04.08.25

Unit - 1

* Introduction to CSS

* CSS stands for 'cascading style sheets. It is a fundamental technology in web development used to style and design HTML content.

* What is CSS?

* CSS describes how HTML elements are to be displayed on screen, paper, or in other media. It controls the layout, colors, fonts, spacing

* Importing a Stylesheet

* In web development, importing a stylesheet means linking an external CSS file to your HTML document so you can apply styles (colors, fonts, layouts, etc.) to your webpage

* there are two main ways to import a stylesheet into an HTML file

1. Using the <link> Tag

* This method imports a CSS file into an HTML document

* program:-

HTML

```
<!DOCTYPE html>
<html>
<head>
  <title>Linked CSS Example</title>
  <link rel="stylesheet" href="style.css">
</head>
<body>
  <h1>Hello, World!</h1>
</body>
</html>
```

css

```
h1 {
  color: darkblue;
  text-align: center;
}
```

2. using @import in css

* this method is written inside a css file to import another css file

* program:

css

```
@import url("theme.css");

h1 {
  font-family: Arial;
}

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

*

method	use location	use case
<link>	In HTML <head>	standard for web pages
@import	Inside css file	Modular css, optional styles

* CSS rules

* CSS rules define how HTML elements should look on a web page - including their layout, color, font, spacing

* Structure of a CSS Rule

A CSS rule consists of a selector, property, and value

* Ex:-

```
p{
   color: blue;
   font-size: 16px;
}
```

*

Selector	Description
*	Universal selector (applies to all elements)
h1, h2	Selects multiple tags
.class name	Selects elements with a class

* program:-

```
body{
   background-color: #f0f0f0;
   margin: 0;
   font-family: Arial, sans-serif;
}

h1{
```

```
    color: navy;
    text-align: center;
}
p {
    line-height : 1.6;
    color: #333;
}
```

* Style Types
* There are Three main types of CSS Styling
* ① Inline CSS
* Style is applied directly to an individual HTML element using the style attribute
* Ex:-

```
<p style="color: red; font-size: 18px; ">
This is an inline styled paragraph </p>
```

* ② Internal CSS
* CSS is written inside a <style> tag within the <head> of an HTML file.
* Ex:-

```
<!DOCTYPE html>
<html>
<head>
    <style>
        p{
            color: blue;
            font-size: 18 px;
        }
```

```
</style>
</head>
<body>
    <p> This is an internally styled paragraph
        </p>

</body>
</html>
```

* ③ External CSS

* CSS is placed in a separate .css file and linked to the HTML using <link>

* Ex: index.html

```
<head>
    <link rel="stylesheet" href="style.css">
</head>
```

style.css

```
p{
 color: green;
 font-size : 18px;
}
```

* Selectors
  * A selector tells the browser which HTML elements to style

* Types of CSS selectors :-
  * ① Universal selector
    * selects all elements

* Ex :-

```
* {
    margin:0;
    padding:0;
}
```

* ② Element selector
  * selects elements by tag name

* Ex :-

```
h1{
    color:red;
}
```

* ③ class selector
  * Targets elements with a specific class attribute

* Ex :-

```
.myclass{
    font-style:italic;
}
```

```
<p class="myclass"> This is styled </p>
```

* ④ ID selector
  * Targets a unique ID in the document.

* Ex:-

```
#header{
    background-color: lightblue;
}
<div id = "header"> My Header </div>
```

* ⑤ Group selector
  * Applies the same style to multiple elements

* Ex:-

```
h1, h2, h3{
    font-family: Arial;
}
```

* Program:-

```
<!DOCTYPE html>
<html>
<head>
    <style>
        h1{
            color: darkblue;
        }
        .highlight{
            background-color: yellow;
        }
        #footer{
            text-align: center;
            font-size: 14px;
        }
        p:hover{
            color: red;
        }
```

```
</style>
</head>
<body>
  <h1> CSS Selectors Example </h1>
  <p> This is a normal paragraph. </p>

