CHAPTER 5.0 – CSS (CASCADED STYLE SHEET

INTRODUCTION

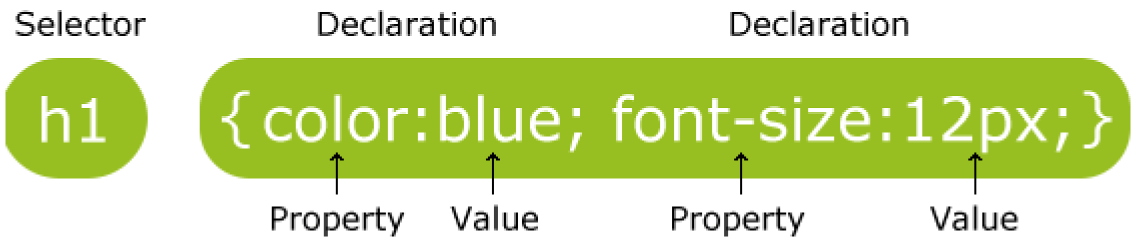
- **CSS** stands for **C**ascading **S**tyle **S**heets
- CSS defines **how HTML elements are to be displayed**
- Styles were added to HTML 4.0 **to solve a problem**
- CSS saves a lot of work
- External Style Sheets are stored in **CSS files**

CSS Solved a Big Problem

- HTML was intended to **define the content** of a document, like:
  - <h1>This is a heading</h1>
  - <p>This is a paragraph.</p>
- When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers.
- Development of large web sites, where fonts and color information were added to every single page, became a long and expensive process.
- To solve this problem, the World Wide Web Consortium created CSS.

CSS Syntax

- A CSS rule set consists of a selector and a declaration block:
- The selector points to the HTML element you want to style.
- The declaration block contains one or more declarations separated by semicolons.
- Each declaration includes a property name and a value, separated by a colon.



Example

- A CSS declaration always ends with a semicolon, and declaration groups are surrounded by curly braces:
  - p {color:red; text-align: center;}
- To make the CSS code more readable, you can put one declaration on each line.
- In the following example all <p> elements will be center-aligned, with a red text color:
- **Example**
- p {
      color: red;
      text-align: center;
  }

CSS Comments

- Comments are used to explain your code, and may help you when you edit the source code at a later date.
- Comments are ignored by browsers.
- A CSS comment starts with /* and ends with */.
- Comments can also span multiple lines:
- **Example**

```
p {
    color: red;
    /* This is a single-line comment */
    text-align: center;
  }

  /* This is
  a multi-line
  comment */
```

CSS Selectors

- CSS selectors allow you to select and manipulate HTML elements.
- CSS selectors are used to "find" (or select) HTML elements based on their id, class, type, attribute, and more.

The element Selector

- The element selector selects elements based on the element name.
- You can select all <p> elements on a page like this: (all <p> elements will be center-aligned, with a red text color)
- **Example**

-
  ```
  p {
      text-align: center;
      color: red;
  }
  ```

The id Selector

- The id selector uses the id attribute of an HTML element to select a specific element.
- An id should be unique within a page, so the id selector is used if you want to select a single, unique element.
- To select an element with a specific id, write a hash character, followed by the id of the element.
- The style rule below will be applied to the HTML element with id="para1":
- **Example**
-
```
#para1 {
    text-align: center;
    color: red;
}
```

The class Selector

- The class selector selects elements with a specific class attribute.
- To select elements with a specific class, write a period character, followed by the name of the class:
- **Example**
-
```
.center {
    text-align: center;
    color: red;
}
```
- You can also specify that only specific HTML elements should be affected by a class.
- **Example**
-
```
p.center {
    text-align: center;
    color: red;
}
```

Grouping Selectors

- If you have elements with the same style definitions, like this:

-
  ```
  h1 {
      text-align: center;
      color: red;
  }
  h2 {
      text-align: center;
      color: red;
  }
  p {
      text-align: center;
      color: red;
  }
  ```

