CSS/JS

1. Create a webpage that displays the div element with the following attributes.

• Font name: Verdana

• Font size: 20 pixels

• Color: red

• Align: center

Note: Use Inline style sheet

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Styled Div Example</title>

</head>

<body>

    <div *style*="font-family: Verdana; font-size: 20px; color: red; text-align: center; ">

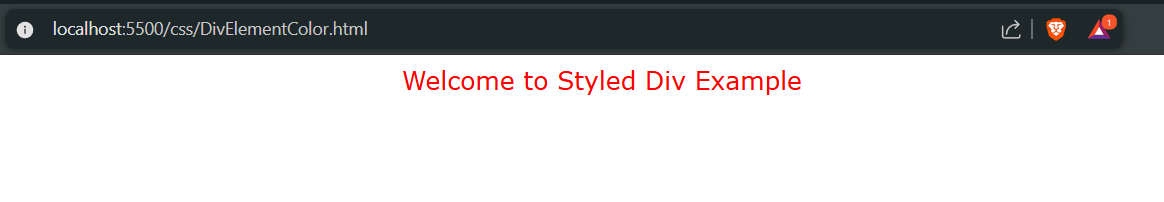
        Welcome to Styled Div Example

    </div>

</body>

</html>

**Output:**



1. Create a webpage that displays the div element with the following text properties.

• Text color: Magenta

• Text alignment: right

• Text decoration: underline

• Text transform: uppercase

• Text indent: 60 pixels

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Styled Div Example</title>

</head>

<body>

    <div *style*="color: magenta; text-align: right; text-decoration: underline; text-transform: uppercase; text-indent: 60px; font-size: 30px;">

        Welcome to Styled Div Example

    </div>

</body>

</html>

**Output:**



1. Using CSS, create a web page that displays text “Welcome to CSS World” with an image as its border,

a. Image repeats itself to fit the contents

b. Image should stretch to fit the contents

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Image Border Example</title>

    <style>

*.img-border-repeat* {

            border: 20px solid transparent;

            padding: 20px 40px;

            font-size: 32px;

            text-align: center;

            margin: 40px auto;

            width: fit-content;

            border-image: url('https://www.w3schools.com/cssref/border.png') 30 repeat;

        }

*.img-border-stretch* {

            border: 20px solid transparent;

            padding: 20px 40px;

            font-size: 32px;

            text-align: center;

            margin: 40px auto;

            width: fit-content;

            border-image: url('https://www.w3schools.com/cssref/border.png') 30 stretch;

        }

    </style>

</head>

<body>

    <div *class*="img-border-repeat">

        Welcome to CSS World<br>

        <span *style*="font-size:14px; color:#888;">(Image border repeats)</span>

    </div>

    <div *class*="img-border-stretch">

        Welcome to CSS World<br>

        <span *style*="font-size:14px; color:#888;">(Image border stretches)</span>

    </div>

</body>

</html>

**Output:**



1. Create a webpage that displays the hyperlink with the following background color properties.

• Set the background color as %FF0000 for unvisited link

• Set the background color as %FFFF00 for visited link

• Set the background color as %FF00FF for mouse over link

• Set the background color as %0000FF for selected link

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Styled Hyperlink Background Colors</title>

    <style>

        a*:link* {

            background-color: #FF0000;

            color: white;

            padding: 8px 16px;

            text-decoration: none;

        }

        a*:visited* {

            background-color: #FFFF00;

            color: black;

        }

        a*:hover* {

            background-color: #FF00FF;

            color: white;

        }

        a*:active* {

            background-color: #0000FF;

            color: white;

        }

    </style>

</head>

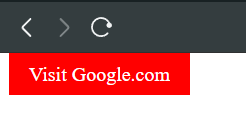
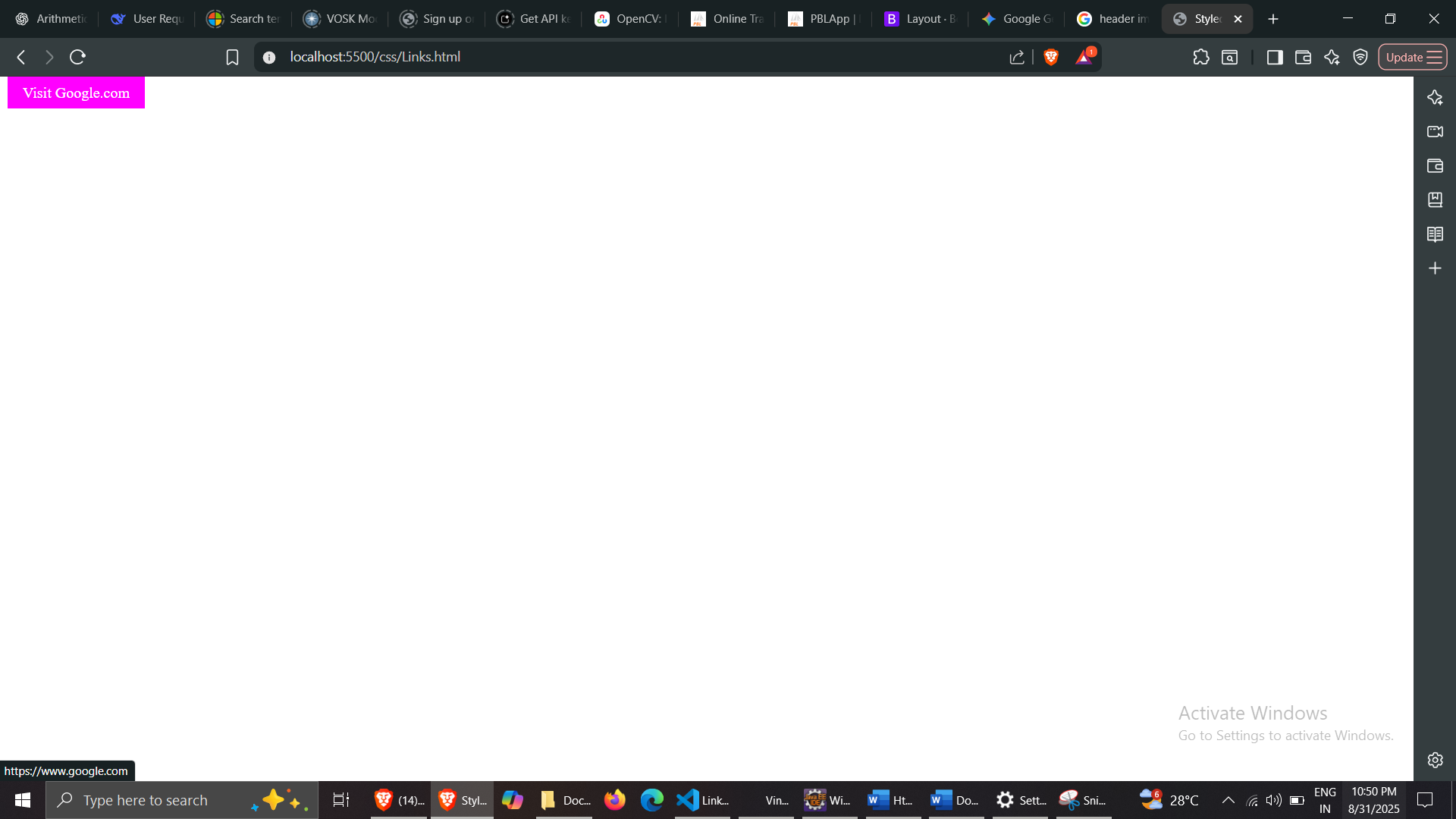
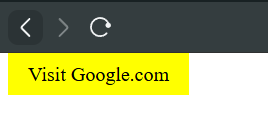
<body>

    <a *href*="https://www.google.com" *target*="\_blank">Visit Google.com</a>

</body>

</html>

**Output:**



1. Create a webpage that displays a list as an unordered list with the following properties.

• List-style-type: square;

• Text align: left

• Color: red

• Text transform: uppercase

• Text decoration: underline

• Font name: Arial

• Font size: 25pixels

• Font weight: bold

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Styled Unordered List</title>

    <style>

        ul*.custom-list* {

            list-style-type: square;

            text-align: left;

            color: red;

            text-transform: uppercase;

            text-decoration: underline;

            font-family: Arial, sans-serif;

            font-size: 25px;

            font-weight: bold;

            padding-left: 40px;

        }

    </style>

</head>

<body>

    <ul *class*="custom-list">

        <li>apple</li>

        <li>banana</li>

        <li>orange</li>

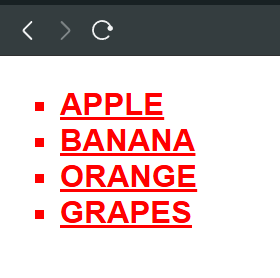
        <li>grapes</li>

    </ul>

</body>

</html>

**Output:**



1. Create a webpage that displays four paragraphs with the following padding properties.

• Set the left padding as 3cm for first paragraph

• Set the right padding as 50% for second paragraph

• Set the bottom padding as 25% for third paragraph

• Set the top padding as 25% for fourth paragraph

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Paragraph Padding Example</title>

    <style>

*.pad-left* {

            padding-left: 3cm;

            background: #f9eaea;

        }

*.pad-right* {

            padding-right: 50%;

            background: #eaf9ea;

        }

*.pad-bottom* {

            padding-bottom: 25%;

            background: #eaeaf9;

        }

*.pad-top* {

            padding-top: 25%;

            background: #f9f9ea;

        }

        p {

            margin: 20px 0;

            border: 1px solid #ccc;

        }

    </style>

</head>

<body>

    <p *class*="pad-left">This paragraph has a left padding of 3cm.</p>

    <p *class*="pad-right">This paragraph has a right padding of 50%.</p>

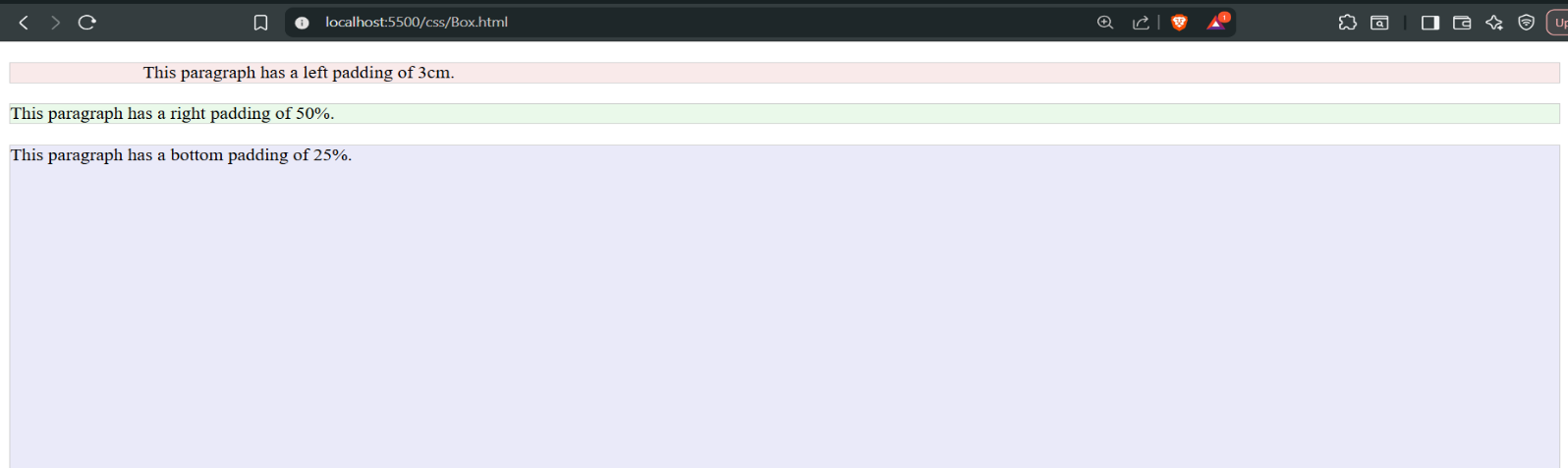
    <p *class*="pad-bottom">This paragraph has a bottom padding of 25%.</p>

    <p *class*="pad-top">This paragraph has a top padding of 25%.</p>

</body>

</html>

**Output:**





1. Create a webpage that displays a white text with black shadow.

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>White Text with Black Shadow</title>

    <style>

        body {

            text-align: center;

            padding-top: 100px;

        }

*.shadow-text* {

            color: #fff;

            font-size: 48px;

            font-family: Arial, sans-serif;

            text-shadow: 3px 3px 8px #000;

        }

    </style>

</head>

<body>

    <div *class*="shadow-text">

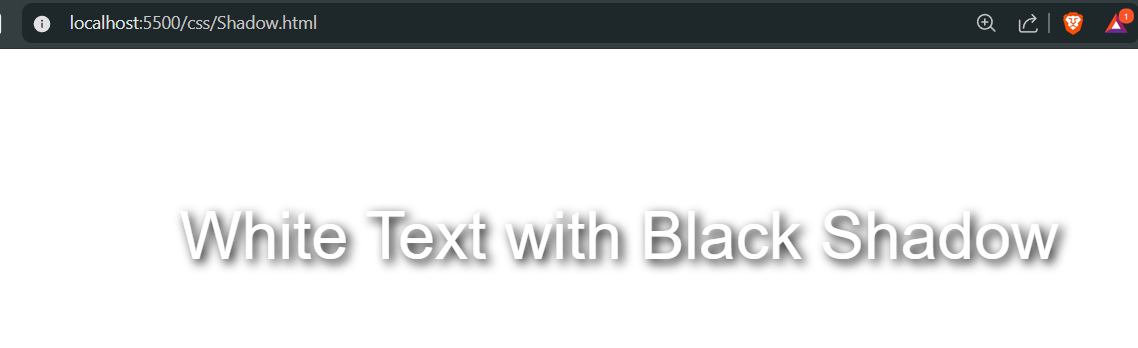
        White Text with Black Shadow

    </div>

</body>

</html>

**Output:**



1. Create a webpage that displays the div element with the following properties.

• Skews the

element 20 degrees along the X-axis, and 10 degrees along the Y-axis

• Set the background color as yellow

• Set the border as 1pixels solid black

• Set the width as 200pixels

• Set the height as 100 pixels

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Skewed Div Example</title>

    <style>

*.skewed-div* {

            width: 200px;

            height: 100px;

            background: yellow;

            border: 1px solid black;

            transform: skew(20deg, 10deg);

            display: flex;

            align-items: center;

            justify-content: center;

            font-family: Arial, sans-serif;

            margin: 60px auto;

        }

    </style>

</head>

<body>

    <div *class*="skewed-div">

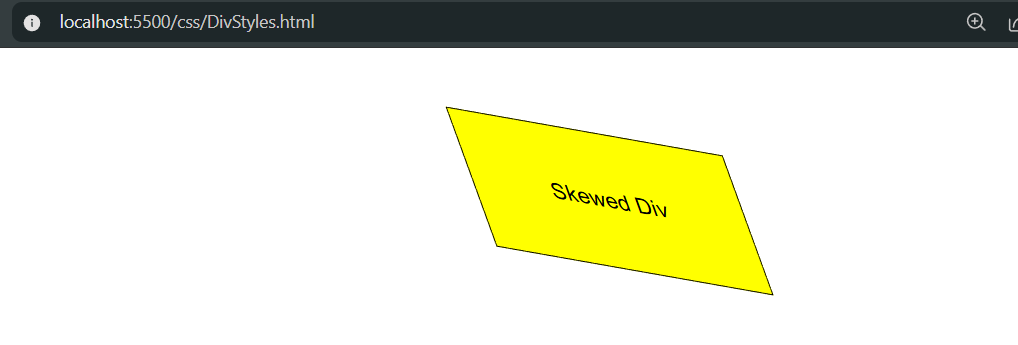
        Skewed Div

    </div>

</body>

</html>

**Output:**



1. Write a JavaScript function that checks whether a passed string is palindrome or not?

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Palindrome Checker</title>

    <script>

        function isPalindrome(str) {

            var reversed = str.split('').reverse().join('');

            return str === reversed;

        }

        function checkPalindrome() {

            var input = document.getElementById('inputStr').*value*;

            var result = isPalindrome(input)

                ? "It's a palindrome!"

                : "It's not a palindrome.";

            document.getElementById('result').*textContent* = result;

        }

    </script>

</head>

<body>

    <h2>Palindrome Checker</h2>

    <input *type*="text" *id*="inputStr" *placeholder*="Enter text">

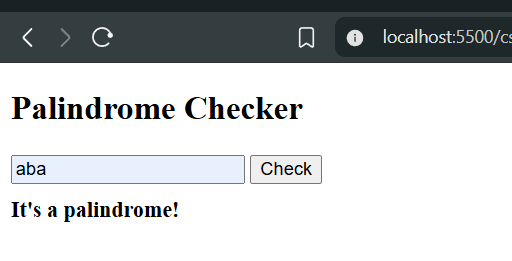
    <button *onclick*="checkPalindrome()">Check</button>

    <div *id*="result" *style*="margin-top:10px; font-weight:bold;"></div>

</body>

</html>

**Output:**



1. Create an HTML page with two textboxes and a Calculate button:

When user enters the numbers and clicks the Calculate button, it has to alert the sum.

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Calculate Sum</title>

    <script>

        function calculateSum() {

            var num1 = parseFloat(document.getElementById('num1').*value*) || 0;

            var num2 = parseFloat(document.getElementById('num2').*value*) || 0;

            alert("Sum: " + (num1 + num2));

        }

    </script>

</head>

<body>

    <h2>Sum Calculator</h2>

    <input *type*="text" *id*="num1" *placeholder*="Enter first number">

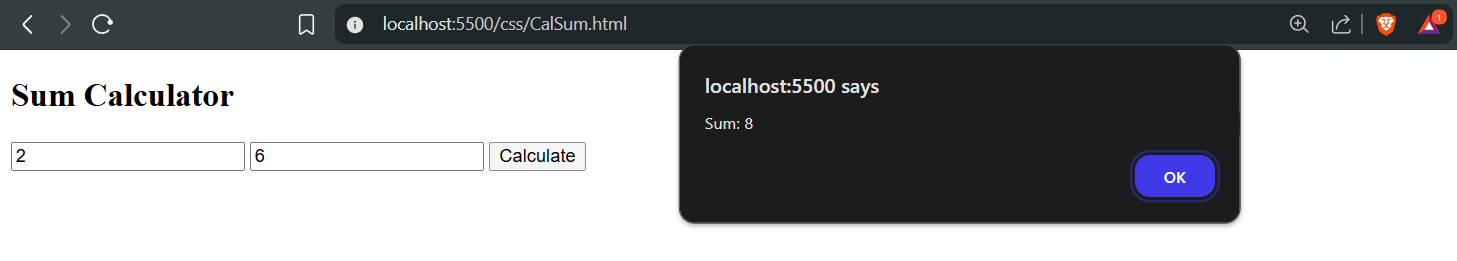
    <input *type*="text" *id*="num2" *placeholder*="Enter second number">

    <button *onclick*="calculateSum()">Calculate</button>

</body>

</html>

**Output:**



1. Create a HTML page with 4 hyperlinks named white, red, blue and pink.

The background colour of the HTML page will change based on the link which is clicked.

Write a javascript function to implement it.

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Background Color Changer</title>

    <script>

        function changeBgColor(color) {

            document.*body*.*style*.*backgroundColor* = color;}

    </script>

</head>

<body>

    <h2>Click a color to change background</h2>

    <a *href*="#" *onclick*="changeBgColor('white')">White</a> |

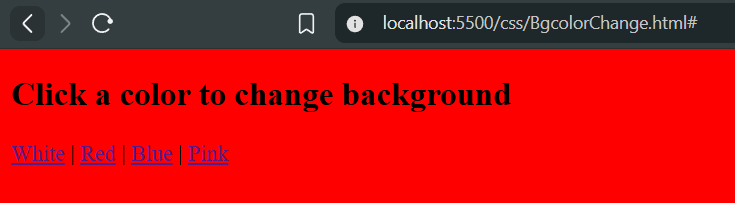
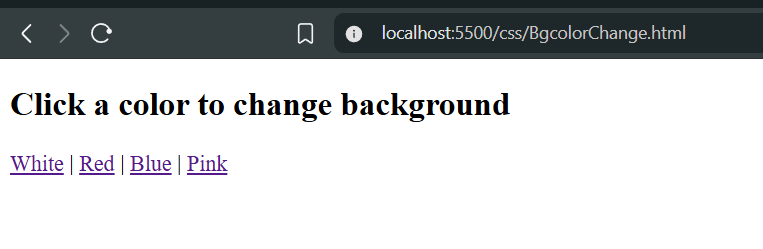
    <a *href*="#" *onclick*="changeBgColor('red')">Red</a> |

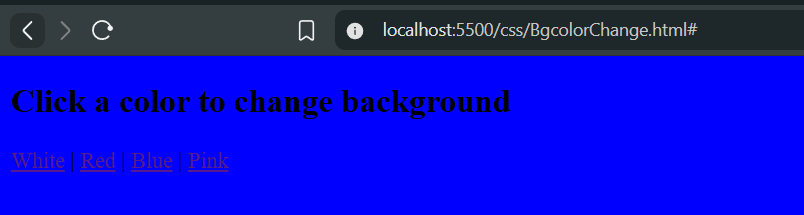
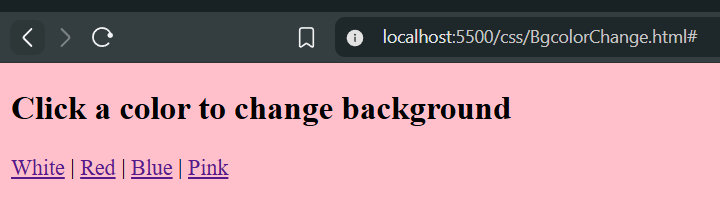
    <a *href*="#" *onclick*="changeBgColor('blue')">Blue</a> |

    <a *href*="#" *onclick*="changeBgColor('pink')">Pink</a>

</body>

</html>

**Output:**



1. Create program to display current time in a textbox (HH:MM:SS) such that the value in the time textbox is dynamic (Time should get updated every second) and not static.[Hint: Use setTimeOut function]

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Dynamic Time Display</title>

    <script>

        function updateTime() {

            var now = new Date();

            var hours = String(now.getHours()).padStart(2, '0');

            var minutes = String(now.getMinutes()).padStart(2, '0');

            var seconds = String(now.getSeconds()).padStart(2, '0');

            var currentTime = hours + ":" + minutes + ":" + seconds;

            document.getElementById('timeBox').*value* = currentTime;

            setTimeout(updateTime, 1000);

        }

        window.*onload* = updateTime;

    </script>

</head>

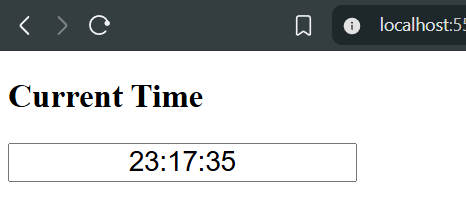
<body>

    <h2>Current Time</h2>

    <input *type*="text" *id*="timeBox" *readonly* *style*="font-size:20px; text-align:center;">

</body>

</html>

**Output:**

1. Write validation functions and modify the onSubmit event Handler in the form code to validate thefollowing form fields:

1. Member number

• Must be entered

• Must be a number

2. Password

• Must be entered

• Must be longer than 4 characters

If an error occurs, use an alert box to display an error message for the first field in error and place the cursor in the first field in error.

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Form Validation Example</title>

    <script>

        function validateForm(form) {

            var memberNo = form.*memberNo*.*value*.trim();

            if (memberNo === "") {

                alert("Member number must be entered.");

                form.*memberNo*.focus();

                return false;

            }

            if (isNaN(memberNo)) {

                alert("Member number must be a number.");

                form.*memberNo*.focus();

                return false;

            }

            var password = form.*password*.*value*;

            if (password === "") {

                alert("Password must be entered.");

                form.*password*.focus();

                return false;

            }

            if (password.length <= 4) {

                alert("Password must be longer than 4 characters.");

                form.*password*.focus();

                return false;

            }

            return true;

        }

    </script>

</head>

<body>

    <h2>Member Login</h2>

    <form *onsubmit*="return validateForm(*this*);">

        <label *for*="memberNo">Member Number:</label>

        <input *type*="text" *id*="memberNo" *name*="memberNo"><br><br>

        <label *for*="password">Password:</label>

        <input *type*="password" *id*="password" *name*="password"><br><br>

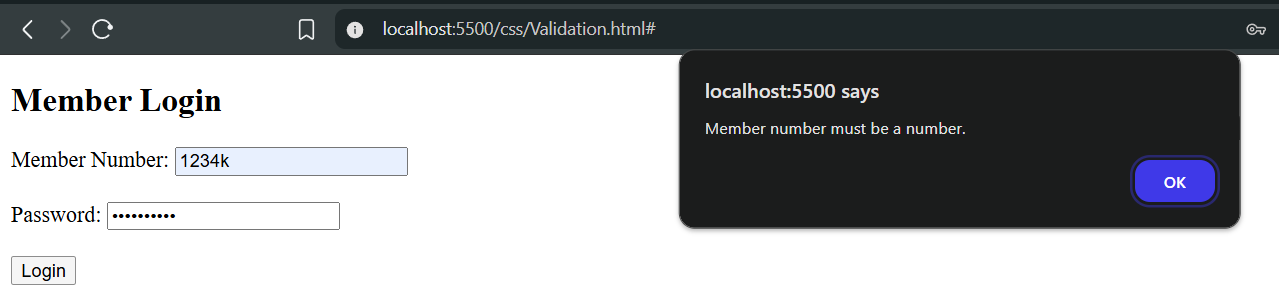
        <input *type*="submit" *value*="Login">

    </form>

</body>

</html>

**Output:**



1. Write a validation function to check a password between 6 to 20 characters which contain at least one numeric digit, one uppercase and one lowercase letter

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Password Validation</title>

    <script>

        function validatePassword(password) {

*// Password between 6 to 20 characters, at least one digit, one uppercase, one lowercase*

            var regex = /^(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{6,20}$/;

            return regex.test(password);

        }

        function checkPassword() {

            var pwd = document.getElementById('password').*value*;

            if (validatePassword(pwd)) {

                alert("Password is valid!");

            } else {

                alert("Password must be 6-20 characters and include at least one numeric digit, one uppercase and one lowercase letter.");

            }

        }

    </script>

</head>

<body>

    <h2>Password Validation</h2>

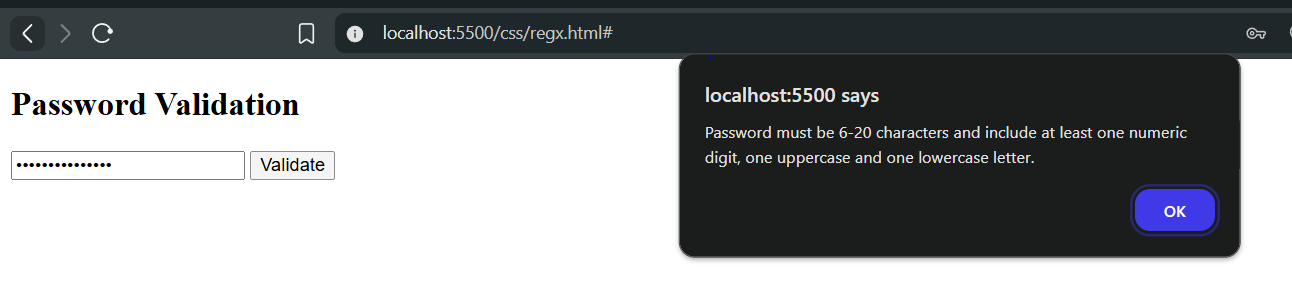
    <input *type*="password" *id*="password" *placeholder*="Enter password">

    <button *onclick*="checkPassword()">Validate</button>

</body>

</html>

**Output:**



1. Create a OldLook.html web page in the current project with the following content

Tags - Contents

H1 – Welcome to my page

P – this is a trial for bootstrap

Table

TR

TH – col1

TH – col2

TR

TD – data1

TD – data2

TR

TD – data3

TD – data4 use bootstrap

**Source Code:**

<!DOCTYPE *html*>

<html>

<head>

    <title>Welcome to my page</title>

*<!-- Bootstrap CSS CDN -->*

    <link *href*="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" *rel*="stylesheet">

</head>

<body *class*="background-color: #f8f9fa;">

    <div *class*="container mt-5">

        <h1 *class*="mb-4">Welcome to my page</h1>

        <p *class*="mb-4">this is a trial for bootstrap</p>

        <table *class*="table table-bordered">

            <tr>

                <th>col1</th>

                <th>col2</th>

            </tr>

            <tr>

                <td>data1</td>

                <td>data2</td>

            </tr>

            <tr>

                <td>data3</td>

                <td>data4</td>

            </tr>

        </table>

    </div>

</body>

</html>

**Output:**

