Experiment:1

1. Lists, Links and Images

a) Write a HTML program, to explain the working of lists.

Note: It should have ordered list, unordered list, nested lists and order list in an unordered list and definition lists

Source Code: lists.html

```
<!DOCTYPE html>
<html>
<head>
    <title>HTML Lists </title>
</head>
<body>
    <h1>HTML Lists</h1>
    <!-- Ordered List -->
    <h2>1. Ordered List (Numbered)</h2>
    <p>Steps to create a simple website:</p>
    <ol>
        <li>Create an HTML file</li>
        <li>Add a head and body section</li>
        <li>Write content using tags</li>
        <li>Save and open in browser</li>
    </ol>
    <!-- Unordered List -->
    <h2>2. Unordered List (Bulleted)</h2>
    <p>Web Technologies :</p>
    <ul type="disc">
        <li>HTML</li>
        <li>CSS</li>
        <li>JavaScript</li>
    </ul>
    <!-- Nested List -->
    <h2>3. Nested List (List inside List)</h2>
    <ul>
        <li>Frontend Technologies
            <ul>
```

```
            <li>HTML</li>
            <li>CSS</li>
            <li>JavaScript</li>
        </ul>
    </li>
    <li>Backend Technologies
        <ul>
            <li>Java</li>
            <li>Python</li>
            <li>PHP</li>
        </ul>
    </li>
</ul>
<!-- Ordered inside Unordered -->
<h2>4. Mixed List (Ordered inside Unordered)</h2>
<ul>
    <li>Frontend Technologies
        <ol>
            <li>HTML</li>
            <li>CSS</li>
            <li>JavaScript</li>
        </ol>
    </li>
    <li>Backend Technologies
        <ol>
            <li>Java</li>
            <li>Python</li>
            <li>PHP</li>
        </ol>
    </li>
</ul>
<!-- Definition List -->
<h2>5. Definition List (Term - Meaning)</h2>
<dl>
    <dt>HTML</dt>
    <dd>Hyper Text Markup Language</dd>
```

```
        <dt>CSS</dt>
        <dd>Cascading Style Sheets</dd>
        <dt>JS</dt>
        <dd>JavaScript</dd>
    </dl>
</body>
</html>
```

Output:

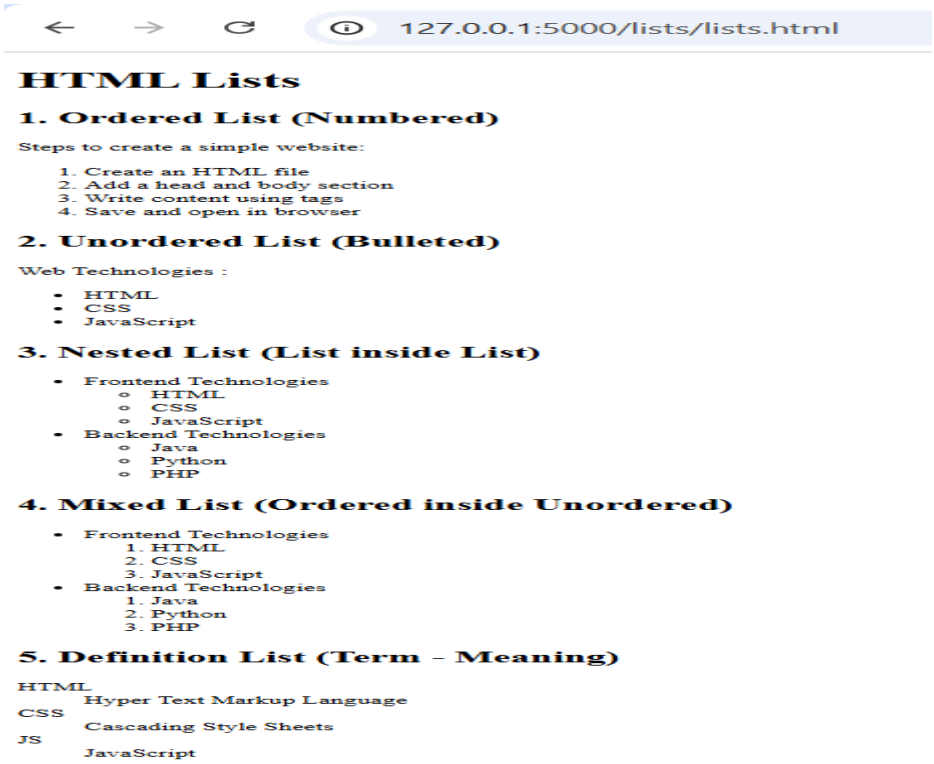


Figure 1: HTML Lists

b) Write a HTML Program, to explain the working of hyperlinks using <a> tag and href, target attributes

Source Code: **hyperlinks.html**

```
<!DOCTYPE html>
<html>
<head>
    <title>Hyperlinks</title>
</head>
<body>
    <h2>Click the links below to visit different websites:</h2>
    <ol>
        <!-- Opens the link in the same tab -->
        <li><a href="https://www.google.com">Google</a><br></li>
        <!-- Opens the link in a new tab or window -->
```

```
          <li><a  href="https://www.hackerrank.com"  target="_blank">Hacker
Rank</a></li>
                            <li><a      href="https://www.kitsguntur.ac.in"
target="_blank">KITS</a></li>
    </ol>
</body>
</html>
```

Output:

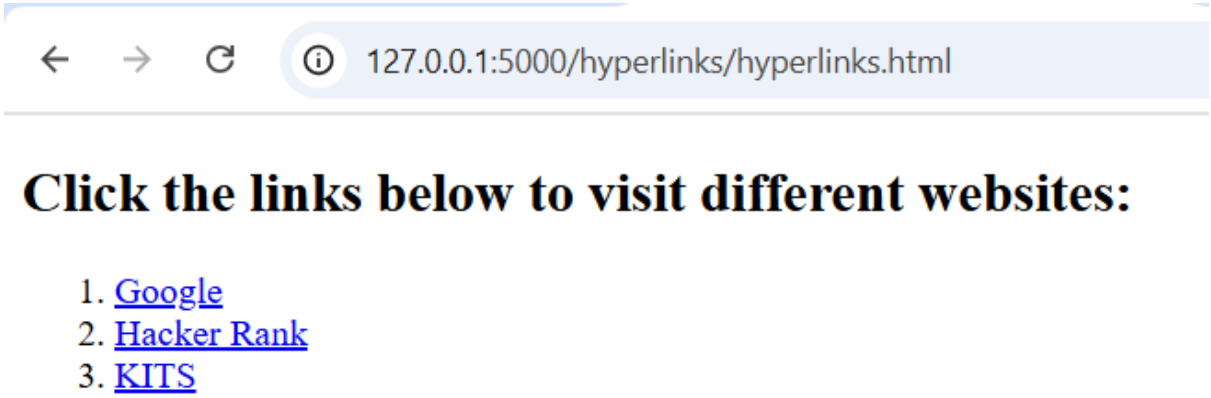


Figure 2: Hyperlinks

c) Create a html document that has your image and your friends image with specific height and width. Also when clicked on the images it should navigate to their respective profiles

