

WP – MODULE 2

PART 1

SWARADH P
Assistant Professor
KMCT CE

Introduction to Cascading Style Sheets (CSS)

Introduction

- HTML5 is used for marking up information to be rendered in a browser.
- We use a W3C technology called **Cascading Style Sheets 3 (CSS3)** to formatting and presenting information. In other words, CSS is used to format the layout of a webpage.
- CSS3 allows you to specify the presentation of elements on a web page (e.g., fonts, spacing, sizes, colors, positioning) separately from the document's structure and content (section headers, body text, links, etc.)

Introduction to Cascading Style Sheets (CSS)

- HTML5 was designed to specify the content and structure of a document.
- This separation of structure from presentation simplifies maintaining and modifying web pages, especially on large-scale websites.
- Though HTML5 has some attributes that control presentation, it's better not to mix presentation with content.
- If a website's presentation is determined entirely by a style sheet, you can simply swap in a new stylesheet to completely change the site's appearance.

Introduction to Cascading Style Sheets (CSS)

- Hence, With CSS, you can **control the color, font, the size of text, the spacing between elements, how elements are positioned and laid out, what background images or background colors** are to be used, different displays for different devices and screen sizes, and much more.
- The word **cascading** means that a style applied to a parent element will also apply to all children elements within the parent.
- So, if you set the color of the body text to "blue", all headings, paragraphs, and other text elements within the body will also get the same color (unless you specify something else)

Introduction to Cascading Style Sheets (CSS)

Using CSS

CSS can be added to HTML documents in **3 ways**:

- **Inline** - By using the style attribute inside HTML elements
- **Internal** - By using a `<style>` element in the `<head>` section
- **External** - By using a `<link>` element to link to an external CSS file

The most common way to add CSS, is to keep the styles in external CSS files.

Introduction to Cascading Style Sheets (CSS)

Inline Styles

- An inline CSS is used to apply a unique style to a single HTML element.
- An inline CSS uses the style attribute of an HTML element.
- Inline styles do not truly separate presentation from content.
- To apply similar styles to multiple elements, use embedded style sheets or external style sheets.
- The following example sets the text color of the <h1> element to blue, and the text color of the <p> element to red:

```
<h1 style="color:blue;">A Blue Heading</h1>
<p style="color:red;">A red paragraph.</p>
```

Introduction to Cascading Style Sheets (CSS)

Inline Styles

```
<!DOCTYPE html>
<html>
<body>

<h1 style="color:blue;">A Blue Heading</h1>

<p style="color:red;">A red paragraph.</p>

</body>
</html>
```

A Blue Heading

A red paragraph.

Introduction to Cascading Style Sheets (CSS)

Inline Styles

```
<!DOCTYPE html>

<!-- Using inline styles -->
<html>
  <head>
    <meta charset = "utf-8">
    <title>Inline Styles</title>
  </head>
  <body>
    <p>This text does not have any style applied to it.</p>

    <!-- The style attribute allows you to declare -->
    <!-- inline styles. Separate multiple -->
    <!-- style properties with a semicolon. -->
    <p style = "font-size: 20pt;">This text has the
    <em>font-size</em> style applied to it, making it 20pt.
    </p>
    <p style = "font-size: 20pt; color: deepskyblue;">
      This text has the <em>font-size</em> and
      <em>color</em> styles applied to it, making it
      20pt and deep sky blue.</p>
  </body>
</html>
```

Introduction to Cascading Style Sheets (CSS)

Inline Styles

This text does not have any style applied to it.

This text has the *font-size* style applied to it, making it 20pt.

This text has the *font-size* and *color* styles applied to it, making it 20pt and deep sky blue.

Introduction to Cascading Style Sheets (CSS)

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.
-

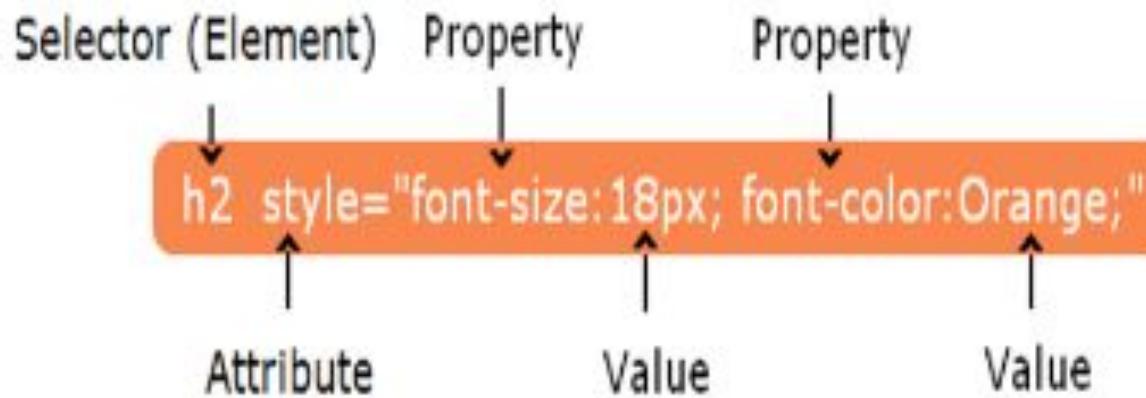
Inline CSS style consists set of rules :

