Web Technology Lab Manual

BCA V SEM
SCHOOL OF COMPUTER APPLICATIONS

GM UNIVERSITY
Innovating Minds

Faculty of Computing & Information Technology

GM University, Davanagere - 577006

www.gmu.ac.in

Prepared by: Mrs. USHA N

PART A – HTML Programs (Static Web Pages)

1. Design a Home Page for an Online Book Store using frames:

- Divide the page into three frames (top, left, right)
- Top Frame: Display site title and navigation links (Home, Login, Registration, Catalogue)
- Left Frame: Display category links (e.g., CSE, ECE, EEE)
- Right Frame: Display default welcome message or contents loaded via left frame links
- Use <frameset> and <frame> tags.

left.html

```
<html>
<head>
  <title>Left Frame</title>
</head>
<body style="background-color:#e0e0e0; font-family:Arial;">
  <h3>Categories</h3>
  <a href="cse.html" target="rightFrame">CSE</a><br>
  <a href="ece.html" target="rightFrame">ECE</a><br>
  <a href="eee.html" target="rightFrame">EEE</a>
</body>
</html>
```

Right.html

```
<html>
<head>
  <title>Right Frame</title>
</head>
<body style="font-family:Arial;">
  <h2>Welcome to the Online Book Store</h2>
  <p>Select a category from the left or use the menu above to explore more options.</p>
</body>
</html>
```

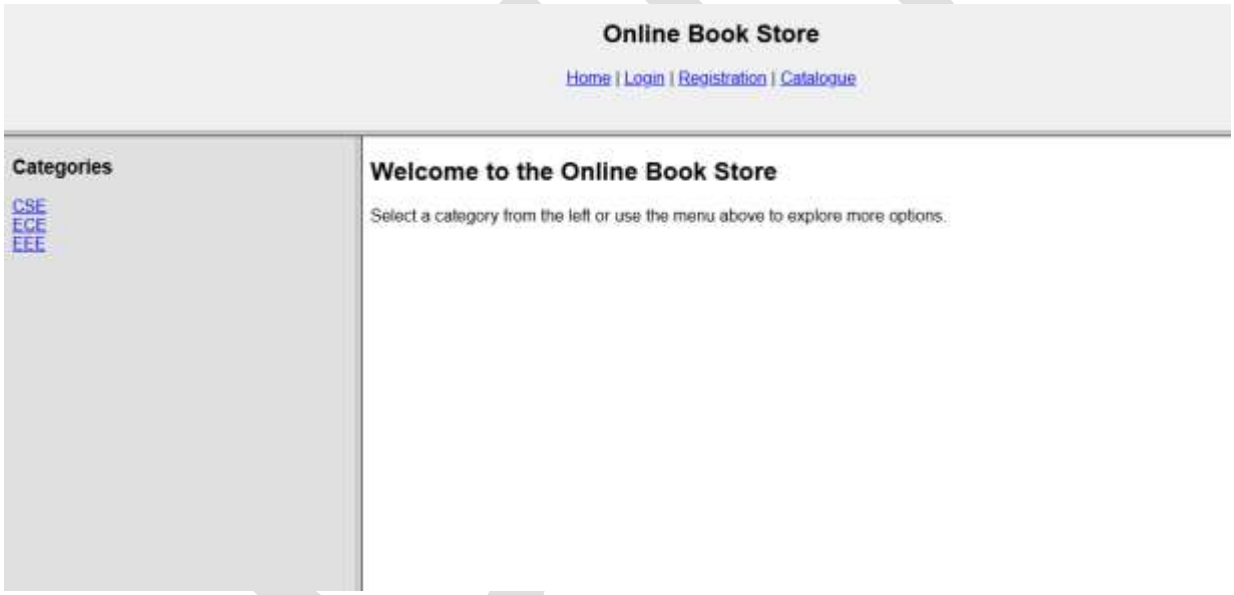
top.html

```
<html>
<head>
  <title>Top Frame</title>
</head>
<body style="text-align:center; background-color:#f0f0f0; font-family:Arial;">
  <h2>Online Book Store</h2>
  <a href="right.html" target="rightFrame">Home</a> |
  <a href="login.html" target="rightFrame">Login</a> |
  <a href="register.html" target="rightFrame">Registration</a> |
  <a href="catalogue.html" target="rightFrame">Catalogue</a>
</body>
</html>
```

index.html

```
<html>
<head>
   <title>Online Book Store</title>
</head>
<!-- Divide the page into frames -->
<frameset rows="20%,80%">
   <!-- Top frame for site title and navigation -->
   <frame src="top.html" name="topFrame">
   <!-- Remaining area is divided into left and right frames -->
   <frameset cols="25%,75%">
      <!-- Left frame for category links -->
      <frame src="left.html" name="leftFrame">
            <!-- Right frame for displaying content -->
      <frame src="right.html" name="rightFrame">
   </frameset>
</frameset>
</html>
```

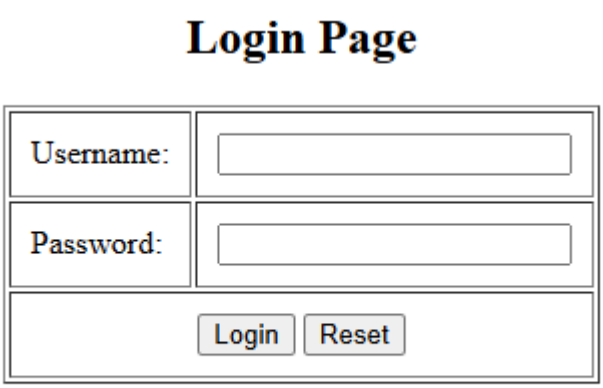
OUTPUT:



2. Create a Login Page using HTML Forms:
- Use <form> tag to collect Username and Password
- Include Submit and Reset buttons
- Use proper <input> types (text, password)
- Align the form using <table> or tags.

```
<html>
<head>
  <title>Login Page</title>
</head>
<body>
  <h2 align="center">Login Page</h2>

  <form action="login_process.html" method="post">
    <table align="center" cellpadding="10" border="1">
      <tr>
        <td><label for="username">Username:</label></td>
        <td><input type="text" id="username" name="username" required></td>
      </tr>
      <tr>
        <td><label for="password">Password:</label></td>
        <td><input type="password" id="password" name="password"
required></td>
      </tr>
      <tr>
        <td colspan="2" align="center">
          <input type="submit" value="Login">
          <input type="reset" value="Reset">
        </td>
      </tr>
    </table>
  </form>
</body>
</html>
```

OUTPUT:



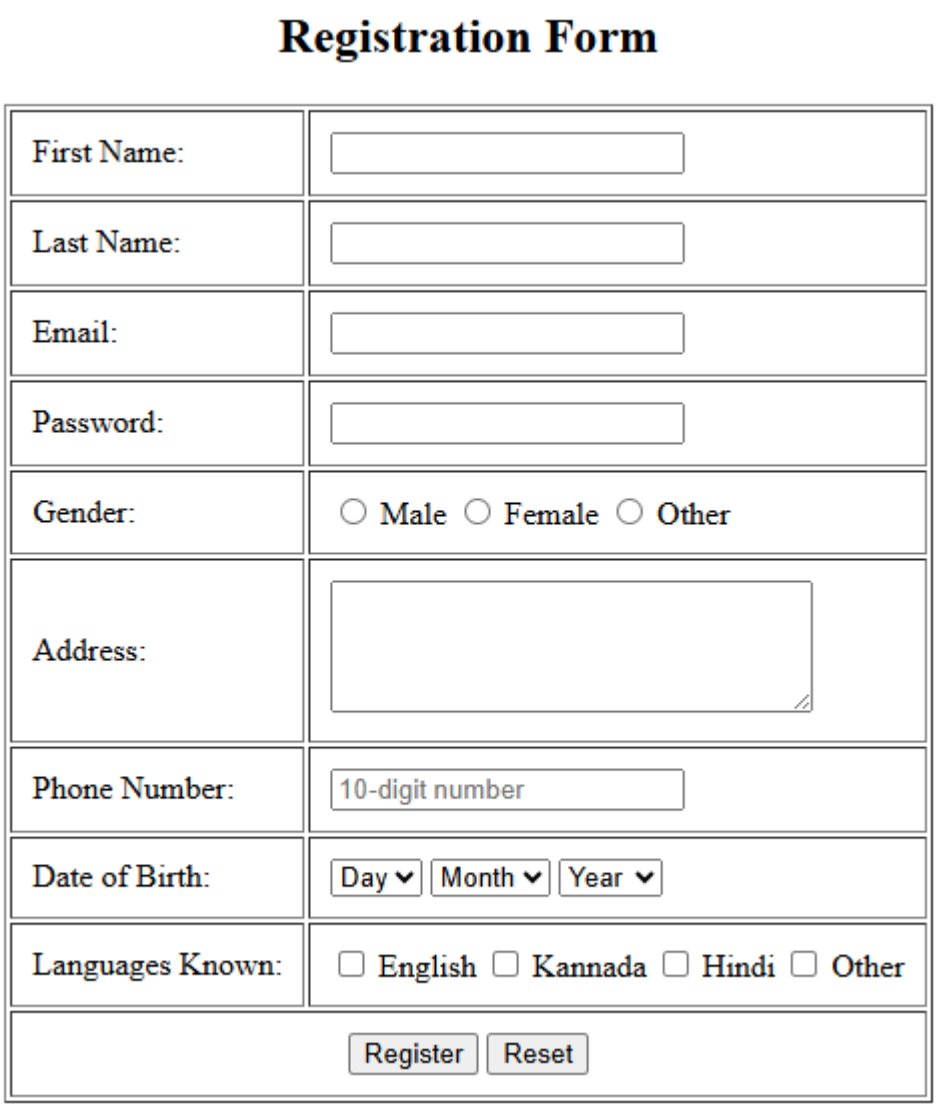
3. Design a Registration Page using HTML Forms:
- Include fields: First Name, Last Name, Email, Password, Gender (radio), Address (textarea), Phone Number, Date of Birth (select boxes), Languages Known (checkboxes)
- Use <form> tag with proper structure and validations through HTML attributes (like 'required').

```
<html>
<head>
  <title>Registration Page</title>
</head>
<body>
  <h2 align="center">Registration Form</h2>
  <form action="submit_registration.html" method="post">
    <table align="center" cellpadding="10" border="1">
      <tr>
        <td><label for="fname">First Name:</label></td>
        <td><input type="text" id="fname" name="fname" required></td>
      </tr>
      <tr>
        <td><label for="lname">Last Name:</label></td>
        <td><input type="text" id="lname" name="lname" required></td>
      </tr>
      <tr>
        <td><label for="email">Email:</label></td>
        <td><input type="email" id="email" name="email" required></td>
      </tr>
      <tr>
        <td><label for="password">Password:</label></td>
        <td><input type="password" id="password" name="password" required></td>
      </tr>
      <tr>
        <td>Gender:</td>
        <td>
          <input type="radio" name="gender" value="Male" required> Male
          <input type="radio" name="gender" value="Female"> Female
          <input type="radio" name="gender" value="Other"> Other
        </td>
      </tr>
      <tr>
        <td><label for="address">Address:</label></td>
        <td><textarea id="address" name="address" rows="4" cols="30"
required></textarea></td>
      </tr>
      <tr>
        <td><label for="phone">Phone Number:</label></td>
        <td><input type="tel" id="phone" name="phone" pattern="[0-9]{10}" required
placeholder="10-digit number"></td>
      </tr>
      <tr>
        <td>Date of Birth:</td>
```

```
    <td>
      <select name="day" required>
        <option value="">Day</option>
        <!-- Days 1 to 31 -->
        <script>
          for (let i = 1; i <= 31; i++) {
            document.write('<option value="' + i + '">' + i + '</option>');
          }
        </script>
      </select>
      <select name="month" required>
        <option value="">Month</option>
        <option value="Jan">Jan</option>
        <option value="Feb">Feb</option>
        <option value="Mar">Mar</option>
        <option value="Apr">Apr</option>
        <option value="May">May</option>
        <option value="Jun">Jun</option>
        <option value="Jul">Jul</option>
        <option value="Aug">Aug</option>
        <option value="Sep">Sep</option>
        <option value="Oct">Oct</option>
        <option value="Nov">Nov</option>
        <option value="Dec">Dec</option>
      </select>
      <select name="year" required>
        <option value="">Year</option>
        <!-- Years 1980 to 2020 -->
        <script>
          for (let y = 1980; y <= 2020; y++) {
            document.write('<option value="' + y + '">' + y + '</option>');
          }
        </script>
      </select>
    </td>
  </tr>

  <tr>
    <td>Languages Known:</td>
    <td>
      <input type="checkbox" name="lang" value="English"> English
      <input type="checkbox" name="lang" value="Kannada"> Kannada
      <input type="checkbox" name="lang" value="Hindi"> Hindi
      <input type="checkbox" name="lang" value="Other"> Other
    </td>
  </tr>
  <tr>
    <td colspan="2" align="center">
      <input type="submit" value="Register">
      <input type="reset" value="Reset">
    </td>
```

```
        </tr>
      </table>
    </form>
  </body>
  </html>
```