Source Code: profile_images.html

```
<!DOCTYPE html>
<html>
<head>
    <title>My Profile and Friend's Profile</title>
</head>
<body>
    <h2>My Profile</h2>
    <a href="my_profile.html" target="_blank">
        <img src="my_image.jpg" alt="My Image" width="200" height="200">
    </a>
    <h2>Friend's Profile</h2>
    <a href="friend_profile.html" target="_blank">
           <img  src="friend_image.jpg"  alt="Friend's  Image"  width="200"
height="200">
```

```
    </a>
</body>
</html>
```

Output:

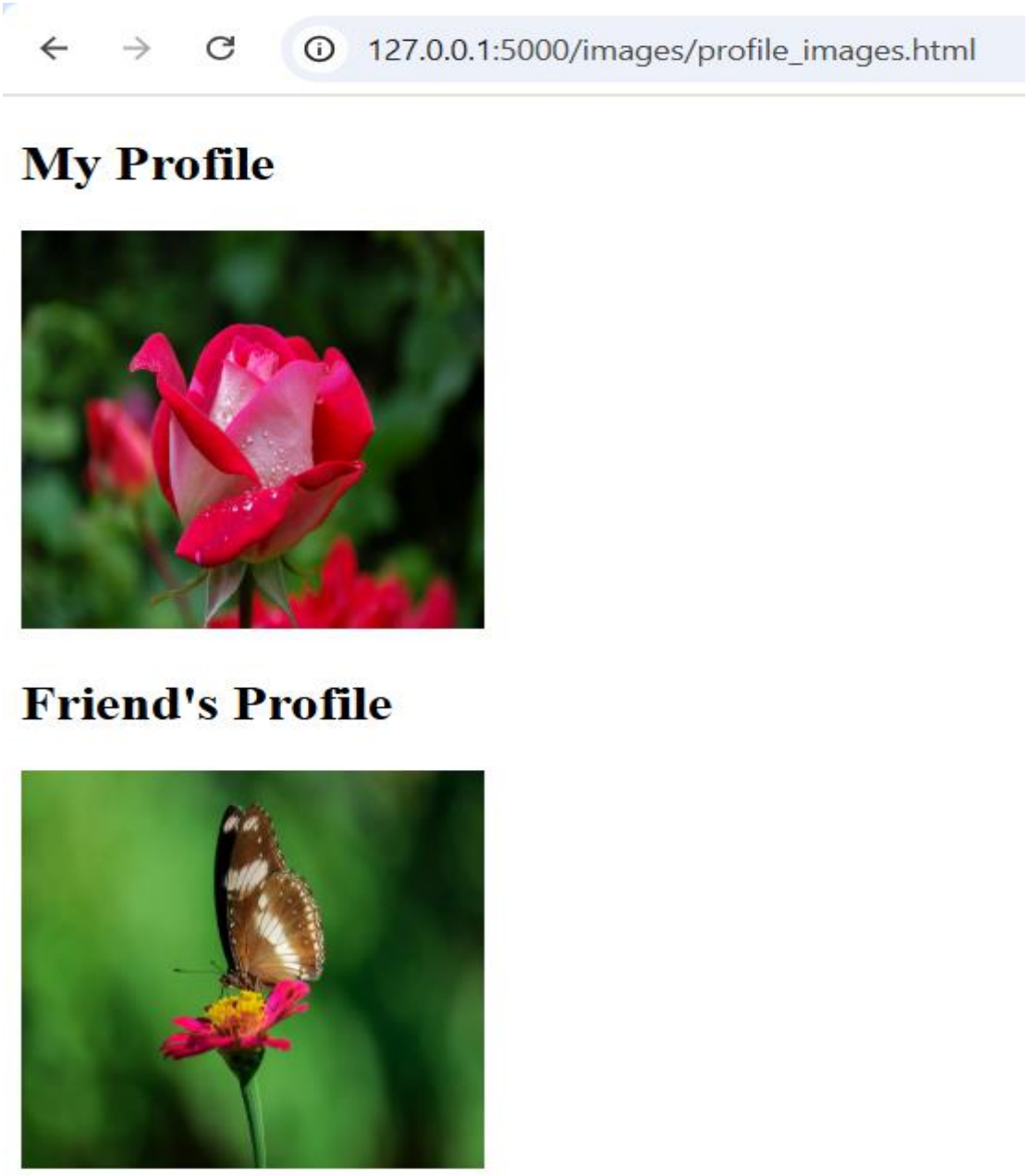


Figure 3: Profile Page

Source Code: my_profile.html

```
<!DOCTYPE html>
<html>
<head>
    <title>My Profile</title>
</head>
```

```
<body>
    <h1>My Profile</h1>
    <img src="my_image.jpg" alt="My Image" width="200" height="200">
    <p>Name: Ravi</p>
    <p>Role: Web Developer</p>
    <p>Location: India</p>
    <a href="profile_images.html">Back to Home</a>
</body>
</html>
```

Output:

