1. Lists, Links and Images

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

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

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Understanding Lists in HTML</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            color: #333;
            line-height: 1.6;
            margin: 20px;
        }
        h1, h2, h3 {
            color: #333;
        }
        ul, ol, dl {
            margin-left: 20px;
        }
        dt {
            font-weight: bold;
        }
        dd {
            margin-left: 20px;
        }
    </style>
</head>
<body>
    <h1>Understanding Lists in HTML</h1>
    <p>In HTML, lists are used to group related items. There are three main types of lists: unordered lists, ordered lists, and definition lists. Lists can also be nested within each other.</p>
    <h2>Unordered List (Bulleted List)</h2>
```

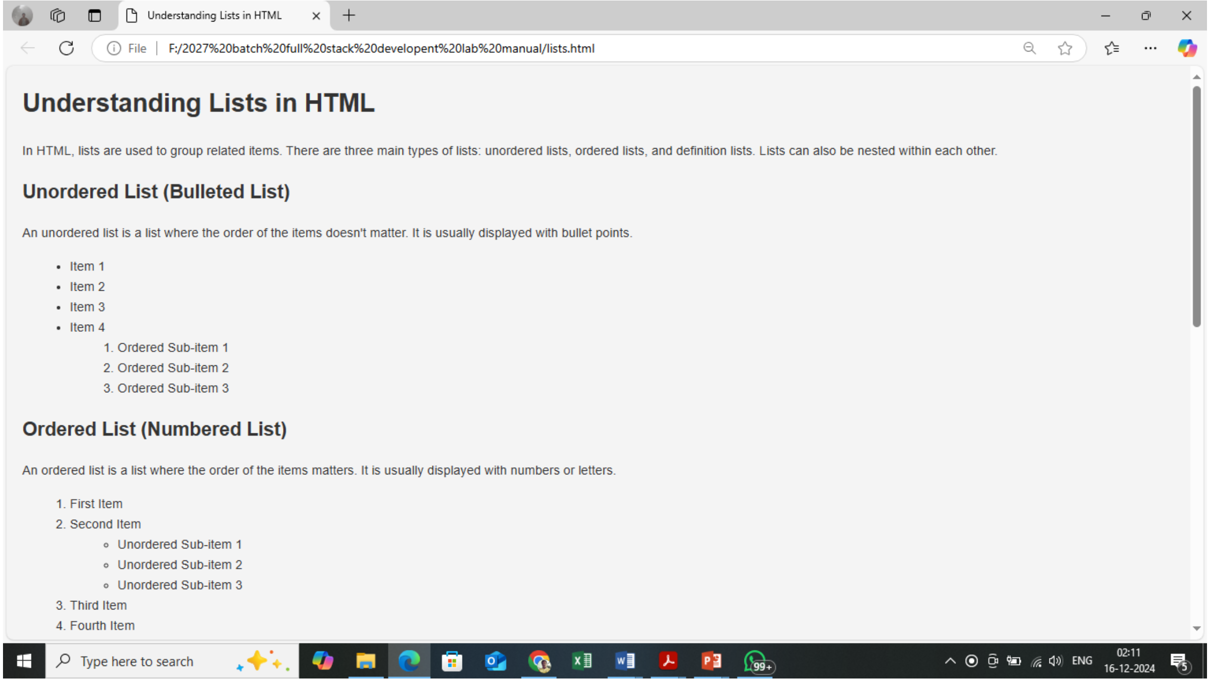
```
<p>An unordered list is a list where the order of the items doesn't matter. It is usually displayed with bullet points.</p>
<ul>
    <li>Item 1</li>
    <li>Item 2</li>
    <li>Item 3</li>
    <li>Item 4
        <ol>
            <li>Ordered Sub-item 1</li>
            <li>Ordered Sub-item 2</li>
            <li>Ordered Sub-item 3</li>
        </ol>
    </li>
</ul>
<h2>Ordered List (Numbered List)</h2>
<p>An ordered list is a list where the order of the items matters. It is usually displayed with numbers or letters.</p>
<ol>
    <li>First Item</li>
    <li>Second Item
        <ul>
            <li>Unordered Sub-item 1</li>
            <li>Unordered Sub-item 2</li>
            <li>Unordered Sub-item 3</li>
        </ul>
    </li>
    <li>Third Item</li>
    <li>Fourth Item</li>
</ol>

