



S. S Jain Subodh P.G. (Autonomous) College

SUBJECT - Internet and Web Technology

TITLE - Cascading Style Sheets

Presented By –

SANGEETA VAIBHAV MEENA

What is CSS?

- CSS stands for Cascading Style Sheet.
- Cascading Style Sheets (CSS) is a style sheet language used to describe the presentation semantics (that is, the look and formatting) of a document written in a markup language.
- A Cascading Style Sheet is a file with a list of formatting instructions.
- Adding Styles (i.e. fonts, colors, spacing etc.) to web documents.
- CSS style sheets are the modern way to control the appearance and layout of our web pages.
- CSS saves a lot of work. It can control the layout of multiple web pages all at once by saving Styles in external .css files.



CSS Syntax

- A CSS comprises of style rules that are interpreted by the browser and then applied to the corresponding elements in our document. A style rule is made of three parts:-

Selector: A selector is an HTML tag at which a style will be applied. This could be any tag like `<h1>` or `<table>` etc.

Property: A property is a type of attribute of HTML tag. Put simply, all the HTML attributes are converted into CSS properties. They could be color, border, etc.

Value: Values are assigned to properties. For example, color property can have the value either red or #F1F1F1 etc.

Syntax:- selector { property: value; }

Examples:- H1 {color:Yellow;font-size:11px;}
table{ border :1px solid #C00; }



CSS Selectors

- CSS selectors are used to “find” or “select” HTML elements based on their element name, id, class, attribute, and more.
- Different types of Selectors are:-
 - ❖ Element Selector
 - ❖ Universal Selector
 - ❖ Descendant Selector
 - ❖ ID Selector
 - ❖ Class Selector
 - ❖ Child Selector
 - ❖ Attribute Selector



The Element Selector

- The element selector selects the HTML elements based on the element name.
- Example:-

```
<!DOCTYPE html>
<html>
<head>
<style> p { color: red;
           text-align: center; }
</style>
</head>
<body>
  <p>Hello World!</p>
  <p>These paragraphs are styled with CSS.</p>
</body>
</html>
```



The Universal Selectors

- Rather than selecting elements of a specific type, the universal selector quite simply matches the name of any element type.

- Example:-

```
<head>
<style>
  * {
    color: #000000;
  }
</style>
</head>
```

- This rule renders the content of every element in our document in black.



The Descendant Selectors

- Suppose we want to apply a style rule to a particular element only when it lies inside a particular element.

- Example:-

```
<head>  
<style>  
  ul em { color: #000000;  
  }  
</style>  
</head>
```

- This style rule will apply to `` element only when it lies inside the `` tag.



The ID Selectors

- The id selector selects the id attribute of an HTML element to select a specific element.
- An id is always unique within the page so it is chosen to select a single, unique element.
- It is written with the hash character (#), followed by the id of the element.
- The style rule below will be applied to the HTML element with id="para1".
- Example:-

```
<html>
<head> <style> #para1 { text-align: center; color: blue; }
        </style> </head>
<body>
    <p id="para1">Hello Javatpoint.com</p>
    <p>This paragraph will not be affected.</p>
</body>
</html>
```




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:

➤ Example:-

```
<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>
```




Difference between CLASS & ID Attribute

- Many Elements can have the same “Class” but they cannot have the same “ID”.
- An “ID” is a unique identifier, used for targeting certain elements or tagging parent elements.
- “Class” is used when we want to consistently style multiple elements throughout the page/site. Classes are useful when we have more than one element that shares the same style.
- A good way to remember this is a class is a type of item and the id is the unique name of an item on the page.
- **ID's are unique**
 - ❖ Each element can have only one ID
 - ❖ Each page can have only one element with that ID
- **Classes are not unique**
 - ❖ We can use the same class on multiple elements.
 - ❖ We can use multiple classes on the same element.



The Child Selectors

- Similar to descendants but have different functionality.

- Example:-

```
<html>
<head> <style>
  body > p{ color : #ff0000; }
</style> </head>
<body>
  <p> Text color is Red here</p>
  <div> <p> Text color is Black here </p> </div>
</body>
</html>
```

- This rule will render all the paragraphs in RED if they are a direct child of the <body> element. Other paragraphs put inside other elements like <div> or <td> would not have any effect of this rule.



The Attribute Selector

- We can also apply styles to HTML elements with particular attributes.
- The style rule below will match all the input elements having a type attribute with a value of text.

- Example:-

```
<style>
  input [ type = "text " ] { color: #000000;
                             }
</style>
```

- The advantage to this method is that the `< input type = "submit" >` element is unaffected, and the color applied only to the desired text fields.



Multiple Style Rules

- We may need to define multiple style rules for a single element. We can define these rules to combine multiple properties and corresponding values into a single block.

- Example:-

```
<style> h1 {  
    color: #36C;  
    font-weight: normal;  
    letter-spacing: .4em;  
    margin-bottom: 1em;  
    text-transform: lowercase; }  
  
</style>
```

- Here all the property and value pairs are separated by a semicolon (;). We can keep them in a single line or multiple lines. For better readability, we keep them in separate lines.



Grouping Selectors

- We can apply a style to many selectors if we have elements with the same style definitions.

- Example:-

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

- It will be better to group the selectors, to minimize the code.
- To group selectors, separate each selector with a comma.