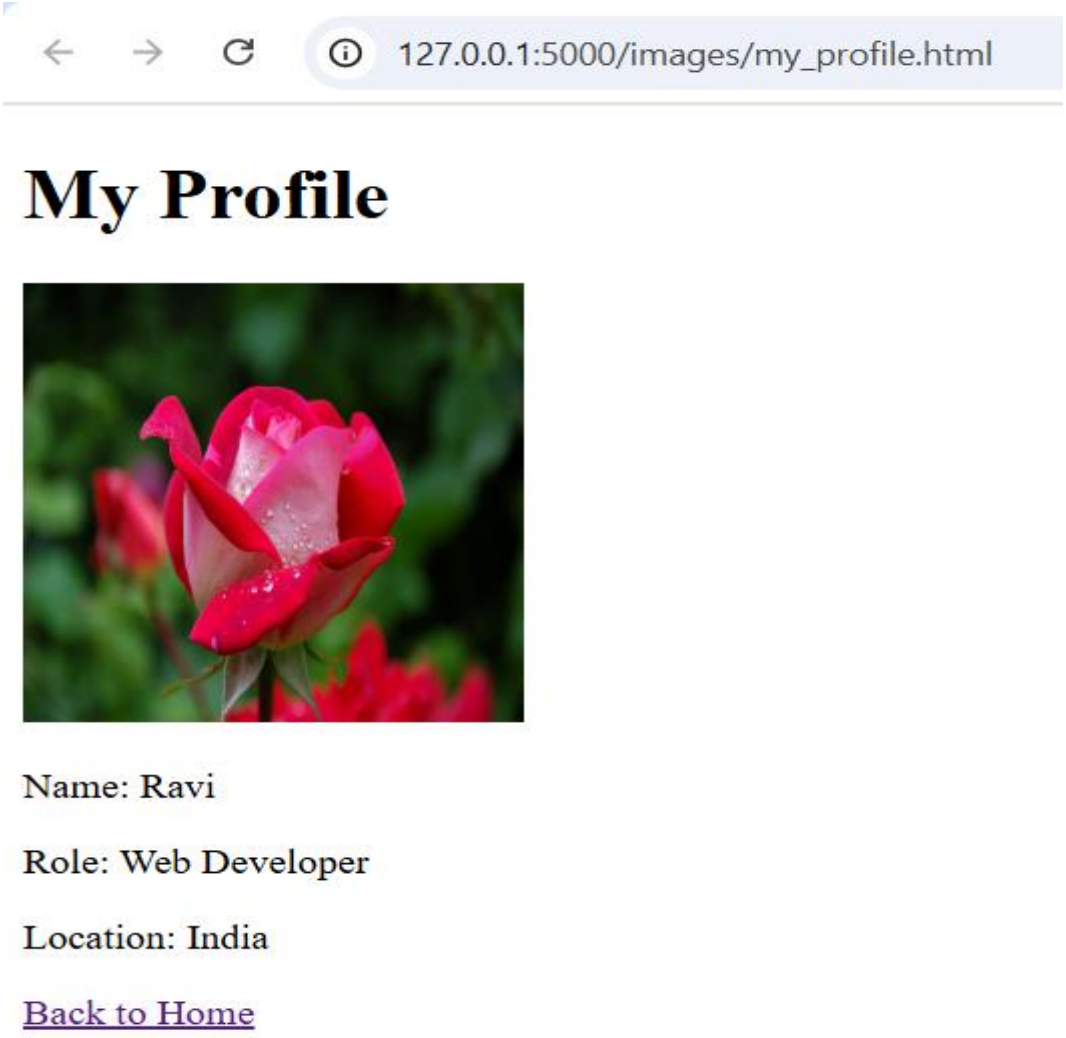


Figure 4: My Profile

Source Code: friend_profile.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Friend's Profile</title>
</head>
<body>
```

```
    <h1>Friend's Profile</h1>
        <img   src="friend_image.jpg"   alt="Friend's   Image"   width="200"
height="200">
    <p>Name: Mahesh</p>
    <p>Role: Software Engineer</p>
    <p>Location: Hyderabad</p>
    <a href="profile_images.html">Back to Home</a>
</body>
</html>
```

Output:

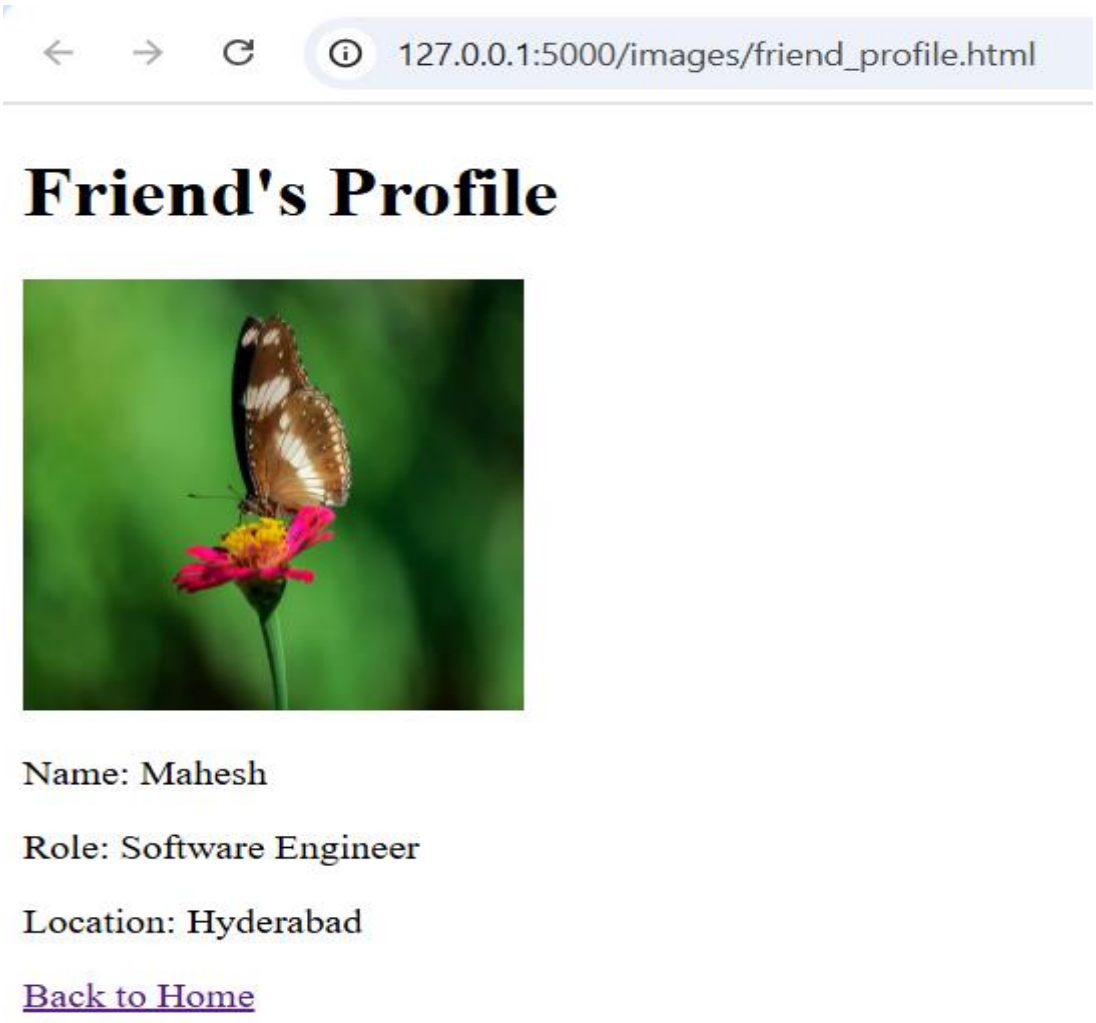


Figure 5: Friend Profile

d) Write a html program, in such a way that, rather than placing large images on a page, the preferred technique is to use thumbnails by setting the height and width parameters to something like to 100*100 pixels. Each thumbnail image is also a link to a fullsized version of the image. Create an image gallery using this technique.

Source Code: image_gallary.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Thumbnail Image Gallery</title>
</head>
<body>
    <h1>Image Gallery using Thumbnails</h1>
    <!-- Thumbnail 1 -->
    <a href="1.jpg" target="_blank">
        <img src="1.jpg" alt="Image 1" width="100" height="100">
    </a>
    <!-- Thumbnail 2 -->
    <a href="2.jpg" target="_blank">
        <img src="2.jpg" alt="Image 2" width="100" height="100">
    </a>
    <!-- Thumbnail 3 -->
    <a href="3.jpg" target="_blank">
        <img src="3.jpg" alt="Image 3" width="100" height="100">
    </a>
    <!-- Thumbnail 4 -->
    <a href="4.jpg" target="_blank">
        <img src="4.jpg" alt="Image 4" width="100" height="100">
    </a>
</body>
</html>
```

Output:

