K. J. Somaiya College of Engineering, Mumbai-77
(A Constituent College of Somaiya Vidyavihar University)

Batch : D-2 **Roll No. : 16010122323**

Experiment : 2

TITLE: Develop and demonstrate the usage of inline, internal and external style sheet using CSS

AIM: To demonstrate usage of CSS

Expected Outcome of Experiment: Use CSS to prepare the layout of web pages.

Books/ Journals/ Websites referred:

1. .

Importance of CSS in designing of a website is to be explained. Explain various ways to use CSS. Also explain how to change background colour of page, adding and editing border types, adding navigation bars, usage of various types of 2D and 3D transformation.

Description of the CSS style code with its effect at output:

Index.**html :**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
  <link rel="stylesheet" href="wpl.css">
</head>
<body>
<div class="container" id="container">
    <div class="form-container sign-up-container">
        <form action="#">
            <h1>Create Account</h1>
            <div class="social-container">
                <a href="#" class="social"><i class="fab fa-facebook-f"></i></a>
                <a href="#" class="social"><i class="fab fa-google-plus-g"></i></a>
                <a href="#" class="social"><i class="fab fa-linkedin-in"></i></a>
            </div>
            <span>or use your email for registration</span>
            <input type="text" placeholder="Name" />
            <input type="email" placeholder="Email" />
            <input type="password" placeholder="Password" />
            <button>Sign Up</button>
        </form>
```

```
    </div>
    <div class="form-container sign-in-container">
        <form action="#">
            <h1>Sign in</h1>
            <div class="social-container">
                <a href="#" class="social"><i class="fab fa-facebook-f"></i></a>
                <a href="#" class="social"><i class="fab fa-google-plus-g"></i></a>
                <a href="#" class="social"><i class="fab fa-linkedin-in"></i></a>
            </div>
            <span>or use your account</span>
            <input type="email" placeholder="Email" />
            <input type="password" placeholder="Password" />
            <a href="#">Forgot your password?</a>
            <button>Sign In</button>
        </form>
    </div>
    <div class="overlay-container">
        <div class="overlay">
            <div class="overlay-panel overlay-left">
                <h1>Welcome Back!</h1>
                <p>To keep connected with us please login with your personal info</p>
                <button class="ghost" id="signIn">Sign In</button>
            </div>
            <div class="overlay-panel overlay-right">
                <h1>Hello, Friend!</h1>
                <p>Enter your personal details and start journey with us</p>
                <button class="ghost" id="signUp">Sign Up</button>
            </div>
        </div>
    </div>
</div>
</body>
</html>
```

Style.css :

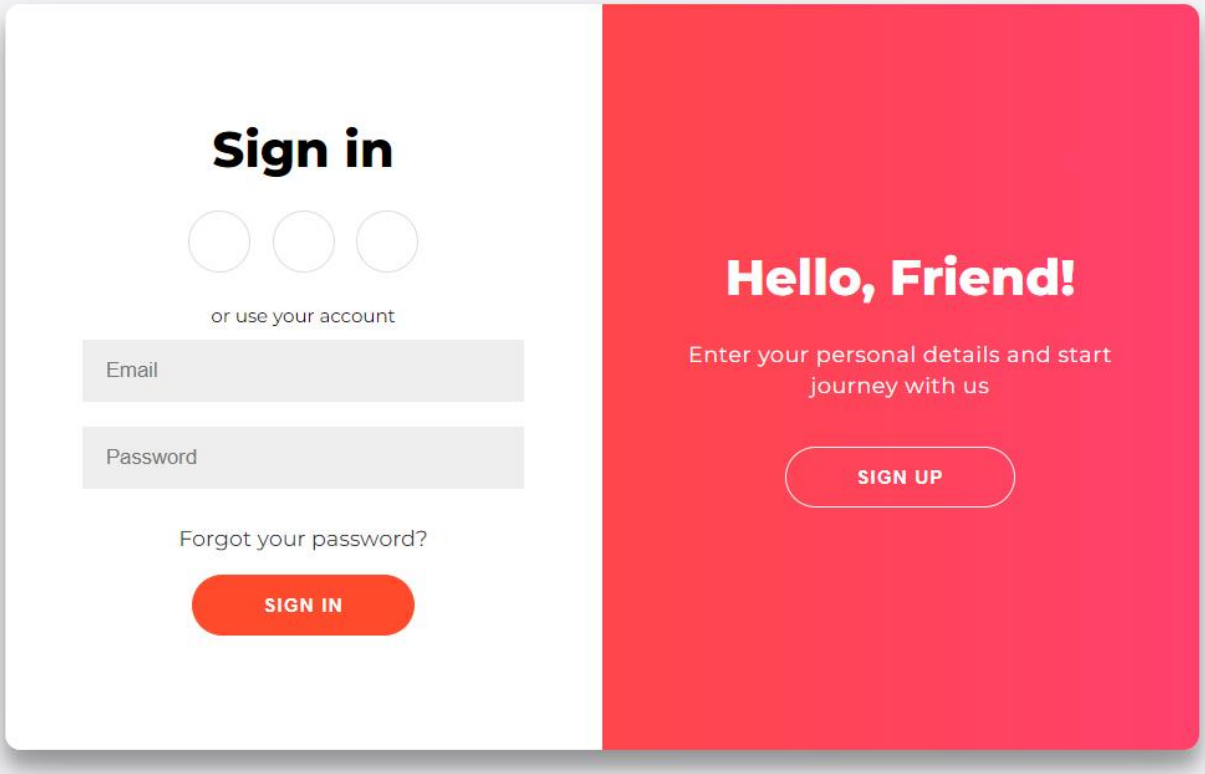
```
@import url('https://fonts.googleapis.com/css?family=Montserrat:400,800');

