Introduction to CSS

Lec-1

What is an CSS ?

1 CSS stands for "Cascading Style Sheet".

2 CSS is used to give the style for HTML elements.

3 It is used to give proper alignment for the web page.

4 Used for Responsive web design.

Ways to connect CSS to our HTML

Way - 1 : Inline CSS

- Although CSS is a different language than HTML, it's possible to write CSS code directly within HTML code using inline style.
- To style an HTML element, we can add **style attribute directly to the opening tag**, after that we can set it equal to any CSS property we would like to apply.

Eg: <p style = 'color: red;'> Learning CSS! </p>

- If we want to add more than one style, we can do so using **semicolon** at the end of every style property we add.

Eg: <p style = 'color: red; font-size:20px'> Learning CSS! </p>

Way - 2 : Internal CSS

- The CSS code inside **<style>** element or tag is often referred as **internal stylesheet.**
- To create an internal stylesheet, a **<style>** element or tag must be placed inside of **<head>** element.

Eg:

```
<head>
    <style>
        **Your CSS code here**
     </style>
</head>
```

Note :

- **Inline Style** and **Internal CSS** is not an effective way to add style to our HTML, but in some circumstances it can be given.

Ways to connect CSS to our HTML

Way-3 : External CSS (Best Practice)

- To link our CSS file with HTML externally, we use **.css** file name extension, like so : **style.css.**
- With an external stylesheet, you can write all CSS code needed to style a page without sacrificing the readability and maintainability of your HTML file.
- We can use **<link>** element to link HTML and CSS file together.
- **<link>** tag or element is a self-closing element and must be placed within **<head>** element in an HTML file.
- **<link>** element basically takes 2 attributes:

1. **href :** Similarly to the href attribute in an **anchor <a></a>** tag, it takes address or a path of a css file as a value.
2. **rel :** This attribute describes relationship between your HTML document and the file which has been linked, since we have to link our css file, we must pass the value as **rel: 'stylesheet'.**

- Eg : **<link href = './style.css' rel = 'stylesheet'>**

Selectors in CSS

- Selectors in CSS are used to target specific HTML elements to be styled by declaration of CSS properties.
- There are 5 major types of Selectors in CSS

1. Simple Selectors
2. Combinator Selectors
3. Attribute Selectors
4. Pseudo Element Selectors
5. Pseudo Class Selectors

Selectors in CSS

1. Simple Selectors :

- Simple Selectors are of 5 types:

(i) 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 target any element using id selector, **hash (#)** symbol is used.
- Eg:

html

```
<h1 id = "demo"> Hey there, World! </h1>
```

css

```
#demo{
    color:'hotpink';
  }
```

(ii) 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.
- Eg:

html

```
<h1 class = "demo"> Hey there, World! </h1>
```

css

```
.demo{
    color:'beige';
  }
```

Selectors in CSS

(iii) Element Selector :

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

html

```html
<p> This a Normal Paragraph </p>
```

css

```css
p{
   font-size: 1.5rem;
}
```

(iv) Group Selector :

- The grouping selector selects all the HTML elements with the same style definitions.
- Eg:

html

```html
<h1> This is an Heading 1 <//h1>
<h2> This is an Heading 2 </h2>
<p> This is a paragraph </p>
```

css

```css
/* Without group selector adding same styles to 3
different elements */
h1{
   font-style: italic;
   font-weight: 600;
}
h2{
   font-style: italic;
   font-weight: 600;
}
p{
   font-style: italic;
   font-weight: 600;
}
/* With group selector */
h1,h2,p{
         font-style: italic;
         font-weight: 600;
}
```

(iv) Universal Selector :

- The universal selector selects all HTML elements on the page.
- It is targeted using **Asterisk (*)** symbol.
- Eg:

css

```css
/* Adding Default CSS using universal selector */
*{
  margin: 0;
  padding: 0;
  box-sizing: border-box;
 }
```

Note :

- **Specificity** is the order by which browser decides which CSS styles will be displayed.
- Order of specificity :

▼ ID Selectors (Most Specific)

▼ Class Selectors (Moderately Specific)

▼ TagName Selectors (Least Specific)

Selectors in CSS

2. Combinator Selectors :

- Combinator selectors are of 4 types:

(i) Descendent Combinator : Targets **direct child** as well as **indirect child.**

Eg:

html

```
<div>
  <p>This is a first paragraph</p>
  <p>This is a second paragraph</p>
  <span>
    <p>This is a first para under span</p>
  </span>
</div>
```

css

```
div p{
       font-size: 2rem;
       font-style: italic;
     }
```

(ii) Child Combinator : Targets **direct child** only.

Eg:

html

```
<div>
  <p>This is a first paragraph</p>
  <p>This is a second paragraph</p>
  <span>
    <p>This is a first para under span</p>
  </span>
</div>
```

css

```
div > p{
    color: darkmagenta;
    font-style: oblique;
    font-size: 3rem;
}
```

Selectors in CSS

(iii) Adjacent Sibling Combinator: Used to select an **element that is directly after a specific element.**

Eg:

html

```
<div>
   <p>This is a first paragraph</p>
   <p>This is a second paragraph</p>
   <span>
    <p>This is a first para under span</p>
   </span>
</div>
 <p>This is a immediate paragraph</p>
 <p>This paragraph is after it</p>
```

css

```
div + p{
  background-color: red;
  color: white;
}
```

(iv) General Sibling Combinator: Used to select **all elements that are next siblings of a specified element.**

Eg:

html

```
<div>
   <p>This is a first paragraph</p>
   <p>This is a second paragraph</p>
   <span>
    <p>This is a first para under span</p>
   </span>
</div>
 <p>This is a immediate paragraph</p>
 <p>This paragraph is after it</p>
```

css

```
div ~ p{
  background-color: blue;
  color: white;
}
```

Selectors in CSS

3. Attribute Selectors :

- The **Attribute Selector** can be used to target HTML elements that already contain attributes.
- Elements of same type can be targeted differently by their attribute or attribute value.
- The most basic syntax is an attribute surrounded by **[] square bracket.**
- Eg:

html

```
<a href="#" target="_blank">This is link1</a> |
<a href="#">This is link 2</a> |
<a href="#" target="_blank">This is link 3</a>
```

css

```
[target]{
   color: red;
   font-size: 1.6rem;
 }
```

- Above example will only select the **<a></a>** tag(s) that contains **target** attribute and apply the properties mentioned in the ruleset.
- Another Eg of Attribute selector:

html

```
<img src="./summer.jpeg" alt ="">
<img src="./winter.jpeg" alt ="">
```

css

```
[src*='summer']{
   width: 550px;
 }
 [src*= 'winter']{
   width: 200px;
 }
```

- In the first CSS ruleset it will look for **img** element with an attribute **src** which contains the string as **summer** and will set the **width** as **550px.**
- In the second CSS ruleset it will look for **img** element with an attribute **src** which contains the string as **winter** and will set the **width** as **200px.**

Selectors in CSS

4. Pseudo Element Selectors :

- Used to target **content only.**
- They are mainly of 6 types:

1. first-letter
2. first-line
3. Before
4. After
5. Selection
6. Marker

(i) first-letter : Used to add a **special style to the first letter of a text.** It can be applied to **block level elements** only.

Eg:

html

```
<p>Hola, Buenos dias</p>
```

css

```
p::first-letter {
  color: #ff0000;
  font-size: xx-large;
}
```

(ii) first-line : Used to add a **special style to the first line of a text.** It can be applied to **block level elements** only.

Eg:

html

```
<p> Hola <br>
Bueons dias
</p>
```

css

```
p::first-line {
  color: red;
  font-size: xx-large;
}
```

(iii) before : Used to **insert some content before the content of a specified element**. Use the **content property to specify the content to insert.**

Eg:

html

```
<h1>Welcome to My Website</h1>
```

css

```
 h1::before {
  content: url(smiley.gif);
}
```

(iv) after : Used to **insert some content after the content of a specified element**. Use the **content property to specify the content to insert.**

Eg:

html

```
<h1>Welcome to My Website</h1>
```

css

```
 h1::after {
  content: url(smiley.gif);
}
```

Selectors in CSS

(v) selection : Used to **style the the part of a text that is selected by a user.**

Eg:

html

```
<h1>Welcome to My Website</h1>
```

css

```
h1::selection{
    background-color: palevioletred;
    color: white;
}
```

(vi) marker: Used to style the **list item markers.**

Eg:

html

```
<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

css

```
::marker{
    color: seagreen;
    font-size: 2.5rem;
}
```

Selectors in CSS

5. Pseudo Class Selectors :

- Pseudo class selectors are broadly of 2 types:

1. Dynamic Pseudo class selectors
2. Structural Pseudo class selectors

A.) Dynamic Pseudo class selectors:

- They're of 4 types:

1. link
2. visited
3. hover
4. active

(i) link: Styles unvisited links

Eg:

html

```
<a href="#">This is a link</a>
```

css

```
a:link{
    color: orange;
    font-size: 3rem;
    text-decoration: none;
}
```

(ii) visited: Styles visited links

Eg:

html

```
<a href="#">This is a link</a>
```

css

```
a:visited{
    color: greenyellow;
}
```

(iii) active: Styles an activated link.

Eg:

html

```
<a href="#">This is a link</a>
```

css

```
a:active{
     color: blue;
}
```

(iv) hover: Styles an link/button on mouse over

Eg:

html

```
<button type="submit">Sign Up</button>
```

css

```
button:hover{
    cursor: pointer;
    background-color: orangered;
    color: black;
}
```

Selectors in CSS

B.) Structural Pseudo class selectors:

- They're of 3 types:

1. first-child
2. last-child
3. nth-child

(i) first-child: It matches a specific element that is the first child of another element.

Eg:

html

```
<div>
  <p>This is some text.</p>
  <p>This is some text.</p>
  <div>
    <p>This is some text.</p>
    <p>This is some text.</p>
  </div>
</div>

<div>
  <p>This is some text.</p>
  <p>This is some text.</p>
</div>
```

css

```
p:first-child {
  color: blue;
}
```

(ii) last-child: It matches a specific element that is the last child of another element.

Eg:

html

```
<div>
  <p>This is some text.</p>
  <p>This is some text.</p>
  <div>
    <p>This is some text.</p>
    <p>This is some text.</p>
  </div>
</div>

<div>
  <p>This is some text.</p>
  <p>This is some text.</p>
</div>
```

css

```
p:last-child {
  color: blue;
}
```

(iii) nth-child: It matches a specific element declared by the user.

Eg:

html

```
<div>
  <p>This is some text.</p>
  <p>This is some text.</p>
  <div>
    <p>This is some text.</p>
    <p>This is some text.</p>
  </div>
</div>

<div>
  <p>This is some text.</p>
  <p>This is some text.</p>
</div>
```

css

```
p:nth-child(2) {
  color: blue;
}
```