CSS

Cascading Style Sheets

What is CSS

Cascading Style Sheets (CSS) is a style sheet language used for describing the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.

Why CSS

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

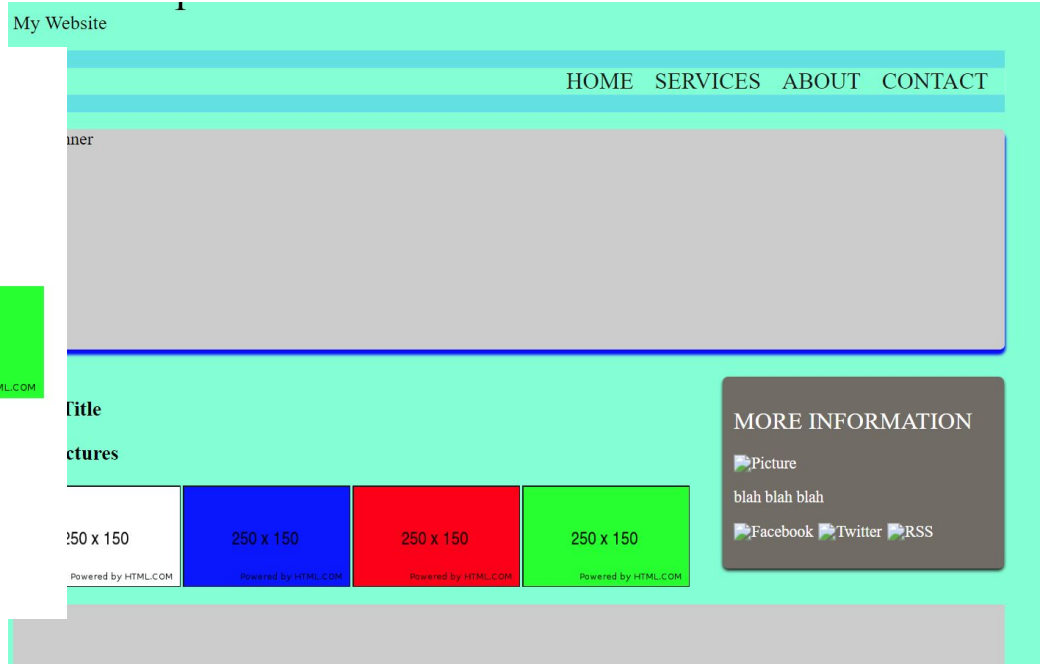
- Determines how elements will be displayed
- Makes your content look GOOD
- CSS saves time and work, can control multiple pages
- CSS can add animation
- CSS can add interaction
- Adapt the way your page looks with CSS, perfect for responsive design

Different browsers

https://en.wikipedia.org/wiki/Comparison_of_browser_engines_(CSS_support)

- some browsers may have different levels of support for technology features to other
- sometimes browsers have bugs, or implement features differently.

Different Look same HTML



CSS reference

https://developer.mozilla.org/en-US/docs/Web/CSS

https://css-tricks.com/almanac/

Keyword index

Note: The property names in this index do **not** include the JavaScript names where they differ from the CSS standard names.

A

:active
additive-symbols (@counter-style)
::after (:after)
align-content
align-items
align-self
all
<an-plus-b>
<angle>
animation
animation-delay
animation-direction
animation-duration
font-language-override
font-optical-sizing
font-size
font-size-adjust
font-stretch
font-stretch (@font-face)
font-style
font-style (@font-face)
font-synthesis
font-variant
font-variant (@font-face)
font-variant-alternates
font-variant-caps
font-variant-east-asian
padding-block-end
padding-block-start
padding-bottom
padding-inline-end
padding-inline-start
padding-left
padding-right
padding-top
@page
page-break-after
page-break-before
page-break-inside
pc
<percentage>

```
selectorlist {
  property: value;
  [more property:value; pairs]
}
```

Style rule examples

```
1  strong {
2    color: red;
3  }
4
5  div.menu-bar li:hover > ul {
6    display: block;
7  }
```

Basics of Styling

```
<link rel="stylesheet" href="style1.css" />
<style>
h1 {color: blue;}
</style>
<h1><font color="green">Test Text 1</font></h1>
<h1 style="color: red;">Test Text 2</h1>
<h1>Test Text 3</h1>
<h1>Test Text 4</h1>
```

example1.html / style1.css

Inline styles

CSS styles to a single, specific element. Using an attribute style you can apply the styling to the element, and all child elements. Notice styling works from the element out, so less direct styling is overwritten as it goes out from the element.

example2.html

```
<p style="color: red;">Test Text 1</p>
<p style="color: #FF0000; font-size:16px; font-
weight:bold ">Test Text 2</p>
<p style="color: #FF0000; font-size:24px">Test Text
3</p>
<p style="color: #FF0000; font-size:1em">Test Text
4</p>
<div style="color:blue;">
<div style="color:yellow">Test1</div>
<div style="font-size:28px">Test2</div>
</div>
```

Try it - inline

- Open you editor
- Create HTML elements
- Change the color
- Change the size

Internal styles

Use the style tag to add styling. This is where selection is really important. As you can see much of CSS is about making selections of the elements.

example3.html

```
<head>
    <title>Example 2</title>
    <style>
        p {
            color: red;
        }

        p:not(:first-child) {
            font-size: 16px;
            font-weight: bold;
        }

        div {
            color: blue;
        }
    </style>
</head>

<body>
    <p>Test Text 1</p>
    <p>Test Text 2</p>
    <p>Test Text 3</p>
    <p>Test Text 4</p>
    <div>
```

Try it

- Open you editor
- Create HTML elements
- Within the style tag select the element by tag and change the color and size.
- Create various different tags apply style.
- Nest tags within other tags, select the tag to add styling

CSS Style external link

Benefits of linking to an external CSS file

- everything is stored within a single file
- once changed/updated, the changes are reflected on all other pages that reference the stylesheet
- makes it easier to maintain larger websites

example4.html

```
!DOCTYPE html>
html>

head>
   <title>Example 4</title>
   <link rel="stylesheet" href="style4.css" /> </head>

body>
   <p>Test Text 1</p>
   <p>Test Text 2</p>
   <p>Test Text 3</p>
   <p>Test Text 4</p>
   <div>
       <div>Test1</div>
       <div>Test2</div>
   </div>
```

Try it

- Open you editor
- Create HTML elements
- Link to an external CSS file

Selection Classes and ID

Classes and IDs are used for selection of elements.

Use classes for the ability to apply styling to multiple elements.

Use ID for applying styling to one specific element.

example5.html

```
    <style>
        .red {
            color: red;
        }

        .blue {
            color: blue;
        }
    </style>
/head>

body>
    <p class="red">Test Text 1</p>
    <p class="red">>Test Text 2</p>
    <p class="blue">>Test Text 3</p>
    <p  class="red">>Test Text 4</p>
    <div  class="blue">>
        <div>Test1</div>
        <div>Test2</div>
    </div>
```

Try it

- Open you editor
- Create HTML elements
- Select elements using the class and id attributes.
- Apply multiple classes to the same element.

Color options

https://developer.mozilla.org/en-US/docs/Web/CSS/color_value

http://paletton.com/

You can use :

- Keyword color ‘red’
- RGB Hex value ‘#ff0000’
- RGB value ‘rgb(255, 0, 0)’
- RGB value ‘rgba(255, 0, 0,.5)’

example6.html

Specification	RGB equivalent	Keyword	RGB hex value	Live keyword
		black	#000000	
		silver	#c0c0c0	
		gray	#808080	
		white	#ffffff	
		maroon	#800000	
		red	#ff0000	
		purple	#800080	
		fuchsia	#ff00ff	
CSS Level 1		green	#008000	
		lime	#00ff00	
		olive	#808000	
		yellow	#ffff00	
		navy	#000080	
		blue	#0000ff	
		teal	#008080	
		aqua	#00ffff	
CSS Level 2 (Revision 1)		orange	#ffa500	

Background Colors / Images

Using divs and spans can help with the selection of elements.

Example8.html

Background shorthand

https://developer.mozilla.org/en-US/docs/Web/CSS/background

```
  .green3 {
      color: rgb(0, 255, 0);
      background-color: yellow;
  }

  .green4 {
      color: rgba(0, 255, 0, 0.5);
      background-color: aqua;
  }
style>
```

CSS display property

https://developer.mozilla.org/en-US/docs/Web/CSS/display

Example : display.css

```
/* <display-outside> values */
display: block;
display: inline;
```

```
/* <display-box> valu
display: contents;
display: none;
```

```
/* <display-legacy> values */
display: inline-block;
display: inline-table;
display: inline-flex;
display: inline-grid;
```

```
/* <display-listitem> values */
display: list-item;
display: list-item block;
 splay: list-item inline;
 splay: list-item flow;
 splay: list-item flow-root;
 splay: list-item block flow;
 splay: list-item block flow-root;
 splay: flow list-item block;
```

List items

Manipulate list items.

https://developer.mozilla.org/en-US/docs/Web/CSS/list-style-type

list-style: none;

display: inline-block;

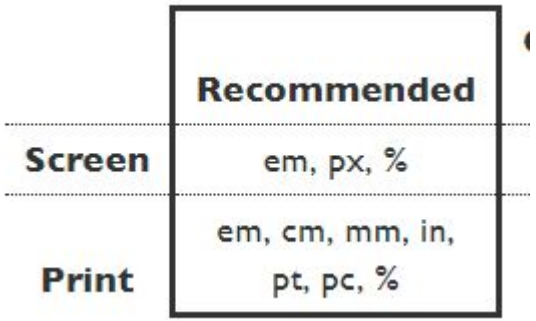
Exercise #1

Fun with colors

1. Open editor
2. Open exercise1.html
3. Create new exercise1.css file
4. Apply CSS properties and values to different element groups
5. Have fun with colors, background and text.
6. Create simple website

CSS measures and properties

https://www.w3.org/Style/Examples/007/units.e
l



Percentage (%) - expressed as a percentage in relation to the containing

Pixel Unit (PX) - Fixed unit size in pixels dots on your computer screen.

EM unit (EM) - Scalable unit. em unit is relative to the font-size of the pa

REM unit (REM) - Scalable unit. The rem unit is relative to the root html element.
*not used often

Divs and Spans

Using divs and spans can help with the selection of elements.

example7.html

Styling text

http://www.blindtextgenerator.com/snippets?snipps=ANY-snippets-lorem

example13.html

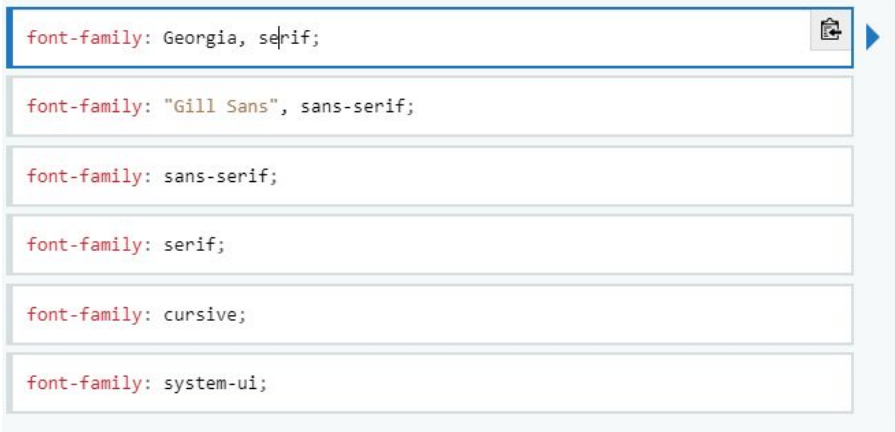
```
.test1 {
    text-align: center;
}

.test2 {
    text-align: left;
    font-style: italic;
}

.test3 {
    text-align: right;
    font-weight: bold;
}

.test4 {
    word-spacing: 20px;
    font-size: 2.5em;
}
```

CSS Font Family

Using divs and spans can help with the selection of elements.

Example9.html

https://developer.mozilla.org/en-US/docs/Web/CSS/font-family



Google Fonts

https://fonts.google.com/

- Text-align
- Font-style
- Font-weight
- Font-size
- font-family

```
.highlight{
    background-color: yellow;
}

.bigText{
    font-size:200%;
    font-weight: bold;
    font-family: fantasy;
    font-style: italic;


}
.oneText{
    text-align: center;

}
li a{
    text-decoration: none;
}
.twoText{
    line-height: 30px;
}
```

Exercise #2

Fun with text

1. Open editor
2. Open exercise2.html
3. Create new exercise2.css file
4. Apply CSS properties and values to different element groups
5. Update the fonts and colors
6. Create your mini webpage

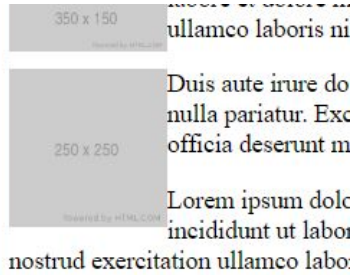
Images CSS floats

USe float to position images inline with text.

Try max-width to limit the size of an image.

example10.html

```
<title>Example 10</title>
<style>
    img{
        float:left;
        max-width: 100px;
    }
</style>
```

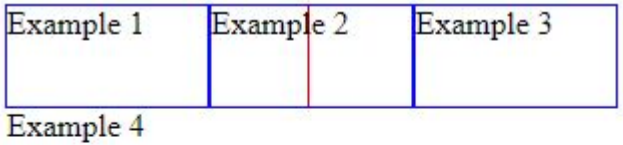


CSS floats

Floats are not just for images.