<h2>Definition List</h2>
<p>A definition list is a list of terms and their corresponding definitions.</p>
<dl>
    <dt>HTML</dt>
    <dd>HyperText Markup Language, the standard language for creating web pages.</dd>
    <dt>CSS</dt>
    <dd>Cascading Style Sheets, used to style HTML elements.</dd>
```

```html
        <dt>JavaScript</dt>
        <dd>A programming language used to create dynamic and interactive effects within web browsers.</dd>
        <dt>Nested Definition List</dt>
        <dd>
            <dl>
                <dt>Term 1</dt>
                <dd>Definition for term 1</dd>
                <dt>Term 2</dt>
                <dd>Definition for term 2</dd>
            </dl>
        </dd>
    </dl>
    <h2>Combining Lists</h2>
    <p>Lists can be combined and nested to create complex structures. Here is an example:</p>
    <ul>
        <li>Item with nested lists
            <ul>
                <li>Sub-item in an unordered list
                    <ol>
                        <li>Ordered sub-item in an unordered list</li>
                        <li>Another ordered sub-item</li>
                    </ol>
                </li>
            </ul>
        </li>
        <li>Another item with nested definition list
            <dl>
                <dt>Term in a nested list</dt>
                <dd>Definition for the term in a nested list</dd>
            </dl>
        </li>
    </ul>
    <h2>Conclusion</h2>
    <p>HTML provides various ways to present data in a list format. Whether you use unordered lists for bullet points, ordered lists for numbered items, or definition lists for terms and definitions, lists help organize information clearly for users. Additionally, lists can be nested within each other to create more complex and organized structures.</p>
```

```
</body>

</html>
```



Understanding Lists in HTML
File
F:/2027%20batch%20full%20stack%20developent%20lab%20manual/lists.html
Type here to search
99+
ENG
02:11
16-12-2024
5

Understanding Lists in HTML

In HTML, lists are used to group related items. There are three main types of lists: unordered lists, ordered lists, and definition lists. Lists can also be nested within each other.

Unordered List (Bulleted List)

An unordered list is a list where the order of the items doesn't matter. It is usually displayed with bullet points.

- Item 1
- Item 2
- Item 3
- Item 4
    1. Ordered Sub-item 1
    2. Ordered Sub-item 2
    3. Ordered Sub-item 3

Ordered List (Numbered List)

An ordered list is a list where the order of the items matters. It is usually displayed with numbers or letters.

1. First Item
2. Second Item
    - Unordered Sub-item 1
    - Unordered Sub-item 2
    - Unordered Sub-item 3
3. Third Item
4. Fourth Item

1.b Write a HTML program, to explain the working of hyperlinks using <a> tag and href, target Attributes.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Understanding Hyperlinks in HTML</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            color: #333;
            line-height: 1.6;
            margin: 20px;
        }
        h1, h2 {
            color: #333;
        }
        p {
            margin-bottom: 20px;
        }
        a {
            color: #1a0dab;
            text-decoration: none;
        }
        a:hover {
            text-decoration: underline;
        }
    </style>
</head>
<body>
    <h1>Understanding Hyperlinks in HTML</h1>

    <p>In HTML, hyperlinks are created using the <code>&lt;a&gt;</code> (anchor) tag. The <code>href</code> attribute specifies the URL of the page the link goes to, and the <code>target</code> attribute specifies where to open the linked document.</p>
    <h2>Basic Hyperlink</h2>
```

```html
    <p>A basic hyperlink is created by using the <code>&lt;a&gt;</code> tag with the <code>href</code> attribute.</p>

    <p><a href="https://www.example.com">Visit Example.com</a></p>

    <p>In the above example, clicking the link will take you to <code>https://www.example.com</code>.</p>

<h2>Open Link in a New Tab</h2>

    <p>To open the link in a new tab, use the <code>target="_blank"</code> attribute.</p>

    <p><a href="https://www.example.com" target="_blank">Visit Example.com in a New Tab</a></p>

    <p>In the above example, clicking the link will open <code>https://www.example.com</code> in a new tab.</p>

  <h2>Open Link in the Same Frame</h2>

    <p>To open the link in the same frame (default behavior), use the <code>target="_self"</code> attribute.</p>

    <p><a href="https://www.example.com" target="_self">Visit Example.com in the Same Frame</a></p>

    <p>In the above example, clicking the link will open <code>https://www.example.com</code> in the same tab or frame.</p>

     <h2>Open Link in a Parent Frame</h2>

    <p>To open the link in the parent frame, use the <code>target="_parent"</code> attribute. This is useful when dealing with frames.</p>

    <p><a href="https://www.example.com" target="_parent">Visit Example.com in the Parent Frame</a></p>

    <h2>Open Link in the Full Body of the Window</h2>

    <p>To open the link in the full body of the window, use the <code>target="_top"</code> attribute. This is also useful when dealing with frames.</p>

    <p><a href="https://www.example.com" target="_top">Visit Example.com in the Full Body of the Window</a></p>

    <h2>Relative Links</h2>

    <p>Links can also be relative, pointing to other pages within the same website.</p>

    <p><a href="contact.html">Contact Us</a></p>

    <p>In the above example, clicking the link will take you to the <code>contact.html</code> page within the same website.</p>

    <h2>Email Links</h2>

    <p>You can also create a link to send an email using the <code>mailto:</code> scheme.</p>

    <p><a href="mailto:example@example.com">Send an Email</a></p>

    <p>In the above example, clicking the link will open the default email client to send an email to <code>example@example.com</code>.</p>

    <h2>Conclusion</h2>

    <p>Hyperlinks are a fundamental part of HTML, allowing users to navigate between different pages and resources. By using the <code>href</code> attribute, you can specify the destination of the link, and by using the <code>target</code> attribute, you can control where the link opens.</p>
  </body>
</html>
```

