MAHARAJA SURAJMAL INSTITUTE

C-4, JanakPuri, New Delhi, 110058

DEPARTMENT OF COMPUTER APPLICATION

WEB TECHNOLOGY LAB

SUBJECT CODE-BCA-105P

Submitted by:

Name: Krishna
Enrollment No.: -
00214902020
Course: BCA-I-A
(Shift 1)

Submitted to:

Ms. Rhythm Choudhary
Assistant Professor
MSI

INDEX

12.	WAP to make use of JavaScript Operator: : I. Arithmetic Operators II. Assignment Operators III. Comparison Operators IV. Logical Operators V. Conditional Operators VI. Type Operators VII. Bitwise Operators	
13.	WAP to make use of following JavaScript Statements: i.if ii. else iii. else if iv. switch v. while vi. do while vii. for viii. for in ix. for of	
14.	WAP to show use of following object in JS. i.date ii.Math iii.string	
15.	WAP to show use of object in JavaScript. i. By object literal ii. By instance of object directly iii. By using object constructor	
16.	WAP to show Validation in JavaScript.	
17.	WAP to show usage of inbuilt functions.	
18	WAP to show usage of i. alert box ii. prompt box iii. comfirm box.	

19	WAP demonstrating i. Factorial of number ii. Prime number in a range iii. Reverse of a number	
20.	WAP to implement event handling using i. onclick ii. mouseover iii. mouseout iv. mousein v. doubleclick vi. mousemove	
21.	WAP in JavaScript to demonstrate i. Blur ii. Focus iii. Reset iv. Submit v. Change vi. load	
22.	WAP to implement event handling using i. keydown ii. keyup	
23.	WAP to show JS can change HTML content.	

24.	WAP to show JS can change HTML attribute value.	
25.	WAP to show JS can change HTML style.	
26.	WAP to show use of Window Object in JS. i. Navigation object ii. JavaScript Screen Object iii. Set-time function	
27.	Create an XML file containing note as the root element and to, from, subject and body as its child element.	
28.	Create a well-formed XML file having 5 tags-food menu, category, name, price and calories. Food_menu is the main parent element that contains category as the child element. Category is the parent element of name, price and calories.	
29.	WAP to display the bookstore details in XML with CSS and internal DTD.	
30.	Create Valid XML doc using external DTD.	

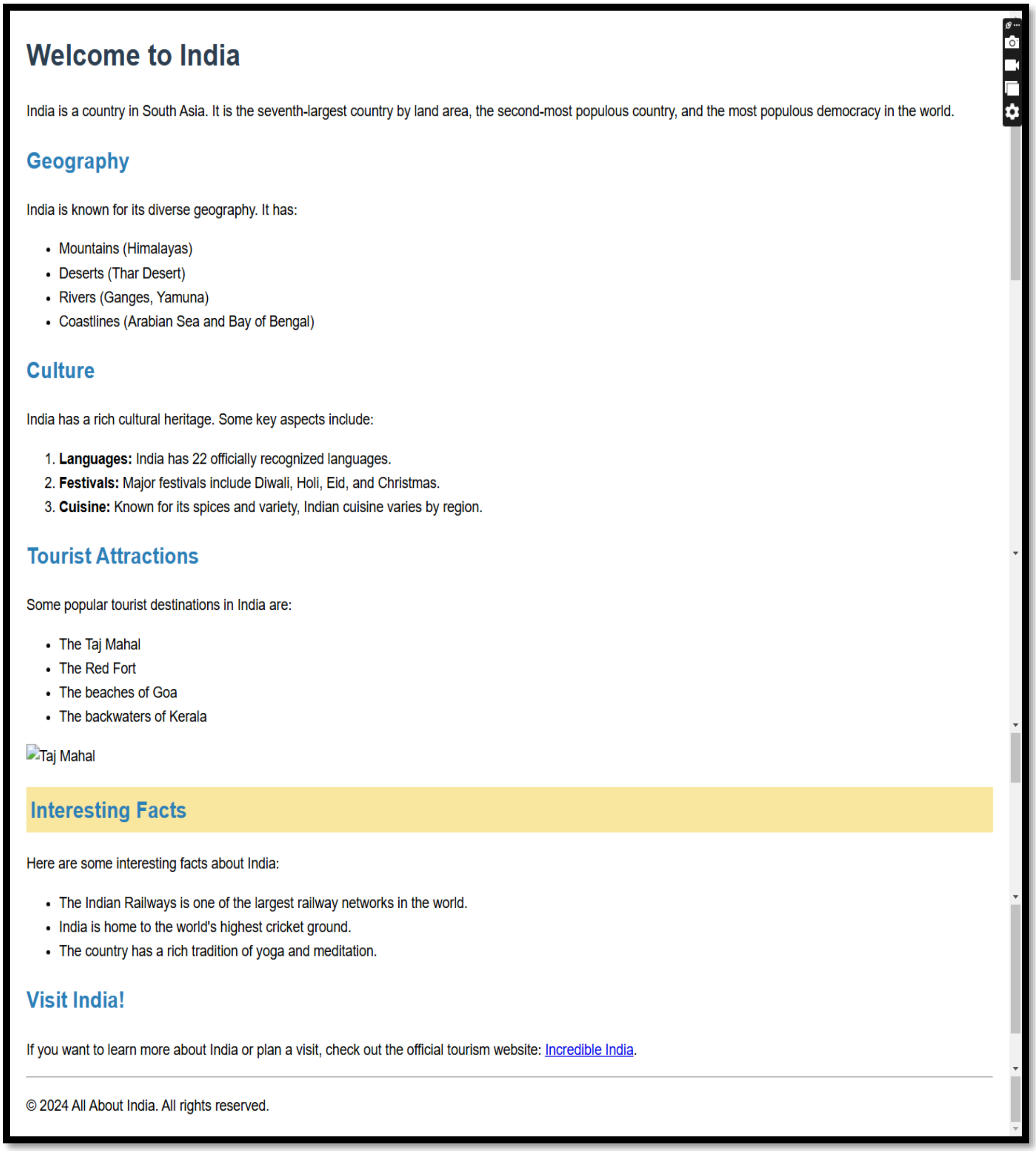
Program 1 : Create your Web Page on India using various basic HTML tags.

Index.html page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>All About India</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      line-height: 1.6;
      margin: 20px;
    }
    h1 {
      color: #2c3e50;
    }
    h2 {
      color: #2980b9;
    }
    .highlight {
      background-color: #f9e79f;
      padding: 5px;
    }
  </style>
</head>
<body>

<h1>Welcome to India</h1>

<p>India is a country in South Asia. It is the seventh-largest country by land area, the
second-most populous country, and the most populous democracy in the world.</p>

<h2>Geography</h2>
<p>India is known for its diverse geography. It has:</p>
<ul>
  <li>Mountains (Himalayas)</li>
  <li>Deserts (Thar Desert)</li>
  <li>Rivers (Ganges, Yamuna)</li>
  <li>Coastlines (Arabian Sea and Bay of Bengal)</li>
</ul>

<h2>Culture</h2>
<p>India has a rich cultural heritage. Some key aspects include:</p>
<ol>
```

```
    <li><strong>Languages:</strong> India has 22 officially recognized languages.</li>
    <li><strong>Festivals:</strong> Major festivals include Diwali, Holi, Eid, and
Christmas.</li>
    <li><strong>Cuisine:</strong> Known for its spices and variety, Indian cuisine varies by
region.</li>
</ol>

<h2>Tourist Attractions</h2>
<p>Some popular tourist destinations in India are:</p>
<ul>
    <li>The Taj Mahal</li>
    <li>The Red Fort</li>
    <li>The beaches of Goa</li>
    <li>The backwaters of Kerala</li>
</ul>

<img
src="https://upload.wikimedia.org/wikipedia/commons/thumb/4/41/Taj_Mahal%2C_Agr
a%2C_India.jpg/800px-Taj_Mahal%2C_Agra%2C_India.jpg" alt="Taj Mahal"
style="width:100%; max-width:600px;">

<h2 class="highlight">Interesting Facts</h2>
<p>Here are some interesting facts about India:</p>
<ul>
    <li>The Indian Railways is one of the largest railway networks in the world.</li>
    <li>India is home to the world's highest cricket ground.</li>
    <li>The country has a rich tradition of yoga and meditation.</li>
</ul>

<h2>Visit India!</h2>
<p>If you want to learn more about India or plan a visit, check out the official tourism
website:
   <a href="https://www.incredibleindia.org" target="_blank">Incredible India</a>.
</p>

<hr>

<footer>
    <p>&copy; 2024 All About India. All rights reserved.</p>
</footer>

</body>
</html>
```

Output



Welcome to India

India is a country in South Asia. It is the seventh-largest country by land area, the second-most populous country, and the most populous democracy in the world.

Geography

India is known for its diverse geography. It has:

- Mountains (Himalayas)
- Deserts (Thar Desert)
- Rivers (Ganges, Yamuna)
- Coastlines (Arabian Sea and Bay of Bengal)

Culture

India has a rich cultural heritage. Some key aspects include:

1. **Languages:** India has 22 officially recognized languages.
2. **Festivals:** Major festivals include Diwali, Holi, Eid, and Christmas.
3. **Cuisine:** Known for its spices and variety, Indian cuisine varies by region.

Tourist Attractions

Some popular tourist destinations in India are:

- The Taj Mahal
- The Red Fort
- The beaches of Goa
- The backwaters of Kerala

Taj Mahal

Interesting Facts

Here are some interesting facts about India:

- The Indian Railways is one of the largest railway networks in the world.
- India is home to the world's highest cricket ground.
- The country has a rich tradition of yoga and meditation.

Visit India!

If you want to learn more about India or plan a visit, check out the official tourism website: Incredible India.

© 2024 All About India. All rights reserved.

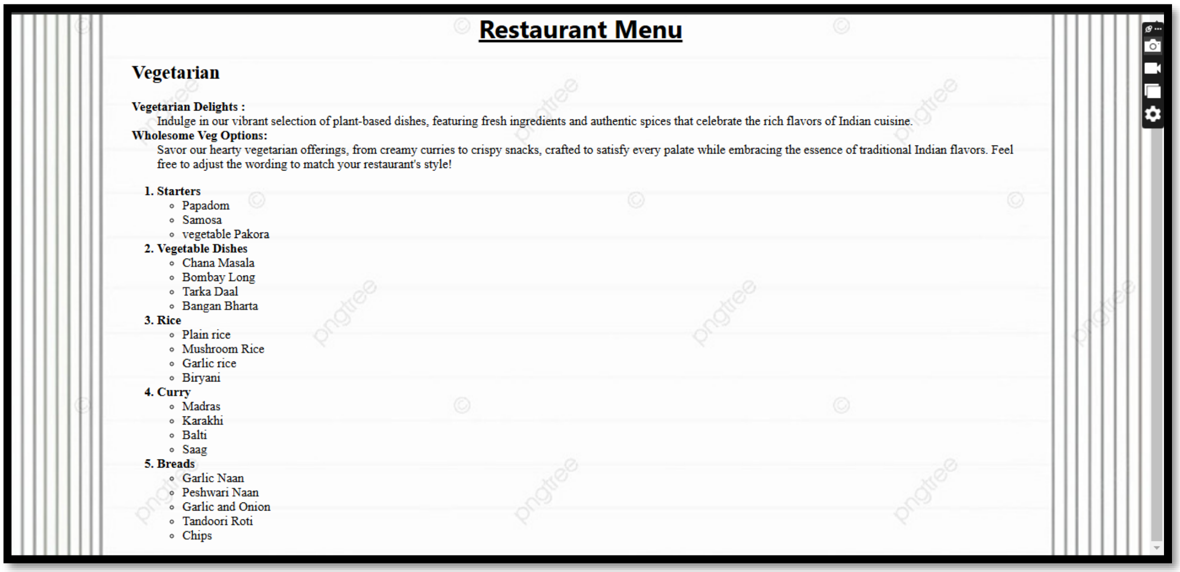
Program 2: Create an Restaurant Menu That use Ordered list, Unorder List, definition list, Nested list.

Index.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Restaurant menu</title>
  <style>
    body{
      background-image:
url(https://png.pngtree.com/thumb_back/fw800/background/20230927/pngtree-white-grey-minimalist-creative-page-border-image_13370233.jpg);
      background-size: cover;
      background-position: center center;
      background-repeat: no-repeat;
      font-size:20px;
    }
    .content{
      margin-left: 10vw;
      margin-right: 10vw;
    }
  </style>
</head>
<body>
  <h1 align="center">
    <font style="font-family:'Segoe UI';"><u>Restaurant Menu </u></font>
  </h1>
  <div class="content">
    <h2>Vegetarian</h2>
    <dl>
      <B><dt>Vegetarian Delights : </dt></B>
      <dd>Indulge in our vibrant selection of plant-based dishes, featuring fresh
ingredients and authentic spices that celebrate the rich flavors of Indian cuisine.</dd>
      <B><dt>Wholesome Veg Options:</dt></B>
      <dd>Savor our hearty vegetarian offerings, from creamy curries to crispy snacks,
crafted to satisfy every palate while embracing the essence of traditional Indian
flavors.</dd>
    </dl>
    <ol>
      <B><li>Starters</li></B>
      <ul>
```

```
        <li>Papadom</li>
        <li>Samosa</li>
        <li>vegetable Pakora</li>
      </ul>
      <B><li>Vegetable Dishes</li></B>
      <ul>
        <li>Chana Masala</li>
        <li>Bombay Long</li>
        <li>Tarka Daal</li>
        <li>Bangan Bharta</li>
      </ul>
      <B><li>Rice</li></B>
      <ul>
        <li>Plain rice</li>
        <li>Mushroom Rice</li>
        <li>Garlic rice</li>
        <li>Biryani</li>
      </ul>
      <B><li>Curry</li></B>
      <ul>
        <li>Madras</li>
        <li>Karakhi</li>
        <li>Balti</li>
        <li>Saag</li>
      </ul>
      <B><li>Breads</li></B>
      <ul>
        <li>Garlic Naan</li>
        <li>Peshwari Naan</li>
        <li>Garlic and Onion</li>
        <li>Tandoori Roti</li>
        <li>Chips</li>
      </ul>
    </ol>
  </div>
</body>
</html>
```

Output



Restaurant Menu

Vegetarian

Vegetarian Delights :
Indulge in our vibrant selection of plant-based dishes, featuring fresh ingredients and authentic spices that celebrate the rich flavors of Indian cuisine.
Wholesome Veg Options:
Savor our hearty vegetarian offerings, from creamy curries to crispy snacks, crafted to satisfy every palate while embracing the essence of traditional Indian flavors. Feel free to adjust the wording to match your restaurant's style!

1. **Starters**
   - Papadom
   - Samosa
   - vegetable Pakora
2. **Vegetable Dishes**
   - Chana Masala
   - Bombay Long
   - Tarka Daal
   - Bangan Bharta
3. **Rice**
   - Plain rice
   - Mushroom Rice
   - Garlic rice
   - Biryani
4. **Curry**
   - Madras
   - Karakhi
   - Balti
   - Saag
5. **Breads**
   - Garlic Naan
   - Peshwari Naan
   - Garlic and Onion
   - Tandoori Roti
   - Chips

Program 3: Design a table and make use of following attributes: colspan, rowspan, thead, tbody, tfoot, width, height, cellpadding, cellspacing etc.

Index.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Tables</title>
</head>
<body>
  <h1 align="center">TABLE AND ITS ATTRIBUTES</h1>
  <center><table width="80%" border="2px" cellpadding="10px">
    <colgroup>
      <col span="5" style="background-color: aliceblue;">
      <col style="background: greenyellow;">
    </colgroup>
    <thead style="background-color: aqua;">
      <tr>
        <th rowspan="2">NAME</th>
        <th colspan="5">SUBJECTS</th>
      </tr>
      <tr>
        <th>English</th>
        <th>Hindi</th>
        <th>Maths</th>
        <th>Science</th>
        <th>Total Percentage</th>
      </tr>
    </thead>
    <tbody style="background-color: rgb(230, 230, 230);">
      <tr>
        <td>Alpha</td>
        <td>87</td>
        <td>89</td>
        <td>90</td>
        <td>84</td>
        <td>87.5</td>
      </tr>
      <tr>
        <td>Beta</td>
        <td>87</td>
        <td>60</td>
        <td>74</td>
        <td>56</td>
        <td>69.25</td>
      </tr>
      <tr>
        <td>Charlie</td>
        <td>85</td>
        <td>96</td>
        <td>85</td>
        <td>79</td>
        <td>86.25</td>
      </tr>
      <tr>
        <td>Delta</td>
        <td>91</td>
        <td>88</td>
        <td>79</td>
        <td>88</td>
        <td>86.5</td>
      </tr>
    </tbody>
    <tfoot style="background-color: greenyellow;">
      <td colspan="6">Highest Marks : &lt; Alpha &gt; : &lt; 87.5% &gt; : &lt; Prize : $10 &gt;</td>
    </tfoot>
  </table></center>
</body>
</html>
```

Output

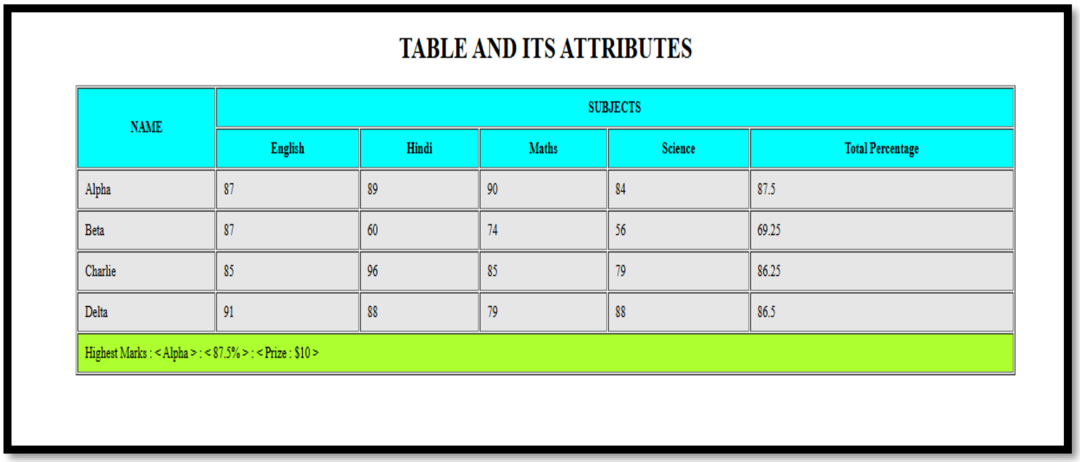


TABLE AND ITS ATTRIBUTES

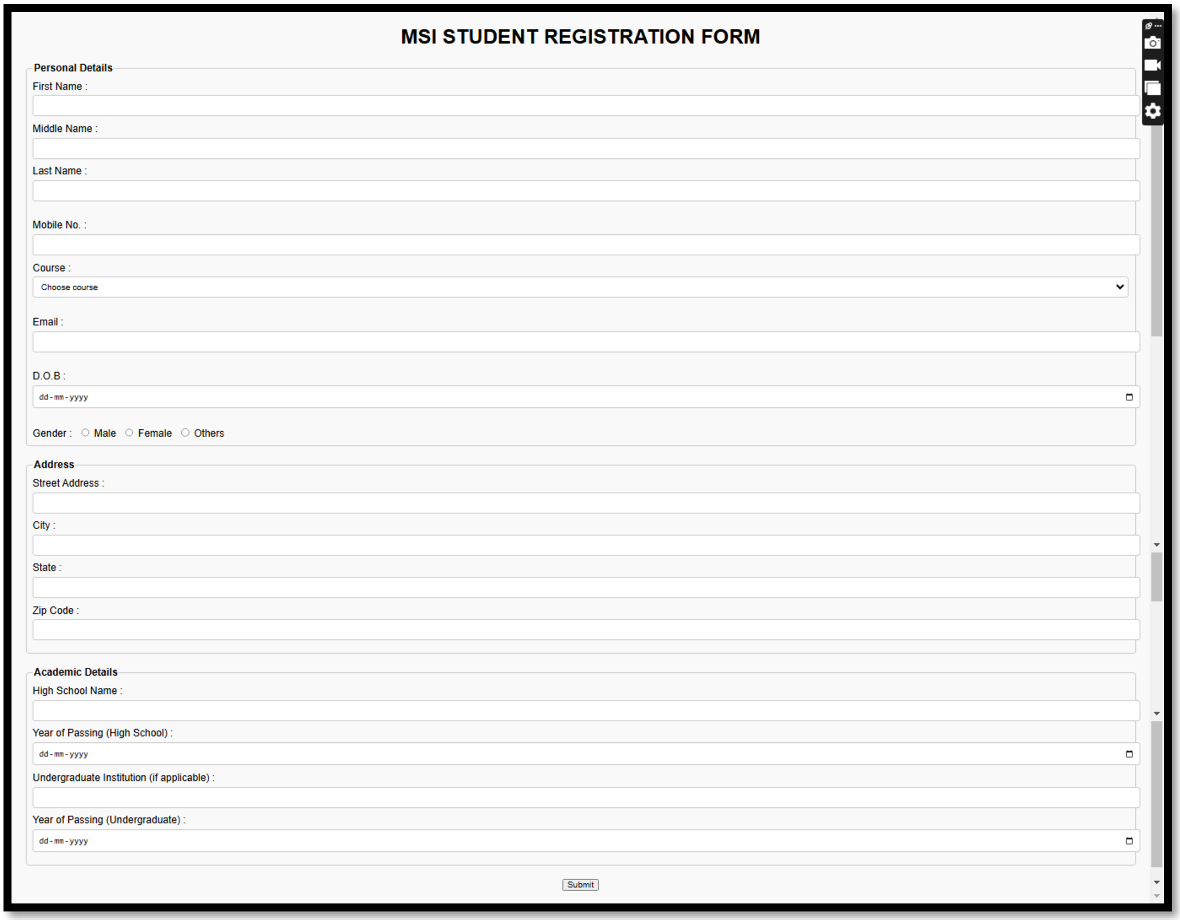
NAME	SUBJECTS				
	English	Hindi	Maths	Science	Total Percentage
Alpha	87	89	90	84	87.5
Beta	87	60	74	56	69.25
Charlie	85	96	85	79	86.25
Delta	91	88	79	88	86.5
Highest Marks : < Alpha > : < 87.5% > : < Prize : $10 >					

Program 4: Create Student registration form for admission in college.

Index.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>MSI Student Registration Form</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f9f9f9;
      margin: 20px;
    }
    fieldset {
      margin-bottom: 20px;
      padding: 10px;
      border-radius: 5px;
      border: 1px solid #ccc;
    }
    legend {
      font-weight: bold;
    }
    input[type="text"], input[type="email"], input[type="date"], select {
      width: 100%;
      padding: 8px;
      margin-top: 5px;
      margin-bottom: 10px;
      border-radius: 4px;
      border: 1px solid #ccc;
    }
    input[type="radio"] {
      margin-left: 10px;
    }
  </style>
</head>
<body>
  <h1 align="center">MSI STUDENT REGISTRATION FORM</h1>
  <form>
    <fieldset>
      <legend>Personal Details</legend>
      First Name : <input type="text" required><br>
      Middle Name : <input type="text"><br>
      Last Name : <input type="text"><br><br>
      Mobile No. : <input type="text" required><br>
      Course :
      <select required>
        <option disabled selected>Choose course</option>
        <option value="BBA">BBA</option>
        <option value="BCA">BCA</option>
        <option value="B.Tech">B.Tech</option>
        <option value="MBA">MBA</option>
      </select><br><br>
      Email : <input type="email" required><br><br>
      D.O.B : <input type="date" required><br><br>
      Gender :
      <input type="radio" name="gen" value="Male"> Male
      <input type="radio" name="gen" value="Female"> Female
      <input type="radio" name="gen" value="Others"> Others
    </fieldset>

    <fieldset>
      <legend>Address</legend>
      Street Address : <input type="text" required><br>
      City : <input type="text" required><br>
      State : <input type="text" required><br>
      Zip Code : <input type="text" required><br>
    </fieldset>

    <fieldset>
      <legend>Academic Details</legend>
      High School Name : <input type="text" required><br>
      Year of Passing (High School) : <input type="date" required><br>
      Undergraduate Institution (if applicable) : <input type="text"><br>
      Year of Passing (Undergraduate) : <input type="date"><br>
    </fieldset>

    <div align="center">
      <button type="submit">Submit</button>
    </div>
  </form>
</body>
</html>
```