Clearing of floats

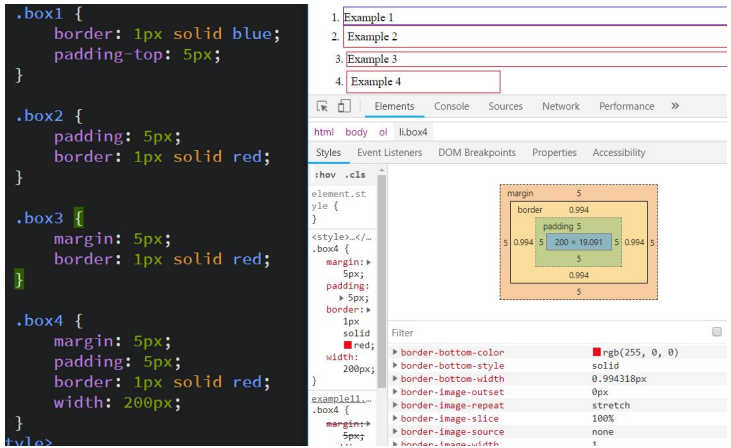
example11.html

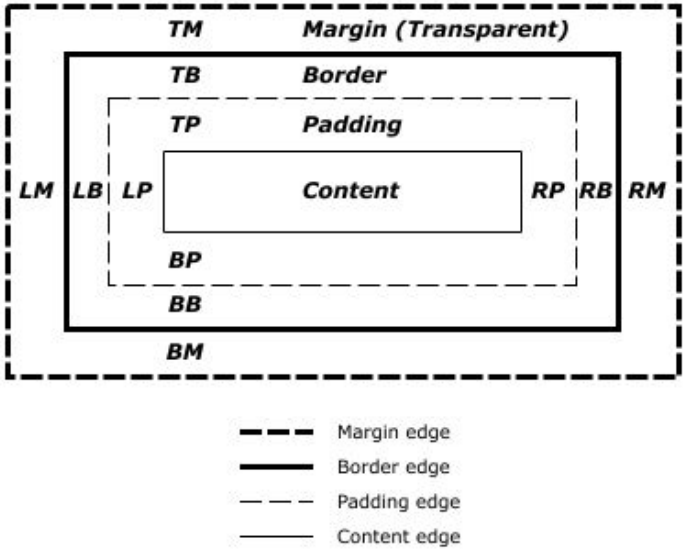


BOX Model

https://www.w3.org/TR/CSS2/box.html

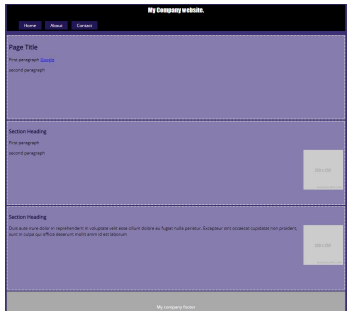
- Border, padding, margin, width/height
- Chrome inspector





Exercise #3

1. create background of body to color #3C3176;
2. make H2 #10073B
3. make section background #867DB0 with dotted border. add small margin top/bottom only to separate the sections. Padding for slight indent of text. Add a minimum height to the sections
4. create nav bar - using padding and colors #231858 background #FFF font;
5. make logo font color white, align center, font 6. size 1.5em and font style fantasy
import your favorite google font and add to the body
7. footer padding and centered text with grey background
8. images to max size of 150px and to be right aligned

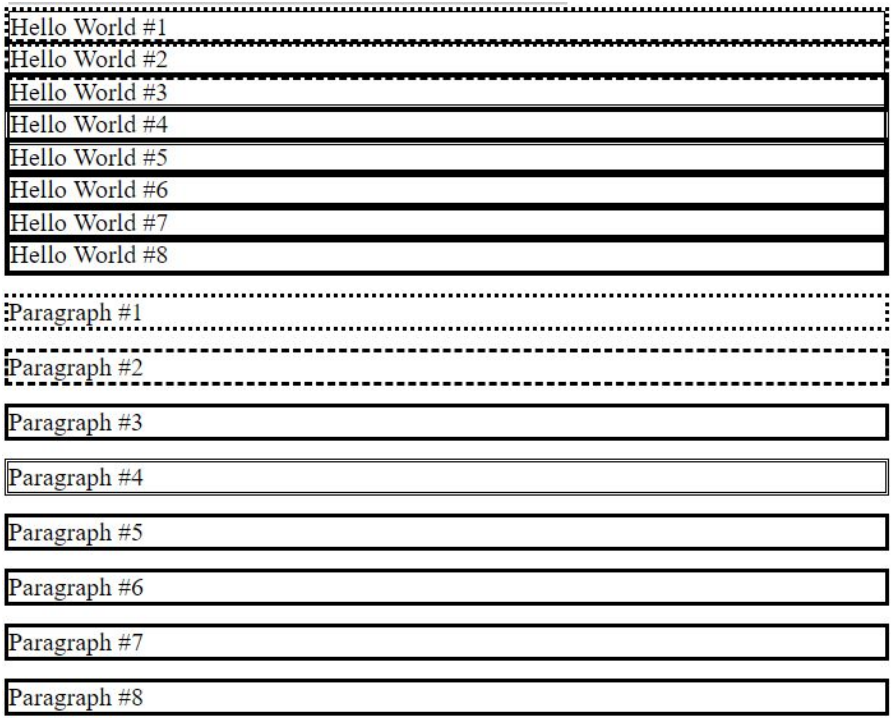


CSS Outline

The outline-style property specifies the style of the outline.

