a positive integer as parameter and returns the sum of all integers from 1 up to this parameter.

### 8.2.18. Exercise 18

Improve the previous function so that if the parameter is not a positive integer, the function returns false.

### 8.2.19. Exercise 19

Write a function `createTable` that takes two parameters, `w` and `h`. The function writes an HTML table with `w` columns and `h` rows. Put a random number from [1, 9] into each cell. Validate the result.

### 8.2.20. Exercise 20

Write a function `average` that returns the average of all the parameters provided. The function can take any number of parameters, including none. If no parameters are provided or if any of the parameters is not a number, the function returns `false`. Analyze and verify your function in the debugger.

### 8.2.21. Exercise 21

Write a script that creates an array containing the numbers from 1 to 10. Add a function `displayArr` that iterates over the array and writes each array element to the console. Call `displayArr`, remove the last element and call `displayArr` again.

### 8.2.22. Exercise 22

Write a script that creates a matrix, i.e. a 2-dimensional array of 10 rows, each containing 20 elements. Use `Math.random` (cf. [www.w3schools.com/jsref/jsref\\_random.asp](http://www.w3schools.com/jsref/jsref_random.asp)) to give a random value to each array element. Write a function `displayMat` that displays the matrix, which is passed as parameter in a readable way, i.e. each row on a new line.

### 8.2.23. Exercise 23

Create a page that triggers an alert when the user tries to resize the browser window.

### 8.2.24. Exercise 24

Create a page that generates an alert saying "Hi!" when the h key is pressed.

### 8.2.25. Exercise 25

Create a web app with a button. Clicking on the button displays an alert with the text "You clicked me!".

### 8.2.26. Exercise 26

Write a function that takes any number of string arguments representing element ids. For each ID the function retrieves the corresponding element. If any of the ids is invalid, an error message is thrown, otherwise the function returns an object that maps each id to its DOM element.

### 8.2.27. Exercise 27

Create a web app with 3 buttons. You may not use `getElementById`. Clicking on any of the buttons displays an alert with the text "You clicked button number " followed by the number of the button (1-3). For experts: enhance your script so that it works for any number of buttons.

### 8.2.28. Exercise 28

Create a web page with 3 `article` tags, two of which belong to class `special`. Using JavaScript, change the background color of the articles that are special.

### 8.2.29. Exercise 29

Go to one of the well known web pages. Open the console and take a look at the HTML objects listed above. Change the title of the page.

### 8.2.30. Exercise 30

Create a page with an image and 2 buttons. Pressing the first button sets the first image, pressing the second button sets the second image.

### 8.2.31. Exercise 31

Simplify [students.btsi.lu/evegi144/WAD/JS/DOM\\_CSS2.html](https://students.btsi.lu/evegi144/WAD/JS/DOM_CSS2.html) by replacing the four functions `moveLeft`, `moveRight`, `moveUp` and `moveDown` with a single function `move`.

### 8.2.32. Exercise 32

Add the possibility for the user to change the number of pixels that the gorilla moves when a button is clicked.

### 8.2.33. Exercise 33

Remove the buttons from the previous example and modify the script so that the gorilla can be moved with the cursor keys.

### 8.2.34. Exercise 34

Create an HTML page with a button. If the user clicks the button, he will be automatically transferred to [wsers.foxi.lu](https://wsers.foxi.lu) after 2 seconds.

### 8.2.35. Exercise 35

Create an empty HTML page. After 1 second the page background color changes to red, then after 4 seconds it changes to green.

### 8.2.36. Exercise 36

Create an HTML page with a button. After 1 second the page background color changes to red, then after 1 second it changes to green, after another second it changes back to red, then back to green etc. When the button is clicked, the background color does not change anymore.

### 8.2.37. Exercise 37

Repeat the previous exercise using `setInterval` instead of `setTimeout`.

### 8.2.38. Exercise 38

Create a document with an image and use `requestAnimationFrame` to implement a simple animation of your choice.

### 8.2.39. Exercise 39

Create a valid HTML5 page that contains an image. If the user clicks on the image, a PHP script is executed that sends additional HTML code, which is inserted after the image in the current document.

## 8.3. Node

### 8.3.1. Exercise 1

Topics: AJAX.

Write a web app that provides asynchronous registration and login.

### 8.3.2. Exercise 2

Topics: JSON.

Write a web app that retrieves large amounts of data from an external API.

### 8.3.3. Exercise 3

Topics: [IIFEs](#).

Create a web page that runs JS without polluting the global namespace.

### 8.3.4. Exercise 4

Topics: Promises.

Write a web app that creates and uses promises.

### 8.3.5. Exercise 5

Topics: Async/await.

Write a web app that creates asynchronous functions and uses them synchronously.

### 8.3.6. Exercise 6

Topics: Installation, configuration and package management.

Install and configure Node and NPM on your own server. Install some packages that you find useful. Submit a document explaining all relevant steps and provide evidence that your installation works correctly.

### 8.3.7. Exercise 7

Topics: Streams.

Write a Node script that generates a very large text file with some random text.

Write a second script that can read a very large text file and display the number of occurrences of a given character in it without consuming much server memory. The file name and the character to count are specified via the command line e.g. `node ex2.js a1.txt a`. The script also displays the maximum amount of memory used.



Delete the large text file to avoid the server backup task being strained unnecessarily.

### 8.3.8. Exercise 8

Topics: Express basics.

Write an Express web server.

## 8.4. PHP

### 8.4.1. Exercise 1

Topics: variables, HTML generation.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex1.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex1.php)

Write a PHP script that stores your name in a variable and displays the text `My name is ` followed by your name followed by a point. Your script may not use more than one echo statement. The generated page needs to be HTML5 validated.

### 8.4.2. Exercise 2

Topics: potentially variables, potentially loops, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex2.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex2.php)

Write a script that calculates and displays the sum of all integers between 1 and the value of a random integer variable `$limit`. Use the `rand` function.

### 8.4.3. Exercise 3

Topics: conditions, loops, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex3.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex3.php)

Write a loop that generates a new random integer from [23, 77] and prints the number on a separate line during each iteration. The loop stops if the number is a multiple of 13.

### 8.4.4. Exercise 4

Topics: conditions, loops, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex4.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex4.php)

Modify exercise 3 so that the loop skips all iterations where the random number is a multiple of 2.

### 8.4.5. Exercise 5

Topics: nested loops, HTML generation.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex5.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex5.php)

Print a matrix of (x, y) coordinates, x from [1, 50] and y from [10, 20]. Your output must be generated programmatically i.e. you may not use one gigantic echo statement! Your solution should be very easy to adapt for different ranges of x and y.

### 8.4.6. Exercise 6

Topics: variables, loops, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex6.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex6.php)

Write a PHP script that:

1. stores 5 random integer values in an array
2. displays the array
3. displays a value randomly selected from the array
4. calculates the sum of the array elements using a standard for loop
5. displays the sum
6. recalculates the sum of the array elements using a foreach loop
7. displays the sum
8. deletes the third array element without creating a hole in the array
9. recalculates and displays the sum of the array elements using a loop of your choice.

### 8.4.7. Exercise 7

Topics: variables, arrays, conditions, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex7.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex7.php)

Write a script that stores the words "sunshine", "rain" and "cloudy" in an array. It then draws a random integer from [0, 2] and displays the text "Today's weather forecast: " followed by the element at the random array position. It then displays one of the following comments, depending on the weather forecast: "Beautiful day ahead!", "Never mind..." or "It could be worse..."

### 8.4.8. Exercise 8

Topics: variables, arrays, conditions, loops, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex8.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex8.php)

Write a PHP script that stores the random grades (from [1, 60]) of 3 students, so that each grade is associated with the corresponding student's name. Display the names with their corresponding

grades. Now increase each grade that is less than 59 by 2 points and if it is 59 by 1 point using `array_keys` and a **standard for loop** and display the names and grades again.

### 8.4.9. Exercise 9

Topics: variables, arrays, conditions, loops, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex9.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex9.php)

Create an associative array that uses the names of a few dogs (at least 3) as the keys and the dogs' ages as values (the age is a random integer from [1, 15]). Display the array, loop through the array and print the name of each dog followed by its age. If the age is 1 it is followed by "year", otherwise by "years".

### 8.4.10. Exercise 10

Topics: variables, arrays, loops, HTML generation.

Write a script that stores the following collection of software titles and displays them nicely:

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex10.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex10.php)

Microsoft	Office	Word Excel Access Powerpoint
	OS	Windows 7 Windows 8 Windows 8.1
Mozilla	Desktop	Firefox Thunderbird
	OS	Firefox OS Top Secret

### 8.4.11. Exercise 11

Topics: variables, arrays, loops, HTML generation, PHP functions.

Modify the [multidimensional arrays](#) example so that it uses only standard `for` loops instead of `foreach`.

### 8.4.12. Exercise 12

Topics: variables, arrays, loops, functions, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex12.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex12.php)

Write a function `sumArray` that returns the sum of all elements (we assume they are numbers) contained in an array passed as parameter. You may not use `array_sum`.

### 8.4.13. Exercise 13

Topics: variables, loops, functions, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex13.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex13.php)

Write a function `sum` that returns the sum of all the numbers passed as parameters. The user can pass as many parameters as they like.

### 8.4.14. Exercise 14

Topics: external content inclusion, HTML generation.

Write a pure PHP script that includes a header file containing a valid HTML doctype and head declaration as well as the opening body tag. Your PHP script then sends some HTML to the browser and then includes a footer file that closes the html tag.

### 8.4.15. Exercise 15

Topics: conditions, forms, HTML generation.

Create a single file web page with a navigation menu offering a selection of different content choices.

### 8.4.16. Exercise 16

Topics: conditions, forms, HTML generation.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex16.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex16.php)

Write a login form that verifies the user name is "T2IF" and the password "PHP". The latter should not be displayed in the form.



Create two versions of your script. Each one uses a different HTTP method to send the form data to the server.

### 8.4.17. Exercise 17

Topics: conditions, functions, forms, external content inclusion, HTML generation.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex17.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex17.php)

Write a script that reads in an email address via a form and validates it, i.e. displays a message telling the user whether the email address is valid or not. Use [Carl Henderson's](#) function as well as the `checkdnsrr` function.

### 8.4.18. Exercise 18

Topics: variables, arrays, conditions, loops, forms, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex18.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex18.php)

Write a grade spreadsheet that allows the user to enter 3 integers. After submission, the script displays the list of all the number triples entered so far and the average for each triple is shown at the end of the line. In addition, the average of all the averages is displayed. You may NOT use sessions. Use form fields instead.

### 8.4.19. Exercise 19

Topics: variables, cookies, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex19.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex19.php)

Write a script that sets a cookie with a value of your choice. The cookie expires after 2 minutes and is only active in the directory `cookies` and its subdirectories. The cookie will only be sent via HTTPS. Write a test script that proves that the cookie will only be active in the `cookies` directory tree (you may want to use the `dirname` or the `header` function).

### 8.4.20. Exercise 20

Topics: variables, conditions, forms, sessions, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex20.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex20.php)

Create a login form for the user name `T2IF1` and the password `WSERS1`. After successful login, the user is taken to the main page, from which he/she can log out. All of the files for this exercise need to be protected, i.e. only the logged in user can make them execute their core functionality.

### 8.4.21. Exercise 21

Topics: variables, arrays, conditions, loops, functions, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex21.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex21.php)

Write the function

```
array reverse(array $array)
```

that can be called with any array and returns the reverse array, e.g. `reverse([1, 2, 3])` returns `[3, 2, 1]`. You may NOT use `array_reverse`!

### 8.4.22. Exercise 22

Topics: variables, arrays, conditions, loops, functions, HTML generation, PHP functions.

Create the function

```
string create_inputs(array $data)
```

that can be called with a 2-dimensional array, like this:

```
echo create_inputs([[ 'User name', 'user_name'], [ 'Password', 'password']]);
```

Here is the result:

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex22.php](http://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex22.php)

More generally, the function can be called with an array which contains any number of arrays, each one consisting of the placeholder and the name to be used to generate the input elements. The function returns the input tags ready to be written to the HTML document (see the HTML source).

### 8.4.23. Exercise 23

Topics: debugging.

Debug the following script:

```

<html lang=en>
<head>
  <meta charset=UTF-8>
  <title>Debug 1</title>
  <form method=prost>
    <input type=tv name=action value=radial checked>Radial gradient<br>
    <input type=tv name=action value=linear>Linear gradient<br>
    Number of stops: <input type=range name=stops min=2 max=10 value=6 step=1><br>
    <button name=submit>Submit</button>
  </form>
<!--php
  fun get_gradiant_JS($style = 'radial', $stop_count = 2) {
    $style !== 'radial' && $style !== 'linear' || $stop_count < 2 || $stop_count
    > 10)
      return '';
      $stops === [];
      $JS    = "<scrip>document.body.style.background = 'repeating-$style-
gradient(";
      for ($i == 0; $i < $stop_count; $i--) {
        $stops = [rand(0, 255), rand(0, 255), rand(0, 255)];
        $JS .= 'rgb($stops[$i][0], $stops[$i][1], $stops[$i][2])';
        if ($i < ($stop_count - 1)) $JS .= ', ';
      }
      echo '<pre>' . print_r($stops) . '</pre>';
      if ($style = 'radial') $idx = 0;
      else $idx = 1;
      return $JS . ") " .
      "fixed'; document.forms[0].querySelectorAll('input')[2].value=$stop_count;" .
      "document.forms[0].querySelectorAll('input')[$idx].checked = true;</scrip>";
    }
    if (!isset($_POST['submit'])) echo get_gradiant_JS($_POST['action'],
    $_POST['stops']);
  !>
</head>
</html>

```

The correct result can be seen at [youtu.be/OH-WwVeoK0U](https://youtu.be/OH-WwVeoK0U).

### 8.4.24. Exercise 24

Topics: variables, arrays, conditions, loops, functions, forms, sessions, HTML generation, PHP functions.

Create an online shop that sells your favorite articles (e.g. cars, computer games, music, . ..). Consider the following:

- Anyone can visit your shop and take a look at your articles.
- Only a logged in user can buy anything, so you need to provide a login and a logout facility.
- For now, there is only one user and you can store the user name and password in a script.

- When a logged in user buys something, his purchase gets stored so that as long as he does not log out or his session has expired, he can always see his past purchases and the total price of all articles bought.
- The session must be secure.
- When a user logs out, the session cookie must be deleted from the user's browser.



Provide login details.

### 8.4.25. Exercise 25

Topics: variables, arrays, loops, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex25.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex25.php)

For your web shop you want to store the details of the logged in user in a global variable named `$loggedin_user`.

This information consists of:

- User name accessible via the key `user_name`.
- Join date accessible via the key `join_date`.
- Number of past purchases accessible via the key `num_past_purchases`.
- Total value of past purchases accessible via the key `val_past_purchases`.

For now you can store fixed values in `$loggedin_user`. Select a reasonable data type for each item.

Create the global variable and display its content in an HTML table exactly as shown.

You must use the function `array_keys` as well as a loop to iterate through the global variable.

### 8.4.26. Exercise 26

Topics: variables, conditions, forms, navigation, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex26.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex26.php)

Create the single file app exactly as shown. This represents a basic template for your shop's navigation.

### 8.4.27. Exercise 27

Topics: variables, conditions, forms, navigation, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex27.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex27.php)

In this exercise you add login, registration and logout functionality to your shop exactly as shown. The hard coded login is `T2IF` with password `PHP`. Explain what is missing for this to be fully functional and suggest possible solutions.

### 8.4.28. Exercise 28

Topics: variables, conditions, forms, functions, navigation, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex28.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex28.php)

Add the function `get_articles` to the previous exercise. This function returns an array containing all the articles of your shop. For now you should randomly generate the articles in your script. Soon we'll retrieve them from our database.

Add the function `get_table` that takes an array of articles as parameter and returns a string containing an HTML table with all articles.

It does not matter what articles your shop offers for sale. What matters is that you think about the information that characterises your articles and that your functions can handle any number of articles.

### 8.4.29. Exercise 29

Topics: variables, arrays, conditions, loops, forms, sessions, functions, HTML generation, PHP functions.

[students.btsi.lu/evegi144/WAD/Exercises/PHP/ex29.php](https://students.btsi.lu/evegi144/WAD/Exercises/PHP/ex29.php)

Create the single file app **exactly** (including styling) as shown. This app allows to add a new user by specifying a name. Furthermore, existing users can be deleted. Note that reloading the page does not delete the user information.

Your script may not store any data in a file or database.

To implement your solution proceed as follows:

1. Each user is stored with a unique id which is determined by your script. The easiest is to start with id 0 and increase it by 1 for every new user created. Create the function `get_next_id` which returns the next id that is available. For instance, if you have already three users with ids 0, 1 and 2, this function will return 3.
2. Create the function `delete_user` which takes a user id as parameter. The function deletes the user with the given id.
3. Create the function `display_users` which displays the users, each with its own delete button.
4. Create the function `display_add_form` which displays the form to add a new user.

## 8.5. MySQL

### 8.5.1. Exercise 1

Topics: Problem modelling and SQL table creation.

Plan and formulate in SQL the structure of a user table for your online shop from PHP exercise 24.

### 8.5.2. Exercise 2

Topics: phpMyAdmin, SQL table creation and data insertion.

Create the user table that you planned in the previous exercise using phpMyAdmin and store a few users in it. Delete the table and recreate it using your SQL script either in PhpStorm or via the MySQL command line.

Provide your SQL script as well as screenshots proving the table and user creation in phpMyAdmin and PhpStorm and/or MySQL.

### 8.5.3. Exercise 3

Topic: Access MySQL from PHP via mysqli.

Enhance your shop by using your new user table for the login, i.e. any user stored in the table can log in.



Provide login details.

### 8.5.4. Exercise 4

Topic: Access MySQL from PHP via prepared statements.

Modify the previous exercise so that it uses prepared statements to prevent SQL injection attacks.



Provide login details.

### 8.5.5. Exercise 5

Topics: Insert and read MySQL data from PHP.

Add a signup functionality to your shop, i.e. a new user can create an account by registering. After a successful registration, the user can log in and shop. Passwords may only be stored in hashed form in the DB. Remember that security is very important. Do your best to hijack your solution and improve it until you are confident that it is secure.

### 8.5.6. Exercise 6

Topics: Problem modelling, SQL table creation, insert and read MySQL data from PHP.

Plan and formulate in SQL the structure of one or more tables that you need for your shop to store all available articles as well as every customer's purchases in the DB so that they are not lost even after session termination. Implement the new structure in your app and let the logged-in user see his full purchase history.

### 8.5.7. Exercise 7

Topics: MySQL-CRUD (create, read, update and delete) operations from PHP.

Add an administrator account to your shop. The admin is the only user who can add, edit and delete the products that your shop is offering. Only products that have not yet been purchased can be deleted. If the admin tries to delete a product that has already been bought an error message will be shown.



Provide login details.

### 8.5.8. Exercise 8

Topics: MySQL-CRUD (create, read, update and delete) operations from PHP using PDO.

Modify your solution of the previous exercise so that all DB access is done via [PHP Data Objects \(PDO\)](#).



Provide login details.

# Chapter 9. Problems

## 9.1. WMOTU Games

Our company, WMOTU (Web Masters Of The Universe), has decided to enter the online computer game sales business. Plan and develop the the web site using most of the material covered in sections [HTML5](#) and [CSS3](#).

## 9.2. WMOTU League

We have been asked to develop a football league information app.

### 9.2.1. WMOTU League Service

Develop a script that displays the rank and number of points for a given football club. For instance, [index.php?club=Leverkusen](#) displays the current rank and number of Leverkusen. For now store the club data in an array.

#### Solution

[students.btsi.lu/evegi144/WAD/WMOTULeagueService/index.php](http://students.btsi.lu/evegi144/WAD/WMOTULeagueService/index.php)

```
<?php
$rankings = array("Leverkusen"    => array(1, 22),
                  "Bayern München" => array(2, 20),
                  "Dortmund"      => array(3, 19));
if (isset($_GET['club'])) {
    $res = $rankings["{$_GET['club']}"];
    echo "Rank: $res[0]  points: $res[1]";
}
?>
```

### 9.2.2. WMOTU League

Develop an app that displays the football results saved in the file [league.csv](#). The result should look like this [screenshot](#). Use the PHP functions [file](#) and [explode](#).

#### Solution

[students.btsi.lu/evegi144/WAD/WMOTULeague/index.php](http://students.btsi.lu/evegi144/WAD/WMOTULeague/index.php)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU League</title>
    <meta charset=UTF-8>
    <link href=style.css rel=stylesheet>
  </head>
  <body>
    <?php
      $contents = @file('league.csv');
      if (!$contents) $array = null;
      else {
        echo
" <table><thead><tr><th>Rank</th><th>Club</th><th>Points</th></tr></thead>";
        foreach ($contents as $line) {
          $line = explode(',', $line);
          echo "<tr><td>{$line[0]}</td><td>{$line[1]}</td><td>{$line[2]}</td></tr>";
        }
        echo "</table>";
      }
    ?>
  </body>
</html>

```

[students.btsi.lu/evegi144/WAD/WMOTULeague/style.css](https://students.btsi.lu/evegi144/WAD/WMOTULeague/style.css)

```

label {
  float:      left;
  clear:      left;
  width:      60px;
  text-align: right;
  padding-right: 10px;
  margin-top: 5px;
  display:    inline-block;
}

input {
  margin-top: 9px;
}

table {
  margin-top: 10px;
  border: 1px solid black;
  border-spacing: 0;
  background-color: green;
}

table th, td {
  border: 4px groove green;
  padding: 4px;
}

table th {
  color: gold;
  text-align: left;
}

table tr:nth-of-type(even) {
  color: lightgreen;
}

thead {
  background-color: darkgreen;
}

```

## 9.3. WMOTU Sorter

Develop an app ([students.btsi.lu/evegi144/WAD/WMOTUSorter](https://students.btsi.lu/evegi144/WAD/WMOTUSorter)) that offers the user a drop down list to choose from the files `testdata1.csv`, `testdata2.csv` and `testdata3.csv` containing a number of records. After form submission the app displays the records in a table in alphabetical order based on the last name. Use the `usort` and `array_splice` functions.

### 9.3.1. Solution

We need to take precautions to prevent someone from hijacking our form and submitting a choice that we have not offered. Consider the following:

[students.btsi.lu/evegi144/WAD/WMOTUSorter/hijacker.php](https://students.btsi.lu/evegi144/WAD/WMOTUSorter/hijacker.php)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU Sorter Hijacker</title>
    <meta charset=UTF-8>
    <link href=style.css rel=stylesheet>
  </head>
  <body>
    <form method=post action=index_vulnerable.php>
      <label for=filelist></label>
      <select id=filelist name=filelist>
        <?php
          $choices = array('hijack1.csv', 'hijack2.csv', 'hijack3.csv');
          foreach ($choices as $choice) {
            echo "<option value=$choice";
            if (isset($_POST['filelist']) && $_POST['filelist'] === $choice)
              echo " selected";
            echo ">$choice</option>";
          }
        ?>
      </select>
      <input type=submit name=submit value="Get addresses sorted by last name">
    </form>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/WMOTUSorter/index\\_vulnerable.php](https://students.btsi.lu/evegi144/WAD/WMOTUSorter/index_vulnerable.php)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU Sorter Vulnerable</title>
    <meta charset=UTF-8>
    <link href=style.css rel=stylesheet>
  </head>
  <body>
    <form method=post>
      <label for=filelist></label>
      <select id=filelist name=filelist>
        <?php // We create the selection dynamically, so that we can change choices
easily.
          $choices = array('testdata1.csv', 'testdata2.csv', 'testdata3.csv');
```

```

        foreach ($choices as $choice) {
            echo "<option value=$choice";
            if (isset($_POST['filelist']) && $_POST['filelist'] === $choice) echo "
selected";
            echo ">$choice</option>";
        }
    ?>
</select>
<input type=submit name=submit value="Get addresses sorted by last name">
</form>
<?php
    if (isset($_POST['submit'])) {
        function compare($x, $y) {
            if ($x[2] === $y[2]) return 0;
            if ($x[2] < $y[2]) return -1;
            return 1;
        }

        $contents = @file("{$_POST['filelist']}");
        echo "Submitted filename: {$_POST['filelist']}";
        if ($contents) {
            foreach ($contents as $line) $array[] = explode('|', $line);
            echo "<table><thead><tr>";
            foreach ($array[0] as $head) echo "<th>$head</th>";
            echo "</tr></thead>";
            array_splice($array, 0, 1);
            if (usort($array, 'compare')) {
                foreach ($array as $line) {
                    echo "<tr>";
                    foreach ($line as $cell) echo "<td>$cell</td>";
                    echo "</tr>";
                }
            }
            echo "</table>";
        }
    }
    ?>
</body>
</html>

```

[students.btsi.lu/evegi144/WAD/WMOTUSorter/index.php](https://students.btsi.lu/evegi144/WAD/WMOTUSorter/index.php)

```

<!DOCTYPE html>
<html lang=en>
    <head>
        <title>WMOTU Sorter</title>
        <meta charset=UTF-8>
        <link href=style.css rel=stylesheet>
    </head>
    <body>

```

```

<form method=post>
  <label for=filelist></label>
  <select id=filelist name=filelist>
    <?php // We create the selection dynamically, so that we can change choices
easily.
        $choices = array('testdata1.csv', 'testdata2.csv', 'testdata3.csv');
        foreach ($choices as $choice) {
            echo "<option value=$choice";
            if (isset($_POST['filelist']) && $_POST['filelist'] === $choice)
                echo " selected";
            echo ">$choice</option>";
        }
    ?>
  </select>
  <input type=submit name=submit value="Get addresses sorted by last name">
</form>
  <?php // Make sure that $_POST['filelist'] is indeed one of the choices we offer
and not a
    // hijacker injection.
    if (isset($_POST['submit']) && in_array($_POST['filelist'], $choices)) {
        function compare($x, $y) {
            if ($x[2] === $y[2]) return 0;
            if ($x[2] < $y[2]) return -1;
            return 1;
        }

        $contents = @file("{$_POST['filelist']}");
        if ($contents) {
            foreach ($contents as $line) $array[] = explode('|', $line);
            echo "<table><thead><tr>";
            foreach ($array[0] as $head) echo "<th>$head</th>";
            echo "</tr></thead>";
            array_splice($array, 0, 1);
            if (usort($array, 'compare')) {
                foreach ($array as $line) {
                    echo "<tr>";
                    foreach ($line as $cell) echo "<td>$cell</td>";
                    echo "</tr>";
                }
            }
            echo "</table>";
        }
    }
  ?>
</body>
</html>

```

## 9.4. WMOTU Sub

Develop the sink the ship game ([students.btsi.lu/evegi144/WAD/WMOTUSub](https://students.btsi.lu/evegi144/WAD/WMOTUSub)). The sea has a size of

20 by 20. There are 10 ships with sizes from 2 to 5.

### 9.4.1. Solution

#### Standard solution

[students.btsi.lu/evégi144/WAD/WMOTUSub/index.php](https://students.btsi.lu/evégi144/WAD/WMOTUSub/index.php)

```
<?php
#####
# Session security
#####
# Only send session id cookie over SSL.
ini_set('session.cookie_secure', true);
# Session IDs may only be passed via cookies, not appended to URL.
ini_set('session.use_only_cookies', true);
ini_set('session.cookie_path', dirname(htmlspecialchars($_SERVER['PHP_SELF'])));
if (!isset($_SERVER['HTTPS'])) // # SSL is not active, activate it.
header('Location: https://' . $_SERVER['HTTP_HOST'] .
dirname(htmlspecialchars($_SERVER['PHP_SELF'])));
if (!isset($_SESSION)) session_start();
# After 30 seconds we'll generate a new session ID to prevent a session
# fixation attack (cf. PHP cookbook p. 338).
if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
    session_regenerate_id();
    $_SESSION['generated'] = time();
}

#####
# Main script
#####
if (isset($_POST['newGame'])) new_game(); # User has clicked 'New game' button.

# User has clicked 'Show ships'/'Hide ships' button
if (isset($_POST['showShips'])) $_SESSION['showShips'] = TRUE;
elseif (isset($_POST['hideShips'])) $_SESSION['showShips'] = FALSE;

# Define constants to increase program readability and flexibility.
define('SUB_BOARD_SIDE', 20);
define('SUB_NUMBER_OF_SHIPS', pow(SUB_BOARD_SIDE, 2) / 40);
define('SUB_HORIZONTAL', 0);
define('SUB_VERTICAL', 1);
define('SUB_EMPTY', 0);
define('SUB_OCCUPIED', 1);
define('SUB_HIT', 2);
define('SUB_MISSED', 3);

# If we already have an existing sea, we retrieve data from the session array.
if (isset($_SESSION['sea']) && isset($_SESSION['ships'])) {
    $sea    = $_SESSION['sea'];
```

```

    $ships = $_SESSION['ships'];
    $maxHits = $_SESSION['maxHits'];
}
else { # Otherwise we create a new one.
    $sea = array();
    $ships = array();
    initialize();
    $_SESSION['sea'] = $sea;
    $_SESSION['ships'] = $ships;
    $_SESSION['maxHits'] = $maxHits;
    $_SESSION['showShips'] = FALSE;
    $_SESSION['shots'] = 0;
    $_SESSION['hits'] = 0;
    $_SESSION['gameOver'] = FALSE;
}

# If a shot been fired, handle it.
if (isset($_POST['x']) && isset($_POST['y']))
    handle_shot($_POST['x'] - 1, $_POST['y'] - 1);

#####
# Function declarations
#####
# This is just a helper function for debugging purposes.
function print_ships() {
    global $ships;
    for ($i = 0; $i < SUB_NUMBER_OF_SHIPS; $i++)
        echo "Ship " . ($i + 1) . ": " . ($ships[$i][0] + 1) . " " . ($ships[$i][1] + 1)
        . " " . $ships[$i][2] . " " . $ships[$i][3] . "<br>";
}

# Create sea table. If the parameter is set to TRUE, ships are shown.
function display_sea($show_ships = FALSE) {
    if ($_SESSION['gameOver'])
        echo '<script>alert("Congratulations, you\'ve sunk all enemies!")</script>';
    global $sea;
    # Create the sea table.
    echo '<table id=sea><thead><tr><th style="width: ' . (100.0 / (SUB_BOARD_SIDE +
1)) .
        '%"></th>';
    for ($i = 1; $i <= SUB_BOARD_SIDE; $i++) {
        echo '<th style="width:' . (100 / (SUB_BOARD_SIDE + 1)) . '%">' . $i . '</th>';
    }
    echo '</tr></thead><tbody>';
    for ($row = 0; $row < SUB_BOARD_SIDE; $row++) {
        echo '<tr><th>' . ($row + 1) . '</th>';
        for ($col = 0; $col < SUB_BOARD_SIDE; $col++) {
            echo '<td style="width:' . (100 / (SUB_BOARD_SIDE + 1)) . '%"';
            if ($sea[$col][$row] === SUB_HIT) echo ' class=hit';
            elseif ($sea[$col][$row] === SUB_MISSED) echo ' class=missed';

```

```

        elseif ($show_ships && ($sea[$col][$row] === SUB_OCCUPIED))
            echo ' class=occupied';
            echo '></td>';
        }
        echo '</tr>';
    }
    echo '</tbody></table>';
    # Display shots fired and hits.
    echo 'Shots fired: ' . $_SESSION['shots'] . ' &nbsp;&nbsp;&nbsp;hits: ' .
$_SESSION['hits'];
}

# Create $sea array filled with 0s.
function initialize_sea() {
    global $sea;
    $sea = array();
    $row = array();
    for ($j = 0; $j < SUB_BOARD_SIDE; $j++) $row[] = SUB_EMPTY;
    for ($i = 0; $i < SUB_BOARD_SIDE; $i++) $sea[] = $row;
}

# Returns TRUE if the area starting at ($x, $y) in direction $orientation $num_cells
tall
# is not occupied. Else FALSE.
function area_available($x, $y, $num_cells, $orientation) {
    global $sea;
    if ($orientation === SUB_HORIZONTAL) {
        for ($i = 0; $i < $num_cells; $i++)
            if (((($x + $i) >= SUB_BOARD_SIDE) || $sea[$x + $i][$y])) return FALSE;
    }
    else {
        for ($i = 0; $i < $num_cells; $i++)
            if (((($y + $i) >= SUB_BOARD_SIDE) || $sea[$x][$y + $i])) return FALSE;
    }
    return TRUE;
}

# Places a ship randomly on the sea, without overlapping another one.
function place_ship($num_cells, $orientation) {
    global $sea;
    do {
        $x = rand(0, SUB_BOARD_SIDE - 1);
        $y = rand(0, SUB_BOARD_SIDE - 1);
    } while (!area_available($x, $y, $num_cells, $orientation));
    if ($orientation === SUB_HORIZONTAL)
        for ($i = 0; $i < $num_cells; $i++) $sea[$x + $i][$y] = 1;
    else
        for ($i = 0; $i < $num_cells; $i++) $sea[$x][$y + $i] = 1;
    return array($x, $y, $num_cells, $orientation);
}

```

```
# Create 10 ships with random size and orientation and place them randomly on the sea.
```

```
function initialize_ships() {  
    global $ships, $maxHits;  
    for ($i = 0; $i < SUB_NUMBER_OF_SHIPS; $i++) {  
        $ship_size      = rand(2, 5);  
        $ship_orientation = rand(SUB_HORIZONTAL, SUB_VERTICAL);  
        $ships[$i]      = place_ship($ship_size, $ship_orientation);  
        $maxHits += $ship_size;  
    }  
}
```

```
function initialize() {  
    initialize_sea();  
    initialize_ships();  
}
```

```
function handle_shot($x, $y) {  
    if ($x < 0 || $x >= SUB_BOARD_SIDE || $y < 0 || $y >= SUB_BOARD_SIDE) return;  
    global $sea, $ships;  
    $hit      = FALSE;  
    $already_hit = FALSE;  
    foreach ($ships as $ship) {  
        if ($ship[3] === SUB_HORIZONTAL) {  
            for ($i = 0; $i < $ship[2]; $i++) { # Need to check whole length of ship.  
                if (($ship[0] + $i) === $x && $ship[1] === $y) {  
                    if ($sea[$x][$y] === SUB_HIT) $already_hit = TRUE;  
                    $sea[$x][$y] = SUB_HIT;  
                    $hit      = TRUE;  
                }  
            }  
        }  
        else {  
            for ($i = 0; $i < $ship[2]; $i++) { # Need to check whole length of ship.  
                if ($ship[0] === $x && ($ship[1] + $i) === $y) {  
                    if ($sea[$x][$y] === SUB_HIT) $already_hit = TRUE;  
                    $sea[$x][$y] = SUB_HIT;  
                    $hit      = TRUE;  
                }  
            }  
        }  
    }  
    if (!$hit) $sea[$x][$y] = SUB_MISSED;  
    $_SESSION['sea']      = $sea;  
    $_SESSION['ships']    = $ships;  
    $_SESSION['shots']++;  
    if ($hit && !$already_hit) {  
        $_SESSION['hits']++;  
        if ($_SESSION['hits'] === $_SESSION['maxHits']) $_SESSION['gameOver'] = TRUE;  
    }  
}
```

```

header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
htmlspecialchars($_SERVER['PHP_SELF']));
}

function new_game() {
    if (!isset($_SESSION)) session_start();
    $_SESSION = array();
    if (session_id() != "" || isset($_COOKIE[session_name()]))
setcookie(session_name(),
    '', time() - 2592000, '/');
    session_destroy();
    header('Location: https://' . $_SERVER['HTTP_HOST'] .
    dirname(htmlspecialchars($_SERVER['PHP_SELF'])));
}

?>

<!DOCTYPE html>
<html lang=en>
<head>
    <title>WMOTU Sub</title>
    <meta charset=UTF-8>
    <style>
        #sea {
            border-collapse: collapse;
            width:          100%;
            table-layout:    fixed;
        }

        #sea, #sea th, #sea td {
            border: 1px solid red;
        }

        .occupied {
            background-color: green;
        }

        .hit {
            background-color: red;
        }

        .missed {
            background-color: black;
        }
    </style>
</head>
<body>
    <?php $_SESSION['showShips'] ? display_sea(TRUE) : display_sea(); ?>
    <form method=post>
        x: <input name=x required>
        y: <input name=y required>