* {
    box-sizing: border-box;
}

body {
    background: #f6f5f7;
    display: flex;
    justify-content: center;
    align-items: center;
    flex-direction: column;
    font-family: 'Montserrat', sans-serif;
    height: 100vh;
    margin: -20px 0 50px;
}

h1 {
    font-weight: bold;
    margin: 0;
```

```
}

h2 {
    text-align: center;
}

p {
    font-size: 14px;
    font-weight: 100;
    line-height: 20px;
    letter-spacing: 0.5px;
    margin: 20px 0 30px;
}

span {
    font-size: 12px;
}

a {
    color: #333;
    font-size: 14px;
    text-decoration: none;
    margin: 15px 0;
}

button {
    border-radius: 20px;
    border: 1px solid #FF4B2B;
    background-color: #FF4B2B;
    color: #FFFFFF;
    font-size: 12px;
    font-weight: bold;
    padding: 12px 45px;
    letter-spacing: 1px;
    text-transform: uppercase;
    transition: transform 80ms ease-in;
}

button:active {
    transform: scale(0.95);
}

button:focus {
    outline: none;
}

button.ghost {
    background-color: transparent;
    border-color: #FFFFFF;
}

form {
    background-color: #FFFFFF;
    display: flex;
    align-items: center;
    justify-content: center;
    flex-direction: column;
```

```
    padding: 0 50px;
    height: 100%;
    text-align: center;
}

input {
    background-color: #eee;
    border: none;
    padding: 12px 15px;
    margin: 8px 0;
    width: 100%;
}

.container {
    background-color: #fff;
    border-radius: 10px;
    box-shadow: 0 14px 28px rgba(0,0,0,0.25),
            0 10px 10px rgba(0,0,0,0.22);
    position: relative;
    overflow: hidden;
    width: 768px;
    max-width: 100%;
    min-height: 480px;
}

.form-container {
    position: absolute;
    top: 0;
    height: 100%;
    transition: all 0.6s ease-in-out;
}

.sign-in-container {
    left: 0;
    width: 50%;
    z-index: 2;
}

.container.right-panel-active .sign-in-container {
    transform: translateX(100%);
}

.sign-up-container {
    left: 0;
    width: 50%;
    opacity: 0;
    z-index: 1;
}

.container.right-panel-active .sign-up-container {
    transform: translateX(100%);
    opacity: 1;
    z-index: 5;
    animation: show 0.6s;
}

@keyframes show {
```

```
    0%, 49.99% {
        opacity: 0;
        z-index: 1;
    }

    50%, 100% {
        opacity: 1;
        z-index: 5;
    }
}

.overlay-container {
    position: absolute;
    top: 0;
    left: 50%;
    width: 50%;
    height: 100%;
    overflow: hidden;
    transition: transform 0.6s ease-in-out;
    z-index: 100;
}