- Selector (Element)
- Style (Attribute)
- Property, and
- Value

Introduction to Cascading Style Sheets (CSS)

Inline Styles

How to write Inline CSS Style



Introduction to Cascading Style Sheets (CSS)

Inline Styles

- Attribute style specifies an element's style.
- Each CSS property (Eg. font-size) is followed by a colon and a value.
- We can declare a particular p element to use a 20-point font size.
- Two properties (Eg. font-size and color) need to be separated by a semicolon.
- For Eg., we can set a given paragraph's color to deepskyblue.
- Hexadecimal codes may be used in place of color names.
- The below Figure depicts the HTML standard color set.

Introduction to Cascading Style Sheets (CSS)

Inline Styles

Color name	Value	Color name	Value
aqua	#00FFFF	navy	#000080
black	#000000	olive	#808000
blue	#0000FF	purple	#800080
fuchsia	#FF00FF	red	#FF0000
gray	#808080	silver	#C0C0C0
green	#008000	teal	#008080
lime	#00FF00	yellow	#FFFF00
maroon	#800000	white	#FFFFFF

Fig. HTML standard colors and hexadecimal RGB values.

Introduction to Cascading Style Sheets (CSS)

Embedded / Internal Style Sheets

- A second technique for using style sheets is embedded style sheets, which enable you to embed a CSS3 document in an HTML5 document's head section.
- An internal CSS is used to define a style for a single HTML page.
- An internal CSS is defined in the <head> section of an HTML page, within a <style> element.

Introduction to Cascading Style Sheets (CSS)

Embedded / Internal Style Sheets

```
<!DOCTYPE html>
<html>
<head>
<style>
body {background-color: powderblue;}
h1 {color: blue;}
p {color: red;}
</style>
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

This is a heading

This is a paragraph.

Introduction to Cascading Style Sheets (CSS)

Embedded / Internal Style Sheets

```
<!DOCTYPE html>

<!-- Fig. 4.3: embedded.html --&gt;
<!-- Embedded style sheet. --&gt;
&lt;html&gt;
  &lt;head&gt;
    &lt;meta charset = "utf-8"&gt;
    &lt;title&gt;Embedded Style Sheet&lt;/title&gt;

    &lt;!-- this begins the style sheet section --&gt;
    &lt;style type = "text/css"&gt;
      em      { font-weight: bold;
                 color: black; }
      h1      { font-family: tahoma, helvetica, sans-serif; }
      p       { font-size: 12pt;
                 font-family: arial, sans-serif; }
      .special { color: purple; }
    &lt;/style&gt;
  &lt;/head&gt;
  &lt;body&gt;
    &lt;!-- this attribute applies the .special style class --&gt;
    &lt;h1 class = "special"&gt;Deitel &amp; Associates, Inc.&lt;/h1&gt;

    &lt;p&gt;Deitel &amp; Associates, Inc. is an authoring and
       corporate training organization specializing in
       programming languages, Internet and web technology,
       iPhone and Android app development, and object
       technology education.&lt;/p&gt;</pre>
```

Introduction to Cascading Style Sheets (CSS)

Embedded / Internal Style Sheets

```
<h1>Clients</h1>
<p class = "special"> The company's clients include many
    <em>Fortune 1000 companies</em>, government agencies,
    branches of the military and business organizations.</p>
</body>
</html>
```



Introduction to Cascading Style Sheets (CSS)

class Attribute in HTML

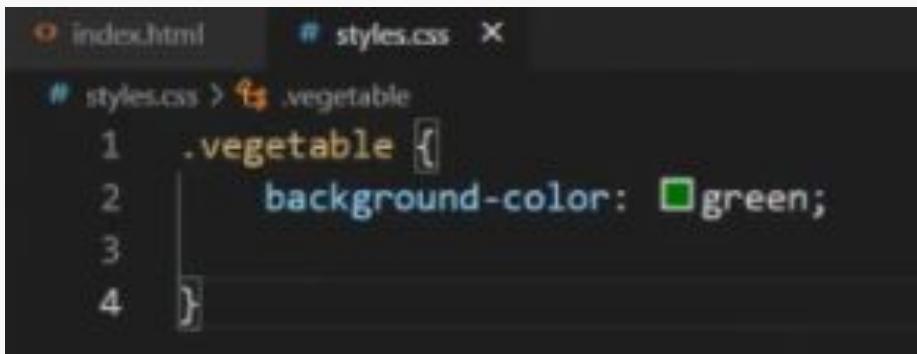
```
<!DOCTYPE html>
<html>
<head>
<title>Learn Html</title>
    <link rel="stylesheet" href="styles.css">
</head>