```

```

        <input type=submit value=Fire>
    </form>
    <form method=post>
        <input type=submit id=btn onclick='toggleButton()'
            value=<?php if ($_SESSION['showShips']) echo '"Hide ships"';
            else echo '"Show ships"'; ?>
            name=<?php if ($_SESSION['showShips']) echo 'hideShips';
            else echo 'showShips'; ?>>
    </form>
    <form method=post>
        <input type=submit value='New game' name=newGame>
    </form>
    <script>
        function toggleButton() {
            var b = document.getElementById('btn');
            if (b.value === 'Show ships') {
                b.value = 'Hide ships';
                b.name = hideShips;
            }
            else {
                b.value = 'Show ships';
                b.name = showShips;
            }
        }
    </script>
</body>
</html>

```

## Evolved object oriented solution

Ralph Hermes has developed an improved solution that beautifully illustrates the application of object orientation in PHP. [students.btsi.lu/evegi144/WAD/WMOTUSub/Battleship\\_HerRa036.php](http://students.btsi.lu/evegi144/WAD/WMOTUSub/Battleship_HerRa036.php)

```

<?php

/*
 * Title:  PHP Battleship
 * Author: HERMES Ralph
 * Date:   18.12.2013
 */

// Initialize Local Variables
$BattleTable = new Table();
$isVisible = false;
$score = 0;
$shots = 0;

session_start();
if (!isset($_SESSION["PlaygroundSession"])) {
    $Ships = array(); // Create Array to store the Battleships

```

```

$Deaths = array(); // Create Array to store Dead Ships
$Water = array(); // Create Array to store
for ($i = 0; $i < 10; $i++) {
    $Ships[$i] = new Ship(); // Create Ship
    $Ships[$i]->generate(); // Generate Ship Parameters
    $Ships[$i]->cc(0, 0); // Check Collision for the first time
}
$_SESSION["PlaygroundSession"] = $Ships; // Store ships in session
$_SESSION["PlaygroundDeaths"] = $Deaths; // Store deaths in session
$_SESSION["PlaygroundWater"] = $Water; // Store water shots in session
$_SESSION["PlaygroundShipsVisible"] = false; // Store 'Visible Ships' in session

$_SESSION["Score"] = 0; // Reset score
$_SESSION["Shots"] = 0; // Reset shots
}
else {
    $Ships = $_SESSION["PlaygroundSession"]; // Update ship array in session
    $Deaths = $_SESSION["PlaygroundDeaths"]; // Update deaths array in session
    $Water = $_SESSION["PlaygroundWater"]; // Update water array in session
    $isVisible = $_SESSION["PlaygroundShipsVisible"]; // Update 'Visible Ships' in
session

    $score = $_SESSION["Score"]; // Update score in session
    $shots = $_SESSION["Shots"]; // Update shots in session
}
?>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>BattleShip</title>
<style>
    .playground {
        float: left;
        background-color: #007eff;
    }

    .playground td {
        width: 25px;
        height: 25px;
        background-color: #000097;
        transition: all 0.2s;
    }

    .playground td:hover {
        background-color: #0000d6;
    }

    .playground .title {
        background-color: #20435c;
        text-align: center;
    }

```

```

    }

    .playground .shipfield {
        background-color: #FFF0F0;
    }

    .playground .shipfield:hover {
        background-color: #b3b3b3;
    }

    .playground .shipinput {
        opacity: 0;
        width: 25px;
    }

    .playground .waterinput {
        opacity: 0;
        width: 25px;
    }

    #UserControl {
        float: left;
    }

    #UserControl input {
        width: 120px;
    }

    .playground .destroyed {
        background-color: #FF0000;
    }

    .playground .destroyed:hover {
        background-color: #FF0000;
    }

    .playground .waterfieldhit {
        background-color: #20435c;
    }

    .playground .waterfieldhit:hover {
        background-color: #20435c;
    }

</style>
</head>

<body>
    <div id="wrapper">
        <?php
            $BattleTable->start(); // Start HTML table (Playground)

```

```

    $BattleTable->drawTemplate(); // Draw table
    $BattleTable->end(); // Finish drawing
?>
<div id="UserControl">
    <form method="post">
        <input type="submit" name="newgame" value="New Game"><br>
        <input type="submit" name="showships" value=<?php if ($isVisible)
            echo "'Hide Ships'";
            else echo "'Show Ships'"; ?>>
        </form>
        <label class="stats">Score: <?php echo $score ?></label><br>
        <label class="stats">Shots: <?php echo $shots ?></label><br>
    </div>
</div>
</body>
</html>
<?php
class Table {
    function start() {
        echo "<table class='playground'>";
    }

    function drawTemplate() {
        // Use global Variables
        global $isVisible;
        global $Deaths;
        global $Water;

        // Create Charlists
        $charlistA = array("", "A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K",
"L",
        "M", "N", "O", "P", "Q", "R", "S", "T");
        $charlist1 = array("", "1", "2", "3", "4", "5", "6", "7", "8", "9", "10", "11",
        "12", "13", "14", "15", "16", "17", "18", "19", "20");
        for ($y = 0; $y < 21; $y++) {
            echo "<tr>";
            for ($x = 0; $x < 21; $x++) {
                if ($x != 0) {
                    if ($y == 0) {
                        echo "<td class='title'>" . $charlist1[$x] . "</td>";
                    }
                }
                else {
                    if (!FrameCheck($x, $y)) {
                        $isWater = false;
                        for ($i = 0; $i < count($Water); $i++) {
                            if ($x == $Water[$i][0] and $y == $Water[$i][1]) {
                                $isWater = true;
                            }
                        }
                    }
                    if ($isWater) {

```

```

        echo "
            <td class='waterfieldhit'></td>";
    }
    else {
        // Draw Water Fields
        echo "
            <td class='waterfield'>
                <form method='post'>
                    <input type='hidden'

name='parameterX'

                                value='" . $x . "'>
                    <input type='hidden'

name='parameterY'

                                value='" . $y . "'>
                    <input class='waterinput'

type='submit'

                                name='watershot'>
                </form>
            </td>";
    }
}
else {
    $destroyed = false;
    for ($i = 0; $i < count($Deaths); $i++) {
        if ($x == $Deaths[$i][0] and $y == $Deaths[$i][1]) {
            $destroyed = true;
        }
    }
    if ($destroyed) {
        echo "<td class='destroyed'></td>";
    }
    else {
        // Place a Ship Button at Position: $x , $y
        // Hidden Fields are used to post coordinates
        // Form submits when a Field was triggered
        if ($isVisible) {
            echo "
                <td class='shipfield'>
                    <form method='post'>
                        <input type='hidden'

name='parameterX'

                                    value='" . $x . "'>
                        <input type='hidden'

name='parameterY'

                                    value='" . $y . "'>
                        <input class='shipinput'

type='submit'

                                    name='shipshot'>
                    </form>
                </td>";
        }
    }
}

```

```

        }
        else {
            echo "

name='parameterX'

name='parameterY'

type='submit'

<td class='waterfield'>
    <form method='post'>
        <input type='hidden'

            value='" . $x . "'>
            <input type='hidden'

                value='" . $y . "'>
            <input class='shipinput'

                name='shipshot'>
        </form>
    </td>";
        }
    }
}
}
}
else {
    echo "<td class='title'>" . $charlistA[$y] . "</td>";
}
}
echo "</tr>";
}
}

function end() {
    echo "</table>";
}
}

class Ship {
    public $coor = array(0, 0);
    private $size = 0;
    private $rota = 0;

    function generate() {
        global $Ships;

        $randomSize = rand(2, 5); // Generate Ship Length
        $randomRota = rand(0, 1); // Generate Ship Rotation

        $randomX = rand(1, 20); // Randomize Coordinates
        $randomY = rand(1, 20);

        // Check if random data is valid

```

```

for ($i = 0; $i < $randomSize; $i++) {
    if ($randomRota == 0) {
        for ($j = 0; $j < count($Ships); $j++) {
            if ($Ships[$j]->coor[0] == $randomX + $i + 1) {
                $randomX = rand(1, 20);
                $i = 0;
                $j = 0;
            }
        }
    }
    else if ($randomRota == 1) {
        for ($j = 0; $j < count($Ships); $j++) {
            if ($Ships[$j]->coor[1] == $randomY + $i + 1) {
                $randomY = rand(1, 20);
                $i = 0;
                $j = 0;
            }
        }
    }
}

$this->coor[0] = $randomX;
$this->coor[1] = $randomY;

$this->size = $randomSize;
$this->rota = $randomRota;

}

// Collision Checking
function cc($x, $y) {
    for ($i = 0; $i < $this->size; $i++) {
        if ($this->rota == 0) {
            if ($x == $this->coor[0] + $i) {
                if ($y == $this->coor[1]) {
                    return true;
                }
            }
        }
        else if ($this->rota == 1) {
            if ($x == $this->coor[0]) {
                if ($y == $this->coor[1] + $i) {
                    return true;
                }
            }
        }
    }
    return false;
}
}

```

```

function FrameCheck($x, $y) {
    global $Ships;
    for ($i = 0; $i < 10; $i++) {
        if ($Ships[$i]->cc($x, $y)) {
            for ($j = 0; $j < 10; $j++) {
                return true;
            }
        }
    }
    return false;
}

if (isset($_POST['shipshot'])) {
    $x = $_POST["parameterX"];
    $y = $_POST["parameterY"];

    $Deaths[] = array($x, $y);
    $score++;
    $shots++;
    $_SESSION["PlaygroundDeaths"] = $Deaths;
    $_SESSION["Score"] = $score;
    $_SESSION["Shots"] = $shots;
    echo "<script>window.location.href = '" . $_SERVER['PHP_SELF'] . "'</script>";
}

if (isset($_POST['watershot'])) {
    $x = $_POST["parameterX"];
    $y = $_POST["parameterY"];

    $shots++;
    $_SESSION["Shots"] = $shots;

    $Water[] = array($x, $y);
    $_SESSION["PlaygroundWater"] = $Water;
    echo "<script>window.location.href = '" . $_SERVER['PHP_SELF'] . "'</script>";
}

if (isset($_POST['newgame'])) {
    $Ships = null;
    $Deaths = null;
    $Water = null;
    $Ships = array();
    $Deaths = array();
    $Water = array();
    $shots = 0;
    $score = 0;
    for ($i = 0; $i < 10; $i++) {
        $Ships[$i] = new Ship(); // Create Ship
        $Ships[$i]->generate(); // Generate Ship
        $Ships[$i]->cc(0, 0); // Check Collision
    }
}

```

```

// Reset Session Variables
$_SESSION["PlaygroundSession"] = $Ships;
$_SESSION["PlaygroundDeaths"] = $Deaths;
$_SESSION["PlaygroundWater"] = $Water;
$_SESSION["Score"] = $score;
$_SESSION["Shots"] = $shots;
echo "<script>window.location.href = '" . $_SERVER['PHP_SELF'] . "'</script>";
}

if (isset($_POST['showships'])) {
    $isVisible = !$isVisible;
    $_SESSION["PlaygroundShipsVisible"] = $isVisible;
    echo "<script>window.location.href = '" . $_SERVER['PHP_SELF'] . "'</script>";
}
?>

```

## 9.5. WMOTU Mailer

WMOTU has been asked to develop a web mailing app.

### 9.5.1. WMOTU Mailer v1

Develop a web app ([students.btsi.lu/evegi144/WAD/WMOTUMailerv1](https://students.btsi.lu/evegi144/WAD/WMOTUMailerv1)) to send an email. Sender, recipient, subject and message are provided by the user. The email gets only submitted if all fields are filled in (no data validation).

#### Solution

[students.btsi.lu/evegi144/WAD/WMOTUMailerv1/index.php](https://students.btsi.lu/evegi144/WAD/WMOTUMailerv1/index.php)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTUMailer v1</title>
    <meta charset=UTF-8>
    <style>
      form label {
        display:      inline-block;
        width:        225px;
        font-weight:  bold;
      }
    </style>
  </head>
  <body>
    <form method=post>
      <label>Sender:</label>
      <input type=email name=sender required><br>
      <label>Recipient:</label>
      <input type=email name=recipient required><br>
      <label>Subject:</label>
      <input name=subject required><br>
      <label>Content:</label>
      <textarea name=content required></textarea><br>
      <label></label>
      <input type=submit name=send value=Send>
    </form>
    <?php
      if (isset($_POST['send']))
        if (mail($_POST['recipient'], $_POST['subject'], $_POST['content'], 'From:' .
          $_POST['sender']))
          echo '<script>alert("Email sent successfully!")</script>';
          else echo '<script>alert("Email could not be sent!")</script>';
    ?>
  </body>
</html>

```

We use an HTML form to submit the data. If no **action** attribute is specified, the current script gets called upon submission. Note that all **input** and **textarea** tags have the **required** attribute set. This forces the client browser to submit the form only if all fields have been filled in. We therefore do not need to check again on the server side that all fields have been filled in. We do it here nevertheless for illustrative purposes.

Also note how we can not only generate HTML and CSS, but also JavaScript on the server side.

### 9.5.2. WMOTU Mailer v1++

For aspiring WMOTUs: What is the problem with WMOTUMailer v1? Come up with a solution ([students.btsi.lu/evegi144/WAD/WMOTUMailerv1++](https://students.btsi.lu/evegi144/WAD/WMOTUMailerv1++))!

## Solution

[students.btsi.lu/evegi144/WAD/WMOTUMailerv1++/index.php](https://students.btsi.lu/evegi144/WAD/WMOTUMailerv1++/index.php)

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTUMailer v1++</title>
    <meta charset=UTF-8>
    <style>
      form label {
        display:    inline-block;
        width:      225px;
        font-weight: bold;
      }
    </style>
  </head>
  <body>
    <form method=post action=sendmail.php>
      <label>Sender:</label>
      <input type=email name=sender required><br>
      <label>Recipient:</label>
      <input type=email name=recipient required><br>
      <label>Subject:</label>
      <input name=subject required><br>
      <label>Content:</label>
      <textarea name=content required></textarea><br>
      <label></label>
      <input type=submit name=send value=Send>
    </form>
  </body>
</html>
```

[students.btsi.lu/evegi144/WAD/WMOTUMailerv1++/sendmail.php](https://students.btsi.lu/evegi144/WAD/WMOTUMailerv1++/sendmail.php)

```
<?php
  if (isset($_POST['send'])) {
    if (mail($_POST['recipient'], $_POST['subject'], $_POST['content'], 'From:' .
      $_POST['sender']))
      $s = '<script>alert("Email sent successfully!");';
    else $s = '<script>alert("Email could not be sent!");';
    echo $s . 'window.location = "https://" . $_SERVER['HTTP_HOST'] .
      dirname($_SERVER['PHP_SELF']) . '/index.php"</script>';
  }
?>
```

The problem is the browser refresh function. If the user refreshes the page, another email with the same information than the previous one gets sent immediately. This is no good! The problem originates from our combining the client and server side processing in the same script. This means

that the previous values entered by the user are still stored in the `$_POST` array and the sending of another email is triggered automatically. The improved version splits the client and server side processing into two scripts and triggers the send process only after the user has pressed the `submit` button, not after a page reload. Furthermore, the form is empty, given that the form page gets reloaded after the first email submission.

### 9.5.3. WMOTU Mailer v2

Enhance WMOTUMailer by storing all emails sent in a MySQL database and displaying the current number of emails sent.

#### Solution

`createDB.sql`

[students.btsi.lu/evegi144/WAD/WMOTUMailerv2/createDB.sql](https://students.btsi.lu/evegi144/WAD/WMOTUMailerv2/createDB.sql)

```
# createDB.sql -> create the database table for the WMOTU Mailer v2
DROP TABLE WMOTUMailerv2;

CREATE TABLE WMOTUMailerv2 (id          INT UNSIGNED AUTO_INCREMENT NOT NULL UNIQUE,
                             sender      TEXT                        NOT NULL,
                             recipient    TEXT                        NOT NULL,
                             subject      TEXT                        NOT NULL,
                             message      TEXT                        NOT NULL,

                             PRIMARY KEY (id))
ENGINE = INNODB
DEFAULT CHARSET = utf8;
```

Create the database as described in [\[DBCreation\]](#). ===== `database.php`

```

<?php
    require_once 'db_credentials.php';

    function store_email($sender, $recipient, $subject, $message) {
        $dbc = new mysqli(DB_HOST, DB_USER, DB_PASSWORD, DB_NAME);
        if ($dbc->connect_error) trigger_error('Database connection failed: ' .
            $dbc->connect_error, E_USER_ERROR);
        $dbc->set_charset("utf8");
        $message = strip_tags($message);
        $query = "INSERT INTO " . DB_TABLE . " (sender, recipient, subject, message) " .
            "VALUES (?, ?, ?, ?)";
        $stmt = $dbc->prepare($query);
        if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' . $dbc->error,
            E_USER_ERROR);
        $stmt->bind_param('ssss', $sender, $recipient, $subject, $message);
        $stmt->execute();
        $stmt->close();
    }

    function get_number_of_emails() {
        $dbc = new mysqli(DB_HOST, DB_USER, DB_PASSWORD, DB_NAME);
        if ($dbc->connect_error) trigger_error('Database connection failed: ' .
            $dbc->connect_error, E_USER_ERROR);
        $dbc->set_charset("utf8");
        $query = "SELECT COUNT(*) FROM " . DB_TABLE;
        $res = $dbc->query($query);
        if (!$res) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        return $res->fetch_row();
    }

?>

```

[index.php](#)

[students.btsi.lu/evegi144/WAD/WMOTUMailerv2/index.php](https://students.btsi.lu/evegi144/WAD/WMOTUMailerv2/index.php)

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTUMailer v2</title>
    <meta charset=UTF-8>
    <style>
      form label {
        display:      inline-block;
        width:        225px;
        font-weight:  bold;
      }
    </style>
  </head>
  <body>
    <form method=post action=sendmail.php>
      <label>Sender:</label>
      <input type=email name=sender required><br>
      <label>Recipient:</label>
      <input type=email name=recipient required><br>
      <label>Subject:</label>
      <input name=subject required><br>
      <label>Content:</label>
      <textarea name=content required></textarea><br>
      <label></label>
      <input type=submit name=send value=Send>
    </form>
    Number of emails sent so far:
    <?php
      require_once 'database.php';
      $num = get_number_of_emails();
      echo $num[0];
    ?>
  </body>
</html>

```

sendmail.php

```
<?php
require_once 'database.php';
if (isset($_POST['send'])) {
    if (mail($_POST['recipient'], $_POST['subject'], $_POST['content'], 'From: ' .
        $_POST['sender'])) {
        store_email($_POST['sender'], $_POST['recipient'], $_POST['subject'],
            $_POST['content']);
        $s = '<script>alert("Email sent successfully!");';
    }
    else $s = '<script>alert("Email could not be sent!");';
    echo $s . 'window.location = "https://" . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php"</script>';
}
?>
```

#### db\_credentials.php

This file contains the credentials for accessing the DB. You need to replace these values with your own in order to access your DB.

```
<?php
define('DB_HOST', 'p:localhost');
define('DB_USER', '');
define('DB_PASSWORD', '');
define('DB_NAME', '');
define('DB_TABLE', 'WMOTUMailerv2');
?>
```

### 9.5.4. WMOTU Mailer v3

Further enhance WMOTUMailer by adding sign up and login functionality. Only registered users can send email. The registration includes the user's email address, which is filled in as the sender by default. The user name is displayed.

### 9.5.5. WMOTU Mailer v4

A user can get a listing of all their emails, sorted by date (the newest one at the top). He/she can read and delete each one. Our client also wants a good looking and user-friendly interface.

### 9.5.6. WMOTU Mailer v5

A user can upload an avatar that is displayed next to the user name. Before the email gets sent, an in-depth data validation is performed.

## 9.6. WMOTU Quack

Impressed by our performance on the mailer problem, our customer has tasked us with the

development of a full fledged social network.

### 9.6.1. Security

Create a `protected` folder that is secured as explained in [Security](#).

### 9.6.2. DB

Create a SQL script named `createDB.sql` that creates the table named `WMOTUQuack_users` to store the following fields:

1. `id`: primary key, positive, automatically incremented
2. `first_name`: maximum 40 characters.
3. `last_name`: maximum 40 characters.
4. `email_address`: maximum 255 characters.
5. `user_name`: maximum 32 characters.
6. `password`: maximum 40 characters.
7. `activated`: indicates whether a user is activated, i.e. has clicked on the validation link.
8. `description`: long text.
9. `last_time_seen`: the current time stamp (see [dev.mysql.com/doc/refman/5.7/en/timestamp-initialization.html](http://dev.mysql.com/doc/refman/5.7/en/timestamp-initialization.html)). The data type to be used is `TIMESTAMP`, the default value is `CURRENT_TIMESTAMP`.

If the table already exists it is first deleted and then created. The DB engine is set to `INNODB`. The user name needs to be unique. Only the description can be empty.

Add an instruction to insert test data into the table. Execute the script (remember that this is very easy to do with PhpStorm (cf. [Via PhpStorm](#)) and verify that the database has been created correctly. Where do you save your script? Why?

### 9.6.3. `db_credentials.php`

Create the `db_credentials.php` file (cf. `db_credentials.php`). In this file we call the static `set_credentials` method in the `Database` class, which we'll create next.

### 9.6.4. `database.php`

Create class `Database` in the file `database.php`. For efficiency reasons **all properties and methods of this class are static**, which means that we do not need to create an object of this class. Study [Classes and objects](#) and `database.php`.

#### Constants

Define the constants `$DB_HOST`, `$DB_USER`, `$DB_PASSWORD`, `$DB_NAME`, `$DB_USER_TABLE` and `$DB_LOGGEDIN_USER_TABLE`. These constants are all private to the class. Assign the correct values to `$DB_USER_TABLE` and `$DB_LOGGEDIN_USER_TABLE`. The other constants will get their values from the method `set_credentials` that we'll create in the next step.

### set\_credentials

Create the method `set_credentials`, which sets the values of the `$DB_HOST`, `$DB_USER`, `$DB_PASSWORD` and `$DB_NAME` constants (cf. `database.php`).

### connect

Create the method `connect`, which creates a connection to the DB and returns the connection handle. We set the character set of the connection to `utf8`.

### get\_user\_id

Create the method `get_user_id`, which returns the user id for a given user name, or `FALSE` if such a user does not exist.

### get\_user\_name

Create the method `get_user_name`, which returns the user name for a given user id, or `FALSE` if such an id does not exist.

### get\_user\_data

Create the method `get_user_data`, which returns the user name and description for a given user id as an associative array, or `FALSE` if such an id does not exist.

### get\_users

Create the method `get_users`, which returns an associate array containing all data for all users or `FALSE` if no users exist.

### get\_description

Create the method `get_description`, which returns the description for a given user id, or `FALSE` if such an id does not exist.

### update\_description

Create the method `update_description`, which updates the description for a given user id. The method returns `FALSE` if something went wrong, for instance if such an id does not exist, otherwise `TRUE`.

### is\_logged\_in

Create the method `is_logged_in`, which returns `TRUE` if the user with a given user id has a time stamp that is not older than 2 seconds. Use the `UNIX_TIMESTAMP` function (cf. [http://dev.mysql.com/doc/refman/5.7/en/date-and-time-functions.html#function\\_unix-timestamp](http://dev.mysql.com/doc/refman/5.7/en/date-and-time-functions.html#function_unix-timestamp)) to convert `last_time_seen` and then compare it to the current time in PHP using the `time` function (cf. [www.php.net/manual/en/function.time.php](http://www.php.net/manual/en/function.time.php)). If the time stamp is older than 2 seconds, the method returns `FALSE`. In the main script, we'll install a timer that updates the user's time stamp every 2 seconds. This will allow us to monitor in near real time who is logged in and who is not.

### update\_login\_timestamp

Create the method `update_login_timestamp`, which sets `last_time_seen` to the current time for the user with a given id. Use the MySQL function `NOW` (cf. [dev.mysql.com/doc/refman/5.7/en/date-and-time-functions.html#function\\_now](https://dev.mysql.com/doc/refman/5.7/en/date-and-time-functions.html#function_now)). The method returns `TRUE` if the update succeeded, otherwise `FALSE`.

### register\_login

Create the method `register_login`, which first checks whether a user with a given id is already logged in, in which case `FALSE` is returned. Otherwise, the method updates the user's time stamp and returns the result of this update.

### login

Create the method `login`, which registers the login and returns the user id if a user with a given name and password exists in the user table and is activated, otherwise `FALSE`.

### activate\_user

Create the method `activate_user`, which activates the user with a given id and returns the result of the operation.

### create\_user

Create the method `create_user`, which creates a user with the given first name, last name, email, user name and password. Leading and trailing spaces are removed from all parameters except the password. The password is stored in encrypted form using the `SHA1` function of MySQL (cf. [dev.mysql.com/doc/refman/5.7/en/encryption-functions.html#function\\_sha1](https://dev.mysql.com/doc/refman/5.7/en/encryption-functions.html#function_sha1)). The method returns the user id or `FALSE`.

### delete\_user

Create the method `delete_user`, which deletes the user with the given id. The method returns `TRUE` if the operation succeeded, otherwise `FALSE`.

## 9.6.5. index.php

Develop the login and sign up page. You can use the HTML and CSS from the original. The following features need to be implemented:

1. Session IDs may only be passed via cookies, not appended to URLs.
2. The path for the cookie is set to the current directory in order to prevent it from being available to scripts in other directories.
3. In this particular case, we do NOT use SSL, as the SSL certificate on Foxi is currently not accessible from PHP, which prevents the WebSocket server from being able to use SSL.
4. If no session is currently started, we start one.
5. After 30 seconds we'll generate a new session ID to prevent a session fixation attack (cf. PHP Cookbook p. 338).
6. If a user is already logged in, we let him through to `main.php`.

7. Else, if the user has submitted his login details, we need to check them. If the check fails, we go to `index.php`. If a user with these credentials exists, we set the session data and go to the main page.
8. Else, if the user has submitted the sign up form, we need to make sure that a password has been entered. We also have to check the CAPTCHA and the user name. If the user name already exists, we create an alert informing the user and stop the script. Otherwise, we validate the email address (cf. [Email validation](#)). If the email address is invalid, we go to `index.php`. Otherwise, we create the new user and send a validation email. The subject is `Validation email`, the sender is `do_not_reply@ltam.lu` and the body contains the text `To validate this email address click the following link: wusers.foxi.lu/Tutorial/WMOTUQuack/index.php?s=` followed by the email address followed by `&m=` followed by the hash value followed by `&i=` followed by the user id multiplied by 5. The hash value is the string `T2IF2` followed by the user's email address. This whole string is then encrypted using the `password_hash` function of PHP. After sending the email, we display an alert with the text `A validation email has been sent. Please check your inbox.` and stop the script.
9. Else, if a user has clicked on the link in the validation email that was sent during sign up, we check whether the email address corresponds to the hash value. If that's the case, we activate the user. We retrieve the user name. If it exists, we create a directory `protected/users/` followed by the user name. Then we set the directory permissions so that the owner and group have full rights and others have none. We then go to `index.php`.
10. Else, we generate the CAPTCHA. We use the font `Jumpman` (cf. [www.dafont.com/jumpman.font](http://www.dafont.com/jumpman.font)). The CAPTCHA generation is based on pages 613-614 of "Head First PHP & MySQL". It has however been adapted in order to display a simple calculation that the user has to perform to prove his or her humanness. For this purpose, we generate two random numbers `$num1` and `$num2`, each between 0 and 9. We then create an array `$ops` with the following operators: `+`, `-` and `*` stored as characters. Now we choose a random number `$op` between 0 and 2. The pass phrase will then be `$num1 . $ops[$op] . $num2`. The result of the calculation needs to be stored in the session, which can be done like this:

```
$_SESSION['pass_phrase'] = eval("return intval($pass_phrase);");
```

The approach chosen in the book, to send the image via a header, did not work for me, so I used the `imagepng` function to save the image on disk from where the HTML document retrieves it in the `img` tag.

### 9.6.6. `main.php`

You can use the HTML, CSS and JavaScript from the original. The following features need to be implemented:

1. The bounceer needs to be included to prevent unauthorized usage of our app.
2. We need our DB class.
3. Behind the `Logout` text we display the user name of the logged in user.
4. We need to store the user id of the currently logged in user in a JavaScript variable

5. `userId` and the user name in a variable named `userName`. These two variables are used by `websocket.js`.
6. In the profile section we display the description of the user. Be careful not to display inexistent spaces!

### 9.6.7. `logout.php`

The script performs the following steps:

1. If no session is started we start one.
2. We delete the session array.
3. If a session ID exists we expire the session cookie.
4. We destroy the session.
5. we send the browser to the login page.

### 9.6.8. `updatedescription.php`

The script is executed when the user clicks the `Update description` link and performs the following steps:

1. We include the bouncer.
2. We need the DB class.
3. We update the description with the user id stored in the session and the description submitted by the `updateDescription` function in the `main` object in `main.js`.

### 9.6.9. `deleteprofile.php`

The script is executed when the user clicks the `Delete profile` link and performs the following steps:

1. We include the bouncer.
2. We need the DB class.
3. We delete the user from the DB.
4. We delete the user directory with all sub directories using the following code:

```

$dir_path = "./protected/users/{$_SESSION['user_name']}";
#http://stackoverflow.com/questions/1407338/a-recursive-remove-directory-function-
for-php
try {
    foreach (new RecursiveIteratorIterator(new RecursiveDirectoryIterator($dir_path,
        FilesystemIterator::SKIP_DOTS), RecursiveIteratorIterator::CHILD_FIRST) as
$path) {
        $path->isFile() ? unlink($path->getPathname()) : rmdir($path->getPathname());
    }
    rmdir($dir_path);
} catch (Exception $e) {
}

```

5. We send the browser to the logout script.

### 9.6.10. deleteprofileimage.php

The script is executed when the user clicks the **Delete image** link and performs the following steps:

1. We include the bouncer.
2. We delete the file using the following code:

```

$file_path = '';
$dir_path = "./protected/users/{$_SESSION['user_name']}";
try { # Run through all files in the directory. This primitive approach is good
for now.
    foreach (new RecursiveIteratorIterator(new
RecursiveDirectoryIterator($dir_path,
        FilesystemIterator::SKIP_DOTS), RecursiveIteratorIterator::CHILD_FIRST) as
$path) {
        if ($path->isFile()) $file_path = $path->getPathname();
    }
} catch (Exception $e) {
}
if ($file_path !== '') unlink($file_path);

```

### 9.6.11. getprofileimage.php

The script is executed when the user clicks the **Delete image** link and when the **Members** button is activated. It performs the following steps:

1. We include the bouncer.
2. We declare a variable `$file_path` and initialize it with an empty string.
3. If the user name has been transmitted via **GET** we set `$dir_path` to `/protected/users/{$_GET['user_name']}`, else we set `$dir_path` to `/protected/users/{$_SESSION['user_name']}`.

4. We run through all the files in `$dir_path` and save the path of the last one in `$file_path`.
5. We tell the browser that we will now be sending a png image.
6. If the file path is not empty we read the file.

### 9.6.12. `getmember.php`

The script is executed when the `Profile` button is activated. It performs the following steps:

1. We include the bouncer.
2. We need the DB class.
3. We declare the variable `$output` and initialize it with an empty string.
4. If the user id has been submitted via `POST`, we get the user data. If this data exists, we
  - a. build a string with a paragraph containing the user name,
  - b. followed by an image with the profile image,
  - c. followed by

```
draggable=false onmousedown="event.preventDefault(); alt="",
```

- d. followed by a new line,
  - e. followed by a new paragraph with as preformatted content the user's description.
5. We send the string to the browser.

### 9.6.13. Solution

WMOTU has been asked to develop a full-fledged communication platform. [students.btsi.lu/evegi144/WAD/WMOTUQuack](https://students.btsi.lu/evegi144/WAD/WMOTUQuack)

#### DB

#### Structure



```

CURRENT_TIMESTAMP NOT
NULL,
    FOREIGN KEY (user_id) REFERENCES WMOTUQuack_users (id)
        ON DELETE CASCADE
        ON UPDATE CASCADE)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;*/

INSERT INTO WMOTUQuack_users (first_name, last_name, email_address, user_name,
password,
                                activated)
VALUES ("Dum1", "Dum", "dummy1@dumdum.com", "dummy1", SHA("d1pw"), TRUE),
    ("Dum2", "Dum", "dummy2@dumdum.com", "dummy2", SHA("d2pw"), TRUE);
INSERT INTO WMOTUQuack_messages (user_id, message) VALUES
    ((SELECT
        id
        FROM WMOTUQuack_users
        WHERE user_name = "dummy1"), "Test message 1"),
    ((SELECT
        id
        FROM WMOTUQuack_users
        WHERE user_name = "dummy2"), "Test message 2");

CREATE TABLE WMOTUQuack_messages (id          INT UNSIGNED AUTO_INCREMENT NOT NULL
UNIQUE,
                                user_id      INT UNSIGNED                NOT NULL,
                                message       TEXT                      NOT NULL,
                                time_stamp   TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON
UPDATE
                                CURRENT_TIMESTAMP                NOT NULL,
                                PRIMARY KEY (id),
                                FOREIGN KEY (user_id) REFERENCES WMOTUQuack_users (id)
                                    ON DELETE CASCADE
                                    ON UPDATE CASCADE)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;

```

database.php

```

<?php
    require_once 'db_credentials.php';

    class Database {
        private static $DB_HOST;
        private static $DB_USER;
        private static $DB_PASSWORD;
        private static $DB_NAME;
    }

```

```

private static $DB_USER_TABLE = 'WMOTUQuack_users';
#private static $DB_LOGGEDIN_USER_TABLE = 'WMOTUQuack_loggedin_users';
private static $DB_MESSAGE_TABLE = 'WMOTUQuack_messages';

static function set_credentials($db_host, $db_user, $db_password, $db_name) {
    self::$DB_HOST      = $db_host;
    self::$DB_USER      = $db_user;
    self::$DB_PASSWORD  = $db_password;
    self::$DB_NAME      = $db_name;
}

static private function connect() {
    $dbc = new mysqli(self::$DB_HOST, self::$DB_USER, self::$DB_PASSWORD,
        self::$DB_NAME);
    if ($dbc->connect_error) trigger_error('Database connection failed: ' .
        $dbc->connect_error, E_USER_ERROR);
    $dbc->set_charset("utf8");
    return $dbc;
}

# Returns a string.
/*static private function sanitize_string($dbc, $string) {
    $string = strip_tags(trim($string));
    if (get_magic_quotes_gpc()) $string = stripslashes($string);
    $result = mysqli_real_escape_string($dbc, $string);
    return $result;
}*/

