Cascading Style Sheets (CSS)

CSS, or **Cascading Style Sheets**, is the language used to style and enhance HTML documents. It defines the presentation of HTML elements on a web page, enabling changes to fonts, colors, sizes, spacing, column layouts, and animations. With CSS, you can adjust font sizes and colors, add backgrounds, and manage the layout, transforming a basic webpage into a visually appealing and user-friendly experience. CSS also simplifies layout management across multiple web pages by using external stylesheets stored in CSS files.

➔ What are the advantages of CSS?

- **Saves Time:** Write CSS once and reuse it across multiple HTML pages.
- **Easy Maintenance:** Change the style globally with a single modification.
- **Search Engine Friendly:** Clean coding technique that improves readability for search engines.
- **Superior Styles:** Offers a wider array of attributes compared to HTML.
- **Offline Browsing:** CSS can store web applications locally using offline cache, allowing offline viewing.

Q1.Difference between CSS 3.0 & 2.0

CSS 3.0	CSS 2.0
Both CSS and HTML were put into a single file, there was no concept of modules there.	CSS splits up different sections code into modules.
There were no new ways of writing the CSS rules.	There are new ways you can write CSS rules with a bunch of CSS selectors.
There is background compatibility maintained with CSS 3.0	There is no background compatibility with CSS 2.0
With CSS 3.0 special fonts can be used such as those in Google Fonts and Typecast.	With CSS 2.0 only web safe fonts can be used.
With CSS 3.0 the selectors were called as a sequence of simple selectors.com	With CSS 2.0 the concept of simple selectors was present.
With CSS 3.0 there is provision for automatically assigning rounded borders to objects21	With CSS 2.0 for rounded borders, coding the CSS styles were complex

CSS 3.0 has the capability to split text into various columns so that each text block appears as a layout of the newspaper.	CSS 2.0, splitting text into multiple columns required complex coding because the standard was not equipped enough to break the text into columns so that it would fit into a box.
CSS 3.0 supports the Border-Box property.	CSS 2.0 doesn't support the Border-Box property.

➔ CSS has three ways to style the HTML:

- **Inline**: Add styles directly to HTML elements using the **style attribute** (limited use).

  Syntax:

  <tagName style = "Value" > Content </tagName>

  For ex: <h1 style = "red"> INLINE CSS </h1>

- **Internal**: Place styles within a ***<style>*** tag inside the HTML file, usually within the <*head*> section.

  Syntax:

  <style>

  Selector {

  Property : Value ;

  }

  </style>

For ex:

<h1> CSS Introduction </h1>

<style>

h1{

color : balck ;

}

</style>

- **External**: Create a separate CSS file with a ***.css*** extension and link it to your HTML file using the <*link*> tag.

Syntax:

<link rel = “stylesheet” href = “ ”>

For ex:

<link rel = “stylesheet” href = “./style.css” >

Selectors:

CSS selectors target the HTML elements on your pages, allowing you to add styles based on their ID, class, type, attribute, and more.

Types of Selectors:

There are four types of selectors in CSS. They are

1. Simple selector.
2. Combinator selector.
3. Pseudo selector.
4. Attribute selector.

Q2.SIMPLE SELECTORS:

It is used to select the HTML elements based on their element name, id, attributes, etc.

Types:

1. Type selector.
2. Id selector.
3. Class selector.
4. Universal selector.
5. Grouping selector.

Type / Element Selector:

Selects HTML elements based on their tag names.

Example:

<!DOCTYPE html >

< html >

<head>

```
        <title>Type Selector </title>
    </head>
    <body>
        < h1> Type selector < /h1>
        <style>
            h1{ color : blue ;}
        < /style >
    < /body>
< /html >
```

Id Selector:

The id selector uses the **id attribute** of an HTML element to select a specific element. An **id** of the element is **unique** on a page to use the id selector.

To select an element with a specific id, write a hash (#) character, followed by the id of the element.

Example:

```
< h1 id = "d" > This is id selector < /h1 >
        #d {
          Color : red ;
            }
```

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:

```
<h1 class = "s"> Hello there! < /h1 >
    .s {
        Color : darkblue ;
        }
```

Universal Selector:

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

Example:

```
* {
  color: blue;
 }
```

Grouping Selector:

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

Example:

```
h1 {
  color: red;
}

h2 {
  color: red;
}

p {
  color: red;
}
```

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

```
h1,h2,p {
  color: red;
}
```

Q3. What are properties & values?

CSS properties are the fundamental building blocks for creating stunning and functional web experiences.

CSS properties are used to set the style or assign behaviour to HTML elements. Each CSS property consists of two parts: property_name and property_value, with the property_value enclosed within double quotes (” “).

Example:

```
#m {
        width: 400px;
        height: 299px;
        background-color: green;
    }
```

From the above example properties are defined at left side (width, height and background-color). Values are defined at right side.