Output



MSI STUDENT REGISTRATION FORM

Personal Details

First Name :

Middle Name :

Last Name :

Mobile No. :

Course :

Choose course

Email :

D.O.B :

dd-mm-yyyy

Gender : ○ Male ○ Female ○ Others

Address

Street Address :

City :

State :

Zip Code :

Academic Details

High School Name :

Year of Passing (High School) :

dd-mm-yyyy

Undergraduate Institution (if applicable) :

Year of Passing (Undergraduate) :

dd-mm-yyyy

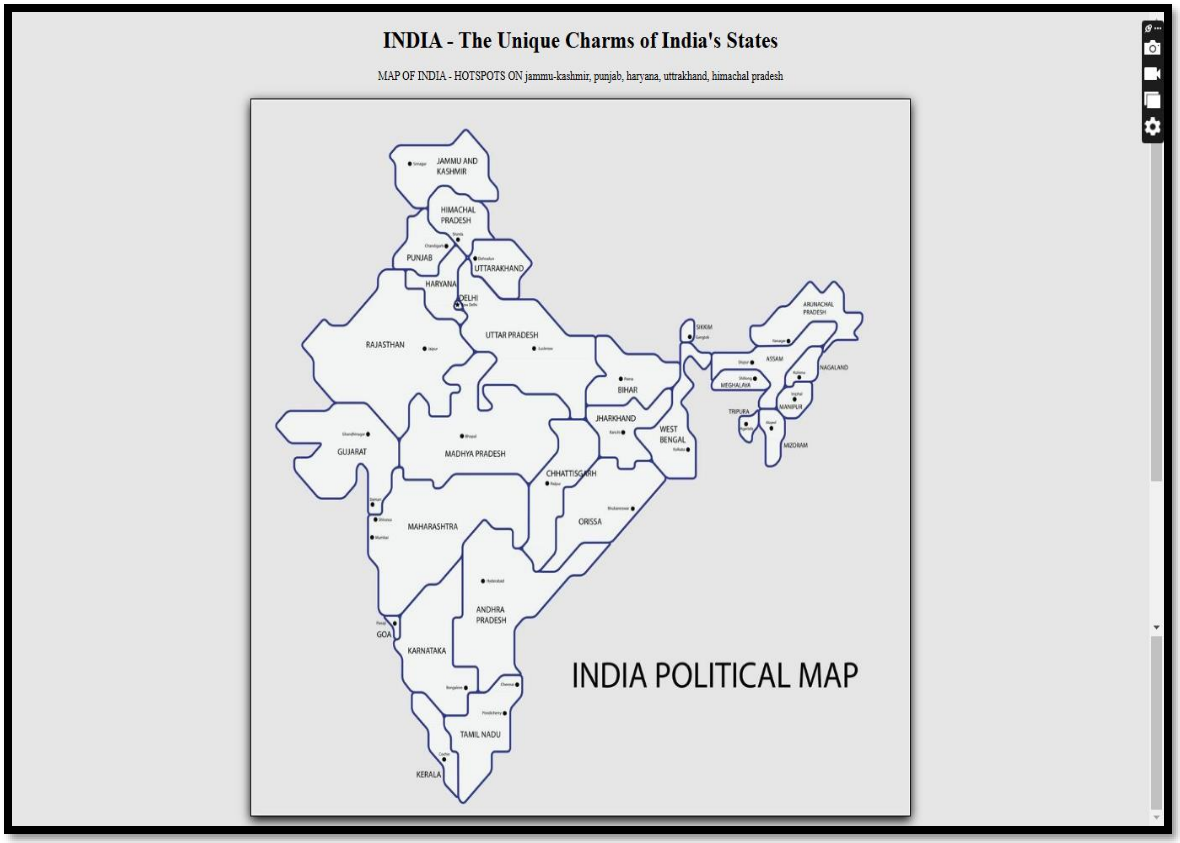
Program 5: Create a web page showing India&#39;s map and create hotspots on five different states and link them to the e-Government sites of the respective states.

Index.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>India</title>
   <style>
      img {
         border: 2px solid black;
         box-shadow: 0px 5px 15px black;
      }
   </style>
</head>
<body bgcolor="#E6E6E6">
   <h1 align="center">INDIA - The Unique Charms of India's States</h1>
   <center>
      <caption align="bottom">MAP OF INDIA - HOTSPOTS ON jammu-kashmir, punjab,
haryana, uttrakhand, himachal pradesh
      </caption><img src="india.jpg" alt="" width="" usemap="#name">
   </center>
   <map name="name">
      <area shape="poly" coords="218,77,314,60,340,40,388,138,302,118,277,148,218,77"
         href="https://jk.gov.in/jammukashmir/" alt="jammu-kashmir">
      <area shape="poly" coords="276,148,224,230,257,231,295,240,320,198,276,148"
href="https://punjab.gov.in/"
         alt="punjab">
      <area shape="poly"
coords="342,214,350,192,383,188,359,139,300,117,279,148,342,214"
         href="https://himachal.nic.in/en-IN/" alt="HP">
      <area shape="poly"
coords="342,214,351,191,410,191,444,225,423,271,395,273,342,214"
         href="https://www.india.gov.in/website-uttarakhand-government-0" alt="UTT">
      <area shape="poly"

coords="342,214,322,197,296,226,295,240,245,233,243,258,275,261,275,282,295,298,319
,300,334,288,316,287,325,231,342,214"
         href="https://haryana.gov.in/" alt="HR">
   </map>
</body>
```

```
</html>
```

Output



INDIA - The Unique Charms of India's States

MAP OF INDIA - HOTSPOTS ON jammu-kashmir, punjab, haryana, uttrakhand, himachal pradesh

Program 6: Create a web page using four frames and link them to an image, web page audio and video.

Index.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>four frame</title>
</head>
  <frameset cols="25%, 25%, 25%, 25%">
    <frame src="image.html" name="imageFrame" title="Image Frame">
    <frame src="wap4.html" name="webpageFrame" title="Web Page Frame">
    <frame src="audio.html" name="audioFrame" title="Audio Frame">
    <frame src="video.html" name="videoFrame" title="Video Frame">
  </frameset>
</html>
```

Image.html page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Image</title>
</head>
<body>
  <h1>This is a page that contains an image</h1>
  <img src="india.jpg" alt="" width="100%">
</body>
</html>
```

Audio.html page

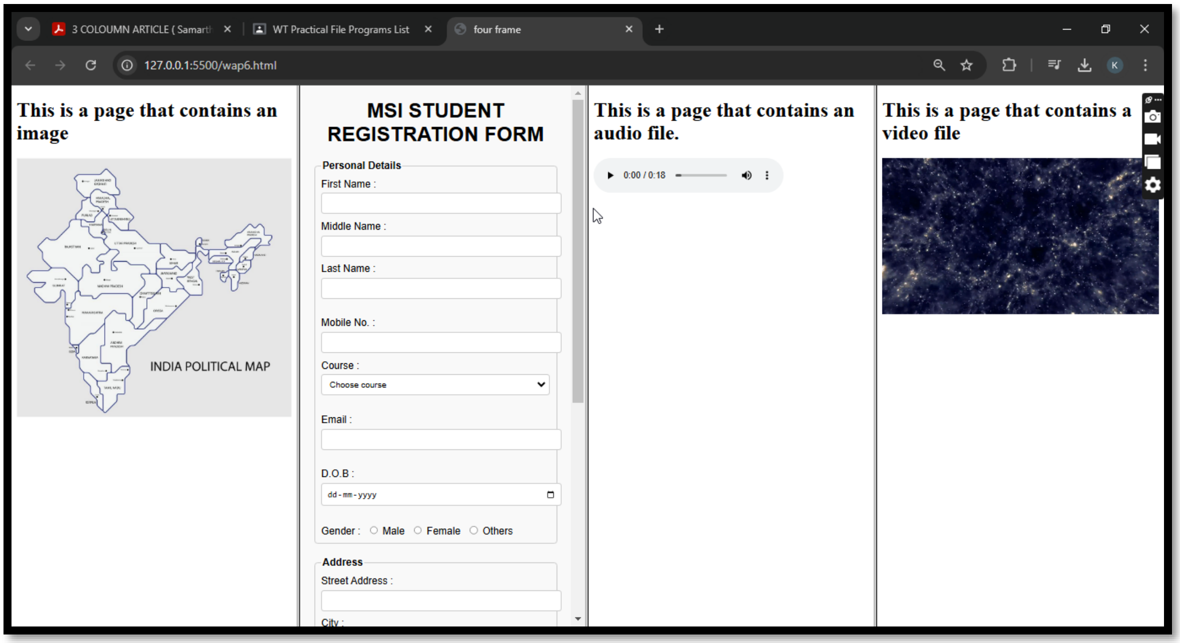
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Audio</title>
</head>
```

```
<body>
   <h1>This is a page that contains an audio file.</h1>
   <audio src="audio.mp3" controls loop autoplay></audio>
</body>
</html>
```

Video.html page

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Video</title>
</head>
<body>
   <h1>This is a page that contains a video file</h1>
   <video src="video.mp4" width="100%" controls loop autoplay></video>
</body>
</html>
```

OUTPUT :



This is a page that contains an image

MSI STUDENT REGISTRATION FORM

Personal Details

First Name :

Middle Name :

Last Name :

Mobile No. :

Course : Choose course

Email :

D.O.B : dd-mm-yyyy

Gender : ○ Male ○ Female ○ Others

Address

Street Address :

City :

This is a page that contains an audio file.

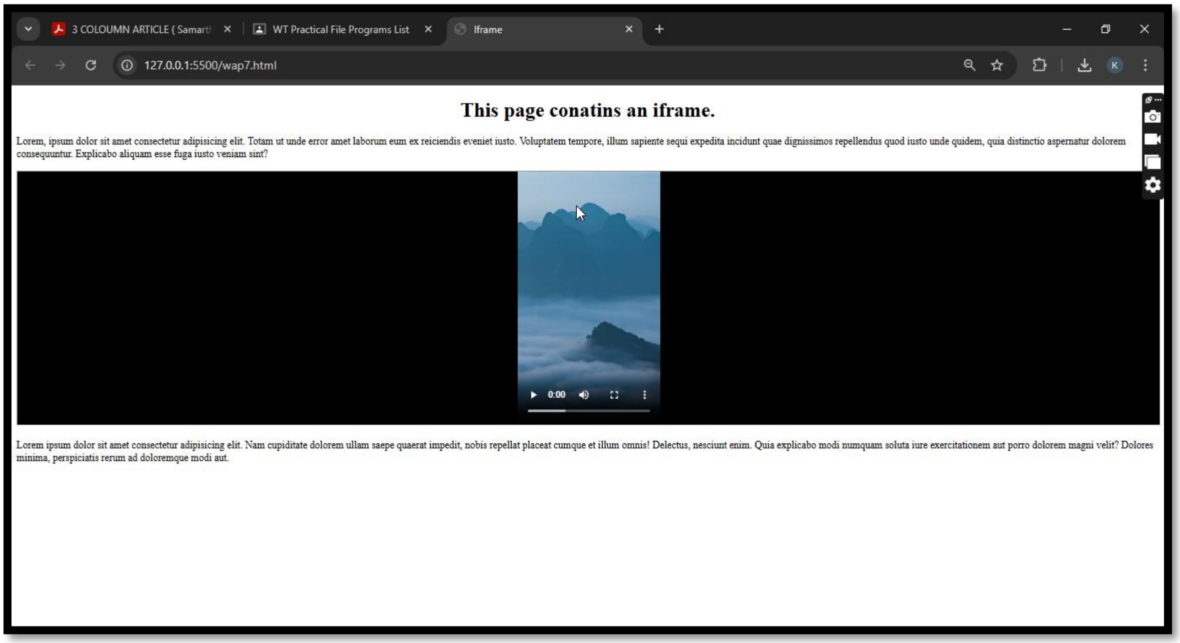
This is a page that contains a video file

Program 7: Create an Iframe and write some lines before and after it.

Index.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Iframe</title>
</head>
<body>
   <h1 align="center">This page contains an iframe.</h1>
   <p>Lorem, ipsum dolor sit amet consectetur adipisicing elit. Totam ut unde error amet laborum eum ex reiciendis eveniet iusto. Voluptatem tempore, illum sapiente sequi expedita incidunt quae dignissimos repellendus quod iusto unde quidem, quia distinctio aspernatur dolorem consequuntur. Explicabo aliquam esse fuga iusto veniam sint?</p>
   <iframe src="https://cdn.pixabay.com/video/2021/04/19/71569-538974129_tiny.mp4" frameborder="4" allowfullscreen width="100%" height="400px"></iframe>
   <p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Nam cupiditate dolorem ullam saepe quaerat impedit, nobis repellat placeat cumque et illum omnis! Delectus, nesciunt enim. Quia explicabo modi numquam soluta iure exercitationem aut porro dolorem magni velit? Dolores minima, perspiciatis rerum ad doloremque modi aut.</p>
</body>
</html>
```

OUTPUT :



127.0.0.1:5500/wap7.html

This page conatins an iframe.

Lorem, ipsum dolor sit amet consectetur adipisicing elit. Totam ut unde error amet laborum eum ex reiciendis eveniet iusto. Voluptatem tempore, illum sapiente sequi expedita incidunt quae dignissimos repellendus quod iusto unde quidem, quia distinctio aspernatur dolorem consequuntur. Explicabo aliquam esse fuga iusto veniam sint?

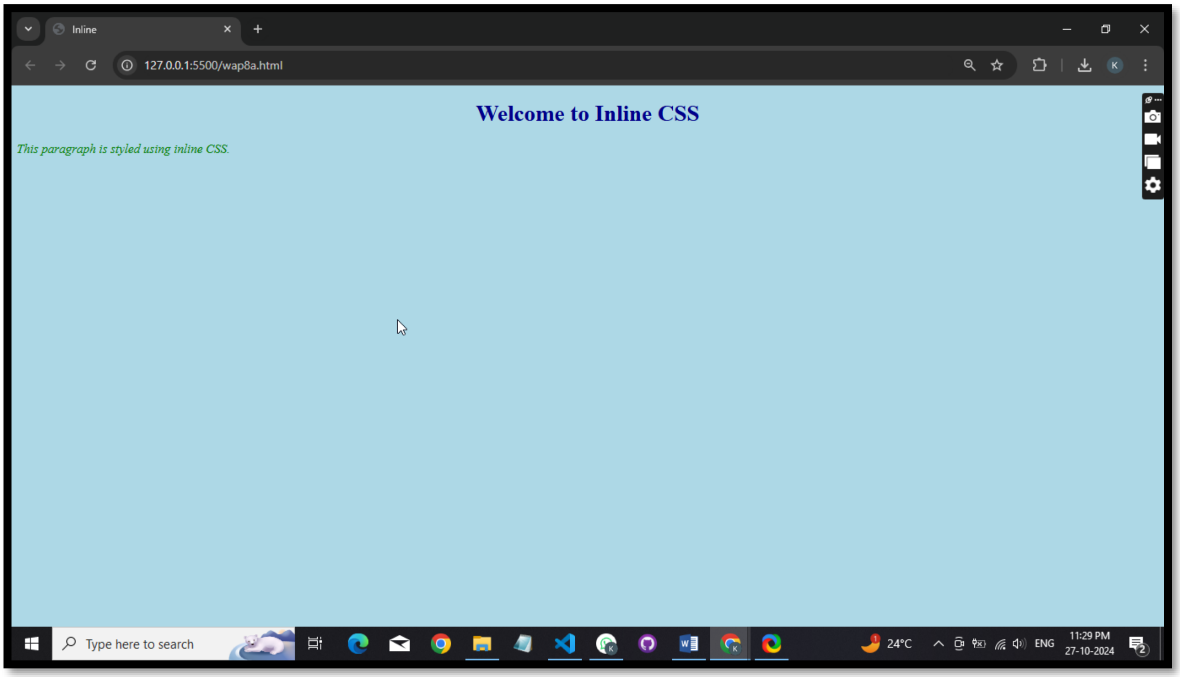
Lorem ipsum dolor sit amet consectetur adipisicing elit. Nam cupiditate dolorem ullam saepe quaerat impedit, nobis repellat placeat cumque et illum omnis! Delectus, nesciunt enim. Quia explicabo modi numquam soluta iure exercitationem aut porro dolorem magni velit? Dolores minima, perspiciatis rerum ad doloremque modi aut.

Program 8: Create 3 similar webpage with inline and internal style sheet and external style sheet and apply all Text styling attributes.

Page1.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Inline</title>
</head>
<body style="background-color: lightblue;">
  <h1 style="color: darkblue; text-align: center; font-size: 36px;">Welcome to Inline CSS</h1>
  <p style="color: green; font-size: 20px; font-style: italic;">This paragraph is styled using inline CSS.</p>
</body>
</html>
```

Output



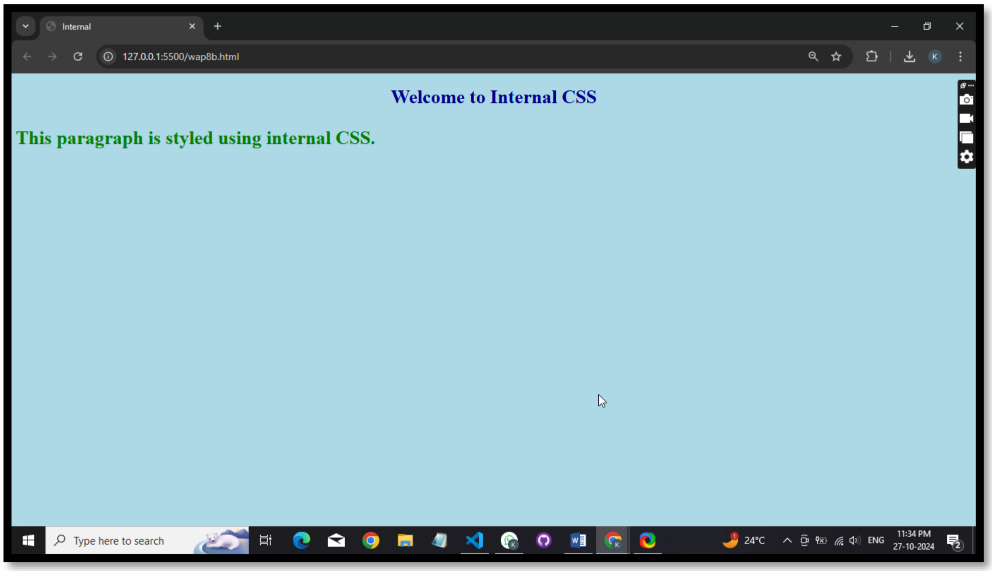
Welcome to Inline CSS

This paragraph is styled using inline CSS.

Page2.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Internal</title>
  <style type="text/css">
    body {
      background-color: lightyellow;
    }
    h1 {
      color: maroon;
      text-align: center;
      font-size: 36px;
    }
    p {
      color: purple;
      font-size: 20px;
      font-weight: bold;
    }
  </style>
</head>
<body>
  <h1>Welcome to Internal CSS</h1>
  <p>This paragraph is styled using internal CSS.</p>
</body>
</html>
```

Output



Welcome to Internal CSS

This paragraph is styled using internal CSS.