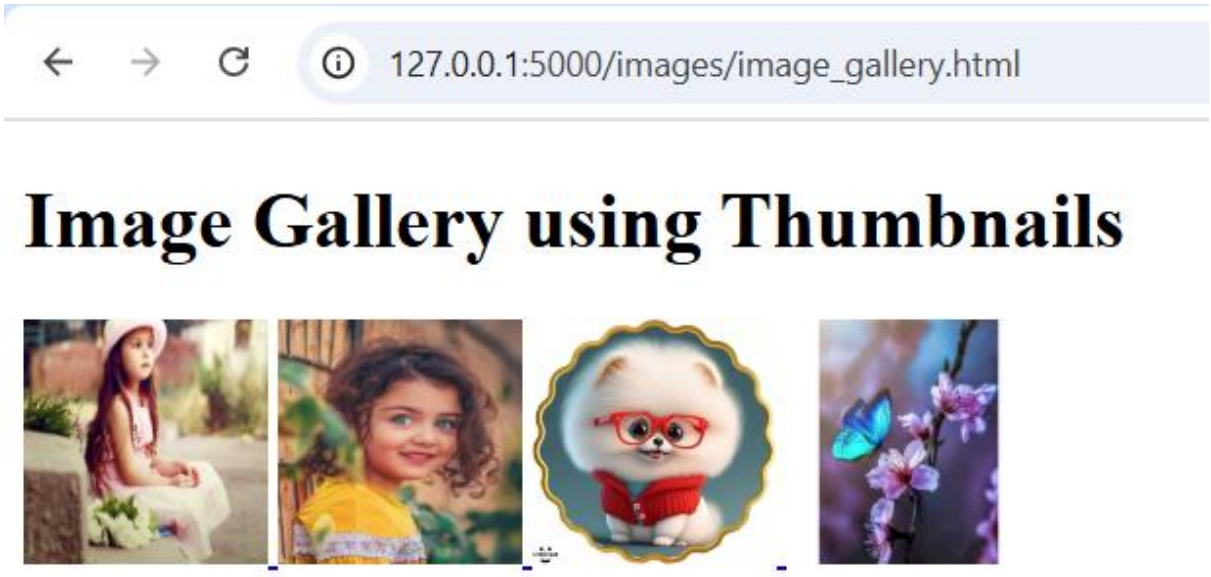


Figure 6: Image Gallery

Experiment 2:

2. HTML, Tables, Forms and Frames

a) Write a HTML program, to explain the working of tables. (use tags: <table>,<tr>,<th>,<td> and attributes: border, rowspan, colspan

Source Code: tables.html

```
<!DOCTYPE html>
<html>
<head>
    <title>HTML Table</title>
    <style>
        th {
            background-color: lightgray;
        }
    </style>
</head>
<body>
    <table border="1" cellpadding="10" cellspacing="0">
        <caption><b>Students Marks Table </b></caption>
        <tr>
            <th rowspan="2">Roll No</th>
            <th rowspan="2">Name</th>
            <th colspan="3">Marks</th>
        </tr>
        <tr>
            <th>HTML</th>
            <th>CSS</th>
            <th>Java Script</th>
        </tr>
        <tr>
            <td>101</td>
            <td>Ravi</td>
            <td>85</td>
            <td>78</td>
            <td>92</td>
        </tr>
        <tr>
            <td>102</td>
            <td>Priya</td>
            <td>90</td>
            <td>88</td>
            <td>95</td>
        </tr>
        <tr>
            <td>103</td>
            <td>Ajay</td>
            <td>75</td>
            <td>80</td>
            <td>85</td>
        </tr>
    </table>
</body>
</html>
```

Output:

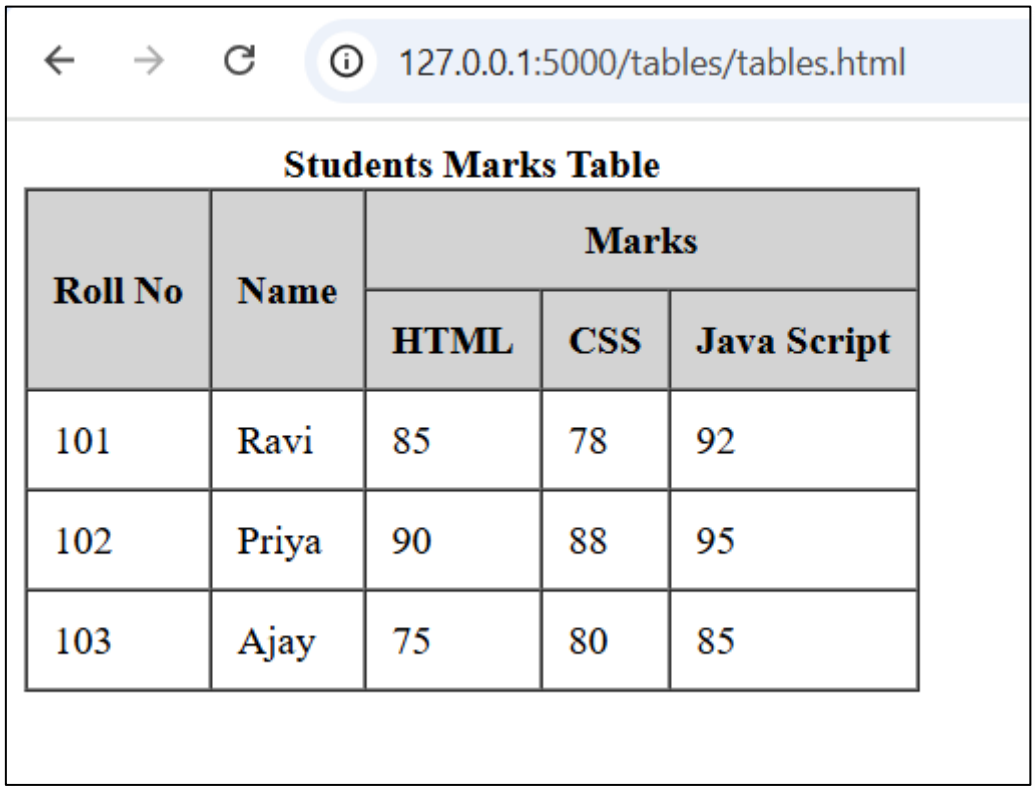


Figure 7: Students Data Table

b) Write a HTML program, to explain the working of tables by preparing a timetable, (Note: Use <caption> tag to set the caption to the table & also use cell spacing, cell padding, border, rowspan, colspan etc.)

Source Code: timetable.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Time Table</title>
    <style>
        th {
            background-color: lightgray;
        }
    </style>
</head>
<body>
    <table border="1" cellspacing="5" cellpadding="10">
        <caption><strong>Weekly Timetable</strong></caption>
        <!-- Header Row -->
        <tr>
            <th>Day</th>
            <th>8:00 - 9:00</th>
            <th>9:00 - 10:00</th>
```

```
    <th>10:00 - 11:00</th>
    <th>11:00 - 12:00<br>(Break)</th>
    <th>12:00 - 1:00</th>
    <th>1:00 - 2:00</th>
    <th>2:00 - 3:00</th>
</tr>
<!-- Monday -->
<tr>
    <td>Monday</td>
    <td>HTML</td>
    <td>C</td>
    <td>Java</td>
    <td rowspan="6" align="center">Break</td>
    <td>HTML</td>
    <td>C</td>
    <td>Java</td>
</tr>
<!-- Tuesday -->
<tr>
    <td>Tuesday</td>
    <td>C</td>
    <td>HTML</td>
    <td>Java</td>
    <td>Java</td>
    <td>HTML</td>
    <td>C</td>
</tr>
<!-- Wednesday -->
<tr>
    <td>Wednesday</td>
    <td colspan="3" align="center">SOC Lab</td>
    <td>C</td>
    <td>Java</td>
    <td>HTML</td>
</tr>
<!-- Thursday -->
<tr>
    <td>Thursday</td>
    <td colspan="3" align="center">SOC Lab</td>
    <td>HTML</td>
    <td>C</td>
    <td>Java</td>
</tr>
<!-- Friday -->
<tr>
    <td>Friday</td>
    <td>C</td>
    <td>HTML</td>
    <td>Java</td>
    <td>Java</td>
    <td>HTML</td>
    <td>C</td>
</tr>
<!-- Saturday -->
```