<body>
    <h1>HTML Classes</h1>

    <h2 class="vegetable">Tomato</h2>
    <h2 class="fruit">Orange</h2>
    <h2 class="vegetable">Onion</h2>
    <h2 class="fruit">Apple</h2>
    <h2 class="fruit">Banana</h2>
    <h2 class="vegetable">Cabbage</h2>
    <h2 class="vegetable">Carrot</h2>
</body>
</html>
```

Introduction to Cascading Style Sheets (CSS)

class Attribute in HTML



The image shows a file explorer interface with two files: 'index.html' and 'styles.css'. The 'index.html' file is a blank document. The 'styles.css' file contains the following CSS code:

```
1 .vegetable {  
2     background-color: green;  
3 }  
4 }
```

HTML Classes

Tomato

Orange

Onion

Apple

Banana

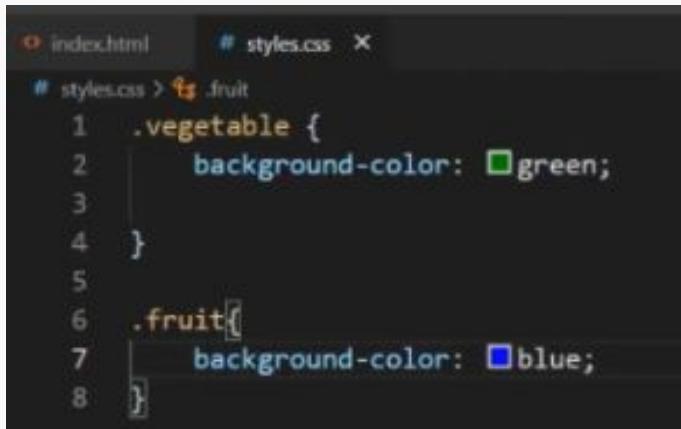
Cabbage

Carrot



Introduction to Cascading Style Sheets (CSS)

class Attribute in HTML



The image shows a code editor interface with two tabs: "index.html" and "# styles.css". The "# styles.css" tab is active, displaying the following CSS code:

```
# styles.css > .fruit
1 .vegetable {
2     background-color: green;
3 }
4 }
5
6 .fruit{
7     background-color: blue;
8 }
```

HTML Classes

Tomato

Orange

Onion

Apple

Banana

Cabbage

Carrot

Introduction to Cascading Style Sheets (CSS)

Embedded / Internal Style Sheets

The **style** Element and MIME Types

- The **style** element defines the embedded style sheet.
- Styles placed in the head apply to matching elements wherever they appear in the body.
- The **style** element's type attribute specifies the MIME (Multipurpose Internet Mail Extensions) type that describes the **style** element's content.
- CSS documents use the MIME type **text/css**.
- The default type for a **style** element in HTML5 is "**text/css**"

Introduction to Cascading Style Sheets (CSS)

MIME type	Description
text/css	CSS documents
image/png	PNG images
text/javascript	JavaScript markup
text/plain	Plain text
image/jpeg	JPEG image
text/html	HTML markup

A few common MIME types.

Introduction to Cascading Style Sheets (CSS)

External Style Sheets

- With an external style sheet, you can change the look of an entire website by changing just one file!
- Each HTML page must include a reference to the external style sheet file inside the `<link>` element, inside the `<head>` section.
- External styles are defined within the `<link>` element, inside the `<head>` section of an HTML page.
- An external style sheet can be written in any text editor, and must be saved with a `.css` extension.
- The external `.css` file should not contain any HTML tags.

Introduction to Cascading Style Sheets (CSS)

External Style Sheets

html file

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" href="mystyle.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

css file (mystyle.css)

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

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

Introduction to Cascading Style Sheets (CSS)

External Style Sheets

Output

This is a heading

This is a paragraph.

Introduction to Cascading Style Sheets (CSS)

External Style Sheets

- In the context of CSS, "rel" refers to the rel attribute of the HTML <link> element.
- This attribute is crucial for specifying the relationship between the current HTML document and the external resource being linked, which in the case of CSS is typically a stylesheet.
- When linking an external CSS file to an HTML document, the rel attribute is set to "stylesheet" within the <link> tag.
- This explicitly informs the browser that the linked file is a CSS stylesheet and should be applied to style the current document.

CSS - Simple Selectors

CSS Selectors

- CSS selectors are used to find or select the HTML elements you want to style.

Simple selectors (select elements based on name, id, class)

The CSS element Selector

Example

Here, all `<p>` elements on the page will be center-aligned, with a red text color:

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

CSS - Simple Selectors

The CSS element Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<p>Every paragraph will be affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>
```

