

These are my Hyper Text Markup Language (HTML), Cascading Style Sheet (CSS), Hypertext PreProcessor (PHP) and Java Script (JS) notes. I keep them here is so I can quickly look up the codes I use most often.

```
<!DOCTYPE html> <head> <meta charset="UTF-8">
<title>Page Title</title>
<meta name="description" content="Site description for search engines." />
<meta name="keywords" content="search, keywords, separated, with, commas" />
<meta name="apple-mobile-web-app-capable" content="yes" />
<meta name="mobile-web-app-capable" content="yes" />
<link rel="shortcut icon" href="16x16.ico" />
<link rel="apple-touch-icon-precomposed" href="57x57.png" />
<link rel="icon" sizes="192x192" href="192x192.png" />
<style type="text/css" media="all">
</style>
</head> <body>
<script> </script>
</body> </html>
```

<!-- This is a comment among HTML. -->

**Line break:** <br />

**Div:** <div style="css"> </div>

**bold, italic** <span class="cssclass">text</span>

**Link:** <a class="cssclass" target="\_blank" href="index.html">Link</a> (new tab)  
<a class="cssclass" target="\_top" href="index.html">Link</a> (main frame)

**Image:** 

**Hypertext Access:** You can make a file on notepad and save it as: .htaccess Since .htaccess is the format, it can appear as having no filename.

**Redirecting:**

ErrorDocument 404 http://...  
ErrorDocument 403 http://...

**Changing the index:**

DirectoryIndex mainpage.html  
This will become the default page instead of index.html

**CSS:** External, Internal, Inline (last overrides first, !important is priority) text-decoration:none!important; /\* This is a comment among CSS. \*/

**Class and id:** .classname #idname

Class names can be repeated but each id must be unique. Classes can be applied to specific elements or a combination of different elements.

img.class { ... } (applies only to pictures of that class)

.class { ... } (applies to all elements of that class)

**Attributes for multiple classes/ids:** (separate with commas) #group5, .group6 { ... }

**Element with multiple classes:** (separate with spaces, ids are unique) class="group1 group2"

<span class="group1 group2"> .group1 { text-decoration:underline; } .group2 { font-style:italic; } </span>

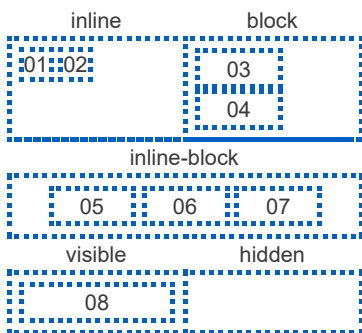
**Screen Orientation CSS:** @media all and (orientation:portrait) { #idname { } }

@media all and (orientation:landscape) { #idname { } }

**Focus:** input:focus (selected) \*:focus { outline: 0px none; } (removes on click outline)

**CSS Links:** a { } a:hover { } a.class:hover { }

**Container element hover:** .outer:hover .inner { ... } (Changes .inner upon hovering on .outer, where .outer contains .inner)

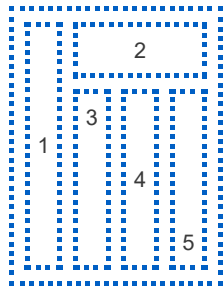
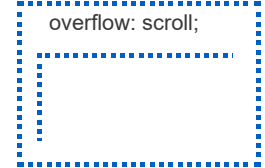


**Display:** display:inline; (elements are next to each other with the space the content occupies)  
display:block; (new elements go to the next line)  
display:inline-block; (elements are next to each other but can be given a specific size)  
display:none; (makes elements invisible, they take up no space)

**Visibility:** visibility:visible; (elements are shown)  
visibility:hidden; (makes elements invisible, they take up the same space)

**Mouse cursors:** cursor:url(design/choco1.cur),url(design/choco2.gif),auto; cursor:auto;  
cursor:default; cursor:pointer; cursor:crosshair; cursor:move; cursor:help; cursor:progress;  
cursor:wait; cursor:text; cursor:n-resize; cursor:ne-resize; cursor:e-resize; cursor:se-resize;  
cursor:s-resize; cursor:sw-resize; cursor:w-resize; cursor:nw-resize; cursor:inherit;