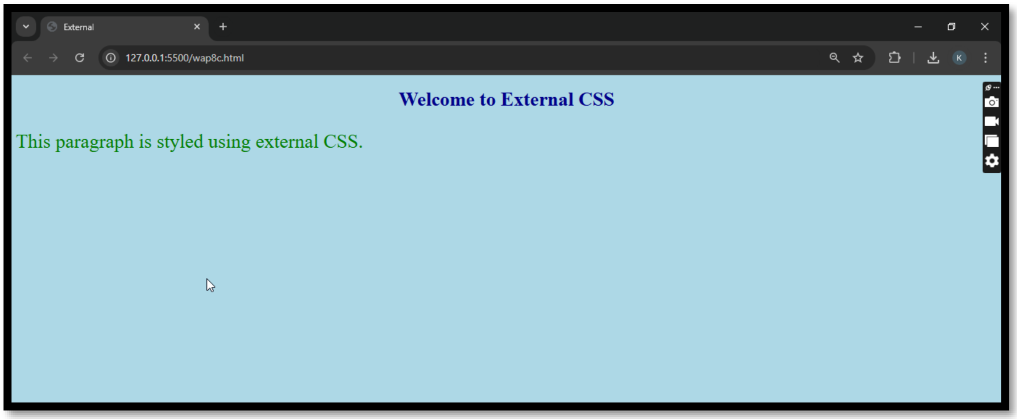
Page3.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>External</title>
   <link rel="stylesheet" href="style.css">
</head>
<body>
   <h1>Welcome to External CSS</h1>
   <p>This paragraph is styled using external CSS.</p>
</body>
</html>
```

Style.css Page

```
body {
   background-color: lightblue;
}
h1 {
   color: darkblue;
   text-align: center;
   font-size: 36px;
}
p {
   color: green;
   font-size: 36px;
   text-decoration:;
}
```

Output



Welcome to External CSS

This paragraph is styled using external CSS.

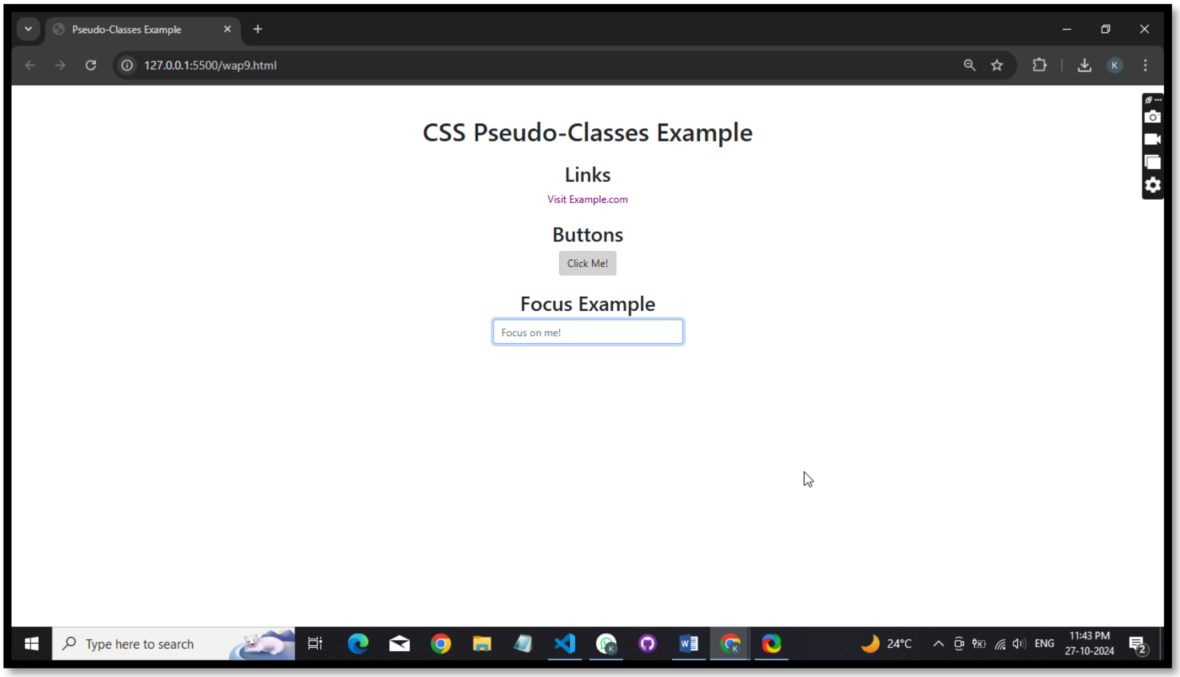
Program 9: Make a Web page that uses all the following pseudo classes. Hover, active, focus, visited.

Index.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Pseudo-Classes Example</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
  <style>
    /* Link styles */
    a {
      color: blue;
      text-decoration: none;
    }
    a:visited {
      color: purple; /* Color for visited links */
    }
    a:hover {
      color: red; /* Color when hovered */
      text-decoration: underline; /* Underline on hover */
    }
    a:active {
      color: green; /* Color when active (clicked) */
    }
    /* Button styles */
    .btn-custom {
      background-color: lightgray;
      border: 1px solid #ccc;
    }
    .btn-custom:hover {
      background-color: darkgray; /* Change on hover */
    }
    .btn-custom:focus {
      outline: 2px solid blue; /* Outline on focus */
    }
  </style>
</head>
<body>
<div class="container mt-5">
```

```
  <h1 class="text-center">CSS Pseudo-Classes Example</h1>
  <div class="text-center mt-4">
    <h2>Links</h2>
    <p>
      <a href="https://www.example.com" target="_blank">Visit Example.com</a>
    </p>
  </div>
  <div class="text-center mt-4">
    <h2>Buttons</h2>
    <button class="btn btn-custom" onclick="alert('Button clicked!')">Click Me!</button>
  </div>
  <div class="text-center mt-4">
    <h2>Focus Example</h2>
    <input type="text" placeholder="Focus on me!" class="form-control" style="width: 300px; margin: auto;">
  </div>
</div>
</body>
</html>
```

Output



CSS Pseudo-Classes Example

Links

Visit Example.com

Buttons

Click Me!

Focus Example

Focus on me!

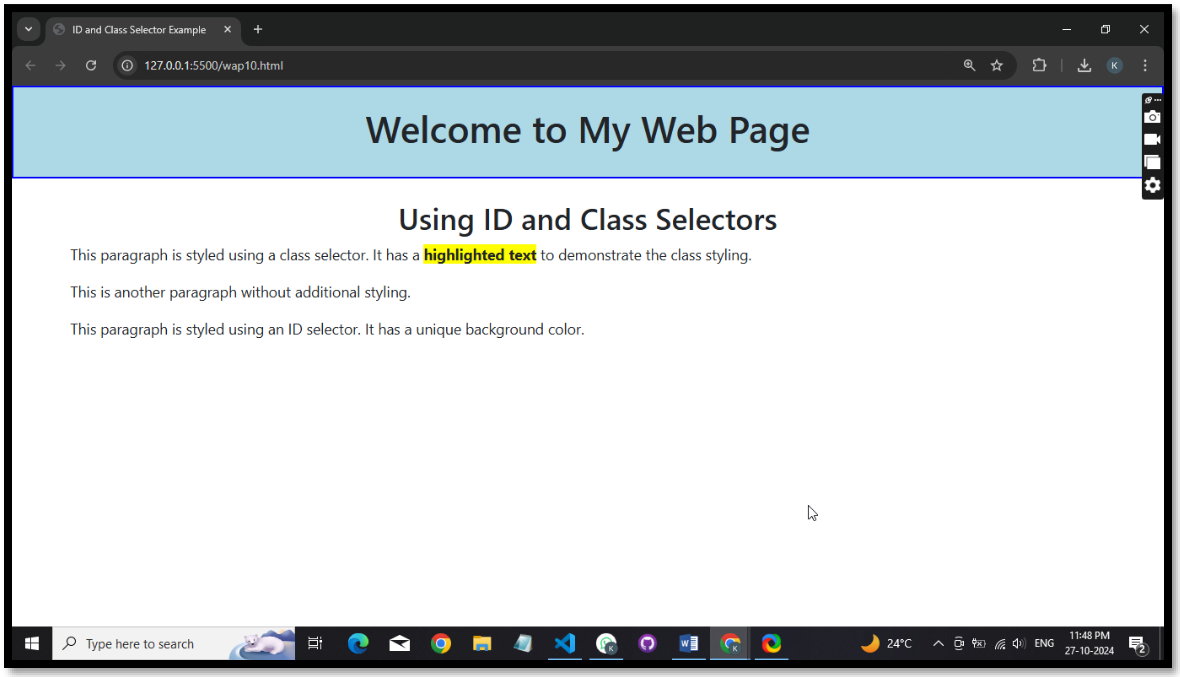
Program 10: Create a web page that use Id and class selector.

Index.html Page

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>ID and Class Selector Example</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
  <style>
    /* ID Selector */
    #header {
      background-color: lightblue;
      padding: 20px;
      text-align: center;
      border: 2px solid blue;
    }
    /* Class Selector */
    .highlight {
      background-color: yellow;
      font-weight: bold;
    }
    .text-center {
      text-align: center;
    }
  </style>
</head>
<body>
<div id="header">
  <h1>Welcome to My Web Page</h1>
</div>
<div class="container mt-4">
  <h2 class="text-center">Using ID and Class Selectors</h2>
  <p>This paragraph is styled using a class selector. It has a <span
class="highlight">highlighted text</span> to demonstrate the class styling.</p>
  <p>This is another paragraph without additional styling.</p>
  <p id="uniqueParagraph">This paragraph is styled using an ID selector. It has a unique
background color.</p>
</div>
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"></sc
```

```
ript>
</body>
</html>
```

Output



Welcome to My Web Page

Using ID and Class Selectors

This paragraph is styled using a class selector. It has a **highlighted text** to demonstrate the class styling.

This is another paragraph without additional styling.

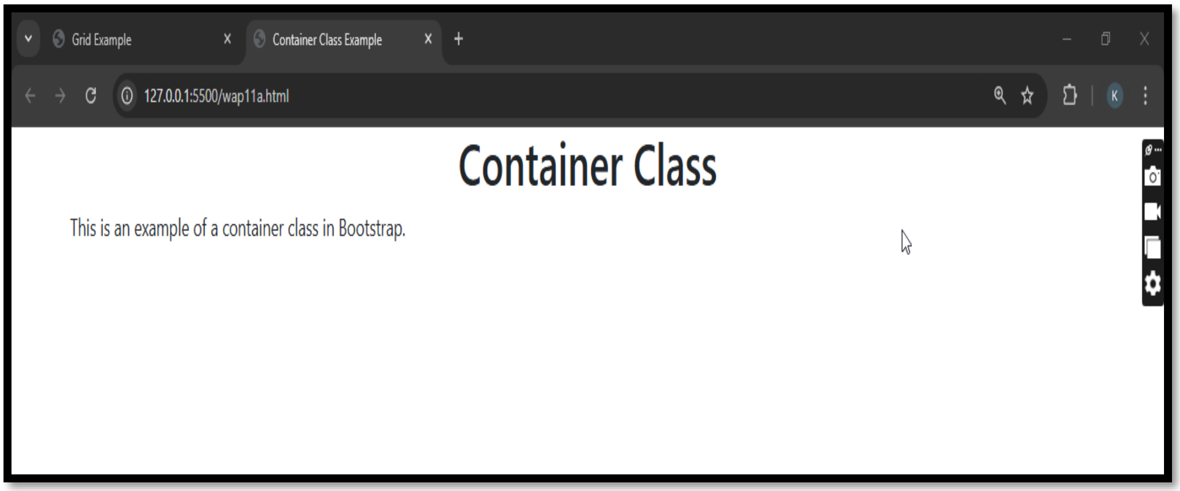
This paragraph is styled using an ID selector. It has a unique background color.

Program 11: Create separate web pages containing: Container Class, Grid, Tables, Image, Button, Typography, Jumbotron, Glyphicons.

Container class

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Container Class Example</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>

<div class="container">
  <h1 class="text-center">Container Class</h1>
  <p>This is an example of a container class in Bootstrap.</p>
</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js">
</script>
</body>
</html>
```

Output



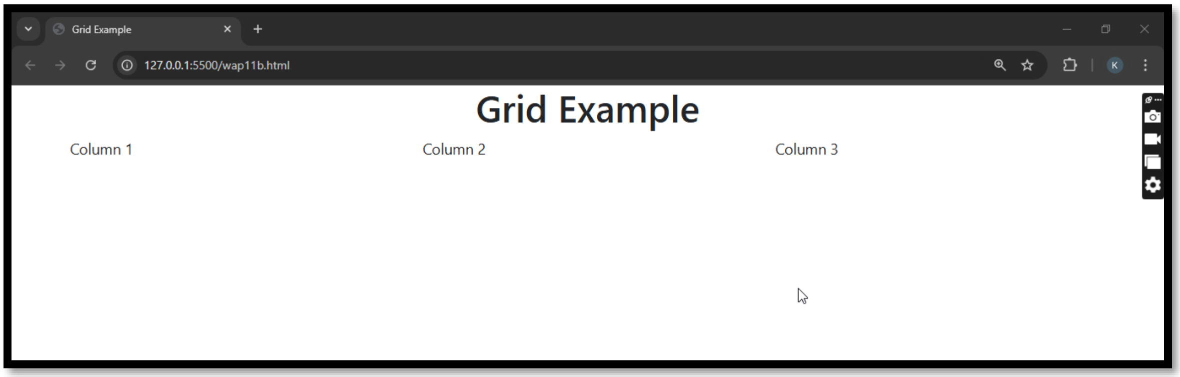
Container Class

This is an example of a container class in Bootstrap.

Grid

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Grid Example</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>

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

<div class="container">
  <div class="row">
    <div class="col">Column 1</div>
    <div class="col">Column 2</div>
    <div class="col">Column 3</div>
  </div>
</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js">
</script>
</body>
</html>
```

Output



Grid Example

Column 1 Column 2 Column 3

Table

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Table Example</title>
  <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>

<h1 class="text-center">Table Example</h1>

<div class="container">
  <table class="table table-bordered">
    <thead>
      <tr>
        <th>Name</th>
        <th>Age</th>
        <th>City</th>
      </tr>
    </thead>
    <tbody>
      <tr>
        <td>John Doe</td>
        <td>30</td>
        <td>New York</td>
      </tr>
      <tr>
        <td>Jane Smith</td>
        <td>25</td>
        <td>Los Angeles</td>
      </tr>
    </tbody>
  </table>
</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"></sc
ript>
</body>
</html>
```

Output

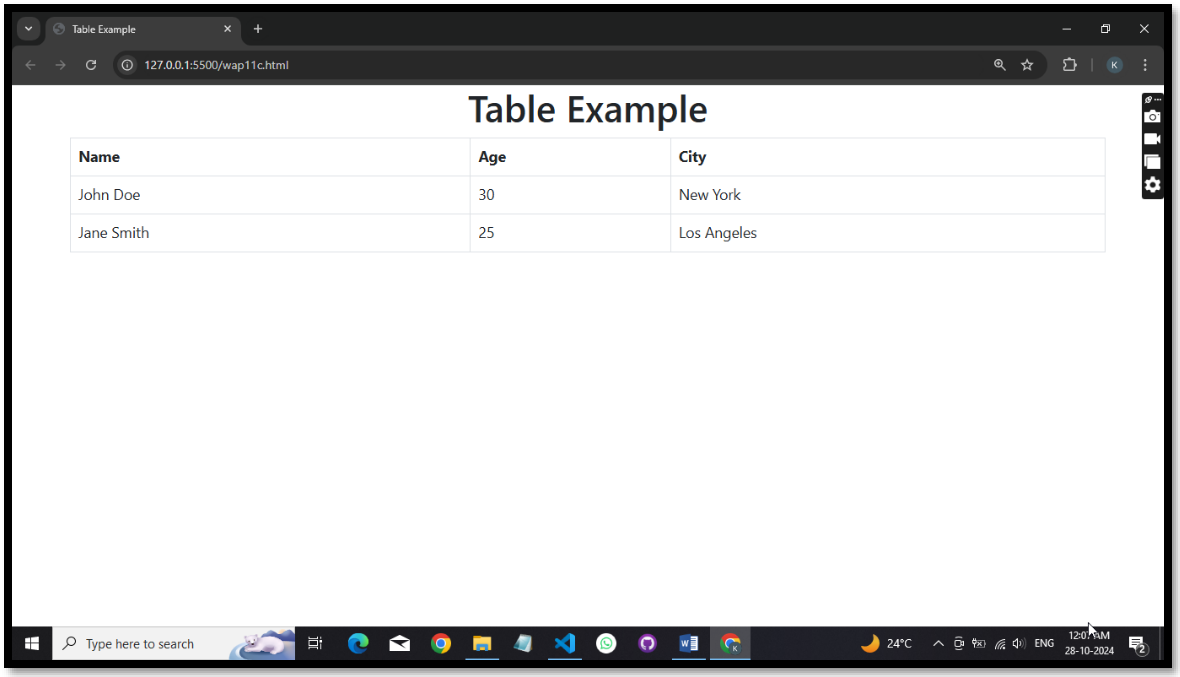


Table Example

Name	Age	City
John Doe	30	New York
Jane Smith	25	Los Angeles

Image

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Image Example</title>
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>

<h1 class="text-center">Image Example</h1>

<div class="container text-center">
   <img src="https://via.placeholder.com/300" alt="Placeholder Image" class="img-fluid">
   <p>This is an example of an image in Bootstrap.</p>
</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js">
</script>
</body>
```

```
</html>
```

Output

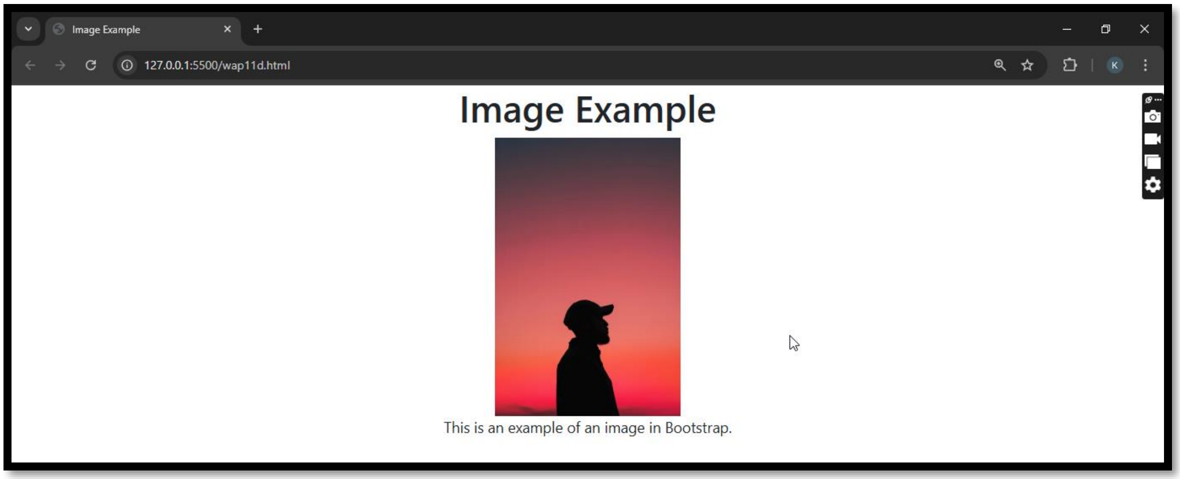


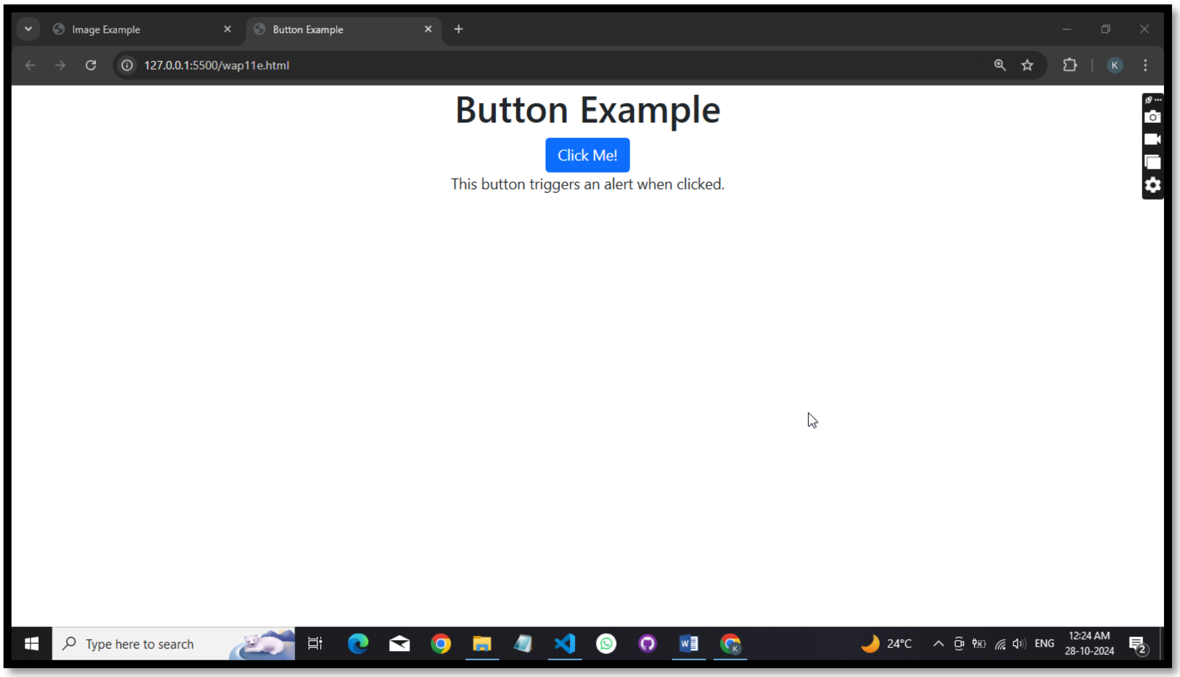
Image Example

This is an example of an image in Bootstrap.

Button

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Button Example</title>
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>

<h1 class="text-center">Button Example</h1>

<div class="container text-center">
   <button class="btn btn-primary" onclick="alert('Button clicked!')">Click Me!</button>
   <p>This button triggers an alert when clicked.</p>
</div>

<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"></sc
ript>
</body>
</html>
```

