Lecture No: 5
Topic: CSS3

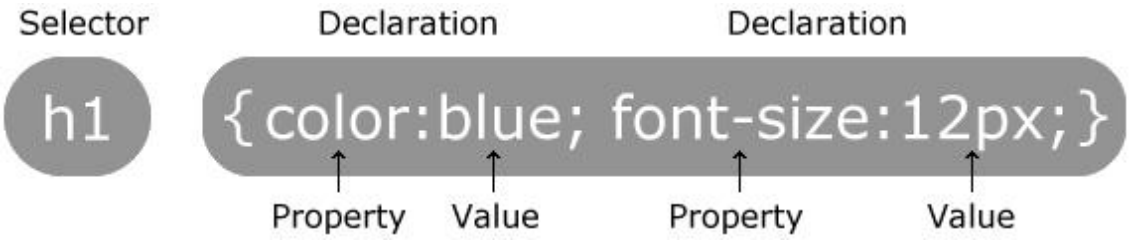
Introduction to CSS3

- The principle of Cascading Style Sheets (CSS) has roots in Standardized Generalized Markup Language (SGML) from the **1980s**. Its goals are to create a consistent look across many webpages and to separate structure from presentation so you can provide different style sheets for printing, browsing, or other scenarios.
- The World Wide Web Consortium (W3C) published CSS Level 1 recommendations in **December 1996**, and then started working on CSS Level 2. (The word recommendation means a formal release of the publication.) In **May 1998**, CSS Level 2 was published, and the W3C started working on various modules of CSS Level 3 (CSS3). Rather than creating one large publication for CSS3, the W3C separated CSS3 into modules that could be published independently.
- **In 2005**, the W3C became stricter with enforcement of the requirements for standards, and already-published standards such as CSS Level 2 Revision 1 (CSS 2.1), CSS3 Selectors, and CSS3 Text were pulled back from Candidate Recommendation to Working Draft level. It wasn't until **June 2011** that the CSS 2.1 recommendation was published. Because CSS3 is modular, the stability of each module differs, and its status differs. More than 50 modules are published from the W3C's CSS Working Group.
- CSS 2.1 is included in CSS3 because CSS3 is being designed to be backward compatible with the CSS 2.1 recommendation. This chapter introduces you to CSS3 by covering many aspects of CSS 2.1. Cascading style sheets provide a means to apply a presentation to an HTML structure by defining how HTML elements are displayed. By using CSS, you can set background and foreground colors, margins, borders, fonts, positions, and much more. You have creative control of the HTML elements, so you can decide what the elements look like and where they display on the screen.

CSS Solved a Big Problem

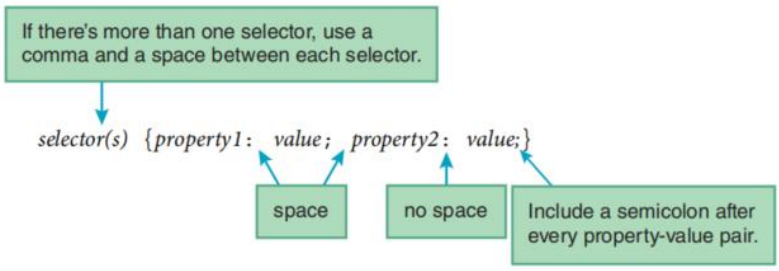
- HTML was **NEVER** intended to contain tags for formatting a web page.
- When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large websites, where fonts and color information were added to every single page, became a long and expensive process.
- To solve this problem, the World Wide Web Consortium (W3C) created CSS.
- CSS removed the style formatting from the HTML page.

CSS Syntax and Selectors

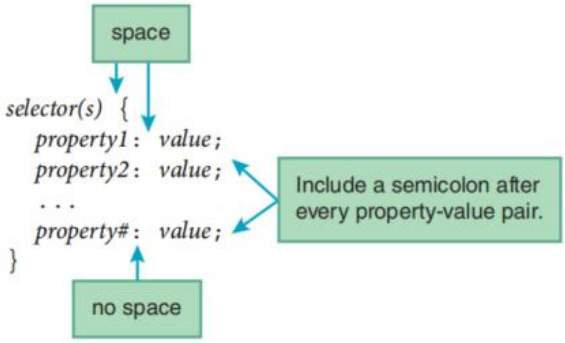


- A CSS rule-set consists of a selector and a declaration block.
- The **selector** points to the HTML element you want to style.
- The **declaration block** contains one or more declarations separated by semicolons.
- Each declaration includes a CSS **property** name and a **value**, separated by a **colon**.
- A CSS declaration always ends with a **semicolon**, and declaration blocks are surrounded by **curly braces**.