CSS Comments

- Comments are used to explain the code, and may help when we 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:

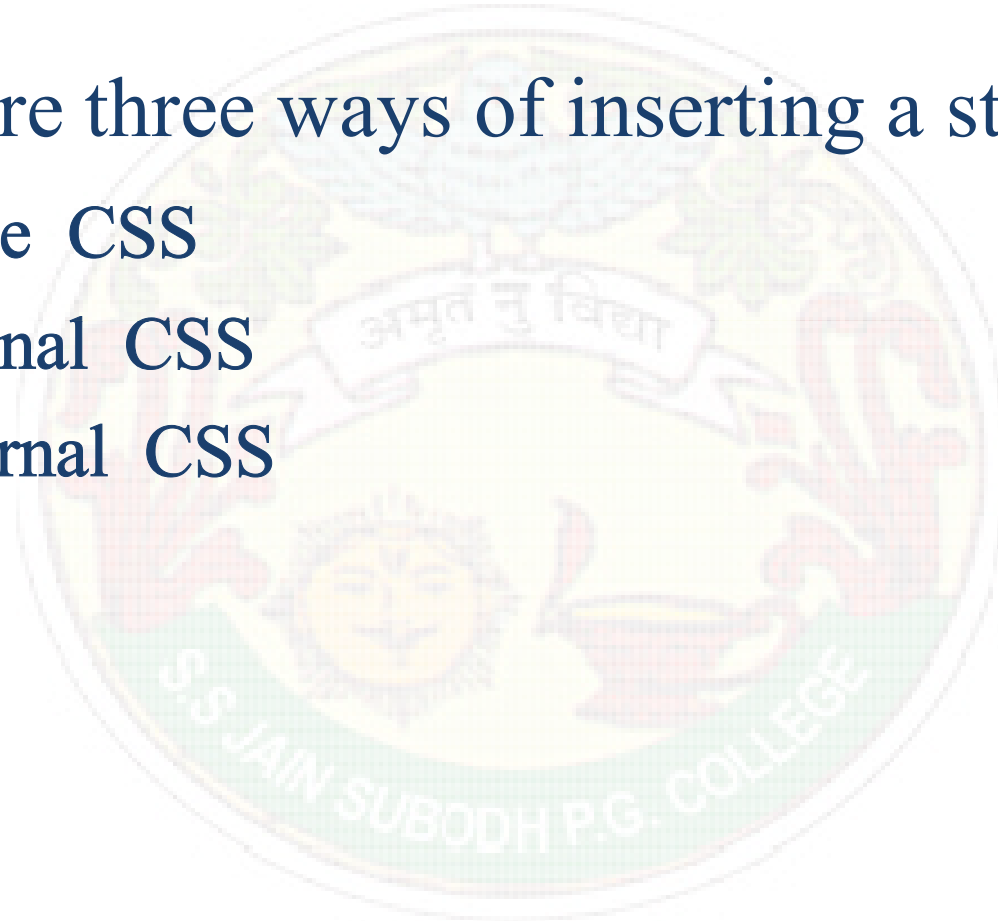
- Example:-

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



How to add CSS?

- There are three ways of inserting a style sheet:-
 - ❖ Inline CSS
 - ❖ Internal CSS
 - ❖ External 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.
- The example below shows how to change the color and the left margin of a `<h1>` element:
- Example:-

```
<!DOCTYPE html>
<html>
<body>
  <h1 style = "color : blue; margin-left : 30px; "> This is a heading </h1>
  <p> This is a paragraph. </p>
</body>
</html>
```



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.
- Example:-

```
<html>
<head>
<style> body{
    background-color : linen ; }
    h1 { color : maroon ;
    margin-left : 40px ; } </style>
</head>
<body>
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
</body>
</html>
```



External CSS

- With an external style sheet, we can change the look of an entire website by changing just one file!
- Each page must include a reference to the external style sheet file inside the `<link>` element. The `<link>` element goes inside the `<head>` section.
- Example:-

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



External CSS

➤ Example :-

```
<!DOCTYPE html>
<html>
<head>
  <link rel = "stylesheet " type = "text/css " href = "mystyle.css " >
</head>
<body>
  <h1>This is a heading</h1>
  <p>This is a paragraph.</p>
</body>
</html>
```

➤ Here is how the "mystyle.css" looks:-

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



Multiple Style Sheets

- If some properties have been defined for the same selector (element) in different style sheets, the value from the last read style sheet will be used.

- Example:-

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
<style> h1 {
        color: orange; }
</style>
</head>
<body>
  <h1>This is a heading</h1>
  <p>The style of this document is a combination of an external
  stylesheet, and internal style. The text color of h1 is Orange.</p>
</body>
</html>
```



CSS Rules Overriding

- We have discussed four ways to include style sheet rules in an HTML document.
- Here is the rule to override any Style Sheet Rule.
 - ❖ Any **Inline Style Sheet** takes the highest priority. So, it will override any rule defined in `<style>...</style>` tags or the rules defined in any external style sheet file.
 - ❖ Any rule defined as **Internal Style Sheet** in `<style>...</style>` tags will override the rules defined in any external style sheet file.
 - ❖ Any rule defined in the **External Style Sheet** file takes the lowest priority, and the rules defined in this file will be applied only when the above two rules are not applicable.



S. S Jain Subodh P.G. (Autonomous) College



Thank You