**Webkit Scrollbars:** `div#scroll4::-webkit-scrollbar { width:15px; height:14px; } /* size of scrollbar */`  
 ♥ `div#scroll4::-webkit-scrollbar-track { background-color:#ff70c3; border-radius: 50px; } /* scrollbar background */`  
 ♠ `div#scroll4::-webkit-scrollbar-thumb { background-color:#2d8fff; border-radius: 50px; } /* scroller */`  
 ♣ `div#scroll4::-webkit-scrollbar-button { background-color:#33c45d; border-radius: 50px; } /* scrollbar button */`  
 ♦ `div#scroll4::-webkit-scrollbar-corner { background-color:#ff9258; border-radius: 50px; } /* corner between bars */`  
 ☆ `div#scroll4::-webkit-scrollbar-button:horizontal:increment { background-image:url(design/scrollright.gif); background-position:center center; background-repeat:no-repeat; width:15px; height:15px; }`  
 ☆ `div#scroll4::-webkit-scrollbar-button:horizontal:decrement { background-image:url(design/scrollleft.gif); background-position:center center; background-repeat:no-repeat; width:15px; height:15px; }`  
 ☆ `div#scroll4::-webkit-scrollbar-button:vertical:increment { background-image:url(design/scrolldown.gif); background-position:center center; background-repeat:no-repeat; width:15px; height:15px; }`  
 ☆ `div#scroll4::-webkit-scrollbar-button:vertical:decrement { background-image:url(design/scrollup.gif); background-position:center center; background-repeat:no-repeat; width:15px; height:15px; }`



**Table:** `<table class="cssclass"> <tr> <td rowspan="2" class="cssclass">1</td> <td colspan="3" class="cssclass">2</td> </tr> <tr> <td class="cssclass">3</td> <td class="cssclass">4</td> <td class="cssclass">5</td> </tr> </table>`

**Cell Spacing:** `border-spacing:0px; border-collapse:collapse; border-collapse:separate;`

**Vertical alignment:** `vertical-align:top; vertical-align:middle; vertical-align:bottom;`

**Lists:** `<ol class="css_class"> <li>item here</li> </ol>`  
`<ul class="css_class"> <li>item here</li> </ul>`

- **Nested Lists:** `<ul style="margin:0px; list-style-image:url(design/star1.gif);"> <li class="level1"> level1 </li> <li class="level2"> level2 </li> </ul>`
  - `li.level1 { margin:0px 0px 0px 0px; } li.level2 { margin:0px 0px 0px 30px; }`

♥ **Before and After Pseudo Classes:** Pictures, unicode symbols and unicode emoji. ♥

♥ `<span class="spanli"> </span> <br />` ♥

♥ `span.spanli:before { content:url(design/star2.gif); }` ♥

♥ `span.spanli:after { padding-left:5px; content:"\2661"; font-size:14px; color:#ffa8fc; text-shadow: -1px 1px 0px #ff3e3e; }` ♥

**Special Characters:** `space &nbsp;`; `< &lt;`; `> &gt;`; `& &amp;`; `ñ &ntilde;`; `é &eacute;`;

**Width and height:** `html { height:100%; width:100%; } body { height:100%; width:100%; }`

**Calculate Size:** `width: calc(100% - 50px);`



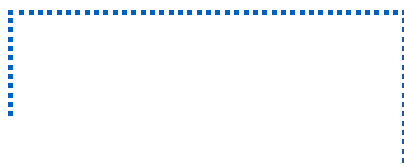
**Resize and keep aspect ratio:** `max-width:100px; max-height:100px;`

**Alignment:** `text-align:center; text-align:justify; text-align:left; text-align:right;` This sets the alignment of other elements aside from text.

**Overflow:** `overflow:hidden;` (no scrollbars)  
`overflow:scroll;` (both scrollbars)  
`overflow-y:hidden; overflow-x:scroll;` (horizontal scrollbar)  
`overflow-y:scroll; overflow-x:hidden;` (vertical scrollbar)



**Position:** `position:relative;` can contain `position:absolute;` and `overflow:hidden;` can be used with layers.  
`position:absolute; left:100px; top:10px;` (content scrolls)  
`position:absolute; right:100px; bottom:10px;` (content scrolls)  
`position:fixed; top:10px; right:50px;` (always visible on screen)  
`position:fixed; bottom:10px; left:50px;` (always visible on screen)  
`position:relative; left:20px; top:10px;` (relative to its original position, can have negative numbers)



**Layers:** (largest are on top) `z-index:3; z-index:2; z-index:1;`

**Make elements float:** `float:left; float:right;`

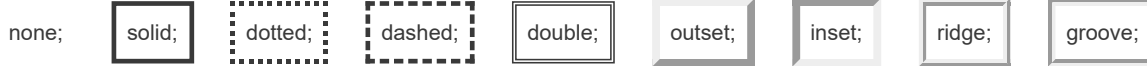
**Padding:** (inside the element border) `padding:10px; padding:20px 50px 30px 40px;`

padding-top:20px; padding-right:50px; padding-bottom:30px; padding-left:40px;