The element Selector



One line CSS



Block CSS rule

- The element selector selects elements based on the element name.
- You can select a
- <p> elements on a page like this (in this case, all <p> elements will be center-aligned, with a red text color):
- **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.
- The id of an element should be unique within a page, so the id selector is used to select one 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":
- **#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.
- In the example below, all HTML elements with class="center" will be red and center-aligned:
- **.center {**
  text-align: center;
  color: red;
  }

Grouping Selectors

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

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

CSS Comments

- Comments are used to explain the code, and may help 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:
- **p {**
  color: red;
  /* This is a single-line comment */
  text-align: center;
  }
  /* This is
  a multi-line
  comment */

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

- With an external style sheet, you can change the look of an entire website by changing just one file!
- `<link rel="stylesheet" href="name-of-external-file">`

  Note the rel="stylesheet" attribute-value pair. rel stands for “relationship,” and its value tells the browser engine what to do with the href file. Having a rel value of stylesheet tells the browser engine to look for CSS rules in the href file and apply them to the current web page.
- Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the <head> section:
- **<head>**
  <link rel="stylesheet" type="text/css" href="mystyle.css">
  </head>

mystyle.css

- **body {**
  background-color: lightblue;
  }

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

Internal Style Sheet

- An internal style sheet may be used if one single page has a unique style.
- Internal styles are defined within the <style> element, inside the <head> section of an HTML page:
- **<head>**
  <style>
  body {
  background-color: linen;
  }
  h1 {
  color: maroon;
  margin-left: 40px;
  }
  </style>
  </head>

Inline Styles

- An inline style may be used to apply a unique style for a single element.
- To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.
- The example below shows how to change the color and the left margin of a <h1> element:
- **<h1 style="color:blue;margin-left:30px;">This is a heading</h1>**

Cascading Order

- What style will be used when there is more than one style specified for an HTML element?

Places Where CSS Rules Can Be Defined, Highest to Lowest Priority
1. In an element's style attribute.
2. In a style element in the web page's head section.
3. In an external file.
4. In the settings defined by a user for a particular browser installation.
5. In the browser's native default settings.

CSS COLORS

- Background Color
  - **<h1 style="background-color:DodgerBlue;">Hello World</h1>**
- Text Color
  - **<h1 style="color:Tomato;">Hello World</h1>**
- Border Color
  - **<h1 style="border:2px solid DodgerBlue;">Hello World</h1>**

Color Values

- In HTML, colors can also be specified using
  - RGB values **rgb(*red, green, blue*)**,
  - HEX values **#*rrggbb***,
  - HSL values **hsl(*hue, saturation, lightness*)**,
  - RGBA values **rgba(*red, green, blue, alpha*)**, and

- HSLA values **hsla(*hue, saturation, lightness, alpha*)**

RGB Values for Color

- RGB stands for red, green, and blue. An RGB value specifies the amounts of red, green, and blue that mix together to form the displayed color. To specify an amount of a color, you can use a percentage, an integer, or a hexadecimal number:
  percentage—0% to 100% for each color
  integer—0 to 255 for each color
  hexadecimal—00 to ff for each color
- The rgba() specifier has been added to allow the alpha, or opacity amount, to be set with a color. Just like rgb(), the first three parameters set red, green, and blue amounts, and are values ranging from 0 to 255. A fourth parameter, the alpha, is a floating point value from 0 to 1 where 0 is completely transparent and 1 is completely opaque. The following declares a red background color that is 50 percent transparent:
  background-color: rgba(255, 0, 0, 0.5);

