Outline of CSS: Cascading Style Sheets

This is an introduction to CSS showing how styles are written, types of style sheets, CSS selectors, the cascade, grouping styles and how styles affect elements in a web page.

What is CSS

CSS, or *styles*, are a way of changing the *presentation* (the appearance) of elements in a web page. Often this means altering the way things look in terms if colour, size, spacing, alignment and so on.

You cannot have CSS without HTML. It is HTML that places elements in a web page. For instance, headings <h1>, <h2> etc., paragraphs <p>, links <a> and images <img>.

It is HTML that gives *structure* to a web page. HTML elements on their own look quite plain. The presentation of HTML elements can be changed with CSS.

What is a style

A style is a set of *properties* with a *name*. Properties are things such as colour, size, margin, padding, alignment and so on. The name of a style is referred to as a *selector.* The selector looks for an HTML element or attribute with its name in order to know what to affect in the page.

The style can be written in one line, as above, but often, for legibility, the various elements of the style are written on separate lines:

```
body {
    font-family: "Helvetica Neue",Helvetica,Arial,sans-serif;
    color: #333333;
    font-size: 16px;
    line-height: 20px;
}
```
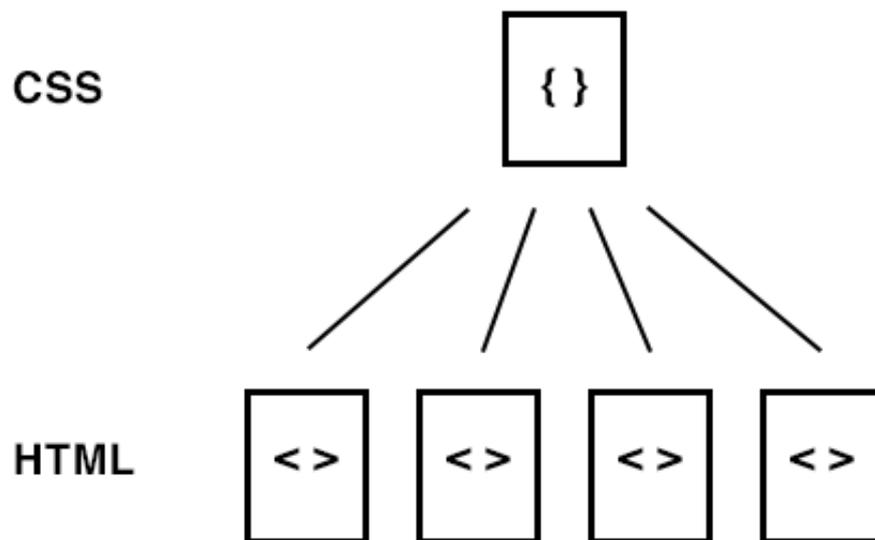
Style sheets
Styles can be written in three locations:

- In an *external* style sheet,
- In an *internal* style sheet
- As an *inline* style

External style sheet
This is the most efficient way to write styles for a web site. The CSS is created as a separate document and the web pages are linked to it. Styles in the external style sheet affect all pages in the site.

In the illustration above, each web page (HTML) has a link in the <head> to the CSS document. The example below shows how the link appears. In this case mystyles.css is the name of the style sheet.

```
<head>
<meta charset="utf-8">
<title>My web page</title>
<link href="mystyles.css" rel="stylesheet" type="text/css">
</head>
```

Internal style sheet

This is a style sheet within the <head> of an HTML document. This style sheet will only affect that particular web page. These are often used in conjunction with external CSS to create styles for one page only, or to utilise the *cascade* (referred to later). Internal CSS is often used when learning to use HTML and CSS:

```
<head>
<meta charset="utf-8">
<title>My web page</title>
<style type="text/css">
body {
    font-family: "Helvetica Neue",Helvetica,Arial,sans-serif;
    color: #333333;
    font-size: 16px;
    line-height: 20px;
}
</style>
</head>
```

Inline style

This is a style that is written inline in the HTML code within the <body> of the document. This is the least used form of writing styles, as it is inefficient compared to external or internal CSS. Every time a style is required, a list of properties for that style has to be written within the code as an *attribute* of the element being affected. In the example below, the style is added as an attribute of the paragraph tag.

```
<p style="font-size:14px; color:#666666">
```

Inline styles are more commonly used when creating HTML emails. This is because some email clients strip the <head> from an HTML document. Therefore, styles need to be written within the <body>.

