



# Introduction to CSS

Sesiunea 3 - 26/04/2021





/Recap\



# Recap

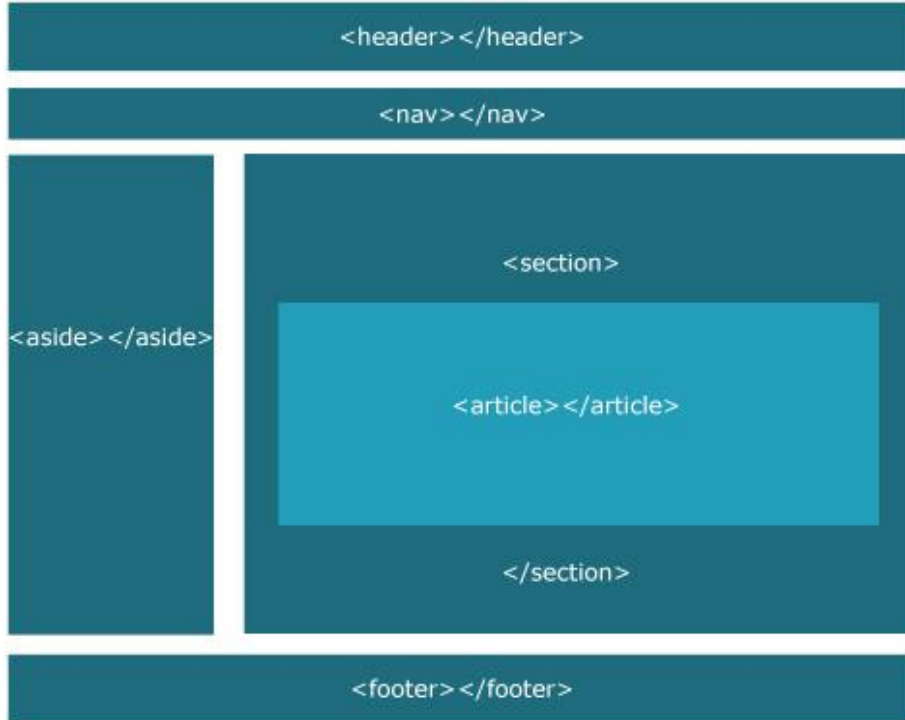
- Basic HTML structure
- Meta tags
- Semantic and non-semantic tags
- Attributes
- Navigation



/Semantic tags: visual representation\



# Semantic Web



- Semantic tags are about giving logical meaning
- The presentation of the content is still something the dev needs to cover
- Main benefits are:
  - SEO
  - Screen readers to cover impaired users
  - Easier navigation
  - Code reusability

# /Navigation\



# Navigation

- Between pages
- Between different websites
- Between sections of the same page (anchors or bookmarks)
- Resources retrieval