**Margin:** (space outside the element border) margin:10px; margin:10px 5px 10px 5px; margin-top:10%; margin-right:5px; margin-bottom:10%; margin-left:5px;

**Horizontally center elements:** text-align:center; (on the element containing it)  
margin-right:auto; margin-left:auto; or margin: 0px auto 0px auto; (on the element)

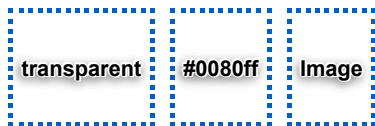
**Borders:** border-width:3px; border-style:solid; border-color:#000000; border:1px solid #000000; border:0px none; border-top-color:#000000; border-right-color:#ffffff; border-bottom-style:dashed; border-left-width:3px;



**Round borders:** horizontal top left, horizontal top right, horizontal bottom right, horizontal bottom left / vertical top left, vertical top right, vertical bottom right, vertical bottom left  
border-radius: 5px 25px 5px 25px / 5px 25px 5px 25px; border-radius: 35px 35px 35px 35px / 35px 35px 35px 35px;

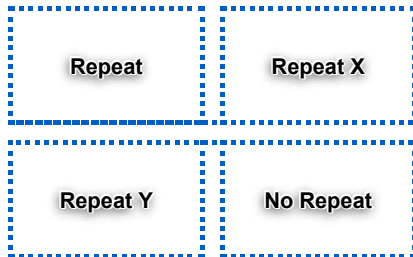
**Vertical Center and Bottom Flex Alignment:** display:flex; align-items:center; align-items:flex-end; height:100%; width:100%; min-height:???px;  
<div class="showbor" style="align-items:center; display:flex; min-height:100px; height:100px; width:150px;"> <div class="showbor" style="width:100px; height:50px; margin:0px auto 0px auto;"> </div> </div>

**Background colors and pictures:** background-color:transparent; background-color:none; background-color:#ffffff; background-image:url(picture.png);

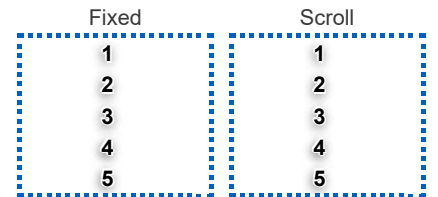


**Background scrolling:** background-attachment:fixed; (the background doesn't move)

background-attachment:scroll; (the background moves with the content)

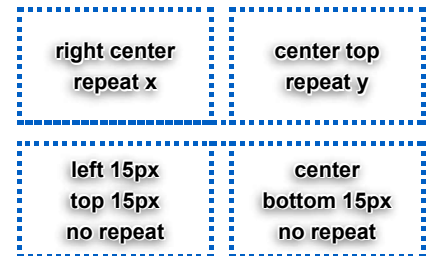


**Tiled background:** background-repeat:no-repeat; (the background is not repeated)  
background-repeat:repeat; (the background is repeated horizontally and vertically)  
background-repeat:repeat-x; (the background is repeated horizontally)  
background-repeat:repeat-y; (the background is repeated vertically)



**Background position:** background-position: 0px 0px;

Other background positions: left top | left center | left bottom | right top | right center | right bottom | center top | center center | center bottom



**Background size:** Use auto to keep the aspect ratio. The % is relative to the element.  
background-size: width height; background-size: auto 100px; background-size: 100% auto;

Full screen picture background: html, body { width:100%; height:100%; }  
body { background-size: 100% auto; background-repeat:no-repeat; background-position:center center; }

**Multiple backgrounds:** The attributes must be in the same order, the first background goes on top. background-image:url(blueback.png),url(skyblueline.png); background-repeat:no-repeat,repeat-x; background-position:center top, center center;



**Opacity:** opacity:1.0; opacity:0.9; opacity:0.8; opacity:0.7; opacity:0.6; opacity:0.5; opacity:0.4; opacity:0.3; opacity:0.2; opacity:0.1;



**Gradients:** background: linear-gradient(start position, start color, end color);  
background:linear-gradient(#0080ff, #c0c0c0);