Selectors

There are three main selector types and two sub category selector types. Every style falls into one of the three main selector types. The three main selector types are:

- **tag**
- **class**
- **id**

The two sub category selectors are:

- **compound**
- **pseudo**

Tag selector
A tag selector is a style that is named after an HTML tag. For example, if a style is named 'h1' then every instance of <h1> in the page will be affected by the style.

```
body {
      background-color: #FCF776;
}

h1 {
      font-size: 22px;
}
```

Class selector
A class selector has a name that is determined by the web designer. This name can be anything he or she wants. The style can be applied to any element or object in a web page. The style is not dependent on any particular HTML tag in the way the tag selector is. The style name in the style sheet is preceded with a full stop:

```
.image-left {
      float: left;
      margin-right: 30px;
}
```

This style is used to float an image to the left of text.

Id
Id means 'identifier'. As with the class selector, the web designer determines the name and to what element or object to apply it to. The selector name in the style sheet is preceded with the hash symbol:

```
#sidebar {
      width: 300px;
      float: left;
}
```

A style used to define a side column.

The difference between class and id selectors
Class and id selectors differ in one significant respect:

- A class selector can be used multiple times within a web page.
- An Id selector should only be used once in a web page.

In both cases, any particular class or ID selector can be used across all pages in a web site.

The difference is to do with *semantics*, the meaning of what is being used.

It may be that a designer wishes to have a style for certain pieces of text to make it appear different from other text. This style may need to be used several times in a page, therefore it would be appropriate to create a *class selector* for the text.

An *id selector* can be used for an element to indicate that it is unique within the page. A typical use for an id selector would be for a layout element such as a *wrapper* <div>, a box that contains all the content for the page. This region is unlikely to be repeated within a page so therefore an id selector can be created to affect its appearance. This indicates to another person editing the page that this is the only time this style is used in the page and that the element it is applied to is unique.

Compound selectors

Compound selectors are styles whose name is made from one or more of the three main selector types, *tag*, *class* or *id*. These styles are used to target a specific element or object within a particular region in the page. A compound selector will ensure that other elements that are the same in other regions in the page will not be affected by the style.

For instance, a style could be written to change the appearance of text links within a part of the page without affecting their appearance elsewhere.

```
#sidebar a {
    color: #FF1F87;
    text-decoration: none;
}
```

A style to change the appearance of text links in a side column.

Pseudo class selectors

These are styles that determine a condition or state for the element or object being targeted. The pseudo part of the selector is preceded by colon. For instance, a link can be made to change in appearance when the mouse is over it. The pseudo class selector for this is `a:hover`.

Other examples of pseudo classes are:

`p:first-line` - changes the appearance of the first line of a paragraph.

`p:first-letter` - changes the appearance of the first letter of a paragraph.

How selectors affect the appearance of elements in a web page.
For selectors to affect the content of a web page their selector name must appear with the HTML document, usually within the <body> of the document.

With *tag* selectors, the fact that the HTML tag they are named after appears in the document is enough.

With *class* and *id* selectors, the selector name must appear in the code as an *attribute* of an HTML element.

An example of a *class name* as an attribute of a <span> tag:

```
<span class="text-1">Some text I have written</span>
```

In the CSS, the selector for this class name would appear like so:

```
.text-1 {
    color: #0200FF;
}
```

An example of an *id name* as an attribute of a <div> tag:

```
<div id="sidebar">some example content</div>
```

In the CSS, the selector for this id name would appear like so:

```
#sidebar {
    width: 300px;
    float: left;
}
```

The Cascade
The *cascade* means an order of priority. This defines what style will be used when there is more than one style for a specific element on the page. The 'cascade' is listed below. Number four has the highest priority:

1. Browser default
2. External Style Sheet
3. Internal Style Sheet
4. Inline Style

How this works:

1. If no style is specified for a web page then the *browser default* is used.
2. If a style for the default font is specified in an *external* style sheet then all web pages linked to that style sheet will use that style. This will override the browser defaults.
3. If, in one of those web pages a different default font is required, then the style, using the same name as used in the external CSS, can be specified in an *internal* style sheet within the <head>. This style will override the style defined in the external style sheet.
4. Finally, an *inline* style for the default font will override an internal style sheet.

Grouped styles

There are times when different styles may use the same properties. Instead of writing these styles out individually, they can be grouped together.

```
h1, h2, h3 {
     color: #0200FF;
}
```

A grouped style defining the colour of headings.

Naming styles

- Styles can named with letters and numbers but do not start the name with a number.
- Upper and lower case can be used but the general convention is to use lower case.
- No spaces. Use a hyphen or underscore to represent a space.
- No special characters, except for the hash and full stop for id and class selectors. There are some particular CSS rules that use characters such as * and >.

Naming an external style sheet

The same general rules apply as when naming an HTML document. The style sheet can have any name you wish. The file name must be appended with '.css' e.g. `global.css`, `main.css`, etc..