OUTPUT:

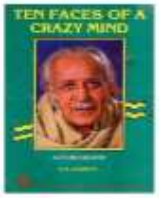
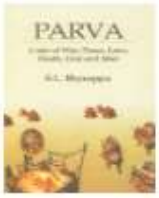


4. Develop a Catalogue Page to display book listings with a table:
   - Use <table> tag with headers: Book Image, Title, Author, Price, and 'Add to Cart' button
   - Use <img> tag for book covers
   - Use <input type='button'> for the cart

```
<html>
<head>
  <title>Book Catalogue</title>
</head>
<body>
  <h2>Book Catalogue</h2>
  <table border="1" cellpadding="10">
    <tr>
      <th>Book Image</th>
      <th>Title</th>
      <th>Author</th>
      <th>Price</th>
      <th>Add to Cart</th>
    </tr>
    <tr>
      <td><img src="C:\Users\Arun Kote\Downloads\JK.jpg" alt="Book 1" width="80" height="100"></td>
      <td>Think about these Things</td>
      <td>Jiddu Krishnamurthy</td>
      <td>₹300</td>
      <td><input type="button" value="Add to Cart"></td>
    </tr>
    <tr>
      <td><img src="C:\Users\Arun Kote\Downloads\SK.jpg" alt="Book 2" width="80" height="100"></td>
      <td>Ten faces of a crazy mind</td>
      <td>Shivram Karanth</td>
      <td>₹450</td>
      <td><input type="button" value="Add to Cart"></td>
    </tr>
    <tr>
      <td><img src="C:\Users\Arun Kote\Downloads\SL.jpg" alt="Book 3" width="80" height="100"></td>
      <td>Parva</td>
      <td>S L Bhayrappa</td>
      <td>₹250</td>
      <td><input type="button" value="Add to Cart"></td>
    </tr>
  </table>
</body>
</html>
```

OUTPUT:

Book Catalogue

Book Image	Title	Author	Price	Add to Cart
	Think about these Things	Jiddu Krishnamurthy	₹300	Add to Cart
	Ten faces of a crazy mind	Shivram Karanth	₹450	Add to Cart
	Parva	S L Bhayrappa	₹250	Add to Cart

5. Create a Shopping Cart Page using HTML Tables:
 - Use <table> to display items in cart with columns: Book Name, Price, Quantity, Amount
 - Add a final row to show the total amount
 - Align data centrally using appropriate attributes.

```
<!DOCTYPE html>
<html>
<head>
  <title>Shopping Cart</title>
</head>
<body>
  <h2 align="center">Shopping Cart</h2>

  <table border="1" align="center" cellpadding="10">
    <tr>
      <th>Book Name</th>
      <th>Price (₹)</th>
      <th>Quantity</th>
      <th>Amount (₹)</th>
    </tr>

    <tr align="center">
      <td>HTML Made Easy</td>
      <td>300</td>
      <td>2</td>
```

```
      <td>600</td>
    </tr>

    <tr align="center">
      <td>Learning JavaScript</td>
      <td>450</td>
      <td>1</td>
      <td>450</td>
    </tr>
    <tr align="center">
      <td>Web Design Basics</td>
      <td>250</td>
      <td>3</td>
      <td>750</td>
    </tr>
    <!-- Total Row -->
    <tr align="center">
      <td colspan="3"><strong>Total Amount</strong></td>
      <td><strong>₹1800</strong></td>
    </tr>
  </table>
</body>
</html>
```

OUTPUT:

Shopping Cart

Book Name	Price (₹)	Quantity	Amount (₹)
HTML Made Easy	300	2	600
Learning JavaScript	450	1	450
Web Design Basics	250	3	750
Total Amount			**₹1800**