**Box Shadow:** box-shadow: horizontal -left +right | vertical -up +down | blur | padding | color

```
box-shadow: -3px -3px 3px 3px #ff8ee8;
```

```
box-shadow: 0px 10px 5px -4px #33a4ff;
```

```
box-shadow: inset 0px 0px 7px 7px #ffff00;
```

**Word Wrap:**

```
word-wrap:break-word;
```

**Font types:**

```
@font-face { font-family: fontname; src:url(fontname.ttf); }
font-family:Verdana; font-family:Arial;
```

```
word-wrap:normal;
```



```
Font sizes: font-size:15px; font-size:10px;
```

```
-webkit-text-size-adjust: none; (prevents mobiles from resizing text)
```

```
Font colors: color:#000000; color:#ffffff; color:#53a8ff; color:#ff8aeb; color:#ffa100;
```

**Font style:**

```
font-style:normal; font-style:italic;
font-weight:normal; font-weight:bold;
text-decoration:none;
text-decoration:line-through;
text-decoration:underline;
TEXT-TRANSFORM:UPPERCASE;
text-transform:lowercase;
```

**Text spacing:**

```
line-height:15px;
line-height:40px;
word-spacing:5px; The word spacing is 5px.
word-spacing:-3px; Thewordspacingis-3px.
letter-spacing:4px; l e t t e r - s p a c i n g : 4 p x ;
letter-spacing:-2px; l e t t e r - s p a c i n g : - 2 p x ;
```

**Text shadow:** Numbers: (first) - left / + right (second) - up / + down (third) blur

```
text-shadow: none;
```

```
text-shadow: -1px 1px 0px #00ffff;
```

```
text-shadow: -1px -1px 0px #ffffff, 1px 1px 0px #ffffff, 1px 0px 0px #ffffff, 0px 1px 0px #ffffff, -1px 0px 0px #ffffff, 0px -1px 0px #ffffff, -1px 2px 7px #000000; font-weight:bold; color:#333333;
```

**Columns:** Number of columns: column-count:3; or column-width:250px;

```
Space between columns: column-gap:30px;
```

```
Line between columns: column-rule: 5px dotted #0080ff; or column-rule-color: #0080ff; column-rule-style: dotted; column-rule-width: 5px;
-moz- and -webkit- are also needed for browser compatibility.
```

```
column-count:2; column-gap:30px; column-rule: 5px dotted #0080ff;
-webkit-column-count:2; -webkit-column-gap:30px; -webkit-column-rule: 5px dotted #0080ff; -moz-column-count:2; -moz-column-gap:30px; -moz-column-rule: 5px dotted #0080ff;
```

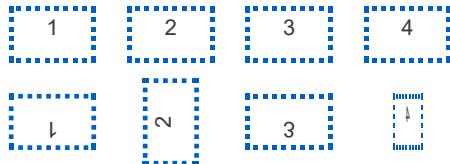
```
column-width: 100px; column-rule-style: dashed; width: 100px; -webkit-column-gap:30px; -moz-column-rule-color: #0080ff; -moz-column-rule-width: 5px;
column-gap:30px; column-rule-width: 5px; -webkit-column-gap:30px; -moz-column-rule-color: #0080ff; -moz-column-rule-width: 5px;
column-rule-color: #0080ff; width: 5px; -webkit-column-rule-color: #0080ff; -moz-column-rule-color: #0080ff; -moz-column-rule-style: dashed;
column-rule-style: dashed; -webkit-column-rule-style: dashed; -moz-column-rule-style: dashed; -moz-column-rule-width: 5px;
```

**Transition:** It can be used with any quantifiable attribute.

```
#transition1 { height:30%; background-color: #57b8ff; transition: background-color 1s, height 2s; }
```

```
#transition1:hover { height:100%; background-color: #ffa6e4; }
```

```
<div style="height:100px; position:relative; display:inline-block; float:right; width:30px;"> <div class="showbor" id="transition1" style="width:100%;"></div></div>
```



**Rotation:** .rotation0 { transition: transform 2s; height:30px; display:inline-block; width:50px; }

```
#rotation1:hover { transform: rotate(180deg); }
```

```
#rotation2:hover { transform: rotate(-90deg); }
```

```
#rotation3:hover { transform: rotateX(180deg); }
```

```
#rotation4:hover { transform: rotateY(-250deg); }
```