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

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Profile Links</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      color: #333;
      text-align: center;
      margin: 20px;
    }
    h1 {
      color: #333;
    }
    .profile {
      display: inline-block;
      margin: 20px;
    }
    .profile img {
      width: 150px;
      height: 150px;
      border-radius: 50%;
      cursor: pointer;
      transition: transform 0.3s;
    }
    .profile img:hover {
      transform: scale(1.1);
    }   </style>
</head>
<body>
  <h1>Profile Links</h1>
```

```
    <div class="profile">
        <a href="https://www.example.com/my-profile" target="_blank">
            <img src="my-image.jpg" alt="My Profile Image">
        </a>
        <p>My Profile</p>
    </div>
    <div class="profile">
        <a href="https://www.example.com/friend-profile" target="_blank">
            <img src="friend-image.jpg" alt="Friend's Profile Image">
        </a>
        <p>Friend's Profile</p>
    </div>
</body>
</html>
```

1. D) Write a HTML program, 1n 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 thumbnails image is also a link a full sized version of the image. Create a image gallery using this technique.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial
scale=1.0">
    <title>Image Gallery</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            color: #333;
            text-align: center;
            margin: 20px;
        }
        h1 {
            color: #333;
        }
        .gallery {
            display: flex;
            flex-wrap: wrap;
            justify-content: center;
        }
        .gallery-item {
            margin: 10px;
        }
        .gallery-item img {
            width: 100px;
            height: 100px;
            border-radius: 5px;
            cursor: pointer;
            transition: transform 0.3s;
        }
        .gallery-item img:hover {
            transform: scale(1.1);
        }
    </style>
</head>
<body>
    <h1>Image Gallery</h1>
    <div class="gallery">
```

```
        <div class="gallery-item">
            <a href="images/image1.jpg" target="_blank">
                <img src="images/thumbnails/image1.jpg" alt="Image 1">
            </a>
        </div>
        <div class="gallery-item">
            <a href="images/image2.jpg" target="_blank">
                <img src="images/thumbnails/image2.jpg" alt="Image 2">
            </a>
        </div>
        <div class="gallery-item">
            <a href="images/image3.jpg" target="_blank">
                <img src="images/thumbnails/image3.jpg" alt="Image 3">
            </a>
        </div>
        <div class="gallery-item">
            <a href="images/image4.jpg" target="_blank">
                <img src="images/thumbnails/image4.jpg" alt="Image 4">
            </a>
        </div>
        <!-- Add more images as needed -->
    </div>
</body>
</html>
```