```
        <tr>
            <td>Saturday</td>
            <td colspan="3" align="center">Java Lab</td>
            <td>Java</td>
            <td>C</td>
            <td>HTML</td>
        </tr>
    </table>
</body>
</html>
```

Output:

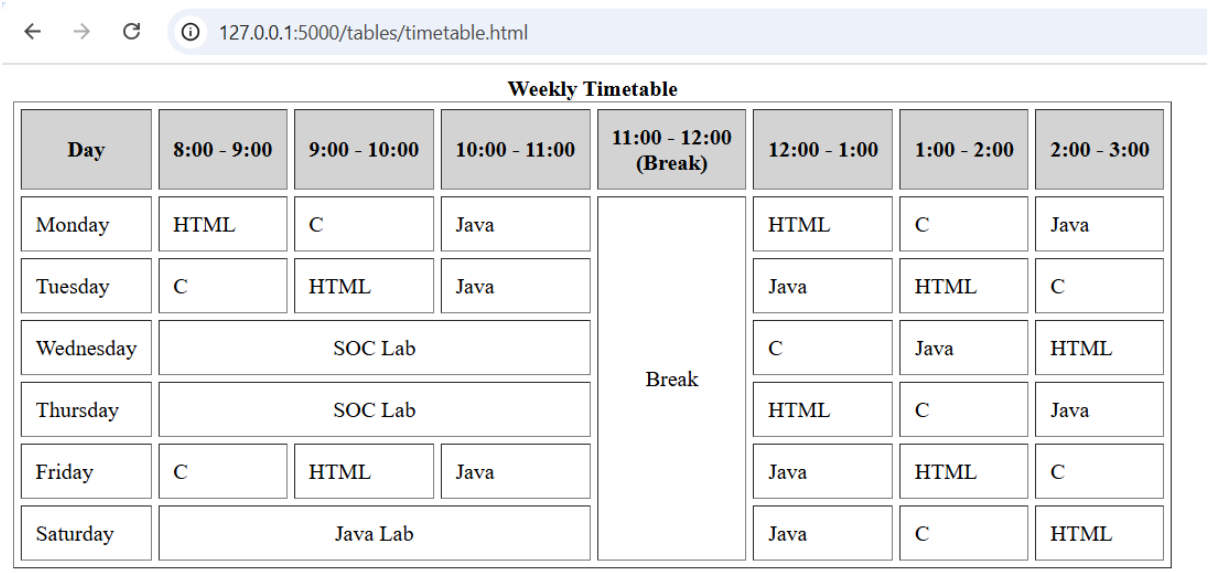


Figure 8: Time Table

c) Write a HTML Program, to explain the working of forms by designing Registration form.(Note: include text field, password field, number of field, date of birth field, checkboxes, radio buttons, list boxes using <Select> & <option> tags, <text area> and two buttons i. e: submit and reset. Use tables to provide a better view.

Source Code: registration_form.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Registration Form</title>
</head>
<body>
    <form>
        <table border="1" cellpadding="8" cellspacing="0">
            <caption><strong>Registration Form</strong></caption>
            <!-- Name -->
            <tr>
                <td><label for="name">Full Name:</label></td>
                <td><input type="text" id="name" name="name" required></td>
```

```
            </tr>
            <!-- Password -->
            <tr>
                <td><label for="password">Password:</label></td>
                  <td><input type="password" id="password" name="password"
required></td>
            </tr>
            <!-- Age -->
            <tr>
                <td><label for="age">Age:</label></td>
                <td><input type="number" id="age" name="age" min="1" max="99"
value="" required></td>
            </tr>
            <!-- Date of Birth -->
            <tr>
                <td><label for="dob">Date of Birth:</label></td>
                <td><input type="date" id="dob" name="dob" required></td>
            </tr>
            <!-- Gender (Radio Buttons) -->
            <tr>
                <td>Gender:</td>
                <td>
                   <input type="radio" id="male" name="gender" value="Male"
required>
                    <label for="male">Male</label>
                           <input  type="radio"  id="female"  name="gender"
value="Female" required>
                    <label for="female">Female</label>
                </td>
            </tr>
            <!-- Hobbies (Checkboxes) -->
            <tr>
                <td>Hobbies:</td>
                <td>
                      <input type="checkbox" name="hobby" value="Reading">
Reading
                       <input type="checkbox" name="hobby" value="Sports">
Sports
                   <input type="checkbox" name="hobby" value="Music"> Music
                </td>
            </tr>
            <!-- Course (Select Box) -->
            <tr>
                <td><label for="course">Select Course:</label></td>
                <td>
                    <select id="course" name="course" required>
                        <option value="">--Select your course--</option>
                        <option value="B.Tech">B.Tech</option>
                        <option value="M.Tech">M.Tech</option>
                        <option value="Ph.D">Ph.D</option>
                    </select>
                </td>
            </tr>
            <!-- Address (Textarea) -->
```

```
            <tr>
                <td><label for="address">Address:</label></td>
                      <td><textarea  id="address"  name="address"  rows="4"
cols="30"></textarea></td>
            </tr>
            <!-- Submit and Reset Buttons -->
            <tr>
                <td colspan="2" align="center">
                    <input type="submit" value="Submit">
                    <input type="reset" value="Reset">
                </td>
            </tr>
        </table>
    </form>
</body>
</html>
```

Output:

Figure 9: Registration Form

d) Write a HTML program, to explain the working of frames, such that page is to be divided into 3 parts on either direction. (Note: first frame->image, second frame->paragraph, third frame->hyperlink. And also make sure of using "no frame" attributes such that frames to be fixed).

