[Lesson 5]

Roi Yehoshua 2018

[DAN.IT EDUCATION]

What we learnt last time?

- Browser default styles
- What is reset.css and why do we need it
- CSS block model
- box-sizing property
- Block display modes
- How to place several block elements on one line with “inline-block”
- How to create columns using and “inline-block”
- Elements tab in Chrome Developer Tools

[Our targets for today]

- CSS selectors
- Selectors weight
- CSS units of measurement

[DAN.IT EDUCATION]

CSS Selectors

- CSS selectors define the elements to which a set of CSS rules apply,
- Different CSS selectors can share same styles.

```
.bigButton, .smallButton {
  color: white;
}
```

- Same selectors may be reused multiple times.

```
.bigButton {
  font-size: 2.4em;
}
```

CSS Selectors

- Universal : * { margin:0; padding:0; }
- Type : p { font-size:2em; }
- Class : .info { background:#ff0; }
  - Multiple : .info.error { color:#900; }
  - Element with class : p.info { color: #111; }

- ID : #info { background:#ff0; }
- Group : td, th, div { font-size:1em; }
- Descendent : div p { color:#f00; }
- Child : div > strong { color:#f00; }
- Adj sibling : p + p { color:#f00; }
- Attribute : input[type="text"] { margin-left: 5px; }

[DAN.IT EDUCATION]

CSS Selectors Demo

```css
*
{
    color:Red;
}

h2
{
    color:Blue;
}

li h2
{
    color:Green;
}

#myListItem1
{
    color:Lime;
}

li.myListItem
{
    color:Navy;
}

.myListItem
{
    color:Orange;
}
```

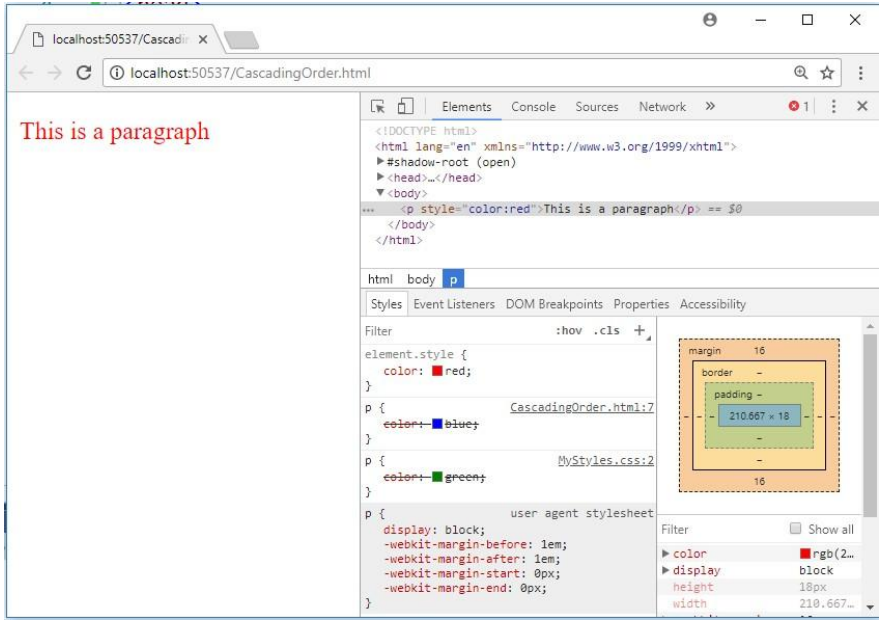
```html
<h2>Css Demo</h2>
this is a css demo
<ul>
    <li>
        <h2>header in list</h2>
        list1 item1
    </li>
</ul>
<ul>
    <li id="myListItem1">list2 item1</li>
    <li class="myListItem">list2 item2</li>
</ul>
<span class="myListItem">test</span>
```

Cascading Order

- What style will be used when there is more than one style specified for an HTML element?
- Inline style (inside a specific HTML element) has the highest priority
- Then external and internal style sheets (in the head section)
- And lastly the browser default
- You can examine which styles have been overridden by which rules using the browser developer tools

Cascading Order

```
<head>
	<link href="MyStyles.css" rel="stylesheet" />
	<style>
		p {
			color: blue;
		}
	</style>
</head>
<body>
	<p style="color:red">This is a paragraph</p>
</body>
```



[DAN.IT] EDUCATION

[Specificity]

- If there are two or more conflicting CSS rules that point to the same element, the browser checks which one is most specific and therefore wins out
  - How to calculate specificity?
  - Start at 0
  - add 1000 for style attribute
  - add 100 for each ID
  - add 10 for each attribute, class or pseudo-class
  - add 1 for each element name or pseudo-element

[DAN.IT] EDUCATION

Specificity

- Consider these three code fragments:

```
A: h1
B: #content h1
C: <h1 style="color: red">Heading</h1>
```

- The specificity of A is 1 (one element)
  The specificity of B is 101 (one ID reference and one element)
  The specificity of C is 1000 (inline styling)
- Since 1 < 101 < 1000, the third rule (C) has a greater level of specificity, and therefore will be applied