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)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-
scale=1.0">
    <title>Understanding HTML Tables</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            color: #333;
            margin: 20px;
        }
        h1 {
            color: #333;
        }
        table {
            width: 100%;
            border-collapse: collapse;
            margin-bottom: 20px;
        }
        table, th, td {
            border: 1px solid #333;
        }
        th, td {
            padding: 10px;
            text-align: left;
        }
        th {
            background-color: #f2f2f2;
        }
    </style>
</head>
<body>
    <h1>Understanding HTML Tables</h1>
```

```html
    <p>In HTML, tables are created using the <code>&lt;table&gt;</code> tag. The table is structured with rows (<code>&lt;tr&gt;</code>) and cells, which can be either header cells (<code>&lt;th&gt;</code>) or data cells (<code>&lt;td&gt;</code>). Attributes like <code>border</code>, <code>rowspan</code>, and <code>colspan</code> can be used to enhance the table's appearance and structure.</p>

    <h2>Basic Table</h2>
    <table border="1">
        <tr>
            <th>Header 1</th>
            <th>Header 2</th>
            <th>Header 3</th>
        </tr>
        <tr>
            <td>Row 1, Cell 1</td>
            <td>Row 1, Cell 2</td>
            <td>Row 1, Cell 3</td>
        </tr>
        <tr>
            <td>Row 2, Cell 1</td>
            <td>Row 2, Cell 2</td>
            <td>Row 2, Cell 3</td>
        </tr>
    </table>

    <h2>Table with Rowspan</h2>
    <p>The <code>rowspan</code> attribute allows a cell to span multiple rows.</p>
    <table border="1">
        <tr>
            <th>Header 1</th>
            <th>Header 2</th>
            <th>Header 3</th>
        </tr>
        <tr>
            <td rowspan="2">Rowspan 2 Rows</td>
            <td>Row 1, Cell 2</td>
            <td>Row 1, Cell 3</td>
        </tr>
        <tr>
            <td>Row 2, Cell 2</td>
            <td>Row 2, Cell 3</td>
        </tr>
    </table>

    <h2>Table with Colspan</h2>
    <p>The <code>colspan</code> attribute allows a cell to
span multiple columns.</p>
    <table border="1">
        <tr>
            <th>Header 1</th>
            <th>Header 2</th>
            <th>Header 3</th>
        </tr>
        <tr>
            <td>Row 1, Cell 1</td>
            <td colspan="2">Colspan 2 Columns</td>
        </tr>
        <tr>
            <td>Row 2, Cell 1</td>
            <td>Row 2, Cell 2</td>
            <td>Row 2, Cell 3</td>
        </tr>
    </table>

    <h2>Complex Table Example</h2>
    <p>This table combines both <code>rowspan</code> and
<code>colspan</code> attributes.</p>
    <table border="1">
        <tr>
            <th>Header 1</th>
            <th>Header 2</th>
            <th>Header 3</th>
            <th>Header 4</th>
        </tr>
        <tr>
            <td rowspan="2">Rowspan 2 Rows</td>
            <td>Row 1, Cell 2</td>
            <td colspan="2">Colspan 2 Columns</td>
    </tr>
    <tr>
      <td>Row 2, Cell 2</td>
      <td>Row 2, Cell 3</td>
      <td>Row 2, Cell 4</td>
    </tr>
    <tr>
      <td>Row 3, Cell 1</td>
      <td colspan="3">Colspan 3 Columns</td>
    </tr>
  </table>