Source Code: frames.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Frames</title>
    <style>
        body {
            margin: 0;
        }
        .container {
            display: flex;
            height: 100vh;
        }
    </style>
</head>
<body>
    <div class="container">
        <iframe src="image.html" width="33%"></iframe>
        <iframe src="paragraph.html" width="33%"></iframe>
        <iframe src="hyperlink.html" width="33%"></iframe>
        <div>
            <noframes>
                <p>Your browser does not support frames/iframes.</p>
                <p>Please update your browser</p>
            </noframes>
</body>
```

```
</html>
```

Source Code: image.html

```
<!DOCTYPE html>
<html>
<head>
    <title>image Frame</title>
    <style>
        html,
        body {
            margin: 0;
            padding: 0;
            height: 100%;
            width: 100%;
        }
        img {
            height: 100%;
            width: 100%;
            object-fit: cover;
            display: block;
        }
    </style>
</head>
<body>
    <img src="image1.jpg" alt="Image is not found">
</body>
</html>
```

Source Code: paragraph.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Paragraph Frame</title>
</head>
<body style="font-size:18px; padding:10px;">
    <p>This is a paragraph loaded in the second frame. It explains how we use separate HTML files in iframes to create a multi-section layout using HTML only.</p>
</body>
</html>
```

Source Code: hyperlink.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Hyperlink Frame</title>
</head>
<body style="text-align:center; padding:20px;">
     <a href="https://www.google.com" target="_blank">Click here to go to Google</a>
</body>
</html>
```

Output:

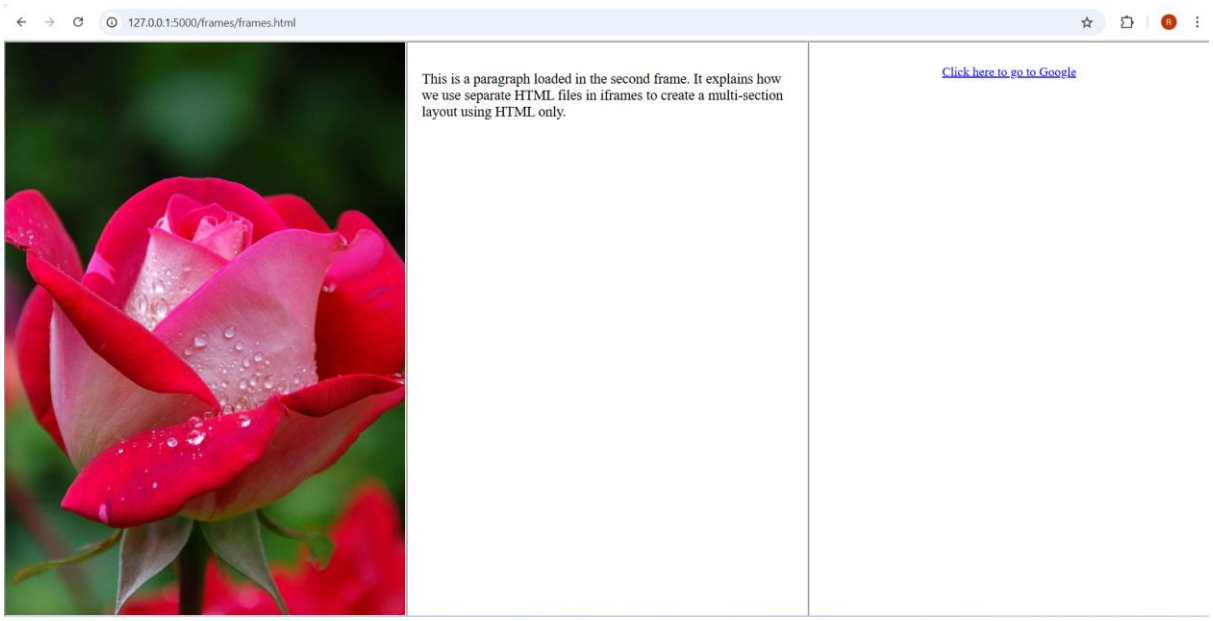


Figure 10: Frames- Image, Paragraph, Hyperlink

Experiment 3:

3. HTML 5 and Cascading, Style Sheets, Types of CSS

a) Write a HTML program that makes use of <article>, <aside>, <figure>, <figcaption>, <footer>, <header>, <main>, <nav>, <section>, , tags

Source Code: html5_tags.html

```
<!DOCTYPE html>
<html>
<head>
  <title>HTML5 Tags</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      margin: 0;
      padding: 0;
    }
    header, nav, main, section, article, aside, footer, div {
      padding: 10px;
      margin: 10px;
      border: 1px solid #ccc;
    }
    nav a {
      margin: 0 10px;
      text-decoration: none;
      color: blue;
    }
    .highlight {
      color: red;
      font-weight: bold;
    }
  </style>
```

```
</head>
<body>
  <!-- Header -->
  <header>
    <h1 align="center">My Blog Website</h1>
    <p align="center">Welcome to my blog</p>
  </header>
  <!-- Navigation -->
  <nav align="center">
    <a href="home.html" target="_blank">HOME</a>
    <a href="about.html" target="_blank">ABOUT US</a>
    <a href="article.html" target="_blank">ARTICLES</a>
    <a href="contact.html" target="_blank">CONTACT US</a>
  </nav>
  <!-- Main Content -->
  <main>
    <section>
      <h2>Technology</h2>
      <!-- Article -->
      <article>
        <h3>What is HTML5?</h3>
        <p>HTML5 is the latest version of HyperText Markup Language used to
structure web pages.</p>
        <!-- Figure and Figcaption -->
        <figure>
          <img  src="html5_logo.png"  width="100"  height="100"  alt="HTML5
Logo">
          <figcaption>Figure: HTML5 Official Logo</figcaption>
        </figure>
        <!-- Using span inside paragraph -->
        <p>HTML5 includes semantic tags like
          <span class="highlight">&lt;article&gt;</span>
          and
          <span class="highlight">&lt;section&gt;</span>.
        </p>
      </article>
    </section>
    <!-- Aside -->
    <aside>
      <h4>Related Links</h4>
      <ul>
        <li><a href="https://www.w3schools.com/css/">CSS3 Tutorial</a></li>
        <li><a              href="https://www.w3schools.com/js/">JavaScript
Basics</a></li>
      </ul>
    </aside>
  </main>
  <!-- Div -->
  <div>
    <p>This content is inside a <span class="highlight">&lt;div&gt;</span>
tag.</p>
  </div>
  <!-- Footer -->
  <footer>
```

```
    <p>&copy; 2025 My Blog | Designed by KITS</p>
  </footer>
</body>
</html>
```

Output:

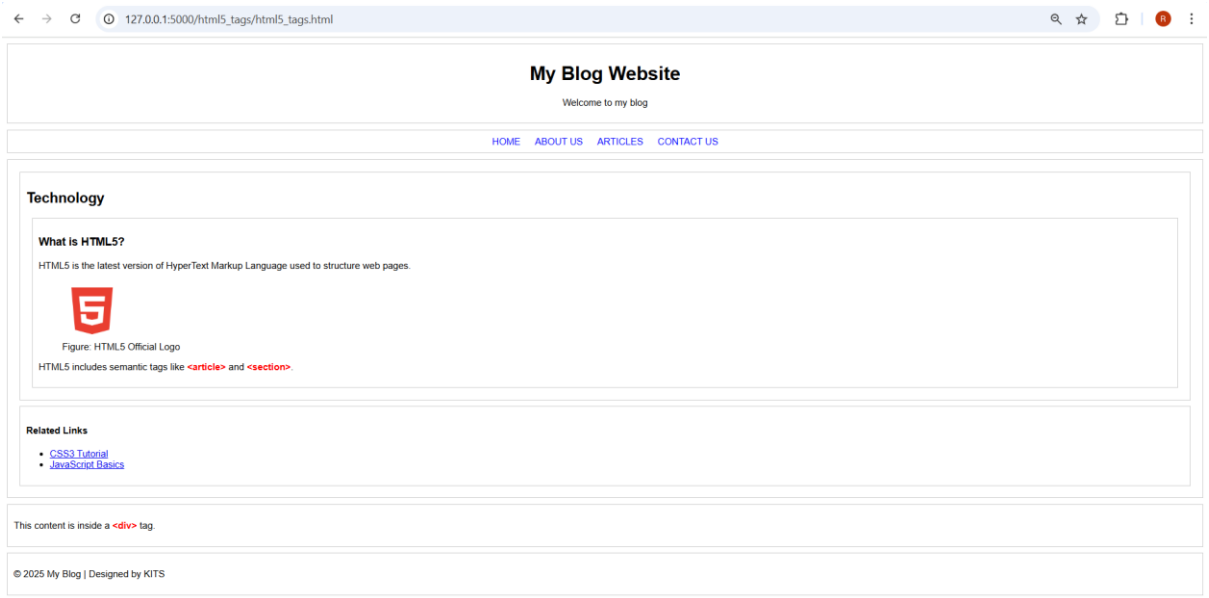


Figure 11: HTML5 Tags

Source Code: home.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Home - My Blog</title>
</head>
<body>
  <h1>Welcome to My Blog</h1>
  <p>This is the home page of my blog website.</p>
</body>
</html>
```

