UNIT - 3 Cascading Style Sheet (CSS)

- Introduction
- Understanding the concept of CSS
- Advantages and disadvantages
- CSS syntax
- Class and ID Selectors
- Types of style sheets

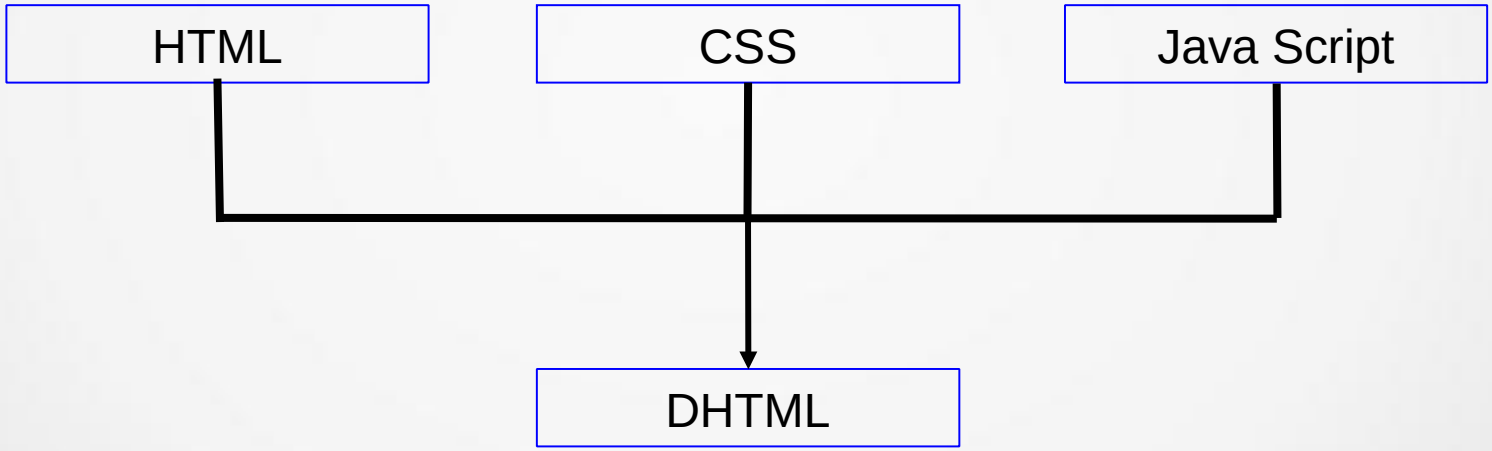
UNIT - 3 Cascading Style Sheet (CSS)

- CSS Properties and Attributes
  - Text Properties and Attributes
  - Font Properties
  - CSS Padding
  - CSS Color
  - CSS Links
  - List Properties
  - Table Properties

•Dynamic HTML - (DHTML)

- Dynamic HTML, or DHTML, is an umbrella term for a collection of technologies used together to create **interactive** and **animated web sites** by using a combination of
  - a static markup language (**such as HTML**),
  - a client-side scripting language (**such as JavaScript**),
  - a presentation definition language (**such as CSS**).

DHTML Architecture



CSS - Introduction

- CSS stands for **Cascading Style Sheets.**
- CSS define **how to display HTML elements.**
- CSS were added to **HTML 4.0 to solve a problem.**
- CSS can **save a lot of work.**
- CSS are **stored in .css files**.

Advatage & Disadvantage of CSS

- **Advantages**
- **CSS saves time** − You can write CSS once and then reuse same sheet in multiple HTML pages.
- **Pages load faster** − If you are using CSS, you do not need to write HTML tag attributes every time.
- **Easy maintenance** − To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.
- **Global web standards** − Now HTML attributes are being deprecated and it is being recommended to use CSS.
- **Offline Browsing** − CSS can store web applications locally with the help of an offline catche.