</body>
</html>
```

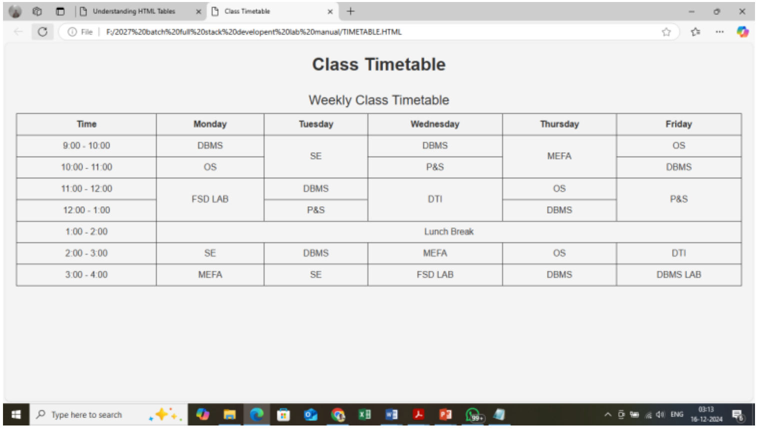
b. Write a HTML program, to explain the working of tables.by preparing a timetable. (Note: use <caption> tag to set the caption to the table & also use cell spacing, cell padding, border, rowspan, colspan etc.)

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Class Timetable</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: #f4f4f4;
      color: #333;
      margin: 20px;
      text-align: center;
    }
    h1 {
            color: #333;
        }
        table {
            width: 100%;
            border-collapse: collapse;
            margin-bottom: 20px;
        }
        table, th, td {
            border: 1px solid #333;
        }
        th, td {
            padding: 10px;
            text-align: center;
        }
        th {
            background-color: #f2f2f2;
        }
        caption {
            caption-side: top;
            font-size: 1.5em;
            margin: 10px;
        }
    </style>
</head>
<body>
    <h1>Class Timetable</h1>

    <table cellspacing="0" cellpadding="10">
        <caption>Weekly Class Timetable</caption>
        <tr>
    <th>Time</th>
    <th>Monday</th>
    <th>Tuesday</th>
    <th>Wednesday</th>
    <th>Thursday</th>
    <th>Friday</th>
</tr>
<tr>
    <td>9:00 - 10:00</td>
    <td>DBMS</td>
    <td rowspan="2">SE</td>
    <td>DBMS</td>
    <td rowspan="2">MEFA</td>
    <td>OS</td>
</tr>
<tr>
    <td>10:00 - 11:00</td>
    <td>OS</td>
    <td>P&S</td>
    <td>DBMS</td>
</tr>
<tr>
    <td>11:00 - 12:00</td>
    <td rowspan="2">FSD LAB</td>
    <td>DBMS</td>
    <td rowspan="2">DTI</td>
    <td>OS</td>
    <td rowspan="2">P&S</td>
</tr>
<tr>
    <td>12:00 - 1:00</td>
```

```
            <td>P&S</td>
            <td>DBMS</td>
        </tr>
        <tr>
            <td>1:00 - 2:00</td>
            <td colspan="5">Lunch Break</td>
        </tr>
        <tr>
            <td>2:00 - 3:00</td>
            <td>SE</td>
            <td>DBMS</td>
            <td>MEFA</td>
            <td>OS</td>
            <td>DTI</td>
        </tr>
        <tr>
            <td>3:00 - 4:00</td>
            <td>MEFA</td>
            <td>SE</td>
            <td>FSD LAB</td>
            <td>DBMS</td>
            <td>DBMS LAB</td>
        </tr>
    </table>

</body>
</html>
```


Understanding HTML Tables
Class Timetable
File | F:/2027%20batch%20full%20stack%20developent%20lab%20manual/TIMETABLE.HTML
Type here to search
03:13
16-12-2024

Class Timetable

Weekly Class Timetable

Time	Monday	Tuesday	Wednesday	Thursday	Friday
9:00 - 10:00	DBMS	SE	DBMS	MEFA	OS
10:00 - 11:00	OS		P&S		DBMS
11:00 - 12:00	FSD LAB	DBMS	DTI	OS	P&S
12:00 - 1:00		P&S		DBMS	
1:00 - 2:00	Lunch Break				
2:00 - 3:00	SE	DBMS	MEFA	OS	DTI
3:00 - 4:00	MEFA	SE	FSD LAB	DBMS	DBMS LAB

1. **C)** Write a HTML program, to explain the working of forms, by designing registration form.

Note: include text field, password field, number field, date of birth field, checkboxes, radio buttons, list boxes using <select> ,<option> tags, <text area> and two buttons submit and reset. Use tables to provide better views.

