Style Sheets

### Principle

- HTML5 follows the separation of concerns design principle: a system should be divided into parts with functionality that overlaps as little as possible.
- Concerns of presentation are dealt with using Cascading Style Sheets (CSS).

### History

- 1996, W3C: CSS Level 1 (CSS1)
  - Covers properties of fonts, texts, boxes; colours, pseudo classes
  - Very limited and buggy support in browsers
- 1998, W3C: CSS Level 2 (CSS2)
  - Covers all of CSS1 plus selectors, media types, box model, tables, bidirectional text, etc
  - Browser support still patchy
- 2011, W3C: CSS Level 2 Revision 1 (CSS2.1)
  - Fixes errors, removes features from CSS2 with insufficient browser supports, adds others
- 2011 onwards, W3C: CSS Level 3
  - Follows a modular specification approach (lots of smaller specifications instead of one very big one)
  - Covers all of CSS2.1 plus media queries, name spaces, flexible layout, etc

### Overview

- CSS Values: Lengths
  - 1996, W3C: CSS Level 1 (CSS1)
    - Covers properties of fonts, texts, boxes; colours, pseudo classes
    - Very limited and buggy support in browsers
  - 1998, W3C: CSS Level 2 (CSS2)
    - Covers all of CSS1 plus selectors, media types, box model, tables, bidirectional text, etc
    - Browser support still patchy
  - 2011, W3C: CSS Level 2 Revision 1 (CSS2.1)
    - Fixes errors, removes features from CSS2 with insufficient browser supports, adds others
  - 2011 onwards, W3C: CSS Level 3
    - Follows a modular specification approach (lots of smaller specifications instead of one very big one)
    - Covers all of CSS2.1 plus media queries, name spaces, flexible layout, etc

### Inline Styles

Using the `style` attribute, one can specify the presentation style for a single HTML element:

- **Styling text**
  - Deals with fonts, text colors, list item style, etc
- **Styling boxes**
  - Deals with properties of boxes:
    - For the purpose of style and layout, all HTML elements are considered to be boxes with the following properties:
      - Size of content, padding, border, margin, and box as well as their style (e.g. colour) can be changed
- **Arranging boxes**
  - Deals with the relative position of boxes to each other

Example:

To make an h1 element red with an italic font shape:

- Set the `font-style` property to italic and
- Set the `color` property to red

```html
<h1 style="font-style: italic; color: red">A Red H1 Heading</h1>
```

### Cascading Style Sheets

- Cascading Style Sheet cover three aspects of style and layout:
  - **Styling text**
    - Deals with fonts, text colors, list item style, etc
  - **Styling boxes**
    - Deals with properties of boxes:
      - For the purpose of style and layout, all HTML elements are considered to be boxes with the following properties:
  - **Arranging boxes**
    - Deals with the relative position of boxes to each other

CSS Values: Lengths

Relative Length Units
- `em`: current font-size
- `ex`: x-height of the current font
- `ch`: width of "0" (zero) in the current font
- `rem`: font-size of the root element (typically 16px)
- `vw`: 1% of the width of the viewport
- `vh`: 1% of the height of the viewport
- `vmin`: 1% of viewport’s smaller dimension
- `vmax`: 1% of viewport’s larger dimension

The viewport is the user’s visible area of a web page, typically determined by the size of the browser window.

Absolute Length Units
- `cm`: centimeter
- `mm`: millimeter
- `in`: inch (1in = 96px = 25.4mm)
- `pt`: point (1pt = 1/72 in)
- `pc`: pica (1pc = 12pt)

* A CSS pixel is the visual angle of one pixel on a device with a pixel density of 96dpi and a distance from the reader of 28in (0.2646mm).
* A web client chose the number of physical pixels that best approximates a CSS pixel
* The multipliers between two absolute length units are always as indicated above

CSS Values: Colours

CSS allows to specify colours in a number of different ways, including:
- **Via colour keywords:** CSS has about 140 names for a wide range of colours, for example: red, green, blue, gold, peachpuff, snow, rebeccapurple
- **Via hexadecimal notation:** #RRGGBB defines a colour as a mix of the colours red, green and blue R (red), G (green), B (blue) are hexadecimal characters (0-9, A-F) representing three numbers between 0 and 255, which in turn, specify the intensity of that colour
  - Example: #FF0000 defines the color red
  - **Via functional notation:** rgb(R,G,B) defines a colour as a mix of the colours red, green and blue R (red), G (green), B (blue) are either decimal numbers between 0 and 255 or percentages 0% to 100% that again specify the intensity of that colour

Mobile Devices and Lengths

- The interplay of CSS length units and mobile devices is problematic
- On mobile devices in portrait orientation, web clients typically render web pages in a viewport that is wider than the physical screen
  - On switching from landscape to portrait orientation, HTML elements appear to shrink when switching from landscape to portrait orientation
- Media queries might not work as expected

The viewport meta tag allows to control this behaviour or web clients

CSS Values: Font sizes

CSS allows to specify the size of a font in a number of different ways, including:
- **Via a length**
- **Via a percentage:** Scales the font relative to the size of the current font
- **Via one of the following keywords:**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Typical scaling factor</th>
</tr>
</thead>
<tbody>
<tr>
<td>xx-large</td>
<td>200%</td>
</tr>
<tr>
<td>x-large</td>
<td>150%</td>
</tr>
<tr>
<td>large</td>
<td>110%</td>
</tr>
<tr>
<td>median</td>
<td>100%</td>
</tr>
<tr>
<td>small</td>
<td>80%</td>
</tr>
<tr>
<td>x-small</td>
<td>60%</td>
</tr>
<tr>
<td>xx-small</td>
<td>55%</td>
</tr>
</tbody>
</table>

Inline Styles: Properties and Values

- **font-family:** Courier, monospace
- **font-style:** italic
- **font-size:** 12pt | larger
- **font-weight:** bold
- **color:** AliceBlue | #F0F8FF
- **background-color:** AliceBlue | #F0F8FF
- **text-decoration:** underline | none
- **text-align:** left | right | center | justify
- **text-indent:** 10pt
- **vertical-align:** top | middle | bottom

This is a right-justified paragraph in a sans serif font with some green text and here is a formatted link.

Example:
```html
<ul style="list-style-type:square; background:Red; padding:20px;"

  <li style="color:Red; padding-top:0.1in"
  
  <span style="color:Black;">second thing</span>
  </li>

  <li style="color:Red; padding-bottom:3em"
  
  <span style="color:Black;">second thing</span>
  </li>

A. first thing
B. second thing
```
**Inline Styles: Properties and Values**

```html
<caption style="color: red; font-style: italic; text-decoration: underline">Student data</caption>
<tr style="background-color: red"><th>Name</th><th>Age</th></tr>
<tr><td>Chris Smith</td><td>19</td></tr>
<tr><td>Pat Jones</td><td>20</td></tr>
<tr><td>Doogie Howser</td><td>15</td></tr>
</table>
```

**Revision and Further Reading**

Read

- Chapter 11: Introducing Cascading Style Sheets
- Chapter 12: Formatting Text
- Chapter 13: Colors and Backgrounds
- Chapter 14: Thinking Inside the Box

of

O'Reilly, 2018.
E-book https://library.liv.ac.uk/record=b5647021