

5. Cascading Style Sheets

1. Cascading Style Sheets (CSS)
 2. Embedded Style Sheets
 3. External Style Sheets
 4. Multiple Selectors
 5. Contextual Selectors
 6. Styles based on class and id attributes
 7. Combining selectors
 8. Further CSS selectors
 9. Some other CSS properties
 10. Links to more information
-

5.1. Cascading Style Sheets (CSS)

- *Cascading Style Sheets (CSS)* allow authors to control the presentation of their documents
- they apply typographic styles and layout instructions to elements on a page
- *cascading* refers to
 - the fact that styles cascade down the document tree, and
 - what happens when several different sources of style information are provided
- a *style* is simply a rule that tells the browser how to render a particular element
- a *rule* associates values with one or more properties of an element
- for example, most elements have a `color` property
- a *stylesheet* is a collection of rules

5.2. Embedded Style Sheets

- use a `<style>` element within the `<head>` element of a document

```
<head>
  <style type="text/css">
    h1 { background: yellow; color: purple; }
  </style>
  ...
</head>
```

- the `type` attribute defines the type of style being included (strictly the *MIME* type)
- cascading style sheets are of type `text/css`
- in this example,
 - `h1` is called the *selector*
 - the curly brace-delimited text is called the *declaration*
 - `background` and `color` are *properties*
 - `yellow` and `purple` are *values*
- all `h1` headings will have purple text on a yellow background:

An h1 heading

5.3. External Style Sheets

- specified by a `<link>` element within the `<head>` element of a document
- an external style sheet is a separate file that can be re-used to apply the same style to numerous documents
- e.g., the `notes.html` files for this module contain the following

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

- the `rel` attribute specifies the relationship between the document and the file referred to, i.e., `notes.css` in the parent directory
- note that the external style sheet can be located anywhere on the web
- the syntax used in external style sheets is the same as for embedded style sheets (without the `style` tags)

5.4. Multiple Selectors

- a selector can comprise a list of elements separated by commas
- all elements in the list have the same property style values applied to them:

```
h1, h2, h3, h4, h5, h6 {
    background: yellow;
    color: purple;
    text-align: center;
}
```

- all headings will have purple text on a yellow background and be centred horizontally:

An h1 heading

An h3 heading

5.5. Contextual Selectors

- element names separated by white space constitute a *contextual* selector:

```
ol li      { list-style: decimal; }
ol ol li   { list-style: lower-alpha; }
ol ol ol li { list-style: lower-roman; }
```

- with these styles
 - the first-level list items of an ordered list are numbered in decimal
 - the second-level list items are numbered with lower-case letters
 - the third-level list items are numbered with lower-case roman numerals
- if there is ambiguity between two contextual styles, the more specific context wins:

```
em  { color: red; }
ul em { color: blue; }
```

- in this example
 - emphasised text will be coloured red

- but emphasised text *within an unordered list* will be coloured blue
- note that in a selector such as `ul em`, the `em` element can occur as a *descendant* of the `ul` element

5.6. Styles based on class and id attributes

- each HTML element can have a `class` and `id` attribute
- often used with the `div` and `span` elements of (X)HTML for styling purposes
- the value of an `id` attribute is meant to be unique within a document
- for example, the following HTML

```
<p class="red">A red paragraph</p>
<p class="green">A green paragraph</p>
<p id="p123">A yellow paragraph</p>
```

may have the following style rules applied

```
.red { color: red; }
.green { color: green; }
#p123 { color: yellow; }
```

where `.` is the class selector and `#` is the id selector

- if we want the class style to apply only to paragraphs, we write

```
p.red { color: red; }
p.green { color: green; }
```

5.7. Combining selectors

- selectors can be combined with one another, e.g.:

```
pre span.keyword, code span.keyword { color: red; }
```

- this specifies that
 - the contents of any `span` element
 - with a `class` attribute value of `keyword`
 - occurring inside a `pre` or `code` element

should be red

- for example, the source for the above `pre` element is:

```
<pre>
pre span.keyword, code span.keyword { color: <span class="keyword">red</span>; }
</pre>
```

5.8. Further CSS selectors

Selector	Example	Selects ...
<code>.class</code>	<code>.keyword</code>	all elements with <code>class="keyword"</code>
<code>#id</code>	<code>#123</code>	the element with <code>id="123"</code>
<code>*</code>	<code>*</code>	all elements
<code>element</code>	<code>p</code>	all <code>p</code> elements
<code>element, element</code>	<code>div, p</code>	all <code>div</code> elements and all <code>p</code> elements
<code>element element</code>	<code>div p</code>	all <code>p</code> elements inside <code>div</code> elements
<code>element > element</code>	<code>div > p</code>	all <code>p</code> elements whose parent is a <code>div</code> element
<code>element ~ element</code>	<code>p ~ ul</code>	all <code>ul</code> elements having a <code>p</code> element as a preceding sibling
<code>[attribute]</code>	<code>[style]</code>	all elements with a <code>style</code> attribute
<code>[attribute = value]</code>	<code>[type="text/javascript"]</code>	all elements with <code>type="text/javascript"</code>
<code>[attribute ^= value]</code>	<code>a[href ^= "http"]</code>	all <code>a</code> elements whose <code>href</code> attribute value begins with <code>"http"</code>
<code>[attribute \$= value]</code>	<code>a[href \$= ".pdf"]</code>	all <code>a</code> elements whose <code>href</code> attribute value ends with <code>".pdf"</code>
<code>[attribute *= value]</code>	<code>a[href *= "bbk"]</code>	all <code>a</code> elements whose <code>href</code> attribute value contains <code>"bbk"</code>

5.9. Some other CSS properties

Property	Some values	Meaning
<code>font-size</code>	<code>large, 24px</code>	<code>px</code> is pixels, also <code>cm, ...</code>
<code>font-family</code>	<code>arial, serif, sans-serif</code>	specific like <code>arial</code> , or generic like <code>serif</code>
<code>font-style</code>	<code>normal, italic</code>	
<code>font-weight</code>	<code>normal, bold</code>	
<code>display</code>	<code>block, inline</code>	
<code>margin-top</code>	<code>1cm, 2em</code>	<code>em</code> is the size of the letter <code>m</code>
<code>line-height</code>	<code>normal, 2</code>	<code>2</code> is twice the current font size
<code>text-decoration</code>	<code>underline, line-through</code>	
<code>list-style-type</code>	<code>circle, lower-alpha</code>	<code>circle</code> for <code>ul</code> , <code>lower-alpha</code> for <code>ol</code>

5.10. Links to more information

- www.w3.org/Style/CSS/Overview.en.html
W3C's CSS home page
- en.wikipedia.org/wiki/Cascading_Style_Sheets
Wikipedia's entry on CSS
- www.w3schools.com/css/
w3schools CSS tutorial