.container.right-panel-active .overlay-container{
    transform: translateX(-100%);
}

.overlay {
    background: #FF416C;
    background: -webkit-linear-gradient(to right, #FF4B2B, #FF416C);
    background: linear-gradient(to right, #FF4B2B, #FF416C);
    background-repeat: no-repeat;
    background-size: cover;
    background-position: 0 0;
    color: #FFFFFF;
    position: relative;
    left: -100%;
    height: 100%;
    width: 200%;
    transform: translateX(0);
    transition: transform 0.6s ease-in-out;
}

.container.right-panel-active .overlay {
    transform: translateX(50%);
}

.overlay-panel {
    position: absolute;
    display: flex;
    align-items: center;
    justify-content: center;
    flex-direction: column;
    padding: 0 40px;
    text-align: center;
    top: 0;
    height: 100%;
    width: 50%;
```

```
    transform: translateX(0);
    transition: transform 0.6s ease-in-out;
}

.overlay-left {
    transform: translateX(-20%);
}

.container.right-panel-active .overlay-left {
    transform: translateX(0);
}

.overlay-right {
    right: 0;
    transform: translateX(0);
}

.container.right-panel-active .overlay-right {
    transform: translateX(20%);
}

.social-container {
    margin: 20px 0;
}

.social-container a {
    border: 1px solid #DDDDDD;
    border-radius: 50%;
    display: inline-flex;
    justify-content: center;
    align-items: center;
    margin: 0 5px;
    height: 40px;
    width: 40px;
}
```



Post Lab Objective with Ans (Min 5):

- **What is the Box model in CSS?**

  The box model in CSS is a fundamental concept that defines how elements are rendered on a webpage. It consists of four parts: content, padding, border, and margin. The content area holds the actual content of the element, padding provides space between the content and the border, border outlines the content and padding, and margin separates the element from other elements on the page.

- **What are the advantages of using CSS?**

The advantages of using CSS include:

1. Separation of concerns: CSS separates the style and presentation of a webpage from its HTML structure, making it easier to maintain and update.
2. Consistency: CSS allows for consistent styling across a website, ensuring a uniform look and feel.
3. Flexibility: CSS offers various layout options and styling techniques, allowing designers to create diverse and visually appealing designs.
4. Efficiency: CSS enables smaller file sizes compared to inline styling or using HTML

attributes, leading to faster page loading times.
5. Accessibility: CSS supports accessibility features, making it easier to create websites that are usable by individuals with disabilities.
6. Device compatibility: CSS allows for responsive design, enabling websites to adapt to different screen sizes and devices seamlessly.
7. SEO benefits: Properly structured CSS can improve search engine optimization by enhancing page readability and structure.

- **What are the limitations of CSS?**

The limitations of CSS include:

1. Browser inconsistencies: Different browsers may interpret CSS rules differently, leading to inconsistencies in how a webpage is displayed.
2. Limited layout capabilities: CSS has limitations in complex layout designs, particularly in comparison to more advanced layout systems like those offered by JavaScript frameworks.
3. Lack of dynamic capabilities: CSS is primarily a static styling language and lacks dynamic functionality, such as variables, loops, and conditional statements.
4. Accessibility challenges: While CSS can improve accessibility, it also poses challenges for accessibility when used improperly, such as with overly complex layouts or lack of semantic markup.
5. Performance overhead: Excessive use of CSS or inefficient stylesheets can negatively impact page load times and overall performance.
6. Learning curve: CSS can have a steep learning curve, especially for beginners, due to its specificity, inheritance, and various layout techniques.
7. Limited browser support for newer features: Some newer CSS features may not be fully supported across all browsers, requiring fallbacks or polyfills for compatibility.

- **What are the different types of Selectors in CSS?**

The different types of selectors in CSS include:

1. Element selectors: Select elements based on their HTML tag name.
2. Class selectors: Select elements based on their class attribute.
3. ID selectors: Select a single element based on its unique ID attribute.
4. Universal selectors: Select all elements on a webpage.
5. Attribute selectors: Select elements based on their attribute values.
6. Pseudo-classes: Select elements based on their state or position in the document.
7. Pseudo-elements: Select and style parts of an element, such as its first line or first letter.