CSS - Simple Selectors

The CSS element Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
p {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<p>Every paragraph will be affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>
```

Every paragraph will be affected by the style.

Me too!

And me!

CSS - Simple Selectors

The CSS id Selector

- The id selector uses the id attribute of an HTML element to select a specific element.
- The id of an element is 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.

Example

The CSS rule below will be applied to the HTML element with id="para1":

```
#para1 {  
    text-align: center;  
    color: red;  
}
```

CSS - Simple Selectors

The CSS id Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
#para1 {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<p id="para1">Hello World!</p>
<p>This paragraph is not affected by the style.</p>

</body>
</html>
```

CSS - Simple Selectors

The CSS id Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
#para1 {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<p id="para1">Hello World!</p>
<p>This paragraph is not affected by the style.</p>

</body>
</html>
```

Hello World!

This paragraph is not affected by the style.

CSS - Simple Selectors

The CSS class Selector

- The class selector selects HTML elements with a specific class attribute.
- To select elements with a specific class, write a period (.) character, followed by the class name.

Example

In this example all HTML elements with class="center" will be red and center-aligned:

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

CSS - Simple Selectors

The CSS class Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1 class="center">Red and center-aligned heading</h1>
<p class="center">Red and center-aligned paragraph.</p>

</body>
</html>
```

CSS - Simple Selectors

The CSS class Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1 class="center">Red and center-aligned heading</h1>
<p class="center">Red and center-aligned paragraph.</p>

</body>
</html>
```

Red and center-aligned heading

Red and center-aligned paragraph.

CSS - Simple Selectors

The CSS class Selector

Example

In this example only <p> elements with class="center" will be red and center-aligned:

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

CSS - Simple Selectors

The CSS class Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and center-aligned.</p>

</body>
</html>
```

CSS - Simple Selectors

The CSS class Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
p.center {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<h1 class="center">This heading will not be affected</h1>
<p class="center">This paragraph will be red and center-aligned.</p>

</body>
</html>
```

This heading will not be affected

This paragraph will be red and center-aligned.

CSS - Simple Selectors

The CSS Universal Selector

- The universal selector (*) selects all HTML elements on the page.

Example

The CSS rule below will affect every HTML element on the page:

```
* {  
    text-align: center;  
    color: blue;  
}
```

CSS - Simple Selectors

The CSS Universal Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
* {
    text-align: center;
    color: blue;
}
</style>
</head>
<body>

<h1>Hello world!</h1>

<p>Every element on the page will be affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>
```

CSS - Simple Selectors

The CSS Universal Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
* {
  text-align: center;
  color: blue;
}
</style>
</head>
<body>

<h1>Hello world!</h1>

<p>Every element on the page will be affected by the style.</p>
<p id="para1">Me too!</p>
<p>And me!</p>

</body>
</html>
```

Hello world!

Every element on the page will be affected by the style.

Me too!

And me!

CSS - Simple Selectors

The CSS Grouping Selector

- The grouping selector selects all the HTML elements with the same style definitions.
- Look at the following CSS code (the h1, h2, and p elements have the same style definitions):

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

CSS - Simple Selectors

The CSS Grouping Selector

- The grouping selector selects all the HTML elements with the same style definitions.
- Look at the following CSS code (the h1, h2, and p elements have the same style definitions):

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

CSS - Simple Selectors

The CSS Grouping Selector

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

CSS - Simple Selectors

The CSS Grouping Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
h1, h2, p {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<h1>Hello World!</h1>
<h2>Smaller heading!</h2>
<p>This is a paragraph.</p>

</body>
</html>
```

CSS - Simple Selectors

The CSS Grouping Selector

```
<!DOCTYPE html>
<html>
<head>
<style>
h1, h2, p {
    text-align: center;
    color: red;
}
</style>
</head>
<body>

<h1>Hello World!</h1>
<h2>Smaller heading!</h2>
<p>This is a paragraph.</p>

</body>
</html>
```

Hello World!

Smaller heading!

This is a paragraph.

THANK YOU