**Animation:** @keyframes anime1 { 0% { height:30%; background-color: #57b8ff; }

```
55% { height:70%; }
```

```
70% { height:70%; background-color: #ffa6e4; }
```

```
100% { height:100%; background-color: #ffa6e4; } }
```

```
animation-name: anime1; (animation name, must be unique)
```

```
animation-duration: 4s; (time to complete the animation, s = second and ms = miliseconds)
```

```
animation-delay: 0s; (time before the animation starts)
```

```
animation-timing-function: linear; (same speed throughout animation, I find that the % are better for setting different speeds so I always use this setting)
```

```
{ animation-name: anime;
animation-duration: 3s;
animation-delay: 0s;
animation-timing-function: linear;
animation-iteration-count: 1;
animation-direction: normal;
animation-play-state: running; }
```

```
@keyframes anime {
```



animation-iteration-count: infinite; (times the animation is repeat, 1, 2, etc. or infinite)  
animation-direction: alternate; (goes back and forth)  
animation-direction: normal; (play the animation one way)  
animation-direction: reverse; (play the animation backwards)  
animation-play-state: running; (self explanatory, the other option is paused)

```
0% { color: #57b8ff; }  
100% { color: #ffa6e4; }
```

**PHP:** PHP is executed server side. PHP variables start with \$. The function syntax and comparison symbols are very similar to Java Script. In PHP . can be used to add multiple variables and data to strings.

**Write code/content:** `<?php echo "..."?>`

**Insert code:** `<?php include("codefile.php"); ?>`

**Page redirect:** `<?php header('Location: index.php'); ?>`

**URL variable values:** Assign value: `index.php?page=1`  
Retrieve value: `<?php $variable = $_GET["page"]; ?>`

```
date('m/d/Y') (full date)  
date("d") (days of the month)  
date("l") (days of the week)  
date("N") (days of the week numbers)  
date("F") (months)  
date("Y") (full year)
```

**Case switch:** Break makes the code stop when it reaches a condition it meets.

```
<?php switch ($variable) { case "value1": echo "code here"; break; case "value2": include("page1.php"); break; default: header('Location: page.php'); } ?>
```

```
<a href="code.php?testvariable=value1">Set to value1</a> | <a href="code.php?testvariable=value2">Set to value2</a> | <a href="code.php">Set to null</a> <?php $testvariable = $_GET["testvariable"]; switch ($testvariable) { case "value1": echo "testvariable is set to value1"; break; case "value2": echo "testvariable is set to value2"; break; default: echo "testvariable is not set"; } ?>
```

**Create a cookie:** Expiration: 3600 = 1 hour `<?php setcookie("name", "value", time()+3600); ?>`

**Set cookie by link:** Make a php file and put only this code in it. It will set the cookie and redirect the page. The cookie will save the current path and work only on that folder. `<?php setcookie("name", "value", time()+7200); header('Location: page.php'); ?>`

**Set cookie for all subfolders:** `<?php setcookie("name", "value", time()+7200, "/"); header('Location: page.php'); ?>`

**Insert code based on cookie value:** `<?php if ($_COOKIE["name"] == "value") { code here } else { code here } ?>`

**Set:** `<?php setcookie("testcookie", "testcookie is set", time()+7200, "/"); header('Location:http://...'); ?>`

**Delete:** (empty value and past date) `<?php setcookie("testcookie", "", time()-7200, "/"); header('Location:http://...'); ?>`

**Gallery Script: CSS:** `.gallery { margin: 20px 10px 0px 10px; width:50px; height:50px; border: 0px none; display:inline-block; }`

**PHP:** `<?php $page = $_GET["page"]; if ($page == null) { $page=1; } ?> Page <?php echo $page; ?> <br /> <?php $perpage = 6; $dirs = "pixel/";`

```
$startat = 2 + ($perpage * ($page - 1)); $pics = scandir($dirs); $thepics = array_slice($pics, $startat, $perpage); foreach ($thepics as $item) { if (($item != ".") && ($item != "..")) { echo '<a target="_blank" href="' . $dirs . $item . '"><div class="gallery" style="background-image:url(' . $dirs . $item . ');></div></a>'; } } ?>
```

```
<br /><br /> <?php $totalpic = 0; foreach ($pics as $item) { if (($item != ".") && ($item != "..")) { $totalpic = $totalpic + 1; } } $lastp = $totalpic / $perpage; $lastp = ceil($lastp); $linkp = 1; echo "| "; while ($lastp>0) { echo '<a href="?page=' . $linkp . '">'. $linkp . '</a> | '; $linkp=$linkp+1; $lastp=$lastp-1; } ?>
```

**Explanation:** `.gallery { margin: 20px 10px 0px 10px; width:50px; height:50px; border: 0px none; display:inline-block; }` Thumbnail style, edit the size and shape.