Output



Button Example

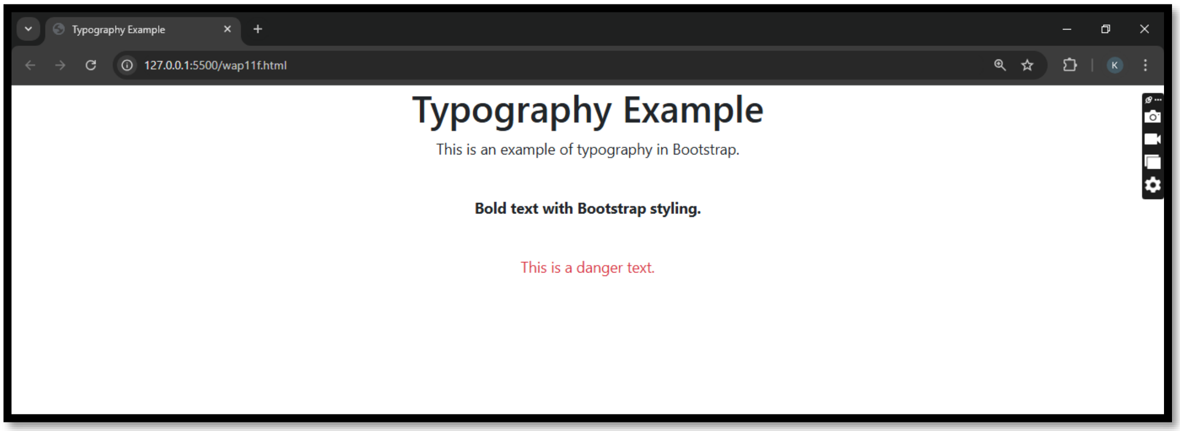
This button triggers an alert when clicked.

Typography

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Typography Example</title>
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>

<h1 class="text-center">Typography Example</h1>

<div class="container text-center">
    <p>This is an example of typography in Bootstrap.</p><br/>
    <p class='fw-bold'>Bold text with Bootstrap styling.</p><br/>
    <p class='text-danger'>This is a danger text.</p><br/>

</div>
```

```
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"></sc
ript>
</body>
</html>
```

Output



Typography Example

This is an example of typography in Bootstrap.

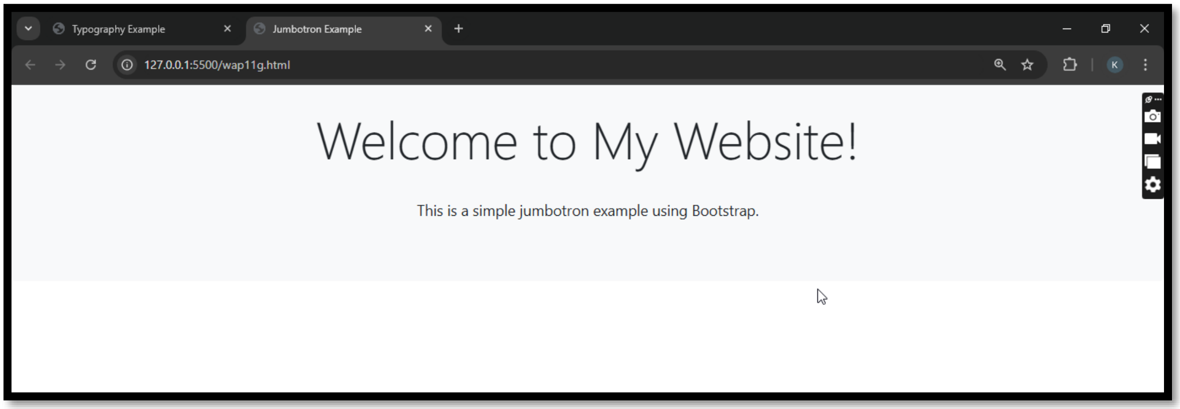
Bold text with Bootstrap styling.

This is a danger text.

Jumbotron

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Jumbotron Example</title>
   <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
rel="stylesheet">
</head>
<body>

<div class="jumbotron text-center bg-light p-4 mb-4 rounded-3">
  <h1 class='display-4'>Welcome to My Website!</h1><br/>
  <p>This is a simple jumbotron example using Bootstrap.</p><br/>
  </div>
<script
src='https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js'>
</script>
 </body>
</html>
```

Output



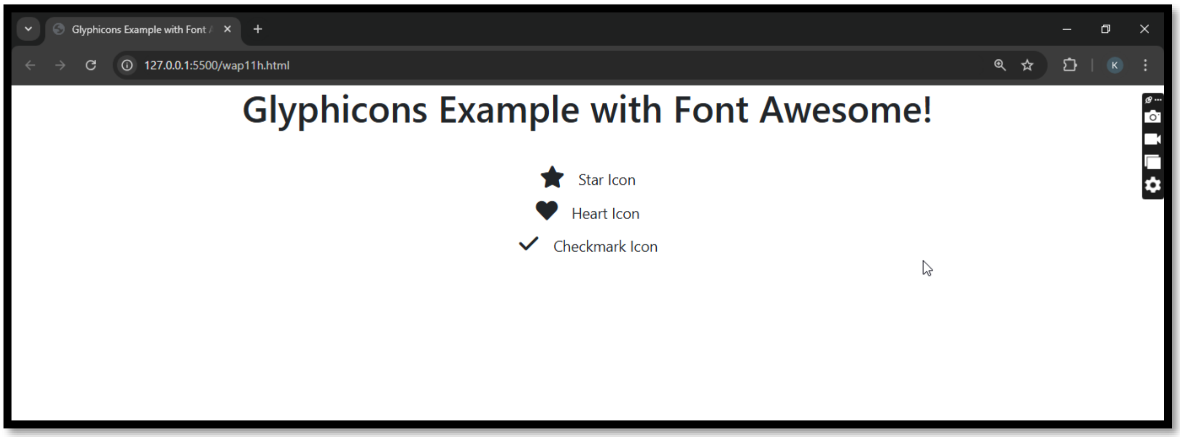
Welcome to My Website!

This is a simple jumbotron example using Bootstrap.

Glyphicons

```
<!DOCTYPE html>
<html lang='en'>
<head>
<meta charset='utf-8'>
<meta name='viewport' content='width=device-width, initial-scale=1'>
<title>Glyphicons Example with Font Awesome</title>
<link href='https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0-
beta3/css/all.min.css' rel='stylesheet'>
<link href='https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css'
rel='stylesheet'>
<style>
.icon { font-size: 24px; margin-right: 10px; }
 </style></head><body>

<h1 class='text-center'>Glyphicons Example with Font Awesome!</h1><br/>

<div class='container text-center'>
<span class='icon'><i class='fas fa-star'></i></span> Star Icon<br/>
<span class='icon'><i class='fas fa-heart'></i></span> Heart Icon<br/>
<span class='icon'><i class='fas fa-check'></i></span> Checkmark Icon<br/>
 </div><script
src='https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js'></scr
ipt>
</body>
</html>
```

Output



Glyphicons Example with Font Awesome!

★ Star Icon

♥ Heart Icon

✓ Checkmark Icon

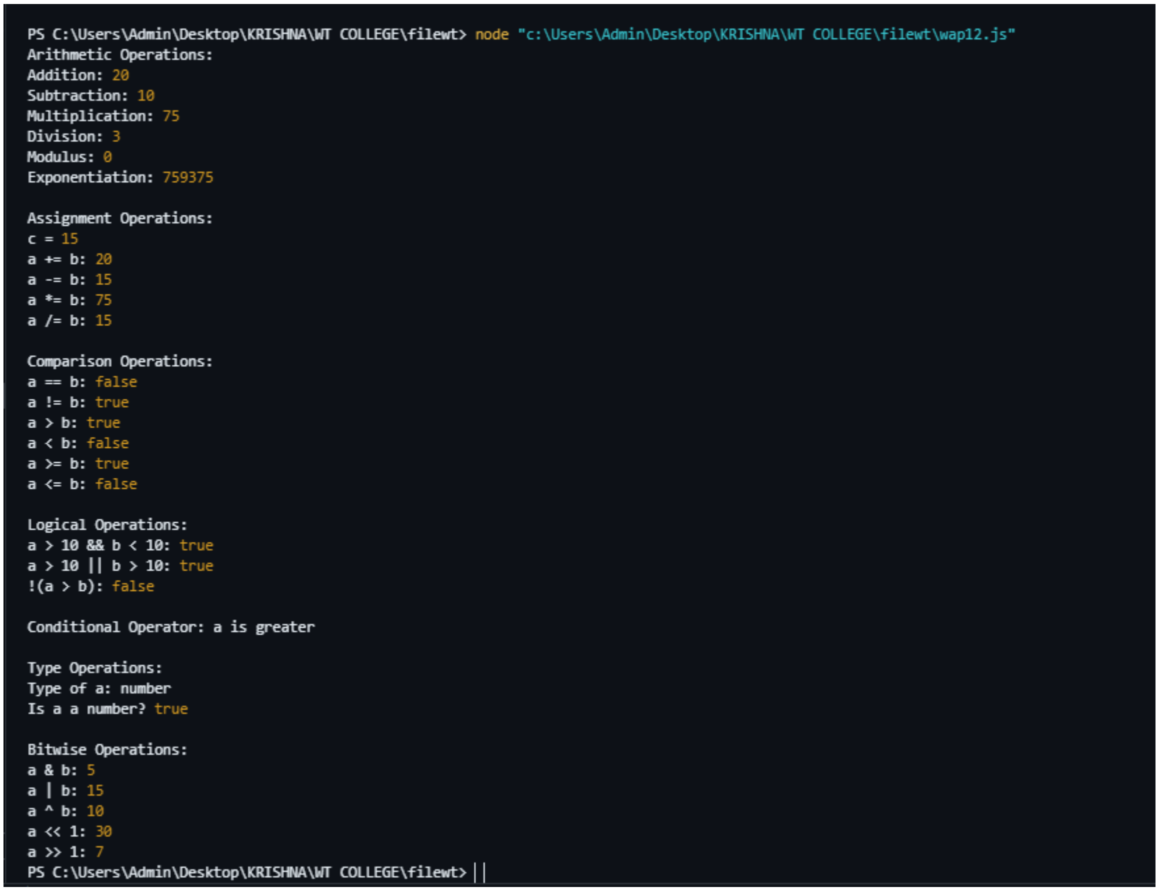
Program 12: WAP to make use of JavaScript Operator.

Code :

```
let a = 15;
let b = 5;

// Arithmetic Operations
console.log("Arithmetic Operations:");
console.log("Addition:", a + b);
console.log("Subtraction:", a - b);
console.log("Multiplication:", a * b);
console.log("Division:", a / b);
console.log("Modulus:", a % b);
console.log("Exponentiation:", a ** b);

// Assignment Operations
console.log("\nAssignment Operations:");
let c = a;
console.log("c =", c);
a += b;
console.log("a += b:", a);
a -= b;
console.log("a -= b:", a);
a *= b;
console.log("a *= b:", a);
a /= b;
console.log("a /= b:", a);

// Comparison Operations
console.log("\nComparison Operations:");
console.log("a == b:", a == b);
console.log("a != b:", a != b);
console.log("a > b:", a > b);
console.log("a < b:", a < b);
console.log("a >= b:", a >= b);
console.log("a <= b:", a <= b);

// Logical Operations
console.log("\nLogical Operations:");
console.log("a > 10 && b < 10:", a > 10 && b < 10);
console.log("a > 10 || b > 10:", a > 10 || b > 10);
console.log("!(a > b):", !(a > b));
```

```
// Conditional (Ternary) Operator
let result = a > b ? "a is greater" : "b is greater";
console.log("\nConditional Operator:", result);

// Type Operations
console.log("\nType Operations:");
console.log("Type of a:", typeof a);
console.log("Is a a number?", typeof a === 'number');

// Bitwise Operations
console.log("\nBitwise Operations:");
console.log("a & b:", a & b);
console.log("a | b:", a | b);
console.log("a ^ b:", a ^ b);
console.log("a << 1:", a << 1);
console.log("a >> 1:", a >> 1);
```

Output :



```
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt> node "c:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt\wap12.js"
Arithmetic Operations:
Addition: 20
Subtraction: 10
Multiplication: 75
Division: 3
Modulus: 0
Exponentiation: 759375

Assignment Operations:
c = 15
a += b: 20
a -= b: 15
a *= b: 75
a /= b: 15

Comparison Operations:
a == b: false
a != b: true
a > b: true
a < b: false
a >= b: true
a <= b: false

Logical Operations:
a > 10 && b < 10: true
a > 10 || b > 10: true
!(a > b): false

Conditional Operator: a is greater

Type Operations:
Type of a: number
Is a a number? true

Bitwise Operations:
a & b: 5
a | b: 15
a ^ b: 10
a << 1: 30
a >> 1: 7
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt>
```

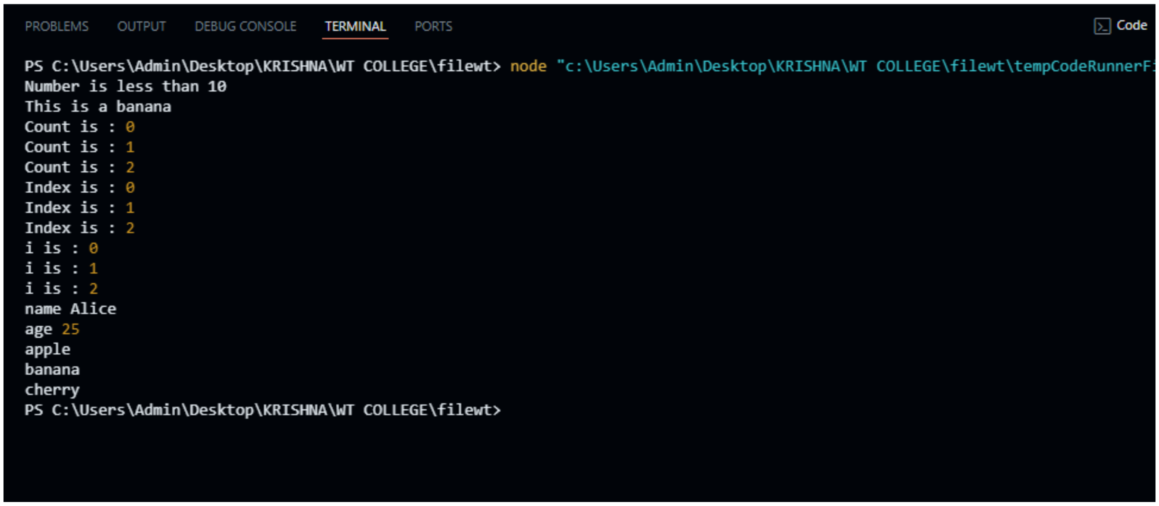
Program 13: WAP to make use of statements : if , else, else if, switch, while, do while, for, for in , for of.

Code :

```
let number = 5;
let fruits = ['apple', 'banana', 'cherry'];
let person = { name: 'Alice', age: 25 };

// Using if, else if, and else
if (number > 10) {
   console.log('Number is greater than 10');
} else if (number === 10) {
   console.log('Number is equal to 10');
} else {
   console.log('Number is less than 10');
}

// Using switch
let fruit = 'banana';
switch (fruit) {
   case 'apple':
      console.log('This is an apple');
      break;
   case 'banana':
      console.log('This is a banana');
      break;
   case 'cherry':
      console.log('This is a cherry');
      break;
   default:
      console.log('Unknown fruit');
}

// Using while
let count = 0;
while (count < 3) {
   console.log("Count is :",count);
   count++;
}

// Using do while
let index = 0;
```

```
do {
   console.log("Index is :",index);
   index++;
} while (index < 3);

// Using for
for (let i = 0; i < 3; i++) {
   console.log("i is :",i);
}

// Using for...in
for (let key in person) {
   console.log(key,person[key]);
}

// Using for...of
for (let fruit of fruits) {
   console.log(fruit);
}
```

Output :



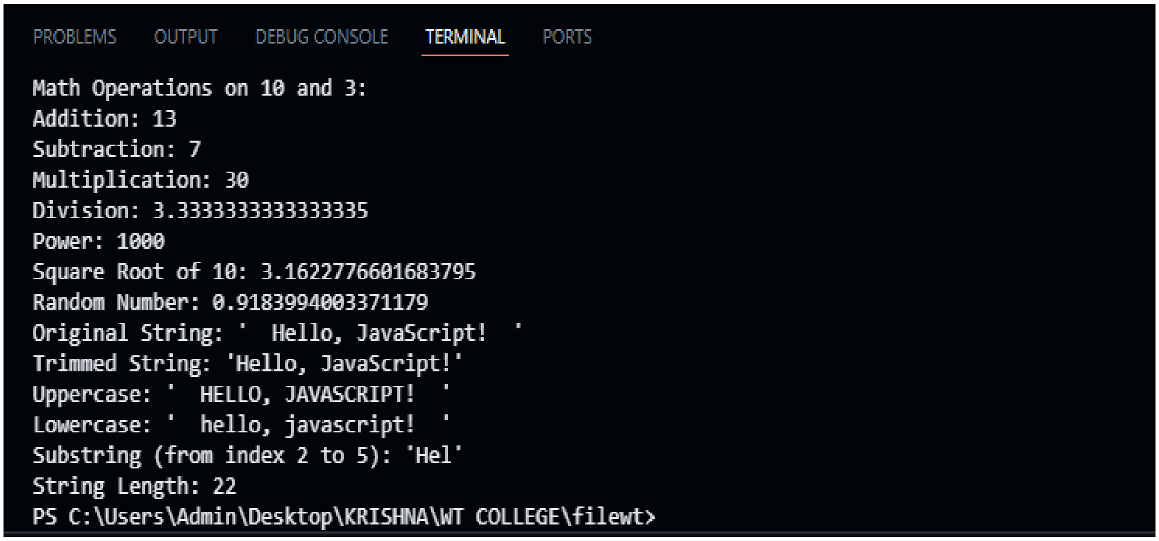
```
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt> node "c:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt\tempCodeRunnerF
Number is less than 10
This is a banana
Count is : 0
Count is : 1
Count is : 2
Index is : 0
Index is : 1
Index is : 2
i is : 0
i is : 1
i is : 2
name Alice
age 25
apple
banana
cherry
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt>
```

Program 14: WAP to make use of following objects : date, math, string.