6. Create a Time Table for your class using HTML Tables:
   - Columns: Days (Mon-Sat), Periods (I to VII)
   - Include breaks using rowspan/colspan
   - Use <table>, <tr>, <th>, and <td> appropriately.

```
<html>
<head>
  <title>Class Time Table</title>
</head>
<body>
  <h2 align="center">Class Time Table</h2>

  <table border="1" align="center" cellpadding="10">
    <tr>
      <th>Period</th>
      <th>Monday</th>
      <th>Tuesday</th>
      <th>Wednesday</th>
      <th>Thursday</th>
      <th>Friday</th>
      <th>Saturday</th>
    </tr>

    <tr align="center">
      <td>I (9:00-10:00)</td>
      <td>Math</td>
      <td>English</td>
      <td>Physics</td>
      <td>Math</td>
      <td>Computer</td>
      <td>English</td>
    </tr>

    <tr align="center">
      <td>II (10:00-11:00)</td>
      <td>English</td>
      <td>Physics</td>
      <td>Math</td>
      <td>Computer</td>
      <td>English</td>
      <td>Physics</td>
    </tr>

    <tr align="center">
      <td>III (11:00-12:00)</td>
      <td>Computer</td>
      <td>Math</td>
      <td>English</td>
      <td>Physics</td>
      <td>Math</td>
```

```
      <td>Computer</td>
    </tr>

    <!-- Break Row -->
    <tr align="center">
      <td colspan="7"><strong>Lunch Break (12:00 -
1:00)</strong></td>
    </tr>

    <tr align="center">
      <td>IV (1:00-2:00)</td>
      <td>Physics</td>
      <td>Computer</td>
      <td>Math</td>
      <td>English</td>
      <td>Physics</td>
      <td>--</td>
    </tr>

    <tr align="center">
      <td>V (2:00-3:00)</td>
      <td>English</td>
      <td>Math</td>
      <td>Computer</td>
      <td>Math</td>
      <td>Computer</td>
      <td>--</td>
    </tr>

    <tr align="center">
      <td>VI (3:00-4:00)</td>
      <td>Lab</td>
      <td>Lab</td>
      <td>Lab</td>
      <td>Lab</td>
      <td>Lab</td>
      <td>--</td>
    </tr>

    <tr align="center">
      <td>VII (4:00-5:00)</td>
      <td>Lab</td>
      <td>Lab</td>
      <td>Lab</td>
      <td>Lab</td>
      <td>Lab</td>
      <td>--</td>
    </tr>
  </table>
</body>
</html>
```

OUTPUT:

Class Time Table

Period	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday
I (9:00-10:00)	Math	English	Physics	Math	Computer	English
II (10:00-11:00)	English	Physics	Math	Computer	English	Physics
III (11:00-12:00)	Computer	Math	English	Physics	Math	Computer
Lunch Break (12:00 - 1:00)						
IV (1:00-2:00)	Physics	Computer	Math	English	Physics	--
V (2:00-3:00)	English	Math	Computer	Math	Computer	--
VI (3:00-4:00)	Lab	Lab	Lab	Lab	Lab	--
VII (4:00-5:00)	Lab	Lab	Lab	Lab	Lab	--

7. Develop a Web Page using Ordered, Unordered and Definition Lists:
   - Use <ol>, <ul>, <li>, and <dl>, <dt>, <dd>
   - Provide appropriate headings and content for each list type
   - Include at least one nested list.

```
<html>
<head>
  <title>List Example Page</title>
</head>
<body>
  <h2>1. Ordered List - Steps to Create a Web Page</h2>
  <ol>
    <li>Open Notepad</li>
    <li>Write HTML Code</li>
    <li>Save with .html extension</li>
    <li>Open in Web Browser</li>
  </ol>

  <h2>2. Unordered List - Web Development Tools</h2>
  <ul>
    <li>VS Code</li>
    <li>Notepad++</li>
    <li>Brackets</li>
    <li>Sublime Text</li>
  </ul>

  <h2>3. Nested List - Web Technologies</h2>
  <ul>
    <li>Frontend
      <ul>
        <li>HTML</li>
        <li>CSS</li>
        <li>JavaScript</li>
      </ul>
    </li>
    <li>Backend
      <ul>
        <li>PHP</li>
        <li>Python</li>
        <li>Node.js</li>
      </ul>
    </li>
  </ul>

  <h2>4. Definition List - HTML Terminology</h2>
  <dl>
    <dt>HTML</dt>
    <dd>HyperText Markup Language used to create web pages.</dd>

    <dt>CSS</dt>
```

```
    <dd>Cascading Style Sheets used for designing web pages.</dd>

    <dt>JavaScript</dt>
    <dd>A scripting language used to make web pages interactive.</dd>
  </dl>
</body>
</html>
```

OUTPUT:

1. Ordered List - Steps to Create a Web Page

1. Open Notepad
2. Write HTML Code
3. Save with .html extension
4. Open in Web Browser

2. Unordered List - Web Development Tools

- VS Code
- Notepad++
- Brackets
- Sublime Text

3. Nested List - Web Technologies

- Frontend
  - HTML
  - CSS
  - JavaScript
- Backend
  - PHP
  - Python
  - Node.js

4. Definition List - HTML Terminology

HTML
: HyperText Markup Language used to create web pages.

CSS
: Cascading Style Sheets used for designing web pages.

JavaScript
: A scripting language used to make web pages interactive.

8. Create a Navigation Web Page using Hyperlinks (internal and external):
   - Use <a> tag to create links
   - Include one internal (within the page using #id) and one external (another file) link
   - Apply link styling using 'alink', 'vlink', and 'link' attributes.

```
<html>
<head>
  <title>Navigation Page</title>
</head>
<body link="blue" alink="red" vlink="green">

  <h2>Welcome to My Navigation Page</h2>

  <!-- External Link -->
  <p>
    Visit <a href="https://www.google.com" target="_blank">Google</a> (External Link)
  </p>

  <!-- Internal Link -->
  <p>
    Go to <a href="G:\PHONES\MEMORIES\Twitter\CH.jpg">Section 2</a> (Internal Link)
  </p>

  <hr>

  <!-- Some Content -->
  <h3 id="section1">Section 1: About Google</h3>
  <p>This is the first section of the page. It talks about web browser google.</p>

  <h3 id="section2">Section 2: My Projects</h3>
        <p>This section includes information about my Favourite cartoon.</p>

  </body>
</html>
```

OUTPUT:

Welcome to My Navigation Page

Visit Google (External Link)

Go to Section 2 (Internal Link)

Section 1: About Google

This is the first section of the page. It talks about web browser google

Section 2: My Projects

This section includes information about my favourite cartoon.