Output:

Figure 12: Home Page

Source Code: about.html

```
<!DOCTYPE html>
<html>
<head>
  <title>About - My Blog</title>
</head>
<body>
  <h1>About Us</h1>
  <p>This blog is created to share web development knowledge.</p>
</body>
</html>
```

Output:

Figure 13: About Us Page

Source Code: article.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Article - My Blog</title>
</head>
<body>
  <h1>What is HTML5?</h1>
  <p>HTML5 is the latest version of HTML used for structuring webpages.</p>
</body>
</html>
```

Output:

What is HTML5?

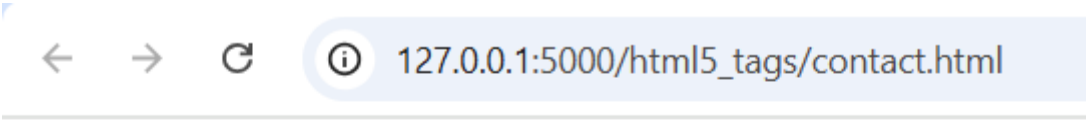
HTML5 is the latest version of HTML used for structuring webpages.

Figure 14: Article Page

Source Code: contact.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Contact - My Blog</title>
</head>
<body>
  <h1>Contact Us</h1>
  <p>Email: contact@myblog.com</p>
</body>
</html>
```

Output:



Contact Us

Email: contact@myblog.com

Figure 15: Contact Us Page

b) Write a HTML program, to embed audio and video into HTML web page

Source Code: media.html

```
<!DOCTYPE html>
html>

<head>
    <title>Embed Audio and Video</title>
</head>

<body>
```

```
    <div>
        <h2>Audio</h2>
        <audio controls>
            <source src="audios/sample.mp3" type="audio/mp3" />
            Your browser does not support the audio element.
        </audio>
        <h2>Video</h2>
        <video width="480" height="270" controls>
            <source src="videos/sample.mp4" type="video/mp4" />
            Your browser does not support the video tag.
        </video>
    </div>
</body>

</html>
```

Output:

c) Write a program to apply different types (or levels of styles or style specification formats)- inline, internal, external styles to HTML elements(identify selector, property and value)

Source Code: types_of_css.html

```
<!DOCTYPE html>
<html>
<head>
    <title>CSS Styles</title>
    <!-- External Style -->
    <link rel="stylesheet" type="text/css" href="styles.css">
    <!-- Internal Style -->
    <style>
        /* Selector: h2, Property: color, Value: blue */
        h2 {
            color: blue;
            font-family: Arial, sans-serif;
        }
        /* Selector: .internal-class, Property: background-color, Value:
lightyellow */
        .internal-class {
            background-color: lightyellow;
            padding: 10px;
            border: 2px solid orange;
            width:600px;
        }
    </style>
</head>
<body>
    <!-- Inline Style -->
    <h1 style="color: red;">
        Inline Style
```

```
    </h1>
    <!-- Internal Style -->
    <p class="internal-class">
        This paragraph uses internal CSS with a class selector.
    </p>
    <!-- External Style -->
    <p class="external-class">
        This paragraph uses external CSS from styles.css.
    </p>
</body>
</html>
```

Source Code: styles.css

```
/* Selector: .external-class, Property: background-color, Value: lightgreen
*/
.external-class {
    background-color: lightgreen;
    font-size: 18px;
    padding: 8px;
    border: 2px solid green;
    width:600px;
}
```

Output:

Experiment 4:

4. Selector forms

a) Write a program to apply different types of selector forms

1. Simple selector(element, id, class, group, universal)

2. Combinator selector(descendant, child, adjacent sibling, general sibling)

3. Pseudo-class selector

4. Pseudo-element selector

5. Attribute selector

Source Code: selectors.html

```
<!DOCTYPE html>
<html>
<head>
  <title>CSS Selector Forms</title>
  <style>
    /* 1. Simple Selectors */
    h1 { color: blue; }               /* element selector */
    #special { color: red; }          /* id selector */
    .highlight { background: yellow;} /* class selector */
    p, li { font-style: italic;}     /* group selector */
    * { font-family: Arial; }         /* universal selector */
```

```
    /* 2. Combinator Selectors */
    div p { color: red; }           /* descendant */
    div > span { color: purple; }     /* child */
    h2 + p { background: lightgray; } /* adjacent sibling */
    h3 ~ ul { border: 1px solid black;} /* general sibling */
    /* 3. Pseudo-class Selector */
    a:hover { color: orange; }
    /* 4. Pseudo-element Selector */
   p::first-letter {
      font-size: 24px;
      color: brown;
      }
    /* 5. Attribute Selector */
    button[id="button"] {
      background-color: blue;
      color: white;
    }
  </style>
</head>
<body>
  <h1>Simple Selector</h1>
  <h2 id="special">ID Selector</h2>
  <p class="highlight">This is a paragraph with a class selector</p>
  <div>
    <p>Descendant selector inside a div</p>
    <span>Child selector span inside div</span>

  </div>
  <!--Select the first <p> element that comes immediately after an <h2>. -
->
  <h2>Heading</h2>
  <p>This paragraph will get lightgray background</p>
  <p>This paragraph will NOT get it </p>
  <!-- Select all <ul> elements that come after an <h2> (not just the first
one). -->
  <h3>Heading</h3>
  <ul><li>First list</li></ul>
  <ul><li>Second list</li></ul>
  <a href="#">Hover over this link</a>
  <!--  all paragraph First Letter is brown -->
  <p>Hello world</p>
  <button type="button" id="button" >Click Me</button>
</body>
</html>
```

Output:

Experiment 5:

5. CSS with Color, Background, Font, Text and CSS Box Model

a) Write program to demonstrate the various ways you can reference a color in CSS.

b) Write a CSS rule that places a background image halfway down the page, tiling it horizontally. The image should remain in place when the user scrolls up or down