Grouping Selectors

- You can group the selectors, to minimize the code.
- To group selectors, separate each selector with a comma.
- In the example below we have grouped the selectors from the code above:
- **Example**
- h1, h2, p {

```
h1, h2, p {
    text-align: center;
    color: red;
}
```

Three Ways to Insert CSS

- There are three ways of inserting a style sheet:
  - External style sheet
  - Internal style sheet
  - Inline style

External Style Sheet

- An external style sheet is ideal when the style is applied to many pages.
- With an external style sheet, you can change the look of an entire Web site by changing just one file.
- Each page must include a link to the style sheet with the <link> tag. The <link> tag goes inside the head section:

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

External Style Sheet

- An external style sheet can be written in any text editor.
- The file should not contain any html tags.
- The style sheet file must be saved with a .css extension.
- An example of a style sheet file called "myStyle.css", is shown below:

-
  ```
  body {
      background-color: lightblue;
  }
  h1 {
      color: navy;
      margin-left: 20px;
  }
  ```

Internal Style Sheet

- An internal style sheet should be used when a single document has a unique style.
- You define internal styles in the head section of an HTML page, inside the <style> tag, like this:
- **Example**
-
```
<head>
<style>
body {
   background-color: linen;
}
h1 {
   color: maroon;
   margin-left: 40px;
}
</style>
</head>
```

Inline Styles

- An inline style loses many of the advantages of a style sheet (by mixing content with presentation).

  (Use this method sparingly!)
- To use inline styles, add the style attribute to the relevant tag.
- The style attribute can contain any CSS property.
- The example shows how to change the color and the left margin of a h1 element:
- **Example**
- <h1 style="color:blue;margin-left:30px;">This is a heading.</h1>

Multiple Style Sheets

- If some properties have been set for the same selector in different style sheets, the values will be inherited from the more specific style sheet.
- For example, assume that an external style sheet has the following properties for the <h1> element:
- h1 {

```
h1 {
    color: navy;
    margin-left: 20px;
}
```

Multiple Style Sheets

- Then, assume that an internal style sheet also has the following property for the <h1> element:
- h1 {
      color: orange;
  }
- If the page with the internal style sheet also links to the external style sheet the properties for the <h1> element will be:
  - color: orange;
    margin-left: 20px;

CSS Background

- CSS background properties are used to define the background effects of an element.
- CSS properties used for background effects:
  - background-color
  - background-image
  - background-repeat
  - background-attachment
  - background-position

Background Color

- The background-color property specifies the background color of an element.
- The background color of a page is set like this:
- **Example**

  -
    ```
    body {
        background-color: #b0c4de;
    }
    ```
- With CSS, a color is most often specified by:
  - a HEX value - like "#ff0000"
  - an RGB value - like "rgb(255,0,0)"
  - a color name - like "red"

CSS Text

- **Text Color**
- The color property is used to set the color of the text.
- With CSS, a color is most often specified by:
  - a HEX value - like "#ff0000"
  - an RGB value - like "rgb(255,0,0)"
  - a color name - like "red"
- **Example**