`<?php $page = $_GET["page"]; if ($page == null) { $page=1; } ?>` Get the variable from the url (gallery.php?page=2), and if it's not there, start on page 1.

`Page <?php echo $page; ?> <br /><br />` Page title and number.

`<?php $perpage = 35; $dirs = "gallery/";` \$perpage is the number of pictures on each page. \$dirs is the directory where they are.

`$startat = 2 + ($perpage * ($page - 1));` The first 2 items in the directory will be . and .. hence a directory with 10 pictures will have an array of

12. We don't need those, so we start at 2, where the actual files start. To that we add the number of pictures per page from previous pages, minus the current one. If \$perpage is 25, Page 1, will start at 2 and show 25 pictures, Page 2 will start at 27 and show 25 more pictures. Since the first 2 don't count, you'll see pictures 1 to 25 on page 1, 26 to 50 on page 2, etc. in alphanumeric order.

`$pics = scandir($dirs); $thepics = array_slice($pics, $startat, $perpage); foreach ($thepics as $item) {` Get the pictures and divide them into pages. The slice code means from this directory, starting here, display this many items.

`if (($item != ".") && ($item != "..")) {` Make sure you have the proper file names (we already skipped . and .. so it's not really needed, but I like to keep it just in case.)

`echo '<a target="_blank" href=" . $dirs . $item . "'><div class="gallery" style="background-image:url(' . $dirs . $item . ');"></div></a>; } } ?>` This code will repeat for each picture, up to the last in that page.

`<br /><br /> <?php $totalpic = 0; foreach ($pics as $item) { if (($item != ".") && ($item != "..")) { $totalpic = $totalpic + 1; } }` To automatically create links to the other pages, we'll need the total number of files. So for each file on the folder, \$totalpic will get +1.

`$lastp = $totalpic / $perpage;` \$lastp is the total number of pages, so if there are 80 files (\$totalpic) and we're showing 25 per page (\$perpage) that's 80/25=3.2

`$lastp = ceil($lastp);` Round that number up, regardless of if it's .5 or less. So \$lastp will be 4. Since there are 80 pictures, page 4 will only have 5, and pages 1 to 3 will have 25 each.

`$linkp = 1; echo "1 ";` Let's generate the page links, starting at 1.

`while ($lastp>0) { echo '<a href="?page=' . $linkp . "'> . $linkp . '</a> | '; $linkp=$linkp+1; $lastp=$lastp-1; } ?>` Every time a link is created, add 1 to \$linkp, which is the page number in the url, and subtract 1 from \$lastp, which is the number of pages left to link. The codes does this as long as \$lastp is more than 0, so that links will stop being made when the last page is reached.

**Fields go inside the form tags:** `<form id="formid" action="#"> </form>`

**Disable editing on field:** `readonly="readonly"`

**Disable textarea resizing:** `textarea { resize: none; }`

**Default check for checkbox and radio buttons:** `checked="checked"`

**Text box:** `document.getElementById("boxname").value`

`<input class="cssclass" id="boxid" value="content" type="text" />`

This is a text box.

This is a textarea.

**Text area:** `document.getElementById("areaname").value`

`<textarea class="cssclass" id="areaid"> Message (value) </textarea>`

**Focus:** `input:focus { css here }` (changes while it's selected)

001 002

Icecream flavor:

Brownie  Chocolate  Cookiesn'Cream

Go!

**Check boxes and radio buttons:**

`document.getElementById("idname").checked==true`

`document.getElementById("idname").checked==false`

**Check box:** `<input type="checkbox" id="cbid" />`

**Radio button:** `<input type="radio" name="radname" id="rad1" />`

`<input type="radio" name="radname" id="rad2" />`

What now?

Videogames  Anime  Fanart

Go!

Snack?

Go!

**Combo box:**

`document.getElementById("comboname").value`

`<select class="cssclass" id="comboid">`

`<option value="nothing_selected" class="cssclass">Question?</option>`

`<option value="variable_value" class="cssclass">Option 1</option>`

`<option value="variable_value" class="cssclass">Option 2</option> </select>`

**PHP Email Form:** The fields in the form need to coincide with the code that will process them. In mailer.php the first part transfers the information from the form into variables. Next, there's the email it will be sent to and the subject of the email. The following part replaces certain characters and limits the length of fields for security. Finally, the variables are placed into the email and sent. The auto reply is optional and can be removed.