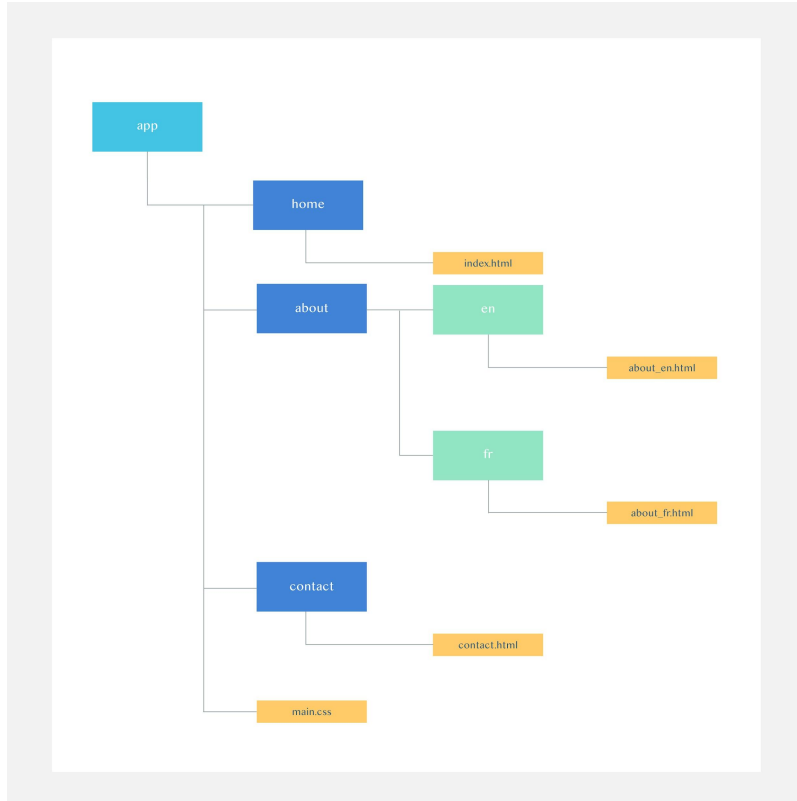


# Navigation: <a> tag

```
<a target="_blank|_self|_parent|_top|framename" href="www.google.com">Ask Google</a>
```



# Navigation: relative paths - navigating folder hierarchy



- Access main.css from index.html: ../main.css
- Access index.html from about\_en.html: ../../home/index.html
- Access main.css from about\_fr.html: ../../../../main.css

# Navigation: bookmarks - navigating the same page

```
<a href="#contact">Contact</a>
```

```
<section id="contact" class="contact-info">This is my contact section</section>
```



/We can do more: Presentation\



# Introduction to **CSS**

- Cascading Style Sheets - a presentational language
- Developed in levels and we are currently at CSS3
- Offer a set of powerful tools for amazing UI creations
- Enables the separation between presentation and content (HTML)
- Includes: layout, palettes, fonts
- There is no magic, just a **priority scheme** to take into account
- And **SELECTORS**



# CSS: types

- Inline styles
- Internal style sheets
- External style sheets



# CSS: inline styles

- Tool: the style attribute
- This type of styling will override any other style
- CSS rules are directly written inside the starting tag using the style attribute
- It'll include a series of CSS property and valid value pairs

# CSS: inline styles

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Page title - Hello, World!</title>
5  </head>
6  <body>
7    <div
8      style="width: 150px;
9        height: 150px;
10       background-color: #280f36;
11       color: #ffc1a0;
12       text-align: center;
13       line-height: 150px">
14      I am coloured div.
15    </div>
16  </body>
17 </html>
```

# CSS: internal style sheets (embedded)

- Tool: the <style> tag
- Defined in the <head> section of the HTML doc inside a {} block (we'll talk selectors in a bit)

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Page title - Hello, World!</title>
5          <style>
6              div {
7                  width: 150px;
8                  height: 150px;
9                  background-color: #280f36;
10                 color: #ffc1a0;
11                 text-align: center;
12                 line-height: 150px
13             }
14         </style>
15     </head>
16     <body>
17         <div>I am coloured div.</div>
18     </body>
19 </html>
```



# CSS: external style sheets

- Tool: the <link> tag (rel, type, href attributes)
- Defined in the <head> section of the HTML doc
- Website speed performance optimization
- The <link> tag defines the between a doc and an external resource
- It's self enclosing tag
- An external CSS can be created once and used on multiple pages
- The content can be treated differently on multiple UIs (e.g. mobile)
- Higher page ranking (SEO)

# CSS: external style sheets

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Page title - Hello, World!</title>
5          <link rel="stylesheet" type="text/css" href="assets/main.css">
6      </head>
7      <body>
8          <div>I am coloured div.</div>
9      </body>
10 </html>
```

```
1  div {
2      width: 150px;
3      height: 150px;
4      background-color: red;
5      color: #ffc1a0;
6      text-align: center;
7      line-height: 150px
8  };
```

# CSS: when to use what

- The inline CSS is limited (does not allow styling pseudo-elements and link states) and it leads to messy code (so, it's a no!)
- To be used only for: testing, quick fixes, email templates, dynamic content (JS generated)
- Internal style sheets can be used for styling a single page (time consuming for more than one page and for repetitive styling needs)
- The external style sheet is the most preferable solution (so use that as much as possible)
- The inline CSS will always override the other ones, but it's still a no (the priority scheme is preferred - we will talk more about it in a bit)

# CSS: external style sheet do have disadvantages

- Pages might not be rendered correctly until the external CSS is loaded
- Linking multiple CSS files can increase the site's download time
- It's still the preferred way

# HTML/CSS: comments