Code :

```
// Using the Date object
function displayCurrentDate() {
   const currentDate = new Date();
   console.log("Current Date and Time: " + currentDate.toString());
}

// Using the Math object
function performMathOperations() {
   const num1 = 10;
   const num2 = 3;

   console.log(`Math Operations on ${num1} and ${num2}:`);
   console.log("Addition: " + (num1 + num2));
   console.log("Subtraction: " + (num1 - num2));
   console.log("Multiplication: " + (num1 * num2));
   console.log("Division: " + (num1 / num2));
   console.log("Power: " + Math.pow(num1, num2));
   console.log("Square Root of " + num1 + ": " + Math.sqrt(num1));
   console.log("Random Number: " + Math.random());
}

// Using the String object
function manipulateStrings() {
   const str = "  Hello, JavaScript!  ";

   console.log("Original String: '" + str + "'");
   console.log("Trimmed String: '" + str.trim() + "'");
   console.log("Uppercase: '" + str.toUpperCase() + "'");
   console.log("Lowercase: '" + str.toLowerCase() + "'");
   console.log("Substring (from index 2 to 5): '" + str.substring(2, 5) + "'");
   console.log("String Length: " + str.length);
}

displayCurrentDate();
performMathOperations();
manipulateStrings();
```

Output :



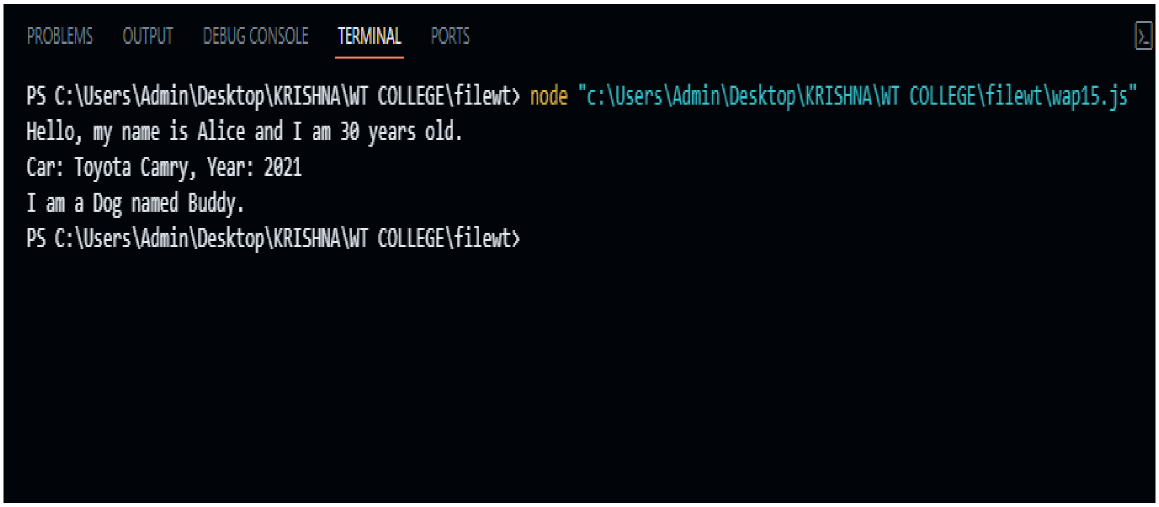
```
Math Operations on 10 and 3:
Addition: 13
Subtraction: 7
Multiplication: 30
Division: 3.3333333333333335
Power: 1000
Square Root of 10: 3.1622776601683795
Random Number: 0.9183994003371179
Original String: '  Hello, JavaScript!  '
Trimmed String: 'Hello, JavaScript!'
Uppercase: '  HELLO, JAVASCRIPT!  '
Lowercase: '  hello, javascript!  '
Substring (from index 2 to 5): 'Hel'
String Length: 22
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt>
```

Program 15: WAP to show use of object in JavaScript: By object literal, By instance of object directly, By using object constructor

Code :

```
// 1. Using Object Literal
const personLiteral = {
   name: 'Alice',
   age: 30,
   greet: function() {
      console.log(`Hello, my name is ${this.name} and I am ${this.age} years old.`);
   }
};

// 2. Creating an Instance Directly
const carInstance = {
   brand: 'Toyota',
   model: 'Camry',
   year: 2021,
   displayInfo: function() {
      console.log(`Car: ${this.brand} ${this.model}, Year: ${this.year}`);
   }
};

// 3. Using an Object Constructor
function Animal(type, name) {
   this.type = type;
   this.name = name;
   this.introduce = function() {
      console.log(`I am a ${this.type} named ${this.name}.`);
   };
}

const dog = new Animal('Dog', 'Buddy');

// Using the objects
personLiteral.greet(); // Output: Hello, my name is Alice and I am 30 years old.
carInstance.displayInfo(); // Output: Car: Toyota Camry, Year: 2021
dog.introduce(); // Output: I am a Dog named Buddy.
```

Output :



```
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt> node "c:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt\wap15.js"
Hello, my name is Alice and I am 30 years old.
Car: Toyota Camry, Year: 2021
I am a Dog named Buddy.
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt>
```

Program 16: WAP to show validation in Java Script.

Code :

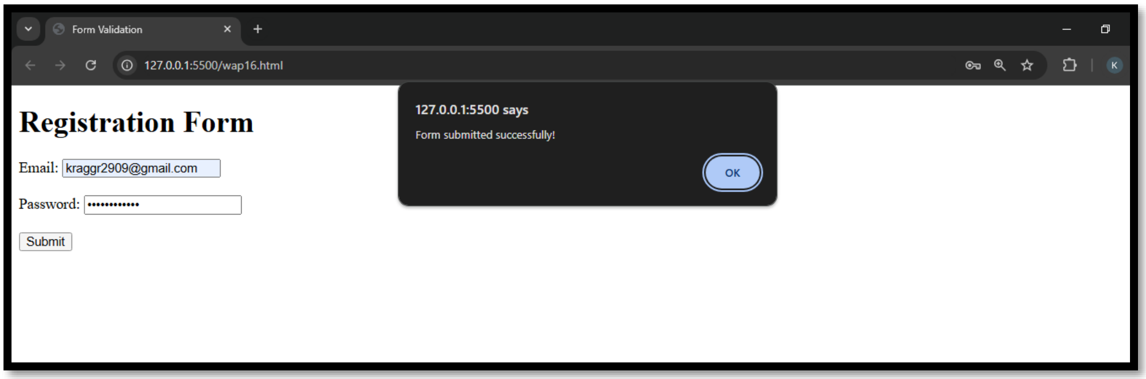
```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Form Validation</title>
  <style>
    .error {
      color: red;
    }
  </style>
</head>
<body>
  <h1>Registration Form</h1>
  <form id="registrationForm">
    <label for="email">Email:</label>
    <input type="text" id="email" name="email" required>
    <span class="error" id="emailError"></span>
    <br><br>

    <label for="password">Password:</label>
    <input type="password" id="password" name="password" required>
    <span class="error" id="passwordError"></span>
    <br><br>

    <button type="submit">Submit</button>
  </form>

  <script src="wap16.js"></script>
</body>
</html>
```

Wap16.js code :

```
document.getElementById('registrationForm').addEventListener('submit',
function(event) {
  event.preventDefault(); // Prevent form submission
```

```
    // Clear previous error messages
    document.getElementById('emailError').textContent = '';
    document.getElementById('passwordError').textContent = '';

    // Get input values
    const email = document.getElementById('email').value;
    const password = document.getElementById('password').value;
    let isValid = true;

    // Validate email
    const emailPattern = /^[^\s@]+@[^\s@]+\.[^\s@]+$/;
    if (!emailPattern.test(email)) {
        document.getElementById('emailError').textContent = 'Please enter a valid email
address.';
        isValid = false;
    }

    // Validate password
    if (password.length < 6) {
        document.getElementById('passwordError').textContent = 'Password must be at
least 6 characters long.';
        isValid = false;
    }

    // If valid, you can submit the form or perform further actions
    if (isValid) {
        alert('Form submitted successfully!');
        // Here you can proceed with form submission, e.g., via AJAX or direct submission.
        // this.submit();
    }
});
```

Output :



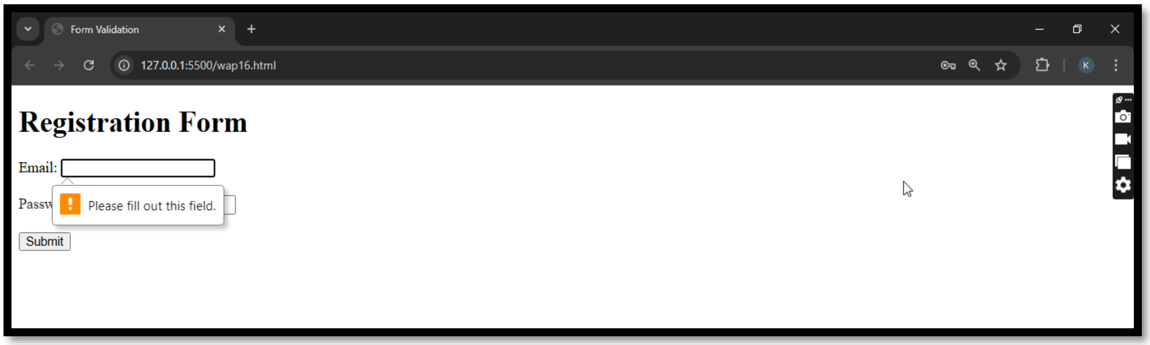
127.0.0.1:5500 says

Form submitted successfully!

Registration Form

Email: kraggr2909@gmail.com

Password: ••••••••••••

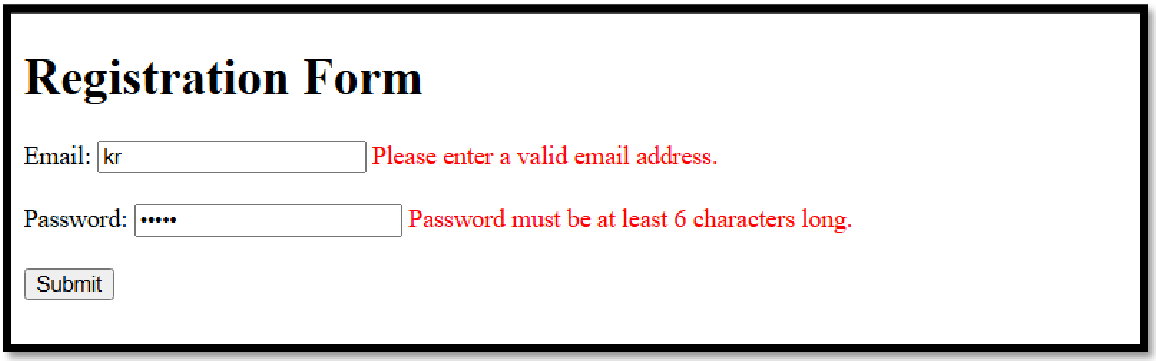


Registration Form

Email:

Passw

Please fill out this field.



Registration Form

Email: kr Please enter a valid email address.

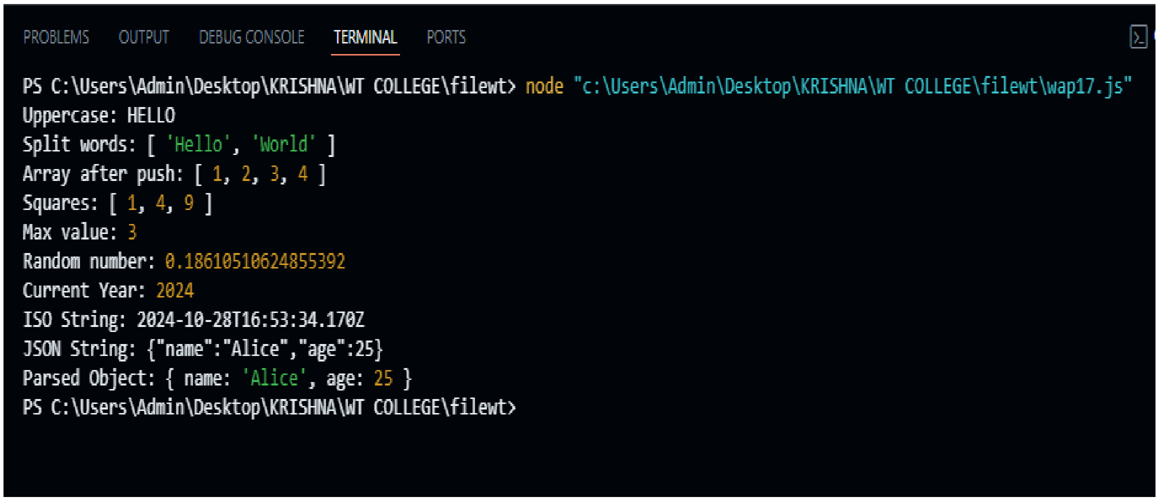
Password: ••••• Password must be at least 6 characters long.

Program 17: WAP to make use of in-built functions.
Code :

```
// String Methods
const str = "hello";
console.log("Uppercase:", str.toUpperCase()); // "HELLO"

const sentence = "Hello World";
const words = sentence.split(" ");
console.log("Split words:", words); // ["Hello", "World"]

// Array Methods
const arr = [1, 2, 3];
arr.push(4);
console.log("Array after push:", arr); // [1, 2, 3, 4]

const numbers = [1, 2, 3];
const squares = numbers.map(num => num * num);
console.log("Squares:", squares); // [1, 4, 9]

// Math Methods
const max = Math.max(1, 3, 2);
console.log("Max value:", max); // 3

const randomNum = Math.random();
console.log("Random number:", randomNum); // e.g., 0.123456789

// Date Methods
const date = new Date();
console.log("Current Year:", date.getFullYear()); // e.g., 2024

console.log("ISO String:", date.toISOString()); // e.g., "2024-10-25T12:34:56.789Z"

// JSON Methods
const obj = { name: "Alice", age: 25 };
const jsonString = JSON.stringify(obj);
console.log("JSON String:", jsonString); // '{"name":"Alice","age":25}'
```

const parsedObj = JSON.parse(jsonString);
console.log("Parsed Object:", parsedObj); // { name: "Alice", age: 25 }

Output :



```
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt> node "c:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt\wap17.js"
Uppercase: HELLO
Split words: [ 'Hello', 'World' ]
Array after push: [ 1, 2, 3, 4 ]
Squares: [ 1, 4, 9 ]
Max value: 3
Random number: 0.18610510624855392
Current Year: 2024
ISO String: 2024-10-28T16:53:34.170Z
JSON String: {"name":"Alice","age":25}
Parsed Object: { name: 'Alice', age: 25 }
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt>
```

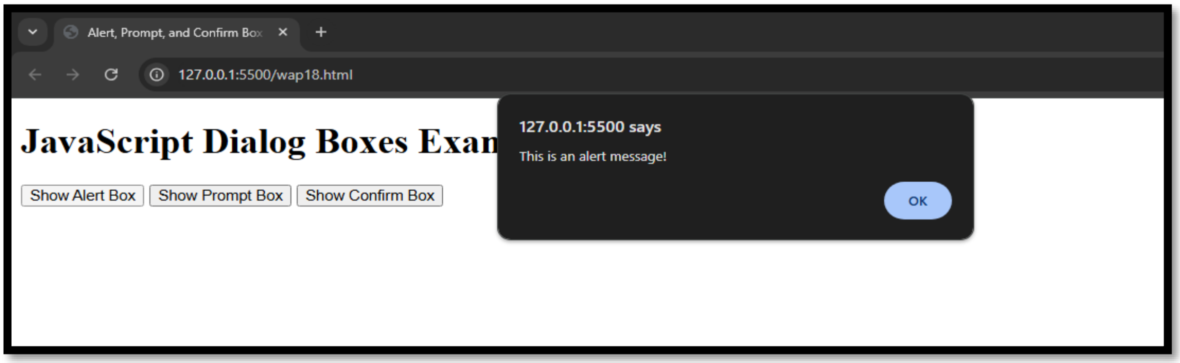
Program 18: WAP to show usage of : alert box, prompt box, confirm box.

Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Alert, Prompt, and Confirm Box Example</title>
   <script type="text/javascript">
      function showAlert() {
         alert("This is an alert message!");
      }

      function showPrompt() {
         var name = prompt("Please enter your name:", "John Doe");
         if (name != null && name != "") {
            alert("Hello " + name + "!");
         } else {
            alert("No name entered.");
         }
      }

      function showConfirm() {
         var result = confirm("Do you want to proceed?");
         if (result) {
            alert("You pressed OK!");
         } else {
            alert("You pressed Cancel!");
         }
      }
   </script>
</head>
<body>
   <h1>JavaScript Dialog Boxes Example</h1>
   <button onclick="showAlert()">Show Alert Box</button>
   <button onclick="showPrompt()">Show Prompt Box</button>
   <button onclick="showConfirm()">Show Confirm Box</button>
</body>
```

```
</html>
```

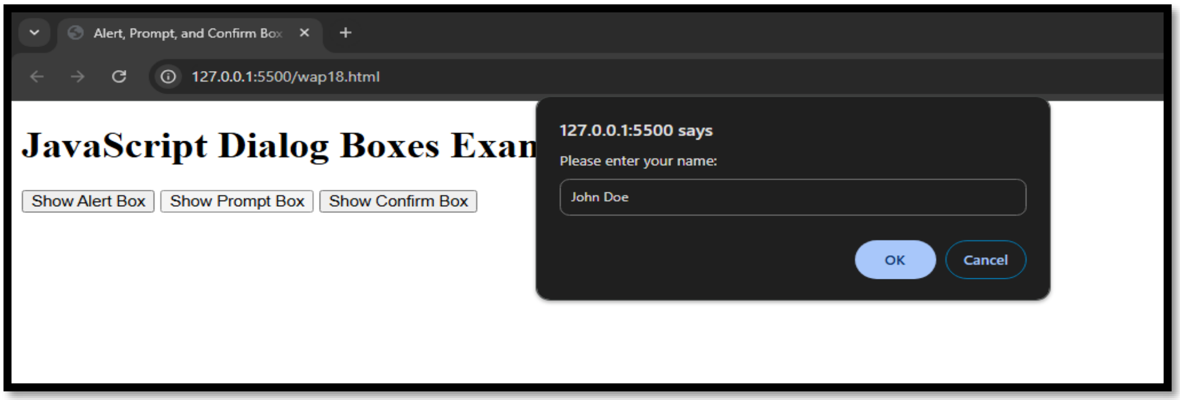
Output :



JavaScript Dialog Boxes Exan

127.0.0.1:5500 says

This is an alert message!

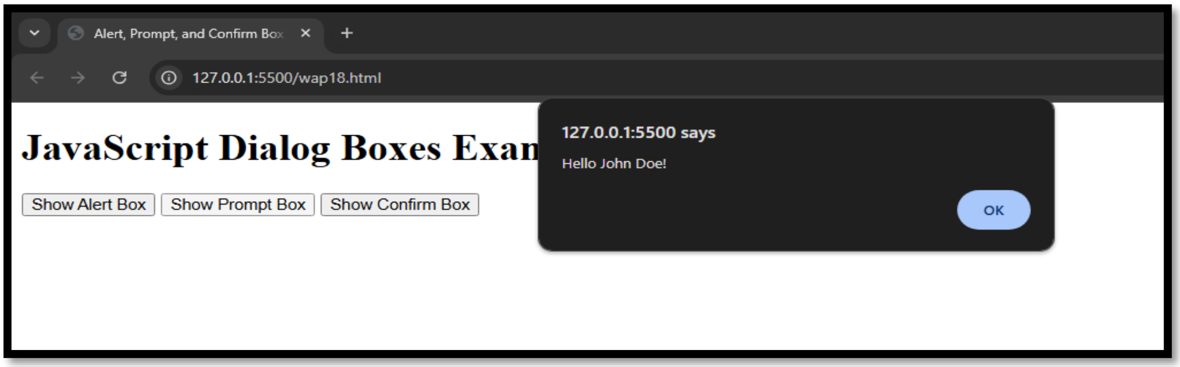


JavaScript Dialog Boxes Exan

127.0.0.1:5500 says

Please enter your name:

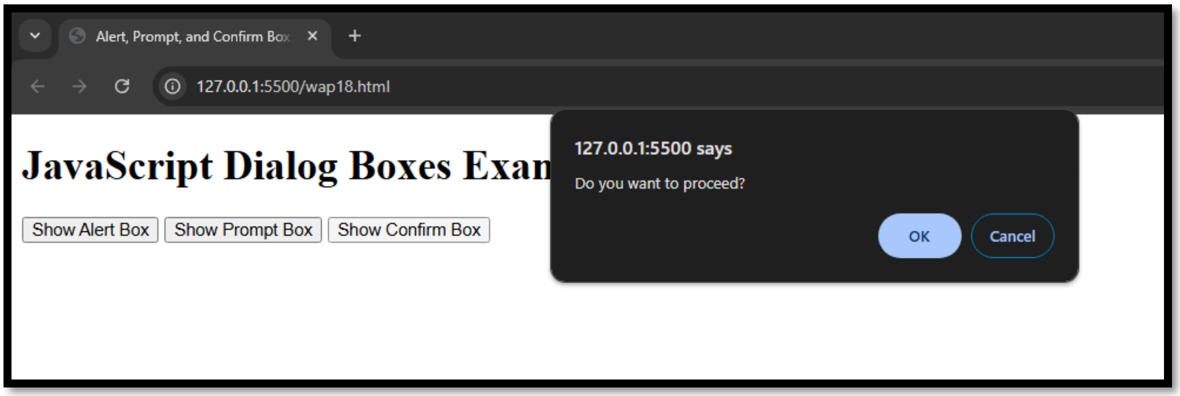
John Doe



JavaScript Dialog Boxes Exan

127.0.0.1:5500 says

Hello John Doe!



JavaScript Dialog Boxes Exan

127.0.0.1:5500 says

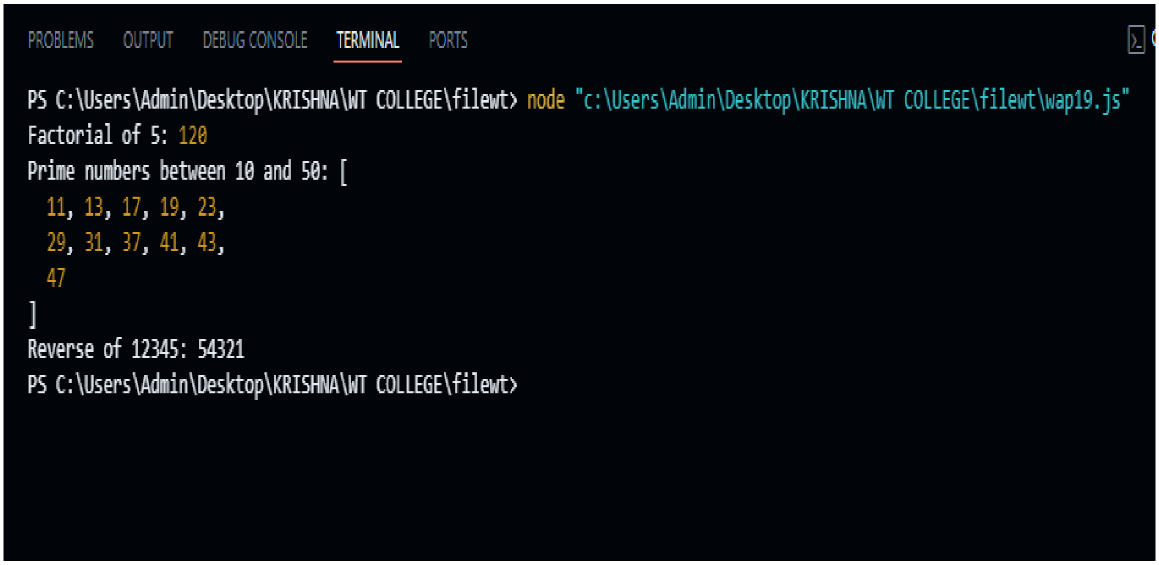
Do you want to proceed?

Program 19: WAP demonstrating : Factorial of number, Prime number in a range, Reverse of a number.

Code :