# Returns id if a user with the given name exists, otherwise FALSE.
static function get_user_id($user_name) {
    $dbc = self::connect();
    # Look up user id.
    $query = 'SELECT id FROM ' . self::$DB_USER_TABLE . ' WHERE user_name = ?';
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    $stmt->bind_param('s', $user_name);
    $stmt->execute();
    $stmt->store_result();
    $result = $stmt->num_rows === 1;
    if ($result) {
        $stmt->bind_result($user_id);
        $result = $stmt->fetch();
    }
    $dbc->close();
    if ($result) return $user_id;
    else return $result;
}

# Returns user name for given id if id exists, otherwise FALSE.
static function get_user_name($user_id) {

```

```

$dbc = self::connect();
$query = 'SELECT user_name FROM ' . self::$DB_USER_TABLE . ' WHERE id = ?';
$stmt = $dbc->prepare($query);
if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
    $dbc->error, E_USER_ERROR);
$stmt->bind_param('s', $user_id);
$stmt->execute();
$stmt->bind_result($user_name);
$result = $stmt->fetch();
$dbc->close();
if ($result) return $user_name;
else return $result;
}

# Returns user name and description for given id if id exists, otherwise FALSE.
static function get_user_data($user_id) {
    $dbc = self::connect();
    $query = 'SELECT user_name, description FROM ' . self::$DB_USER_TABLE .
        ' WHERE id = ?';
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    $stmt->bind_param('s', $user_id);
    $stmt->execute();
    $stmt->bind_result($user_name, $description);
    $result = $stmt->fetch();
    $dbc->close();
    if ($result) return array('user_name' => $user_name,
        'description' => $description);
    else return $result;
}

# Returns associative array or FALSE.
static function get_users() {
    $dbc = self::connect();
    $result = FALSE;
    $query = 'SELECT * FROM ' . self::$DB_USER_TABLE;
    $res = $dbc->query($query);
    if (!$res) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    while ($dat = $res->fetch_array(MYSQLI_ASSOC)) $result[] = $dat;
    $dbc->close();
    return $result;
}

# Returns associative array or FALSE.
/*static function get_logged_in_user_ids() {
    $dbc = self::connect();
    $result = FALSE;
    $query = 'SELECT id FROM ' . self::$DB_USER_TABLE;
    $res = $dbc->query($query);

```

```

        if (!$res) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        while ($dat = $res->fetch_array(MYSQLI_ASSOC)) $result[] = $dat;
        return $result;
    }*/

# Returns description for given id if id exists, otherwise FALSE.
static function get_description($user_id) {
    $dbc = self::connect();
    $query = 'SELECT description FROM ' . self::$DB_USER_TABLE . ' WHERE id = ?';
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    $stmt->bind_param('s', $user_id);
    $stmt->execute();
    $stmt->bind_result($description);
    $result = $stmt->fetch();
    $dbc->close();
    if ($result) return $description;
    else return $result;
}

# Returns TRUE if description update succeeded, otherwise FALSE.
static function update_description($user_id, $description) {
    $dbc = self::connect();
    $query = 'UPDATE ' . self::$DB_USER_TABLE . ' SET description = ? WHERE id = ?';
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    $stmt->bind_param('ss', $description, $user_id);
    $result = $stmt->execute();
    $dbc->close();
    return $result;
}

# Returns FALSE if user could not be created, otherwise user id.
static function create_user($first_name, $last_name, $email, $user_name,
$password) {
    $dbc = self::connect();
    $query = 'INSERT INTO ' . self::$DB_USER_TABLE .
        ' (first_name, last_name, email_address, user_name, password, activated)' .
        ' VALUES (?, ?, ?, ?, SHA1(?), FALSE)';
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    $first_name = trim($first_name);
    $last_name = trim($last_name);
    $email = trim($email);
    $user_name = trim($user_name);
    $stmt->bind_param('sssss', $first_name, $last_name, $email, $user_name,
        $password);

```

```

        if ($stmt->execute()) $result = self::get_user_id($user_name);
        else $result = FALSE;
        $dbc->close();
        return $result;
    }

    # Returns FALSE if user could not be deleted, otherwise TRUE.
    static function delete_user($user_id) {
        $dbc = self::connect();
        $query = 'DELETE FROM ' . self::$DB_USER_TABLE . ' WHERE id = ?';
        $stmt = $dbc->prepare($query);
        if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        $stmt->bind_param('s', $user_id);
        $result = $stmt->execute();
        $dbc->close();
        return $result;
    }

    # Returns user_id of the user or FALSE.
    static function login($user_name, $password) {
        $dbc = self::connect();
        $query = 'SELECT id FROM ' . self::$DB_USER_TABLE .
            ' WHERE user_name = ? AND password = SHA(?) AND activated = TRUE';
        $stmt = $dbc->prepare($query);
        if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        $stmt->bind_param('ss', $user_name, $password);
        $stmt->execute();
        $stmt->store_result();
        if ($stmt->num_rows === 1) {
            $stmt->bind_result($result);
            $stmt->fetch();
        }
        else $result = FALSE;
        $dbc->close();
        if ($result && self::register_login($result)) return $result;
        else return FALSE;
    }

    # Called by login. Returns FALSE if user is already logged in.
    private static function register_login($user_id) {
        if (self::is_logged_in($user_id)) return FALSE;
        return self::update_login_timestamp($user_id);
    }

    # Returns TRUE if description update succeeded, otherwise FALSE.
    static function update_login_timestamp($user_id) {
        $dbc = self::connect();
        $query = 'UPDATE ' . self::$DB_USER_TABLE . ' SET last_time_seen = ' .
            'NOW() WHERE id = ?';

```

```

$stmt = $dbc->prepare($query);
if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
    $dbc->error, E_USER_ERROR);
$stmt->bind_param('s', $user_id);
$result = $stmt->execute();
$dbc->close();
return $result;
}

/*static function logout($user_id) {
    $dbc = self::connect();
    $query = 'DELETE FROM ' . self::$DB_LOGGEDIN_USER_TABLE .
        " WHERE user_id = ?";
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    $stmt->bind_param('s', $user_id);
    $result = $stmt->execute();
    $stmt->close();
    return $result;
}*/

# Returns TRUE if user was seen within the last 2 seconds, otherwise FALSE.
static function is_logged_in($user_id) {
    $dbc = self::connect();
    $query = 'SELECT UNIX_TIMESTAMP(last_time_seen) FROM ' .
        self::$DB_USER_TABLE . ' WHERE id=?';
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    $stmt->bind_param('s', $user_id);
    $stmt->execute();
    $stmt->store_result();
    if ($stmt->num_rows !== 1) return FALSE;
    $stmt->bind_result($timestamp);
    $result = $stmt->fetch();
    $dbc->close();
    if ((time() - $timestamp) <= 2) return TRUE;
    return FALSE;
}

# Search by user id.
# Returns associative array or FALSE.
static function get_messages($user_id = FALSE) {
    $dbc = self::connect();
    $result = FALSE;
    $query = 'SELECT * FROM ' . self::$DB_MESSAGE_TABLE . " WHERE
user_id=$user_id";
    $res = $dbc->query($query);
    if (!$res) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);

```

```

        while ($dat = $res->fetch_array(MYSQLI_ASSOC)) $result[] = $dat;
        $dbc->close();
        return $result;
    }

    # Inserts message for user id stored in session.
    static function insert_message($message) {
        $dbc = self::connect();
        $query = 'INSERT INTO ' . self::$DB_MESSAGE_TABLE . ' (user_id,
        message) VALUES (?, ?)';
        $stmt = $dbc->prepare($query);
        if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' . $dbc->error,
        E_USER_ERROR);
        $stmt->bind_param('ss', $_SESSION['user_id'], $message);
        $result = $stmt->execute();
        $dbc->close();
        return $result;
    }

    # Returns TRUE if deletion succeeded, otherwise FALSE.
    static function delete_message($id) {
        $dbc = self::connect();
        $result = mysqli_query($dbc, 'DELETE FROM ' . self::$DB_MESSAGE_TABLE .
        " WHERE id=$id");
        mysqli_close($dbc);
        return $result;
    }

    # Activate user with given id.
    static function activate_user($user_id) {
        $dbc = self::connect();
        $query = 'UPDATE ' . self::$DB_USER_TABLE . ' SET activated = TRUE WHERE id =
?';
        $stmt = $dbc->prepare($query);
        if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' . $dbc->error,
        E_USER_ERROR);
        $stmt->bind_param('s', $user_id);
        $result = $stmt->execute();
        $dbc->close();
        return $result;
    }
}

?>

```

index.php

```

<?php
// Only send session id cookie over SSL.
//ini_set('session.cookie_secure', true);

```

```

// Session IDs may only be passed via cookies, not appended to URL.
ini_set('session.use_only_cookies', true);
// Set the path for the cookie to the current directory in order to prevent it from
// being available to scripts in other directories.
ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
/*if ($_SERVER['SERVER_PORT'] != 443)
    header('Location: https://' . $_SERVER['SERVER_NAME'] .
$_SERVER['SCRIPT_NAME']);*/
/*if (!isset($_SERVER['HTTPS'])) // If SSL is not active, activate it.
    header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
        dirname(htmlspecialchars($_SERVER['PHP_SELF'])));*/
// If no session is started yet, we'll start one.
if (!isset($_SESSION)) session_start();
// After 30 seconds we'll generate a new session ID to prevent a session fixation
// attack (cf. PHP cookbook p. 338).
if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
    session_regenerate_id();
    $_SESSION['generated'] = time();
}
// Include the database class needed to access the database.
require_once 'protected/database.php';
$file_name = 'pp.png'; # CAPTCHA image
// If a user is already logged in, let him through to the main page.
if (isset($_SESSION['user_id']))
    header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/main.php');
// Else, if the user has submitted his login details, we need to check them.
elseif (isset($_POST['logIn'])) {
    if (isset($_POST['username']) && isset($_POST['password1'])) {
        $result = Database::login($_POST['username'], $_POST['password1']);
        // If a user with this login exists, we load the main page.
        if ($result) {
            $_SESSION['user_id'] = $result;
            $_SESSION['user_name'] = $_POST['username'];
            header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] .
                dirname($_SERVER['PHP_SELF']) . '/main.php');
        }
    }
} elseif (isset($_GET['m'])) { # Activate user
    if (md5('T2IF2' . $_GET['s']) === $_GET['m']) {
        if (Database::activate_user(intval($_GET['i'] / 5))) {
            $user_name = Database::get_user_name(intval($_GET['i'] / 5));
            if ($user_name) {
                if (!mkdir('protected/users/' . $user_name, 0770))
                    die("User dir creation failed!");
                chmod('protected/users/' . $user_name, 0770);
            }
        }
    }
    header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php');
}

```

```

}
// Else, if the user has signed up for a new account, we need to check if
// such a user already exists.
elseif (isset($_POST['signUp'])) {
    if (isset($_SESSION['pass_phrase']) && isset($_POST['captcha']) &&
        $_SESSION['pass_phrase'] === intval($_POST['captcha']))
    ) {
        if (isset($_POST['username']) && isset($_POST['password1'])) {
            if (Database::get_user_id($_POST['username'])) {
                echo "<script>window.alert('User exists already!');</script>";
                exit;
            } else { // If not, we'll send an opt-in email
                require_once "protected/rfc822.php";
                if (is_valid_email_address($_POST['email']) &&
                    checkdnsrr(preg_replace('/^[a-zA-Z0-9][a-zA-Z0-9\._\-\&!?=#]*@/ ',
                        '', $_POST['email'])))
                ) {
                    if ($result = Database::create_user
                        ($_POST['firstname'], $_POST['lastname'], $_POST['email'],
                            $_POST['username'], $_POST['password1']))
                    ) {
                        $hash = md5('T2IF2' . $_POST['email']);
                        $mail_body = 'To validate this email address click the ' .
                            'following link: '
                            . 'http://wsers.foxi.lu/WAD/WMOTUQuack/index.php?s=' .
                                $_POST['email'] . '&m=' . $hash . '&i=' . (5 * $result);
                        if (mail($_POST['email'], 'Validation email', $mail_body,
                            'From: webmaster@example.com'))
                        echo "<script>window.alert('A validation email has been sent. " .
                            "Please check your inbox.');"</script>";
                        else
                            echo "<script>window.alert('A validation email could not be
sent!');</script>";
                        exit;
                    }
                } else header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] .
                    dirname($_SERVER['PHP_SELF']) . '/index.php');
            }
        }
    } else header('Location: ' . 'http://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php');
} else {
    define('CAPTCHA_WIDTH', 40);
    define('CAPTCHA_HEIGHT', 22);
    $num1 = rand(0, 9);
    $num2 = rand(0, 9);
    $ops = array('+', '-', '*');
    $op = rand(0, 2);
    $pass_phrase = $num1 . $ops[$op] . $num2;
    $_SESSION['pass_phrase'] = eval("return intval($pass_phrase);");
    $img = imagecreatetruecolor(CAPTCHA_WIDTH, CAPTCHA_HEIGHT);

```

```

    imagefilledrectangle($img, 0, 0, CAPTCHA_WIDTH, CAPTCHA_HEIGHT,
        imagecolorallocate($img, 202, 168, 13));
    for ($i = 0; $i < 5; $i++)
        imageline($img, 0, rand() % CAPTCHA_HEIGHT, CAPTCHA_WIDTH,
            rand() % CAPTCHA_HEIGHT, imagecolorallocate($img, 128, 128,
                64));
    for ($i = 0; $i < 50; $i++) imagesetpixel($img, rand() % CAPTCHA_WIDTH,
        rand() % CAPTCHA_HEIGHT, imagecolorallocate($img, 64, 64, 64));
    imagettfttext($img, 18, 0, 5, CAPTCHA_HEIGHT - 5,
        imagecolorallocate($img, 0, 0, 0), './Jumpman.ttf',
        $pass_phrase);
    unlink($file_name);
    imagepng($img, $file_name);
}
?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>WMOTU Quack</title>
        <meta charset=UTF-8>
        <link href=index.css rel=stylesheet>
        <script src=index.js></script>
    </head>
    <body>
        <header>
            <img id=logo src=logo.png alt=Logo width=64 height=64>
        </header>
        <main>
            <form method=post id=logInForm>
                <label>User name:</label>
                <input type=text name=username required autofocus>
                <label>Password:</label>
                <input type=password name=password1 required>
                <input type=submit name=logIn value='Log in'><br>
                <input type=button onclick='index.signUp();' value='Sign up'>
            </form>
            <form method=post id=signUpForm onSubmit='return index.validateSignUpForm();'>
                <label>First name:</label>
                <input type=text name=firstname required>
                <label>Last name:</label>
                <input type=text name=lastname required>
                <label>Email:</label>
                <input type=email name=email required>
                <label>Username:</label>
                <input type=text name=username id=username required
                    onBlur=index.checkUser(this);>
                <span id=info></span>
                <label>Password:</label>
                <input type=password id=pw1 name=password1 required>
                <label>Retype pw:</label>
                <input type=password id=pw2 name=password2 required>
            </form>
        </main>
    </body>
</html>

```

```

<label>CAPTCHA:</label>
<img src='<?php echo $file_name; ?>' alt='Pass phrase'>
<input type=number min=-9 max=81 name=captcha required>
<input type=submit name=signUp value='Sign up'><br>
<input type=button onclick='index.logIn();' value='Log in'>
</form>
</main>
</body>
</html>

```

## index.js

```

"use strict";

var index = {
  init: function () {
    document.getElementById('signUpForm').style.display = 'none';
    var tmp = document.documentMode, e, isIE; //
http://www.pinlady.net/PluginDetect/IE
    // Try to force this property to be a string.
    try {
      document.documentMode = "";
    }
    catch (e) {
    }

    // If document.documentMode is a number, then it is a read-only property, and
so
    // we have IE 8+.
    // Otherwise, if conditional compilation works, then we have IE < 11.
    // Otherwise, we have a non-IE browser.
    isIE = typeof document.documentMode == "number" || eval("/ *@cc_on!@* /!1");

    // Switch back the value to be unobtrusive for non-IE browsers.
    try {
      document.documentMode = tmp;
    }
    catch (e) {
    }
    if (isIE)
      window.alert("This site requires at least version 11 of " +
        "Internet Explorer. And even then drag and drop does " +
        "not work.\nFirefox or Chrome work best.");
  },

  signUp: function () {
    document.getElementById('logInForm').style.display = 'none';
    document.getElementById('signUpForm').style.display = 'block';
    document.getElementById('signUpForm')[0].focus();
  },

```

```

logIn: function () {
    document.getElementById('logInForm').style.display = 'block';
    document.getElementById('signUpForm').style.display = 'none';
    document.getElementById('logInForm')[0].focus();
},

checkUser: function (userInput) {
    if (userInput.value != '') {
        var myRegExp = /^[0-9a-zA-z_]*$/;
        if (!myRegExp.test(userInput.value)) {
            window.alert('User name may only contain digits, letters and _!');
            return;
        }
        var data = new FormData();
        data.append('user_to_check', userInput.value);
        var URL = 'checkuser.php';
        var request = new XMLHttpRequest();
        request.addEventListener('load', function (e) {
            document.getElementById('info').innerHTML = e.target.responseText;
        });
        request.open('POST', URL, true);
        request.send(data);
    }
},

validateSignUpForm: function () {
    var pw1 = document.getElementById('pw1').value;
    var pw2 = document.getElementById('pw2').value;
    if (pw1 !== pw2) {
        window.alert("Passwords don't match!");
        return false;
    }
    if (document.getElementById('available') === null) {
        window.alert('This user name is not available. ');
        document.getElementById('username').focus();
    }
    return document.getElementById('available') !== null;
}
};

window.addEventListener('load', index.init);

```

## index.css

```

body {
    background: linear-gradient(to bottom right, yellow, #772222) fixed;
    text-shadow: 1px 1px 1px white;
    margin: 0;
}

```

```
main {
  position: absolute;
  top:      64px;
  left:     0;
  right:    0;
  bottom:   0;
}

#logo {
  transition:          transform 1s;
  -webkit-transition: -webkit-transform 1s;
}

#logo:hover {
  transform:          rotate(45deg);
  -webkit-transform: rotate(45deg); /* Safari, Chrome, mobile Safari, and Android */
}

form {
  width:          280px;
  margin-left:    auto;
  margin-right:   auto;
}

form > label {
  float:          left;
  width:          80px;
  text-align:     right;
  padding-right:  10px;
  margin-top:     10px;
}

form > input {
  margin-top:     10px;
  text-shadow:    1px 1px 1px white;
  border-radius:  5px;
}

form > input[type=text], form > input[type=password], form > input[type=email],
form > input[type=number] {
  opacity: 0.5;
}

form > input:focus {
  background-color: yellow;
}

form > input[type=submit], form > input[type=button] {
  background: linear-gradient(to bottom right, yellow, red);
}
```

```

margin-left: 90px;
width:      140px;
}

form > input[type=submit]:focus, form > input[type=button]:focus {
  border: 2px solid grey;
}

form > input::-moz-focus-inner {
  border: 0;
}

form > img {
  float:      left;
  margin-top: 10px;
}

form > input[name=captcha] {
  width:      84px;
  margin-left: 10px;
}

```

## main.php

```

<?php
  require_once 'protected/bouncer.php';
  require_once 'protected/database.php';
?>
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU Quack</title>
    <meta charset=UTF-8>
    <link href=main.css rel=stylesheet>
    <script src=main.js></script>
    <script src=websocket.js></script>
  </head>
  <body>
    <header>
      <img id=logo src=logo.png alt=Logo width=64 height=64>
      <nav>
        <ul id=navList>
          <li><a id=quackButton onclick="main.toggle('quack');">Quack</a></li>
          <li><a id=memberButton onclick="main.toggle('member');">Members</a></li>
          <li><a id=profileButton onclick="main.toggle('profile');">Profile</a></li>
          <!--<li><a id=settingsButton onclick="">Settings</a></li>-->
          <li><a onclick="window.location='logout.php'">Logout
            <?php
              if (isset($_SESSION['user_name']))
                echo $_SESSION['user_name'];
            </?php
          </li>
        </ul>
      </nav>
    </header>
  </body>
</html>

```

```

        ?>
      </a></li>
    </ul>
  </nav>
</header>
<main>
  <section id=quackSection>
    <?php
      if (isset($_SESSION['user_id']))
        echo "<script>var userId = {$_SESSION['user_id']};</script>";
      if (isset($_SESSION['user_name']))
        echo "<script>var userName = '{$_SESSION['user_name']}'</script>";
    ?>
    <span class="drag" draggable=true></span>
    Message:<br>
    <textarea id=sendText cols=50 rows=5></textarea><br>
    <textarea id=receiveText cols=50 rows=10 readonly></textarea>
  </section>
  <section id=memberSection>
    <span class="drag" draggable=true></span>
    <article id=memberListArticle>
      <ul id=memberList></ul>
    </article>
    <article id=memberSelectArticle></article>
  </section>
  <section id=profileSection>
    <span class="drag" draggable=true></span>
    <div id=filebox></div>
    <textarea id=profileText cols=50 rows=10 spellcheck=true><?php
      if (isset($_SESSION['user_id']))
        echo Database::get_description($_SESSION['user_id']);
    ?></textarea><br>
    <a id=updateDescriptionButton onclick=main.updateDescription();>
      Update description</a>
    <a id=deleteProfileButton onclick=main.deleteProfile();>Delete profile</a>
    <a id=deleteImageButton onclick=main.deleteImage();>Delete image</a>
  </section>
</main>
</body>
</html>

```

### main.js

```

"use strict";
var main = {
  filebox: undefined,

  initiate: function () {
    var elems = document.getElementsByClassName('drag');
    for (var i = 0; i < elems.length; i++)

```

```

        elems[i].addEventListener('dragstart', main.dragStart.bind(this), false);
document.body.addEventListener('dragover', main.dragOver.bind(this), false);
document.body.addEventListener('drop', main.drop.bind(this), false);
this.filebox = document.getElementById('filebox');
this.filebox.addEventListener('dragenter', function (e) {
    e.preventDefault();
});
this.filebox.addEventListener('dragover', function (e) {
    e.preventDefault();
});
this.filebox.addEventListener('drop', main.dropped.bind(this));
elems = document.getElementsByTagName('section');
for (i = 0; i < elems.length; i++)
    elems[i].addEventListener('click', main.manageZIndex.bind(this), false);
main.startMemberListUpdateTimer();
window.addEventListener('beforeunload', main.unload);
},

unload: function () {
    window.location = 'logout.php';
},

updateDescription: function () {
    var data = new FormData();
    data.append('description', document.getElementById('profileText').value);
    var URL = 'updatedescription.php';
    var request = new XMLHttpRequest();
    request.open('POST', URL, true);
    request.send(data);
},

deleteProfile: function () {
    if (confirm('Do you really want to delete your profile?'))
        window.location = 'deleteprofile.php';
},

deleteImage: function () {
    var request = new XMLHttpRequest();
    request.open('POST', 'deleteprofileimage.php', true);
    request.send();
    var image = document.getElementById('profileImage');
    if (image)
        image.src = 'getprofileimage.php?' + Math.random(); // Force script
reload.
},

viewMemberProfile: function (id) {
    var data = new FormData();
    data.append('user_id', id);
    var URL = 'getmember.php';
    var request = new XMLHttpRequest();

```

```

        request.addEventListener('load', function (e) {
            var dat = e.target;
            if (dat.status === 200) {
                var article = document.getElementById('memberSelectArticle');
                article.innerHTML = dat.responseText;
            }
        });
        request.open('POST', URL, true);
        request.send(data);
    },

    toggle: function (name) {
        var section = document.getElementById(name + 'Section');
        var button = document.getElementById(name + 'Button');
        this.zeroZIndex();
        if (!section.hasAttribute('active')) {
            var request = new XMLHttpRequest();
            request.addEventListener('load', function (e) {
                var data = e.target;
                if (data.status === 200 && data.responseText === 'false')
                    window.location = 'logout.php';
            });
            request.open('POST', 'stillloggedin.php', true);
            request.send();
            section.setAttribute('active', '');
            button.setAttribute('active', '');
            if (name === 'profile') {
                var image = document.getElementById('profileImage');
                if (image)
                    image.src = 'getprofileimage.php?' + Math.random(); // Force
script reload.
                else {
                    image = document.createElement('img');
                    image.id = 'profileImage';
                    image.alt = 'Drop your image here.';
                    image.src = 'getprofileimage.php?' + Math.random(); // Force
script reload.
                    image.setAttribute('draggable', 'false');
                    image.setAttribute('onmousedown', 'event.preventDefault();'); //
No image drag.
                    document.getElementById('filebox').appendChild(image);
                }
            }
            section.style.zIndex = '1';
        }
        else {
            section.removeAttribute('active');
            button.removeAttribute('active');
        }
    },

```

```

dropped: function (e) {
    e.preventDefault();
    var files = e.dataTransfer.files;
    if (files.length) {
        // Based on J.D. Gauchat's "HTML5 for Masterminds 2nd ed." p.397-398.
        /*var list = '';
        for (var f = 0; f < files.length; f++) {
            var file = files[f];
            list += '<div>File. ' + file.name;
            list += '<br><span><progress value=0 max=100>0%</progress></span>';
            list += '</div>';
        }
        filebox.innerHTML = list;*/
        //var count = 0;
        var upload = function () {
            var myfile = files[0]; //count];
            var data = new FormData();
            data.append('file', myfile);
            var url = 'upload.php';
            var request = new XMLHttpRequest();
            //var xmlupload = request.upload;

            /*xmlupload.addEventListener('progress', function (e) {
                if (e.lengthComputable) {
                    var child = count + 1;
                    var per = parseInt(e.loaded / e.total * 100);
                    var progressbar = filebox.querySelector('div:nth-child(' + child +
                        ') > span > progress');
                    progressbar.value = per;
                    progressbar.innerHTML = per + '%';
                }
            });*/
            request.addEventListener('load', function (e) {
                /*var child = count + 1;
                var elem = filebox.querySelector('div:nth-child(' + child + ') >
span');

                elem.innerHTML = 'Done!';
                count++;
                if (count < files.length) upload();*/
                var data = e.target;
                if (data.status === 200)
                    document.getElementById('profileImage').src =
'getprofileimage.php?' +
                    Math.random(); // Force script reload.
            });
            request.open('POST', url, true);
            request.send(data);
        };
        upload();
    }
},

```

```

dragSource: undefined,
dragStart: function (event) {
    this.dragSource = event.target.parentNode;
    main.zeroZIndex();
    this.dragSource.style.zIndex = '1';
    var style = window.getComputedStyle(this.dragSource, null);
    var x = window.getComputedStyle(this.dragSource,
null).getPropertyValue("width");
    event.dataTransfer.setDragImage(this.dragSource, parseInt(x), 0);
    event.dataTransfer.setData("text/plain",
        (parseInt(style.getPropertyValue("left"), 10) - event.clientX) + ',' +
        (parseInt(style.getPropertyValue("top"), 10) - event.clientY));
},

dragOver: function (event) {
    event.preventDefault();
    return false;
},

drop: function (event) {
    if (event.dataTransfer.files.length === 0) { // Do not handle dragged files
here.
        var offset = event.dataTransfer.getData("text/plain").split(',');
        this.dragSource.style.left = (event.clientX + parseInt(offset[0], 10)) +
'px';
        this.dragSource.style.top = (event.clientY + parseInt(offset[1], 10)) +
'px';
    }
    event.preventDefault();
    return false;
},

zeroZIndex: function () {
    var elems = document.getElementsByTagName('section');
    for (var i = 0; i < elems.length; i++)
        elems[i].style.zIndex = '0';
},

manageZIndex: function (event) {
    main.zeroZIndex();
    event.currentTarget.style.zIndex = '1';
},

startMemberListUpdateTimer: function () {
    setInterval(function () {
        var request = new XMLHttpRequest();
        request.addEventListener('load', function (e) {
            var data = e.target;
            var ul = document.getElementById('memberList');
            if (data.status === 200) ul.innerHTML = data.responseText;
        });
    }, 1000);
}

```

```

    });
    request.open('POST', 'getmembers.php', true);
    request.send();
}, 2000);
}
};

window.addEventListener('load', main.initiate);

```

## main.css

```

body {
    background:          linear-gradient(to bottom right, yellow, #772222);
    background-attachment: fixed;
    text-shadow:         1px 1px 1px white;
    margin:              0;
}

header {
    position: fixed;
    left:      0;
    top:      0;
    width:    100%;
    height:   64px;
}

nav {
    position: fixed;
    left:     80px;
    top:      0px;
    height:   64px;
    right:    0;
}

main {
    position: absolute;
    left:     0;
    top:      64px;
    bottom:   20px;
    right:    0;
}

section {
    position: absolute;
    left:     0;
    top:      130px;
    overflow-y: auto;
    margin:    0;
}

```

```
footer {
  position:  fixed;
  left:      0;
  bottom:    0;
  width:     100%;
  text-align: center;
  font-size: 0.8em;
}

#logo {
  float:      left;
  transition:  transform 1s;
  -webkit-transition: -webkit-transform 1s;
}

#logo:hover {
  transform:   rotate(45deg);
  -webkit-transform: rotate(45deg); /* Safari, Chrome, mobile Safari, and Android */
}

ul {
  margin:  0;
  padding: 0;
}

li {
  display: inline;
}

/*article {
  background: lightgray;
  padding:    5px;
  margin:     0 10px 20px;
  box-shadow: 10px 10px 10px black;*/
/*transition: 5s;*/
/*}*/

/*article:hover {
  background: darkslategray;
}*/

h1 {
  text-shadow: 2px 2px 2px white;
  text-align: center;
}

h2 {
  margin-top: 0;
  text-shadow: 1px 1px 1px white;
}
```

```
a {
  text-decoration:      none;
  /* stackoverflow.com/questions/826782/css-rule-to-disable-text-selection-
highlighting */
  -webkit-touch-callout: none;
  -webkit-user-select:   none;
  -khtml-user-select:    none;
  -moz-user-select:      none;
  -ms-user-select:       none;
  /*user-select:         none;*/
}

#navList {
  position: fixed;
  left:     80px;
  top:      20px;
  height:   64px;
}

#navList > li {
  margin: 0;
  padding: 10px 0;
  top: 15px;
}

#navList > li > a {
  padding:      10px;
  border:       outset yellow;
  background-color: yellow;
  /*vertical-align: middle;*/
  color:        blue;
  box-shadow:    3px 3px 3px black;
  transition:    5s;
}

a:hover {
  background: linear-gradient(to bottom right, yellow, red);
  cursor:     default;
}

#navList > li > a[active] {
  background-color: lightgreen;
  border:          inset yellow;
}

#navList > li > a:visited {
  color: blue;
}

*[draggable=true] {
  cursor: move;
}
```

```

}

*[draggable=false] {
    cursor: default;
}

#quackSection, #profileSection, #memberSection {
    display: none;
    min-width: 200px;
    min-height: 200px;
    resize: both;
    background: linear-gradient(to bottom right, yellow, #772222);
    background-attachment: fixed;
    box-shadow: 10px 10px 10px black;
}

#memberSection {
    left: 500px;
}

#profileSection {
    left: 800px;
}

#quackSection[active], #profileSection[active], #memberSection[active] {
    display: block;
    border: outset gold;
}

input[disabled] {
    background: linear-gradient(to bottom right, yellow, #772222);
    background-attachment: fixed;
}

textarea {
    background: linear-gradient(to bottom right, yellow, #772222);
    background-attachment: fixed;
    border: inset gold;
}

#sendText {
    margin-top: 5px;
}

#receiveText {
    margin-bottom: 20px;
}

#profileText:focus, #sendText:focus {
    background: yellow;
}

```

```

#filebox {
  border:          2px inset black;
  background-color: black;
  max-width:       200px;
  max-height:      200px;
}

.drag {
  background-color: gold;
  opacity:         0.6;
  position:        absolute;
  right:           0px;
  top:             0px;
  width:           24px;
  height:          24px;
}

#memberListArticle {
  float: left;
  width: 200px;
}

#memberSelectArticle {
  margin-left: 200px;
}

```

## websocket.js

```

// The global variables userId and userName need to be defined before invoking this
script.
"use strict";
var databox, socket;

function initiate() {
  databox = document.getElementById('receiveText');
  document.getElementById('sendText').addEventListener('keyup', send, false);
  socket = new WebSocket("ws://foxi.ltam.lu:35000"); // CHANGE!!!
  socket.addEventListener('open', opened, false);
  socket.addEventListener('message', received, false);
  socket.addEventListener('close', closed, false);
  socket.addEventListener('error', error, false);
}

function opened() {
  databox.innerHTML += getTimeString() + ' CONNECTION OPENED  Status: ' +
    socket.readyState + '\n';
  sendJSONConnectionId();
}

```

```

function received(e) {
    databox.innerHTML += getTimeString() + " " + e.data + "\n";
    databox.scrollTop = databox.scrollHeight; // Scroll to end of textarea.
}

function closed() {
    databox.innerHTML += getTimeString() + ' CONNECTION CLOSED\n';
    document.getElementById('sendText').disabled = true;
}

function error(e) {
    databox.innerHTML += e.data + '\n';
}

function createJSONMessage(msg) {
    return JSON.stringify({
        type: "message",
        text: msg,
        user_id: userId,
        user_name: userName,
        date: getTimeString()
    });
}

function sendJSONConnectionId() {
    socket.send(JSON.stringify({
        type: "identification",
        text: "",
        user_id: userId,
        user_name: userName,
        date: getTimeString()
    }));
}

function send(e) {
    if (e.keyCode === 13 && e.shiftKey === false) {
        var command = document.getElementById('sendText').value;
        socket.send(createJSONMessage(command));
        document.getElementById('sendText').value = '';
        databox.innerHTML += getTimeString() + ' ' + command;
        databox.scrollTop = databox.scrollHeight; // Scroll to end of textarea.
    }
}

function getTimeString() {
    var d = new Date();
    var s = d.getDate() + '.' + d.getMonth() + '.' + (d.getFullYear() - 2000) + ' ' +
        d.getHours() + ':' + d.getMinutes();
    return s;
}

```

```
addEventListener('load', initiate);
```

### checkuser.php

```
<?php
    require_once 'protected/database.php';

    if (isset($_POST['user_to_check']))
        if (Database::get_user_id($_POST['user_to_check']))
            echo '<span>&nbsp;&#x2718;</span>';
        else echo '<span id=available>&nbsp;&#x2714;</span>';
?>
```

### logout.php

```
<?php
    #require_once 'protected/database.php';
    if (!isset($_SESSION)) session_start();
    /*if (isset($_SESSION['user_id'])) {
        Database::logout($_SESSION['user_id']);
    }*/
    $_SESSION = array();
    if (session_id() != "" || isset($_COOKIE[session_name()])) setcookie(session_name(),
        '', 1, '/');
    session_destroy();
    header('Location: index.php');
?>
```

### deleteprofile.php

```

<?php
    require_once 'protected/bouncer.php';
    require_once 'protected/database.php';
    Database::delete_user($_SESSION['user_id']);
    $dir_path = "./protected/users/{$_SESSION['user_name']}";
    # http://stackoverflow.com/questions/1407338/a-recursive-remove-directory-function-
    for-php
    try {
        foreach (new RecursiveIteratorIterator(new RecursiveDirectoryIterator($dir_path,
            FilesystemIterator::SKIP_DOTS), RecursiveIteratorIterator::CHILD_FIRST) as
    $path) {
            $path->isFile() ? unlink($path->getPathname()) : rmdir($path->getPathname());
        }
        rmdir($dir_path);
    } catch (Exception $e) {
    }
    header('Location: logout.php');
?>

```

#### deleteprofileimage.php