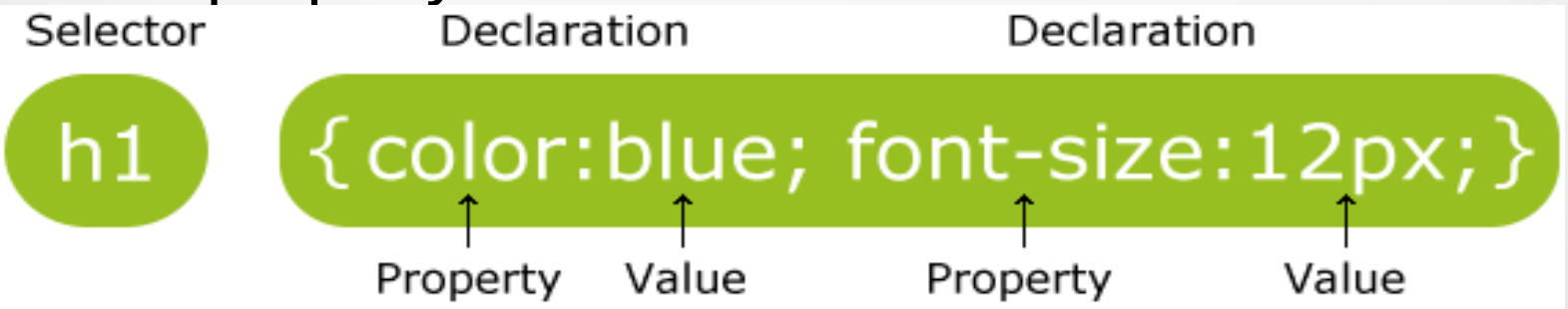
Advatage & Disadvantage of CSS

- **Disadvantages**
- CSS works differently on different browsers. IE and Opera supports CSS as different logic.

- You will need to test your web pages with multiple browsers in order to ensure compatibility when using style sheets.

CSS Syntax

- A CSS rule has two main parts: **a selector,** and one or more **declarations:**
  - The selector is **normally the HTML element** you want to style.
  - Each declaration **consists of a property and a value**.
  - The property is the style attribute you want to change. Each property has a value.



CSS Example

- A CSS declaration always ends with a semicolon,
  and declaration groups are surrounded by curly brackets:

  P{color:red;text-align:center;}

- More readable form is:

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

DEMO_1.html

CSS Grouping Selectors

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

```
h2 {
    text-align: center;
    color: yellow;
}
```

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

CSS Comments

- Comments are used to explain your code, and may help you when you edit the source code at a later date. Comments are ignored by browsers.
- A CSS comment begins with "/*", and ends with "*/", like this:
  - ***/*This is a comment*/***

```
p
{
text-align:center;
/*This is another comment*/
color:black;
font-family:arial;
}
```

The **id Selector** and **class selector**

- In addition to setting a style for a HTML element, CSS allows you to specify your own selectors called
  - **"id"** and
  - **"class"**

The id Selector

- The id selector is used to s**pecify a style for a single, unique element.**
- The id selector uses the id attribute of the HTML element, and is **defined with a "#"**.
- The style rule below will be applied to the element with **id="para1":**
- **Example :**

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

DEMO_2.html

The class Selector

- The class selector is **used to specify a style for a group of elements.** Unlike the id selector, the class selector is **most often used on several elements.**
- This a**llows you to set a particular style for many HTML elements with the same class.**
- The class selector uses the HTML class attribute, and is **defined with a "."**

The class Selector

- In the example below, all HTML elements with class="center" will be center-aligned:
- Example:

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

- You can also specify that only specific HTML elements should be affected by a class. In the example below, all p elements with class="center" will be center-aligned:
- Example:

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

DEMO_3.html

Type of Style Sheets

- There are three ways of inserting a style sheet:
  - **External style sheet**
  - **Internal style sheet**
  - **Inline style**

Type of Style Sheets

- **External style sheet**
  - An external style sheet is ideal when the style is applied to many pages.
  - With an external style sheet, you can change the look of an entire Web site by changing one file. Each page must link to the style sheet using the **<link> tag.** The <link> tag goes inside the head section:

Type of Style Sheets

- **External style sheet**

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

Type of Style Sheets

- **External style sheet - How to write external style sheet**
  - An external style sheet can be written in any text editor. The file should not contain any html tags. Your style sheet should be saved with a .css extension. An example of a style sheet file is shown below:

```
hr {color:sienna;}

p {margin-left:20px;}

body {background-image:url("images/back40.gif");}
```

DEMO_4.html

Type of Style Sheets

- **Internal style sheet**
  - Internal style sheet should be used when a single document has a unique style. You define internal styles in the head section of an HTML page, **by using the <style> tag**.