  <p class = "highlight" > This is a highlighted
      paragraph </p>
    <div id = "footer" > This is the footer. </div>
</body>
</html>
```

Fonts and Typography

* Fonts and typography are crucial in web development to make websites readable, aesthetic, and user-friendly. Typography involves font selection, size, line spacing, weight, and layout which directly affects user experience

* Typography
  * Font family (e.g., Arial, Times new Roman)
  * Font size
  * Font weight (boldness)
  * Line height (spacing between lines)
  * Letter spacing (space between characters)
  * Text alignment
  * Text decoration (underline, etc.)

★ Font properties

property	description
font-family	specifies the font
font-size	sets the size of the font
font-weight	controls boldness
line-height	spacing between lines
Letter-spacing	space between characters
text-align	aligns text (left, center, right, justify)
text-transform	uppercase / lowercase

★ program:-

```
<!DOCTYPE html>
<html>
<head>
  <link
href="https://fonts.googleapis.com/css2?family
=Roboto&display=swap" rel="stylesheet">
  <style>
    body{
      font-family: 'Roboto', sans-serif;
      font-size: 18px;
      line-height: 1.6;
      color: #333;
      text-align: justify;
      margin: 20px;
    }
```

```
h1{
  font-size: 32px;
  text-align: center;
  text-transform: uppercase;
}
p{
  letter-spacing : 0.5px;
}
</style>
</head>
<body>
<h1>Web Typography</h1>
<p> Typography interact with a website </p>
</body>
</html>
```

Managing Text Styles

* Text style refers to the visual appearance of text on a webpage. It includes properties that control how text looks - such as its font, size, weight, color, decoration, and transformation

  * Font - Font
  * Size - Size
  * Light - weight

* **font-family** - font-family sets the typeface for text. It defines a list of fonts to try in order

Ex:-
font-family: Arial, Sans-serif;
"Hii, how are you, I'm font-family"

font-size - Font-size sets the size of text.
It defines a list of font size

Ex:-
font-size: 18px;
"Hii",
font-size: 28px;
"how are you"

font-weight: controls the boldness or thickness of text

Ex:-
normal, bold, or numeric values like 100 (thin) to 900 (extra bold)

font-style: Defines the slant or style of the font, such as normal, italic, or oblique

Ex:-
font-style: italic; makes text slanted

Font-color: Defines the colors of the text

Ex:-
color: red;
color: #333333;
color: rgb(0,0,0);

* CSS colors:-
  sets the color of the text

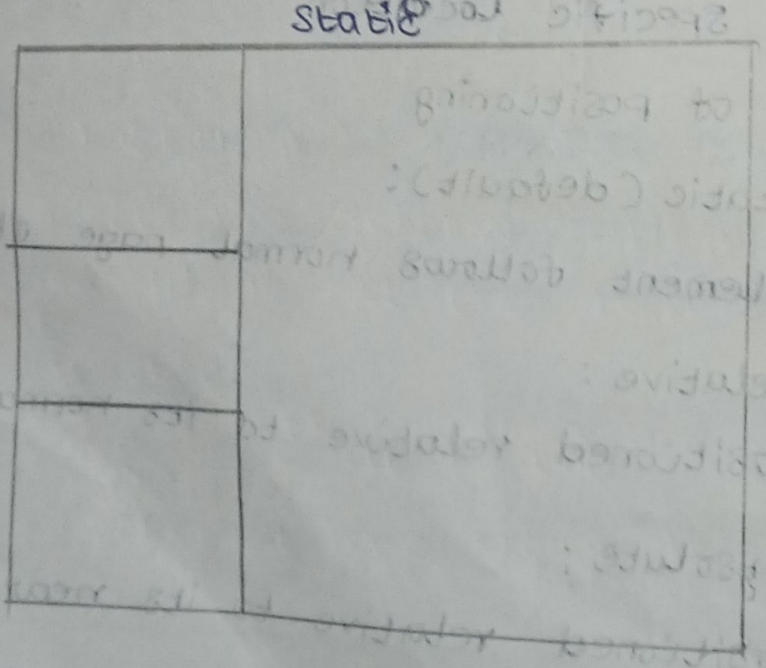
* 1. Named colors
  * predefined color names in CSS
  * Easy to use, but limited in variety
  * Ex:-
    color: red; color: blue; color: green;