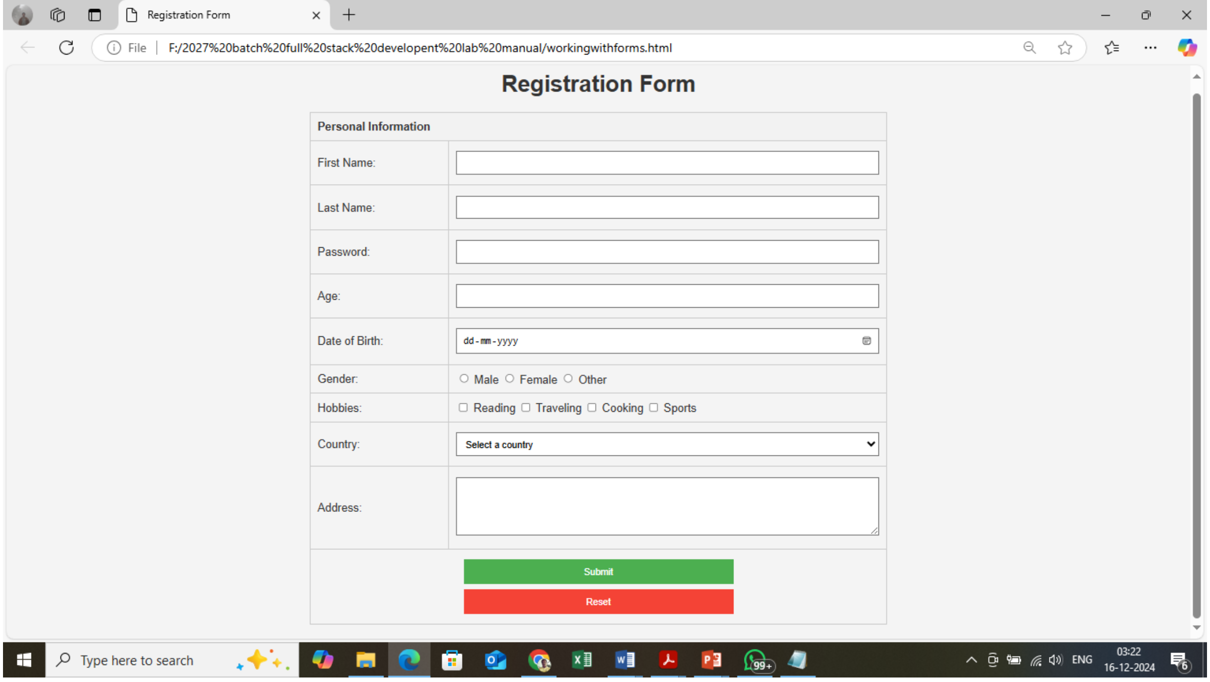
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport"
content="width=device-width, initial-
scale=1.0">
    <title>Registration Form</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #f4f4f4;
            color: #333;
            margin: 20px;
        }
        h1 {
            text-align: center;
            color: #333;
        }
        table {
            margin: 0 auto;
            border-collapse: collapse;
            width: 50%;
        }
        table, th, td {
            border: 1px solid #ccc;
        }
        th, td {
            padding: 10px;
            text-align: left;
        }
        th {
            background-color: #f2f2f2;
        }
        input[type="text"],
input[type="password"],
input[type="number"],    input[type="date"],
select, textarea {
            width: 100%;
            padding: 8px;
            margin: 4px 0;
            box-sizing: border-box;
        }
        input[type="submit"],
input[type="reset"] {
            width: 48%;
            padding: 10px;
            margin: 4px 1%;
```

```
      background-color: #4CAF50;
      color: white;
      border: none;
      cursor: pointer;
    }
    input[type="reset"] {
      background-color: #f44336;
    }
input[type="submit"]:hover,
input[type="reset"]:hover {
      opacity: 0.9;
    }
  </style>
</head>
<body>
  <h1>Registration Form</h1>
  <form action="#" method="post">
    <table>
      <tr>
        <th colspan="2">Personal
Information</th>
      </tr>
      <tr>
        <td>First Name:</td>
        <td><input type="text"
name="firstname" required></td>
      </tr>
      <tr>
        <td>Last Name:</td>
        <td><input type="text"
name="lastname" required></td>
```

```
      </tr>
      <tr>
        <td>Password:</td>
        <td><input type="password"
name="password" required></td>
      </tr>
      <tr>
        <td>Age:</td>
        <td><input type="number"
name="age" required></td>
      </tr>
      <tr>
        <td>Date of Birth:</td>
        <td><input type="date"
name="dob" required></td>
      </tr>
      <tr>
        <td>Gender:</td>
        <td>
          <input type="radio"
name="gender" value="male" required>
Male
          <input type="radio"
name="gender" value="female" required>
Female
          <input type="radio"
name="gender" value="other" required>
Other
        </td>
      </tr>
      <tr>
```