```
// Function to calculate factorial
function factorial(num) {
  if (num < 0) {
    return "Factorial is not defined for negative numbers.";
  }
  let result = 1;
  for (let i = 1; i <= num; i++) {
    result *= i;
  }
  return result;
}

// Function to find prime numbers in a range
function findPrimes(start, end) {
  let primes = [];

  for (let num = start; num <= end; num++) {
    if (num > 1) {
      let isPrime = true;
      for (let i = 2; i <= Math.sqrt(num); i++) {
        if (num % i === 0) {
          isPrime = false;
          break;
        }
      }
      if (isPrime) {
        primes.push(num);
      }
    }
  }
  return primes;
}

// Function to reverse a number
function reverseNumber(num) {
   return num.toString().split('').reverse().join('');
}

   console.log("Factorial of 5:", factorial(5));
console.log("Prime numbers between 10 and 50:", findPrimes(10, 50));
console.log("Reverse of 12345:", reverseNumber(12345));
```

Output :



```
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt> node "c:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt\wap19.js"
Factorial of 5: 120
Prime numbers between 10 and 50: [
  11, 13, 17, 19, 23,
  29, 31, 37, 41, 43,
  47
]
Reverse of 12345: 54321
PS C:\Users\Admin\Desktop\KRISHNA\WT COLLEGE\filewt>
```

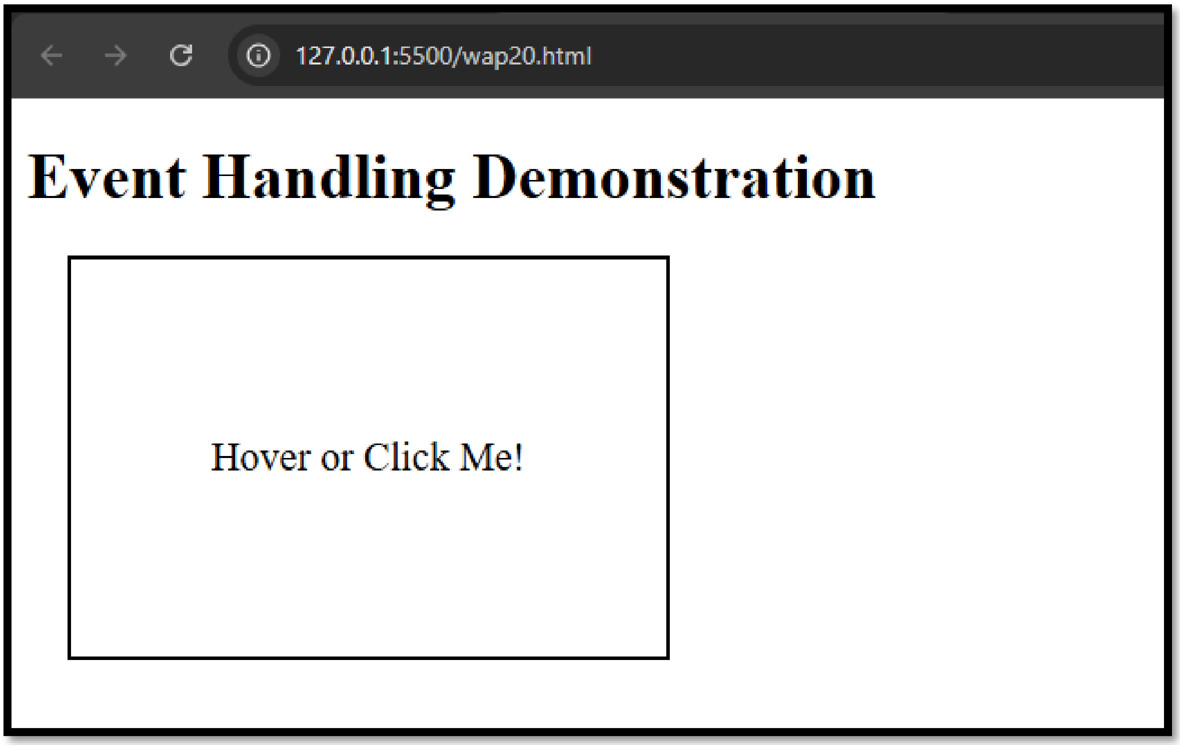
Program 20: WAP to implement event handling using onclick, mouseover, mouseout, mousein, doubleclick, mousemove

Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Event Handling Example</title>
   <style>
      #eventBox {
         width: 300px;
         height: 200px;
         border: 2px solid #000;
         margin: 20px;
         text-align: center;
         line-height: 200px;
         font-size: 20px;
         transition: background-color 0.3s;
      }
   </style>
</head>
<body>

<h1>Event Handling Demonstration</h1>

<div id="eventBox">Hover or Click Me!</div>

<script>
   const eventBox = document.getElementById('eventBox');

   // Onclick Event
   eventBox.onclick = function() {
      alert('Box clicked!');
   };

   // Mouseover Event
   eventBox.onmouseover = function() {
```

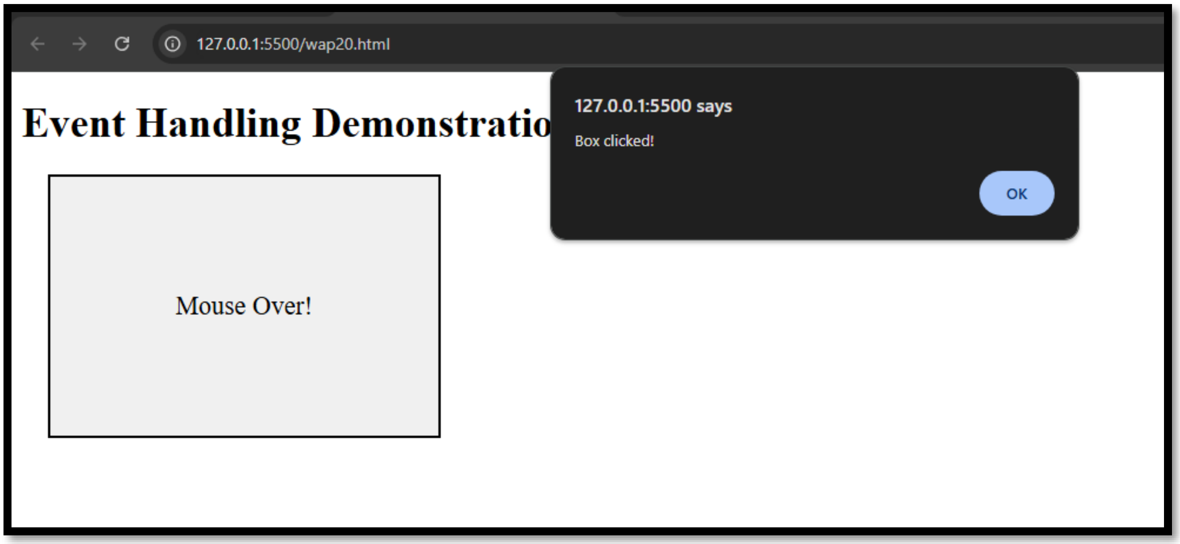
```
            eventBox.style.backgroundColor = '#f0f0f0';
            eventBox.innerHTML = 'Mouse Over!';
        };

        // Mouseout Event
        eventBox.onmouseout = function() {
            eventBox.style.backgroundColor = '#fff';
            eventBox.innerHTML = 'Hover or Click Me!';
        };

        // Mouseenter Event
        eventBox.onmouseenter = function() {
            console.log('Mouse entered the box.');
        };

        // Double Click Event
        eventBox.ondblclick = function() {
            alert('Box double-clicked!');
        };

        // Mousemove Event
        eventBox.onmousemove = function(event) {
            const x = event.clientX - eventBox.getBoundingClientRect().left;
            const y = event.clientY - eventBox.getBoundingClientRect().top;
            console.log(`Mouse position: X: ${x}, Y: ${y}`);
        };
    </script>
    </body>
    </html>
```

Output :


127.0.0.1:5500/wap20.html

Event Handling Demonstration

Hover or Click Me!


127.0.0.1:5500/wap20.html

Event Handling Demonstratio

Mouse Over!

127.0.0.1:5500 says

Box clicked!

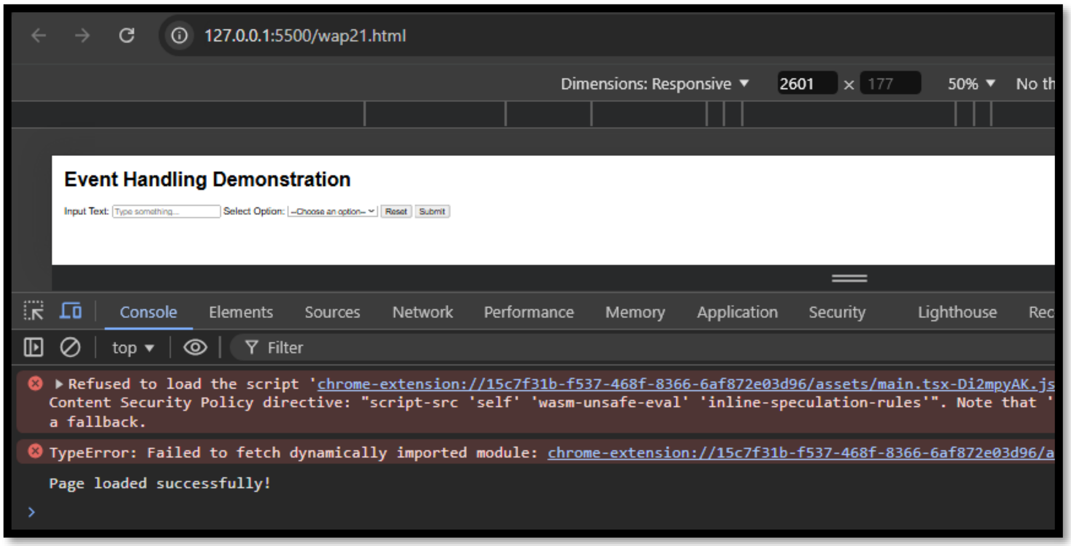
Program 21 : WAP in JavaScript to demonstrate Blur, Focus, Reset, Submit, Change, load.

Code :

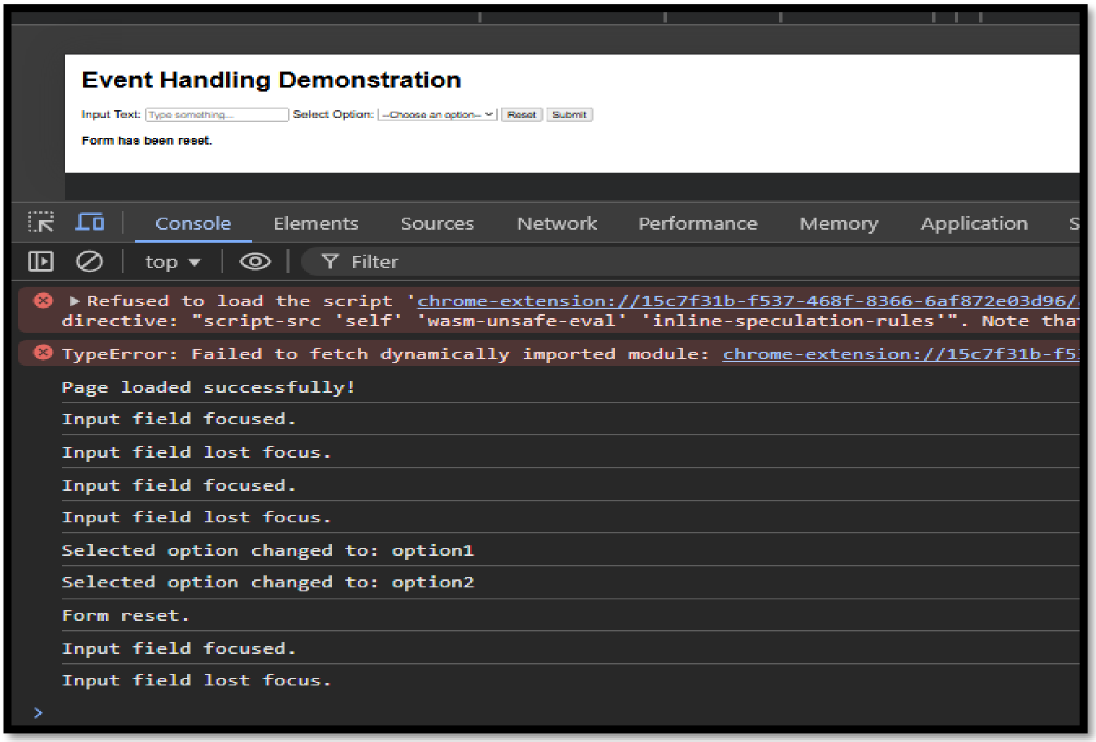
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Event Handling Example</title>
   <style>
      body {
         font-family: Arial, sans-serif;
         margin: 20px;
      }
      #message {
         margin-top: 20px;
         font-weight: bold;
      }
   </style>
</head>
<body>

<h1>Event Handling Demonstration</h1>

<form id="myForm">
   <label for="textInput">Input Text:</label>
   <input type="text" id="textInput" placeholder="Type something..." required>

   <label for="selectInput">Select Option:</label>
   <select id="selectInput">
      <option value="">--Choose an option--</option>
      <option value="option1">Option 1</option>
      <option value="option2">Option 2</option>
   </select>

   <button type="reset">Reset</button>
   <input type="submit" value="Submit">
</form>
```

```
<div id="message"></div>

<script>
    // Load event
    window.onload = function() {
        console.log("Page loaded successfully!");
    };

    // Focus event
    document.getElementById('textInput').onfocus = function() {
        this.style.backgroundColor = '#e0f7fa';
        console.log("Input field focused.");
    };

    // Blur event
    document.getElementById('textInput').onblur = function() {
        this.style.backgroundColor = '';
        console.log("Input field lost focus.");
    };

    // Change event
    document.getElementById('selectInput').onchange = function() {
        console.log("Selected option changed to: " + this.value);
    };

    // Submit event
    document.getElementById('myForm').onsubmit = function(event) {
        event.preventDefault(); // Prevent actual form submission for demonstration
        const messageDiv = document.getElementById('message');
        messageDiv.innerHTML = "Form submitted successfully! Input: " +
document.getElementById('textInput').value;
        console.log("Form submitted.");
    };

    // Reset event
    document.getElementById('myForm').onreset = function() {
        const messageDiv = document.getElementById('message');
        messageDiv.innerHTML = "Form has been reset.";
        console.log("Form reset.");
    };
</script>

</body>
</html>
```

Output :

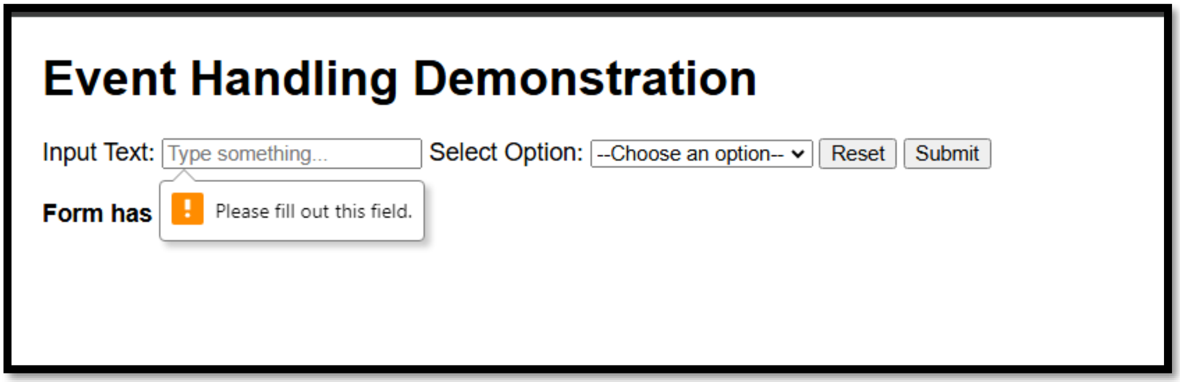


```
▸ Refused to load the script 'chrome-extension://15c7f31b-f537-468f-8366-6af872e03d96/assets/main.tsx-Di2mpyAK.js
Content Security Policy directive: "script-src 'self' 'wasm-unsafe-eval' 'inline-speculation-rules'". Note that '
a fallback.
TypeError: Failed to fetch dynamically imported module: chrome-extension://15c7f31b-f537-468f-8366-6af872e03d96/a
Page loaded successfully!
>
```


Event Handling Demonstration
Input Text: Type something...
Select Option: --Choose an option-- ▾
Reset
Submit
Console
Elements
Sources
Network
Performance
Memory
Application
S
top ▾
Filter

Form has been reset.