Type of Style Sheets

- **Internal style sheet**

```
<head>
   <style type="text/css">
        hr {color:sienna;}
       p {margin-left:20px;}
       body {background-image:url("images/back40.gif");}
   </style>
</head>
```

DEMO_1.html

Type of Style Sheets

- **Inline style sheet**
  - An inline style loses many of the advantages of style sheets by mixing content with presentation.
  - To use inline styles you use the style attribute in the relevant tag. The style attribute can contain any CSS property. The example shows how to change the color and the left margin of a paragraph.

```
<p style="color:blue;margin-left:20px">
    This is a paragraph.
</p>
```

DEMO_5.html

CSS PROPERTIES AND ATTRIBUTES

- Text Properties and Attributes
- Font Properties
- CSS Padding
- CSS Color
- List Properties
- Table Properties
- CSS Links

Text Properties

- **Text Color**
  - The color property is used to set the color of the text.
  - With CSS, a color is most often specified by:
    - a HEX value - like "#ff0000"
    - an RGB value - like "rgb(255,0,0)"
    - a color name - like “red”
- The default color for a page is defined in the body selector.
- **Example:**

```
body {color:blue;}
h1 {color:#00ff00;}
h2 {color:rgb(255,0,0);}
```

Text Properties

- **Text Alignment**
  - The text-align property is used to set the horizontal alignment of a text. Text can be centered or left or right or justified.
  - When text-align is set to "justify", each line is stretched so that every line has equal width, and the left and right margins are straight (like in magazines and newspapers).
  - **Example**

h1 {text-align:center;}

p.date {text-align:right;}

p.main {text-align:justify;}

DEMO_6.html

Text Properties

- **Text Decoration**
  - The text-decoration property is used to set or remove decorations from text.
  - The text-decoration property is mostly used to remove underlines from links for design purposes:
  - Example:

h1 {text-decoration:overline;}

h2 {text-decoration:line-through;}

h3 {text-decoration:underline;}

h4 {text-decoration:blink;}

DEMO_7.html

Text Properties

- **Text Transformation**
  - The text-transform property is used to specify uppercase and lowercase letters in a text.
  - It can be used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word.
  - **Example:**

p{text-transform:uppercase;}

p{text-transform:lowercase;}

p{text-transform:capitalize;}

Text Properties

- **Text Indent**
  - The text-Indent property is used to specify indents of the first line.
  - The value of indents should be in either px, in or %
  - Example:

    p{text-indent: 1.5in;}

    p{text-indent: 10%;}

    p{text-indent: 50px;}

DEMO_8.html

Text Properties

- **Text Vertical align**
  - The text-vertical align property is used to specify vertical position.
  - The value for Vertical align are Baseline, sub, super, top, text-top, middle, bottom, text-bottom, percentage
  - Example:

    p{vertical-align:top;}

    p{vertical-align:bottom;}

DEMO_9.html

Text Properties

- **Letter Spacing and Word Spacing**
  - Letter sapcing attribute is used to increase and decrease the space between the characters of a word.
  - Word Spacing is used to increase and decrease the space between words in text.
  - **Example:**

h1{letter-spacing : 2px;}

h2 { letter-spacing : -3px}

p {word-spacing : 30px}

DEMO_10.html

Text Properties

- **Line Height**
  - This attribute is used to set the distances between the lines in any text in a Web page.
  - Example:

p { line-height : 90%}

DEMO_11.html

CSS FONT

- **FONT**
  - CSS font properties define the **font family, boldness, size, and the style of a text.**
- **CSS Font Families**
  - In CSS, there are two types of font family names:
    - **generic family** - a group of font families with a similar look (like "Serif" or "Monospace")
    - **font family** - a specific font family (like "Times New Roman" or "Arial")