```
<!-- Type selector -->
<!--
|   It can be multiline also
-->
<div class="selectors-wrapper">
  <div class="selector-section">
    <div class="description-wrapper">
      <span class="text-weighted">Type</span>
      <span> selector:</span>
      <span class="text-weighted">a</span>
      <span> { }</span>
    </div>
    <div class="circles-wrapper">
      <p class="circle">p</p>
      <div class="circle">div</div>
      <a class="circle">a</a>
    </div>
  </div>
<!-- End of Type Selector -->
```

```
/* CSS - type selector */
/*
|   It can be multi line also
*/
.circles-wrapper a {
  background-color: ■ #fe9c7f;
  color: ■ #fff;
  border-color: ■ #fff;
}
```



/What about all these colors?\



# CSS: color/background-color/border-color

IndianRed	CD5C5C	205,92,92	GreenYellow	ADFF2F	173,255,47	Cornsilk	FFF8DC	255,248,220
LightCoral	F08080	240,128,128	Chartreuse	7FFF00	127,255,0	BlanchedAlmond	FFEBCD	255,235,205
Salmon	FA8072	250,128,114	LawnGreen	7CFC00	124,252,0	Bisque	FFE4C4	255,228,196
DarkSalmon	E9967A	233,150,122	Lime	00FF00	0,255,0	NavajoWhite	FFDEAD	255,222,173
LightSalmon	FFA07A	255,160,122	LimeGreen	32CD32	50,205,50	Wheat	F5DEB3	245,222,179
Crimson	DC143C	220,20,60	PaleGreen	98FB98	152,251,152	BurlyWood	DEB887	222,184,135
Red	FF0000	255,0,0	LightGreen	90EE90	144,238,144	Tan	D2B48C	210,180,140
FireBrick	B22222	178,34,34	MediumSpringGreen	00FA9A	0,250,154	RosyBrown	BC8F8F	188,143,143
DarkRed	8B0000	139,0,0	SpringGreen	00FF7F	0,255,127	SandyBrown	F4A460	244,164,96
<b>Pinks</b>			MediumSeaGreen	3CB371	60,179,113	Goldenrod	DAA520	218,165,32
Pink	FFC0CB	255,192,203	SeaGreen	2E8B57	46,139,87	DarkGoldenrod	B8860B	184,134,11
LightPink	FFB6C1	255,182,193	ForestGreen	228B22	34,139,34	Peru	CD853F	205,133,63
HotPink	FF69B4	255,105,180	Green	008000	0,128,0	Chocolate	D2691E	210,105,30
DeepPink	FF1493	255,20,147	DarkGreen	006400	0,100,0	SaddleBrown	8B4513	139,69,19
MediumVioletRed	C71585	199,21,133	YellowGreen	9ACD32	154,205,50	Sienna	A0522D	160,82,45
PaleVioletRed	DB7093	219,112,147	OliveDrab	6B8E23	107,142,35	Brown	A52A2A	165,42,42

**CSS**: hex and RGB (Red/Green/Blue)



RED : #FF0000 or rgb(255,0,0)

GREEN : #00FF00 or rgb(0,255,0)

BLUE : #0000FF or rgb(0,0,255)



/Priority scheme\



# CSS: priority scheme

- bottom > top
- inline > internal > external (bottom > top)
- selectors
- !important (**almost never**)




/Exercise: given the following HTML doc\



# Assignment #1

- Add the following set of CSS rules as inline styling

```
✓ div {  
    width: 100px;  
    height: 100px;  
    border-radius: 50%;  
    background-color:  #5c3069;  
}
```

# Assignment #2

- Add the following set of CSS rules as internal styling

```
article {  
    width: 200px;  
    height: 200px;  
    border-radius: 50%;  
    background-color: #fff;  
    display: flex;  
    align-items: center;  
    justify-content: center;  
}
```

# Assignment #3

- Add the following set of CSS rules as external styling

```
section {  
    width: 300px;  
    height: 300px;  
    border-radius: 50%;  
    background-color: #5c3069;  
    display: flex;  
    align-items: center;  
    justify-content: center;  
}
```



/Q&A\





# Thank you

**Next: CSS Selectors - Part#1**