-
  ```
  body {
      color: blue;
  }
  h1 {
      color: #00ff00;
  }

  h2 {
      color: rgb(255,0,0);
  }
  ```

Background Color

- **Text Alignment**
- The text-align property is used to set the horizontal alignment of a text.
- Text can be centered, or aligned to the left or right, or justified.
- When text-align is set to "justify", each line is stretched so that every line has equal width, and the left and right margins are straight (like in magazines and newspapers).
- **Example**
-
```
h1 {
    text-align: center;
}
p.date {
    text-align: right;
}
p.main {
    text-align: justify;
}
```

Background Color

- **Text Decoration**
- The text-decoration property is used to set or remove decorations from text.
- The text-decoration property is mostly used to remove underlines from links for design purposes:
- **Example**
- a { text-decoration: none; }
- It can also be used to decorate text. E.g.
-
```
h1 {
    text-decoration: overline;
}
h2 {
    text-decoration: line-through;
}
h3 {
    text-decoration: underline;
}
```

Text Transformation

- The text-transform property is used to specify uppercase and lowercase letters in a text.
- It can be used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word.
- **Example**
-
```
p.uppercase {
    text-transform: uppercase;
}
p.lowercase {
    text-transform: lowercase;
}
p.capitalize {
    text-transform: capitalize;
}
```

Text Indentation

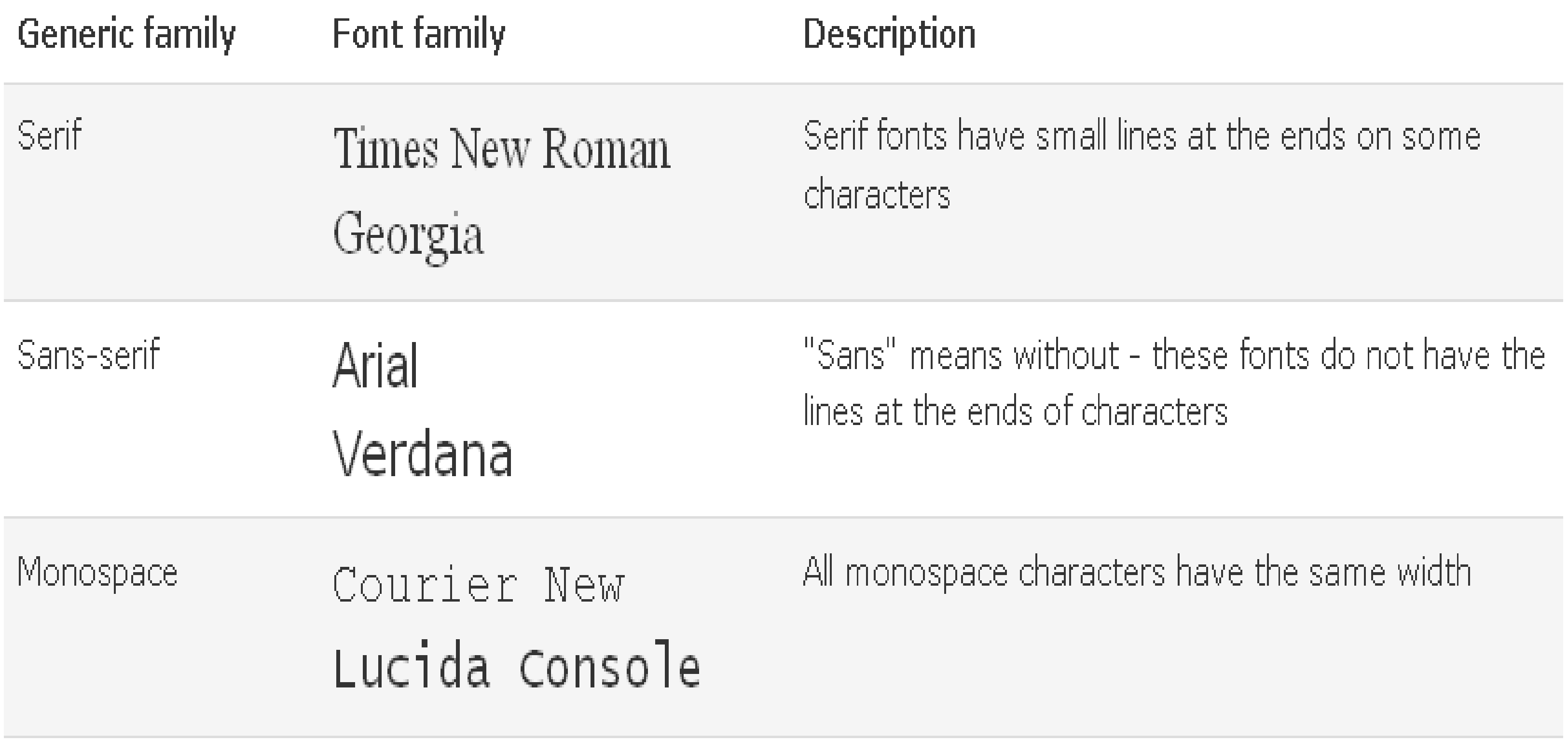
- The text-indent property is used to specify the indentation of the first line of a text.
- **Example**