```
▸ Refused to load the script 'chrome-extension://15c7f31b-f537-468f-8366-6af872e03d96/
directive: "script-src 'self' 'wasm-unsafe-eval' 'inline-speculation-rules'". Note tha
TypeError: Failed to fetch dynamically imported module: chrome-extension://15c7f31b-f5
Page loaded successfully!
Input field focused.
Input field lost focus.
Input field focused.
Input field lost focus.
Selected option changed to: option1
Selected option changed to: option2
Form reset.
Input field focused.
Input field lost focus.
>
```



Event Handling Demonstration

Input Text: Type something... Select Option: --Choose an option--

Form has

Please fill out this field.

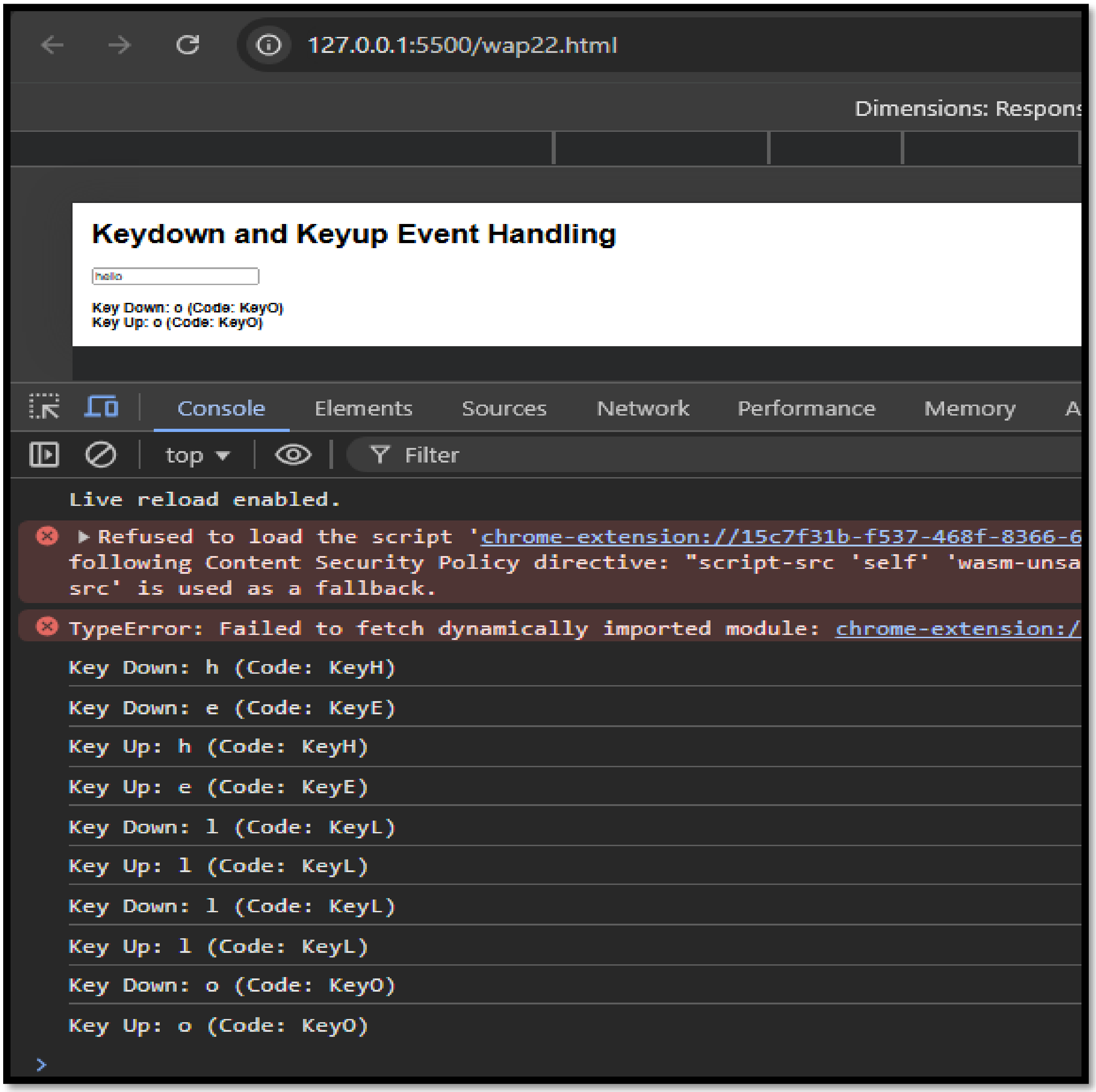
Program 22 : WAP to implement event handling using : keydown, keyup.

Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Keydown and Keyup Event Handling</title>
   <style>
      body {
         font-family: Arial, sans-serif;
         margin: 20px;
      }
      #output {
         margin-top: 20px;
         font-weight: bold;
      }
   </style>
</head>
<body>

<h1>Keydown and Keyup Event Handling</h1>

<input type="text" id="textInput" placeholder="Type something..." />
<div id="output"></div>

<script>
   const textInput = document.getElementById('textInput');
   const output = document.getElementById('output');

   // Keydown event
   textInput.addEventListener('keydown', function(event) {
      output.innerHTML = `Key Down: ${event.key} (Code: ${event.code})`;
      console.log(`Key Down: ${event.key} (Code: ${event.code})`);
   });

   // Keyup event
```

```
    textInput.addEventListener('keyup', function(event) {
        output.innerHTML += `<br>Key Up: ${event.key} (Code: ${event.code})`;
        console.log(`Key Up: ${event.key} (Code: ${event.code})`);
    });
</script>

</body>
</html>
```

Output :



Keydown and Keyup Event Handling

hello

Key Down: o (Code: KeyO)
Key Up: o (Code: KeyO)

```
Live reload enabled.
▶ Refused to load the script 'chrome-extension://15c7f31b-f537-468f-8366-6
following Content Security Policy directive: "script-src 'self' 'wasm-unsa
src' is used as a fallback.
TypeError: Failed to fetch dynamically imported module: chrome-extension:/
Key Down: h (Code: KeyH)
Key Down: e (Code: KeyE)
Key Up: h (Code: KeyH)
Key Up: e (Code: KeyE)
Key Down: l (Code: KeyL)
Key Up: l (Code: KeyL)
Key Down: l (Code: KeyL)
Key Up: l (Code: KeyL)
Key Down: o (Code: KeyO)
Key Up: o (Code: KeyO)
>
```

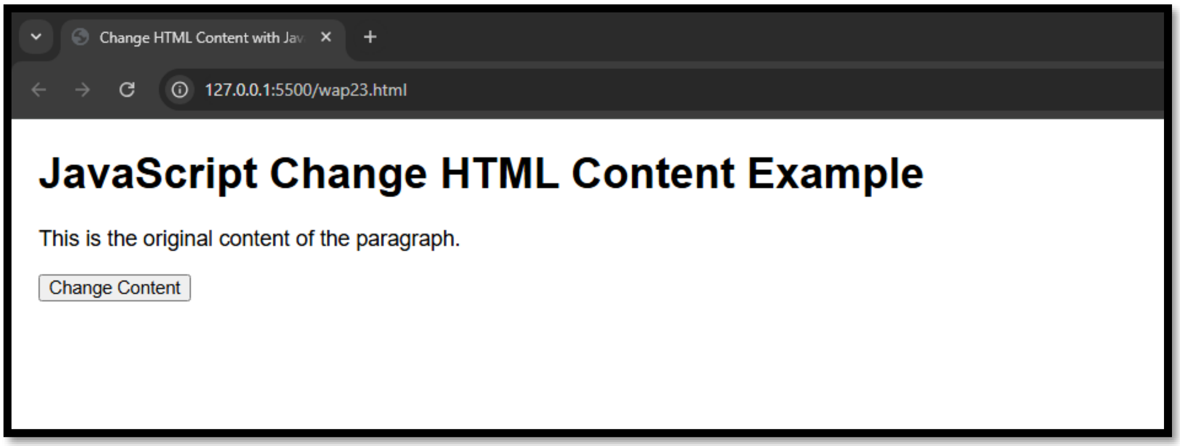
Program 23: WAP to show JS can change HTML content.

Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Change HTML Content with JavaScript</title>
   <style>
      body {
         font-family: Arial, sans-serif;
         margin: 20px;
      }
      #output {
         margin-top: 20px;
         font-weight: bold;
      }
   </style>
</head>
<body>

<h1>JavaScript Change HTML Content Example</h1>

<p id="myParagraph">This is the original content of the paragraph.</p>

<button id="changeContentButton">Change Content</button>

<div id="output"></div>

<script>
   // Function to change the content of the paragraph
   function changeContent() {
      document.getElementById("myParagraph").innerHTML = "This is the new content
of the paragraph!";
      document.getElementById("output").innerHTML = "The paragraph content has
been changed.";
   }
```

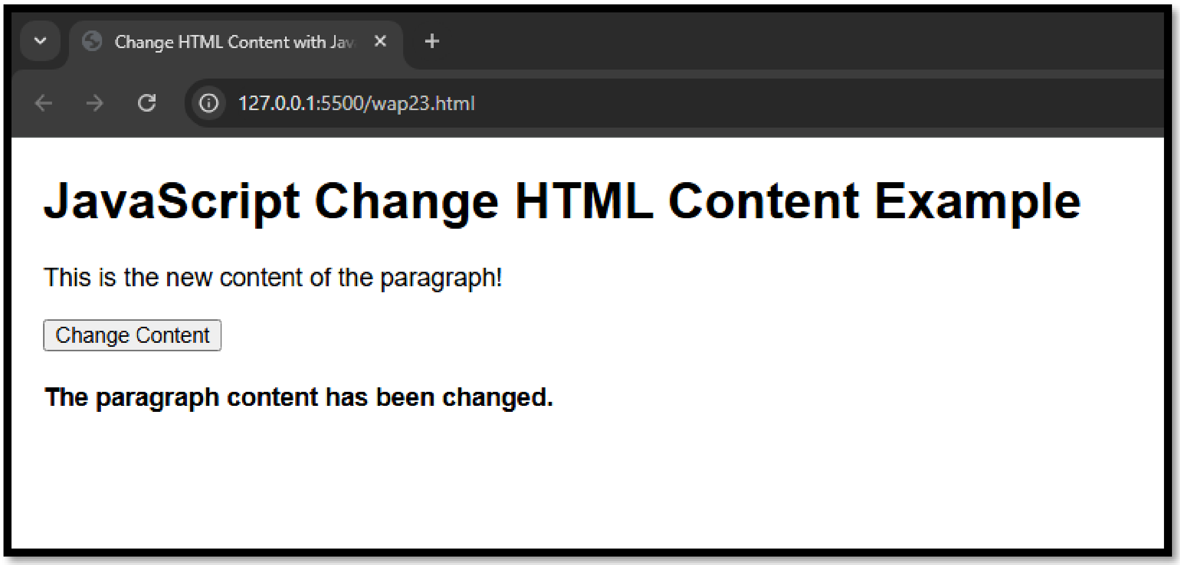
```
    // Adding event listener to the button
    document.getElementById("changeContentButton").onclick = changeContent;
</script>

</body>
</html>
```

Output :



JavaScript Change HTML Content Example

This is the original content of the paragraph.



JavaScript Change HTML Content Example

This is the new content of the paragraph!

The paragraph content has been changed.

Program 24: WAP to show JS can change HTML attribute value.

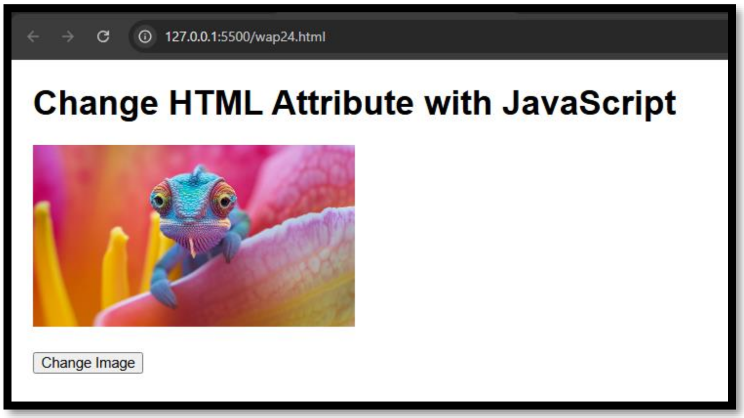
Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Change HTML Attribute Example</title>
   <style>
      body {
         font-family: Arial, sans-serif;
         margin: 20px;
      }
      #output {
         margin-top: 20px;
         font-weight: bold;
      }
   </style>
</head>
<body>

<h1>Change HTML Attribute with JavaScript</h1>

<img id="myImage"
src="https://letsenhance.io/static/8f5e523ee6b2479e26ecc91b9c25261e/1015f/MainAf
ter.jpg" alt="Placeholder Image" width="300" />
<br><br>
<button id="changeImageButton">Change Image</button>

<div id="output"></div>

<script>
   // Function to change the src attribute of the image
   function changeImage() {
      const imgElement = document.getElementById("myImage");
      imgElement.src = "https://i0.wp.com/picjumbo.com/wp-
content/uploads/silhouette-of-young-blonde-with-short-hair-on-orange-background-
free-image.jpeg?h=800&quality=80"; // New image URL
```

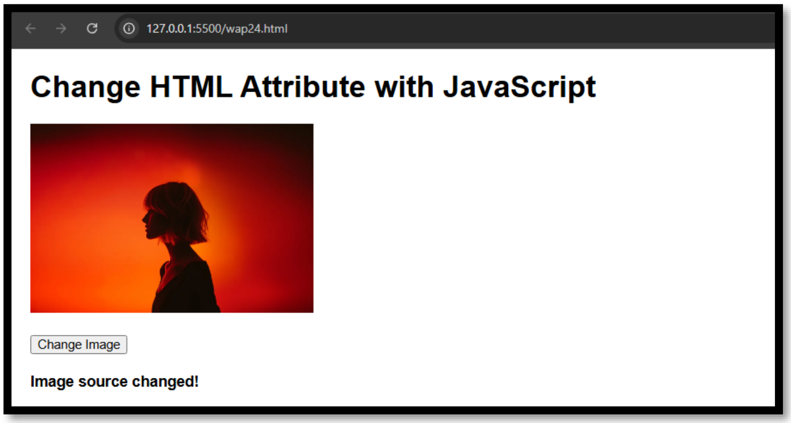
```
      document.getElementById("output").innerHTML = "Image source changed!";
    }

    // Adding event listener to the button
    document.getElementById("changeImageButton").onclick = changeImage;
</script>

</body>
</html>
```

```
    // Adding event listener to the button
    document.getElementById("changeImageButton").onclick = changeImage;
</script>

</body>
</html>
```

Output :



Change HTML Attribute with JavaScript



Change HTML Attribute with JavaScript

Image source changed!

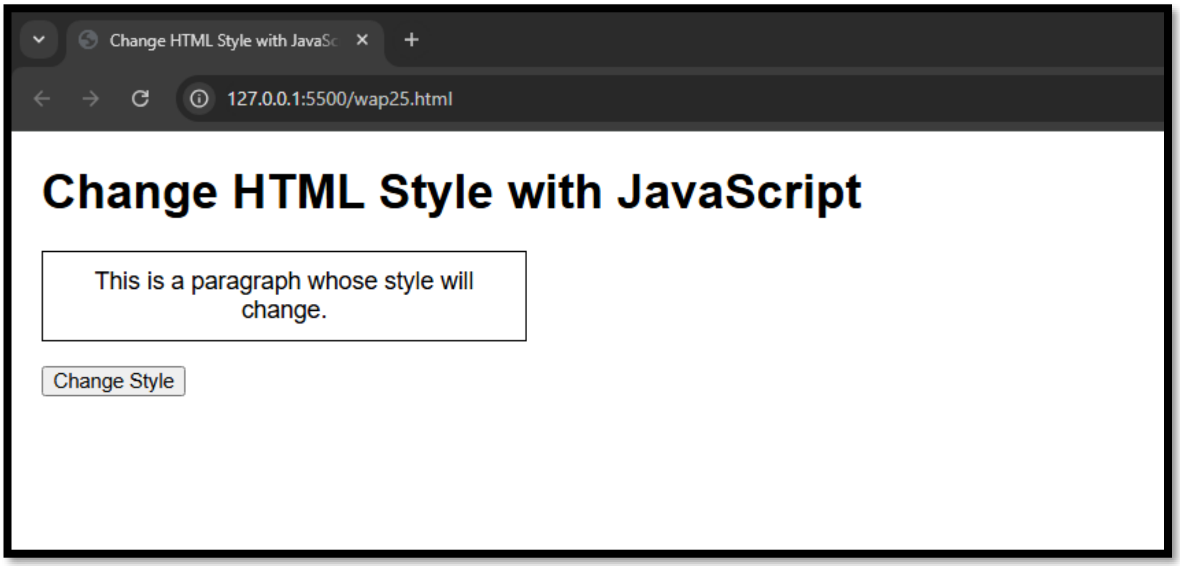
Program 25 : WAP to show JS can change HTML style.

Code :

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Change HTML Style with JavaScript</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 20px;
    }
    #myParagraph {
      padding: 10px;
      border: 1px solid #000;
      width: 300px;
      text-align: center;
    }
  </style>
</head>
<body>

<h1>Change HTML Style with JavaScript</h1>

<p id="myParagraph">This is a paragraph whose style will change.</p>

<button id="changeStyleButton">Change Style</button>

<script>
  // Function to change the style of the paragraph
  function changeStyle() {
    const paragraph = document.getElementById("myParagraph");
    paragraph.style.backgroundColor = "#FF5733"; // Change background color
    paragraph.style.color = "#FFFFFF"; // Change text color
    paragraph.style.fontSize = "20px"; // Change font size
    paragraph.style.borderRadius = "5px"; // Add border radius
  }
```

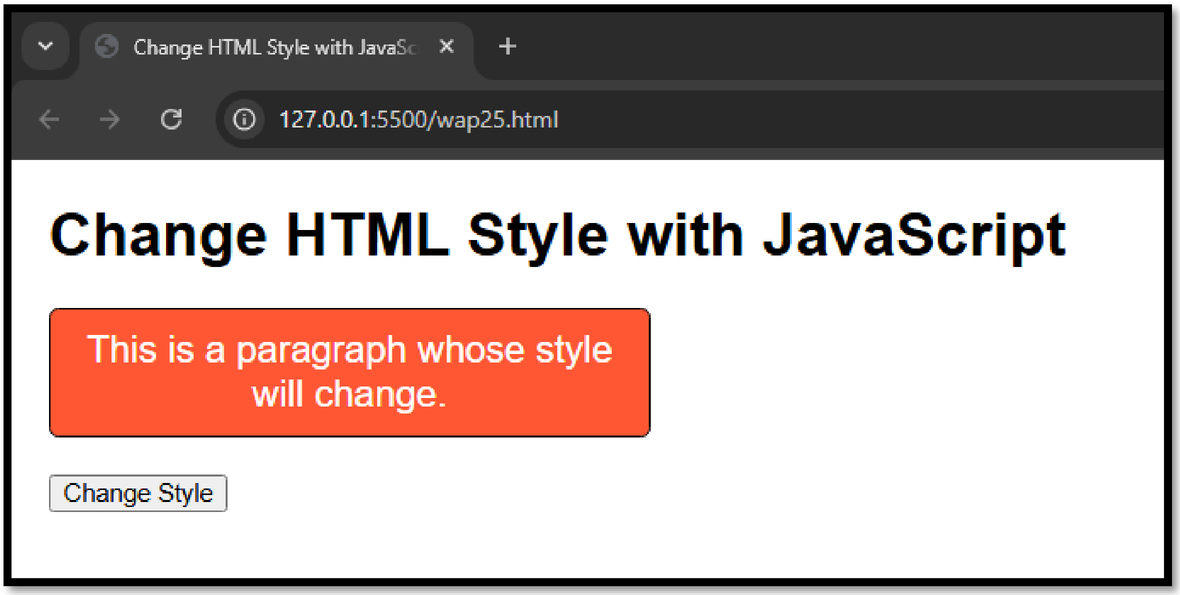
```
    // Adding event listener to the button
    document.getElementById("changeStyleButton").onclick = changeStyle;
</script>

</body>
</html>
```

Output :



Change HTML Style with JavaScript

This is a paragraph whose style will change.



Change HTML Style with JavaScript

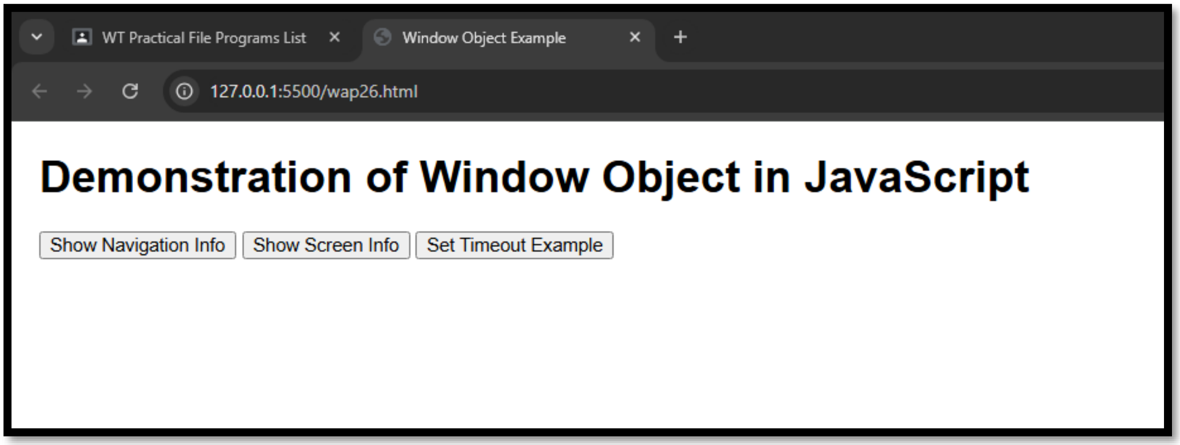
This is a paragraph whose style will change.

Program 26 : WAP to show use of Window Object in JS.
Navigation object, JavaScript Screen Object, Set-time function.
Code :

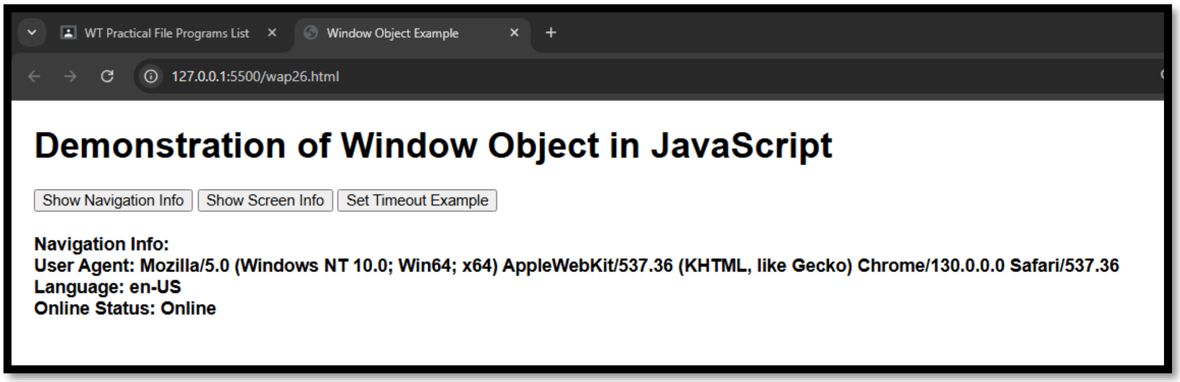
```
<!DOCTYPE html>
<html lang="en">
<head>
   <meta charset="UTF-8">
   <meta name="viewport" content="width=device-width, initial-scale=1.0">
   <title>Window Object Example</title>
   <style>
      body {
         font-family: Arial, sans-serif;
         margin: 20px;
      }
      #output {
         margin-top: 20px;
         font-weight: bold;
      }
   </style>
</head>
<body>