* 2. Hexadecimal (#rrggbb)
  * Represents colors using hex values
  * rr, gg, bb = Red, Green, Blue (in hex: 00 to FF)
  * Ex:-
    color: #ff0000; (pure red)

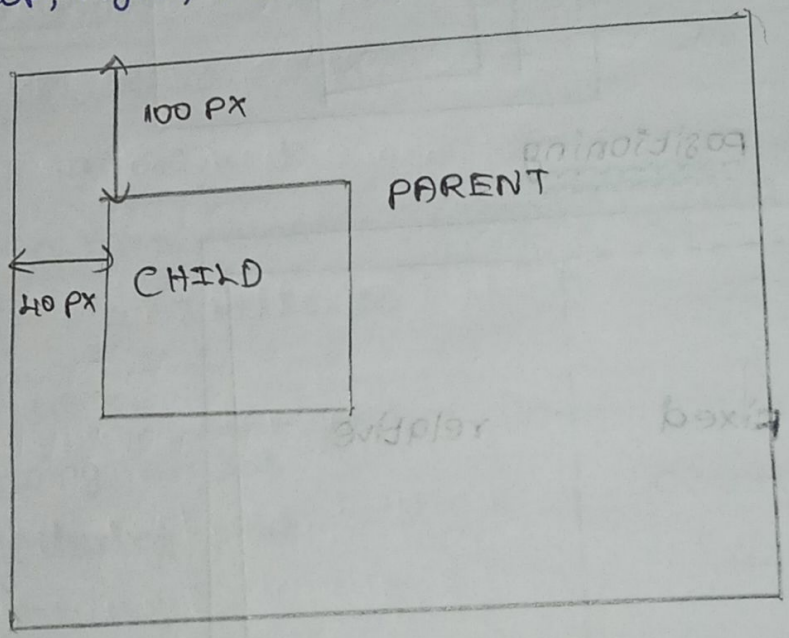
---

* Positioning Elements
* positioning controls how an element is placed in the document layout, either in normal flow or at a specific location
* Types of positioning
  * static (default):
    * Element follows normal page flow
    * Relative:
    * positioned relative to its normal position
    * Absolute:
    * positioned relative to its nearest
    * positioned ancestor (or <body> if none)

* Fixed:
  * Stays fixed to the viewport (doesn't move on scroll)
* Sticky:
  * Switches between relative and fixed depending on scroll position.
* Static positioning
  * Static (the default), it follows the normal document flow - meaning it appears where it naturally fits in the layout, based on the surrounding elements
    * It does not respond to top, bottom, left or right properties.
    * Elements are placed one after another, like blocks stacking vertically

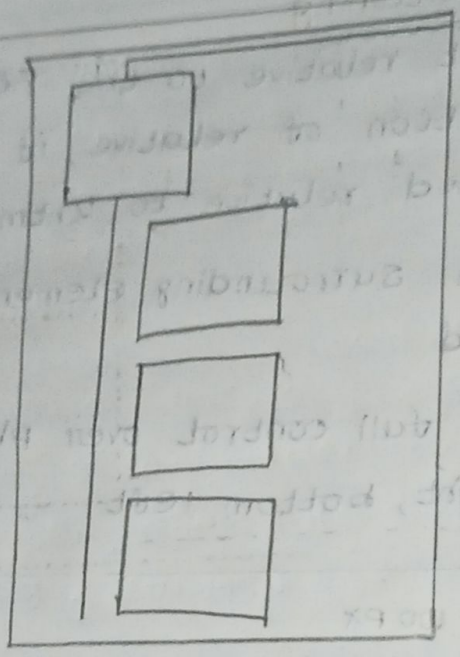


* Absolute positioning
* positions it relative to the nearest ancestor with a position of relative, if anone exists, it's positioned relative to <html>
    * Ignores surrounding elements - no space is reserved
    * Allows full control over placement using top, right, bottom, left

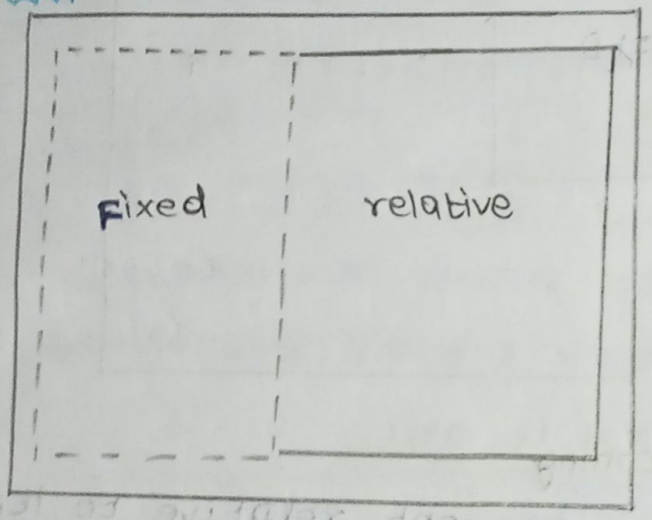


* Relative positioning
    * positions the element relative to its original position
    * keeps the space it originally occupied

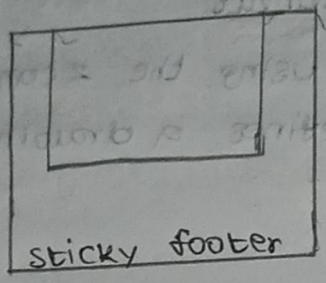
Moves visually using top, left, right, bottom

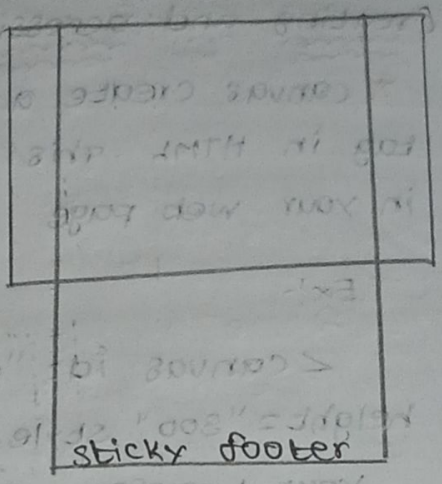


* Fixed positioning



* sticky positioning
  * Acts like relative until a scroll threshold is met, then becomes fixed
* characteristics:
  * Based on scroll position
  * parent must have enough height for it to work





The HTML5 canvas

* what is canvas?
  * An HTML element (<canvas>) used to draw graphics via Javascript
* common uses
  * 1. Drawing shapes
  * 2. Rendering text
  * 3. Animations
  * 4. Game graphics
* syntax:-
* <canvas id="mycanvas" width="500" height="300">
  Your browser does not support the HTML5 canvas tag </canvas>
* Drawing shapes
  * Draw a Line
  * Draw a Rectangle
  * Draw a circle
  * Draw Text

* Creating and accessing a canvas

* canvas create a and using the <canvas> tag in HTML5 this tag defines a drawing area in your web page

* Ex:-

* <canvas id="mycanvas" width="400" height="300" style="border:1px solid #000000;">
your browser does not support the HTML5 canvas tag </canvas>

* program:-

