

CSS



CSS?

- Stands for **C**ascading **S**tylesheet
- How HTML elements are displayed
 - Formatted to different media



CSS - Files

CSS can be imported (in the head) from a **.css** file

```
<link rel="stylesheet" type="text/css" href="main.css">
```

Or it can be coded in the **<style>** tag

```
<style>
  html {
    margin: 0;
    color: rgb(0,0,0);
  }
</style>
```

CSS - Files

Also it can be added to each tag (in the body) using “**style**” attribute

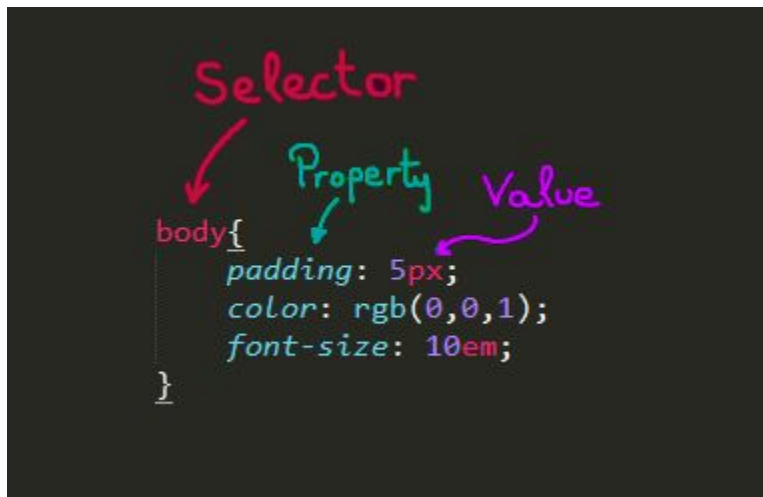
```
<p style="font-size: 10em; margin-left: 30px;">Some text.</p>
```

However, it is **not** recommended because:

- Higher priority
- Organization / visualization of styles
- More difficult to change
- Unnecessarily mix HTML with CSS

CSS - Syntax

- Selectors: HTML elements, classes or IDs
- Property: what is modified (from default)
- Value



CSS - Selectors

- HTML Elements: h1, a, html, body
- ID
 - Unique
 - #container, #img-logo
- Class
 - For groups
 - A tag can have one class or more
 - .highlight, .footer-img

CSS - Selectors

- Priorities
 - Inside the CSS
 - ID
 - Class
 - Element
 - !important
- Loading styles
 - Inline styles
 - Internal stylesheets
 - External stylesheets
 - Browser styles

CSS - Selectors

```
/* NAVIGATING */
```

```
article p {} /* All <p> inside <article> */
```

```
nav > ul {} /* ALL <ul> that are direct children of <nav> */
```

```
span.im-text {} /* ALL <span> with class "im-text" */
```

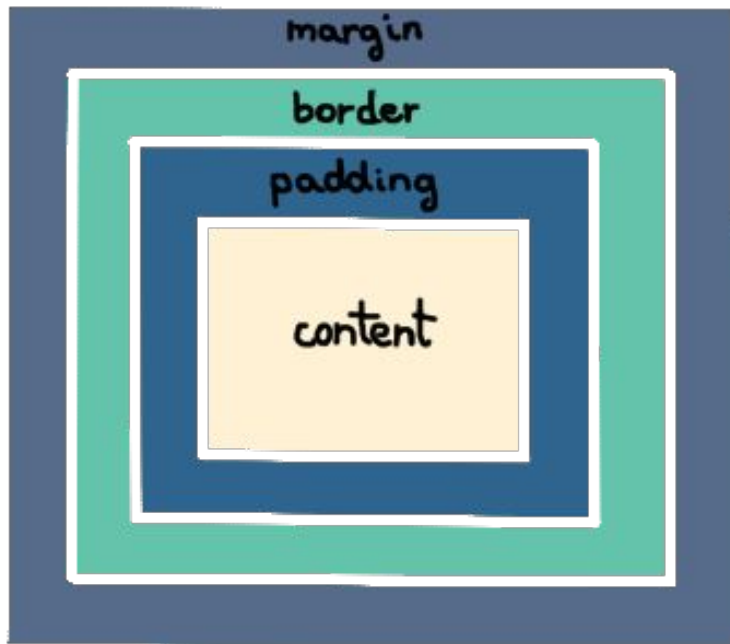
```
.nav-text + span {} /* ALL <span> immediately after something with class "nav-text" */
```

```
ol:first-child {} /* The first child of every <ol> in the document */
```

```
/* GROUPING SELECTORS */
```

```
li#special, p.head-text {} /* ALL <li> with ID "special" and all <p> with class "head-text" */
```


CSS - Margin & padding



CSS - Position

- Describes the type of positioning method of the element
 - static: render in order
 - absolute: relative to its first ancestor not static
 - relative: relative to its static position (affected by left, top ...)
 - fixed: relative to browser window
 - sticky: relative + fixed
 - needs a container
 - needs at least one: top, bottom, left, right

CSS - Floats

- To float elements
- Does **not** work with absolute and fixed positions
- Related properties
 - clear: to prevent elements to float around
 - overflow: to fix the element if it overflows outside the container
 - Check **clearfix** hack

CSS - Display

- To change display behavior of a tag
 - inline
 - block
 - inline-block: like inline but with block advantages (resizing, margin,...)
 - flex
 - grid

CSS - Flexible Box

- To organize containers in an easy way
- The container **parent** needs to be “display: flex”
- Children **position** is set in the **parent**

```
#flex-parent {  
    display: flex;  
    justify-content: center;  
    flex-direction: column-reverse;  
}
```

CSS - Grid Layout

- To make flexible tables easily with divs

```
.grid-container {  
  display: grid;  
  grid-template-columns: auto auto auto;  
  padding: 2px;  
  background-color: rgb(0,0,0);  
}  
.grid-container div{  
  background-color: rgb(255,255,255);  
  margin: 5px;  
  font-size: 30px;  
  text-align: center;  
}
```

```
<div class="grid-container">  
  <div>1</div>  
  <div>2</div>  
  <div>3</div>  
  <div>4</div>  
  <div>5</div>  
  <div>6</div>  
  <div>7</div>  
  <div>8</div>  
  <div>9</div>  
</div>
```

1	2	3
4	5	6
7	8	9

CSS - Pseudo-classes

- To capture events, to get the state of an element
- Order is important!

```
a:link {} /* Unvisited link */  
  
a:visited {} /* Visited link */  
  
a:hover {} /* Mouse over link */  
  
a:active {} /* Selected link */
```

CSS - @media

- To adapt the content to different media type/devices
 - screen sizes
 - other media
- Rule that checks
 - width and height of viewport and device
 - orientation (landscape/portrait)
 - resolution

```
@media screen and (min-width: 800px) {  
  body {  
    background-color: rgb(0, 1, 1);  
  }  
}
```


CSS - Other

- Adding fonts
- Other cool properties

```
@font-face {  
    font-family: mywebfont;  
    src: url("webfont.woff");  
}
```

```
body{  
    font-family: mywebfont;  
}
```

```
div {  
    overflow: scroll;  
    border-radius: 25px;  
    text-shadow: 2px 2px;  
    box-shadow: 2px 2px;  
    font-smooth: antialiased;  
    opacity: 0.5;  
    box-sizing: border-box;  
}
```