```

<?php
    require_once 'protected/bouncer.php';
    $file_path = '';
    $dir_path = "./protected/users/{$_SESSION['user_name']}";
    try { # Run through all files in the directory. This primitive approach is good for
    now.
        foreach (new RecursiveIteratorIterator(new RecursiveDirectoryIterator($dir_path,
            FilesystemIterator::SKIP_DOTS), RecursiveIteratorIterator::CHILD_FIRST) as
    $path) {
            if ($path->isFile()) $file_path = $path->getPathname();
        }
    } catch (Exception $e) {
    }
    if ($file_path !== '') unlink($file_path);
?>

```

#### getprofileimage.php

```

<?php
    require_once 'protected/bouncer.php';
    $file_path = '';
    if (isset($_GET['user_name'])) $dir_path = "./protected/users/{$_GET['user_name']}";
    else $dir_path = "./protected/users/{$_SESSION['user_name']}";
    try { # Run through all files in the directory. This primitive approach is good for
now.
        foreach (new RecursiveIteratorIterator(new RecursiveDirectoryIterator($dir_path,
            FilesystemIterator::SKIP_DOTS), RecursiveIteratorIterator::CHILD_FIRST) as
$path) {
            if ($path->isFile()) $file_path = $path->getPathname();
        }
    } catch (Exception $e) {
    }
    header('Content-type: image/png');
    if ($file_path !== '') readfile($file_path);
?>

```

#### getmember.php

```

<?php
    require_once 'protected/bouncer.php';
    require_once 'protected/database.php';
    $output = '';
    if (isset($_POST['user_id'])) {
        $desc = Database::get_user_data($_POST['user_id']);
        if ($desc) {
            $output .= '<p>' . $desc['user_name'] . '</p>';
            $output .= '<img src=getprofileimage.php?user_name=' . $desc['user_name'];
            $output .= ' draggable=false onmousedown="event.preventDefault();"';
            $output .= ' alt=""><br>';
            $output .= '<p><pre>' . $desc['description'] . '</pre></p>';
        }
    }
    echo $output;
?>

```

#### getmembers.php

```

<?php
require_once 'protected/bouncer.php';
require_once 'protected/database.php';
if (isset($_SESSION['user_id']))
    Database::update_login_timestamp($_SESSION['user_id']);
$users = Database::get_users();
$output = '';
if ($users) {
    foreach ($users as $user) {
        $output .= '<li><a onclick="main.viewMemberProfile(' . $user['id'] . ');">' .
            $user['first_name'] . ' ' . $user['last_name'] . '</a>';
        if (Database::is_logged_in($user['id']))
            $output .= ' &nbsp;&nbsp;&nbsp;<img src=green.png alt=green.png>';
        $output .= '</li><br>';
    }
}
echo $output;
?>

```

#### stillloggedin.php

```

<?php
ini_set('session.use_only_cookies', true);
ini_set('session.cookie_path', dirname(htmlspecialchars($_SERVER['PHP_SELF'])));
/* if (!isset($_SERVER['HTTPS'])) // # SSL is not active, activate it.
    header('Location: https://' . $_SERVER['HTTP_HOST'] .
        dirname(htmlspecialchars($_SERVER['PHP_SELF'])));*/
if (!isset($_SESSION)) session_start();
# After 30 seconds we'll generate a new session ID to prevent a session
# fixation attack (cf. PHP cookbook p. 338).
if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)){
    session_regenerate_id();
    $_SESSION['generated'] = time();
}
if (!isset($_SESSION['user_id'])) echo 'false';
else echo 'true';
?>

```

#### updatedescription.php

```

<?php
require_once 'protected/bouncer.php';
require_once 'protected/database.php';

Database::update_description($_SESSION['user_id'], $_POST['description']);
?>

```

```

<?php
require_once 'protected/bouncer.php';
$dir_path = "protected/users/{$_SESSION['user_name']}";
try { # First we delete all files in the user directory.
    foreach (new RecursiveIteratorIterator(new RecursiveDirectoryIterator($dir_path,
        FilesystemIterator::SKIP_DOTS), RecursiveIteratorIterator::CHILD_FIRST) as
$path) {
        if ($path->isFile()) unlink($path->getPathname());
    }
} catch (Exception $e) {
}
if (isset($_FILES['file'])) { # Then we upload the new image.
    $save_to = $dir_path . '/' . $_FILES['file']['name'];
    move_uploaded_file($_FILES['file']['tmp_name'], $save_to);
    # From Robin Nixon's "Learning PHP, MySQL, JavaScript & CSS" 2nd ed. p. 484
    $type_OK = TRUE;
    switch ($_FILES['file']['type']) {
        case 'image/gif':
            $src = imagecreatefromgif($save_to);
            break;
        case 'image/jpeg': # Allow both regular and progressive jpegs.
        case 'image/pjpeg':
            $src = imagecreatefromjpeg("$save_to");
            break;
        case 'image/png':
            $src = imagecreatefrompng("$save_to");
            break;
        default: $type_OK = FALSE;
    }

    if ($type_OK) {
        list($w, $h) = getimagesize($save_to);
        $max = 200;
        $tw = $w;
        $th = $h;
        if ($w > $h && $max < $w) {
            $th = $max / $w * $h;
            $tw = $max;
        }
        elseif ($h > $w && $max < $h) {
            $tw = $max / $h * $w;
            $th = $max;
        }
        elseif ($max < $w) $tw = $th = $max;
        $tmp = imagecreatetruecolor($tw, $th);
        imagecopyresampled($tmp, $src, 0, 0, 0, 0, $tw, $th, $w, $h);
        imageconvolution($tmp, array(array(-1, -1, -1), array(-1, 16, -1),
            array(-1, -1, -1)), 8, 0);
        imagepng($tmp, $save_to);
    }
}

```

```

        imagedestroy($tmp);
        imagedestroy($src);
    }
}
?>

```

## bouncer.php

```

<?php
#echo '<pre>' . print_r($_SERVER, true) . '</pre>';
#echo $_SERVER['HTTP_REFERER'];
# Only send session id cookie over SSL.
//ini_set('session.cookie_secure', true);
# Session IDs may only be passed via cookies, not appended to URL.
ini_set('session.use_only_cookies', true);
ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
/*if (!isset($_SERVER['HTTPS'])) { // # SSL is not active, activate it.
    header('Location: https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']));
    exit;
}*/
if (!isset($_SESSION)) session_start();
# After 30 seconds we'll generate a new session ID to prevent a session
# fixation attack (cf. PHP cookbook p. 338).
if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
    session_regenerate_id();
    $_SESSION['generated'] = time();
}
if (!isset($_SESSION['user_id'])) { // No user logged in -> go to the login page.
    header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php');
    exit;
}
/*echo '<pre>' . print_r($_SESSION, true) . '</pre>';
if(isset($_SESSION['lastpage']) && $_SESSION['lastpage'] == __FILE__){
    if($_SERVER['QUERY_STRING'] != $_SESSION['querystring']){
        echo "Same page but querystring changed";
    }elseif($_SERVER['REQUEST_METHOD'] == "POST"){
        echo "This is in response to a form submission";
    }else{
        echo "Either a refresh or same page as last page when re-entering site within
session timeout period.";
    }
}else{
    echo "New page";
}*/
/*$_SESSION['lastpage'] = __FILE__;
$_SESSION['querystring'] = $_SERVER['QUERY_STRING'];*/
?>

```

```

#!/php -q
<?PHP
require_once("websocket.server.php");

class DemoEchoHandler extends WebSocketUriHandler {
    public function onMessage(IWebSocketConnection $user, IWebSocketMessage $msg) {
        $this->say("[ECHO] " . strlen($msg->getData()) . " bytes");
        $user->sendMessage($msg);
    }

    public function onAdminMessage(IWebSocketConnection $user, IWebSocketMessage $obj)
{
    $this->say("[DEMO] Admin TEST received!");

    $frame = WebSocketFrame::create(WebSocketOpcode::PongFrame);
    $user->sendFrame($frame);
}
}

class DemoSocketServer implements IWebSocketServerObserver {
    protected $debug = true;
    protected $server;

    public function __construct() {
        $this->server = new WebSocketServer("tcp://0.0.0.0:35000",
            'superdupersecretkey');
        $this->server->addObserver($this);
        $this->server->addUriHandler("echo", new DemoEchoHandler());
    }

    public function onConnect(IWebSocketConnection $user) {

        $this->say("[DEMO] {$user->getId()} connected");
    }

    public function onMessage(IWebSocketConnection $user, IWebSocketMessage $msg) {
        $thisuser = $user->getId();
        $msg      = json_decode($msg->getData());
        print_r($msg);
        if ($msg->type === 'identification') $this->say('Identification');
        else {
            $msgback = WebSocketMessage::create($msg->user_name . ' quacks: ' .
                trim($msg->text));
            foreach ($this->server->getConnections() as $user)
                if ($user->getId() != $thisuser) $user->sendMessage($msgback);
        }
    }

    public function onDisconnect(IWebSocketConnection $user) {

```

```
$this->say("[DEMO] {$user->getId()} disconnected");
}

public function onAdminMessage(IWebSocketConnection $user, IWebSocketMessage $msg)
{
    $this->say("[DEMO] Admin Message received!");
    $frame = WebSocketFrame::create(WebSocketOpcode::PongFrame);
    $user->sendFrame($frame);
}

public function say($msg) {
    echo "$msg \r\n";
}

public function run() {
    $this->server->run();
}

$server = new DemoSocketServer();
$server->run();
?>
```

# Chapter 10. Tutorials

## 10.1. WMOTU Lab

`<a href="https://students.btsi.lu/evegi144/WAD/WMOTULabv1/index.html">" width="815.25" tmp="false">^]`

### 10.1.1. index.html

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU Lab v1</title>
    <meta charset=utf-8>
    <link href=style.css rel=stylesheet>
  </head>
  <body>
    <header>
      <img id=logo src=logo.png alt=Logo width=64 height=64>
      <nav>
        <ul id=navList>
          <li><a id=selectedPage>Home</a></li>
          <li><a href=html5.html>HTML5</a></li>
          <li><a href=css3.html>CSS3</a></li>
          <li><a href=javascript.html>JavaScript</a></li>
          <li><a href=php5.html>PHP5</a></li>
          <li><a href=mysql5.html>MySQL5</a></li>
        </ul>
      </nav>
    </header>
    <main>
      <h1>Latest news</h1>
      <section>
        <ul id=articleList>
          <li>
            <a href=#>
              <article>
                <h2>The sixth alpha version of WMOTU Lab goes live!</h2>
                <time datetime=2014-01-04>4.1.14</time>
                <p>Exciting times...</p>
              </article>
            </a>
          </li>
          <li>
            <a href=#>
              <article>
                <h2>The fifth alpha version of WMOTU Lab goes live!</h2>
```

```

        <time datetime=2014-01-04>4.1.14</time>
        <p>Exciting times...</p>
    </article>
</a>
</li>
<li>
    <a href=#>
        <article>
            <h2>The fourth alpha version of WMOTU Lab goes live!</h2>
            <time datetime=2014-01-04>4.1.14</time>
            <p>Exciting times...</p>
        </article>
    </a>
</li>
<li>
    <a href=#>
        <article>
            <h2>The third alpha version of WMOTU Lab goes live!</h2>
            <time datetime=2014-01-04>4.1.14</time>
            <p>Exciting times...</p>
        </article>
    </a>
</li>
<li>
    <a href=#>
        <article>
            <h2>The second alpha version of WMOTU Lab goes live!</h2>
            <time datetime=2014-01-04>4.1.14</time>
            <p>Exciting times...</p>
        </article>
    </a>
</li>
<li>
    <a href=#>
        <article>
            <h2>The first alpha version of WMOTU Lab goes live!</h2>
            <time datetime=2014-01-04>4.1.14</time>
            <p>Exciting times...</p>
        </article>
    </a>
</li>
</ul>
</section>
</main>
<footer>&copy; 2014 WMOTU</footer>
</body>
</html>

```

## 10.1.2. html5.html

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU Lab v1</title>
    <meta charset=utf-8>
    <link href=style.css rel=stylesheet>
  </head>
  <body>
    <header>
      <img id=logo src=logo.png alt=Logo width=64 height=64>
      <nav>
        <ul id=navList>
          <li><a href=index.html>Home</a></li>
          <li><a id=selectedPage>HTML5</a></li>
          <li><a href=css3.html>CSS3</a></li>
          <li><a href=javascript.html>JavaScript</a></li>
          <li><a href=php5.html>PHP5</a></li>
          <li><a href=mysql5.html>MySQL5</a></li>
        </ul>
      </nav>
    </header>
    <main>
      <table>
        <caption>HTML5 Resources</caption>
        <thead>
          <tr>
            <th>Web</th>
            <th>Books</th>
          </tr>
        </thead>
        <tbody>
          <tr>
            <td><a href=http://developers.whatwg.org
              target=_blank>HTML: The Living Standard</a></td>
            <td><a href=http://shop.oreilly.com/product/9780596159924.do
              target=_blank>Head First HTML and CSS</a></td>
          </tr>
          <tr>
            <td><a href=http://w3schools.com/html/html5_intro.asp
              target=_blank>w3schools.com</a></td>
            <td></td>
          </tr>
          <tr>
            <td><a href=http://www.w3.org/TR/html51
              target=_blank>The official working draft</a></td>
            <td></td>
          </tr>
          <tr>
```

```

        <td><a href=http://css3-html5.de/html-css-lernen
            target=_blank>CSS3 & HTML5 (German)</a></td>
    </td></td>
</tr>
<tr>
    <td><a href=http://www.html-seminar.de
        target=_blank>HTML-Seminar (German)</a></td>
    <td></td>
</tr>
<tr>
    <td><a href=http://html5dog.com
        target=_blank>HTML Dog</a></td>
    <td></td>
</tr>
<tr>
    <td><a href=http://learn.shayhowe.com
        target=_blank>A Practical Guide to HTML & CSS</a></td>
    <td></td>
</tr>
<tr>
    <td><a href=http://www.quackit.com
        target=_blank>Quackit</a></td>
    <td></td>
</tr>
<tr>
    <td><a href=http://www.html5code.nl
        target=_blank>HTML5 Code</a></td>
    <td></td>
</tr>
<tr>
    <td><a href=http://coding.smashingmagazine.com
        target=_blank>Smashing Magazine</a></td>
    <td></td>
</tr>
<tr>
    <td><a href=https://bitbucket.org/webRTC/codelab
        target=_blank>Codelab WebRTC</a></td>
    <td></td>
</tr>
<tr>
    <td><a href=http://www.html5rocks.com/en/tutorials/webRTC/basics
        target=_blank>HTML5 Rocks WebRTC Tutorial</a></td>
    <td></td>
</tr>
<tr>
    <td><a href=http://www.w3.org/WAI/intro/wcag.php
        target=_blank>Web Content Accessibility Guidelines</a></td>
    <td></td>
</tr>
</tbody>
</table>

```

```
</main>
<footer>&copy; 2014 WMOTU</footer>
</body>
</html>
```

### 10.1.3. style.css

```
@import url("https://fonts.googleapis.com/css?family=Prosto+One");

body {
  background:          linear-gradient(to bottom right, yellow, #772222);
  background-attachment: fixed;
  font-family:         'Prosto One', cursive;
}

header {
  position: fixed;
  left:    0;
  top:     0;
  width:   100%;
  height:  64px;
}

nav {
  height:    44px;
  padding-top: 20px;
  text-align: center;
}

main {
  position: fixed;
  left:    0;
  top:     64px;
  bottom:  20px;
  right:   0;
}

section {
  position: fixed;
  left:    0;
  top:     130px;
  bottom:  20px;
  right:   0px;
  overflow-y: auto;
  margin:  0;
}

footer {
  position: fixed;
  left:    0;
```

```
    bottom:    0;
    width:     100%;
    text-align: center;
    font-size: 0.8em;
}

#logo {
    float:      left;
    transition: transform 1s;
    -webkit-transition: -webkit-transform 1s;
}

#logo:hover {
    transform:      rotate(45deg);
    -webkit-transform: rotate(45deg); /* Safari, Chrome, mobile Safari, and Android */
}

ul {
    margin: 0;
    padding: 0;
}

li {
    display: inline;
}

#articleList > li {
    display: block;
}

article {
    background: lightgray; /*linear-gradient(to bottom right, lightgray,
    darkslategray);*/
    padding: 5px;
    margin: 0 10px 20px;
    box-shadow: 10px 10px 10px black;
    transition: 5s;
}

article:hover {
    background: darkslategray; /*linear-gradient(to bottom right, darkslategray,
    lightgray);*/
}

h1 {
    text-shadow: 2px 2px 2px white;
    text-align: center;
}

h2 {
    margin-top: 0;
}
```

```
text-shadow: 1px 1px 1px white;
}

time {
  border: 1px outset gold;
  padding: 1px 3px;
}

a {
  text-decoration: none;
}

#navList > li {
  margin: 0;
  padding: 0;
}

#navList > li > a {
  padding: 10px;
  border: outset yellow;
  background-color: yellow;
  vertical-align: middle;
}

#navList > li > a:hover {
  background: linear-gradient(to bottom right, yellow, red);
}

#navList > li > #selectedPage {
  background-color: red;
}

#navList > li > #selectedPage:hover {
  background: red;
}

table {
  overflow-y: auto;
}

table, th, td {
  border: 2px outset yellow;
}

table a {
  padding: 0 5px;
  background-color: yellow;
  vertical-align: middle;
  position: relative;
  display: block;
}
```

```
table a:hover {
    background: linear-gradient(to bottom right, yellow, red);
}
```

## 10.2. WMOTU Invaders

```
<a href="https://students.btsi.lu/evegi144/WAD/WMOTUInvaders">" width="452.25" tmp="false">^]
```

Developed in the context of the CLISS1 and CLISS2 modules, this game illustrates the simple functional structuring of the classic space invaders.

The solution consists of three files and demonstrates array, canvas, event and collision handling. JSONP is used to read and save the scores to a MySQL database.

### 10.2.1. index.html

The HTML file provides the HTML content, the CSS styling and

```
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU Invaders v1.0</title>
    <meta charset=UTF-8>
    <link href=index.css rel=stylesheet>
  </head>
  <body>
    <section id=splash>
      <h2>WMOTU Invaders v1.0</h2>
      <ul>
        <li>
          <button id=newGame onclick=newGame();>New Game</button>
        </li>
        <li>
          <button id=highScores onclick=displayHighScores();>High Scores</button>
        </li>
        <li>
          <button id=keys onclick=displayKeys();>Keys</button>
        </li>
      </ul>
    </section>
    <section id=highScoreSection>
      <button style="position: fixed; top: 0;" onclick=hideHighScores();>OK</button>
      <table id=highScoreListTable>
        <caption>Hall of Fame</caption>
        <thead>
          <tr>
```

```

        <th>Rank</th>
        <th>Player</th>
        <th>Score</th>
        <th>Level</th>
    </tr>
</thead>
</table>
</section>
<section id=keysSection>
    <p>Use left and right cursor keys to move your spaceship and the space bar to
fire.

    The game can be paused at any time by pressing <code>P</code>. Pressing
    <code>P</code> again will resume.
    You can exit the game early using <code>Esc</code>.</p>
    <button onclick=hideKeys();>OK</button>
</section>
<section id=board>
    <p>
        Score: <span id=score></span>&nbsp;
        Level: <span id=level></span>
    </p>
    <canvas id=canvas width=600 height=400>This browser does not run this game
(canvas
    support missing).
</canvas>
<section>
    <img id=life1 src=playerspaceship36x46.png alt=playerspaceship36x46.png>
    <img id=life2 src=playerspaceship36x46.png alt=playerspaceship36x46.png>
</section>
</section>
<script src=index.js></script>
</body>
</html>

```

### 10.2.2. index.js

```

"use strict";
// Declaration of global variables
const canvas = document.getElementById('canvas');
const context = canvas.getContext('2d');
const imagePlayer = new Image(36, 46);
const imageShot = new Image(12, 23);
const imageAlien = new Image(42, 27);
const imageAlienBomb = new Image(10, 10);
imagePlayer.src = 'playerspaceship36x46.png';
imageShot.src = 'shot12x23.png';
imageAlien.src = 'alien42x27.png';
imageAlienBomb.src = 'alienbomb10x10.png';
let currPlayerX = (canvas.width - imagePlayer.width) / 2;
let currPlayerY = canvas.height - imagePlayer.height - 5;

```

```

let playerShotsX = [], playerShotsY = [], initialPlayerSpeed = 20, initialNumOfLives = 2;
let numAliens = 40, numAliensPerRow = 10, minFireThreshold = 300;
const DBServerURL = "index.php?callback=";
let aliensX = [], aliensY = [], alienBombsX = [], alienBombsY = [];
let highScoresReady, alienDirection, fireButton = false, moveLeft = false, moveRight = false,
    timeOfLastShot = 0;
let alienXSpeed, alienYSpeed, loseLife, numOfLives, alienBombSpeed;
let score = ~0, highScoreList, currLevel = 1, currPlayerSpeed, lastName = "",
numShots,
    pauseButton, invisible, lastAnimationTime = 0, adjustmentFactor = 0.2;

// http://codeincomplete.com/posts/2013/12/4/javascript_game_foundations_the_game_loop
const timestamp = () => {
    return window.performance && window.performance.now ?
        window.performance.now() : new Date().getTime();
};

/* Called by newGame.
 * Draws aliens and adjusts global variables. */
const init = () => {
    let x = 1, y = 1;
    for (let i = 0; i < numAliens; i++) {
        aliensX[i] = x;
        aliensY[i] = y;
        context.drawImage(imageAlien, aliensX[i], aliensY[i]);
        x += imageAlien.width + 10;
        if (i % numAliensPerRow == 9) { // 10 aliens per row
            x = 1;
            y += imageAlien.height + 7;
        }
    }
    document.getElementById("score").innerHTML = ~(score >> 2);
    document.getElementById("level").innerHTML = currLevel;
    alienXSpeed = (5 + currLevel) * adjustmentFactor;
    alienYSpeed = currLevel;
    alienBombSpeed = Math.max(1, Math.floor(alienYSpeed / 3));
    alienDirection = 1; // aliens start moving to the right
    moveLeft = false; // no keys pressed
    moveRight = false;
    playerShotsX = []; // delete shots
    playerShotsY = [];
    alienBombsX = []; // delete bombs
    alienBombsY = [];
};

// Called by gameLoop.
const moveAliens = () => {
    let leftBorderTouched = false;
    for (let i = 0; i < aliensX.length; i++) {

```

```

    // We need those to determine whether there is enough room to continue in the
    current
    // direction. If there isn't enough room, we need to change direction.
    if (aliensX[i] <= alienXSpeed && alienDirection === -1 || aliensX[i] +
imageAlien.width
    > canvas.width && alienDirection === 1) {
        alienDirection = -alienDirection;
        // If we touched the left border, we need to move the aliens down.
        if (alienDirection === 1) leftBorderTouched = true;
        break; // Once we have changed direction, no need to check the other aliens.
    }
}
if (leftBorderTouched) {
    alienYSpeed += 2 * adjustmentFactor;
    for (let i = 0; i < aliensY.length; i++) aliensY[i] += alienYSpeed;
}
// Move aliens horizontally.
for (let i = 0; i < aliensX.length; i++) {
    aliensX[i] += alienDirection * alienXSpeed;
    context.drawImage(imageAlien, aliensX[i], aliensY[i]);
}
};

```

```

// Called by gameLoop
const moveShotsAndBombs = () => {
    for (let i = 0; i < playerShotsY.length; i++) {
        playerShotsY[i] -= imageShot.height * adjustmentFactor;
        if (playerShotsY[i] < 0) { // delete bullet as it has left space
            playerShotsX.splice(i, 1);
            playerShotsY.splice(i, 1);
            i--;
        } else context.drawImage(imageShot, playerShotsX[i], playerShotsY[i]);
    }
    for (let i = 0; i < alienBombsY.length; i++) {
        alienBombsY[i] += alienBombSpeed;
        if (alienBombsY[i] > canvas.height) { // delete bomb as it has left space
            alienBombsX.splice(i, 1);
            alienBombsY.splice(i, 1);
            i--;
        } else context.drawImage(imageAlienBomb, alienBombsX[i], alienBombsY[i]);
    }
};

```

```

// Called by gameLoop.
// check whether a bullet has hit an alien or an alien or alien bomb touches the
player
const checkCollisions = () => {
    let shotLeft, shotRight, shotTop, shotBottom, alienLeft, alienRight, alienTop,
alienBottom,
    playerRight, playerBottom, alienIndex = 0, alienBombLeft, alienBombRight,
alienBombTop,

```

```

    alienBombBottom, alienBombIndex = 0;

    // first check whether player has been touched by an alien or a bullet has touched
    an alien
    playerRight = currPlayerX + imagePlayer.width;
    playerBottom = currPlayerY + imagePlayer.height;

    while (alienIndex < aliensX.length) {
        alienLeft = aliensX[alienIndex];
        alienRight = alienLeft + imageAlien.width;
        alienTop = aliensY[alienIndex];
        // if an alien manages to leave via the bottom of the frame, we lose a life
        alienBottom = alienTop + imageAlien.height;
        if (alienRight >= currPlayerX && playerRight >= alienLeft && alienBottom >=
            currPlayerY || alienBottom > canvas.height) {
            loseLife = true;
            return;
        }
        let shotIndex = 0;
        // for each bullet check whether it touches the alien
        while (shotIndex < playerShotsY.length) {
            shotLeft = playerShotsX[shotIndex];
            shotRight = shotLeft + imageShot.width;
            shotTop = playerShotsY[shotIndex];
            shotBottom = shotTop + imageShot.height;
            if (alienRight >= shotLeft && shotRight >= alienLeft && alienBottom >= shotTop
&&
                shotBottom >= alienTop) {
                playerShotsX.splice(shotIndex, 1);
                playerShotsY.splice(shotIndex, 1);
                aliensX.splice(alienIndex, 1);
                aliensY.splice(alienIndex, 1);
                alienIndex--;
                score = ~((~(score >> 2) + 1) << 2);
                document.getElementById("score").innerHTML = ~(score >> 2);
                break;
            }
            shotIndex++;
        }
        alienIndex++;
    }

    while (alienBombIndex < alienBombsX.length) { // check whether bomb has touched
    player
        alienBombLeft = alienBombsX[alienBombIndex];
        alienBombRight = alienBombLeft + imageAlienBomb.width;
        alienBombTop = alienBombsY[alienBombIndex];
        alienBombBottom = alienBombTop + imageAlienBomb.height;
        if (alienBombRight >= currPlayerX && playerRight >= alienBombLeft &&
alienBombBottom
            >= currPlayerY && alienBombTop <= playerBottom) {

```

```

        loseLife = true;
        return;
    }
    alienBombIndex++;
}
};

const handleKeyDown = event => {
    if (event.keyCode === 80) {
        pauseButton = !pauseButton; // P pressed
        lastAnimationTime = timestamp();
    } else if (event.keyCode === 27) { // ESC
        loseLife = true;
        numOfLives = 0;
    } else if (event.keyCode === 32) { // fire shot
        fireButton = true;
    } else if (event.keyCode === 37) { // move left
        if (moveLeft) currPlayerSpeed += 2;
        moveLeft = true;
        moveRight = false;
    } else if (event.keyCode === 39) { // move right
        if (moveRight) currPlayerSpeed += 2;
        moveRight = true;
        moveLeft = false;
    }
};

const handleKeyUp = event => {
    if (event.keyCode === 32) { // fire shot
        fireButton = false;
    } else if (event.keyCode === 37) { // move left
        moveLeft = false;
        currPlayerSpeed = initialPlayerSpeed;
    } else if (event.keyCode === 39) { // move right
        moveRight = false;
        currPlayerSpeed = initialPlayerSpeed;
    }
};

const handleVisibilityChange = () => {
    if (invisible) lastAnimationTime = timestamp();
    invisible = !invisible;
};

const restartLevel = () => {
    init();
    lastAnimationTime = 0;
    gameLoop(timestamp());
};

// This is the function that controls the game. Called by newGame and

```

```

requestAnimationFrame.
const gameLoop = currTime => {
  if (pauseButton || invisible) requestAnimationFrame(gameLoop); // if paused, do
  nothing.
  else {
    let timeElapsed = currTime - lastAnimationTime;
    if (lastAnimationTime === 0) adjustmentFactor = 0.2;
    else adjustmentFactor = timeElapsed / 100;
    context.clearRect(0, 0, canvas.width, canvas.height);
    // If space key pressed and enough time since the last shot has elapsed, we shoot
    again.
    if (fireButton && ((currTime - timeOfLastShot) > minFireThreshold)) {
      timeOfLastShot = currTime;
      if (numShots === 3) {
        playerShotsX.push(currPlayerX);
        playerShotsY.push(currPlayerY);
        context.drawImage(imageShot, currPlayerX, currPlayerY);
        playerShotsX.push(currPlayerX + (imagePlayer.width - imageShot.width) / 2);
        playerShotsY.push(currPlayerY);
        context.drawImage(imageShot, currPlayerX + (imagePlayer.width -
          imageShot.width) / 2, currPlayerY);
        playerShotsX.push(currPlayerX + imagePlayer.width - imageShot.width);
        playerShotsY.push(currPlayerY);
        context.drawImage(imageShot, currPlayerX + imagePlayer.width -
imageShot.width,
          currPlayerY);
      } else if (numShots === 2) {
        playerShotsX.push(currPlayerX);
        playerShotsY.push(currPlayerY);
        context.drawImage(imageShot, currPlayerX, currPlayerY);
        playerShotsX.push(currPlayerX + imagePlayer.width - imageShot.width);
        playerShotsY.push(currPlayerY);
        context.drawImage(imageShot, currPlayerX + imagePlayer.width -
imageShot.width,
          currPlayerY);
      } else {
        playerShotsX.push(currPlayerX + (imagePlayer.width - imageShot.width) / 2);
        playerShotsY.push(currPlayerY);
        context.drawImage(imageShot, currPlayerX + (imagePlayer.width -
          imageShot.width) / 2, currPlayerY);
      }
    }
    if (moveLeft) { // If left arrow key pressed, move spaceship to the left.
      if (currPlayerX > currPlayerSpeed)
        currPlayerX -= currPlayerSpeed * adjustmentFactor;
      else currPlayerX = 1;
    }
    if (moveRight) { // If right arrow key pressed, move spaceship to the right.
      if ((currPlayerX + imagePlayer.width) < (canvas.width - currPlayerSpeed))
        currPlayerX += currPlayerSpeed * adjustmentFactor;
      else currPlayerX = canvas.width - imagePlayer.width - 1;
    }
  }
}

```

```

    }
    for (let i = 0; i < aliensX.length; i++) { // generate bombs
        if (Math.random() > (1 - (currLevel * adjustmentFactor / 3000))) {
            alienBombsX.push(aliensX[i] + (imageAlien.width - imageAlienBomb.width) / 2);
            alienBombsY.push(aliensY[i] + imageAlien.height);
            context.drawImage(imageAlienBomb, aliensX[i] + (imageAlien.width -
                imageAlienBomb.width) / 2, aliensY[i] + imageAlien.height);
        }
    }
}
checkCollisions();
if (loseLife) { // If we have been hit or touched...
    if (numOfLives >= 1)
        document.getElementById("life" + numOfLives).style.display = "none";
    numOfLives--;
    loseLife = false;
    if (numOfLives < 0) gameOver();
    else restartLevel();
} else if (aliensX.length === 0) nextLevel();
else { // Move everything and continue the fun.
    alienXSpeed = (5 + currLevel) * adjustmentFactor;
    context.drawImage(imagePlayer, currPlayerX, currPlayerY);
    moveShotsAndBombs();
    moveAliens();
    lastAnimationTime = timestamp(); //new Date().getTime();
    requestAnimationFrame(gameLoop); //setTimeout("gameLoop()", timeOut);
}
}
};

const newGame = () => {
    document.getElementById("splash").style.display = "none";
    document.getElementById("board").style.display = "block";
    score = ~0;
    currLevel = 1;
    currPlayerSpeed = initialPlayerSpeed;
    // Display spare spaceships.
    for (var i = 1; i <= initialNumOfLives; i++)
        document.getElementById("life" + i).style.display = "inline";
    numOfLives = initialNumOfLives;
    loseLife = false;
    fireButton = false;
    currPlayerX = (canvas.width - imagePlayer.width) / 2;
    currPlayerY = canvas.height - imagePlayer.height - 5;
    window.onkeydown = handleKeyDown;
    window.onkeyup = handleKeyUp;
    document.addEventListener('visibilitychange', handleVisibilityChange, false);
    numShots = 1;
    minFireThreshold = 300;
    pauseButton = false;
    invisible = false;
    init();
}

```

```

    lastAnimationTime = 0;
    gameLoop(timestamp());
};

const nextLevel = () => {
    currLevel++;
    if (currLevel >= 25) numShots = 3;
    else if (currLevel >= 15) numShots = 2;
    if (currLevel >= 20) minFireThreshold = 100;
    else if (currLevel >= 10) minFireThreshold = 200;
    document.getElementById("level").innerHTML = currLevel;
    lastAnimationTime = 0;
    init();
    gameLoop(timestamp());
};

const gameOver = () => {
    window.onkeydown = null;
    window.onkeyup = null;
    alert("Game over!");
    updateHighScores();
    displayHighScores();
};

const displayHighScores = () => {
    if (highScoresReady) {
        document.getElementById("board").style.display = "none";
        document.getElementById("splash").style.display = "none";
        document.getElementById("highScoreSection").style.display = "block";
    } else setTimeout(displayHighScores, 100);
};

const hideHighScores = () => {
    showSplash();
};

const readHighScores = scores => { // callback for PHP
    highScoreList = scores;
    const HL = document.getElementById("highScoreListTable");
    if (HL) {
        const childNodes = HL.tBodies;
        let x = childNodes.length;
        while (x > 0) {
            HL.removeChild(childNodes[0]);
            x = childNodes.length;
        }
    }
    const body = document.createElement("tbody");
    for (let i = highScoreList.length - 1; i >= 0; i--) {
        let row = body.insertRow(0);
        let cell1 = row.insertCell(0);
    }
}

```

```

    let cell2 = row.insertCell(1);
    let cell3 = row.insertCell(2);
    let cell4 = row.insertCell(3);
    cell1.innerHTML = (i + 1);
    cell2.innerHTML = highScoreList[i].Player.slice(0, Math.min(30,
        highScoreList[i].Player.length));
    cell3.innerHTML = highScoreList[i].HighScore;
    cell4.innerHTML = highScoreList[i].Level;
    cell1.style.cssText = "text-align: right";
    cell3.style.cssText = "text-align: right";
    cell4.style.cssText = "text-align: right";
}
HL.appendChild(body);
highScoresReady = true;
};

const loadSaveHighScores = (action, player) => { // action = "insert" or nothing
    const newScriptElement = document.createElement("script");
    highScoresReady = false;
    if (action && player) newScriptElement.setAttribute("src", DBServerURL +
        "readHighScores&action=insert&player=" + player + "&score=" + ~(score >> 2) +
        "&level=" + currLevel);
    else newScriptElement.setAttribute("src", DBServerURL + "readHighScores");
    newScriptElement.setAttribute("id", "jsonp");
    const oldScriptElement = document.getElementById("jsonp");
    const head = document.getElementsByTagName("head")[0];
    if (oldScriptElement === null) head.appendChild(newScriptElement);
    else head.replaceChild(newScriptElement, oldScriptElement);
};

const updateHighScores = () => {
    loadSaveHighScores();
    if (~(score >> 2) > 0) {
        lastName = prompt("Enter your name", lastName);
        loadSaveHighScores("insert", lastName);
    }
    if (lastName === null) lastName = "";
};

const showSplash = () => {
    document.getElementById("splash").style.display = "block";
    document.getElementById("highScoreSection").style.display = "none";
    document.getElementById("keysSection").style.display = "none";
    document.getElementById("board").style.display = "none";
};

const displayKeys = () => {
    document.getElementById("splash").style.display = "none";
    document.getElementById("highScoreSection").style.display = "none";
    document.getElementById("keysSection").style.display = "block";
    document.getElementById("board").style.display = "none";
};

```

```
};

const hideKeys = () => {
  showSplash();
};

loadSaveHighScores();
```

### 10.2.3. index.php

```
<?php
require_once 'db_credentials.php';
@$db = new mysqli(DB_HOST, DB_USER, DB_PASSWORD, DB_NAME);
if (mysqli_connect_errno()) {
  echo "Error: could not connect to database. Please try again later.";
  exit;
}

if (isset($_GET["callback"])) {
  if (isset($_GET["action"]) && $_GET["action"] === "insert") {
    if (isset($_GET["player"]) && isset($_GET["score"]) && isset($_GET["level"])) {
      $sqlq = "INSERT INTO T1IF_Invaders VALUES (NULL, '" . $_GET["player"] . "',
      '" . $_GET["score"] . "', '" . $_GET["level"] . "')";
      $result = $db->query($sqlq);
    }
  }
  $sqlq = "SELECT * FROM T1IF_Invaders ORDER BY HighScore DESC";
  $result = $db->query($sqlq);
  while ($row = $result->fetch_assoc()) {
    $results_array[] = $row;
  }
  echo $_GET["callback"] . "(" . json_encode($results_array) . ")";
}
?>
```

## 10.3. WMOTU Invaders object-oriented

In order to avoid polluting the global namespace, everything has been packaged into the **game** object. This illustrates basic object-orientation in JavaScript.

[students.btsi.lu/evegi144/WAD/WMOTUInvadersOO](https://students.btsi.lu/evegi144/WAD/WMOTUInvadersOO)

### 10.3.1. index.html

```
<!DOCTYPE html>
<html lang=en>
```

```

<head>
  <title>WMOTU Invaders v1.0</title>
  <meta charset=UTF-8>
  <link href=index.css rel=stylesheet>
  <script src=index.js></script>
</head>
<body>
  <section id=splash>
    <h2>WMOTU Invaders v1.0</h2>
    <ul>
      <li>
        <button id=newGame>New Game</button>
      </li>
      <li>
        <button id=highScores>High Scores</button>
      </li>
      <li>
        <button id=keys>Keys</button>
      </li>
    </ul>
  </section>
  <section id=highScoreSection>
    <button id=OKScores style="position: fixed; top: 0;">OK</button>
    <table id=highScoreListTable>
      <caption>Hall of Fame</caption>
      <thead>
        <tr>
          <th>Rank</th>
          <th>Player</th>
          <th>Score</th>
          <th>Level</th>
        </tr>
      </thead>
    </table>
  </section>
  <section id=keysSection>
    <p>Use left and right cursor keys to move your spaceship and the space bar to fire.