`<form name="mailing" action="mailer.php" method="post"> <input name="uname" id="uname" value="Name: " type="text" /> <input`

```
name="umail" id="umail" value="Email: " type="text" /> <input name="usite" id="usite" value="Site: " type="text" /> <input name="ururl" id="ururl" value="URL: " type="text" /> <textarea name="uspeak" id="uspeak" rows="3" cols="22">Message: </textarea> <input type="submit" value="Send" /> </form>
```

**mailer.php**

```
<?php $yourname = $_POST['uname']; $youremail = $_POST['umail']; $thesite = $_POST['usite']; $thesurl = $_POST['ururl']; $speak = $_POST['uspeak'];
$to = 'YOUR EMAIL'; $email_subject = "EMAIL SUBJECT";
$yourname = str_replace("=", "", $yourname); $yourname = str_replace("\'", "\&quot;", $yourname); $yourname = substr($yourname, 0, 30);
$youremail = str_replace("=", "", $youremail); $youremail = str_replace("\'", "\&quot;", $youremail); $youremail = substr($youremail, 0, 50);
$thesite = str_replace("=", "", $thesite); $thesite = str_replace("\'", "\&quot;", $thesite); $thesite = substr($thesite, 0, 30);
$thesurl = str_replace("=", "", $thesurl); $thesurl = str_replace("\'", "\&quot;", $thesurl); $thesurl = substr($thesurl, 0, 100);
$speak = str_replace("=", "", $speak); $speak = str_replace("\'", "\&quot;", $speak); $speak = substr($speak, 0, 7000);
$email_body = "Name: $yourname \n Email: $youremail \n Site: $thesite \n Url: $thesurl \n Message: $speak";
mail($to, $email_subject, $email_body);
$autoreply = "AUTO REPLY TEXT";
$yourmail = $youremail; $yourmail = str_replace('Email: ', "", $yourmail); $yourmail = str_replace('Email:', "", $yourmail);
mail($yourmail, 'AUTO REPLY SUBJECT', $autoreply);
header('Location: URL AFTER EMAIL IS SENT'); ?>
(Error page CSS & HTML goes here.)
```

*/\* This is a comment among JS. \*/*

**Java Script CSS:** background-color is backgroundColor and so on  
Delete the - and add capitals to the second word to change CSS to JS.

```
document.getElementById("cssjs02").style.backgroundImage = "url('back.png')";
```

Operation Symbols		
=	&&	
assign value	and	or

**Functions:** function fuctname() { actions... }  
function fuctname() { if (condition) { action } if (condition) { action } else { action } }

```
id="mouse3"
onmouseover="change4();"
onmouseout="change5();"

id="mouse2"
onclick="change7();"
```

**Mouse events:** onmouseover="function();" onmouseout="function();" onclick="function();"

```
function change4() { document.getElementById("mouse3").src="design/Parfait_Oreo.gif"; }
function change5() { document.getElementById("mouse3").src="design/Parfait_Choco_Chip.gif"; }
```

```
var clickit="candy"; function change7() { switch (clickit) {
case "candy": document.getElementById("mouse2").src="design/Parfait_Red_Velvet.gif";
clickit="sweet"; break;
```

```
case "sweet": document.getElementById("mouse2").src="design/Parfait_Chocolate_2.gif"; clickit="candy"; break;
default: document.getElementById("mouse2").src="design/Parfait_Chocolate_2.gif"; clickit="candy"; break; } }
```

```
onchange="function();"
function change8() { document.getElementById("cha1").style.backgroundColor = "#0047ab";
document.getElementById("cha1").style.color = "#87cefa"; }
```

**Reset**

```
Click this div to see the mouse position.
Mouse position Y:
Mouse position X:
```

**Mouse Position:** function trackmouse(event) { event.clientX; event.clientY;  
document.getElementById("posy").innerHTML = event.clientY;  
document.getElementById("posx").innerHTML = event.clientX; }

**Mouse Sparkles:** Disable loops on the gif so it plays once.  
<body onclick="sparkling();"> 

```
function sparkling() { var sparktop = event.clientY - 15;
var sparkleft = event.clientX - 15;
document.getElementById("spark3").style.top = sparktop + "px";
document.getElementById("spark3").style.left = sparkleft + "px";
document.getElementById("spark3").src = "design/sparkles3.gif"; }
```

Comparison Symbols		
<	>	==
less	greater	equal
<=	>=	!=
less/equal	greater/equal	not equal

**Alerts:** alert("text here " + variable + "more text..." + "\n"); \n (new line)  
**JS Alert Popup** var listen = "Listen!"; function heylisten() { alert( "Hey! \n" + listen ); }