<h1>Demonstration of Window Object in JavaScript</h1>

<button id="navButton">Show Navigation Info</button>
<button id="screenButton">Show Screen Info</button>
<button id="setTimeoutButton">Set Timeout Example</button>

<div id="output"></div>

<script>
   // Function to show Navigation object information
   function showNavigationInfo() {
      const navInfo = `
         User Agent: ${navigator.userAgent}<br>
         Language: ${navigator.language}<br>
         Online Status: ${navigator.onLine ? 'Online' : 'Offline'}
      `;
```

```
        document.getElementById("output").innerHTML = `<strong>Navigation
Info:</strong><br>${navInfo}`;
    }

    // Function to show Screen object information
    function showScreenInfo() {
        const screenInfo = `
            Screen Width: ${screen.width}px<br>
            Screen Height: ${screen.height}px<br>
            Available Width: ${screen.availWidth}px<br>
            Available Height: ${screen.availHeight}px<br>
            Color Depth: ${screen.colorDepth} bits<br>
            Pixel Depth: ${screen.pixelDepth} bits
        `;
        document.getElementById("output").innerHTML = `<strong>Screen
Info:</strong><br>${screenInfo}`;
    }

    // Function to demonstrate setTimeout
    function setTimeoutExample() {
        document.getElementById("output").innerHTML = "Waiting for 3 seconds...";
        setTimeout(() => {
            document.getElementById("output").innerHTML = "3 seconds have passed!";
        }, 3000);
    }

    // Adding event listeners to buttons
    document.getElementById("navButton").onclick = showNavigationInfo;
    document.getElementById("screenButton").onclick = showScreenInfo;
    document.getElementById("setTimeoutButton").onclick = setTimeoutExample;
</script>

</body>
</html>
```

Output :

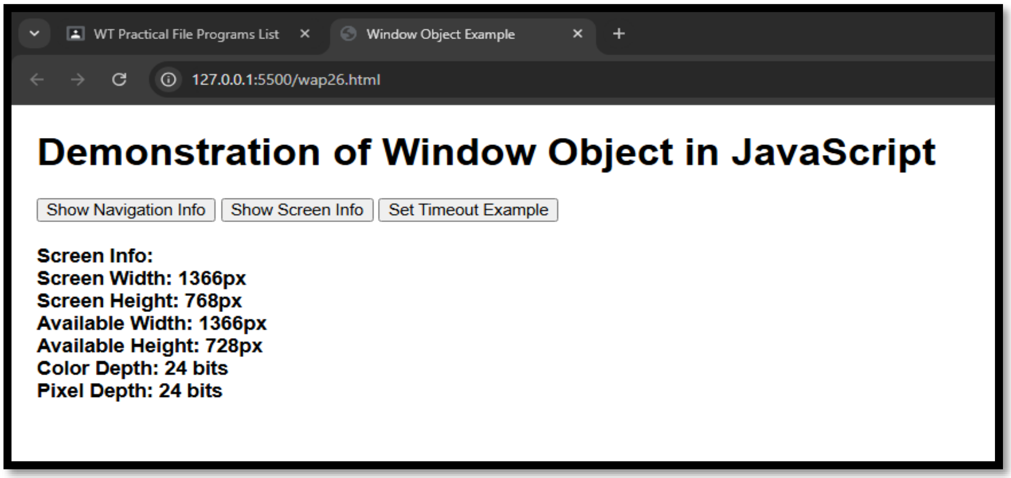


Demonstration of Window Object in JavaScript



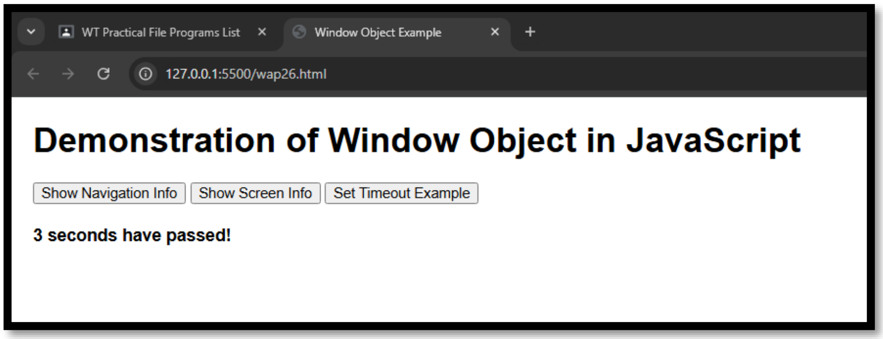
Demonstration of Window Object in JavaScript

```
Navigation Info:
User Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/130.0.0.0 Safari/537.36
Language: en-US
Online Status: Online
```



Demonstration of Window Object in JavaScript

```
Screen Info:
Screen Width: 1366px
Screen Height: 768px
Available Width: 1366px
Available Height: 728px
Color Depth: 24 bits
Pixel Depth: 24 bits
```



Demonstration of Window Object in JavaScript

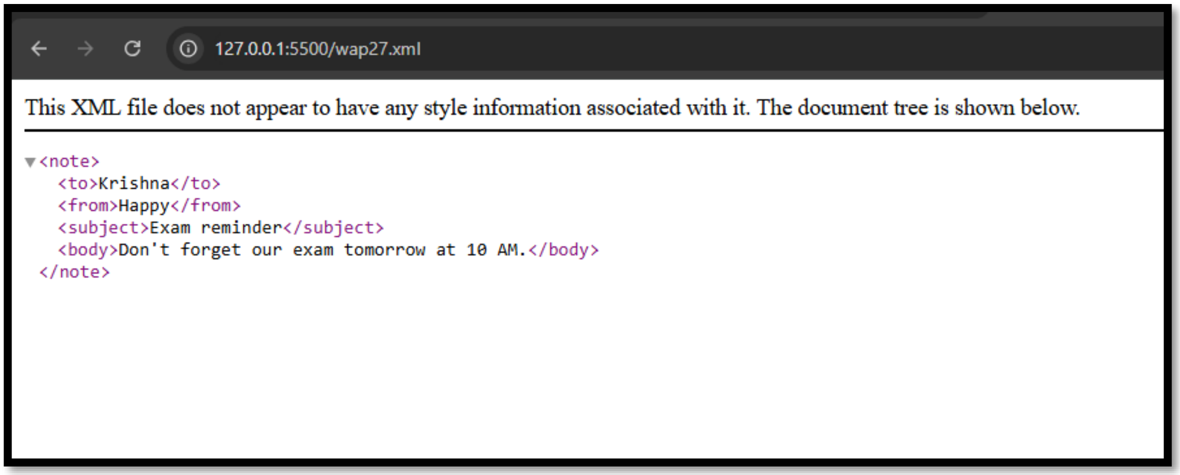
```
3 seconds have passed!
```

Program 27 : Create an XML file containing note as the root element and to, from, subject and body as its child element.

Code :

```
<?xml version="1.0" encoding="UTF-8"?>
<note>
   <to>Krishna</to>
   <from>Happy</from>
   <subject>Exam reminder</subject>
   <body>Don't forget our exam tomorrow at 10 AM.</body>
</note>
```

OUTPUT :



This XML file does not appear to have any style information associated with it. The document tree is shown below.

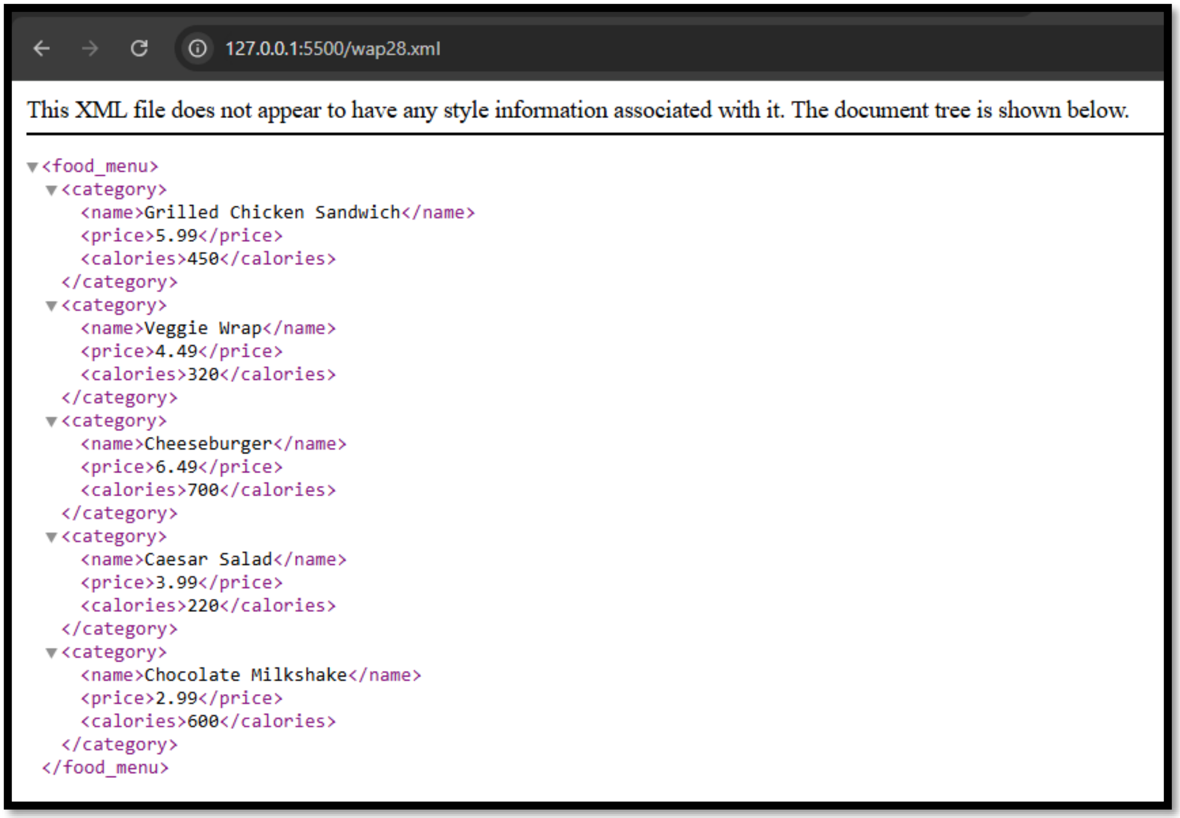
```
▼<note>
   <to>Krishna</to>
   <from>Happy</from>
   <subject>Exam reminder</subject>
   <body>Don't forget our exam tomorrow at 10 AM.</body>
 </note>
```

Program 28 : Create a well-formed XML file having 5 tags-food menu, category, name, price and calories. Food_menu is the main parent element that contains category as the child element. Category is the parent element of name, price and calories.

Code :

```
<?xml version="1.0" encoding="UTF-8"?>
<food_menu>
   <category>
      <name>Grilled Chicken Sandwich</name>
      <price>5.99</price>
      <calories>450</calories>
   </category>
   <category>
      <name>Veggie Wrap</name>
      <price>4.49</price>
      <calories>320</calories>
   </category>
   <category>
      <name>Cheeseburger</name>
      <price>6.49</price>
      <calories>700</calories>
   </category>
   <category>
      <name>Caesar Salad</name>
      <price>3.99</price>
      <calories>220</calories>
   </category>
   <category>
      <name>Chocolate Milkshake</name>
      <price>2.99</price>
      <calories>600</calories>
   </category>
</food_menu>
```

OUTPUT :



This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<food_menu>
  ▼<category>
     <name>Grilled Chicken Sandwich</name>
     <price>5.99</price>
     <calories>450</calories>
   </category>
  ▼<category>
     <name>Veggie Wrap</name>
     <price>4.49</price>
     <calories>320</calories>
   </category>
  ▼<category>
     <name>Cheeseburger</name>
     <price>6.49</price>
     <calories>700</calories>
   </category>
  ▼<category>
     <name>Caesar Salad</name>
     <price>3.99</price>
     <calories>220</calories>
   </category>
  ▼<category>
     <name>Chocolate Milkshake</name>
     <price>2.99</price>
     <calories>600</calories>
   </category>
 </food_menu>
```

Program 29 : WAP to display the bookstore details in XML with CSS and internal DTD.

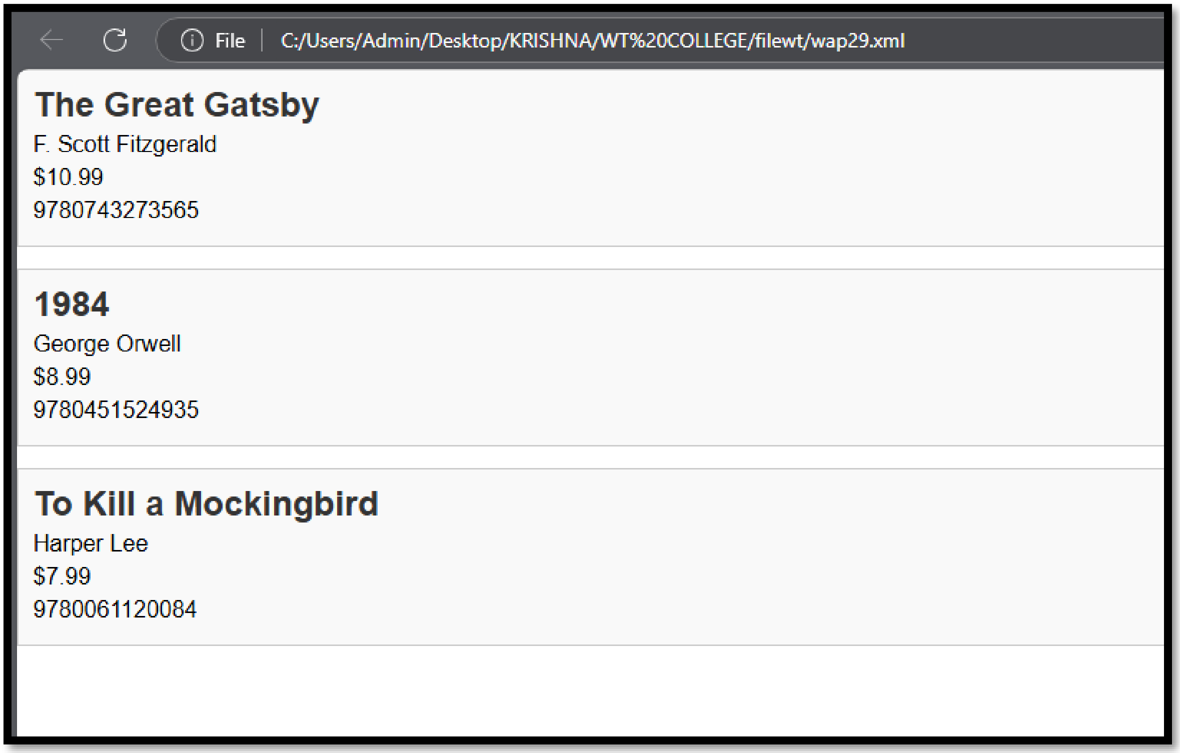
CODE :

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE bookstore [
  <!ELEMENT bookstore (book+)>
  <!ELEMENT book (title, author, price, isbn)>
  <!ELEMENT title (#PCDATA)>
  <!ELEMENT author (#PCDATA)>
  <!ELEMENT price (#PCDATA)>
  <!ELEMENT isbn (#PCDATA)>
]>
<?xml-stylesheet type="text/css" href="style12.css"?>
<bookstore>
  <book>
    <title>The Great Gatsby</title>
    <author>F. Scott Fitzgerald</author>
    <price>$10.99</price>
    <isbn>9780743273565</isbn>
  </book>
  <book>
    <title>1984</title>
    <author>George Orwell</author>
    <price>$8.99</price>
    <isbn>9780451524935</isbn>
  </book>
  <book>
    <title>To Kill a Mockingbird</title>
    <author>Harper Lee</author>
    <price>$7.99</price>
    <isbn>9780061120084</isbn>
  </book>
</bookstore>
```

STYLE12.css

```
bookstore {
  display: block;
  font-family: Arial, sans-serif;
}
book {
  display: block;
  margin-bottom: 15px;
  padding: 10px;
  border: 1px solid #ccc;
  background-color: #f9f9f9;
}
title {
  display: block;
  font-size: 1.5em;
  font-weight: bold;
  color: #333;
}
author, price, isbn {
  display: block;
  margin: 5px 0;
}
```

OUTPUT :



File
C:/Users/Admin/Desktop/KRISHNA/WT%20COLLEGE/filewt/wap29.xml

```
The Great Gatsby
F. Scott Fitzgerald
$10.99
9780743273565

1984
George Orwell
$8.99
9780451524935

To Kill a Mockingbird
Harper Lee
$7.99
9780061120084
```

Program 30 : Create Valid XML doc using external DTD.

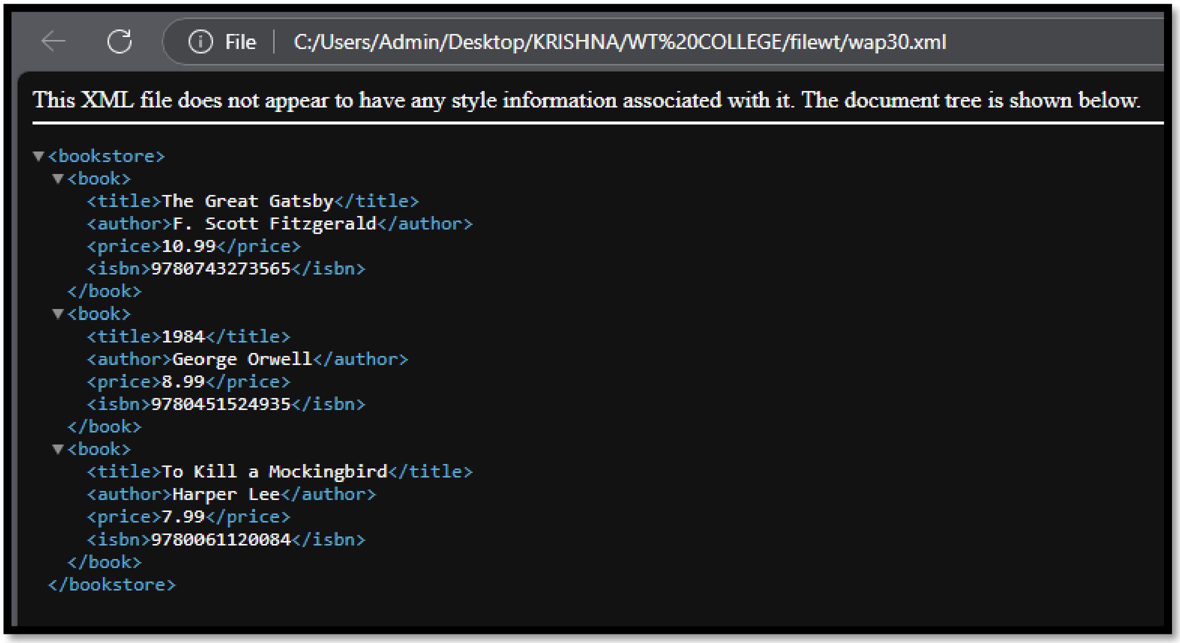
CODE :

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE bookstore SYSTEM "bookstore.dtd">
<bookstore>
  <book>
    <title>The Great Gatsby</title>
    <author>F. Scott Fitzgerald</author>
    <price>10.99</price>
    <isbn>9780743273565</isbn>
  </book>
  <book>
    <title>1984</title>
    <author>George Orwell</author>
    <price>8.99</price>
    <isbn>9780451524935</isbn>
  </book>
  <book>
    <title>To Kill a Mockingbird</title>
    <author>Harper Lee</author>
    <price>7.99</price>
    <isbn>9780061120084</isbn>
  </book>
</bookstore>
```

BOOKSTORE.DTD :

```
<!ELEMENT bookstore (book+)>
<!ELEMENT book (title, author, price, isbn)>
<!ELEMENT title (#PCDATA)>
<!ELEMENT author (#PCDATA)>
<!ELEMENT price (#PCDATA)>
<!ELEMENT isbn (#PCDATA)>
```

OUTPUT :



File
C:/Users/Admin/Desktop/KRISHNA/WT%20COLLEGE/filewt/wap30.xml

This XML file does not appear to have any style information associated with it. The document tree is shown below.

```
▼<bookstore>
  ▼<book>
     <title>The Great Gatsby</title>
     <author>F. Scott Fitzgerald</author>
     <price>10.99</price>
     <isbn>9780743273565</isbn>
   </book>
  ▼<book>
     <title>1984</title>
     <author>George Orwell</author>
     <price>8.99</price>
     <isbn>9780451524935</isbn>
   </book>
  ▼<book>
     <title>To Kill a Mockingbird</title>
     <author>Harper Lee</author>
     <price>7.99</price>
     <isbn>9780061120084</isbn>
   </book>
 </bookstore>
```