PART B – JavaScript Programs (Dynamic Interaction)

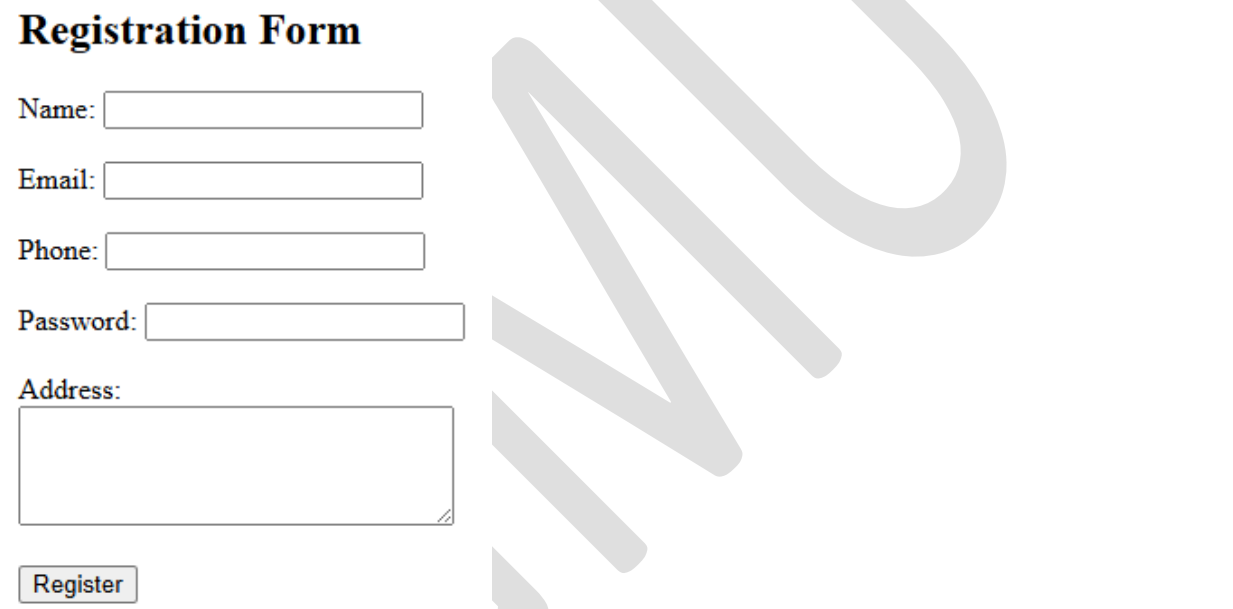
1. Validate Registration Form using JavaScript:

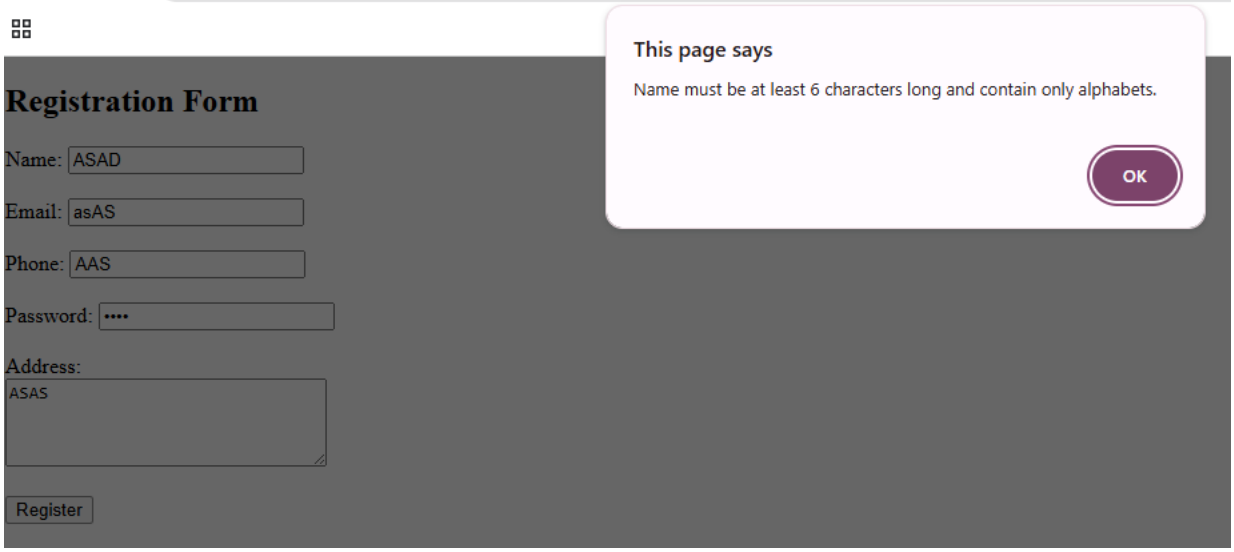
- Validate: Name (min 6 chars, alphabets only), Email (standard pattern), Phone (10 digits), Password (min 6 chars), Empty check for Address
- Use alert to show error messages
- Trigger on form submission.

```
<!DOCTYPE html>
<html>
<head>
  <title>Registration Form Validation</title>
  <script>
    function validateForm() {
      var name = document.forms["regForm"]["name"].value;
      var email = document.forms["regForm"]["email"].value;
      var phone = document.forms["regForm"]["phone"].value;
      var password = document.forms["regForm"]["password"].value;
      var address = document.forms["regForm"]["address"].value;

      // Name validation (min 6 chars, alphabets only)
      var namePattern = /^[A-Za-z ]{6,}$/;
      if (!namePattern.test(name)) {
        alert("Name must be at least 6 characters long and contain only alphabets.");
        return false;
      }

      // Email validation
      var emailPattern = /^[^ ]+@[^ ]+\.[a-z]{2,3}$/;
      if (!emailPattern.test(email)) {
        alert("Enter a valid email address.");
        return false;
      }

      // Phone number validation (10 digits)
      var phonePattern = /^[0-9]{10}$/;
      if (!phonePattern.test(phone)) {
        alert("Phone number must be exactly 10 digits.");
        return false;
      }

      // Password validation (min 6 chars)
      if (password.length < 6) {
        alert("Password must be at least 6 characters long.");
        return false;
      }

      // Address empty check
      if (address.trim() === "") {
        alert("Address cannot be empty.");
        return false;
      }

      // All validations passed
```

```
        alert("Form submitted successfully!");
        return true;
      }
    </script>
</head>
<body>
    <h2>Registration Form</h2>
    <form name="regForm" onsubmit="return validateForm()">
      Name: <input type="text" name="name"><br><br>
      Email: <input type="text" name="email"><br><br>
      Phone: <input type="text" name="phone"><br><br>
      Password: <input type="password" name="password"><br><br>
      Address:<br>
      <textarea name="address" rows="4" cols="30"></textarea><br><br>
      <input type="submit" value="Register">
    </form>
</body>
</html>
```