```
<!DOCTYPE html>
<html>
<head>
<title> canvas Access Example </title>
</head>
<body>

<!--canvas Element-->
<canvas id="mycanvas" width="400" height="300" style="border:1px solid black;">
your browser does not support canvas </canvas>

<!--Javascript to Access and draw-->
<script>
const canvas = document.getElementById("mycanvas");
const ctx = canvas.getContext("2d");
```

```
// Draw a blue rectangle
ctx.fillStyle = "blue";
ctx.fillRect(50, 50, 150); // x, y, width, height
</script>
</body>
</html>
```

Writing Text to the Canvas

* The HTML5 canvas allows you to display text using JavaScript. You can control the font, size, color, position, alignment, and outline or fill the text.

* 1. Basic methods to write text

method	Description
ctx.fillText(text, x, y)	Draws filled text
ctx.strokeText (text, x, y)	Draws outlined text

* 2. Set Text properties

Property	Example	Description
ctx.font	"20px Arial"	Sets font size and family
ctx.fillStyle	"blue"	Sets the color for fillText()
ctx.strokeStyle	"red"	Sets the color for strokeText()
ctx.textAlign	"center", "left", "right"	Align the text

ctx.textBaseline	"top", "middle", "bottom"	vertical alignment

Drawing lines on canvas