Example outline.css

```
.test1 {outline-style: dotted; }
.test2 {outline-style: dashed;}
.test3 {outline-style: solid;}
.test4 {outline-style: double;}
.test5 {outline-style: groove;}
.test6 {outline-style: ridge;}
.test7 {outline-style: inset;}
.test8 {outline-style: outset;}
```



CSS positioning

https://developer.mozilla.org/en-US/docs/Web/CSS/position

- **Static :** The default value places the item in the normal flow
- **Relative :** The item is placed in the normal flow, and then shifted or offset from that position. Subsequent flow items are laid out as if the item had not been moved.
- **Absolute :** Specifies absolute positioning. The element is positioned in relation to its nearest non-static ancestor.
- **Fixed :** The item is absolutely positioned in a fixed position on the screen even as the rest of the document is scrolled

example12.html

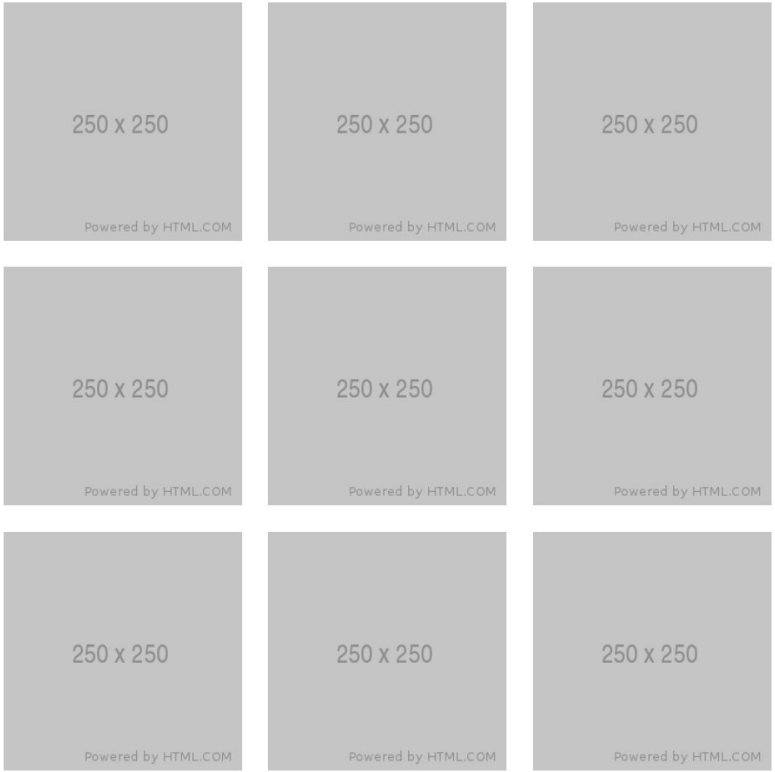
CSS positioning

- left,top,right,bottom

example12.html

Exercise #4

Making the image Grid?



CSS for hyperlinks

a:link - default - unvisited link

a:visited - visited link

a:hover - when the mouse cursor is over the link

a:active -when the link is clicked

Example links.css

CSS Pseudo Class

https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes

A CSS pseudo-class is a keyword added to a selector that specifies a special state of the selected element(s). For example, :hover can be used to change a button's color when the user hovers over it.

```
a:active{
    color:aqua;
}
div:hover{
    background-color: yellow;
    font-size: 2em;
}
  .container div:last-child{
    background: purple;
}*/
.container div:nth-child(2){
    background: yellow;
}
```

CSS for tables

The way a table is styled should be done using CSS. Lots of options to make your table look good.

Example tables.css

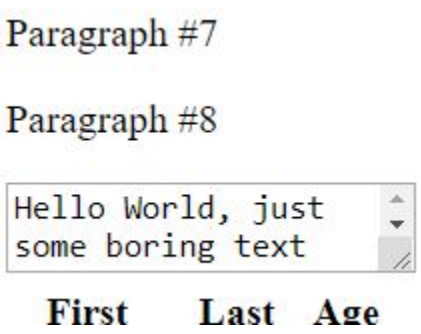
First	Last	Age
Laurence	Svekis	40
John	Smith	50
Jane	Johnson	33
Mike	Jones	25

Overflow and MaxWidth/Height

Overflow tells the browser how to handle content that does not fit within the element.

Max width and height force on responsive when available to restrict the height value to the max.

Example : *overflow.css*



Align Elements

Aligning of elements can be done with several options available including margins, padding, display positions.

Example align.css