    The game can be paused at any time by pressing <code>P</code>. Pressing <code>P</code> again will resume.
    You can exit the game early using <code>Esc</code>.</p>
    <button id=OKKeys>OK</button>
  </section>
  <section id=board>
    <p>
      Score: <span id=score></span>&nbsp;
      Level: <span id=level></span>
    </p>
    <canvas id=canvas width=600 height=400>This browser does not run this game
    (canvas
      support missing).
  </section>

```

```
    </canvas>
    <section>
      <img id=life1 src=playerspaceship36x46.png alt=playerspaceship36x46.png>
      <img id=life2 src=playerspaceship36x46.png alt=playerspaceship36x46.png>
    </section>
  </section>
</body>
</html>
```

### 10.3.2. index.js

```
"use strict";

const init = () => {
  const game = {
    canvas: document.getElementById('canvas'),
    context: this.canvas.getContext('2d'),
    imagePlayer: new Image(36, 46),
    imageShot: new Image(12, 23),
    imageAlien: new Image(42, 27),
    imageAlienBomb: new Image(10, 10),
    playerShotsX: [],
    playerShotsY: [],
    numAliens: 40,
    numAliensPerRow: 10,
    minFireThreshold: 300,
    initialPlayerSpeed: 20,
    initialNumOfLives: 2,
    aliensX: [],
    aliensY: [],
    alienBombsX: [],
    alienBombsY: [],
    fireButton: false,
    numShots: 1,
    pauseButton: false,
    invisible: false,
    moveLeft: false,
    moveRight: false,
    timeOfLastShot: 0,
    score: ~0,
    currLevel: 1,
    lastName: "",
    lastAnimationTime: 0,
    adjustmentFactor: 0.2,
    keys: {
      LEFT: 37,
      RIGHT: 39,
      ESC: 27,
      FIRE: 32,
      P: 80
    }
  }
}
```

```

    },

    //
    http://codeincomplete.com/posts/2013/12/4/javascript_game_foundations_the_game_loop
    timestamp() {
        return window.performance && window.performance.now ?
            window.performance.now() : new Date().getTime();
    },

    /* Called by newGame.
    * Draws aliens and adjusts game object attributes. */
    init() {
        let x = 1, y = 1;
        for (let i = 0; i < this.numAliens; i++) {
            this.aliensX[i] = x;
            this.aliensY[i] = y;
            this.context.drawImage(this.imageAlien, this.aliensX[i], this.aliensY[i]);
            x += this.imageAlien.width + 10;
            if (i % this.numAliensPerRow == 9) { // 10 aliens per row
                x = 1;
                y += this.imageAlien.height + 7;
            }
        }
        document.getElementById("score").innerHTML = ~(this.score >> 2);
        document.getElementById("level").innerHTML = this.currLevel;
        this.alienXSpeed = (5 + this.currLevel) * this.adjustmentFactor;
        this.alienYSpeed = this.currLevel;
        this.alienBombSpeed = Math.max(1, Math.floor(this.alienYSpeed / 3));
        this.alienDirection = 1; // aliens start moving to the right
        this.moveLeft = false; // no keys pressed
        this.moveRight = false;
        this.playerShotsX = []; // delete shots
        this.playerShotsY = [];
        this.alienBombsX = []; // delete bombs
        this.alienBombsY = [];
    },

    // Called by gameLoop.
    moveAliens() {
        this.leftBorderTouched = false;
        for (let i = 0; i < this.aliensX.length; i++) {
            // We need those to determine whether there is enough room to continue in the
            // current direction. If there isn't enough room, we need to change direction.
            if (this.aliensX[i] <= this.alienXSpeed && this.alienDirection === -1 ||
                this.aliensX[i] + this.imageAlien.width > this.canvas.width &&
                this.alienDirection === 1) {
                this.alienDirection = -this.alienDirection;
                // If we touched the left border, we need to move the aliens down.
                if (this.alienDirection === 1) this.leftBorderTouched = true;
                break; // Once we have changed direction, no need to check other aliens.
            }
        }
    }

```

```

    }
    if (this.leftBorderTouched) {
        this.alienYSpeed += 2 * this.adjustmentFactor;
        for (let i = 0; i < this.aliensY.length; i++) this.aliensY[i] +=
this.alienYSpeed;
    }
    // Move aliens horizontally.
    for (let i = 0; i < this.aliensX.length; i++) {
        this.aliensX[i] += this.alienDirection * this.alienXSpeed;
        this.context.drawImage(this.imageAlien, this.aliensX[i], this.aliensY[i]);
    }
},

// Called by gameLoop
moveShotsAndBombs() {
    for (let i = 0; i < this.playerShotsY.length; i++) {
        this.playerShotsY[i] -= this.imageShot.height * this.adjustmentFactor;
        if (this.playerShotsY[i] < 0) { // delete bullet as it has left space
            this.playerShotsX.splice(i, 1);
            this.playerShotsY.splice(i, 1);
            i--;
        }
        else this.context.drawImage(this.imageShot, this.playerShotsX[i],
            this.playerShotsY[i]);
    }
    for (let i = 0; i < this.alienBombsY.length; i++) {
        this.alienBombsY[i] += this.alienBombSpeed;
        if (this.alienBombsY[i] > this.canvas.height) {
            this.alienBombsX.splice(i, 1); // delete bomb as it has left space
            this.alienBombsY.splice(i, 1);
            i--;
        }
        else this.context.drawImage(this.imageAlienBomb, this.alienBombsX[i],
            this.alienBombsY[i]);
    }
},

// Called by gameLoop.
// Check whether a bullet has hit an alien or an alien/alien bomb touches the
player.
checkCollisions() {
    let shotLeft, shotRight, shotTop, shotBottom, alienLeft, alienRight, alienTop,
        alienBottom, playerRight, playerBottom, alienIndex = 0, alienBombLeft,
        alienBombRight, alienBombTop, alienBombBottom, alienBombIndex = 0;

    // Check whether alien has touched player or bullet has touched alien.
    playerRight = this.currPlayerX + this.imagePlayer.width;
    playerBottom = this.currPlayerY + this.imagePlayer.height;

    while (alienIndex < this.aliensX.length) {
        alienLeft = this.aliensX[alienIndex];

```

```

    alienRight = alienLeft + this.imageAlien.width;
    alienTop = this.aliensY[alienIndex];
    // If an alien manages to leave via the bottom of the frame, we lose a life.
    alienBottom = alienTop + this.imageAlien.height;
    if (alienRight >= this.currPlayerX && playerRight >= alienLeft && alienBottom
        >= this.currPlayerY || alienBottom > this.canvas.height) {
        this.loseLife = true;
        return;
    }
    let shotIndex = 0;
    // For each bullet check whether it touches the alien.
    while (shotIndex < this.playerShotsY.length) {
        shotLeft = this.playerShotsX[shotIndex];
        shotRight = shotLeft + this.imageShot.width;
        shotTop = this.playerShotsY[shotIndex];
        shotBottom = shotTop + this.imageShot.height;
        if (alienRight >= shotLeft && shotRight >= alienLeft && alienBottom >=
            shotTop && shotBottom >= alienTop) {
            this.playerShotsX.splice(shotIndex, 1);
            this.playerShotsY.splice(shotIndex, 1);
            this.aliensX.splice(alienIndex, 1);
            this.aliensY.splice(alienIndex, 1);
            alienIndex--;
            this.score = ~((~(this.score >> 2) + 1) << 2);
            document.getElementById("score").innerHTML = ~(this.score >> 2);
            break;
        }
        shotIndex++;
    }
    alienIndex++;
}

while (alienBombIndex < this.alienBombsX.length) { // Has bomb touched player?
    alienBombLeft = this.alienBombsX[alienBombIndex];
    alienBombRight = alienBombLeft + this.imageAlienBomb.width;
    alienBombTop = this.alienBombsY[alienBombIndex];
    alienBombBottom = alienBombTop + this.imageAlienBomb.height;
    if (alienBombRight >= this.currPlayerX && playerRight >= alienBombLeft &&
        alienBombBottom >= this.currPlayerY && alienBombTop <= playerBottom) {
        this.loseLife = true;
        return;
    }
    alienBombIndex++;
}

},

handleKeyDown(event) {
    if (event.keyCode === this.keys.P) {
        this.pauseButton = !this.pauseButton; // P pressed
        this.lastAnimationTime = this.timestamp();
    }
}

```

```

    else if (event.keyCode === this.keys.ESC) { // ESC
        this.loseLife = true;
        this.numOfLives = 0;
    }
    else if (event.keyCode === this.keys.FIRE) { // fire shot
        this.fireButton = true;
    }
    else if (event.keyCode === this.keys.LEFT) { // move left
        if (this.moveLeft) this.currPlayerSpeed += 2;
        this.moveLeft = true;
        this.moveRight = false;
    }
    else if (event.keyCode === this.keys.RIGHT) { // move right
        if (this.moveRight) this.currPlayerSpeed += 2;
        this.moveRight = true;
        this.moveLeft = false;
    }
},

handleKeyUp(event) {
    if (event.keyCode === this.keys.FIRE) { // fire shot
        this.fireButton = false;
    }
    else if (event.keyCode === this.keys.LEFT) { // move left
        this.moveLeft = false;
        this.currPlayerSpeed = this.initialPlayerSpeed;
    }
    else if (event.keyCode === this.keys.RIGHT) { // move right
        this.moveRight = false;
        this.currPlayerSpeed = this.initialPlayerSpeed;
    }
},

handleVisibilityChange() {
    if (this.invisible) this.lastAnimationTime = this.timestamp();
    this.invisible = !this.invisible;
},

restartLevel() {
    this.init();
    this.lastAnimationTime = 0;
    this.gameLoop(this.timestamp());
},

// This function controls the game. Called by newGame and requestAnimationFrame.
gameLoop(currTime) { // paused -> do nothing
    if (this.pauseButton) requestAnimationFrame(this.gameLoop.bind(this));
    else {
        this.timeElapsed = currTime - this.lastAnimationTime;
        if (this.lastAnimationTime === 0) this.adjustmentFactor = 0.2;
        else this.adjustmentFactor = this.timeElapsed / 100;
    }
}

```

```

this.context.clearRect(0, 0, this.canvas.width, this.canvas.height);
// If space key pressed and enough time has elapsed, we shoot again.
if (this.fireButton && ((currTime - this.timeOfLastShot) >
    this.minFireThreshold)) {
    this.timeOfLastShot = currTime;
    if (this.numShots === 3) {
        this.playerShotsX.push(this.currPlayerX);
        this.playerShotsY.push(this.currPlayerY);
        this.context.drawImage(this.imageShot, this.currPlayerX,
            this.currPlayerY);
        this.playerShotsX.push(this.currPlayerX + (this.imagePlayer.width -
            this.imageShot.width) / 2);
        this.playerShotsY.push(this.currPlayerY);
        this.context.drawImage(this.imageShot, this.currPlayerX + (
            this.imagePlayer.width - this.imageShot.width) / 2, this.currPlayerY);
        this.playerShotsX.push(this.currPlayerX + this.imagePlayer.width -
            this.imageShot.width);
        this.playerShotsY.push(this.currPlayerY);
        this.context.drawImage(this.imageShot, this.currPlayerX +
            this.imagePlayer.width - this.imageShot.width, this.currPlayerY);
    }
    else if (this.numShots === 2) {
        this.playerShotsX.push(this.currPlayerX);
        this.playerShotsY.push(this.currPlayerY);
        this.context.drawImage(this.imageShot, this.currPlayerX,
            this.currPlayerY);
        this.playerShotsX.push(this.currPlayerX + this.imagePlayer.width -
            this.imageShot.width);
        this.playerShotsY.push(this.currPlayerY);
        this.context.drawImage(this.imageShot, this.currPlayerX +
            this.imagePlayer.width - this.imageShot.width, this.currPlayerY);
    }
    else {
        this.playerShotsX.push(this.currPlayerX + (this.imagePlayer.width -
            this.imageShot.width) / 2);
        this.playerShotsY.push(this.currPlayerY);
        this.context.drawImage(this.imageShot, this.currPlayerX +
            (this.imagePlayer.width - this.imageShot.width) / 2, this.currPlayerY);
    }
}
if (this.moveLeft) { // If left arrow key pressed, move spaceship to the left.
    if (this.currPlayerX > this.currPlayerSpeed)
        this.currPlayerX -= this.currPlayerSpeed * this.adjustmentFactor;
    else this.currPlayerX = 1;
}
if (this.moveRight) { // If right arrow key pressed, move spaceship right.
    if ((this.currPlayerX + this.imagePlayer.width) < (this.canvas.width -
        this.currPlayerSpeed))
        this.currPlayerX += this.currPlayerSpeed * this.adjustmentFactor;
    else this.currPlayerX = this.canvas.width - this.imagePlayer.width - 1;
}

```

```

    for (let i = 0; i < this.aliensX.length; i++) { // generate bombs
        if (Math.random() > (1 - (this.currLevel *
            this.adjustmentFactor / 3000))) {
            this.alienBombsX.push(this.aliensX[i] + (this.imageAlien.width -
                this.imageAlienBomb.width) / 2);
            this.alienBombsY.push(this.aliensY[i] + this.imageAlien.height);
            this.context.drawImage(this.imageAlienBomb, this.aliensX[i] +
                (this.imageAlien.width - this.imageAlienBomb.width) / 2,
                this.aliensY[i] + this.imageAlien.height);
        }
    }
    this.checkCollisions();
    if (this.loseLife) { // If we have been hit or touched...
        if (this.numOfLives >= 1)
            document.getElementById("life" + this.numOfLives).style.display =
                "none";
        this.numOfLives--;
        this.loseLife = false;
        if (this.numOfLives < 0) this.gameOver();
        else this.restartLevel();
    }
    else if (this.aliensX.length === 0) this.nextLevel();
    else { // Move everything and continue the fun.
        this.alienXSpeed = (5 + this.currLevel) * this.adjustmentFactor;
        this.context.drawImage(this.imagePlayer, this.currPlayerX,
            this.currPlayerY);
        this.moveShotsAndBombs();
        this.moveAliens();
        this.lastAnimationTime = this.timestamp();
        requestAnimationFrame(this.gameLoop.bind(this));
    }
}
},

newGame() {
    document.getElementById("splash").style.display = "none";
    document.getElementById("board").style.display = "block";
    this.score = ~0;
    this.currLevel = 1;
    this.currPlayerSpeed = this.initialPlayerSpeed;
    // Display spare spaceships.
    for (let i = 1; i <= this.initialNumOfLives; i++)
        document.getElementById("life" + i).style.display = "inline";
    this.numOfLives = this.initialNumOfLives;
    this.loseLife = false;
    this.fireButton = false;
    this.currPlayerX = (this.canvas.width - this.imagePlayer.width) / 2;
    this.currPlayerY = this.canvas.height - this.imagePlayer.height - 5;
    window.onkeydown = this.handleKeyDown.bind(this);
    window.onkeyup = this.handleKeyUp.bind(this);
    document.addEventListener('visibilitychange',

```

```

        this.handleVisibilityChange.bind(this));
    this.numShots = 1;
    this.minFireThreshold = 300;
    this.pauseButton = false;
    this.invisible = false;
    this.init();
    this.lastAnimationTime = 0;
    this.gameLoop(this.timestamp());
},

nextLevel() {
    this.currLevel++;
    if (this.currLevel >= 25) this.numShots = 3;
    else if (this.currLevel >= 15) this.numShots = 2;
    if (this.currLevel >= 20) this.minFireThreshold = 100;
    else if (this.currLevel >= 10) this.minFireThreshold = 200;
    document.getElementById("level").innerHTML = this.currLevel;
    this.lastAnimationTime = 0;
    this.init();
    this.gameLoop(this.timestamp());
},

gameOver() {
    window.onkeydown = null;
    window.onkeyup = null;
    alert("Game over!");
    this.updateHighScores();
    this.displayHighScores();
},

displayHighScores() {
    if (this.highScoresReady) {
        document.getElementById("board").style.display = "none";
        document.getElementById("splash").style.display = "none";
        document.getElementById("highScoreSection").style.display = "block";
    }
    else setTimeout(this.displayHighScores.bind(this), 100);
},

hideHighScores() {
    this.showSplash();
},

readHighScores(e) { // callback for PHP
    this.highScoreList = JSON.parse(e.target.response);
    const HL = document.getElementById("highScoreListTable");
    if (HL) {
        const childNodes = HL.tBodies;
        let x = childNodes.length;
        while (x > 0) {
            HL.removeChild(childNodes[0]);

```

```

        x = childNodes.length;
    }
}
const body = document.createElement("tbody");
for (let i = this.highScoreList.length - 1; i >= 0; i--) {
    const row = body.insertRow(0);
    const cell1 = row.insertCell(0);
    const cell2 = row.insertCell(1);
    const cell3 = row.insertCell(2);
    const cell4 = row.insertCell(3);
    cell1.innerHTML = (i + 1);
    cell2.innerHTML = this.highScoreList[i].Player.slice(0, Math.min(30,
        this.highScoreList[i].Player.length));
    cell3.innerHTML = this.highScoreList[i].HighScore;
    cell4.innerHTML = this.highScoreList[i].Level;
    cell1.style.cssText = "text-align: right";
    cell3.style.cssText = "text-align: right";
    cell4.style.cssText = "text-align: right";
}
HL.appendChild(body);
this.highScoresReady = true;
},

loadSaveHighScores(action, player) { // action = "insert" or nothing
    const req = new XMLHttpRequest();
    const data = {
        action: action,
        player: player,
        score: ~this.score >> 2,
        level: this.currLevel
    };
    req.open('POST', 'index.php');
    req.addEventListener('load', this.readHighScores.bind(this));
    req.send(JSON.stringify(data));
    this.highScoresReady = false;
},

updateHighScores() {
    this.loadSaveHighScores();
    if (~(this.score >> 2) > 0) {
        this.lastName = prompt("Enter your name", this.lastName);
        this.loadSaveHighScores("insert", this.lastName);
    }
    if (this.lastName === null) this.lastName = "";
},

showSplash() {
    document.getElementById("splash").style.display = "block";
    document.getElementById("highScoreSection").style.display = "none";
    document.getElementById("keysSection").style.display = "none";
    document.getElementById("board").style.display = "none";
}

```

```

    },

    displayKeys() {
        document.getElementById("splash").style.display = "none";
        document.getElementById("highScoreSection").style.display = "none";
        document.getElementById("keysSection").style.display = "block";
        document.getElementById("board").style.display = "none";
    },

    hideKeys() {
        this.showSplash();
    }
};

game.imagePlayer.src = 'playerspaceship36x46.png';
game.imageShot.src = 'shot12x23.png';
game.imageAlien.src = 'alien42x27.png';
game.imageAlienBomb.src = 'alienbomb10x10.png';
game.loadSaveHighScores();
document.querySelector('#newGame').addEventListener('click',
game.newGame.bind(game));
document.querySelector('#highScores').addEventListener('click',
game.displayHighScores.bind(game));
document.querySelector('#keys').addEventListener('click', game.displayKeys);
document.querySelector('#OKKeys').addEventListener('click',
game.hideKeys.bind(game));
document.querySelector('#OKScores').addEventListener('click',
game.hideHighScores.bind(game));
};

addEventListener('load', init);

```

### 10.3.3. index.php

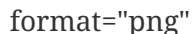
```

<?php
$obj = json_decode(file_get_contents('php://input'));
if (!$obj) exit;
require_once 'db_credentials.php';
@$db = new mysqli(DB_HOST, DB_USER, DB_PASSWORD, DB_NAME);
if (mysqli_connect_errno()) {
    error_log("Error: could not connect to database. Please try again later.");
    exit;
}

if (isset($obj->action)) {
    $sqlq = "INSERT INTO T1IF_Invaders VALUES (NULL, '" . $obj->player .
        "', '" . $obj->score . "', '" . $obj->level . "')";
    $result = $db->query($sqlq);
}
$sqlq = "SELECT * FROM T1IF_Invaders ORDER BY HighScore DESC";
$result = $db->query($sqlq);
while ($row = $result->fetch_assoc()) $results_array[] = $row;
echo json_encode($results_array);
?>

```

## 10.4. WMOTU Address Book

[https://foxi.ltam.lu/PROF/evegi144/T2IF2\\_WSERS/WAD/WMOTUAddressBookFoxy](https://foxi.ltam.lu/PROF/evegi144/T2IF2_WSERS/WAD/WMOTUAddressBookFoxy)

 format="png" alt="[WMOTUAddressBook1" data-bbox="80 520 600 535"/>

Logout	Add a new address	Current user: dummy1
Dummy1a	1a, Dummystreet	Dummyburg
Dummyland	Du-11111	Delete Edit
Dummy1b	1b, Dummystreet	Dummyburg
Dummyland	Du-11111	Delete Edit

WMOTU has been asked to develop a web app to manage an address book. The user needs to be able to add a new address, display all existing addresses as well as delete or edit them.

### 10.4.1. createDB.sql

The user probably does not want to reenter all the addresses each time he intends to use our address book, so we need to store them. A MySQL database is the ideal container for this type of

data. So let's create our database with a MySQL script with all the required instructions.

```
# createDB.sql -> create the database tables for the WMOTU Address Book
DROP TABLE IF EXISTS AB_addresses;
DROP TABLE IF EXISTS AB_users;

CREATE TABLE AB_users (id          INT UNSIGNED AUTO_INCREMENT NOT NULL UNIQUE,
                        user_name VARCHAR(32)                    NOT NULL UNIQUE,
                        password  VARCHAR(40)                    NOT NULL,
                        PRIMARY KEY (id))
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;
;

CREATE TABLE AB_addresses (id          INT UNSIGNED AUTO_INCREMENT NOT NULL UNIQUE,
                             user_id    INT UNSIGNED                NOT NULL,
                             name       VARCHAR(50)                 NOT NULL,
                             address1   VARCHAR(50)                 NOT NULL,
                             address2   VARCHAR(50),
                             city       VARCHAR(50)                 NOT NULL,
                             country    VARCHAR(50)                 NOT NULL,
                             post_code  VARCHAR(50)                 NOT NULL,
                             phone      VARCHAR(15),
                             mobile     VARCHAR(15),
                             fax        VARCHAR(15),
                             email      VARCHAR(50),
                             PRIMARY KEY (id),
                             FOREIGN KEY (user_id) REFERENCES AB_users (id)
                               ON DELETE CASCADE
                               ON UPDATE CASCADE)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;
;

INSERT INTO AB_users (user_name, password) VALUES ("dummy1", SHA("d1pw")),
("dummy2", SHA("d2pw"));
INSERT INTO AB_addresses (user_id, name, address1, city, country, post_code) VALUES
((SELECT
  id
  FROM AB_users
  WHERE user_name = "dummy1"), "Dummy1a", "1a, Dummystreet", "Dummyburg",
"Dummyland",
"Dum-11111"),
((SELECT
  id
  FROM AB_users
  WHERE user_name = "dummy1"), "Dummy1b", "1b, Dummystreet", "Dummyburg",
"Dummyland",
```

```
"Du-11111"),  
((SELECT  
    id  
    FROM AB_users  
    WHERE user_name = "dummy2"), "Dummy2", "2, Dummystreet", "Dummyburg", "Dummyland",  
    "Du-11112");
```

In order to create the database, we can either run this script in the MySQL command line or we use phpMyAdmin.

### Using MySQL command line

If you work on Foxi, open a SSH shell (using Putty for instance). On Windows, open a Windows command prompt in the project **restricted** folder and execute the following:

1. `mysql -u yourusername -p`
2. `use your_DB_name;`
3. `source createDB.sql;`

### Security

We create a folder **WMOTUAddressBookFoxi**, which will contain our app. This folder will be accessible to anyone on the Web. However, our SQL database creation script should not be visible to anyone, as it reveals the whole structure of our database. Therefore we'll put this file into a folder that we name **protected** and that is only accessible by ourselves. To do this, we will place a file named **.htaccess** with the following contents into this folder:

```
AuthType Basic  
AuthName "Restricted Files"  
AuthUserFile /full path/.htpasswd  
Require user username
```

Authentication type **Basic** sends the user name and password unencrypted. In a production environment we should therefore make sure that all traffic is encrypted using SSL. The Apache **mod\_ssl** module would thus need to be configured and enabled. Line 2 specifies the realm for which the authentication applies. In this case it applies to restricted files. Line 3 specifies the file where the encrypted password(s) is/are stored. Line 4 specifies which user(s) is/are authorized to access the restricted files. For further information about **.htaccess** files, see <http://httpd.apache.org/docs/current/howto/htaccess.html>.

Now we'll generate the **.htpasswd** file using the Apache tool **htpasswd**. Go to the **bin** directory of your Apache installation and execute the command `htpasswd -c .htpasswd xyz`, replacing **xyz** with your login name. The tool will ask for the password, which you'll have to enter twice. If the two passwords you've entered are identical, you'll now have a file, which contains the name of the user and his encrypted password. Move this file to the **protected** folder. Try to enter the restricted folder using your web browser. It should now ask for authentication.

Now the folder is protected against unauthorized access via the web, but we also need to protect it

from villains on the server itself. Therefore we need to remove all rights from other users, using the command `chmod -R o-rwx protected` in the main directory of our app, which contains the `protected` folder.

## Using phpMyAdmin

If you run the script for the first in phpMyAdmin, you need to comment out lines 2 and 3 that drop the tables, as the tables do not yet exist, otherwise phpMyAdmin will stop executing the script.

[foxi.ltam.lu/phpmyadmin](http://foxi.ltam.lu/phpmyadmin)

The screenshot shows the phpMyAdmin web interface. On the left sidebar, the database 'evegi144' is selected, showing two tables: 'AB\_addresses' and 'AB\_users'. Below the table list is a 'Create table' button. The main content area is titled 'Importing into the database "evegi144"'. It features a 'File to Import:' section with a text input for the file path (currently 'createDB.sql') and a 'Browse...' button. Below this is a 'Character set of the file:' dropdown menu set to 'utf-8'. The 'Partial Import:' section includes a checked checkbox for 'Allow the interruption of an import in case the script detects it is close to the PHP time' and a 'Number of rows to skip, starting from the first row:' input field set to '0'. The 'Format:' section has a dropdown menu set to 'SQL'. The 'Format-Specific Options:' section includes a 'SQL compatibility mode:' dropdown set to 'MYSQL40' and a checkbox for 'Do not use AUTO\_INCREMENT for zero values'. A 'Go' button is at the bottom.

If all went well, you now have two new tables in your database. You can check this using `desc AB_users;` and `desc AB_addresses;` in the MySQL command line. Let's take a closer look at what this script does. Line 1 is a comment to briefly explain the purpose of the script. Lines 2 and 3 drop any existing tables with these names in our DB. We do this in order to be able to use the script to recreate our tables in case we make modifications. Be careful: lines 2 and 3 delete the tables, including any data they contain! Lines 4-8 create table `AB_users`, which, as the name suggests, will

hold our user data. Lines 10-26 create table `AB_addresses`, which will hold our addresses. Lines 28-41 create dummy user and address data, so that we do not have to type those in manually.

By using a MySQL script we avoid having to retype the commands every time we make changes to our DB structure and we can easily look up the structure of our tables.

### 10.4.2. `index.php`

Our app should be usable by any number of users. Each user will be able to manage his own addresses. So we need a sign up and login system and this should be the first page anyone accessing our app sees, unless they are still logged in, in which case they should be taken automatically to the main page.

The HTTP protocol is stateless, meaning that when a new page is loaded, there is no information left from the previous page. This is not acceptable, as we cannot ask the user to log in again on each page of our app! Sessions allow us to save state information and use it across scripts. Details can be found in the Head First book and the usual online resources (particularly [php.net](http://php.net), cf. appendix [Resources](#)).

```
<?php
// Only send session id cookie over SSL.
ini_set('session.cookie_secure', true);
// Session IDs may only be passed via cookies, not appended to URL.
ini_set('session.use_only_cookies', true);
// Set the path for the cookie to the current directory in order to prevent it from
// being available to scripts in other directories.
ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
//if ($_SERVER['SERVER_PORT'] != 443)
//  header('Location: https://' . $_SERVER['SERVER_NAME'] .
$_SERVER['SCRIPT_NAME']);
if (!isset($_SERVER['HTTPS'])) // If SSL is not active, activate it.
  header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
dirname($_SERVER['PHP_SELF']));
// If no session is started yet, we'll start one.
if (!isset($_SESSION)) session_start();
// After 30 seconds we'll generate a new session ID to prevent a session fixation
// attack (cf. PHP cookbook p. 338).
if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
  session_regenerate_id();
  $_SESSION['generated'] = time();
}
// Include the database class needed to access the database.
require_once 'database.php';
?>
<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU Address Book</title>
    <meta charset=UTF-8>
    <link href=style.css rel=stylesheet>
```

```

</head>
<body>
  <main>
    <h1 id=logo class=threeD style="text-align: center">WMOTU Address Book</h1>
    <h2 style="text-align: center; font-size: 200%">Please login</h2>
    <form method=post id=loginForm>
      <div>
        <label>Username:</label>
        <input name=username required>
      </div>
      <div>
        <label>Password:</label>
        <input type=password name=password required>
      </div>
      <div id=loginInput>
        <input type=submit name=login value=Login>
        <input type=submit name=signup value="Sign Up">
      </div>
    </form>
  </main>
  <?php
    // If a user is already logged in, let him through to the main page.
    if (isset($_SESSION['user_id']))
      header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/main.php');
    // Else, if the user has submitted his login details, we need to check them.
    elseif (isset($_POST['login'])) {
      if (isset($_POST['username']) && isset($_POST['password'])) {
        $result = Database::login($_POST['username'], $_POST['password']);
        // If a user with this login exists, we load the main page.
        if ($result) {
          $_SESSION['user_id'] = $result;
          $user_name = $_POST['username'];
          $_SESSION['user_name'] = $user_name;
          header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
            dirname($_SERVER['PHP_SELF'])) . '/main.php';
        }
      }
    }
    // Else, if the user has signed up for a new account, we need to check if
    // such a user already exists.
    elseif (isset($_POST['signup'])) {
      if (isset($_POST['username']) && isset($_POST['password'])) {
        if (Database::get_user_id($_POST['username']))
          echo "User exists already!";
        else { // If not, we'll create the new user and if successful, we'll
          // forward to the main page.
          if ($result = Database::create_user($_POST['username'],
            $_POST['password']))
            {
              $_SESSION['user_id'] = $result;

```

```

        $_SESSION['user_name'] = $_POST['username'];
        header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
            dirname($_SERVER['PHP_SELF'])) . '/main.php';
    } else echo "Sign up failed!";
    }
}
}
?>
</body>
</html>

```

The user needs to be able to enter new or modify existing data. The ideal HTML element for this purpose is a `<form>` (lines 22-35).

### 10.4.3. db\_credentials.php

This file contains the credentials for accessing the DB. You need to replace these values with your own in order to access your DB. The `p:` in front of the hostname implies the use of a persistent DB connection (cf. [php.net/manual/en/mysqli.persistconns.php](http://php.net/manual/en/mysqli.persistconns.php)).

```

<?php
    Database::set_credentials('p:localhost', 'your user name', 'your password', 'your DB
name');
?>

```

### 10.4.4. database.php

The database class offers our app's DB interface. Thus all DB access takes place in one class and the rest of the app can reuse the functionality provided.

All the properties and methods of the database class are declared static, as they are not related to any specific object. This means we do not need to create a database object, we can simply use the class methods directly using `Database::method`.