```
ctx.beginPath();
ctx.moveTo(100,100); //Start point
ctx.lineTo(800,100); //End point
ctx.strokeStyle = "red";
ctx.lineWidth = 5;
ctx.stroke();
```

* beginPath(): Starts a new path
* moveTo(x,y): Sets the start point
* lineTo(x,y): Sets the end point
* stroke(): Actually draws the line

Basic steps to Draw a Line

To draw a line, you'll typically follow these steps:

* Create the <canvas> element in HTML
* Access the canvas in Javascript
* Use canvas 2D context methods to draw the line

* Working with curves
* HTML5 canvas supports two types of curves:
  Quadratic Bezier curves - one control point
  * Cubic Bezier curves - Two control points

```
ctx.beginPath();
ctx.moveTo(50,200); // start point
ctx.quadraticCurveTo(150,100,250,200); // control and
                                       end point

ctx.strokeStyle="purple";
ctx.stroke();
```

* Rectangle and quardratic curve

```
//Draw rectangle
ctx.fillStyle="orange";
ctx.fillRect(100,80,150,70);
//Draw quadratic curve
ctx.beginPath();
ctx.quadraticCurveTo(300,100,550,200);
ctx.strokeStyle="purple";
ctx.linewidth=3;
ctx.stroke();
```

* Bezier curves

```
//Draw cubic Bezier curve
ctx.beginPath();
ctx.moveTo(50,250);
ctx.bezierCurveTo(150,350,450,150,550,250);
```

```
ctx.strokeStyle = "green";
ctx.lineWidth = 3;
ctx.stroke();
</script>
</body>
</html>
```

Drawing Lines

1. Horizontal line (<hr>)

   Simplest way to draw a horizontal line

★ Ex:-

```
<hr style="border: none; height: 2px;
  background-color: black;">
```

2. CSS Borders

   Use borders on elments to simulate lines

★ Ex:-

```
<div style="border-bottom: 2px solid black;
    width: 200px;"></div>
```

3. Line with ::before or ::after

★

```
.line::after {
  content: "";
  display: block;
  height: 1px;
  background: black;
  margin-top: 10px;
}
```

4. CSS transform for diagonal line

* To rotate a line:

```
<div style="
    width: 200px;
    height: 2px;
    background-color: black;
    transform: rotate(45deg);"></div>
```

Manipulating Images

* 1. Basic Manipulation with HTML & CSS

* html

```
<img src="image.jpg" alt="example"
  width="300" height="200">
```

CSS

```
img{
  width: 300px;
  height: auto;
  border-radius: 10px;
}
```

* 2. Responsive design

* html

```
<picture>
  <source media="(min-width: 800px)"
    srcset="large.jpg">
  <source media="(min-width: 500px)"
    srcset="medium.jpg">
  <img src="small.jpg" alt="Responsive image">
</picture>
```

* Some types:-
  * Manipulating with javascript
  * Advanced manipulation with canvas
  * Using SVG for effects and clipping
  * Exporting edited images

The <audio> and <video> element

* plays audio files such as music, podcasts, or sound effects.

Attribute	Description
controls	displays built in audio controls
autoplay	starts playing automatically
loop	Repeats the audio when finished
muted	starts with audio muted
preload	Hints to the browser on how to load audio

* plays video files directly in the browser
* Ex:-
  * A custom audio or video player
  * A playlist
  * Live streaming
  * Audio visualization

HTML5 Features

Element	purpose
<header>	Intro or top of a section/ page
<footer>	Bottom of a section/page
<article>	self-contained content block
<section>	Thematic group content
<nav>	navigation menus
<aside>	sidebar or tangential content
<main>	main content of page

Geolocation and the GPS services

- GPS (Global positioning system)
- Wi-Fi networks
- Cell Tower triangulation
- Ip address approximation

Use case	Example
Maps and directions	Google maps, waze
Nearby services/ products	Food delivery, ride sharing apps
Location based content	Local news, weather or alerts
Travel apps	Track movements, show trails/routs

* The <audio> and <video> element

* Ex:-

```
<video width = "600" height = "380" controls>
<source src = "aa.mp4" type = "video/mp4">
your browser does not support HTML video
</video>
<input type = "button" value = "click" name = "bt1">
</body>
</html>
```

* Ex:-

```
<audio width = "300" height = "140" controls>
<source src = "aa.mp4" type = "audio/mp4">
your browser does not support HTML audio
</audio>
<input type = "button" value = "click" name = "bt1">
</body>
</html>
```