**JS Intervals:** 1000 = 1 second | 60000 = 1 minute

**Repeats actions at set time intervals:** `idvariable = setInterval( function fname() { ... }, 1000);`

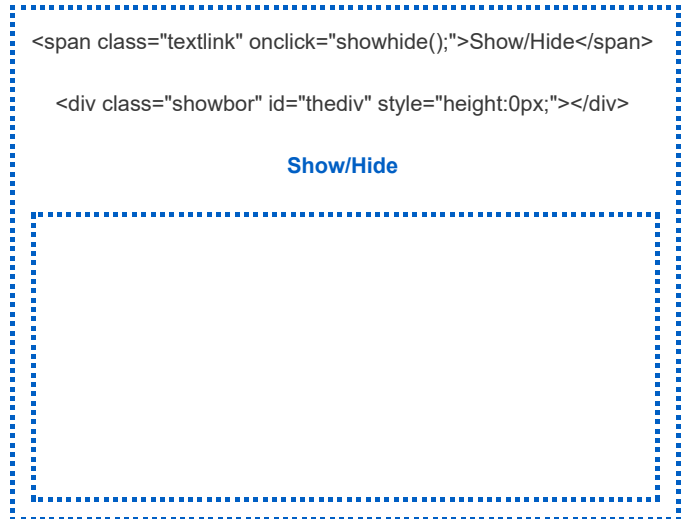
**Stop the interval from repeating:** `function fname() { clearInterval(idvariable); }`

**Performs an action once after a set amount of time:** `idvariable = setTimeout( function fname() { ... }, 3000);`

**Stop Automatically:** `function fname1() { thetimer = setInterval( function fname2() { ... }, 1000); setTimeout( function fname3() { clearInterval(timer); }, 5000); }`

**Show or hide page sections:** `<script>`

```
var speed = 3;
var fullsize = 100;
var newsize = 0; var compute; var timesize;
function showdiv() {
  timesize = setInterval( function countin() {
    newsize = newsize + speed;
    compute = fullsize - newsize;
    document.getElementById("thediv").style.height = newsize + "px";
    if (compute < speed) { clearInterval(timesize);
    document.getElementById("thediv").style.height = fullsize + "px";
    newsize = fullsize; } }, 15); }
function hidedit() {
  timesize = setInterval( function countout() {
    newsize = newsize - speed;
    document.getElementById("thediv").style.height = newsize + "px";
    if (newsiz < speed) { clearInterval(timesize);
    document.getElementById("thediv").style.height = "0px";
    newsize = 0; } }, 15); }
function showhide() { if (newsiz == 0) { showdiv(); } if (newsiz == fullsize) { hidedit(); } }
</script>
```



```
document.getElementById("js01").innerHTML = document.getElementById("js00").clientHeight; 50
document.getElementById("js02").innerHTML = document.getElementById("js00").clientWidth; 30
```

```
var somevar1 = "KAWAII"; document.getElementById("js03").innerHTML = somevar1.toLowerCase(); kawaii
```

```
var somevar2 = "tech"; document.getElementById("js04").innerHTML = somevar2.toUpperCase(); TECH
```

```
var somevar3 = "Chocolate Cake"; document.getElementById("js05").innerHTML = somevar3.length; 14 (Characters and spaces)
```

```
var somevar4 = "many words"; document.getElementById("js06").innerHTML = somevar4.match("word"); word (Must be same case)
```

```
var somevar5 = "M?"; somevar5 = somevar5.replace(/M/g, "01001101 "); somevar5 = somevar5.replace(/&?/g, "00111111 ");
document.getElementById("js07").innerHTML = somevar5; 01001101 00111111 (\ is used to escape characters)
```

```
var somevar6 = 2.7; somevar6 = Math.floor(somevar6); document.getElementById("js08").innerHTML = somevar6; 2 (round down)
```

```
var somevar7 = 2.3; somevar7 = Math.ceil(somevar7); document.getElementById("js09").innerHTML = somevar7; 3 (round up)
```

```
var somevar9 = 2.145; somevar9 = parseFloat(somevar9).toFixed(2); document.getElementById("js11").innerHTML = somevar9; 2.15
```

```
var somevar8 = Math.floor((20)*Math.random()+1); document.getElementById("js10").innerHTML = somevar8; 11 (random number 1-20)
```

```
var somevar10 = "0A"; somevar10 = parseInt(somevar10); document.getElementById("js12").innerHTML = somevar10; 0
```