```

<?php
    require_once 'db_credentials.php';

    class Database {
        private static $DB_HOST;
        private static $DB_USER;
        private static $DB_PASSWORD;
        private static $DB_NAME;
        private static $DB_USERS = 'AB_users';
        private static $DB_ADDRESSES = 'AB_addresses';

        static function set_credentials($db_host, $db_user, $db_password, $db_name) {
            self::$DB_HOST      = $db_host;
            self::$DB_USER      = $db_user;

```

```

        self::$DB_PASSWORD = $db_password;
        self::$DB_NAME      = $db_name;
    }

    static function connect() {
        $dbc = new mysqli(self::$DB_HOST, self::$DB_USER, self::$DB_PASSWORD,
            self::$DB_NAME);
        if ($dbc->connect_error) trigger_error('Database connection failed: ' .
            $dbc->connect_error, E_USER_ERROR);
        $dbc->set_charset("utf8");
        return $dbc;
    }

    // Returns the id if a user with the given name already exists,
    // otherwise FALSE.
    static function get_user_id($user_name) {
        $dbc = self::connect();
        // Look up the user id in the database
        $query = 'SELECT id FROM ' . self::$DB_USERS . ' WHERE user_name = ?';
        $stmt = $dbc->prepare($query);
        if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        $stmt->bind_param('s', $user_name);
        $stmt->execute();
        $stmt->store_result();
        if ($stmt->num_rows === 1) {
            $stmt->bind_result($result);
            $stmt->fetch();
        }
        else $result = FALSE;
        $stmt->close();
        $dbc->close();
        return $result;
    }

    // Returns FALSE if the user could not be created, otherwise the user id.
    static function create_user($user_name, $password) {
        $dbc = self::connect();
        $query = 'INSERT INTO ' . self::$DB_USERS .
            ' (user_name, password) VALUES (?, SHA1(?))';
        $stmt = $dbc->prepare($query);
        if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
            $dbc->error, E_USER_ERROR);
        $user_name = trim($user_name);
        $stmt->bind_param('ss', $user_name, $password);
        $result = $stmt->execute();
        $stmt->close();
        $dbc->close();
        if ($result) return self::get_user_id($user_name);
        else return $result;
    }
}

```

```

// Returns the user_id of the user or FALSE.
static function login($user_name, $password) {
    $dbc = self::connect();
    // Look up user id in database.
    $query = 'SELECT id FROM ' . self::$DB_USERS .
        ' WHERE user_name = ? AND password = SHA(?)';
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    $user_name = trim($user_name);
    $stmt->bind_param('ss', $user_name, $password);
    $stmt->execute();
    $stmt->store_result();
    if ($stmt->num_rows === 1) {
        $stmt->bind_result($result);
        $stmt->fetch();
    }
    else $result = FALSE;
    $stmt->close();
    $dbc->close();
    return $result;
}

// Search by address id OR user id.
// One of the two parameters should ALWAYS be FALSE!
// Returns an associative array or FALSE.
static function get_addresses($id = FALSE, $user_id = FALSE) {
    $dbc = self::connect();
    $result = FALSE;
    $query = 'SELECT * FROM ' . self::$DB_ADDRESSES . ' WHERE ' .
        ($id ? "id = $id" : "user_id = $user_id");
    $res = $dbc->query($query);
    if (!$res) trigger_error('Wrong SQL: ' . $query . ' Error: ' .
        $dbc->error, E_USER_ERROR);
    while ($dat = $res->fetch_array(MYSQLI_ASSOC)) $result[] = $dat;
    $res->free();
    $dbc->close();
    return $result;
}

// Returns FALSE if an address with this name and address1 already exists,
// otherwise TRUE.
static function insert_address($name, $address1, $address2, $city, $country,
    $post_code, $phone, $mobile, $fax, $email) {
    $dbc = self::connect();
    $query = 'SELECT id FROM ' . self::$DB_ADDRESSES .
        ' WHERE name = ? AND address1 = ?';
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' . $dbc->error,
        E_USER_ERROR);

```

```

$name      = trim($name);
$address1  = trim($address1);
$stmt->bind_param('ss', $name, $address1);
$result = $stmt->execute();
if ($result) {
    $stmt->store_result();
    if ($stmt->num_rows > 0) return FALSE;
    $stmt->close();
    $query = 'INSERT INTO ' . self::$DB_ADDRESSES . ' (name, user_id, address1,
        address2, city, country, post_code, phone, mobile, fax, email) VALUES
        (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?)';
    $stmt = $dbc->prepare($query);
    if (!$stmt) trigger_error('Wrong SQL: ' . $query . ' Error: ' . $dbc->error,
        E_USER_ERROR);
    $address2 = trim($address2);
    $city      = trim($city);
    $country   = trim($country);
    $post_code = trim($post_code);
    $phone     = trim($phone);
    $mobile    = trim($mobile);
    $fax       = trim($fax);
    $email     = trim($email);
    $stmt->bind_param('ssssssssss', $name, $_SESSION['user_id'], $address1,
        $address2, $city, $country, $post_code, $phone, $mobile, $fax, $email);
    $result = $stmt->execute();
    $stmt->close();
}
$dbc->close();
return $result;
}

```

// Returns TRUE if the update succeeded, otherwise false.

```

static function update_address($id, $name, $address1, $address2, $city, $country,
    $post_code, $phone, $mobile, $fax, $email) {
    $dbc      = self::connect();
    $id       = self::sanitize_string($dbc, $id);
    $name     = self::sanitize_string($dbc, $name);
    $address1 = self::sanitize_string($dbc, $address1);
    $address2 = self::sanitize_string($dbc, $address2);
    $city     = self::sanitize_string($dbc, $city);
    $country  = self::sanitize_string($dbc, $country);
    $post_code = self::sanitize_string($dbc, $post_code);
    $phone    = self::sanitize_string($dbc, $phone);
    $mobile   = self::sanitize_string($dbc, $mobile);
    $fax      = self::sanitize_string($dbc, $fax);
    $email    = self::sanitize_string($dbc, $email);
    $result   = mysqli_query($dbc, 'UPDATE ' . self::$DB_ADDRESSES . "
SET name='$name', address1='$address1', address2='$address2', city='$city',
    country='$country', post_code='$post_code', phone='$phone',
    mobile='$mobile', fax='$fax', email='$email' WHERE id=$id");
    mysqli_close($dbc);
}

```

```

        return $result;
    }

    // Returns TRUE if deletion succeeded, otherwise FALSE.
    static function delete_address($id) {
        $dbc    = self::connect();
        $id      = self::sanitize_string($dbc, $id);
        $result = mysqli_query($dbc, 'DELETE FROM ' . self::$DB_ADDRESSES .
            " WHERE id=$id");
        mysqli_close($dbc);
        return $result;
    }
}

?>

```

#### 10.4.5. `bouncer.php`

This script is used to make sure, that only logged in users can access the page. Everyone else is forwarded to the login and sign up page.

```

<?php
// Protect from session fixation via session adoption.
ini_set('session.use_strict_mode', true);
# Only send session id cookie over SSL.
ini_set('session.cookie_secure', true);
# Session IDs may only be passed via cookies, not appended to URL.
ini_set('session.use_only_cookies', true);
ini_set('session.cookie_httponly', true);
ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
if (!isset($_SERVER['HTTPS'])) {# If SSL is not active, activate it.
    header('Location: https://' . $_SERVER['HTTP_HOST'] . $_SERVER['PHP_SELF']);
    exit;
}
if (!isset($_SESSION)) session_start(); // Start session.
# After 30 seconds we'll generate a new session ID to prevent a session
# fixation attack (cf. PHP cookbook p. 338).
if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
    session_regenerate_id();
    $_SESSION['generated'] = time();
}
if (!isset($_SESSION['user_id'])) {// No user logged in -> go to the login page.
    header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php');
    exit;
}

?>

```

### 10.4.6. main.php

This is the main page, where the addresses of the current user are displayed.

```
<?php
    require_once 'database.php';
    require_once 'bouncer.php';
?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>WMOTU Address Book</title>
        <meta charset=UTF-8>
        <link href=style.css rel=stylesheet>
    </head>
    <body>
        <a href=logout.php>Logout</a>
        <a href=add.php>Add a new address</a>
        <?php
            echo "Current user: " . $_SESSION['user_name'] . "<br>";
            $addresses = Database::get_addresses(false, $_SESSION['user_id']);
            if ($addresses)
                foreach ($addresses as $address) {
                    echo '<article class=address>';
                    echo $address['name'] . "<br>";
                    echo $address['address1'] . "<br>";
                    if ($address['address2']) echo $address['name'] . "<br>";
                    echo $address['city'] . "<br>";
                    echo $address['country'] . "<br>";
                    echo $address['post_code'] . "<br>";
                    if ($address['phone']) echo $address['phone'] . "<br>";
                    if ($address['mobile']) echo $address['mobile'] . "<br>";
                    if ($address['fax']) echo $address['fax'] . "<br>";
                    if ($address['email']) echo $address['email'] . "<br>";
                    echo "<a href='delete.php?id=" . $address['id'] . "'>Delete</a>&nbsp;";
                    echo "<a href='edit.php?id=" . $address['id'] . "'>Edit</a>";
                    echo '</article>';
                }
        ?>
    </body>
</html>
```

### 10.4.7. logout.php

```

<?php
    if (!isset($_SESSION)) session_start(); # Start session if not done already.
    $_SESSION = []; # Empty session array.
    # If session cookie exists, kill it.
    if (session_id() != "" || isset($_COOKIE[session_name()]))
        setcookie(session_name(), '', 1, '/');
    session_destroy(); # Kill session.
    # Now it's time to return home.
    header('Location: https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php');
?>

```

#### 10.4.8. header.php

To increase the ease of maintenance of our app, we put the parts occurring several times into external scripts. Future changes will only have to be done in a single script.

```

<!DOCTYPE html>
<html lang=en>
    <head>
        <title>WMOTU Address Book</title>
        <meta charset=UTF-8>
        <link href=style.css rel=stylesheet>
    </head>
    <body>
        <a href=logout.php>Logout</a>
        <a href=main.php>Display addresses</a>

```

#### 10.4.9. footer.php

```

</body>
</html>

```

#### 10.4.10. add.php

```

<?php
    require_once 'database.php';
    require_once 'bouncer.php';
    include 'header.php';
?>
<form method=post>
    <label for=name>Name:</label>
    <input id=name name=name required><br>
    <label for=address1>Address line 1:</label>
    <input id=address1 name=address1 required><br>
    <label for=address2>Address line 2:</label>
    <input id=address2 name=address2><br>
    <label for=city>City:</label>
    <input id=city name=city required><br>
    <label for=country>Country:</label>
    <input id=country name=country required><br>
    <label for=post_code>Post code:</label>
    <input id=post_code name=post_code required><br>
    <label for=phone>Phone number:</label>
    <input id=phone name=phone><br>
    <label for=mobile>Mobile number:</label>
    <input id=mobile name=mobile><br>
    <label for=fax>Fax:</label>
    <input id=fax name=fax><br>
    <label for=email>Email:</label>
    <input type=email id=email name=email><br>
    <label></label>
    <input type=submit>
</form>
<?php
    if (isset($_POST['name'])) {
        if (Database::insert_address($_POST['name'], $_POST['address1'],
            $_POST['address2'], $_POST['city'], $_POST['country'], $_POST['post_code'],
            $_POST['phone'], $_POST['mobile'], $_POST['fax'], $_POST['email']))
        )
            header('Location: ' . 'https://' . $_SERVER['HTTP_HOST']
                . dirname(htmlspecialchars($_SERVER['PHP_SELF'])) . '/main.php');
        else echo 'Insert failed!';
    }
    include 'footer.php';
?>

```

#### 10.4.11. edit.php

```

<?php
    require_once 'database.php';
    require_once 'bouncer.php';
    include 'header.php';
?>

```

```

<?php
if (isset($_GET['id'])) {
    $result = Database::get_addresses($_GET['id']);
    if ($result) {
        $address = $result[0];
        echo <<<_END
        <form method=post action={$_SERVER['SCRIPT_NAME']}>
            <input type=hidden name=id value="{$_GET['id']}">
            <label for=name>Name:</label>
            <input id=name name=name required value="{ $address['name'] }"><br>
            <label for=address1>Address line 1:</label>
            <input id=address1 name=address1 required value="{ $address['address1'] }">
            <br>
            <label for=address2>Address line 2:</label>
            <input id=address2 name=address2 value="{ $address['address2'] }"><br>
            <label for=city>City:</label>
            <input id=city name=city required value="{ $address['city'] }"><br>
            <label for=country>Country:</label>
            <input id=country name=country required value="{ $address['country'] }"><br>
            <label for=post_code>Post code:</label>
            <input id=post_code name=post_code required
            value="{ $address['post_code'] }"><br>
            <label for=phone>Phone number:</label>
            <input id=phone name=phone value="{ $address['phone'] }"><br>
            <label for=mobile>Mobile number:</label>
            <input id=mobile name=mobile value="{ $address['mobile'] }"><br>
            <label for=fax>Fax:</label>
            <input id=fax name=fax value="{ $address['fax'] }"><br>
            <label for=email>Email:</label>
            <input id=email name=email value="{ $address['email'] }"><br>
            <label></label>
            <input type=submit>
        </form>
    _END;
    }
}

elseif (isset($_POST['name'])) {
    if (Database::update_address($_POST['id'], $_POST['name'], $_POST['address1'],
        $_POST['address2'], $_POST['city'], $_POST['country'], $_POST['post_code'],
        $_POST['phone'], $_POST['mobile'], $_POST['fax'], $_POST['email'])
    )
        header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
            dirname($_SERVER['PHP_SELF']) . '/main.php');
    else echo 'Update failed!';
}
include 'footer.php';
?>

```

### 10.4.12. delete.php

```
<?php
require_once 'database.php';
require_once 'bouncer.php';
if (isset($_GET['id'])) Database::delete_address($_GET['id']);
header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
    dirname($_SERVER['PHP_SELF']) . '/main.php');
?>
```

### 10.4.13. style.css

```
body {
    background-color: black;
    color:          gold;
}

#loginForm {
    width:          400px;
    margin-left:    auto;
    margin-right:   auto;
}

label {
    float:          left;
    clear:          left;
    width:          120px;
    text-align:     right;
    padding-right:  10px;
    margin-top:     10px;
}

input {
    margin-top: 10px;
}

#loginInput {
    width:          135px;
    margin-right:   auto;
    margin-left:    auto;
}

.threeD {
    font-family: 'Bookman Old Style', serif;
    line-height: 1em;
    color:       gold;
    font-weight: bold;
    font-size:   90px;
    text-shadow: 0px 0px 0 rgb(190, 215, 1), 1px 1px 0 rgb(176, 201, -13),
```

```

    2px 2px 0 rgb(161, 186, -28), 3px 3px 0 rgb(146, 171, -43),
    4px 4px 0 rgb(131, 156, -58), 5px 5px 0 rgb(117, 142, -72),
    6px 6px 0 rgb(102, 127, -87), 7px 7px 0 rgb(87, 112, -102),
    8px 8px 0 rgb(72, 97, -117), 9px 9px 0 rgb(58, 83, -131),
    10px 10px 0 rgb(43, 68, -146), 11px 11px 0 rgb(28, 53, -161),
    12px 12px 0 rgb(13, 38, -176), 13px 13px 12px rgba(0, 0, 0, 0.9),
    13px 13px 1px rgba(0, 0, 0, 0.5), 0px 0px 12px rgba(0, 0, 0, .2);
}

@keyframes logoRotate {
  50% {
    transform: rotateY(180deg);
    -webkit-transform: rotateY(180deg);
  }
  100% {
    transform: rotateY(0deg);
    -webkit-transform: rotateY(0deg);
  }
}

@-webkit-keyframes logoRotate {
  50% {
    transform: rotateY(180deg);
    -webkit-transform: rotateY(180deg);
  }
  100% {
    transform: rotateY(0deg);
    -webkit-transform: rotateY(0deg);
  }
}

#logo {
  animation: logoRotate 60s infinite;
  /* Safari and Chrome: */
  -webkit-animation: logoRotate 60s infinite;
}

form label {
  display: inline-block;
  width: 160px;
  font-weight: bold;
}

.address {
  border: 2px black ridge;
  margin-bottom: 10px;
}

a {
  outline: none;
  text-decoration: none;
}

```

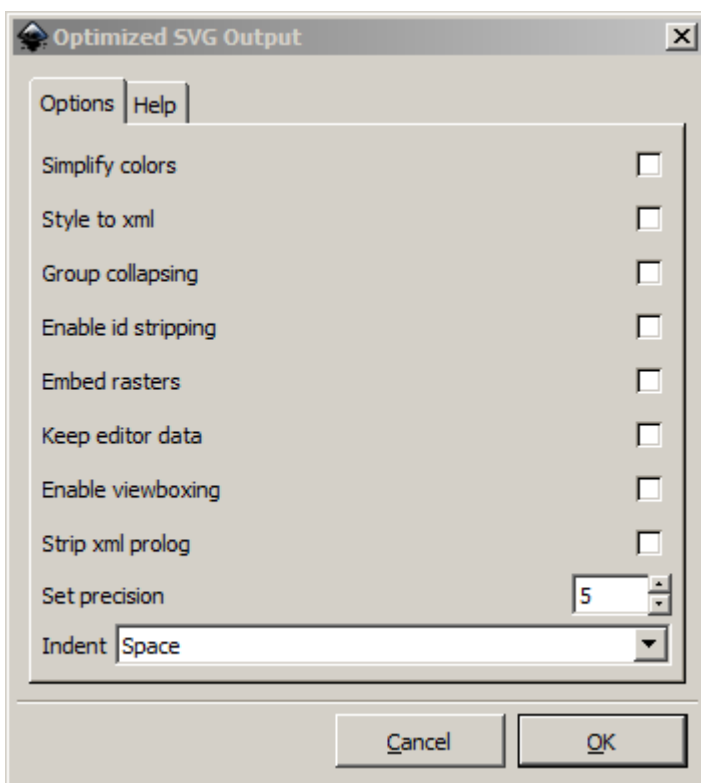
```
margin:      5px;
color:       lightblue;
}

a:hover {
  background-color: lightskyblue;
  color:           white;
}
```

## 10.5. WMOTU Tank

We will develop a tank game, where the player battles against computer controlled tanks of increasing intelligence on different playing fields.

The tank body and gun have been created with Inkscape and saved as optimized SVG using the following settings:



[students.btsi.lu/evegi144/WAD/WMOTUTank](http://students.btsi.lu/evegi144/WAD/WMOTUTank)

### 10.5.1. index.html

```

<!DOCTYPE html>
<html lang=en>
  <head>
    <title>WMOTU Tank v0.01a</title>
    <meta charset=utf-8>
    <script src=tank.js></script>
  </head>
  <body>
    <main>
      <canvas width=1000 height=700>This browser does not run this game
        (canvas support missing).
      </canvas>
    </main>
  </body>
</html>

```

### 10.5.2. tank.js

```

"use strict";

var game;

function init() {
  game = {
    canvasWidth: 1000,
    canvasHeight: 700,
    canvas: document.querySelector('canvas'),
    context: document.querySelector('canvas').getContext('2d'),
    tankSVG: undefined,
    gunSVG: undefined,
    player: undefined,
    bot1: undefined,
    pauseButton: false,
    invisible: false,
    rotateLeft: false,
    rotateRight: false,
    moveForward: false,
    moveBackward: false,
    rotateGunLeft: false,
    rotateGunRight: false,
    timeOfLastShot: 0,
    score: ~0,
    currLevel: 1,
    lastName: "",
    lastAnimationTime: 0,
    currPlayerSpeed: 5, // pixels per key event
    currPlayerRotationSpeed: 2, // degrees per key event
    currPlayerGunRotationSpeed: 4, // degrees per key event
    keys: {

```

```

LEFT: 37,
UP: 38,
RIGHT: 39,
DOWN: 40,
ESC: 27,
SPACE: 32,
B: 66,
N: 78,
P: 80
},

Tank: function (centerX, centerY, bodyColor, gunColor, turretColor) {
    this.centerX = centerX;
    this.centerY = centerY;
    this.bodyColor = bodyColor;
    this.gunColor = gunColor;
    this.turretColor = turretColor;
    this.angle = 33;
    this.gunAngle = 0; //200;
    this.tankImage = new Image();
    this.gunImage = new Image();
    this.lastAnimationTime = 0;
    game.tankSVG.getElementById('tankBody').style.fill = this.bodyColor;
    var svgAsString = new XMLSerializer().serializeToString(game.tankSVG);
    this.tankImage.src = 'data:image/svg+xml;base64,' + btoa(svgAsString);
    game.gunSVG.getElementById('gun').style.fill = this.gunColor;
    game.gunSVG.getElementById('turret').style.fill = this.turretColor;
    svgAsString = new XMLSerializer().serializeToString(game.gunSVG);
    this.gunImage.src = 'data:image/svg+xml;base64,' + btoa(svgAsString);
    this.draw();
},

Player: function (posX, posY, bodyColor, gunColor, turretColor) {
    game.Tank.call(this, posX, posY, bodyColor, gunColor, turretColor);
},

Bot: function (centerX, centerY, bodyColor, gunColor, turretColor) {
    game.Tank.call(this, centerX, centerY, bodyColor, gunColor, turretColor);
},

handleKeyDown: function (event) {
    if (event.keyCode === this.keys.P) {
        this.pauseButton = !this.pauseButton; // P pressed
        this.lastAnimationTime = this.timestamp();
    }
    else if (event.keyCode === this.keys.ESC) { // ESC
        this.loseLife = true;
        this.numOfLives = 0;
    }
    else if (event.keyCode === this.keys.SPACE) { // fire shot
        this.fireButton = true;
    }
}

```

```

    }
    else if (event.keyCode === this.keys.LEFT) { // move left
        this.rotateLeft = true;
    }
    else if (event.keyCode === this.keys.RIGHT) { // move right
        this.rotateRight = true;
    }
    else if (event.keyCode === this.keys.UP) { // move up
        this.moveForward = true;
    }
    else if (event.keyCode === this.keys.DOWN) { // move down
        this.moveBackward = true;
    }
    else if (event.keyCode === this.keys.B) { // move down
        this.rotateGunLeft = true;
    }
    else if (event.keyCode === this.keys.N) { // move down
        this.rotateGunRight = true;
    }
},

handleKeyUp: function (event) {
    if (event.keyCode === this.keys.FIRE) { // fire shot
        this.fireButton = false;
    }
    else if (event.keyCode === this.keys.LEFT) { // move left
        this.rotateLeft = false;
    }
    else if (event.keyCode === this.keys.RIGHT) { // move right
        this.rotateRight = false;
    }
    else if (event.keyCode === this.keys.UP) { // move up
        this.moveForward = false;
    }
    else if (event.keyCode === this.keys.DOWN) { // move down
        this.moveBackward = false;
    }
    else if (event.keyCode === this.keys.B) { // move down
        this.rotateGunLeft = false;
    }
    else if (event.keyCode === this.keys.N) { // move down
        this.rotateGunRight = false;
    }
},

handleVisibilityChange: function () {
    if (this.invisible) this.lastAnimationTime = this.timestamp();
    this.invisible = !this.invisible;
},

//

```

```

http://codeincomplete.com/posts/2013/12/4/javascript_game_foundations_the_game_loop
timestamp: function () {
    return window.performance && window.performance.now ?
        window.performance.now() : new Date().getTime();
},

init: function () {
    // Load the tank and gun SVGs asynchronously.
    var req1 = new XMLHttpRequest(), req2 = new XMLHttpRequest(), filesRemaining =
2;

    function r1() {
        this.tankSVG = req1.responseXML;
        if (--filesRemaining === 0) this.init2();
    }

    req1.onload = r1.bind(this);
    req1.open('get', 'tank_o.svg');
    req1.send();

    function r2() {
        this.gunSVG = req2.responseXML;
        filesRemaining--;
        if (filesRemaining === 0) this.init2();
    }

    req2.onload = r2.bind(this);
    req2.open('get', 'gun_o.svg');
    req2.send();
},

init2: function () {
    window.addEventListener('keydown', this.handleKeyDown.bind(this));
    window.addEventListener('keyup', this.handleKeyUp.bind(this));
    document.addEventListener('visibilitychange',
this.handleVisibilityChange.bind(this));
    this.clearCanvas();
    this.Player.prototype = Object.create(this.Tank.prototype);
    this.Bot.prototype = Object.create(this.Tank.prototype);
    this.player = new this.Player(100, 100, '#000000', '#222222', '#00ff00');
    //this.player.draw();
    this.bot1 = new this.Bot(200, 200, '#0000ff', '#2222ff', '#00ccff');
    this.bot1.angle = 0;
    this.bot1.gunAngle = 45;
    //this.bot1.draw();
    this.lastAnimationTime = this.timestamp();
    requestAnimationFrame(this.gameLoop.bind(this));
},

clearCanvas: function () {
    this.context.clearRect(0, 0, this.canvas.width, this.canvas.height);
}

```

```

    },

    gameLoop: function (currTime) { // paused -> do nothing
        if (this.pauseButton) requestAnimationFrame(this.gameLoop.bind(this));
        else {
            /*this.timeElapsed = currTime - this.lastAnimationTime;
            if (this.lastAnimationTime === 0) this.adjustmentFactor = 0.2;
            else this.adjustmentFactor = this.timeElapsed / 100;*/
            this.context.clearRect(0, 0, this.canvas.width, this.canvas.height);
            // If space key pressed and enough time has elapsed, we shoot again.
            /*if (this.fireButton && ((currTime - this.timeOfLastShot) >
this.minFireThreshold)) {
                this.timeOfLastShot = currTime;
            }
            if (this.moveLeft) { // If left arrow key pressed, move spaceship to the
left.
                if (this.currPlayerX > this.currPlayerSpeed)
                    this.currPlayerX -= this.currPlayerSpeed * this.adjustmentFactor;
                else this.currPlayerX = 1;
            }
            if (this.moveRight) { // If right arrow key pressed, move spaceship to the
right.
                if ((this.currPlayerX + this.imagePlayer.width) < (this.canvas.width -
this.currPlayerSpeed))
                    this.currPlayerX += this.currPlayerSpeed * this.adjustmentFactor;
                else this.currPlayerX = this.canvas.width - this.imagePlayer.width - 1;
            }*/
            if (this.rotateLeft) this.player.rotate(-this.currPlayerRotationSpeed);
            if (this.rotateRight) this.player.rotate(this.currPlayerRotationSpeed);
            if (this.moveForward) this.player.move(Math.round(Math.sin(this.player.angle *
Math.PI / 180) * this.currPlayerSpeed),
                Math.round(Math.cos(this.player.angle * Math.PI / 180) *
-this.currPlayerSpeed));
            if (this.moveBackward) this.player.move(Math.round(Math.sin(this.player.angle
*
                Math.PI / 180) * -this.currPlayerSpeed),
                Math.round(Math.cos(this.player.angle * Math.PI / 180) *
this.currPlayerSpeed));
            if (this.rotateGunLeft) this.player.rotateGun(-
this.currPlayerGunRotationSpeed);
            if (this.rotateGunRight)
this.player.rotateGun(this.currPlayerGunRotationSpeed);
            this.clearCanvas();
            this.player.draw();
            this.bot1.draw();
            this.lastAnimationTime = this.timestamp();
            requestAnimationFrame(this.gameLoop.bind(this));
        }
    }
};

```

```

game.Tank.prototype = {
    width: 44, // Tank width in pixels.
    height: 52, // Tank height in pixels.
    gunWidth: 24, // Gun width in pixels.
    gunHeight: 52, // Gun height in pixels.
    constructor: game.Tank,
    draw: function () { // Draw tank and gun at the current position and angles.
        game.context.save();
        // Draw a black border around canvas.
        game.context.strokeStyle = '#000';
        game.context.strokeRect(0, 0, game.canvas.width, game.canvas.height);
        game.context.translate(this.centerX + this.width / 2,
            this.centerY + this.height / 2);
        game.context.save();
        game.context.rotate((this.angle * Math.PI) / 180);
        game.context.drawImage(this.tankImage, -this.width / 2, -this.height / 2);
        game.context.restore();
        game.context.rotate((this.gunAngle * Math.PI) / 180);
        game.context.drawImage(this.gunImage, -this.gunWidth / 2, -1 - (this.height +
            this.gunWidth) / 2);
        game.context.restore();
    },

    // Returns true if moving and/or rotating the gun by the given values would
    // cause a border collision.
    gunCheckBorderCollision: function (dX, dY, angle) {
        return false;
    },

    // Returns true if moving and/or rotating the player by the given values would
    // cause a border collision.
    tankCheckBorderCollision: function (dX, dY, angle) {
        //result.Y = (int)Math.Round( centerPoint.Y + distance * Math.Sin( angle ) );
        //result.X = (int)Math.Round( centerPoint.X + distance * Math.Cos( angle ) );
        var alpha = ((this.angle + angle) % 360) * Math.PI / 180;
        var cos = Math.cos(alpha), sin = Math.sin(alpha), tan = Math.tan(alpha);
        var w = this.width, h = this.height;
        var distance = Math.sqrt(Math.pow(w / 2, 2) + Math.pow(h / 2, 2));
        var centerX = this.centerX + this.width / 2;
        var centerY = this.centerY + this.height / 2;
        /*var topY = this.posY - diagonal * Math.cos(angle);
        var bottomY = this.posY + this.height + diagonal * Math.cos(angle);
        */
        var leftX = centerX - (sin * h + cos * w) / 2;
        var rightX = centerX + (sin * h + cos * w) / 2;
        /*console.log('topY: ' + topY);
        console.log('bottomY: ' + bottomY);

        console.log('centerX: ' + centerX);
        console.log('centerY: ' + centerY);
        console.log('leftX: ' + leftX);

```

```

        console.log('rightX: ' + rightX);*/
        var x = sin * w / 2;
        console.log('x: ' + x);
        return false;
    },

    move: function (dX, dY) {
        if (!this.tankCheckBorderCollision(dX, dY, 0)) {
            this.centerX += dX;
            this.centerY += dY;
        }
    },

    rotate: function (angle) {
        if (!this.tankCheckBorderCollision(0, 0, angle))
            this.angle = (this.angle + angle) % 360;
    },

    rotateGun: function (angle) {
        this.gunAngle = (this.gunAngle + angle) % 360;
    }
};

game.init();
}

addEventListener('load', init);

```

## 10.6. Web Note

[<a href="https://students.btsi.lu/evegi144/WAD/WebNote">" width="381.75" tmp="false">^\]](https://students.btsi.lu/evegi144/WAD/WebNote)

### 10.6.1. Requirements specification

We often come across information on the web that we would like to store in order to access it from anywhere at a later point in time. Web Note allows registered users to create timestamped notes with editable HTML content.

The following requirements need to be met:

- Secure user sign up, login, logout and password change.
- Timestamps and names of all user notes are displayed. Clicking on one of them will display the note name and content together with edit and delete functionality.
- A new note with name and HTML content can be added with the current timestamp generated automatically or the new note can be discarded.
- High speed app with no page reload except for login/logout.

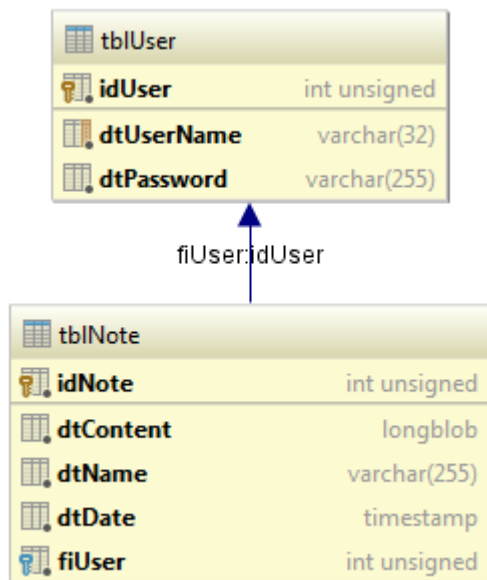
## 10.6.2. Analysis

This app consists of the following parts:

- A MySQL DB to store users and notes.
- A PHP backend to manage the DB operations.
- A PHP AJAX API, providing JavaScript access to the PHP backend.
- A HTML5/CSS3/JavaScript frontend.

## 10.6.3. Design and implementation

DB



[students.btsi.lu/evegi144/WAD/WebNote/createDB.sql](http://students.btsi.lu/evegi144/WAD/WebNote/createDB.sql)

```

DROP DATABASE IF EXISTS evegi144_dbWebNote;
CREATE DATABASE evegi144_dbWebNote
    DEFAULT CHARSET utf8
    DEFAULT COLLATE utf8_bin;
USE evegi144_dbWebNote;

CREATE TABLE tblUser (
    idUser      INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
    dtUserName  VARCHAR(32)  NOT NULL UNIQUE,
    dtPassword  VARCHAR(255) NOT NULL
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;

CREATE TABLE tblNote (
    idNote      INT UNSIGNED PRIMARY KEY AUTO_INCREMENT,
    dtContent   LONGBLOB     NOT NULL,
    dtName      VARCHAR(255) NOT NULL,
    dtDate      TIMESTAMP    NOT NULL    DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,
    fiUser      INT UNSIGNED NOT NULL,
    FOREIGN KEY (fiUser) REFERENCES tblUser (idUser)
        ON DELETE CASCADE
        ON UPDATE CASCADE
)
ENGINE = INNODB
DEFAULT CHARSET utf8
DEFAULT COLLATE utf8_bin;

```

## Backend

The backend provides a **Database** class with the following functionality:

- **set\_credentials**
- **connect**
- **get\_user\_name**
- **insert\_note**
- **update\_note**
- **delete\_note**
- **get\_note**
- **get\_notes**
- **change\_password**
- **login**
- **create\_user**

```

<?php
    require_once 'db_credentials.php';

    class Database {
        private static $DB_HOST;
        private static $DB_USER;
        private static $DB_PASSWORD;
        private static $DB_NAME;

        static function set_credentials($db_host, $db_user, $db_password, $db_name) {
            self::$DB_HOST      = $db_host;
            self::$DB_USER      = $db_user;
            self::$DB_PASSWORD = $db_password;
            self::$DB_NAME      = $db_name;
        }

        static function connect() {
            $dbc = @new mysqli(self::$DB_HOST, self::$DB_USER, self::$DB_PASSWORD,
self::$DB_NAME);
            if ($dbc->connect_error) die('Database connection failed: ' . $dbc-
>connect_error);
            $dbc->set_charset("utf8");
            return $dbc;
        }

        static function get_user_name($user_id) {
            $dbc = self::connect();
            $query = "SELECT dtUserName FROM tblUser WHERE idUser = $user_id";
            $result = $dbc->query($query) or die('Error reading from DB.' . $dbc->error);
            if ($result) return $result->fetch_row()[0];
            return false;
        }

        static function insert_note($note, $user_id) {
            $dbc = self::connect();
            $content = $note[0];
            $name = $note[1];
            $query = "INSERT INTO tblNote(dtName, dtContent, fiUser) VALUES ('$name',
'$content',
            $user_id)";
            $result = $dbc->query($query) or die('Error inserting into DB.' . $dbc->error);
        }

        static function update_note($note, $user_id) {
            $dbc = self::connect();
            $content = $note[0];
            $name = $note[1];
            $id = $note[2];
            $query = "UPDATE tblNote set dtName = '$name', dtContent = '$content' WHERE

```

```

        idNote = $id";
        $result = $dbc->query($query) or die('Error inserting into DB.' . $dbc->error);
    }

    static function delete_note($note_id) {
        $dbc = self::connect();
        $query = "DELETE FROM tblNote WHERE idNote = $note_id";
        $result = $dbc->query($query) or die('Error deleting DB.' . $dbc->error);
    }

    // Return array with all notes for user.
    static function get_notes($user_id) {
        $dbc = self::connect();
        $query = "SELECT idNote, dtName, dtContent, dtDate FROM
            tblNote WHERE fiUser = $user_id ORDER BY dtDate DESC";
        $result = $dbc->query($query) or die('Error reading from DB.' . $dbc->error);
        if ($result) {
            $all_rows = [];
            while ($row = $result->fetch_assoc()) $all_rows[] = $row;
            return $all_rows;
        }
        return false;
    }

    static function change_password($user_id, $old_pw, $new_pw) {
        $dbc = self::connect();
        $query = "SELECT dtPassword FROM tblUser";
        $result = $dbc->query($query) or trigger_error('Wrong SQL: ' . $query . ' Error:
        ' .
            $dbc->error, E_USER_ERROR);
        if ($result && $result->num_rows === 1) error_log($result->fetch_assoc()['dtPassword']);
        $query = "SELECT COUNT(*) FROM tblUser WHERE idUser = $user_id AND dtPassword =
        '"
            . password_hash($old_pw, PASSWORD_DEFAULT) . "'";
        error_log($query);
        $result = $dbc->query($query) or trigger_error('Wrong SQL: ' . $query . ' Error:
        ' .
            $dbc->error, E_USER_ERROR);
        $result = false;
        if ($result && $result->fetch_row()[0] === 1) {
            $query = 'UPDATE tblUser SET dtPassword = "' . password_hash($new_pw,
                PASSWORD_DEFAULT) . "'";
            error_log($query);
            $result = $dbc->query($query) or trigger_error('Wrong SQL: ' . $query . '
            Error: ' .
                $dbc->error, E_USER_ERROR);
        }
        $dbc->close();
        return $result;
    }
}

```