[DAN.IT] EDUCATION

CSS Units

- CSS has several different units for expressing a length
- Many CSS properties take "length" values, such as width, margin, padding, font-size,border-width, etc.
- Length is a number followed by a length unit, such as 10px, 2em, etc.
- A whitespace cannot appear between the number and the unit
- However, if the value is 0, the unit can be omitted
- For some CSS properties, negative lengths are allowed
- There are two types of length units: absolute and relative

CSS Units

- **px** - most simple unit. Represent one pixel of the screen.
- **%** - percent of the element first parent container that defines a size boundaries.
- **em** - relative to the font size of element container.
- **rem** - relative to the font size of the document root element.
- **vh**, **vw** - percent of the current screen height (vh) or screen width (vw).
- **vmin**, **vmax** - lowest from vh and vw for vmin and highest for vmax.

Absolute Lengths

- Absolute length units are fixed and a length expressed in absolute units will appear as exactly that size
- Absolute length units are not recommended for use on screen, because of the variety of screen sizes
- However, they can be used if the output medium is known, such as for print layout

Unit	Description
cm	centimeters
mm	millimeters
in	inches (1in = 96px = 2.54cm)
px	pixels (1px = 1/96th of 1in) Pixels (px) are relative to the viewing device. For low-dpi devices, 1px is one device pixel (dot) of the display. For printers and high resolution screens 1px implies multiple device pixels.
pt	points (1pt = 1/72 of 1in)
pc	picas (1pc = 12 pt)

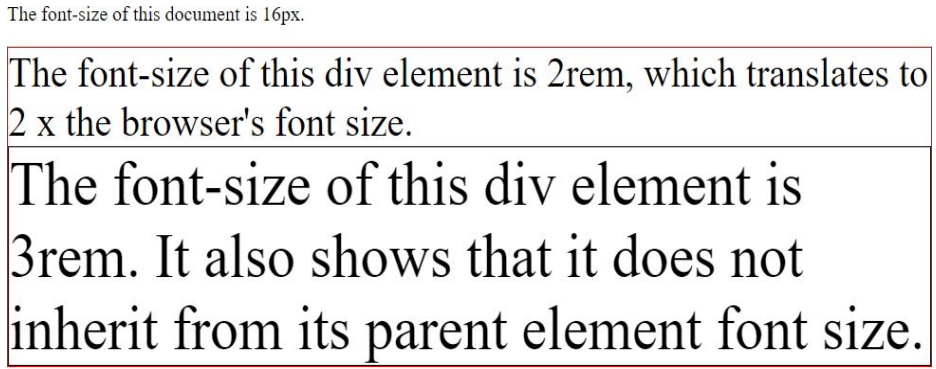
Relative Lengths

- Relative length units specify a length relative to another length property
- Relative length units scales better between different rendering devices

Unit	Description
em	Relative to the font-size of the element (2em means 2 times the size of the current font)
ex	Relative to the x-height of the current font (rarely used)
ch	Relative to width of the "0" (zero)
rem	Relative to font-size of the root element
vw	Relative to 1% of the width of the viewport Viewport = the browser window size. If the viewport is 50cm wide, 1vw = 0.5cm.
vh	Relative to 1% of the height of the viewport
vmin	Relative to 1% of viewport's smaller dimension
vmax	Relative to 1% of viewport's larger dimension
%	Relative to the parent element

Example for rem Units

```
<style>
    html {
        font-size: 16px;
    }
    div {
        font-size: 3rem;
        border: 1px solid black;
    }
    #top-div {
        font-size: 2rem;
        border: 1px solid red;
    }
</style>
```




```
<body>
    <p>The font-size of this document is 16px.</p>
    <div id="top-div">
        The font-size of this div element is 2rem, which translates to 2 x the browser's font size.
        <div>The font-size of this div element is 3rem. It also shows that it does not inherit from its  parent
element font size.</div>
    </div>
</body>
```

[DAN.IT] EDUCATION

[vw/vh Units]

- Sometimes there are elements within our design that we'd like to ensure can fit into the viewport in their entirety, even if the user resizes the browser
- With vw/vh, we can size elements relative to the size of the viewport
- 1 vw/vh unit reflects 1/100th the width/height of the viewport

```
<style>
    h1 {
        font-size: 20vw;
    }
</style>

<body>
    <h1>Hello</h1>

    <p>Resize the width of the browser window
to  see how the font-size of h1 changes.</p>
    <p>1vw = 1% of viewport width.</p>
</body>
```

[DAN.IT EDUCATION]

Control questions

1. What is CSS selector?
2. Which CSS selectors do you know?
3. Which attribute has higher weight: “class” or “id”?
4. What is the weight of the following selector? `h2#page-header a.logo-text.text-white`
5. How to overwrite inline styles placed in HTML using CSS?
6. Name some absolute and relative css units
7. When do we use absolute units and when relative?
8. What is the difference between em and rem css units?
9. What is the difference between **%** and **vh/vw** units?