-
  ```
  p {
      text-indent: 50px;
  }
  ```

CSS Font

- CSS font properties define the font family, boldness, size, and the style of a text.
- **CSS Font Families**
- In CSS, there are two types of font family names:
  - **generic family** - a group of font families with a similar look (like "Serif" or "Monospace")
  - **font family** - a specific font family (like "Times New Roman" or "Arial")

CSS Font



Generic family	Font family	Description
Serif	Times New Roman Georgia	Serif fonts have small lines at the ends on some characters
Sans-serif	Arial Verdana	"Sans" means without - these fonts do not have the lines at the ends of characters
Monospace	Courier New Lucida Console	All monospace characters have the same width

Font Family

- The font-family property should hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font.
- Start with the font you want, and end with a generic family, to let the browser pick a similar font in the generic family, if no other fonts are available.
- **Example**

-
  ```
  p {
      font-family: "Times New Roman", Times, serif;
  }
  ```

Font Style

- The font-style property is mostly used to specify italic text.
- This property has three values:
  - normal - The text is shown normally
  - italic - The text is shown in italics
  - oblique - The text is "leaning" (oblique is very similar to italic, but less supported)
- **Example**
-
```
p.normal {
   font-style: normal;
}
p.italic {
   font-style: italic;
}
p.oblique {
   font-style: oblique;
}
```

Font Size

- **Set Font Size With Pixels**
- Setting the text size with pixels gives you full control over the text size:
- **Example**

-
  ```
  h1 {
      font-size: 40px;
  }
  h2 {
      font-size: 30px;
  }
  p {
      font-size: 14px;
  }
  ```

Font Size

- **Set Font Size With Em**
- To allow users to resize the text (in the browser menu), many developers use em instead of pixels.
- 1em is equal to the current font size. The default text size in browsers is 16px. So, the default size of 1em is 16px.
- The size can be calculated from pixels to em using this formula: *pixels*/16=*em*
- **Example**
- h1 {

```
h1 {
    font-size: 2.5em; /* 40px/16=2.5em */
}
h2 {
    font-size: 1.875em; /* 30px/16=1.875em */
}
p {
    font-size: 0.875em; /* 14px/16=0.875em */
}
```

Styling Links

- Links can be styled with any CSS property (e.g. color, font-family, background, etc.).
- In addition, links can be styled differently depending on what **state** they are in.
- The four links states are:
  - a:link - a normal, unvisited link
  - a:visited - a link the user has visited
  - a:hover - a link when the user mouses over it
  - a:active - a link the moment it is clicked

Styling Links

- **Example**
- /* unvisited link */

```
/* unvisited link */
a:link {
    color: #FF0000;
}
/* visited link */
a:visited {
    color: #00FF00;
}
/* mouse over link */
a:hover {
    color: #FF00FF;
}
/* selected link */
a:active {
    color: #0000FF;
}
```

Styling Links

- When setting the style for several link states, there are some order rules:
  - a:hover MUST come after a:link and a:visited
  - a:active MUST come after a:hover

Common Link Styles

- The text-decoration property is mostly used to remove underlines from links:
- **Example**
-
```
a:link {
    text-decoration: none;
}
a:visited {
    text-decoration: none;
}
a:hover {
    text-decoration: underline;
}
a:active {
    text-decoration: underline;
}
```

Common Link Styles

- **Background Color**
- The background-color property specifies the background color for links:
- **Example**
- a:link {
     background-color: #B2FF99;
  }
  a:visited {
     background-color: #FFFF85;
  }
  a:hover {
     background-color: #FF704D;
  }
  a:active {
     background-color: #FF704D;
  }