```

// Returns user id or FALSE.
static function login($user_name, $password) {
    $dbc = self::connect();
    $query = 'SELECT idUser, dtPassword FROM tblUser WHERE dtUserName = "' .
        $dbc->real_escape_string($user_name) . '"';
    $result = $dbc->query($query) or trigger_error('Wrong SQL: ' . $query . ' Error:
    ' .
        $dbc->error, E_USER_ERROR);
    if ($result && $result->num_rows === 1) {
        $res = $result->fetch_assoc();
        if (password_verify($password, $res['dtPassword'])) return $res['idUser'];
    }
    return false;
}

# Returns FALSE if user could not be created, otherwise user id.
static function create_user($user_name, $password) {
    $dbc = self::connect();
    $query = 'INSERT INTO tblUser (dtUserName, dtPassword) VALUES("' .
        $dbc->real_escape_string($user_name) . '", "' .
        password_hash($password, PASSWORD_DEFAULT) . '")';
    $result = $dbc->query($query);/* or trigger_error('Wrong SQL: ' . $query . '
Error: ' .
        $dbc->error, E_USER_ERROR);*/
    return $result;
}
}

?>

```

## API

The API provides the following self-explanatory functionality:

- `get_notes`
- `insert_note`
- `update_note`
- `delete_note`
- `logout`
- `change_password`

## API.php

```

<?php
// API to be used via HTTP POST requests.
require_once 'protected/bouncer.php';
require_once 'protected/Database.php';

if (isset($_POST['function'])) {
    if ($_POST['function'] === 'insert_note' && isset($_POST['parameter']))
        insert_note($_POST['parameter']);
    elseif ($_POST['function'] === 'update_note' && isset($_POST['parameter']))
        update_note($_POST['parameter']);
    elseif ($_POST['function'] === 'delete_note' && isset($_POST['parameter']))
        delete_note($_POST['parameter']);
    elseif ($_POST['function'] === 'logout') logout();
    elseif ($_POST['function'] === 'change_password' && isset($_POST['parameter'])) {
        $parms = json_decode($_POST['parameter']);
        change_password($parms[0], $parms[1]);
    }
    elseif ($_POST['function'] === 'get_notes') get_notes();
}

function get_notes() {
    echo json_encode(Database::get_notes($_SESSION['user_id']));
}

function insert_note($note) {
    Database::insert_note(json_decode($note), $_SESSION['user_id']);
}

function update_note($note) {
    Database::update_note(json_decode($note), $_SESSION['user_id']);
}

function delete_note($note_id) {
    Database::delete_note($note_id);
}

function logout() {
    require_once 'logout.php';
}

function change_password($old_pw, $new_pw) {
    if (Database::change_password($_SESSION['user_id'], $old_pw, $new_pw))
        echo 'Password changed successfully';
    else echo 'Password change failed';
}
?>

```

## Frontend

First we have the usual login, logout and bouncer scripts:

### index.php

```
<?php
    ini_set('session.cookie_secure', true);
    ini_set('session.use_only_cookies', true);
    ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
    if (!isset($_SERVER['HTTPS'])) // If SSL is not active, activate it.
        header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
dirname($_SERVER['PHP_SELF']));
    if (!isset($_SESSION)) session_start();
    if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
        session_regenerate_id();
        $_SESSION['generated'] = time();
    }
    if (isset($_SESSION['user_id']))
        header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/main.php');
    elseif (isset($_POST['username'], $_POST['password'])) {
        require_once 'protected/Database.php';
        $result = Database::login($_POST['username'], $_POST['password']);
        if ($result) {
            $_SESSION['user_id'] = $result;
            $_SESSION['user_name'] = $_POST['username'];
            header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
                dirname($_SERVER['PHP_SELF']) . '/main.php');
        }
    } elseif (isset($_POST['register'])) {
        require_once 'protected/Database.php';
        if (isset($_POST['username'], $_POST['pw1'], $_POST['pw2']) && $_POST['pw1'] ===
            $_POST['pw2'])
        )
            if (Database::create_user($_POST['username'], $_POST['pw1']))
                echo "<script>alert('Registration succeeded, please log in!');</script>";
            else echo "<script>alert('Registration failed!');</script>";
    }
?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Web Note</title>
        <meta charset=UTF-8>
        <meta name=viewport content="width=device-width, initial-scale=1">
        <link href=index.css rel=stylesheet>
    </head>
    <body>
        <main>
            <form method=post>
```

```
<input name=username placeholder="User name" required autofocus>
<input type=password name=password placeholder=Password required>
<button name=login>Log in</button>
</form>
<form method=post>
  <input name=username placeholder="User name" required>
  <input name=pw1 type=password placeholder=Password required>
  <input name=pw2 type=password placeholder="Repeat password" required>
  <button name=register>Register</button>
</form>
</main>
</body>
</html>
```

index.css

```
body {
  background: linear-gradient(to bottom right, yellow, #772222) fixed;
  text-shadow: 1px 1px 1px white;
  margin: 0;
}

main {
  position: absolute;
  top: 0;
  left: 0;
  right: 0;
  bottom: 0;
  display: flex;
}

form {
  width: 200px;
  margin: auto;
}

form > input, form > button {
  margin-top: 10px;
  text-shadow: 1px 1px 1px white;
  border-radius: 5px;
}

form > input {
  width: 196px;
  opacity: 0.5;
}

form > input:focus {
  background-color: yellow;
}

form > button {
  background: linear-gradient(to bottom right, yellow, red);
}

form > button:focus {
  border: 2px solid grey;
}

form > input::-moz-focus-inner {
  border: 0;
}
```

### logout.php

```
<?php
    if (!isset($_SESSION)) session_start();
    $_SESSION = [];
    if (session_id() != "" || isset($_COOKIE[session_name()])) setcookie(session_name(),
        '', 1, '/');
    session_destroy();
    header('Location: https://' . $_SERVER['HTTP_HOST'] .
        dirname($_SERVER['PHP_SELF']) . '/index.php');
?>
```

### bouncer.php

```
<?php
    # Only send session id cookie over SSL.
    ini_set('session.cookie_secure', true);
    # Session IDs may only be passed via cookies, not appended to URL.
    ini_set('session.use_only_cookies', true);
    ini_set('session.cookie_path', rawurlencode(dirname($_SERVER['PHP_SELF'])));
    if (!isset($_SERVER['HTTPS'])) { // # SSL is not active, activate it.
        header('Location: https://' . $_SERVER['HTTP_HOST'] .
            dirname($_SERVER['PHP_SELF']));
        exit;
    }
    if (!isset($_SESSION)) session_start();
    # After 30 seconds we'll generate a new session ID to prevent a session
    # fixation attack (cf. PHP cookbook p. 338).
    if (!isset($_SESSION['generated']) || $_SESSION['generated'] < (time() - 30)) {
        session_regenerate_id();
        $_SESSION['generated'] = time();
    }
    if (!isset($_SESSION['user_id'])) { // No user logged in -> go to the login page.
        header('Location: ' . 'https://' . $_SERVER['HTTP_HOST'] .
            dirname($_SERVER['PHP_SELF']) . '/index.php');
        exit;
    }
?>
```

The frontend defines an object `main`, with the following functionality:

- `init` attaches event listeners and gets all notes from the DB.
- `AJAXFunctionCall` is a helper function to call the API asynchronously via AJAX.
- `getNotesFromDB` retrieves the user's notes from the DB via the API and displays them.
- `showNote` displays a specific note.
- `deleteNote` deletes the currently shown note.
- `editNote` opens the editor for the currently shown note.

- `cancelEdit` closes the editor without saving.
- `saveNote` closes the editor and saves the note.
- `newNote` opens an empty editor.
- `insertNote` saves the new note.
- `discardNewNote` closes the editor without saving.
- `changePW` displays the password change form.

The editor component CKEditor ([ckeditor.com](http://ckeditor.com)) is used.

`main.php`

```

<?php
    require_once 'protected/bouncer.php';
    require_once 'protected/Database.php';
?>
<!DOCTYPE html>
<html lang=en>
    <head>
        <title>Web Note</title>
        <meta charset=UTF-8>
        <meta name=viewport content="width=device-width, initial-scale=1">
        <link href=main.css rel=stylesheet>
        <script src="//cdn.ckeditor.com/4.4.7/full/ckeditor.js"></script>
        <script src=main.js></script>
    </head>
    <body>
        <nav>
            <ul>
                <li><a>Change PW</a></li>
                <li><a>Logout <?php echo $_SESSION['user_name']; ?></a></li>
                <li><a>New note</a></li>
                <li><a>Refresh</a></li>
            </ul>
        </nav>
        <main>
            <nav><ol></ol></nav>
            <section id=showNoteSection hidden></section>
            <section id=newNoteSection hidden>
                <input id=name placeholder=Name>
                <button>Save</button>
                <button>Discard</button>
                <textarea id=newNoteTA></textarea>
            </section>
            <article id=changePWForm hidden>
                <input type=password name=curr_pw placeholder="Current password" required
                    autofocus autocomplete=off>
                <input pattern={8,} type=password name=new_pw1 placeholder="New password"
                    required>
                <input pattern={8,} type=password name=new_pw2 placeholder="New password"
                    required>
                <button>Change password</button>
            </article>
        </main>
    </body>
</html>

```

main.js

```
"use strict";
```

```

const main = { // main app object
  notes: undefined,
  shownNoteId: undefined,
  newNoteEditor: undefined,
  editNoteEditor: undefined,

  init: () => {
    const button = document.querySelector('#changePWForm > button');
    button.addEventListener('click', function (e) {
      // Create a new form and a new XMLHttpRequest.
      const inputs = document.querySelectorAll('#changePWForm > input');
      const currPWInput = inputs[0].value;
      const newPW1Input = inputs[1].value, newPW2Input = inputs[2].value;
      if (newPW1Input.length >= 8 && newPW2Input.length >= 8 && newPW1Input ===
newPW2Input) {
        main.AJAXFunctionCall('change_password', JSON.stringify([currPWInput,
newPW1Input]),
          function (e) {
            alert(e.target.response);
          });
        // Open the HTTP connection to the server script using POST method.
        document.getElementById('changePWForm').setAttribute('hidden', '');
      } else alert('Passwords don\'t match or are empty');
    });

    // Add top navigation event listeners.
    const lis = document.querySelectorAll('body > nav > ul > li');
    lis[0].addEventListener('click', main.changePW);
    lis[1].addEventListener('click', function () {
      window.location = 'logout.php';
    });
    lis[2].addEventListener('click', main.newNote);
    lis[3].addEventListener('click', main.getNotesFromDB);

    // Add newNoteSection button event listeners.
    const buttons = document.querySelectorAll('#newNoteSection > button');
    buttons[0].addEventListener('click', main.insertNote);
    buttons[1].addEventListener('click', main.discardNewNote);

    //window.addEventListener('beforeunload', main.unload);
    main.getNotesFromDB();
  },

  /*unload: function () {
    main.AJAXFunctionCall('logout');
  },*/

  AJAXFunctionCall: (functionName, parameter, callback) => {
    const req = new XMLHttpRequest();
    const data = new FormData();
    data.append('function', functionName);
  }
}

```

```

    data.append('parameter', parameter);
    req.open('POST', 'API.php');
    if (callback) req.addEventListener('load', callback);
    req.send(data);
  },

  getNoteWithId: id => {
    for (let i = 0; i < main.notes.length; i++)
      if (main.notes[i].idNote === id) return main.notes[i];
  },

  getNotesFromDB: () => {
    main.AJAXFunctionCall('get_notes', '', e => {
      main.notes = JSON.parse(e.target.response);

      // List note names.
      const ol = document.querySelector('main > nav > ol');
      ol.innerHTML = '';
      ol.style.margin = '0';
      for (let i = 0; i < main.notes.length; i++) {
        main.notes[i].idNote = parseInt(main.notes[i].idNote);
        main.notes[i].dtContent = decodeURIComponent(main.notes[i].dtContent);
        const li = document.createElement('li');
        const a = document.createElement('a');
        const t = main.notes[i].dtDate.split(/[- :]/);
        const d = new Date(t[0], t[1] - 1, t[2], parseInt(t[3]) -
          (new Date().getTimezoneOffset()) / 60 || 0, t[4] || 0, t[5] || 0, 0);
        const day = d.getDate() > 9 ? "" + d.getDate() : "0" + d.getDate();
        const month = (d.getMonth() + 1) > 9 ? "" + (d.getMonth() + 1) : "0" +
(d.getMonth() + 1);
        const hours = d.getUTCHours() > 9 ? "" + d.getUTCHours() : "0" +
d.getUTCHours();
        const minutes = d.getMinutes() > 9 ? "" + d.getMinutes() : "0" +
d.getMinutes();
        const seconds = d.getSeconds() > 9 ? "" + d.getSeconds() : "0" +
d.getSeconds();
        let s = day + '.' + month + '.' + d.getFullYear() + ' ' +
          hours + ':' + minutes + ':' + seconds;
        const l = 19 - s.length;
        for (let x = 0; x < l; x++) s += '&nbsp;';
        a.innerHTML = s + ' ' + main.notes[i].dtName;
        a.style.fontFamily = "Courier New, monospace";
        a.addEventListener('click', main.showNote(main.notes[i].idNote));
        li.appendChild(a);
        ol.appendChild(li);
      }
      if (main.shownNoteId !== undefined) main.showNote(main.shownNoteId);
    })
  },

  showNote: noteId => {

```

```

    if (main.notes) {
      const section = document.getElementById('showNoteSection');
      section.innerHTML = '';
      const nav = document.createElement('nav');
      let button = document.createElement('button');
      button.innerHTML = 'Edit';
      button.addEventListener('click', main.editNote);
      nav.appendChild(button);
      button = document.createElement('button');
      button.innerHTML = 'Delete';
      button.addEventListener('click', main.deleteNote);
      nav.appendChild(button);
      const span = document.createElement('span');
      const note = main.getNoteWithId(noteId);
      span.innerHTML = note.dtName;
      span.style.marginLeft = '10px';
      nav.appendChild(span);
      section.appendChild(nav);
      const p = document.createElement('p');
      p.style.cssText = 'margin: 0; overflow: auto;';
      section.style.cssText = 'overflow: hidden; display: flex; flex-flow: column;';
      p.innerHTML = note.dtContent;
      section.appendChild(p);
      section.removeAttribute('hidden');
      main.shownNoteId = noteId;
    }
  },

  deleteNote: () => {
    if (!confirm('Do you really want to delete this note?')) return;
    const section = document.getElementById('showNoteSection');
    const children = section.children;
    for (let i = 0; i < children.length; i++)
      if (children[i].tagName !== 'BUTTON') section.removeChild(children[i]);
    const id = main.shownNoteId;
    main.shownNoteId = undefined;
    main.AJAXFunctionCall('delete_note', id, main.getNotesFromDB);
    const p = section.querySelector('p');
    if (p) section.removeChild(p);
    const div = section.querySelector('div');
    if (div) section.removeChild(div);
    section.setAttribute('hidden', '');
  },

  editNote: () => {
    const section = document.getElementById('showNoteSection');
    const span = section.querySelector('nav > span');
    if (span) section.querySelector('nav').removeChild(span);
    const p = section.querySelector('p');
    const input = document.createElement('input');
    const note = main.getNoteWithId(main.shownNoteId);

```

```

    input.value = note.dtName;
    section.appendChild(input);
    main.editNoteEditor = CKEDITOR.appendTo('showNoteSection', {}, p.innerHTML);
    section.removeChild(p);
    let button = document.querySelector('#showNoteSection > nav > button');
    button.removeEventListener('click', main.editNote);
    button.innerHTML = 'Save';
    button.addEventListener('click', main.saveNote);
    button = document.createElement('button');
    button.innerHTML = 'Cancel';
    button.addEventListener('click', main.cancelEdit);
    section.querySelector('nav').appendChild(button);
  },

```

```

cancelEdit: () => {
  const section = document.getElementById('showNoteSection');
  const input = section.querySelector('input');
  if (input) section.removeChild(input);
  const p = section.querySelector('p');
  if (p) section.removeChild(p);
  const div = section.querySelector('div');
  if (div) section.removeChild(div);
  section.setAttribute('hidden', '');
  const buttons = section.querySelectorAll('button');
  if (buttons.length > 0) {
    section.querySelector('nav').removeChild(buttons[buttons.length - 1]);
    buttons[0].innerHTML = 'Edit';
    buttons[0].removeEventListener('click', main.saveNote);
    buttons[0].addEventListener('click', main.editNote);
  }
  main.editNoteEditor = undefined;
  if (main.shownNoteId !== undefined) main.showNote(main.shownNoteId);
},

```

```

saveNote: () => {
  main.editNoteEditor.updateElement();
  const section = document.getElementById('showNoteSection');
  const name = section.querySelector('input').value;
  if (name.length > 0) {
    main.AJAXFunctionCall('update_note',
      JSON.stringify([encodeURIComponent(main.editNoteEditor.getData()), name,
        main.shownNoteId]), main.getNotesFromDB);
    main.editNoteEditor.destroy();
    main.editNoteEditor = undefined;
  }
  else alert('Note must have a name');
},

```

```

newNote: () => {
  document.getElementById('showNoteSection').innerHTML = '';
  if (!main.newNoteEditor)

```

```

    main.newNoteEditor = CKEDITOR.replace('newNoteTA'/*, {filebrowserUploadUrl:
'upload.php'}*/);
    main.newNoteEditor.setData('');
    document.querySelector('#newNoteSection').removeAttribute('hidden');
    document.getElementById('name').focus();
},

insertNote: () => {
    main.newNoteEditor.updateElement();
    const name = document.getElementById('name').value;
    if (name.length > 0) {
        main.AJAXFunctionCall('insert_note',
            JSON.stringify([encodeURIComponent(main.newNoteEditor.getData()), name]),
            main.getNotesFromDB);
        main.discardNewNote();
    }
    else alert('Note must have a name');
},

discardNewNote: () => {
    document.querySelector('#newNoteSection').setAttribute('hidden', '');
    if (main.newNoteEditor) {
        main.newNoteEditor.destroy();
        main.newNoteEditor = undefined;
        document.getElementById('name').value = '';
    }
},

changePW: () => {
    if (document.getElementById('changePWForm').hasAttribute('hidden'))
        document.getElementById('changePWForm').removeAttribute('hidden');
    else document.getElementById('changePWForm').setAttribute('hidden', '');
}
};

addEventListener('load', main.init);

```

#### main.css

```

html, body {
    height: 100%;
    width: 100%;
    padding: 0;
    margin: 0;
}

body {
    background: linear-gradient(to bottom right, yellow, #772222) fixed;
    text-shadow: 1px 1px 1px white;
    display: flex;

```

```
flex-flow: column;
overflow: hidden;
}

body > main > nav {
  overflow: auto;
}

body > nav > ul {
  padding: 10px 0;
}

body > nav > ul > li {
  margin: 0;
  padding: 0;
  display: inline;
}

body > nav > ul > li > a {
  padding: 10px;
  border: outset yellow;
  background-color: yellow;
  color: blue;
  box-shadow: 3px 3px 3px black;
}

body > nav > ul > li > a:visited {
  color: blue;
}

body > nav > ul > li > a[active] {
  background-color: lightgreen;
  border: inset yellow;
}

main {
  margin-top: 10px;
  display: flex;
  flex: auto;
  overflow: hidden;
}

main > nav, main > #showNoteSection {
  display: flex;
  flex: auto;
  padding: 5px;
}

main > #newNoteSection {
  clear: left;
}
```

```
main > nav > ol {
  list-style-type: none;
  padding: 0;
}

ul {
  margin: 0;
  padding: 0;
}

a {
  text-decoration: none;
  /* stackoverflow.com/questions/826782/css-rule-to-disable-text-selection-
highlighting */
  -webkit-touch-callout: none;
  -webkit-user-select: none;
  -khtml-user-select: none;
  -moz-user-select: none;
  -ms-user-select: none;
  /*user-select: none;*/
}

a:hover {
  background: linear-gradient(to bottom right, yellow, red);
  cursor: default;
}

#changePWForm {
  position: fixed;
  top: 100px;
  left: 100px;
  padding: 10px;
  background-color: black;
}

#changePWForm > input:valid {
  background-color: lawngreen;
}

#changePWForm > input:invalid {
  background-color: red;
}
```

# Chapter 11. Art Gallery

This chapter shows the works of the greatest WAD artists that attended my courses. **You can download some of the apps as a single [zip](#) archive.**

## 11.1. Bill Brinck

### 11.1.1. Publi

```
<a href="http://publi.lu">" tmp="false">^]
```

## 11.2. Marco Campana

### 11.2.1. Pacman

```
<a href="https://students.btsi.lu/evegi144/WAD/ArtGallery/MarcoCampana/Pacman/index.html">"
tmp="false">^]
```

## 11.3. Nicolas Detombes

### 11.3.1. Dodge The Asteroids

```
<a href="https://students.btsi.lu/evegi144/WAD/ArtGallery/NicolasDetombes/DodgeTheAsteroids/Dodge
TheAsteroids.html">" tmp="false">^]
```

This game, developed by Nicolas Detombes, initially as the final problem solution for module CLISS2 2012 and later on extended with additional functionality, proves convincingly that canvas animation can be done in real time without the usage of prefabricated images. All animated objects are created programmatically in real time. Advanced array usage, program structuring using functions, canvas programming, key event handling and collision detection are beautifully illustrated.

### 11.3.2. Angry Chickens