OUTPUT:





2. Validate Login Form using JavaScript:
 - Validate Username and Password fields (non-empty, basic length check)
 - Show error if any field is empty
 - Use 'onsubmit' or button 'onclick' for validation.

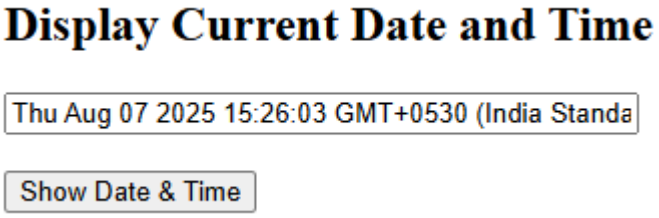
```
<!DOCTYPE html>
<html>
<head>
 <title>Login Validation</title>
 <script>
  function validateLoginForm() {
   var username = document.getElementById("username").value;
   var password = document.getElementById("password").value;

   if (username === "") {
    alert("Please enter your username.");
    return false;
   }
   if (username.length < 4) {
    alert("Username must be at least 4 characters.");
    return false;
   }
   if (password === "") {
    alert("Please enter your password.");
    return false;
   }
   if (password.length < 6) {
    alert("Password must be at least 6 characters.");
    return false;
   }

   alert("Login Successful!");
   return true;
  }
 </script>
</head>
<body>
 <h2>Login Form</h2>
 <form>
  <label>Username:</label>
  <input type="text" id="username"><br><br>

  <label>Password:</label>
  <input type="password" id="password"><br><br>

  <button type="button" onclick="validateLoginForm()">Login</button>
 </form>
</body>
</html>
```

OUTPUT:

3. Use JavaScript to display current Date and Time in a text box:
 - Create an <input> of type text
 - Add a button to trigger JavaScript function
 - Use new Date() to get current date and time and display in textbox.

```
<!DOCTYPE html>
<html>
<head>
 <title>Show Date and Time</title>
 <script>
  function showDateTime() {
   var current = new Date(); // Get current date and time
   document.getElementById("datetimeBox").value = current;
  }
 </script>
</head>
<body>
 <h2>Display Current Date and Time</h2>

 <input type="text" id="datetimeBox" size="40" readonly>
 <br><br>
 <button onclick="showDateTime()">Show Date & Time</button>

</body>
</html>
```

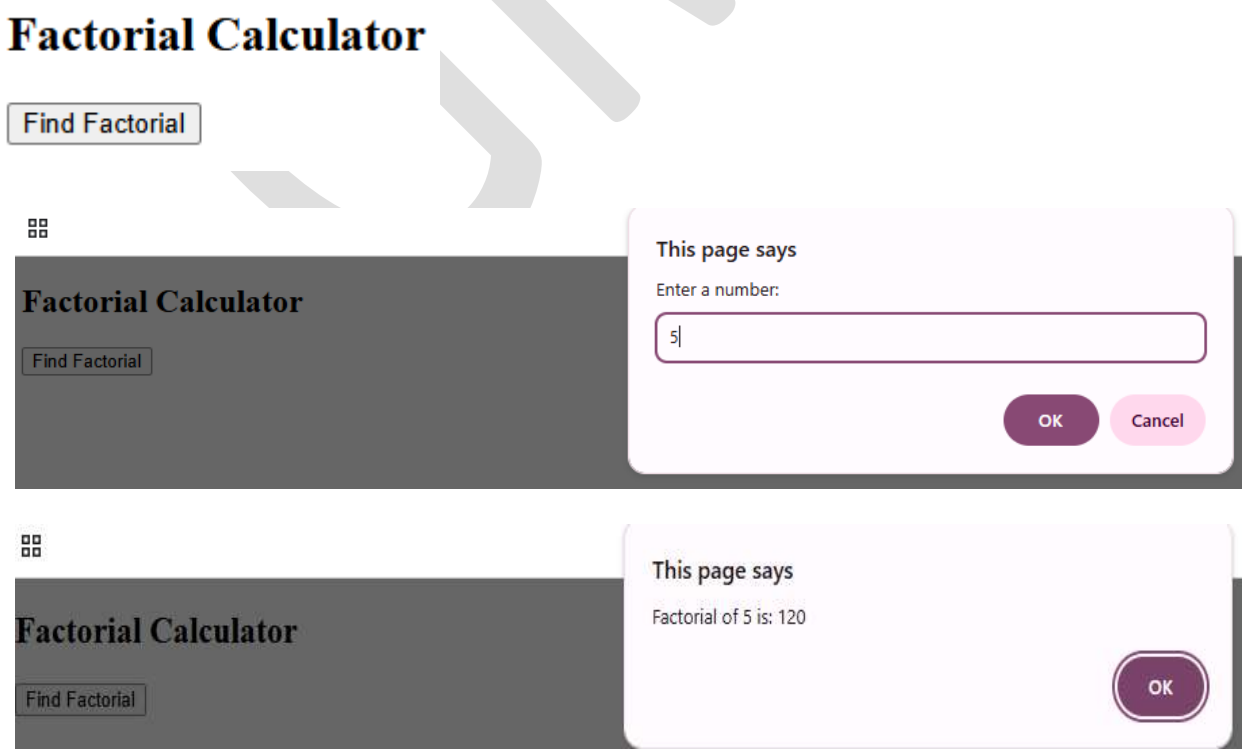
OUTPUT:



4. Create a JavaScript program to find factorial using prompt() and alert():
- Use prompt() to get user input
- Calculate factorial using loop
- Display result using alert().

```
<html>
<head>
 <title>Factorial using JavaScript</title>
 <script>
  function findFactorial() {
   var num = prompt("Enter a number:");  // Get user input
   num = parseInt(num);  // Convert to integer

   if (isNaN(num) || num < 0) {
    alert("Please enter a valid non-negative number.");
    return;
   }

   var fact = 1;
   for (var i = 1; i <= num; i++) {
    fact *= i;
   }

   alert("Factorial of " + num + " is: " + fact);
  }
 </script>
</head>
<body>
 <h2>Factorial Calculator</h2>
 <button onclick="findFactorial()">Find Factorial</button>
</body>
</html>
```

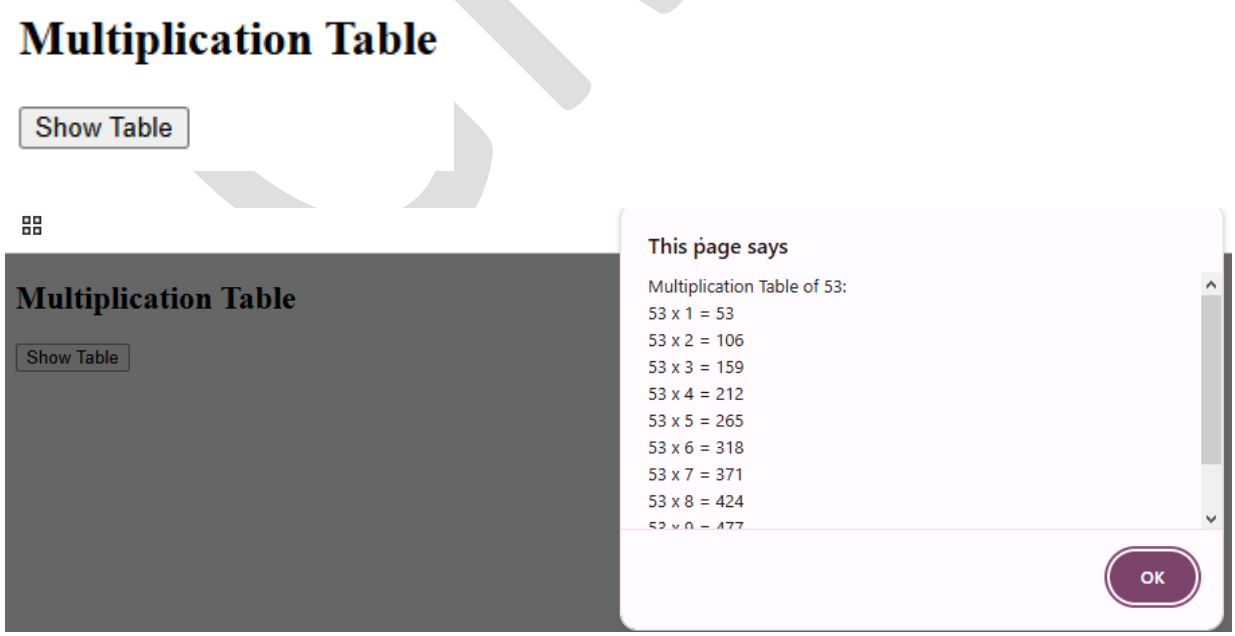
OUTPUT:



5. Develop a script to display a multiplication table using JavaScript:
- Take a number input via prompt()
- Use loop to generate table (1 to 10)
- Display results using alert() or with innerHTML.

```
<html>
<head>
 <title>Multiplication Table</title>
 <script>
  function showTable() {
   var num = prompt("Enter a number:");
   num = parseInt(num);

   if (isNaN(num)) {
    alert("Please enter a valid number.");
    return;
   }

   var result = "Multiplication Table of " + num + ":\n";
   for (var i = 1; i <= 10; i++) {
    result += num + " x " + i + " = " + (num * i) + "\n";
   }

   alert(result);
  }
 </script>
</head>
<body>
 <h2>Multiplication Table</h2>
 <button onclick="showTable()">Show Table</button>
</body>
</html>
```

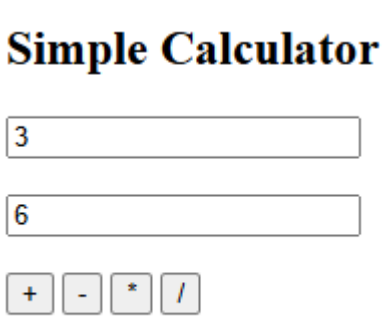
OUTPUT:



6. Create a simple calculator using JavaScript:
- Use <input> fields to get two numbers
- Add buttons for '+', '-', '*', '/'
- On button click, perform respective operation and show result using alert() or in a .

```
<html>
<head>
 <title>Simple Calculator</title>
 <script>
  function calculate(op) {
   var num1 = parseFloat(document.getElementById("num1").value);
   var num2 = parseFloat(document.getElementById("num2").value);
   var result;

   if (isNaN(num1) || isNaN(num2)) {
    alert("Please enter valid numbers");
    return;
   }

   if (op === '+') {
    result = num1 + num2;
   } else if (op === '-') {
    result = num1 - num2;
   } else if (op === '*') {
    result = num1 * num2;
   } else if (op === '/') {
    if (num2 === 0) {
     alert("Cannot divide by zero");
     return;
    }
    result = num1 / num2;
   }

   document.getElementById("output").innerHTML = "Result: " + result;
  }
 </script>
</head>
<body>
 <h2>Simple Calculator</h2>
 <input type="text" id="num1" placeholder="Enter first number"> <br><br>
 <input type="text" id="num2" placeholder="Enter second number"> <br><br>

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

 <p id="output"></p>
</body>
</html>
```

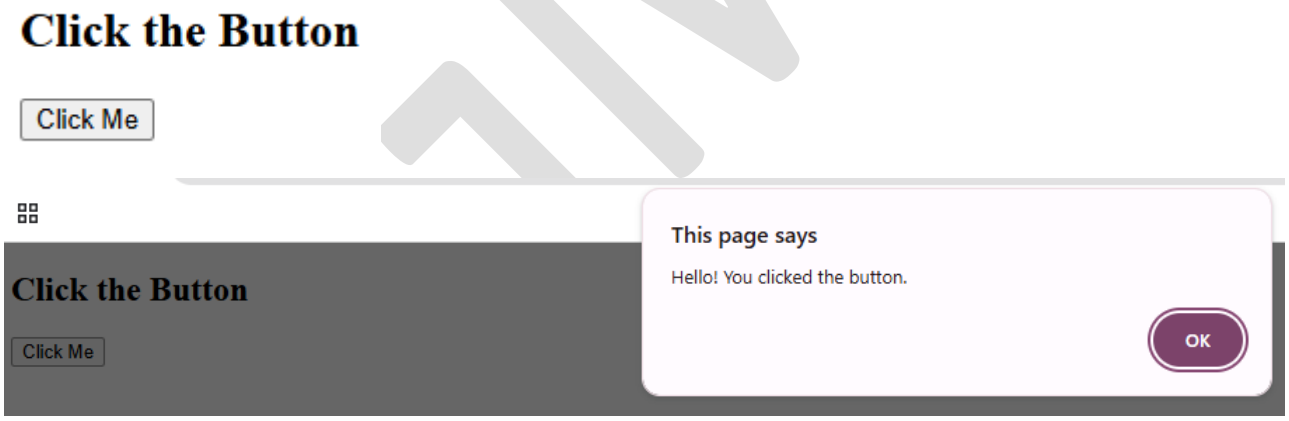
OUTPUT:



7. Use JavaScript to display a message when a button is clicked (onclick):
 - Create a button using <button> or <input type='button'>
 - Use onclick event handler to display a custom message using alert() or innerHTML.

```
<html>
<head>
 <title>Button Click Message</title>
 <script>
  function showMessage() {
   alert("Hello! You clicked the button.");
  }
 </script>
</head>
<body>
 <h2>Click the Button</h2>
 <button onclick="showMessage()">Click Me</button>
</body>
</html>
```

OUTPUT:



8. **Create an Interactive Product Showcase Web Page"**

Requirements:

HTML

- Display at least 3 products in a grid layout.
- Each product should have an image, name, price, and a short description.
- Add an “Add to Cart” button for each product.

CSS

- Style the layout to be visually appealing (use colors, borders, hover effects).
- Make the page responsive using simple media queries.

JavaScript

- When the “Add to Cart” button is clicked:
  - Display a message like “Product added to cart” (using alert() or dynamically in the page).
  - Keep track of the number of items in the cart and update a counter at the top of the page without refreshing.

```
<html>
<head>
  <title>Simple Product Showcase</title>
</head>
<body>
<h1>Product Showcase</h1>
<!-- Cart Count -->
<p>Cart Items: <span id="cartCount">0</span></p>
<!-- Product 1 -->
<img src="C:\Users\Arun Kote\Downloads\HF.jpg" alt="Product 1" width="100" height="50"><br>
<b>Wireless Headphones</b><br>
Price: $49.99<br>
<button onclick="addToCart()">Add to Cart</button>
<hr>
<!-- Product 2 -->
<img src="C:\Users\Arun Kote\Downloads\SW.jpg" width="100" height="50" alt="Product 2"><br>
<b>Smart Watch</b><br>
Price: $89.99<br>
<button onclick="addToCart()">Add to Cart</button>
<hr>
<!-- Product 3 -->
<img src="C:\Users\Arun Kote\Downloads\BS.jpg" width="100" height="50" alt="Product 3"><br>
<b>Bluetooth Speaker</b><br>
Price: $39.99<br>
<button onclick="addToCart()">Add to Cart</button>
<hr>
<marquee behavior="scroll" direction="right">
   Happy Shopping!
</marquee>
<script>
let cartCount = 0;
function addToCart() {
  cartCount++; // increase count
  document.getElementById("cartCount").innerText = cartCount; // update on page
  alert("Product added to cart!");
}
</script>
</body>
</html>
```

OUTPUT:

Product Showcase

Cart Items: 0

Wireless Headphones
Price: $49.99
Add to Cart

Smart Watch
Price: $89.99
Add to Cart

Bluetooth Speaker
Price: $39.99
Add to Cart

Happy Shopping!