CSS FONT

- **CSS Font Families**
  - The font family of a text is set with the font-family property.
  - More than one font family is specified in a comma-separated list:
  - **Example:**

    p{font-family:"Times New Roman", Times, serif;}

CSS FONT

- **Font Style**
  - The font-style property is mostly used to specify italic text.
  - This property has three values:
    - normal - The text is shown normally
    - italic - The text is shown in italics
    - oblique - The text is "leaning”
  - **Example:**

p.normal {font-style:normal;}

p.italic {font-style:italic;}

p.oblique {font-style:oblique;}

DEMO_12.html

CSS FONT

- **Font Size**
  - *The font-size property sets the size of the text. The font-size value can be an absolute or relative size.*
  - ***Absolute size:***
    - *Sets the text to a specified size*
    - *Does not allow a user to change the text size in all browsers (bad for accessibility reasons)*
    - *Absolute size is useful when the physical size of the output is known*
  - ***Relative size:***
    - *Sets the size relative to surrounding elements*
    - *Allows a user to change the text size in browsers.*

CSS FONT

- **Set Font Size With Pixels**
  - *Setting the text size with pixels, The text can be resized in all browsers using the zoom tool, gives you full control over the text size:*
  - *Example:*

```
h1 {font-size:40px;}

p {font-size:14px;}
```

DEMO_13.html

CSS FONT

- **Set Font Size With Em**
  - *To avoid the resizing problem with Internet Explorer, many developers use em instead of pixels.*
  - *The em size unit is recommended by the W3C.*
  - ***1em is equal to the current font size. The default text size in browsers is 16px.*** *So, the default size of 1em is 16px.*
  - ***The size can be calculated from pixels to em using this formula: pixels/16=em***
  - *Example:*

```
h1 {font-size:2.5em;} /* 40px/16=2.5em */

p {font-size:0.875em;} /* 14px/16=0.875em */
```

  - *With the em size, it is possible to adjust the text size in all browsers.*

DEMO_14.html

CSS FONT

- **Set Font Weight**
  - The font-weight property specifies the weight of a font:
  - **Example:**

```
p.normal {font-weight: normal;}
p.thick {font-weight: bold;}
```

DEMO_15.html

CSS FONT

- **Set Font Variant**
  - The font-variant property specifies whether or not a text should be displayed in a small-caps font.
  - In a small-caps font, all lowercase letters are converted to uppercase letters. However, the converted uppercase letters appears in a smaller font size than the original uppercase letters in the text.
  - **Example:**

    p.normal { font-variant: normal; }

    p.small {font-variant: small-caps;}

DEMO_16.html

CSS Padding Properties

- The CSS padding properties are used to generate space around content.
- The padding properties set the size of the white space between the element content and the element border.
- There are CSS properties for setting the padding for each side of an element (top, right, bottom, and left):
  - **padding-top**
  - **padding-right**
  - **padding-bottom**
  - **padding-left**

CSS Padding Properties

- The CSS padding properties define the white space between the element content and the element border.
- All the padding properties can have the following values:
  - **length** - specifies a padding in px, pt, cm, etc.
  - **%** - specifies a padding in % of the width of the containing element
  - **Inherit** - specifies that the padding should be inherited from the parent element.
- **Example:**

```
p {
    padding-top: 50px;
    padding-right: 30px;
    padding-bottom: 50px;
    padding-left: 80px; }
```

DEMO_17.html

CSS Padding Properties

- **CSS Padding - Shorthand Property**
- To shorten the code, it is possible to specify all the padding properties in one property.
- **Example:**

```
p {
    padding: 50px 30px 50px 80px;
}
```

DEMO_17.html

CSS Colors

- Colors are displayed combining **RED, GREEN, and BLUE** light.
- CSS colors are defined using a hexadecimal (HEX) notation for the combination of Red, Green, and Blue color values (RGB).
- The lowest value that can be given to one of the light **sources is 0 (HEX 00). The highest value is 255 (HEX FF).**
- HEX values are written as 3 double digit numbers, starting with a # sign.

CSS Colors

Color	Color HEX	Color RGB
	#000000	rgb(0,0,0)
	#FF0000	rgb(255,0,0)
	#00FF00	rgb(0,255,0)
	#0000FF	rgb(0,0,255)
	#FFFF00	rgb(255,255,0)
	#00FFFF	rgb(0,255,255)
	#FF00FF	rgb(255,0,255)
	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)