HSL and HSLA

- CSS3 also supports **HSL** colors, which stands for **Hue**, **Saturation**, and **Lightness**. HSL is based on a color wheel that is full color at the edges and fades to gray in the center. Now extend the wheel into a cylinder that is black at the bottom, white at the top, and full color in the middle.
- It is specified using hsl(h, s, l). **Hue** is a value from 0 to 360 that maps to the degrees on the color wheel. 0 is red, 120 is green, 240 is blue, and 360 is back around to red. **Saturation** is the percentage of color where 0% is completely gray and 100% full color. **Lightness** is the percent of lightness where 0% is black, 50% is full color, and 100% is white.
-

CSS Backgrounds

- The CSS background properties are used to define the background effects for elements.
- CSS background properties:
  - background-color
  - background-image
  - background-repeat
  - background-attachment
  - background-position

Example:

```
body {
  background-image: url("img_tree.png");
  background-repeat: no-repeat;
  background-position: right top;
  background-attachment: fixed;
}
```

Background - Shorthand property

- To shorten the code, it is also possible to specify all the background properties in one single property. This is called a **shorthand property**.
- The shorthand property for background is background:

- **body {**
  background: #ffffff url("img_tree.png") no-repeat right top;
  }

CSS Borders

- The CSS border properties allow you to specify the style, width, and color of an element's border.
- The **border-style** property specifies what kind of border to display.

The following values are allowed:

- **dotted** - Defines a dotted border
- **dashed** - Defines a dashed border
- **solid -** Defines a solid border
- **double** - Defines a double border
- **groove** - Defines a 3D grooved border. The effect depends on the border-color value
- **ridge** - Defines a 3D ridged border. The effect depends on the border-color value
- **inset** - Defines a 3D inset border. The effect depends on the border-color value
- **outset -** Defines a 3D outset border. The effect depends on the border-color value
- **none -** Defines no border
- **hidden -** Defines a hidden border

Example:

- p.dotted {border-style: dotted;}
  p.dashed {border-style: dashed;}
  p.solid {border-style: solid;}
  p.double {border-style: double;}
  p.groove {border-style: groove;}

Border Width

- The **border-width** property specifies the width of the four borders.
- The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick.
- The border-width property can have from one to four values (for the top border, right border, bottom border, and the left border).

Border Color

- The **border-color** property is used to set the color of the four borders.
- The color can be set by:
  - name - specify a color name, like "red"
  - Hex - specify a hex value, like "#ff0000"
  - RGB - specify a RGB value, like "rgb(255,0,0)"
  - transparent
- The border-color property can have from one to four values (for the top border, right border, bottom border, and the left border).

Border - Individual Sides

- In CSS, there are also properties for specifying each of the borders (top, right, bottom, and left):
- **p {**
  border-top-style: dotted;
  border-right-style: solid;
  border-bottom-style: dotted;

```
    border-left-style: solid;
}
```

Border - Shorthand Property

- The border property is a shorthand property for the following individual border properties:
  - border-width
  - border-style (required)
  - border-color
- **p {**
  border: 5px solid red;
  }

Left Border

- **p {**
  border-left: 6px solid red;
  background-color: lightgrey;
  }

Bottom Border

- **p {**
  border-bottom: 6px solid red;
  background-color: lightgrey;
  }

Rounded Borders

The **border-radius** property is used to add rounded borders to an element:

```
p {
  border: 2px solid red;
  border-radius: 5px;
}
```

CSS Height and Width

- The height and width properties are used to set the height and width of an element.
- The height and width can be set to auto (this is default. Means that the browser calculates the height and width), or be specified in length values, like px, cm, etc., or in percent (%) of the containing block.

Setting max-width

- The **max-width** property is used to set the maximum width of an element.
- The max-width can be specified in length values, like px, cm, etc., or in percent (%) of the containing block, or set to none (this is default. Means that there is no maximum width).

References:

- Carey. P (2018). *New perspectives on HTML5, CSS3, and Javascript.* Australia : Cengage Learning
- Jenkins, S. (2013). Web Design All-in-One for Dummies. John Wiley & Sons.
- Parker, J. (2021). HTML for Beginners: A Complete Beginners Guide to Learn Html in 1 Hour and Master Your Web Designing.
- W3Schools online web tutorials. (n.d.). https://www.w3schools.com/
- Sklar, J. (2012). *Web Design Principles*.
- Castro, E., & Hyslop, B. (2013). *HTML and CSS: Visual QuickStart Guide*. Peachpit Press.