Source Code: css_color_bg.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>CSS Color, Background</title>
  <style>
    /* a. Different ways to apply color */
    h1 { color: red; }                /* named color */
    h2 { color: #00ff00; }            /* hex code */
    h3 { color: rgb(0, 0, 255); }     /* RGB */
    p { color: rgba(255, 165, 0, 0.5);} /* RGBA with opacity */
    /* b. Background image */
    body {
      background-image: url("https://www.w3schools.com/css/img_tree.png");
      background-repeat: repeat-x;       /* horizontal tiling */
      background-attachment: fixed;      /* stays fixed when scrolling */
      background-position: center 50%;   /* halfway down the page */
    }
  </style>
</head>
<body>
  <!-- Colors -->
  <h1>Color : Named Color (Red)</h1>
  <h2>Color : Hex (#00ff00)</h2>
  <h3>Color : RGB (0,0,255)</h3>
  <p>Color  : RGBA with opacity</p>
</body>
</html>
```

Output:

5. CSS with Font, Text and CSS Box Model

c) Write a program using the following terms related to CSS font and text:

i. font-size ii. font-weight iii. font-style iv. text-decoration, v. text-transformation vi. text-alignment

d) Write a program, to explain the importance of CSS Box model using

i. Content ii. Border iii. Margin iv. padding

Source Code: css_font_text_box.html

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <title>CSS Font, Text and CSS Box Model</title>
  <style>
```

```
    /* a. Font & Text properties */
    .text-demo {
    font-size: 20px;
    font-weight: bold;
    font-style: italic;
    text-decoration: underline;
    text-transform: uppercase;
    text-align: justify;
    color: blue;
    background: yellow;
    display: inline-block;   /* shrink width to content */
    padding: 5px 10px;      /* optional: space inside background */
    }
   /* b. CSS Box Model */
    .box {
      margin: 20px;          /* space outside */
      border: 3px solid black; /* border */
      padding: 15px;         /* space inside */
      width: 200px;          /* content area */
      background: #f0e68c;
    }
  </style>
</head>
<body>
  <!-- Font & Text -->
  <div class="text-demo">
    This is CSS Font and Text Demonstration
  </div>
  <!-- Box Model -->
  <div class="box">
    This is a Box Model (content + padding + border + margin).
  </div>
</body>
</html>
```

Output:

Experiment 6:

6. Applying JavaScript - internal and external, I/O, Type Conversion

a) Write a program to embed internal and external JavaScript in a web page

Source Code: js_internal_external.html

```
<!DOCTYPE html>
<html>
<head>
    <title>internal & external JS</title>
</head>
<body>
    <h1>internal and external JS</h1>
    <button onclick="internaljs()">Internal JS </button>
    <button onclick="externaljs()">External JS </button>
```

```
    <!-- Internal JavaScript -->
    <script>
        function internaljs() {
            alert("internal JS!");
        }
    </script>

    <!-- External JavaScript -->
    <script src="external.js"></script>

</body>
</html>
```

external.js

```
function externalJS() {
    alert("External JS!");
}
```

Output:

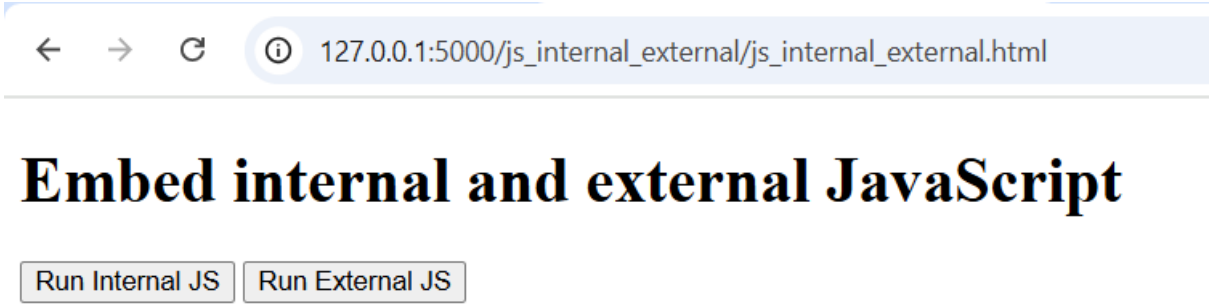


Figure 16: HTML Page – Embed Internal and External JavaScript

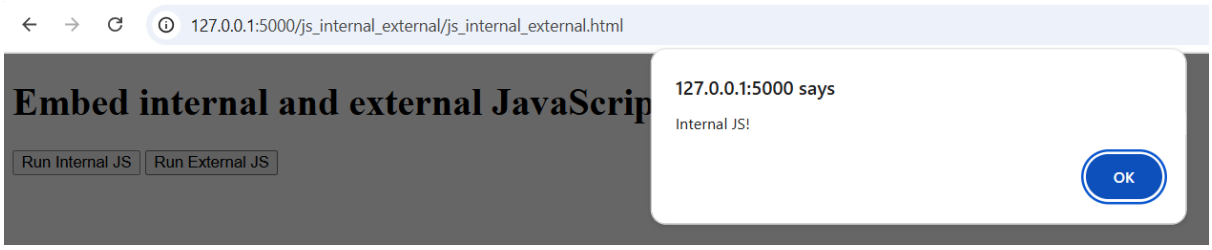


Figure 17: Output of Internal JavaScript

Figure 18:Output of External JavaScript

b) Write a program to explain the different ways for displaying output-

Source Code: js_output_methods.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Display output in JavaScript</title>
    <script>
        // 1. Using alert()
        function showAlert() {
            alert("This is displayed using alert()");
        }
        // 2. Using console.log()
        function showConsole() {
            console.log("This   is   displayed   in   the   console   using
console.log()");
        }
        // 3. Using document.write()
        function showDocumentWrite() {
            document.write("This   is   displayed   using   document.write()
<br>");
        }
        // 4. Using innerHTML
        function showInnerHTML() {
            document.getElementById("outputDiv").innerHTML   =   "This   is
displayed using innerHTML";
        }
    </script>
</head>
<body>
    <h1>Different Ways to Display Output</h1>
    <button onclick="showAlert()">Alert</button>
    <button onclick="showConsole()">Console.log</button>
    <button onclick="showDocumentWrite()">Document.write</button>
    <button onclick="showInnerHTML()">InnerHTML</button>
    <div id="outputDiv" style="margin-top:20px; color:blue;"></div>
</body>
</html>
```

Output:

c) Write a program to explain the different ways for taking input

Source Code: js_input_methods.html

```
<!DOCTYPE html>
<html>
<head>
    <title>Input Methods in JavaScript</title>
    <script>
        // 1. Using prompt()
        function takePrompt() {
            let name = prompt("Enter your name:");
            alert("Your name is  " + name );
```

```
        }

        // 2. Using input field and button
        function takeInputField() {
            let age = document.getElementById("age").value;
            alert("Your age is " + age );
        }
    </script>
</head>
<body>
    <h1>Different Ways to Take Input</h1>

    <!-- Button for prompt() -->
    <button onclick="takePrompt()">Input using prompt()</button>

    <br><br>

    <!-- Input field and button -->
    <input type="number" id="age" placeholder="Enter your age">
    <button onclick="takeInputField()">Input using input field</button>
</body>
</html>
```

Output: