|  |  |
| --- | --- |
| **Ex no:01** | **Create HTML Page with Lists, Tables, and Hyperlinks** |
| **Date:** |

**AIM:**  
To design a basic website using HTML to demonstrate text formatting and image insertion.

**ALGORITHMM:**

**Step 1:** Start with <!DOCTYPE html>, <html>, <head>, and <body> tags.

**Step 2**: Inside <head>, add a <title> for the webpage.

**Step 3:** Use ordered (<ol>) and unordered (<ul>) lists to display items.

**Step 4:** Add table using <table>, with <tr> for rows and <td> for cells.

**Step 5:** Use <th> to define table headers.

**Step 6:** Insert hyperlinks using <a href="URL">Link Text</a>.

**Step 7:** Apply basic formatting for neat display.

**Step 8:** Close all opened tags properly.

**Step 9:** Save the file with .html extension.

**Step 10:** Open in a browser to check lists, table, and links.

**SOURCE CODE:**

<!DOCTYPE html>

<html>

<head>

<title>Text Formatting and Images</title>

<style>

body {

font-family: Verdana;

background-color: #e8f0fe;

padding: 20px;

}

</style>

</head>

<body>

<h1>Welcome to My Website</h1>

<p>This is a <b>bold</b> paragraph.</p>

<p>This is an <i>italicized</i> word.</p>

<p>This is an <u>underlined</u> sentence.</p>

<p><strong>Combining</strong> <em>multiple</em> <u>formats</u>.</p>

<h2>About Us</h2>

<p>We offer tutorials in HTML, CSS, JavaScript, and more.</p>

<h2>Our Logo</h2>

<img src="logo.png" alt="Logo" width="200" height="100">

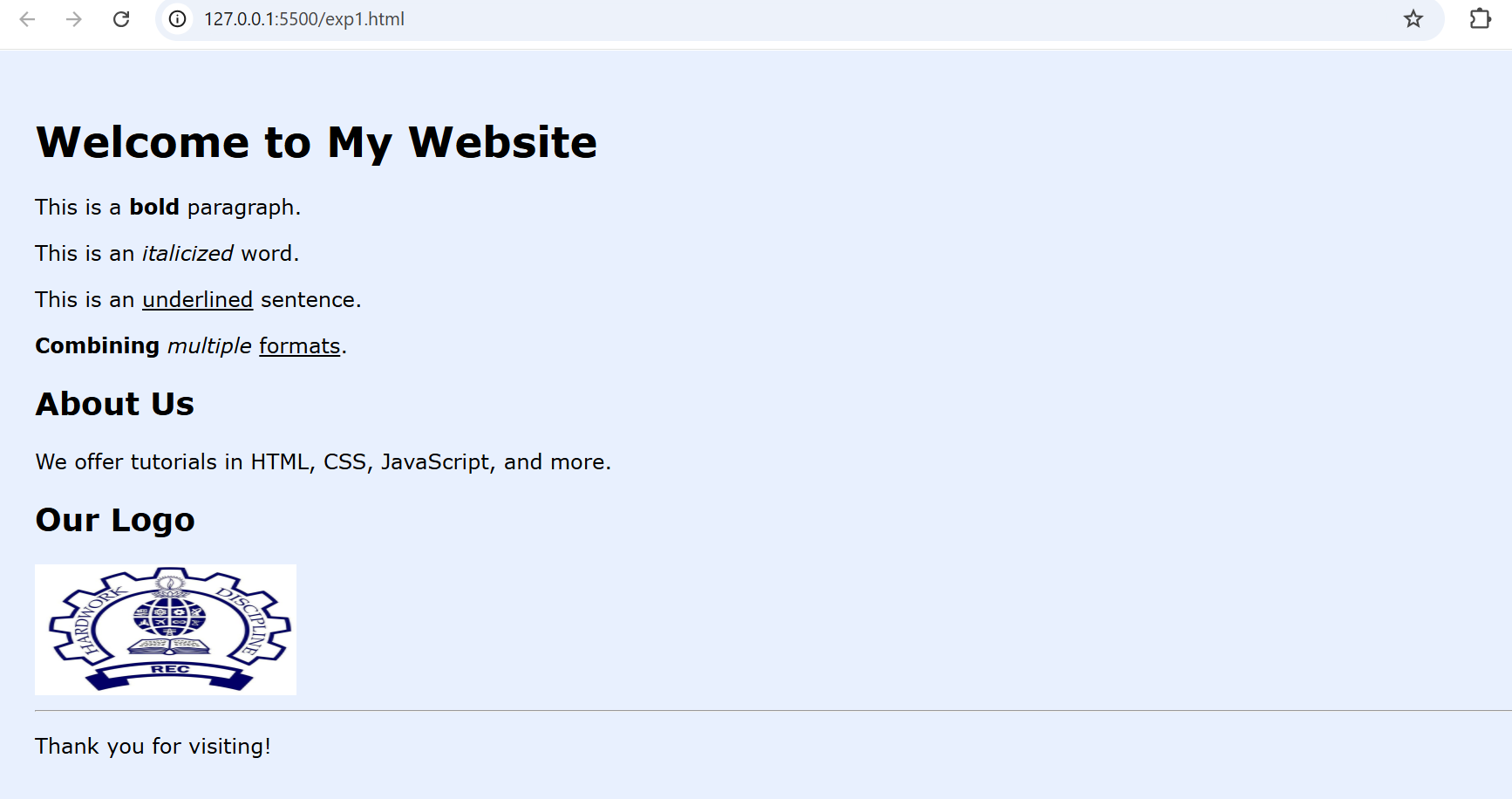
<hr>

<p>Thank you for visiting!</p>

</body>

</html>

**OUTPUT:**



**RESULT:**

We have designed a basic website using HTML to demonstrate text formatting and image insertion.

|  |  |
| --- | --- |
| **Ex no:02** | **Basic Website using HTML (Text Formatting + Image)** |
| **Date:** |

**AIM:**  
To write a HTML program for creation of forms, links, and tables.

**ALGORITHM:**

**Step 1:** Start with <!DOCTYPE html>, <html>, <head>, and <body> tags.

**Step 2:** Inside <head>, add the <title> for the page.

**Step 3:** Start the <body> section for the main content.

**Step 4:** Add headings (<h1> to <h6>) to structure the page.

**Step 5:** Insert paragraphs (<p>) for regular text.

**Step 6:** Apply formatting tags like <b>, <i>, and <u> for styling.

**Step 7:** Use <hr> to add horizontal lines and <br> for line breaks.

**Step 8:** Insert an image with the <img> tag and set src and alt.

**Step 9:** Close the </body> and </html> tags.

**Step 10:** Save the file as .html and view it in a browser.

**SOURCE CODE:**

<!DOCTYPE html>

<html>

<head>

  <title>Forms, Links, and Tables Example</title>

  <style>

    body {

      font-family: Arial, sans-serif;

      background-color: #f9f9f9;

      padding: 20px;

    }

    form, table {

      margin-bottom: 20px;

    }

    table {

      border-collapse: collapse;

      width: 60%;

    }

    th, td {

      border: 1px solid #333;

      padding: 8px;

      text-align: center;

    }

  </style>

</head>

<body>

  <h1>Welcome to My Web Page</h1>

  <hr>

  <p>This page demonstrates the creation of <b>forms</b>, <i>links</i>, and <u>tables</u> using HTML.</p>

  <h2>User Information Form</h2>

  <form action="#">

    <label for="name">Name:</label><br>

    <input type="text" id="name" name="name" required><br><br>

    <label for="email">Email:</label><br>

    <input type="email" id="email" name="email" required><br><br>

    <input type="submit" value="Submit">

  </form>

  <hr>

  <h2>Useful Link</h2>

  <p>Visit <a href="https://www.w3schools.com" target="\_blank"><b>W3Schools</b></a> for learning web development!</p>

  <hr>

  <h2>Sample Student Table</h2>

  <table>

    <tr>

      <th>Student Name</th>

      <th>Roll Number</th>

    </tr>

    <tr>

      <td>Alice</td>

      <td>101</td>

    </tr>

    <tr>

      <td>Bob</td>

      <td>102</td>

    </tr>

    <tr>

      <td>Charlie</td>

      <td>103</td>

    </tr>

  </table>

  <hr>

  <h2>Sample Image</h2>

  <img src="logo.png" alt="Sample Image" width="300" height="200">

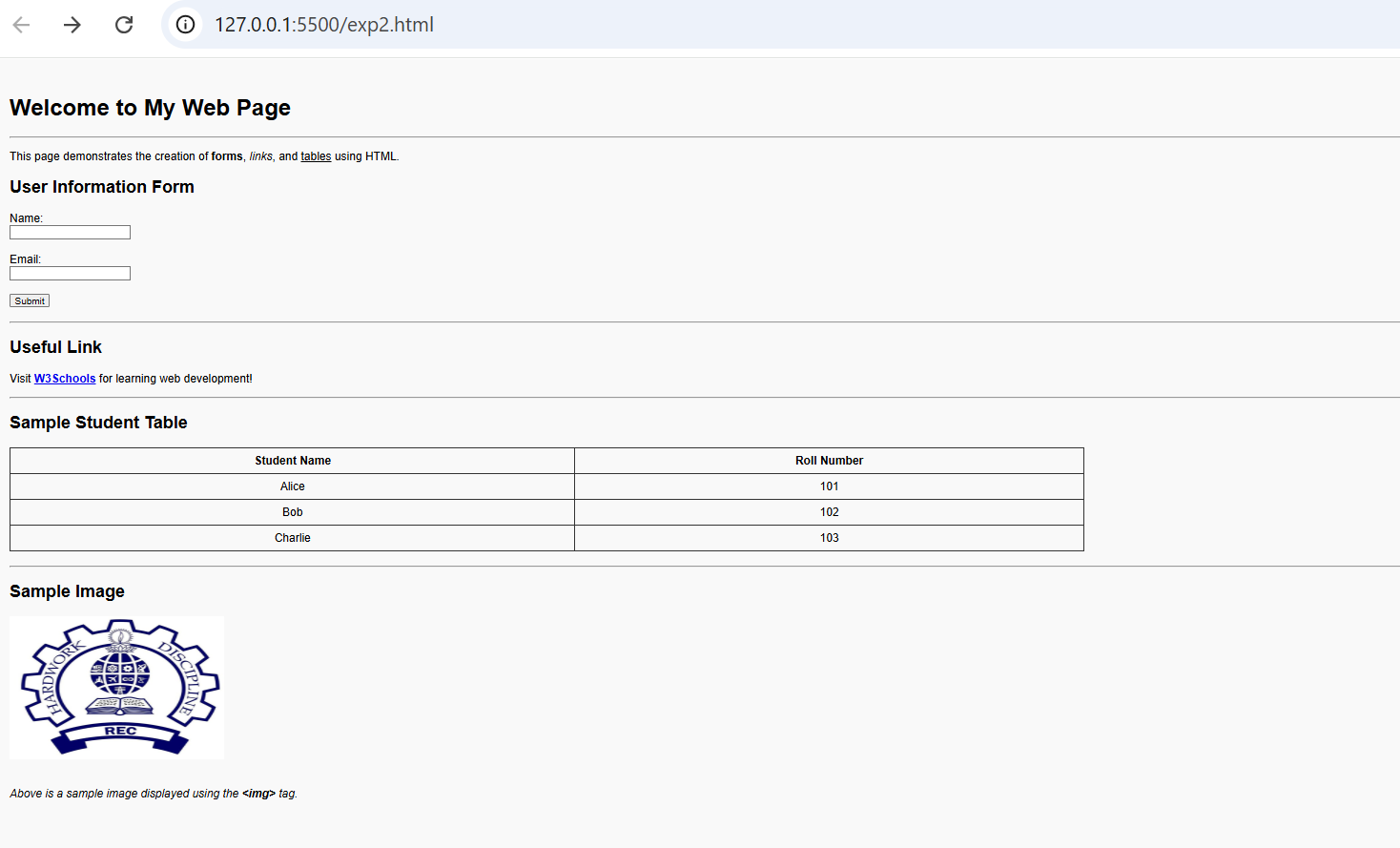
  <br><br>

  <p><i>Above is a sample image displayed using the <b>&lt;img&gt;</b> tag.</i></p>

</body>

</html>

**OUTPUT:**



**RESULT:**

Thus we have written a HTML program for creation of forms, links, and tables.

|  |  |
| --- | --- |
| **Ex no:03** | **Create a web page with HTML5 with image map, hotspot and information when hotspot is clicked** |
| **Date:** |

**AIM:**

create an image map in a webpage, identify hotspots, and display related information when clicked.

**ALGORITHM:**

**Step 1:** Create a basic HTML page with a <form> inside <body>.

**Step 2:** Add input fields for name, email, password, etc., using <input> tags.

**Step 3:** Write a JavaScript function to check if all fields are filled.

**Step 4:** Use if-else conditions to validate inputs like email format.

**Step 5:** Display alerts or error messages if validation fails.

**Step 6:** Call the JavaScript function when the form is submitted using onsubmit.

**Step 7:** Prevent form submission if any field is incorrect.

**Step 8:** Save the HTML file with embedded or linked JavaScript.

**Step 9:** Open the file in a browser and test by submitting the form.

**Step 10:** Debug and fix errors if the form validation doesn't work properly.

**SOURCE CODE:**

<!DOCTYPE html>

<html>

<head>

<title>Image Map Example</title>

<style>

body {

font-family: Arial;

background-color: #f0f8ff;

padding: 20px;

}

</style>

</head>

<body>

<h2>India Map - Image Map</h2>

<p>Click on a region to learn more about it.</p>

<!-- Replace with your actual image and usemap -->

<img src="india-map.jpg" alt="India Map" usemap="#indiastates" width="600" height="400">

<map name="indiastates">

<area shape="rect" coords="120,90,180,150" alt="Delhi" href="https://en.wikipedia.org/wiki/Delhi" target="\_blank">

<area shape="circle" coords="300,200,40" alt="Mumbai" href="https://en.wikipedia.org/wiki/Mumbai" target="\_blank">

<area shape="poly" coords="400,300,420,320,410,350,390,330" alt="Chennai" href="https://en.wikipedia.org/wiki/Chennai" target="\_blank">

</map>

</body>

</html>

**OUTPUT:**



**RESULT:**

Thus we have created an image map in a webpage, identify hotspots, and display related information when clicked.

|  |  |
| --- | --- |
| **Ex no:04** | **Create a webpage with all types of CSS** |
| **Date:** |

**AIM:**  
To create a structured web page using HTML5 semantic elements.

**ALGORITHM:**

**Step 1:** Create an HTML structure with <!DOCTYPE html>, <html>, <head>, and <body> tags.

**Step 2:** Inside <head>, add a <title> for the page.

**Step 3:** Add a <style> tag inside <head> or link an external CSS file.

**Step 4:** Define CSS rules for body, headings, paragraphs, etc.

**Step 5:** Set background color, font size, font family, and text color using CSS.

**Step 6:** Apply margins, padding, and borders to elements.

**Step 7:** Style buttons, images, and links using CSS properties.

**Step 8:** Save the HTML and CSS files properly.

**Step 9:** Open the HTML file in a browser to check the styling.

**Step 10:** Make necessary adjustments if the page needs improvements.

**SOURCE CODE:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>HTML5 Semantic Page</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #eef2f3;

margin: 0;

}

header, nav, section, article, aside, footer {

padding: 15px;

margin: 10px;

border-radius: 8px;

}

header {

background-color: #4CAF50;

color: white;

text-align: center;

}

nav {

background-color: #ddd;

}

section {

display: flex;

}

article {

flex: 2;

background-color: #fff;

}

aside {

flex: 1;

background-color: #f4f4f4;

}

footer {

background-color: #333;

color: white;

text-align: center;

}

</style>

</head>

<body>

<header>

<h1>My HTML5 Webpage</h1>

</header>

<nav>

<a href="#">Home</a> |

<a href="#">About</a> |

<a href="#">Contact</a>

</nav>

<section>

<article>

<h2>Main Article</h2>

<p>This section represents the main content area.</p>

</article>

<aside>

<h3>Side Info</h3>

<p>This is some additional information.</p>

</aside>

</section>

<footer>

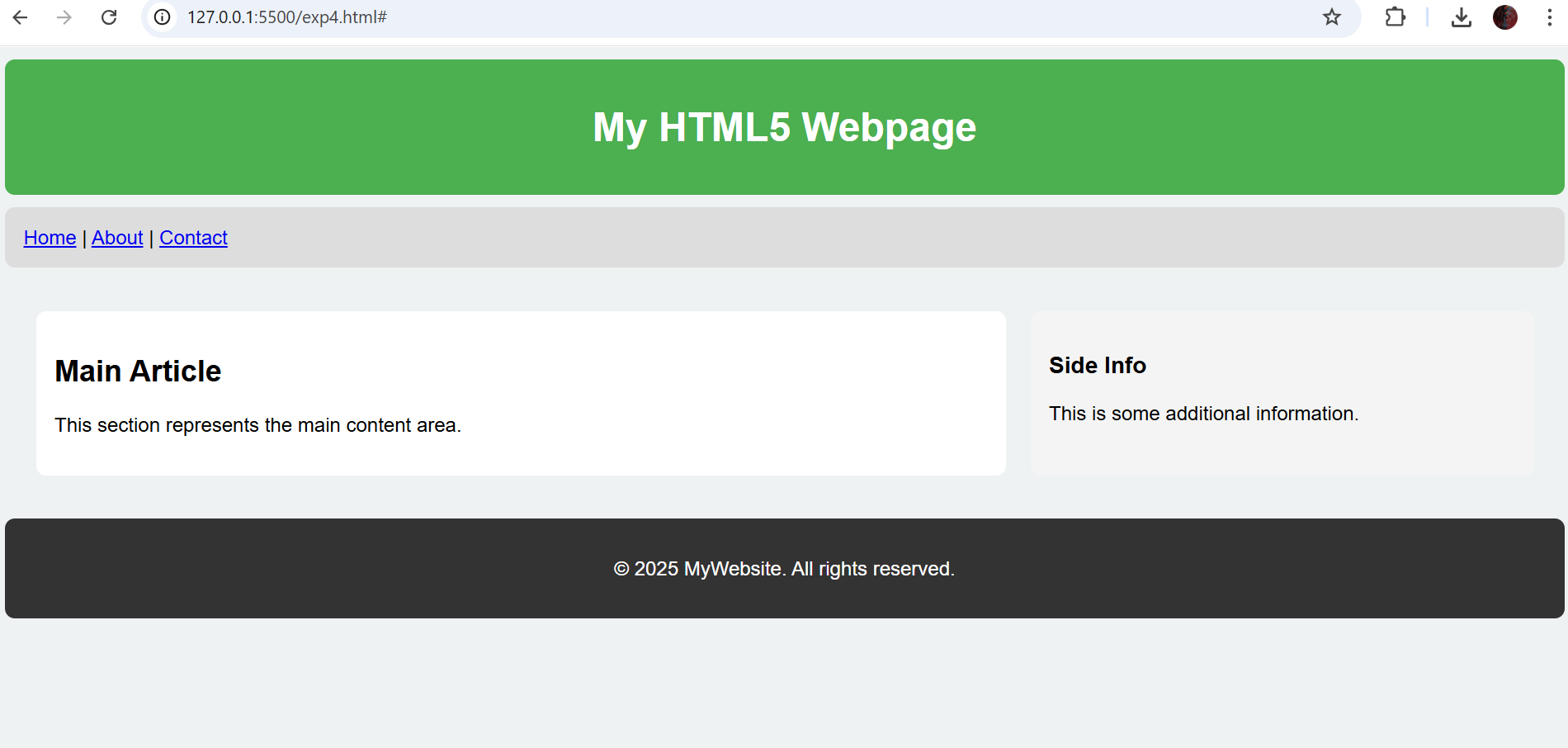
<p>&copy; 2025 MyWebsite. All rights reserved.</p>

</footer>

</body>

</html>

**OUTPUT:**

****

**RESULT:**

We have created a structured web page using HTML5 semantic elements.

|  |  |
| --- | --- |
| **Ex no:05** | **Create a Simple Calculator Using JavaScript** |
| **Date:** |

**AIM:**

To develop a basic calculator that performs addition, subtraction, multiplication, and division using JavaScript.

**ALGORITHMM:**

**Step1:** Create an HTML form with two input fields.

**Step2:** Add buttons for Add, Subtract, Multiply, and Divide.

**Step3:** Write JavaScript functions to do operations.

**Step4:** Read the input numbers inside JavaScript.

**Step5:** Perform calculations based on button clicked.

**Step6:** Display the result inside a div or paragraph.

**Step7:** Check for invalid input like division by zero.

**Step8:** Add proper event handlers for each button.

**Step9:** Save the file and open in browser.

**Step10:** Verify all calculator functions work properly.

**SOURCE CODE**

<!DOCTYPE html>

<html>

<head>

<title>Simple Calculator</title>

<style>

body {

font-family: Arial;

background-color: #f2f2f2;

padding: 20px;

}

input, button {

padding: 8px;

margin: 5px;

}

#result {

font-weight: bold;

margin-top: 10px;

}

</style>

</head>

<body>

<h2>Simple Calculator</h2>

<label>Enter First Number:</label>

<input type="number" id="num1"><br>

<label>Enter Second Number:</label>

<input type="number" id="num2"><br>

<button onclick="calculate('+')">Add</button>

<button onclick="calculate('-')">Subtract</button>

<button onclick="calculate('\*')">Multiply</button>

<button onclick="calculate('/')">Divide</button>

<p id="result"></p>

<script>

function calculate(operator) {

var n1 = parseFloat(document.getElementById("num1").value);

var n2 = parseFloat(document.getElementById("num2").value);

var result;

if (isNaN(n1) || isNaN(n2)) {

document.getElementById("result").innerText = "Please enter valid numbers.";

return;

}

switch (operator) {

case '+':

result = n1 + n2;

break;

case '-':

result = n1 - n2;

break;

case '\*':

result = n1 \* n2;

break;

case '/':

result = n2 !== 0 ? (n1 / n2) : "Cannot divide by zero";

break;

}

document.getElementById("result").innerText = "Result: " + result;

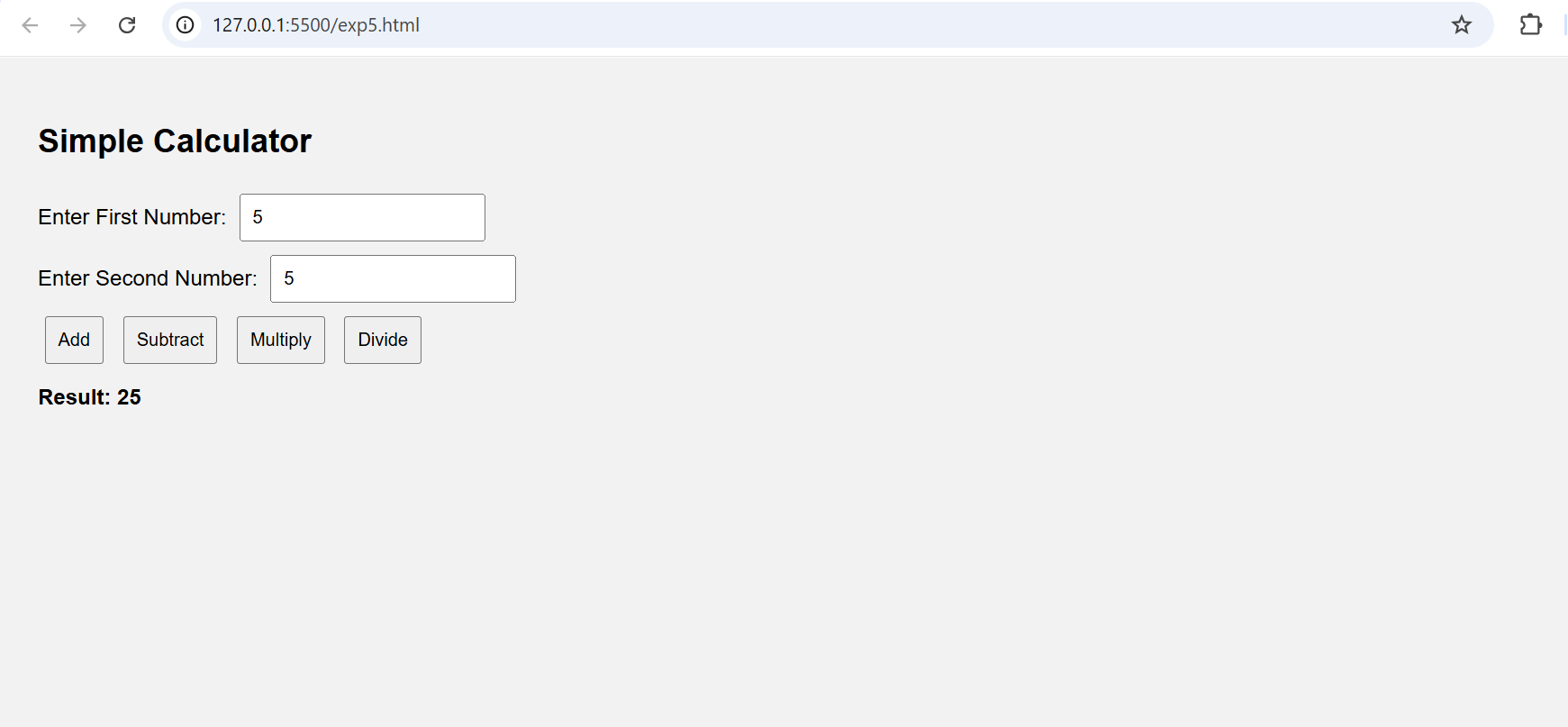
}

</script>

</body>

</html>

**OUTPUT:**

****

**RESULT:**

We have developed a basic calculator that performs addition, subtraction, multiplication, and division using JavaScript.

|  |  |
| --- | --- |
| **Ex no:06** | **Design a Registration Form** |
| **Date:** |

**AIM:**

To design a registration form that validate fields such as name, email, and password using JavaScript before submitting the form.

**ALGORITHM:**

**Step1:** Design an HTML page with <input> fields for two numbers.

**Step2:**  Create buttons for operations like Add, Subtract, Multiply, Divide.

**Step3:**  Write JavaScript functions to perform each operation.

**Step4:** Fetch numbers from input fields inside functions.

**Step5:** Perform calculations and store results.

**Step6:** Display the output inside a paragraph or div.

**Step7:** Use onclick event for buttons to call the functions.

**Step8:** Check for invalid inputs like dividing by zero.

**Step9:** Close all tags and save the file.

**Step10:** Open the HTML file and test all operations.

·

**SOURCE CODE**

<!DOCTYPE html>

<html>

<head>

<title>Form Validation</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #eef;

padding: 20px;

}

.error {

color: red;

}

input {

padding: 5px;

margin: 5px;

}

</style>

</head>

<body>

<h2>Registration Form</h2>

<form name="regForm" onsubmit="return validateForm()">

Name: <input type="text" name="name"><br>

Email: <input type="text" name="email"><br>

Password: <input type="password" name="password"><br>

<input type="submit" value="Submit">

</form>

<p id="errorMsg" class="error"></p>

<script>

function validateForm() {

let name = document.forms["regForm"]["name"].value;

let email = document.forms["regForm"]["email"].value;

let password = document.forms["regForm"]["password"].value;

let errorMsg = document.getElementById("errorMsg");

if (name == "" || email == "" || password == "") {

errorMsg.innerHTML = "All fields must be filled out.";

return false;

}

let emailPattern = /^[^ ]+@[^ ]+\.[a-z]{2,3}$/;

if (!email.match(emailPattern)) {

errorMsg.innerHTML = "Invalid email format.";

return false;

}

if (password.length < 6) {

errorMsg.innerHTML = "Password must be at least 6 characters.";

return false;

}

errorMsg.innerHTML = "";

alert("Form submitted successfully!");

return true;

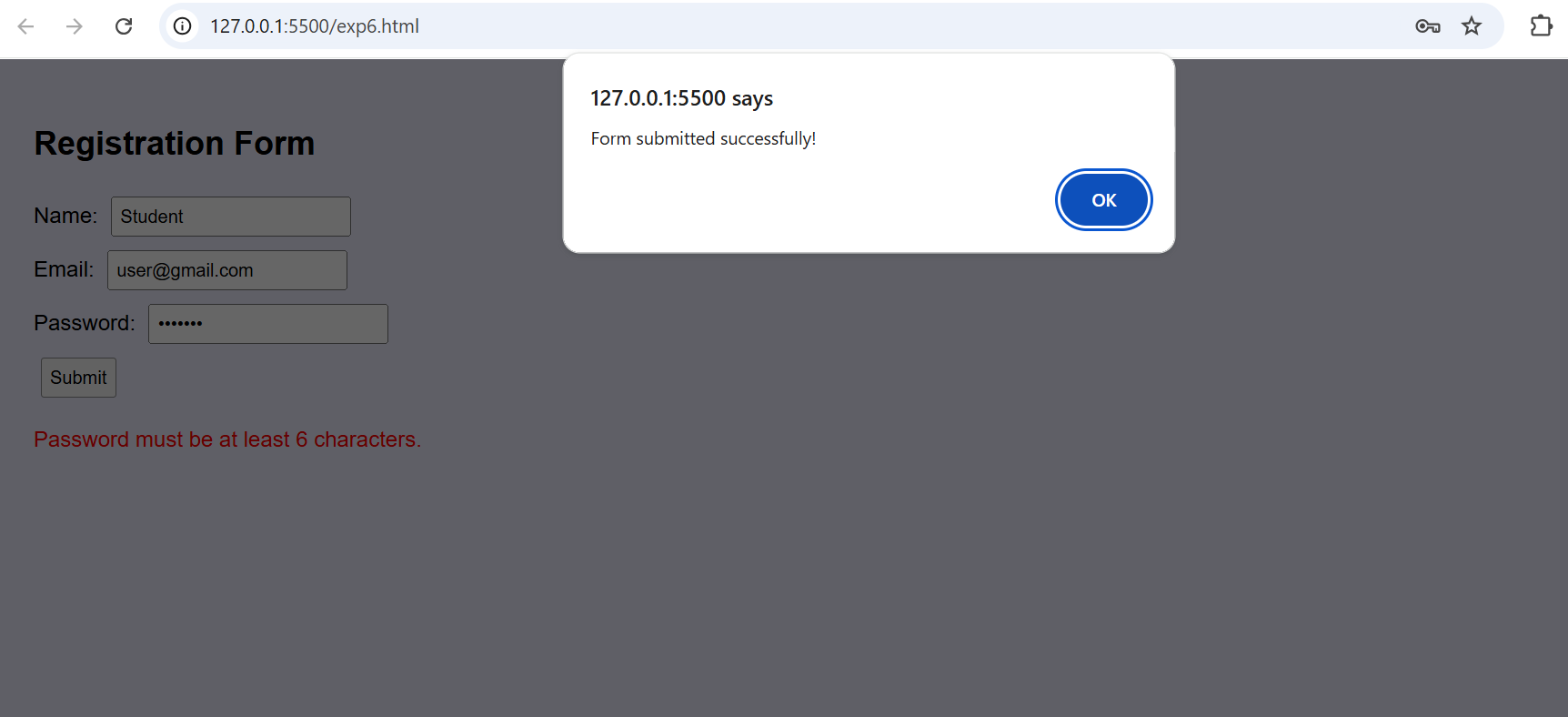
}

</script>

</body>

</html>

**OUTPUT:**

****

**RESULT:**

Thus we have created a Registration form that validates fields such as name, email, and password using JavaScript before submitting the form.

|  |  |
| --- | --- |
| **Ex no:07** | **Develop a responsive website using bootstrap** |
| **Date:** |

**AIM:**

To develop a responsive website using Bootstrap that adjusts to different screen sizes (e.g., desktop, tablet, mobile).

**ALGORITHM:**

**Step 1:** Include the Bootstrap CSS and JS files.

**Step 2:** Set up the basic HTML structure for the web page.

**Step 3:** Create a navigation bar that adapts to different screen sizes.

**Step 4:** Use Bootstrap's grid system to create a responsive layout.

**Step 5:** Add images, text, or any other content within grid columns.

**Step 6:** Make the layout responsive by using appropriate Bootstrap classes.

**Step 7:** Add a footer that stays at the bottom of the page.

**Step 8:** Test the website on different screen sizes to ensure responsiveness.

**Step 9:** Style the website with custom CSS if needed.

**Step 10:** Deploy the website and test it in various browsers.

**SOURCECODE**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Responsive Website</title>

<link href="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.css" rel="stylesheet">

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-light bg-light">

<a class="navbar-brand" href="#">My Website</a>

<button class="navbar-toggler" type="button" data-toggle="collapse" data-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false" aria-label="Toggle navigation">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNav">

<ul class="navbar-nav ml-auto">

<li class="nav-item active">

<a class="nav-link" href="#">Home</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">About</a>

</li>

<li class="nav-item">

<a class="nav-link" href="#">Contact</a>

</li>

</ul>

</div>

</nav>

<div class="container my-5">

<div class="row">

<div class="col-md-6">

<h2>Welcome to My Website</h2>

<p>This is a responsive website built using Bootstrap.</p>

</div>

<div class="col-md-6">

<img src="logo.png" class="img-fluid" alt="Responsive Image">

</div>

</div>

</div>

<footer class="bg-light py-3">

<div class="container text-center">

<p>&copy; 2025 My Website</p>

</div>

</footer>

<script src="https://code.jquery.com/jquery-3.5.1.slim.min.js"></script>

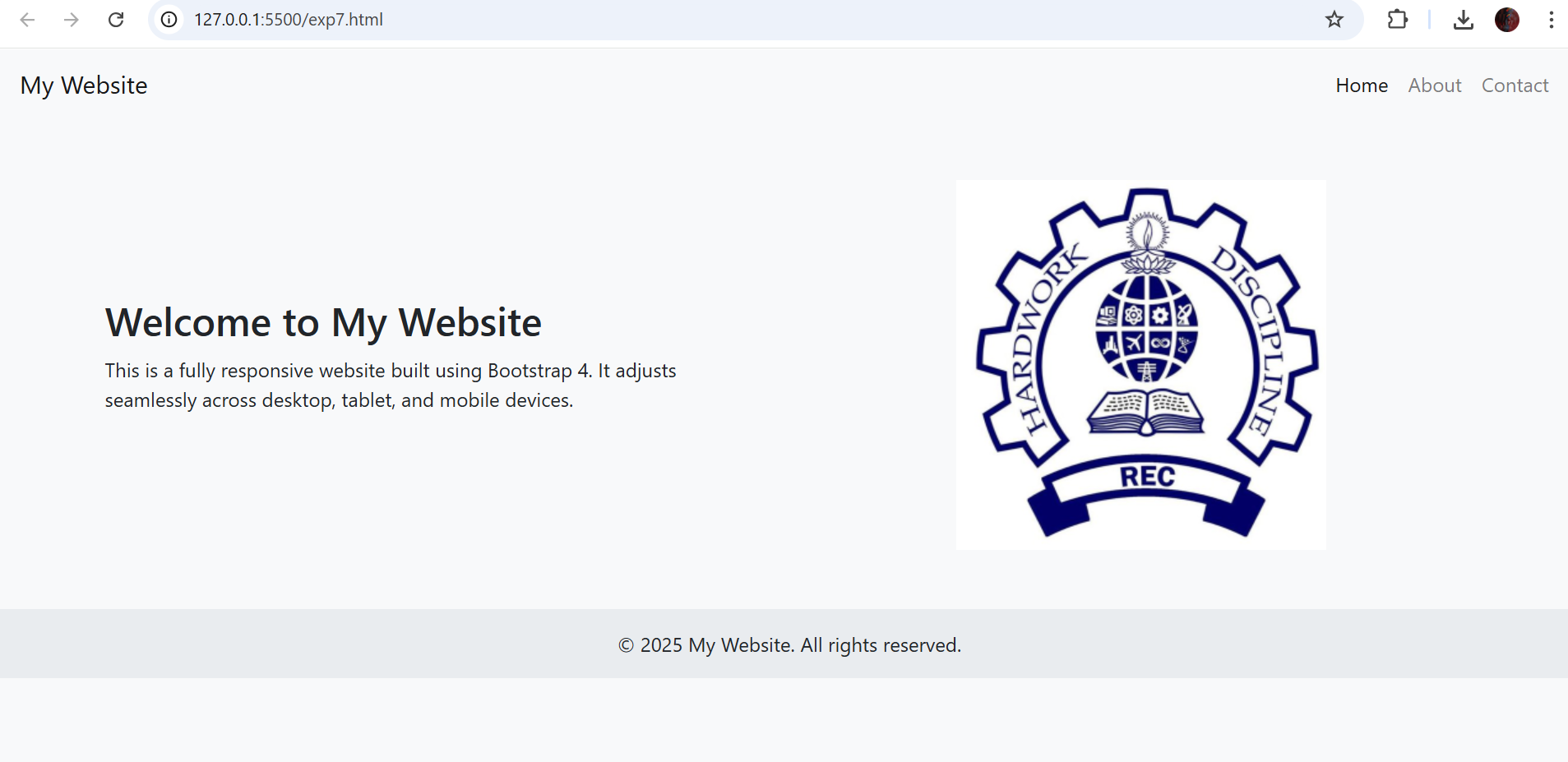
<script src="https://cdn.jsdelivr.net/npm/@popperjs/core@2.5.2/dist/umd/popper.min.js"></script>

<script src="https://maxcdn.bootstrapcdn.com/bootstrap/4.5.2/js/bootstrap.min.js"></script>

</body>

</html>

**OUTPUT**

****

**RESULT:**

A responsive website with a navigation bar, content area, and footer. The layout adapts automatically based on the screen size (desktop, tablet, mobile).

|  |  |
| --- | --- |
| **Ex no:08** | **Design a web page with grid system using Bootstrap** |
| **Date:** |

**AIM:**

To create a web page layout using Bootstrap's Grid System.

**ALGORITHM:**

**Step 1:** Create a new HTML file.

**Step 2:** Link Bootstrap CSS and JS files (via CDN).

**Step 3:** Create a container (<div class="container">).

**Step 4:** Create a row (<div class="row">).

**Step 5:** Add columns using Bootstrap classes like col-md-4.

**Step 6:** Add some sample content inside each column.

**Step 7:** Repeat rows and columns if needed.

**Step 8:** Style with background colors for better visualization.

**Step 9:** Save the HTML file.

**Step 10:** Open it in a browser to view the grid layout.

**SOURCE CODE**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Bootstrap Grid System</title>

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

</head>

<body>

<div class="container mt-4">

<h1 class="text-center">Bootstrap Grid Example</h1>

<div class="row">

<div class="col-md-4 bg-primary text-white p-3">Column 1</div>

<div class="col-md-4 bg-success text-white p-3">Column 2</div>

<div class="col-md-4 bg-danger text-white p-3">Column 3</div>

</div>

<div class="row mt-4">

<div class="col-md-6 bg-warning text-dark p-3">Column 4</div>

<div class="col-md-6 bg-info text-white p-3">Column 5</div>

</div>

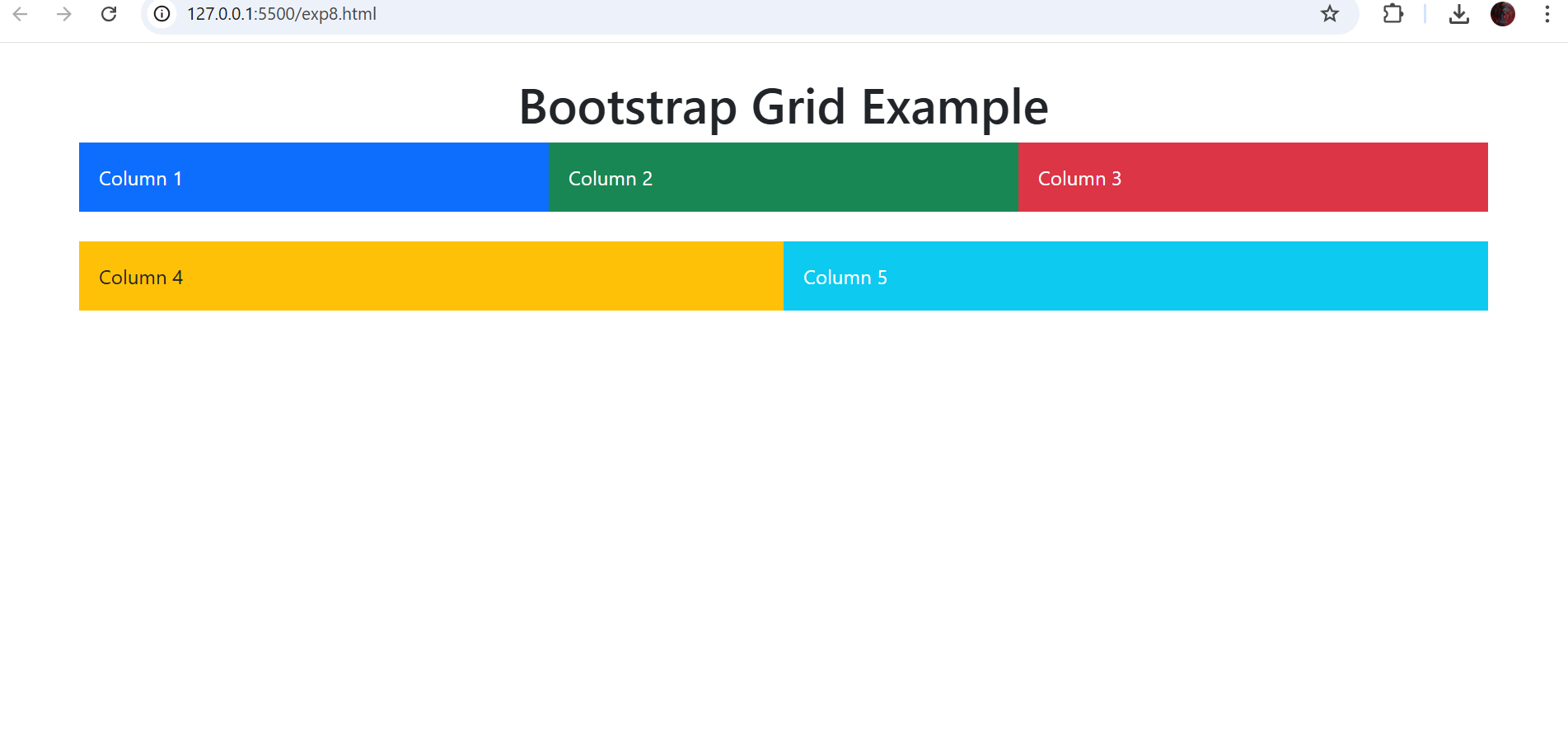
</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

</body>

</html>

**OUTPUT:**

****

**RESULT:**

Thus we have created A web page showing a grid with 3 columns in the first row and 2 columns in the second row, using Bootstrap.

|  |  |
| --- | --- |
| **Ex no:09** | **Design a web page with dropdown navigation bar and pagination** |
| **Date:** |

**AIM:**

To create a navigation bar with a dropdown menu and a pagination component using Bootstrap.

**ALGORITHM:**

**Step 1:** Create a new HTML file.

**Step 2:** Link Bootstrap CSS and JS files.

**Step 3:** Create a <nav> tag for the navigation bar.

**Step 4:** Add brand name and links inside the navbar.

**Step 5:** Add a dropdown menu inside the navbar.

**Step 6:** Create the dropdown items inside it.

**Step 7:** Below the navbar, create a pagination component.

**Step 8:** Style the navbar and pagination.

**Step 9:** Save the HTML file.

**Step 10:** Open it in a browser and test the dropdown and pagination.

**SOURCE CODE**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Dropdown Navbar and Pagination</title>

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

</head>

<body>

<nav class="navbar navbar-expand-lg navbar-dark bg-dark">

<div class="container-fluid">

<a class="navbar-brand" href="#">MySite</a>

<button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNavDropdown">

<span class="navbar-toggler-icon"></span>

</button>

<div class="collapse navbar-collapse" id="navbarNavDropdown">

<ul class="navbar-nav">

<li class="nav-item">

<a class="nav-link active" href="#">Home</a>

</li>

<li class="nav-item dropdown">

<a class="nav-link dropdown-toggle" href="#" id="navbarDropdownMenuLink" role="button" data-bs-toggle="dropdown">

Services

</a>

<ul class="dropdown-menu">

<li><a class="dropdown-item" href="#">Web Design</a></li>

<li><a class="dropdown-item" href="#">Development</a></li>

<li><a class="dropdown-item" href="#">SEO</a></li>

</ul>

</li>

</ul>

</div>

</div>

</nav>

<div class="container mt-5">

<h2 class="text-center">Pagination Example</h2>

<nav aria-label="Page navigation">

<ul class="pagination justify-content-center">

<li class="page-item disabled">

<a class="page-link" href="#">Previous</a>

</li>

<li class="page-item"><a class="page-link" href="#">1</a></li>

<li class="page-item"><a class="page-link" href="#">2</a></li>

<li class="page-item"><a class="page-link" href="#">3</a></li>

<li class="page-item">

<a class="page-link" href="#">Next</a>

</li>

</ul>

</nav>

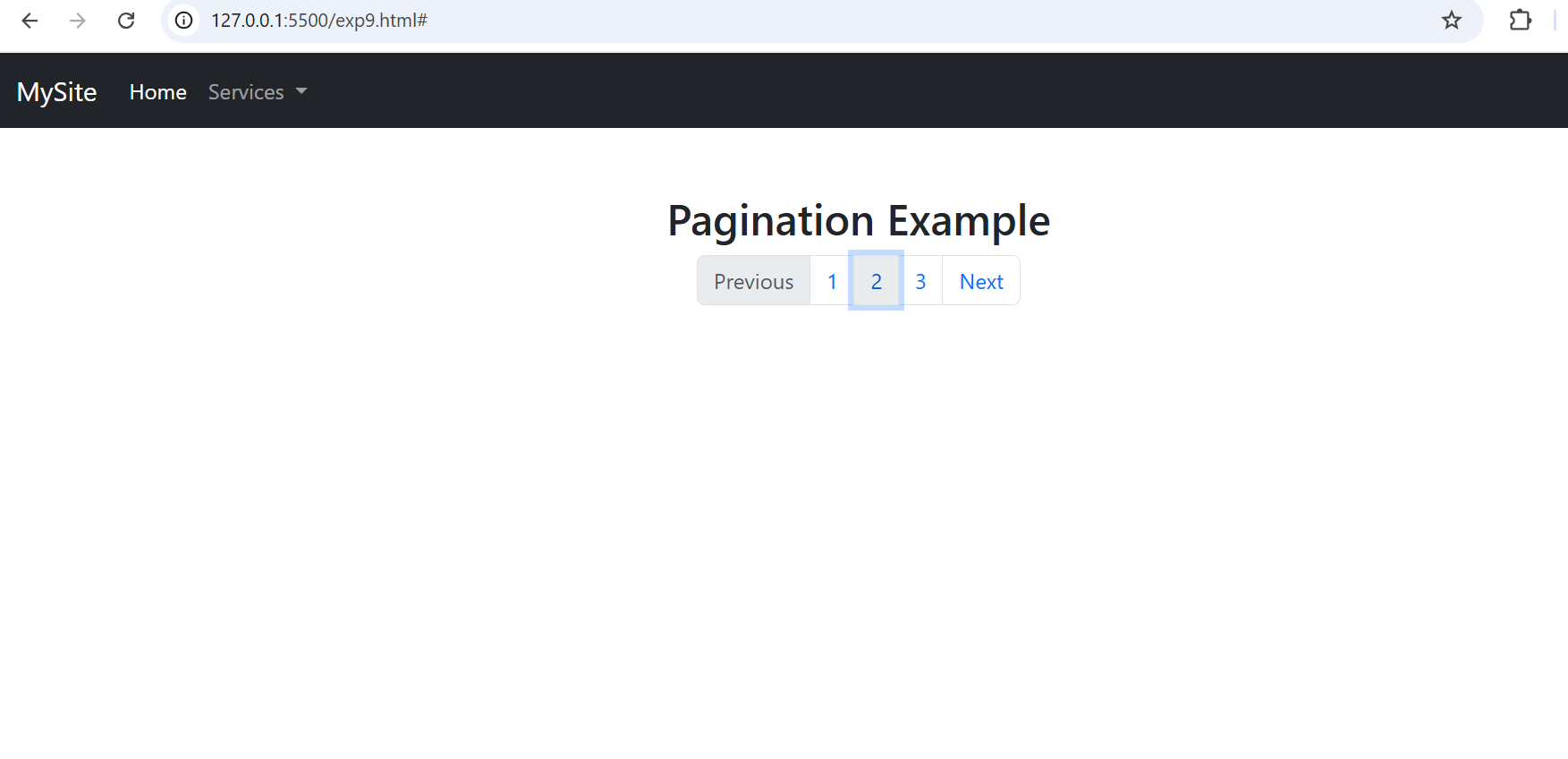
</div>

<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></script>

</body>

</html>

**OUTPUT:**

****

**RESULT:**

Thus we have created a navigation bar with a dropdown menu and a pagination component using Bootstrap.

|  |  |
| --- | --- |
| **Ex no:10** | **Design a web page with jQuery selector** |
| **Date:** |

**AIM:**

To create a web page and demonstrate using jQuery selectors to manipulate HTML elements.

**ALGORITHM:**

**Step 1:** Create a new HTML file.

**Step 2:** Link jQuery library (via CDN).

**Step 3:** Create some HTML elements like heading, paragraph, button.

**Step 4:** Write a <script> tag.

**Step 5:** Use jQuery to select elements (like $("h1"), $(".class"), $("#id")).

**Step 6:** Apply effects like hide(), show(), toggle().

**Step 7:** Attach a click event to the button.

**Step 8:** Write jQuery code inside $(document).ready().

**Step 9:** Save the HTML file.

**Step 10:** Open it in a browser and test the selectors.

**SOURCE CODE**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>jQuery Selector Example</title>

<script src="https://code.jquery.com/jquery-3.7.1.min.js"></script>

</head>

<body>

<h1 id="mainHeading">Welcome to jQuery Selector Example</h1>

<p class="info">This is a paragraph with class 'info'.</p>

<p>This is a normal paragraph.</p>

<button id="toggleButton">Toggle Info Paragraph</button>

<script>

$(document).ready(function(){

$("#toggleButton").click(function(){

$(".info").toggle();

});

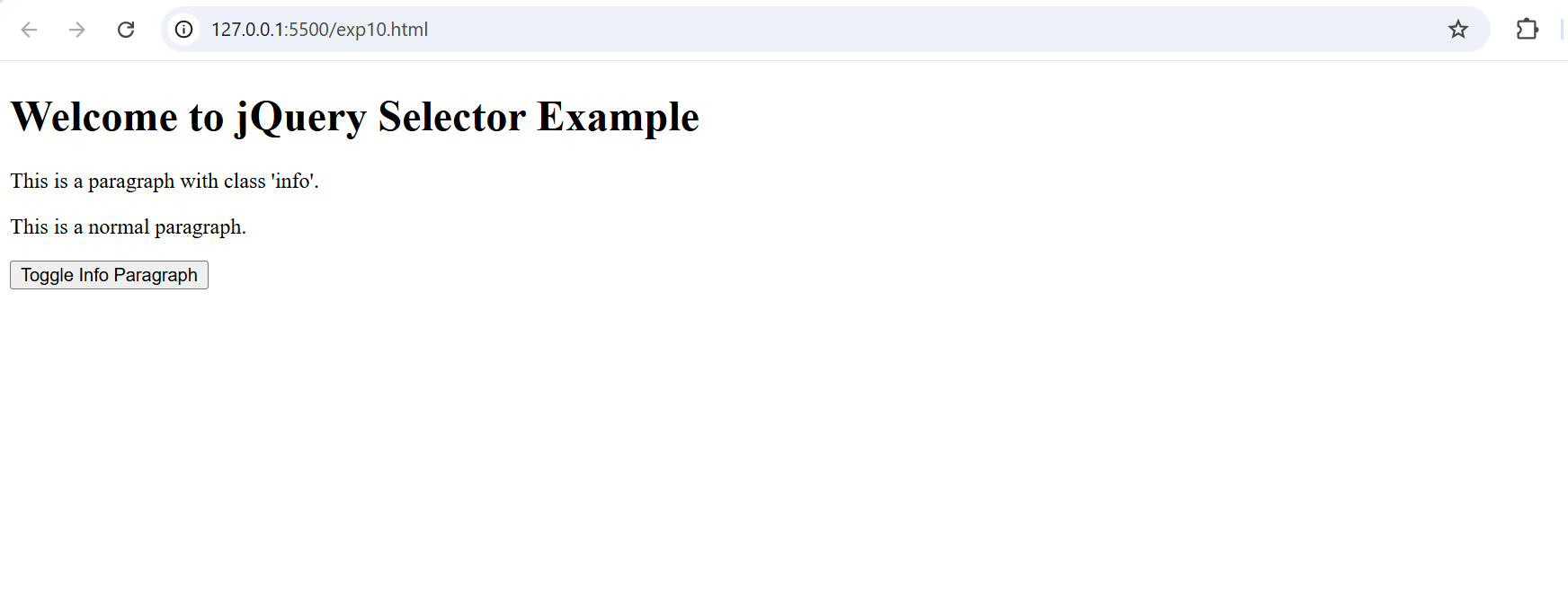
});

</script>

</body>

</html>

**OUTPUT:**

****

**RESULT:**

Thus we have created such that When the user clicks the button, the paragraph with class info will show/hide (toggle) using jQuery selector.

|  |  |
| --- | --- |
| **Ex no:11** | **Design a simple web page using jQuery for Animation Effects** |
| **Date:** |

**AIM:**

To create a webpage that demonstrates basic jQuery animation effects such as hide, show, fade, and slide.

**ALGORITHMM:**

**Step 1:** Create an HTML page with a div or box.

**Step 2:** Link the jQuery library using a CDN in <head>.

**Step 3:** Add buttons like Hide, Show, FadeIn, FadeOut, SlideUp, SlideDown.

**Step 4:** Write jQuery code to perform animations.

**Step 5:**Use jQuery methods like .hide(), .show(), .fadeIn(), .fadeOut().

**Step 6:** Bind button clicks to respective jQuery functions.

**Step 7:** Save the file and open in browser.

**Step 8:** Click buttons and check if animation works.

**Step 9:**Fix errors if any animation does not happen.

**Step 10:** Test all animation effects.

**SOURCE CODE:**

<!DOCTYPE html>

<html>

<head>

<title>jQuery Animation Effects</title>

<script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>

<style>

#box {

width: 200px;

height: 200px;

background-color: tomato;

margin: 20px auto;

text-align: center;

line-height: 200px;

font-weight: bold;

color: white;

font-size: 20px;

}

button {

margin: 5px;

padding: 10px 15px;

}

</style>

</head>

<body>

<h2 style="text-align:center;">jQuery Animation Demo</h2>

<div id="box">Animate Me!</div>

<div style="text-align:center;">

<button id="hideBtn">Hide</button>

<button id="showBtn">Show</button>

<button id="fadeOutBtn">Fade Out</button>

<button id="fadeInBtn">Fade In</button>

<button id="slideUpBtn">Slide Up</button>

<button id="slideDownBtn">Slide Down</button>

<button id="toggleBtn">Toggle</button>

</div>

<script>

$(document).ready(function(){

$("#hideBtn").click(function(){

$("#box").hide();

});

$("#showBtn").click(function(){

$("#box").show();

});

$("#fadeOutBtn").click(function(){

$("#box").fadeOut();

});

$("#fadeInBtn").click(function(){

$("#box").fadeIn();

});

$("#slideUpBtn").click(function(){

$("#box").slideUp();

});

$("#slideDownBtn").click(function(){

$("#box").slideDown();

});

$("#toggleBtn").click(function(){

$("#box").toggle();

});

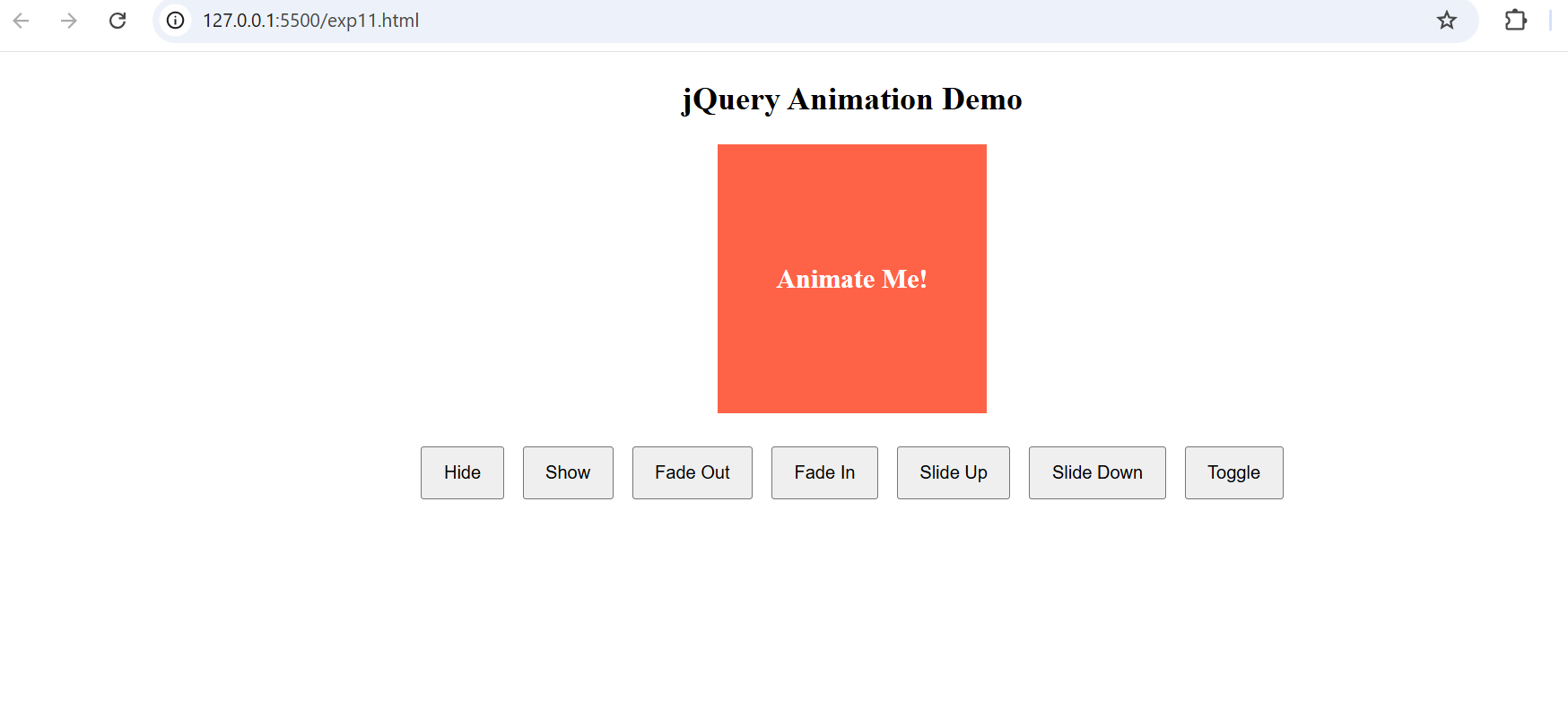
});

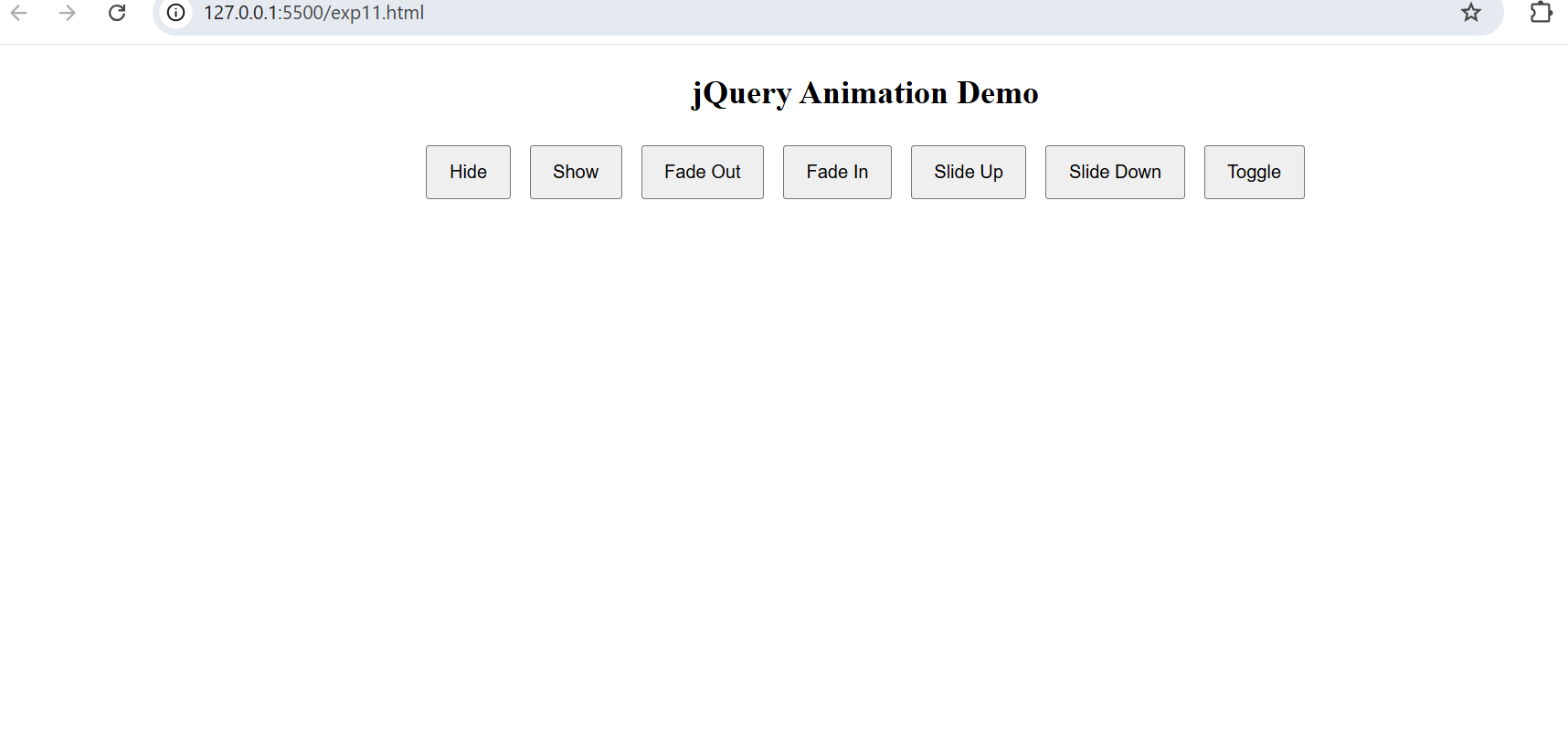
</script>

</body>

</html>

**OUTPUT:**

****

****

**RESULT:**

Thus we have created a webpage that demonstrates basic jQuery animation effects such as

hide, show, fade, and slide.

|  |  |
| --- | --- |
| **Ex no:12** | **A web page to calculate the factorial of a number using php** |
| **Date:** |

**AIM:**

To create a simple web page where users can input a number, and the page will display its factorial.

**ALGORITHM:**

**Step 1:** Create an HTML form to accept a number input.

**Step 2:** Define a PHP script to handle the form submission.

**Step 3:** Retrieve the number input using $\_POST.

**Step 4:** Define a function to calculate the factorial.

**Step 5:** Use a loop to calculate the factorial of the entered number.

**Step 6:** Display the result on the web page.

**Step 7:** If the input is invalid (e.g., negative or non-numeric), show an error message.

**Step 8:** Allow users to enter another number for calculation.

**Step 9:** Style the form using basic CSS.

**Step 10:** Test the page with various inputs and display the correct factorial.

**SOURCE CODE:**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Factorial Calculator</title>

</head>

<body>

<h2>Factorial Calculator</h2>

<form method="post">

<label for="number">Enter a number:</label>

<input type="number" id="number" name="number" required>

<input type="submit" value="Calculate">

</form>

<?php

if ($\_SERVER["REQUEST\_METHOD"] == "POST") {

$number = $\_POST['number'];

if (is\_numeric($number) && $number >= 0) {

function factorial($n) {

$result = 1;

for ($i = 1; $i <= $n; $i++) {

$result \*= $i;

}

return $result;

}

$result = factorial($number);

echo "<p>The factorial of $number is: $result</p>";

} else {

echo "<p>Please enter a valid positive number.</p>";

}

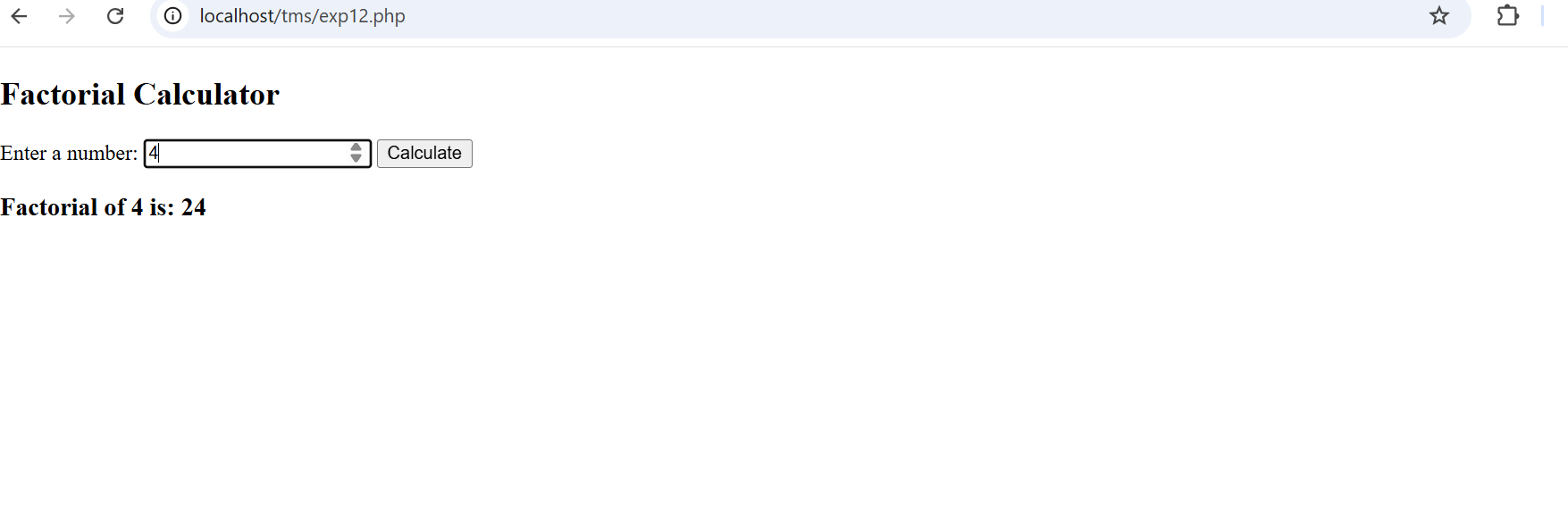
}

?>

</body>

</html>

**OUTPUT:**

****

**RESULT:**

We have created a simple web page where users can input a number, and the page will display its factorial.

|  |  |
| --- | --- |
| **Ex no:13** | **A Web page to Perform Arithmetic Operations Using PHP** |
| **Date:** |

**AIM:**

To create a PHP webpage that performs arithmetic operations (addition, subtraction, multiplication, division) on two numbers entered by the user.

**ALGORITHMM:**

**Step 1: Start with a basic HTML form inside a .php file.**

**Step 2: Create two input fields for the numbers.**

**Step 3: Add a dropdown menu to select the operation (Addition, Subtraction, Multiplication, Division).**

**Step 4: Set the form method to POST to securely send data.**

**Step 5: Retrieve the input values in PHP using $\_POST.**

**Step 6: Apply conditional logic (if-else statements) to check which operation is selected.**

**Step 7:** Perform the selected arithmetic operation and calculate the result.

**Step 8:** Display the calculated result dynamically after the form is submitted.

**Step 9:** Handle special cases like division by zero to avoid errors.

**Step 10:** Save the file with a .php extension and run it using a local server (like XAMPP) to test.

**SOURCE CODE**

<!DOCTYPE html>

<html>

<head>

<title>Arithmetic Operations</title>

</head>

<body>

<h2>Arithmetic Operation Calculator</h2>

<form method="post">

Enter First Number: <input type="text" name="num1" required><br><br>

Enter Second Number: <input type="text" name="num2" required><br><br>

Select Operation:

<select name="operation">

<option value="add">Addition</option>

<option value="subtract">Subtraction</option>

<option value="multiply">Multiplication</option>

<option value="divide">Division</option>

</select><br><br>

<input type="submit" name="submit" value="Calculate">

</form>

<?php

if(isset($\_POST['submit'])){

$num1 = $\_POST['num1'];

$num2 = $\_POST['num2'];

$operation = $\_POST['operation'];

if($operation == "add"){

$result = $num1 + $num2;

echo "<h3>Result of Addition: $result</h3>";

}

elseif($operation == "subtract"){

$result = $num1 - $num2;

echo "<h3>Result of Subtraction: $result</h3>";

}

elseif($operation == "multiply"){

$result = $num1 \* $num2;

echo "<h3>Result of Multiplication: $result</h3>";

}

elseif($operation == "divide"){

if($num2 != 0){

$result = $num1 / $num2;

echo "<h3>Result of Division: $result</h3>";

} else {

echo "<h3>Cannot divide by zero!</h3>";

}

} else {

echo "<h3>Invalid Operation Selected</h3>";

}

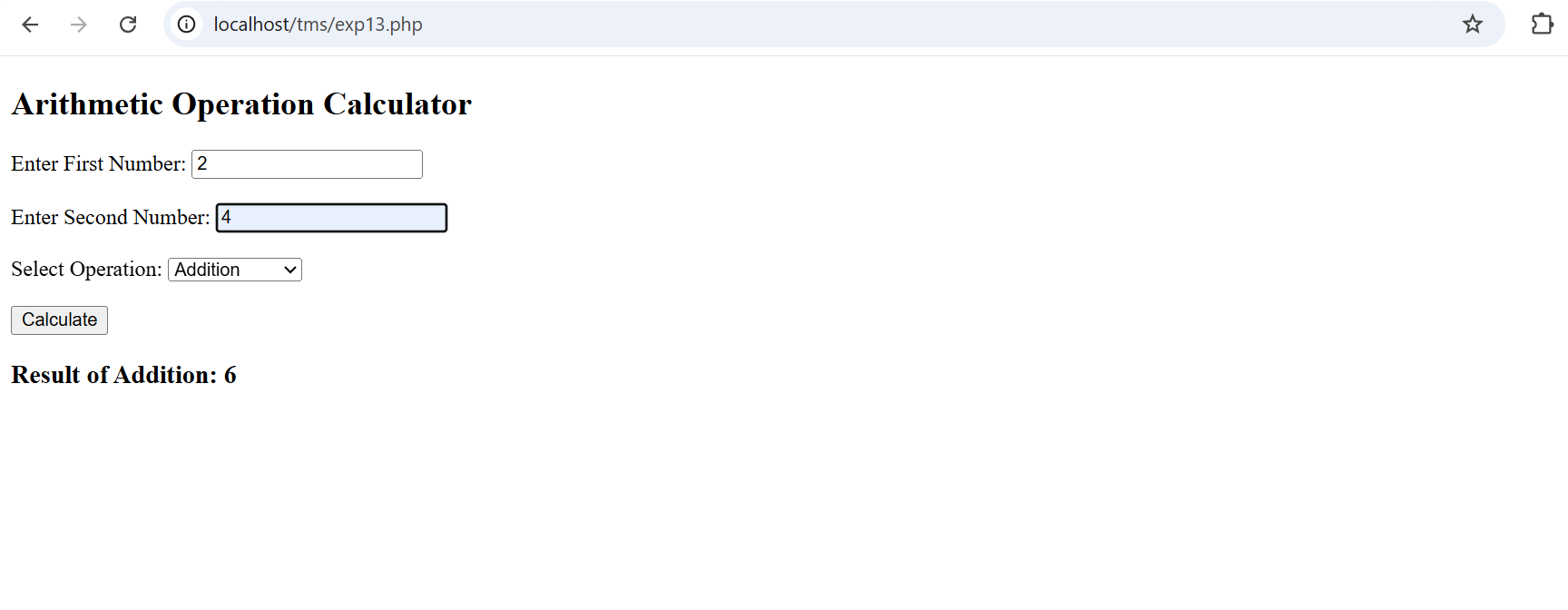
}

?>

</body>

</html>

**OUTPUT:**

****

**RESULT:**

Thus, the PHP webpage for performing arithmetic operations was successfully created and tested.

|  |  |
| --- | --- |
| **Ex no:14** | **Program using regular expression in PHP** |
| **Date:** |

**AIM:**

To write a PHP program that uses regular expressions to validate strings like email addresses and phone numbers.

**ALGORITHMM:**

**Step 1:** Open a new PHP file and use <?php and ?> tags to start the script.

**Step 2:** Define sample input strings for email and phone number.

**Step 3:** Create a regular expression pattern to validate the email format.

**Step 4:** Create a regular expression pattern to validate the phone number (10 digits).

**Step 5:** Use preg\_match() function to check if the email matches the email pattern.

**Step 6:** Use preg\_match() function to check if the phone number matches the phone number pattern.

**Step 7:** If the email matches the pattern, display a success message ("Email is valid").

**Step 8:** If the email does not match the pattern, display an invalid message ("Invalid email format").

**Step 9:** If the phone number matches the pattern, display a success message ("Phone number is valid").

**Step 10:** If the phone number does not match the pattern, display an invalid message ("Invalid phone number format").

**Step 11:** Save the PHP file with a .php extension and run it on a local server like XAMPP or WAMP.

**Step 12:** Test the code with different sample inputs to check the validation results.

**SOURCE CODE**

<?php

// Example for validating an email

$email = "example@gmail.com";

if (preg\_match("/^[a-zA-Z0-9.\_%+-]+@[a-zA-Z0-9.-]+\.[a-zA-Z]{2,}$/", $email)) {

echo "Valid Email Address.<br>";

} else {

echo "Invalid Email Address.<br>";

}

// Example for validating a phone number

$phone = "9876543210";

if (preg\_match("/^[0-9]{10}$/", $phone)) {

echo "Valid Phone Number.";

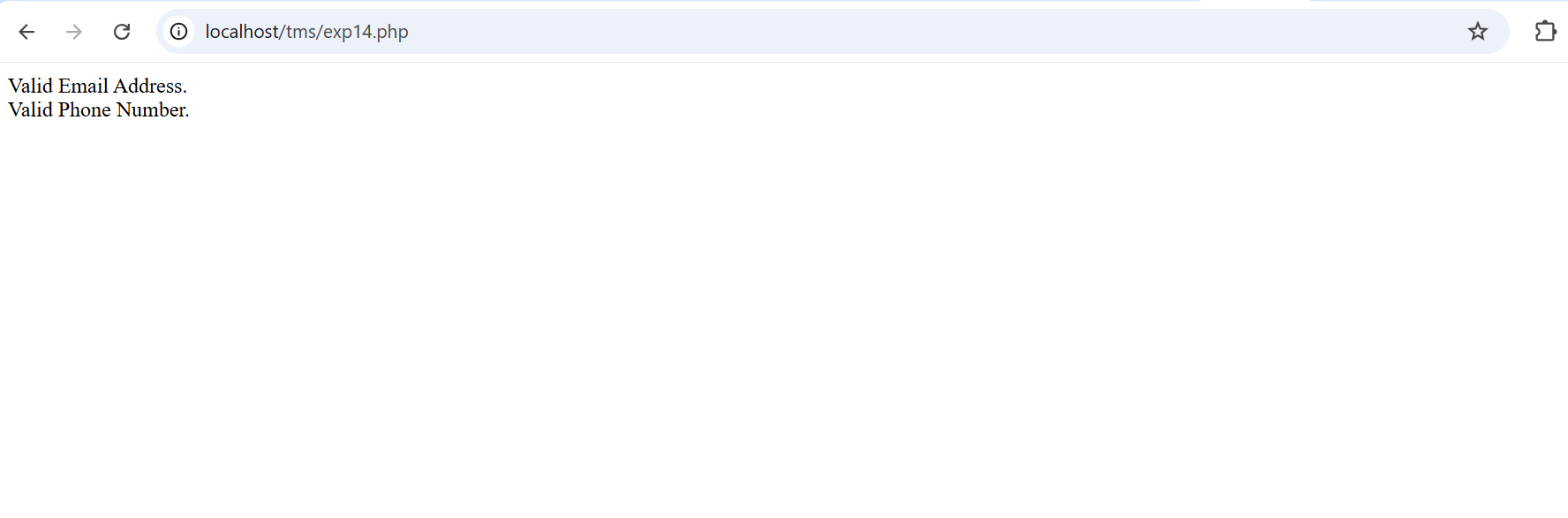
} else {

echo "Invalid Phone Number.";

}

?>

**OUTPUT**

****

**RESULT:**

Thus, a PHP program using regular expressions to validate email addresses and phone numbers was successfully developed and tested.