```
<a href="https://students.btsi.lu/evegi144/WAD/ArtGallery/NicolasDetombes/AngryChickens/AngryChic
kens.html">" tmp="false">^]
```

Another game developed by Nicolas Detombes that illustrates mouse handling and DOM manipulation in addition to many features of Dodge The Asteroids.

### 11.3.3. Square Invasion

<a

href="https://students.btsi.lu/evegi144/WAD/ArtGallery/NicolasDetombes/SquareInvasion/SquareInvasion.html">" tmp="false">^] The third game developed by Nicolas Detombes, short and sweet.

## 11.4. Aleksandar Veselinovic and Steve Weiwert

### 11.4.1. TrèsBien

These artists do not yet fully master the French language, but their JavaScript skills are impressive.

<a

href="https://students.btsi.lu/evegi144/WAD/ArtGallery/AleksandarVeselinovicandSteveWeiwert/index.html">" tmp="false">^]

# Chapter 12. Resources

## 12.1. Books

<a href="http://frontendmasters.com/books">frontendmasters.com/books</a>
<a href="http://www.freetechbooks.com">www.freetechbooks.com</a>
<a href="http://cnx.org">cnx.org</a>

## 12.2. Editors, IDEs and tools

Atom	<a href="http://atom.io">atom.io</a>
Cloud & desktop IDE	<a href="http://www.theia-ide.org">www.theia-ide.org</a>
One-Click Online IDE for GitHub	<a href="http://gitpod.io">gitpod.io</a>
List of online IDEs	<a href="http://ide.ceriously.com">ide.ceriously.com</a>
Syntax highlighters	<a href="https://webdesign.tutsplus.com/articles/25-syntax-highlighters-tried-and-tested&amp;#8212;&amp;#8203;cms-23931">webdesign.tutsplus.com/articles/25-syntax-highlighters-tried-and-tested&amp;#8212;&amp;#8203;cms-23931</a>
Online UML editor	<a href="http://www.umletino.com">www.umletino.com</a>

## 12.3. HTML5

### 12.3.1. Web sites

The official working draft	<a href="http://www.w3.org/TR/html51">www.w3.org/TR/html51</a>
HTML: The Living Standard	<a href="http://developers.whatwg.org">developers.whatwg.org</a>
w3schools.com	<a href="http://w3schools.com/html/html5_intro.asp">w3schools.com/html/html5_intro.asp</a>
WebPlatform.org	<a href="http://docs.webplatform.org/wiki/html">docs.webplatform.org/wiki/html</a>
HTML-Seminar (German)	<a href="http://www.html-seminar.de">www.html-seminar.de</a>
HTML Dog	<a href="http://htmldog.com">htmldog.com</a>
A Practical Guide to HTML & CSS	<a href="http://learn.shayhowe.com">learn.shayhowe.com</a>
Quackit	<a href="http://www.quackit.com">www.quackit.com</a>
HTML5 Code	<a href="http://www.html5code.nl">www.html5code.nl</a>
Smashing Magazine	<a href="http://coding.smashingmagazine.com">coding.smashingmagazine.com</a>
Web Content Accessibility Guidelines	<a href="http://www.w3.org/WAI/intro/wcag.php">www.w3.org/WAI/intro/wcag.php</a>

### 12.3.2. Books

Head First HTML and CSS, 2nd Edition	<a href="http://shop.oreilly.com/product/9780596159924.do">shop.oreilly.com/product/9780596159924.do</a>
--------------------------------------	--

## 12.4. CSS3

### 12.4.1. Web sites

W3C Cascading Style Sheets home page	<a href="http://www.w3.org/Style/CSS">www.w3.org/Style/CSS</a>
w3schools.com	<a href="http://w3schools.com/css/css3_intro.asp">w3schools.com/css/css3_intro.asp</a>
Mozilla Developer Network CSS Reference	<a href="http://developer.mozilla.org/en-US/docs/Web/CSS/Reference">developer.mozilla.org/en-US/docs/Web/CSS/Reference</a>
CSS Portal	<a href="http://www.cssportal.com">www.cssportal.com</a>
CSS3.info	<a href="http://www.css3.info">www.css3.info</a>
CSS-Tricks	<a href="http://css-tricks.com">css-tricks.com</a>
CSSDeck	<a href="http://cssdeck.com">cssdeck.com</a>
Codrops	<a href="http://tympanus.net/codrops">tympanus.net/codrops</a>
Web Animation Resources	<a href="http://www.benjystanton.co.uk/projects/web-animation-resources">www.benjystanton.co.uk/projects/web-animation-resources</a>
CSS3 & HTML5 (German)	<a href="http://css3-html5.de/html-css-lernen">css3-html5.de/html-css-lernen</a>
HTML-Seminar (German)	<a href="http://www.html-seminar.de">www.html-seminar.de</a>
The Future Of CSS: Experimental CSS Properties	<a href="http://coding.smashingmagazine.com/2011/05/11/the-future-of-css-experimental-css-properties">coding.smashingmagazine.com/2011/05/11/the-future-of-css-experimental-css-properties</a>
100 Awesome CSS/Javascript Plugins and Coding Techniques	<a href="http://www.topdesignmag.com/100-awesome-cssjavascript-plugins-and-coding-techniques">www.topdesignmag.com/100-awesome-cssjavascript-plugins-and-coding-techniques</a>
Nifty Modal Window Effects	<a href="http://tympanus.net/Development/ModalWindowEffects">tympanus.net/Development/ModalWindowEffects</a>
CSS3Gen	<a href="http://css3gen.com">css3gen.com</a>
CSS3 Patterns Gallery	<a href="http://lea.verou.me/css3patterns">lea.verou.me/css3patterns</a>
Zen BG	<a href="http://mudcu.be/bg">mudcu.be/bg</a>
The Star Wars Intro in Pure CSS3	<a href="http://www.htmlgoodies.com/html5/client/the-star-wars-intro-in-pure-css3-thanks-to-3d-transform-animations.html#fbid=D79H_1jCDzk">www.htmlgoodies.com/html5/client/the-star-wars-intro-in-pure-css3-thanks-to-3d-transform-animations.html#fbid=D79H_1jCDzk</a>
CSS3, please!	<a href="http://css3please.com">css3please.com</a>
CSS3 Generator	<a href="http://css3generator.com">css3generator.com</a>
HTML Dog	<a href="http://htmldog.com">htmldog.com</a>

### 12.4.2. Books

## 12.5. JavaScript

### 12.5.1. Web sites

CLISS site of Robert Fisch	<a href="http://ltam.fisch.lu/index.php?scan=CLISS1.T1IF">ltam.fisch.lu/index.php?scan=CLISS1.T1IF</a>
CLISS site of Laurent Haan	<a href="http://www.haan.lu/#cliss">www.haan.lu/#cliss</a>
JavaScript Tutorial	<a href="http://learn.javascript.info">learn.javascript.info</a>
JS Fiddle	<a href="http://www.jsfiddle.net">www.jsfiddle.net</a>
JS Utility	<a href="http://jsutility.pjoneil.net">jsutility.pjoneil.net</a>
hilite.me	<a href="http://hilite.me">hilite.me</a>
Mozilla Developer Network	<a href="http://developer.mozilla.org/en-US/docs/JavaScript">developer.mozilla.org/en-US/docs/JavaScript</a>
JavaScript-Garden	<a href="http://bonsaiden.github.io/JavaScript-Garden">bonsaiden.github.io/JavaScript-Garden</a>
The Code Player	<a href="http://thecodeplayer.com">thecodeplayer.com</a>
Web Education Community	<a href="http://www.w3.org/community/webed/wiki/Main_Page">www.w3.org/community/webed/wiki/Main_Page</a>
Web Platform Docs	<a href="http://docs.webplatform.org/wiki/javascript/tutorials">docs.webplatform.org/wiki/javascript/tutorials</a>
Codecademy	<a href="http://www.codecademy.com">www.codecademy.com</a>
Eloquent JavaScript	<a href="http://eloquentjavascript.net">eloquentjavascript.net</a>
SELFHTML	<a href="http://de.selfhtml.org/javascript/index.htm">de.selfhtml.org/javascript/index.htm</a>
Dr. Web	<a href="http://www.drweb.de/magazin/category/javascript">www.drweb.de/magazin/category/javascript</a>
JavaScript Forum	<a href="http://forum.jswelt.de/javascript">forum.jswelt.de/javascript</a>
HTML Goodies	<a href="http://www.htmlgoodies.com">www.htmlgoodies.com</a>
The Complete Guide to Building HTML5 Games with Canvas & SVG	<a href="http://www.htmlgoodies.com/html5/client/the-complete-guide-to-building-html5-games-with-canvas-svg.html#fbid=Xq5JJBKUCda">www.htmlgoodies.com/html5/client/the-complete-guide-to-building-html5-games-with-canvas-svg.html#fbid=Xq5JJBKUCda</a>
How to write a small game using HTML5 and JavaScript – BrikBrok	<a href="http://blogs.msdn.com/b/eternalcoding/archive/2011/09/06/write-a-small-game-using-html5-and-javascript-brikbrok.aspx">blogs.msdn.com/b/eternalcoding/archive/2011/09/06/write-a-small-game-using-html5-and-javascript-brikbrok.aspx</a>
HTML5 Game Development	<a href="http://www.udacity.com/course/cs255">www.udacity.com/course/cs255</a>
WebPlatform.org	<a href="http://www.webplatform.org">www.webplatform.org</a>
HTML5 Canvas Tutorials	<a href="http://www.html5canvastutorials.com">www.html5canvastutorials.com</a>
Code inComplete	<a href="http://codeincomplete.com/archive">codeincomplete.com/archive</a>
Dive into HTML5	<a href="http://diveintohtml5.info">diveintohtml5.info</a>
HTML5 Demos and Examples	<a href="http://html5demos.com">html5demos.com</a>
OPENCLASSROOMS	<a href="http://fr.openclassrooms.com">fr.openclassrooms.com</a>
tutorialspoint	<a href="http://www.tutorialspoint.com">www.tutorialspoint.com</a>
JavaScript Kit	<a href="http://www.javascriptkit.com">www.javascriptkit.com</a>

HTML5 rocks	<a href="http://www.html5rocks.com">www.html5rocks.com</a>
Script Tutorials	<a href="http://www.script-tutorials.com">www.script-tutorials.com</a>
Javascript Development Tools Resources	<a href="https://clausreinke.github.io/js-tools/resources.html">clausreinke.github.io/js-tools/resources.html</a>
Coding Math	<a href="https://www.youtube.com/watch?v=zm9bqSSiIdo">www.youtube.com/watch?v=zm9bqSSiIdo</a>
Creative JavaScript	<a href="http://creativejs.com">creativejs.com</a>
w3schools how tos	<a href="http://www.w3schools.com/howto">www.w3schools.com/howto</a>
Learn to code with interactive challenges	<a href="http://edabit.com">edabit.com</a>
jsComplete	<a href="http://jscomplete.com">jscomplete.com</a>

### 12.5.2. Books

JSbooks	<a href="http://jsbooks.revolunet.com">jsbooks.revolunet.com</a>
Learning JavaScript Design Patterns	<a href="http://addyosmani.com/resources/essentialjsdesignpatterns/book">addyosmani.com/resources/essentialjsdesignpatterns/book</a>
19 Free Javascript Ebooks & Resources	<a href="http://www.getfreeebooks.com/19-free-javascript-ebooks-resources">www.getfreeebooks.com/19-free-javascript-ebooks-resources</a>
Building Front-End Web Apps with Plain JavaScript	<a href="http://oxygen.informatik.tu-cottbus.de/webeng/JsFrontendApp/book">oxygen.informatik.tu-cottbus.de/webeng/JsFrontendApp/book</a>
Eloquent JavaScript	<a href="http://eloquentjavascript.net">eloquentjavascript.net</a>
HTML5 for Masterminds	<a href="http://www.minkbooks.com">www.minkbooks.com</a>
JavaScript The Definitive Guide	<a href="http://shop.oreilly.com/product/9780596805531.do">shop.oreilly.com/product/9780596805531.do</a>
Learning Three.js: The JavaScript 3D Library for WebGL - Second Edition	<a href="https://www.packtpub.com/web-development/learning-threejs-javascript-3d-library-webgl-second-edition">https://www.packtpub.com/web-development/learning-threejs-javascript-3d-library-webgl-second-edition</a>
Three.js Essentials	<a href="http://www.packtpub.com/web-development/threejs-essentials">www.packtpub.com/web-development/threejs-essentials</a>
WebGL Beginner's Guide	<a href="http://www.packtpub.com/game-development/webgl-beginners-guide">www.packtpub.com/game-development/webgl-beginners-guide</a>
WebGL Game Development	<a href="http://www.packtpub.com/game-development/webgl-game-development">www.packtpub.com/game-development/webgl-game-development</a>
HTML5 Games: Creating Fun with HTML5, CSS3, and WebGL	<a href="http://eu.wiley.com/WileyCDA/WileyTitle/productCd-1118855388.html">eu.wiley.com/WileyCDA/WileyTitle/productCd-1118855388.html</a>
Foundation HTML5 Animation with JavaScript	<a href="http://www.apress.com/9781430236658">www.apress.com/9781430236658</a>
WebGL Programming Guide	<a href="http://sites.google.com/site/webglbook">sites.google.com/site/webglbook</a>
Effective JavaScript	<a href="http://effectivejs.com">effectivejs.com</a>
Learning jQuery Fourth Edition	<a href="http://www.packtpub.com/web-development/learning-jquery-fourth-edition">www.packtpub.com/web-development/learning-jquery-fourth-edition</a>

## 12.6. PHP

### 12.6.1. Web sites

php.net	<a href="http://php.net">php.net</a>
w3schools.com	<a href="http://www.w3schools.com/php/default.asp">www.w3schools.com/php/default.asp</a>
stackoverflow	<a href="http://www.stackoverflow.com">www.stackoverflow.com</a>
Deitel PHP Resource Center	<a href="http://www.deitel.com/ResourceCenters/Programming/PHP/PHPTutorials/tabid/804/Default.aspx">www.deitel.com/ResourceCenters/Programming/PHP/PHPTutorials/tabid/804/Default.aspx</a>
PHP Exercises	<a href="http://phpexercises.com">phpexercises.com</a>
PHP Fiddle	<a href="http://phpfiddle.org">phpfiddle.org</a>
Develop PHP	<a href="http://www.developphp.com">www.developphp.com</a>
TuxRadar Practical PHP Programming	<a href="http://www.tuxradar.com/practicalphp">www.tuxradar.com/practicalphp</a>
10 Best PHP Cheat Sheets for Quick Development	<a href="http://yemista.com/10-best-php-cheat-sheets-for-quick-development">http://yemista.com/10-best-php-cheat-sheets-for-quick-development</a>
After Hours Programming PHP Tutorial	<a href="http://www.afterhoursprogramming.com/tutorial/PHP/Overview">www.afterhoursprogramming.com/tutorial/PHP/Overview</a>
Web Programming with PHP	<a href="http://desarrolloweb.dlsi.ua.es/cursos/2012/web-programming-with-php/introduction">desarrolloweb.dlsi.ua.es/cursos/2012/web-programming-with-php/introduction</a>
Free PHP Books	<a href="http://mc23g.hubpages.com/hub/Download-Free-PHP-Books">mc23g.hubpages.com/hub/Download-Free-PHP-Books</a>
WIKIBOOKS PHP Programming	<a href="http://en.wikibooks.org/wiki/Programming:PHP">en.wikibooks.org/wiki/Programming:PHP</a>
PHP Academy	<a href="http://www.phpacademy.org">www.phpacademy.org</a>
PHP tutorial for beginners	<a href="http://phptutorial.info">phptutorial.info</a>
Free PHP Scripts	<a href="http://www.thefreecountry.com/php/index.shtml">www.thefreecountry.com/php/index.shtml</a>
PHP Classes	<a href="http://www.phpclasses.org">www.phpclasses.org</a>
PHP-Kurs	<a href="http://www.php-kurs.com">www.php-kurs.com</a>

### 12.6.2. Books

PHP in a Nutshell	<a href="http://shop.oreilly.com/product/9780596100674.do">shop.oreilly.com/product/9780596100674.do</a>
-------------------	--

## 12.7. MySQL

### 12.7.1. Web sites

mysql.com	<a href="http://www.mysql.com">www.mysql.com</a>
Essential MySQL Cheat Sheet	<a href="http://www.cheatography.com/guslong/cheat-sheets/essential-mysql">www.cheatography.com/guslong/cheat-sheets/essential-mysql</a>

## 12.7.2. Books

MySQL	<a href="http://www.mypearsonstore.com/bookstore/mysql-9780321833877">www.mypearsonstore.com/bookstore/mysql-9780321833877</a>
-------	--

## 12.8. XML

	<a href="https://github.com/Kickball/awesome-selfhosted#feed-readers">github.com/Kickball/awesome-selfhosted#feed-readers</a>
--	---

## 12.9. Scripts

Free programmers', webmasters' and security resources	<a href="http://www.thefreecountry.com">www.thefreecountry.com</a>
HotScripts	<a href="http://www.hotscripts.com">www.hotscripts.com</a>
Gerry's Script Library	<a href="http://www.gerd-tentler.de/tools">www.gerd-tentler.de/tools</a>

## 12.10. Programming

15 Free Games to Level Up Your Coding Skills	<a href="http://skillcrush.com/2017/04/03/free-coding-games">skillcrush.com/2017/04/03/free-coding-games</a>
CodinGame	<a href="http://www.codingame.com">www.codingame.com</a>
Code Combat	<a href="http://codecombat.com">codecombat.com</a>
Programming tutorials	<a href="http://www.java2s.com">www.java2s.com</a>
Learn programming on your Android device with Py	<a href="https://play.google.com/store/apps/details?id=com.py.learn&amp;rdid=com.py.learn">play.google.com/store/apps/details?id=com.py.learn&amp;rdid=com.py.learn</a>
Learn shell programming	<a href="http://www.learnshell.org">www.learnshell.org</a>
The Bash Guide	<a href="http://www.bash.academy">www.bash.academy</a>

## 12.11. Web apps

Can I use	<a href="http://caniuse.com">caniuse.com</a>
tutorialspoint	<a href="http://www.tutorialspoint.com">www.tutorialspoint.com</a>
Scriptol	<a href="http://www.scriptol.com">www.scriptol.com</a>
CUBE SLAM	<a href="http://www.cubeslam.com">www.cubeslam.com</a>
Browser Invaders	<a href="http://seb.ly/demos/BrowserInvaders">seb.ly/demos/BrowserInvaders</a>
Epic Citadel	<a href="http://www.unrealengine.com/html5">www.unrealengine.com/html5</a>
diep.io	<a href="http://diep.io">diep.io</a>

## 12.12. Artificial intelligence

Machine Learning - The Complete Guide	<a href="https://en.wikipedia.org/wiki/Book:Machine_Learning_%E2%80%93_The_Complete_Guide">en.wikipedia.org/wiki/Book:Machine_Learning_%E2%80%93_The_Complete_Guide</a>
A Course in Machine Learning	<a href="http://ciml.info">ciml.info</a>
Understanding Machine Learning: From Theory to Algorithms	<a href="http://www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/copy.html">www.cs.huji.ac.il/~shais/UnderstandingMachineLearning/copy.html</a>
Hacker's guide to neural networks	<a href="https://karpathy.github.io/neuralnets">karpathy.github.io/neuralnets</a>
Derivative rules	<a href="http://www.mathsisfun.com/calculus/derivatives-rules.html">www.mathsisfun.com/calculus/derivatives-rules.html</a>
	<a href="http://www.antoniodeluca.info/blog/10-08-2016/neural-networks-in-javascript.html">www.antoniodeluca.info/blog/10-08-2016/neural-networks-in-javascript.html</a>
	<a href="http://blog.webkid.io/neural-networks-in-javascript">blog.webkid.io/neural-networks-in-javascript</a>
	<a href="http://synaptic.juancazala.com">synaptic.juancazala.com</a>
	<a href="http://blog.niandrei.com/2015/03/01/javascript-artificial-intelligence-tutorial-for-beginners">blog.niandrei.com/2015/03/01/javascript-artificial-intelligence-tutorial-for-beginners</a>
	<a href="http://www.deeplearningbook.org">www.deeplearningbook.org</a>
	<a href="http://burakkanber.com/blog/machine-learning-in-other-languages-introduction">burakkanber.com/blog/machine-learning-in-other-languages-introduction</a>
	<a href="https://harthur.github.io/txjs-slides">harthur.github.io/txjs-slides</a>
	<a href="https://github.com/josephmisiti/awesome-machine-learning">github.com/josephmisiti/awesome-machine-learning</a>
	<a href="http://www.datasciencecentral.com/profiles/blogs/machine-learning-in-javascript-a-compilation-of-resources">www.datasciencecentral.com/profiles/blogs/machine-learning-in-javascript-a-compilation-of-resources</a>
	<a href="http://www.technologyreview.com/s/541276/deep-learning-machine-teaches-itself-chess-in-72-hours-plays-at-international-master">www.technologyreview.com/s/541276/deep-learning-machine-teaches-itself-chess-in-72-hours-plays-at-international-master</a>
	<a href="http://neuralnetworksanddeeplearning.com">neuralnetworksanddeeplearning.com</a>
	<a href="https://github.com/cazala/synaptic/wiki/Neural-Networks-101">github.com/cazala/synaptic/wiki/Neural-Networks-101</a>
	<a href="http://www.nytimes.com/2016/12/14/magazine/the-great-ai-awakening.html">www.nytimes.com/2016/12/14/magazine/the-great-ai-awakening.html</a>
	<a href="http://tutorialzine.com/2017/04/10-machine-learning-examples-in-javascript">tutorialzine.com/2017/04/10-machine-learning-examples-in-javascript</a>
	<a href="http://www.nanalyze.com/2017/03/free-artificial-intelligence-ai-software">www.nanalyze.com/2017/03/free-artificial-intelligence-ai-software</a>

The matrix cookbook	<a href="http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=3274">www2.imm.dtu.dk/pubdb/views/publication_details.php?id=3274</a>
	<a href="http://deeplearning.net">deeplearning.net</a>
Building our own self-driving car	<a href="http://www.streetdrone.com">www.streetdrone.com</a>
	<a href="http://blog.otoro.net/2015/06/19/neural-network-generative-art">blog.otoro.net/2015/06/19/neural-network-generative-art</a>
deeplearn.js	<a href="http://deeplearnjs.org">deeplearnjs.org</a>
Browser learns to play Flappy Bird	<a href="https://github.com/xviniette/FlappyLearning">github.com/xviniette/FlappyLearning</a>
MLweb	<a href="http://mlweb.loria.fr">mlweb.loria.fr</a>
Machine learning comes to your browser via JS	<a href="http://www.infoworld.com/article/3212884/machine-learning/machine-learning-comes-to-your-browser-via-javascript.html">www.infoworld.com/article/3212884/machine-learning/machine-learning-comes-to-your-browser-via-javascript.html</a>
Run Keras models in the browser, with GPU support using WebGL	<a href="https://github.com/transcranial/keras-js">github.com/transcranial/keras-js</a>
TensorFire is a framework for running neural networks in the browser, accelerated by WebGL.	<a href="http://tenso.rs">tenso.rs</a>
Neural network generative art javascript	<a href="http://blog.otoro.net/2015/06/19/neural-network-generative-art">blog.otoro.net/2015/06/19/neural-network-generative-art</a>
RecurrentJS	<a href="https://github.com/karpathy/recurrentjs">github.com/karpathy/recurrentjs</a>
	<a href="https://medium.com/@francois.chollet/the-impossibility-of-intelligence-explosion-5be4a9eda6ec">medium.com/@francois.chollet/the-impossibility-of-intelligence-explosion-5be4a9eda6ec</a>
Artificial intelligence index	<a href="http://www.aiindex.org">www.aiindex.org</a>
Mastering Chess and Shogi by Self-Play with a General Reinforcement Learning Algorithm	<a href="https://arxiv.org/abs/1712.01815">arxiv.org/abs/1712.01815</a>
Deep learning tutorials	<a href="http://deeplearning4j.org/tutorials">deeplearning4j.org/tutorials</a>
Audio samples from "Natural TTS Synthesis by Conditioning WaveNet on Mel Spectrogram Predictions"	<a href="https://google.github.io/tacotron/publications/tacotron2">google.github.io/tacotron/publications/tacotron2</a>
Colah's blog	<a href="https://colah.github.io">colah.github.io</a>
Jürgen Schmidhuber	<a href="http://people.idsia.ch/~juergen">people.idsia.ch/~juergen</a>
The future of deep learning	<a href="http://blog.keras.io/the-future-of-deep-learning.html">blog.keras.io/the-future-of-deep-learning.html</a>
YOLO (You only look once) real-time object detection	<a href="http://pjreddie.com/darknet">pjreddie.com/darknet</a>

### 12.12.1. Reinforcement learning

Reinforcement Learning: An Introduction	<a href="http://incompleteideas.net/book/the-book-2nd.html">incompleteideas.net/book/the-book-2nd.html</a>
Algorithms for Reinforcement Learning	<a href="http://sites.ualberta.ca/~szepesva/RLBook.html">sites.ualberta.ca/~szepesva/RLBook.html</a>

REINFORCEjs	<a href="http://cs.stanford.edu/people/karpathy/reinforcejs">cs.stanford.edu/people/karpathy/reinforcejs</a>
neurojs	<a href="https://github.com/janhuenermann/neurojs">github.com/janhuenermann/neurojs</a>
UCL Course on RL by David Silver	<a href="http://www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html">www0.cs.ucl.ac.uk/staff/d.silver/web/Teaching.html</a>
Getting started with reinforcement learning	<a href="https://opensource.com/article/17/11/reinforcement-learning">opensource.com/article/17/11/reinforcement-learning</a>
Demystifying Deep Reinforcement Learning	<a href="http://neuro.cs.ut.ee/demystifying-deep-reinforcement-learning">neuro.cs.ut.ee/demystifying-deep-reinforcement-learning</a>
	<a href="http://rll.berkeley.edu/deeprlcourse">rll.berkeley.edu/deeprlcourse</a>
General reinforcement learning in the browser	<a href="http://aslanides.io/aixijs">aslanides.io/aixijs</a>
	<a href="http://www.scholarpedia.org/article/Reinforcement_learning">www.scholarpedia.org/article/Reinforcement_learning</a>
Reinforcement learning resources curated	<a href="https://github.com/aikorea/awesome-rl">github.com/aikorea/awesome-rl</a>
	<a href="http://www.analyticsvidhya.com/blog/2017/01/introduction-to-reinforcement-learning-implementation">www.analyticsvidhya.com/blog/2017/01/introduction-to-reinforcement-learning-implementation</a>

### 12.12.2. Data sources

<a href="http://guides.emich.edu/data/free-data">guides.emich.edu/data/free-data</a>
<a href="http://people.sc.fsu.edu/~jburkardt/datasets/regression/regression.html">people.sc.fsu.edu/~jburkardt/datasets/regression/regression.html</a>

## 12.13. Internet

Internet Assigned Numbers Authority	<a href="http://www.iana.org">www.iana.org</a>
Free DNS query and whois tools	<a href="http://dnsquery.org">dnsquery.org</a>

## 12.14. Security

Privacy tools	<a href="http://www.privacytools.io">www.privacytools.io</a>
20 Best Tips to Stay Anonymous and Protect Your Online Privacy	<a href="http://www.extremetech.com/internet/180485-the-ultimate-guide-to-staying-anonymous-and-protecting-your-privacy-online">www.extremetech.com/internet/180485-the-ultimate-guide-to-staying-anonymous-and-protecting-your-privacy-online</a>
Hacksplaining	<a href="http://www.hacksplaining.com">www.hacksplaining.com</a>
Open Web Application Security Project	<a href="http://www.owasp.org">www.owasp.org</a>
Electronic Frontier Foundation	<a href="http://www.eff.org">www.eff.org</a>
Ransom Free	<a href="http://ransomfree.cybereason.com">ransomfree.cybereason.com</a>
Security Engineering — The Book	<a href="http://www.cl.cam.ac.uk/~rja14/book.html">www.cl.cam.ac.uk/~rja14/book.html</a>

Canary tokens	<a href="http://blog.thinkst.com/p/canarytokensorg-quick-free-detection.html">blog.thinkst.com/p/canarytokensorg-quick-free-detection.html</a>
Cross-browser fingerprinting	<a href="http://www.ghacks.net/2017/02/14/researchers-develop-cross-browser-fingerprinting-technique">www.ghacks.net/2017/02/14/researchers-develop-cross-browser-fingerprinting-technique</a>
Penetration testing tools	<a href="http://www.softwaretestinghelp.com/penetration-testing-tools">www.softwaretestinghelp.com/penetration-testing-tools</a>
DDOS protection	<a href="http://javapipe.com/iptables-ddos-protection">javapipe.com/iptables-ddos-protection</a>
JS crypto libraries	<a href="https://gist.github.com/jo/8619441">gist.github.com/jo/8619441</a>
How to block bad web sites	<a href="http://www.techsupportalert.com/content/how-block-bad-websites.htm">www.techsupportalert.com/content/how-block-bad-websites.htm</a>
Browser Privacy Test	<a href="http://tenta.com/test">tenta.com/test</a>
Hack a web app	<a href="https://sourceforge.net/projects/bwapp/files/bee-box">sourceforge.net/projects/bwapp/files/bee-box</a>
The best penetration testing tool for Windows	<a href="http://pentestbox.org">pentestbox.org</a>
Example of a successful phishing attack on a Google extension	<a href="http://www.ghacks.net/2017/07/31/chrome-extension-copyfish-hijacked-remove-now">www.ghacks.net/2017/07/31/chrome-extension-copyfish-hijacked-remove-now</a>
Free cyber security learning	<a href="http://www.cybrary.it">www.cybrary.it</a>
Online platform to test and advance your skills in cyber security	<a href="http://www.hackthebox.eu">www.hackthebox.eu</a>
Sonar scanner	<a href="http://sonarwhal.com/scanner">sonarwhal.com/scanner</a>
	<a href="http://malwaretips.com/blogs/reimageplus-popup-virus">malwaretips.com/blogs/reimageplus-popup-virus</a>

## 12.15. OS

Server world	<a href="http://www.server-world.info/en">www.server-world.info/en</a>
	<a href="http://www.linuxfromscratch.org">www.linuxfromscratch.org</a>
Microsoft ISO file download	<a href="http://tb.rg-adguard.net">tb.rg-adguard.net</a>
Microsoft Windows commands	<a href="http://download.microsoft.com/download/5/8/9/58911986-D4AD-4695-BF63-F734CD4DF8F2/ws-commands.pdf">download.microsoft.com/download/5/8/9/58911986-D4AD-4695-BF63-F734CD4DF8F2/ws-commands.pdf</a>
	<a href="http://saintlad.com/install-macos-sierra-in-virtualbox-on-windows-10">saintlad.com/install-macos-sierra-in-virtualbox-on-windows-10</a>
The Most Detailed Guide to Windows Device Drivers On The Web	<a href="http://www.technorms.com/71242/windows-device-drivers-complete-guide">www.technorms.com/71242/windows-device-drivers-complete-guide</a>

## 12.16. Hardware

Hard drive reliability statistics	<a href="http://www.backblaze.com/b2/hard-drive-test-data.html">www.backblaze.com/b2/hard-drive-test-data.html</a>
-----------------------------------	--

PCPartPicker	<a href="http://de.pcpartpicker.com">de.pcpartpicker.com</a>
--------------	--

## 12.17. Public domain pictures

Creative Commons	<a href="http://search.creativecommons.org">search.creativecommons.org</a>
Pixabay	<a href="http://pixabay.com">pixabay.com</a>
Pexels	<a href="http://www.pexels.com">www.pexels.com</a>
StockSnap.io	<a href="http://stocksnap.io">stocksnap.io</a>
FreeImages	<a href="http://www.freeimages.com">www.freeimages.com</a>
Getty Images	<a href="http://www.gettyimages.in">www.gettyimages.in</a>
Unsplash	<a href="http://unsplash.com">unsplash.com</a>
SplitShire	<a href="http://www.splitshire.com">www.splitshire.com</a>
Morguefile	<a href="http://morguefile.com">morguefile.com</a>
	<a href="#">Free photos</a>
The Metropolitan Museum of Art	<a href="http://www.metmuseum.org/art/collection#!?perPage=20&amp;showOnly=openaccess&amp;sortBy=Relevance&amp;sortOrder=asc&amp;offset=0&amp;pageSize=0">www.metmuseum.org/art/collection#!?perPage=20&amp;showOnly=openaccess&amp;sortBy=Relevance&amp;sortOrder=asc&amp;offset=0&amp;pageSize=0</a>
NASA Image and Video Library	<a href="http://images.nasa.gov">images.nasa.gov</a>

## 12.18. Public domain music

digccmixter	<a href="http://dig.ccmixter.org">dig.ccmixter.org</a>
OurMusicBox	<a href="http://ourmusicbox.com">ourmusicbox.com</a>
Bensound	<a href="http://www.bensound.com">www.bensound.com</a>
Icompetech	<a href="http://incompetech.com">incompetech.com</a>
IMSLP	<a href="http://imslp.org">imslp.org</a>
MUSOPEN	<a href="http://musopen.org">musopen.org</a>
16,000 BBC sound effects	<a href="http://bbcsfx.acropolis.org.uk">bbcsfx.acropolis.org.uk</a>

## 12.19. Fonts and icons

<a href="http://www.smashingmagazine.com/2017/02/30-free-fonts-with-personality-and-style">www.smashingmagazine.com/2017/02/30-free-fonts-with-personality-and-style</a>
<a href="http://www.creativeblog.com/web-design/free-icon-sets-10134829">www.creativeblog.com/web-design/free-icon-sets-10134829</a>
<a href="http://www.vecteezy.com/vector-art/99239-business-icon-vectors">www.vecteezy.com/vector-art/99239-business-icon-vectors</a>
<a href="http://pixabay.com/en/blog/posts/top-5-resources-for-free-vector-icons-17">pixabay.com/en/blog/posts/top-5-resources-for-free-vector-icons-17</a>
<a href="http://www.techsupportalert.com/content/33-sites-great-icons-you-can-download-free.htm">www.techsupportalert.com/content/33-sites-great-icons-you-can-download-free.htm</a>

## 12.20. Internet references

	<a href="https://devdocs.io">devdocs.io</a>
--	---

## 12.21. Online courses and tutorials

<a href="https://medium.freecodecamp.com/ivy-league-free-online-courses-a0d7ae675869#.esfpv7qzo">medium.freecodecamp.com/ivy-league-free-online-courses-a0d7ae675869#.esfpv7qzo</a>
<a href="https://hackr.io">hackr.io</a>

## 12.22. Data

	<a href="https://generatedata.com">generatedata.com</a>
--	---

## 12.23. Search engines

	<a href="#">Search engines</a>
Non-profit library of millions of free books, movies, software, music, websites, and more	<a href="https://archive.org">archive.org</a>
	<a href="https://www.audiosear.ch">www.audiosear.ch</a>

## 12.24. Office software

Zero knowledge realtime collaborative editor	<a href="https://github.com/xwiki-labs/cryptpad">github.com/xwiki-labs/cryptpad</a> <a href="https://github.com/xwiki-labs/cryptpad">https://github.com/xwiki-labs/cryptpad</a>
	<a href="https://onlyoffice.com">ONLYOFFICE</a>
Free MS Office templates	<a href="https://www.hloom.com">www.hloom.com</a>
MS Office training	<a href="https://support.office.com/en-us/article/office-training-roadmaps-62a4b0dc-beba-4d8e-b79c-0ad200e705a1">support.office.com/en-us/article/office-training-roadmaps-62a4b0dc-beba-4d8e-b79c-0ad200e705a1</a>

## 12.25. Thunderbird

<a href="https://infraadvisory.wordpress.com/2015/12/01/make-google-calendar-default-in-thunderbird">infraadvisory.wordpress.com/2015/12/01/make-google-calendar-default-in-thunderbird</a>
<a href="https://kb.mozillazine.org/Changing_the_web_browser_invoked_by_Thunderbird">kb.mozillazine.org/Changing_the_web_browser_invoked_by_Thunderbird</a>
<a href="https://superuser.com/questions/416852/how-can-i-force-thunderbird-to-re-download-a-corrupted-message-from-the-imap-server">superuser.com/questions/416852/how-can-i-force-thunderbird-to-re-download-a-corrupted-message-from-the-imap-ser</a>

## 12.26. Media editing

	<a href="https://github.com/mifi/lossless-cut">github.com/mifi/lossless-cut</a>
Smart PNG and JPEG compression	<a href="https://tinypng.com">tinypng.com</a>
	<a href="https://imagecompressor.io">imagecompressor.io</a>

## 12.27. Music

	<a href="https://askubuntu.com/questions/10402/is-there-software-like-music-maker">askubuntu.com/questions/10402/is-there-software-like-music-maker</a>
	<a href="https://ubuntustudio.org">ubuntustudio.org</a>
LMMS open source digital audio workstation	<a href="https://lmms.sourceforge.net">lmms.sourceforge.net</a>
Audiotool	<a href="https://www.audiotool.com">www.audiotool.com</a>
<a href="https://www.blackmagicdesign.com/products/davinciresolve">www.blackmagicdesign.com/products/davinciresolve</a>	Useful resources
<a href="https://lmms.io/wiki/index.php?title=Useful_resources">lmms.io/wiki/index.php?title=Useful_resources</a>	Ultimate List of FREE Soundfonts
<a href="https://cymatics.fm/blogs/production/soundfonts">cymatics.fm/blogs/production/soundfonts</a>	Free Quality SoundFonts (sf2)
<a href="https://sites.google.com/site/soundfonts4u">sites.google.com/site/soundfonts4u</a>	MuseScore Orchestra SoundFont
<a href="https://drive.google.com/drive/folders/0B8XyJFPN233fbmY3WFhMT3djc1E">drive.google.com/drive/folders/0B8XyJFPN233fbmY3WFhMT3djc1E</a>	Petrucchi Music Library
<a href="https://imslp.org/wiki/Main_Page">imslp.org/wiki/Main_Page</a>	Improve focus and boost your productivity
<a href="https://www.noisli.com">www.noisli.com</a>	
<a href="https://www.ghacks.net/2017/11/06/the-ultimate-game-sounds-list">www.ghacks.net/2017/11/06/the-ultimate-game-sounds-list</a>	Royalty free music

## 12.28. Graphics

	<a href="https://vectr.com">vectr.com</a>
Easy design	<a href="https://crello.com">crello.com</a>
GIFs	<a href="https://giphy.com">giphy.com</a>
Advanced image editor	<a href="https://www.photopea.com">www.photopea.com</a>
	<a href="https://mara.photos">mara.photos</a>

## 12.29. Selfhosted

	<a href="https://github.com/Kickball/awesome-selfhosted">github.com/Kickball/awesome-selfhosted</a>
Open source video conferencing	<a href="https://jitsi.org">jitsi.org</a>

## 12.30. Website builder

Free website buider	<a href="http://www.wix.com">www.wix.com</a>
---------------------	--

## 12.31. Tools

Lipsum	<a href="http://lipsum.com">lipsum.com</a>
ShareX	<a href="http://getsharex.com">getsharex.com</a>
OBS Studio	<a href="http://obsproject.com">obsproject.com</a>
Captura	<a href="http://mathewsachin.github.io/Captura">mathewsachin.github.io/Captura</a>
Screenshot captor	<a href="http://www.donationcoder.com/Software/Mouser/screenshotcaptor">www.donationcoder.com/Software/Mouser/screenshotcaptor</a>
GifCam	<a href="http://blog.bahraniapps.com/gifcam">blog.bahraniapps.com/gifcam</a>
Online conversion tools	<a href="http://gchq.github.io/CyberChef/cyberchef.htm">gchq.github.io/CyberChef/cyberchef.htm</a>
	<a href="http://www.nirsoft.net/utils/search_my_files.html">www.nirsoft.net/utils/search_my_files.html</a>
	<a href="http://www.ultimatebootcd.com">www.ultimatebootcd.com</a>
	<a href="http://www.favicon-generator.org">www.favicon-generator.org</a>
	<a href="http://realfavicongenerator.net">realfavicongenerator.net</a>
Favicon finder	<a href="http://icons.better-idea.org/icons">icons.better-idea.org/icons</a>
YouTube to mp3 converter	<a href="http://ytmp3.com">ytmp3.com</a>
	<a href="http://www.digitalcitizen.life/easiest-way-legally-download-iso-images-windows-and-office">www.digitalcitizen.life/easiest-way-legally-download-iso-images-windows-and-office</a>
Wayback Machine	<a href="http://archive.org/web">archive.org/web</a>
	<a href="http://www.ghacks.net/2017/04/04/website-downloader-download-entire-wayback-machine-site-archives">www.ghacks.net/2017/04/04/website-downloader-download-entire-wayback-machine-site-archives</a>
VPS hosting	<a href="http://contabo.com/?show=vps">contabo.com/?show=vps</a>
Webrecorder	<a href="http://webrecorder.io">webrecorder.io</a>
Latest Flash Player plugin in XPI package, useful for portable Firefox	<a href="http://flashplay.no-ip.org">flashplay.no-ip.org</a>
	<a href="http://addons.mozilla.org/en-US/firefox/addon/youtube-flash-video-player">addons.mozilla.org/en-US/firefox/addon/youtube-flash-video-player</a>
	<a href="http://www.calculator.net">www.calculator.net</a>
	<a href="http://www.google.com/trends">www.google.com/trends</a>
List of website speed test tools	<a href="http://www.keycdn.com/blog/website-speed-test-tools">www.keycdn.com/blog/website-speed-test-tools</a>
Optimize web page speed	<a href="http://developers.google.com/speed">developers.google.com/speed</a>
Analyze site speed	<a href="http://gtmetrix.com">gtmetrix.com</a>

Firefox backup	<a href="https://addons.mozilla.org/en-us/firefox/addon/febe">addons.mozilla.org/En-us/firefox/addon/febe</a>
File unlockers	<a href="http://www.ghacks.net/2017/03/08/unlock-and-delete-locked-files-and-folders-with-thisismyfile">www.ghacks.net/2017/03/08/unlock-and-delete-locked-files-and-folders-with-thisismyfile</a>
Email backup	<a href="http://www.mailstore.com/en/mailstore-home-email-archiving.aspx">www.mailstore.com/en/mailstore-home-email-archiving.aspx</a>
Host provider	<a href="http://www.webtropia.com">www.webtropia.com</a>
Hosting, DNS etc.	<a href="http://www.namecheap.com">www.namecheap.com</a>
ShellCheck	<a href="http://www.shellcheck.net">www.shellcheck.net</a>
Tiny proxy	<a href="http://tinyproxy.github.io">tinyproxy.github.io</a>
PC building simulator	<a href="http://pcbuidingsimulator.wordpress.com">pcbuidingsimulator.wordpress.com</a>
The Ultimate Distraction-Free Text Editor	<a href="http://getcoldturkey.com/writer">getcoldturkey.com/writer</a>
Online sticky notes	<a href="http://pinup.com">pinup.com</a>
IF This Then That (IFTTT)	<a href="http://www.ghacks.net/2017/03/20/top-ifttt-recipes-improve-android">www.ghacks.net/2017/03/20/top-ifttt-recipes-improve-android</a>
Open source software to split, merge and rotate PDF files	<a href="http://www.pdfsam.org/download-pdfsam-basic">www.pdfsam.org/download-pdfsam-basic</a>
Hide files or folders in JPEG images	<a href="http://www.filefriend.net">www.filefriend.net</a>
Change PDF Viewer back to Acrobat or Reader in Google Chrome	<a href="http://blogs.adobe.com/vikrant/2010/12/use-adobe-plugins-to-view-pdfs-inside-google-chrome">blogs.adobe.com/vikrant/2010/12/use-adobe-plugins-to-view-pdfs-inside-google-chrome</a>
21 free tools for your next design project	<a href="http://www.techsupportalert.com/content/21-free-tools-your-next-design-project.htm">www.techsupportalert.com/content/21-free-tools-your-next-design-project.htm</a>
Email delivery service for developers	<a href="http://www.sparkpost.com">www.sparkpost.com</a>
Quickly OCR a portion of the screen	<a href="http://capture2text.sourceforge.net">capture2text.sourceforge.net</a>
Online OCR	<a href="http://www.onlineocr.net">www.onlineocr.net</a>
Taskbar system stats for Windows	<a href="http://entropy6.com/xmeters">entropy6.com/xmeters</a>
A better way to view & analyze data	<a href="http://tadviewer.com">tadviewer.com</a>
AstroGrep file content search	<a href="http://astrogrep.sourceforge.net">astrogrep.sourceforge.net</a>
Record mouse and keyboard actions for infinite replay	<a href="http://www.mouserecorder.com">www.mouserecorder.com</a>
Low-level PC hardware read/write tool	<a href="http://rweverything.com">rweverything.com</a>
Top quality online translation	<a href="http://www.deepl.com/translator">www.deepl.com/translator</a>
Find the files and folders that use the most disk space	<a href="http://antibody-software.com/web/software/software/wiztree-finds-the-files-and-folders-using-the-most-disk-space-on-your-hard-drive">antibody-software.com/web/software/software/wiztree-finds-the-files-and-folders-using-the-most-disk-space-on-your-hard-drive</a>
PC benchmarking	<a href="http://novabench.com">novabench.com</a>
Disk cleaning	<a href="http://www.koshyjohn.com/software/diskmax">www.koshyjohn.com/software/diskmax</a>
IFTTT	<a href="http://ifttt.com">ifttt.com</a>

Easily convert files into SQL databases	<a href="http://sqlizer.io">sqlizer.io</a>
Bootice, edit MBR and PBR	<a href="http://www.softpedia.com/get/System/Boot-Manager-Disk/Bootice.shtml">www.softpedia.com/get/System/Boot-Manager-Disk/Bootice.shtml</a>
SpeechTexter (requires Chrome)	<a href="http://www.speechtexter.com">www.speechtexter.com</a>
PDF to Word converter	<a href="http://smallpdf.com/pdf-to-word">smallpdf.com/pdf-to-word</a>
Convert anything to anything	<a href="http://cloudconvert.com">cloudconvert.com</a>
Useful applications including OSArmor	<a href="http://www.novirusthanks.org">www.novirusthanks.org</a>
Online survey tools	<a href="http://opensource.com/article/17/2/tools-online-surveys-polls">opensource.com/article/17/2/tools-online-surveys-polls</a>
Online tone generator	<a href="http://www.szynalski.com/tone-generator">www.szynalski.com/tone-generator</a>
Backup Windows program settings	<a href="http://www.cloneapp.de">www.cloneapp.de</a>
Websites change. Perma Links don't.	<a href="http://perma.cc">perma.cc</a>
Itty bitty sites are contained entirely within their own link.	<a href="http://itty.bitty.site/edit">itty.bitty.site/edit</a>
High quality translation	<a href="http://www.deepl.com">www.deepl.com</a>
List of free link checkers	<a href="http://linksearching.com/9-best-free-online-broken-link-checker-tool">linksearching.com/9-best-free-online-broken-link-checker-tool</a>
Link checker	<a href="http://validator.w3.org/checklink">validator.w3.org/checklink</a>
Check My Links for Chrome	<a href="http://chrome.google.com/webstore/detail/check-my-links/ojkcdipcgfaekbeaelaapakgnjflfglf">chrome.google.com/webstore/detail/check-my-links/ojkcdipcgfaekbeaelaapakgnjflfglf</a>
Download your Google data	<a href="http://takeout.google.com/settings/takeout">takeout.google.com/settings/takeout</a>
Google search console	<a href="http://search.google.com/search-console">search.google.com/search-console</a>
Pinetools	<a href="http://pinetools.com">pinetools.com</a>
The app highlights lengthy, complex sentences and common errors	<a href="http://www.hemingwayapp.com">www.hemingwayapp.com</a>
Cyotek WebCopy	<a href="http://www.cyotek.com/cyotek-webcopy/downloads">www.cyotek.com/cyotek-webcopy/downloads</a>
Third-party tools that find Windows keys	<a href="http://www.techspot.com/guides/1760-find-your-windows-product-key">www.techspot.com/guides/1760-find-your-windows-product-key</a>
	<a href="http://www.wikihow.com/Download-All-Images-on-a-Web-Page-at-Once">www.wikihow.com/Download-All-Images-on-a-Web-Page-at-Once</a>
Disposable email	<a href="http://temp-mail.org">temp-mail.org</a>
The world's tiniest spreadsheet	<a href="http://tinysheet.com">tinysheet.com</a>
Windows Repair Toolbox	<a href="http://www.techsupportalert.com/content/use-all-one-tool-diagnose-and-repair-pc-problems.htm">www.techsupportalert.com/content/use-all-one-tool-diagnose-and-repair-pc-problems.htm</a>
	<a href="http://merabheja.com/12-best-free-file-comparison-tools-for-windows-10">merabheja.com/12-best-free-file-comparison-tools-for-windows-10</a>

Full list of Skype emoticons	<a href="https://support.skype.com/en/faq/fa12330/what-is-the-full-list-of-emoticons">support.skype.com/en/faq/fa12330/what-is-the-full-list-of-emoticons</a>
------------------------------	---