```
             <td>Hobbies:</td>
             <td>
               <input type="checkbox"
     name="hobby1" value="Reading"> Reading
               <input type="checkbox"
     name="hobby2" value="Traveling">
     Traveling
               <input type="checkbox"
     name="hobby3" value="Cooking"> Cooking
               <input type="checkbox"
     name="hobby4" value="Sports"> Sports
             </td>
           </tr>
           <tr>
             <td>Country:</td>
             <td>
               <select name="country"
     required>
<option value="">Select a country</option
<option value="USA">USA</option>
<option value="Canada">Canada</option>
<option value="UK">UK</option>
<option value="Australia">Australia</option
<option value="India">India</option>
</select>
      </td>
      </tr>
      <tr>
```

```
            <td>Address:</td>
      <td><textarea name="address" rows="4"
     required></textarea></td>
     </tr>
            <tr>
            <td colspan="2" style="text-align: center;">
           <input type="submit" value="Submit">
           <input type="reset" value="Reset">
                   </td>
                 </tr>
               </table>
             </form>
           </body>
           </html>
```


Registration Form
F:/2027%20batch%20full%20stack%20developent%20lab%20manual/workingwithforms.html
Type here to search
ENG
03:22
16-12-2024

Registration Form

Personal Information	
First Name:	
Last Name:	
Password:	
Age:	
Date of Birth:	dd-mm-yyyy
Gender:	○ Male ○ Female ○ Other
Hobbies:	□ Reading □ Traveling □ Cooking □ Sports
Country:	Select a country
Address:	
Submit Reset	

2. 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 attribute such that frames to be fixed).

```
<!DOCTYPE html>
<html>
<head>
  <title>Frames Example</title>
</head>
<frameset rows="33%, 33%, 34%"
frameborder="0" framespacing="0">
  <frame src="image.html"
name="image_frame" frameborder="0"
scrolling="no">
  <frame src="paragraph.html"
name="paragraph_frame"
frameborder="0" scrolling="no">
  <frame src="hyperlink.html"
name="hyperlink_frame"
frameborder="0" scrolling="no">
</frameset>
</html>

image.html
<!DOCTYPE html>
<html>
<head>
  <title>Image Frame</title>
</head>
<body>
```

```
  <img src="image.jpg" width="100%"
height="100%">
</body>
</html>
```

paragraph.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Paragraph Frame</title>
</head>
<body>
  <p>This is a paragraph of text in the
second frame.</p>
</body>
</html>
```

hyperlink.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Hyperlink Frame</title>
</head>
<body>
  <p><a href="(link unavailable)">Visit
(link unavailable)</a></p>
</body></html>
```

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>, <div>, <span>, tags.

```
<!DOCTYPE html>
```

```
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>HTML5 Semantic Tags Example</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
        }
        header, footer {
            background-color: #333;
            color: white;
            padding: 10px;
            text-align: center;
        }
        nav {
            background-color: #f4f4f4;
            padding: 10px;
            text-align: center;
        }
        main {
            margin: 20px;
        }
        section {
            background-color: #e2e2e2;
            padding: 15px;
```

```
      margin-bottom: 20px;
    }
    article {
      background-color: #f9f9f9;
      padding: 15px;
      margin: 10px 0;
    }
    aside {
      background-color: #e9e9e9;
      padding: 10px;
      margin-top: 10px;
    }
    figure {
      display: inline-block;
      margin: 10px;
    }
    figcaption {
      font-size: 0.9em;
      text-align: center;
    }
    .content {
      display: flex;
      justify-content: space-between;
    }
  </style>
</head>
<body>
<header>
  <h1>Welcome to My Website</h1>
  <p>Your one-stop destination for
information</p>
```

```
</header>
<nav>
    <ul>
        <li><a href="#home">Home</a></li>
        <li><a href="#about">About</a></li>
        <li><a
href="#services">Services</a></li>
        <li><a
href="#contact">Contact</a></li>
    </ul>
</nav>
<main>
    <section>
        <h2>Introduction</h2>
        <p>This is the main section of the page
where the content will be displayed.</p>
    </section>
    <div class="content">
        <article>
            <h2>Article 1: Exploring
HTML5</h2>
<p>HTML5 is the latest version of HTML
and includes many new features and
improvements...</p>
        </article>
        <article>
      <h2>Article 2: Understanding
CSS3</h2>
<p>CSS3 is the latest evolution of the
Cascading Style Sheets language...</p>
        </article>
```

```
  </div>
  <aside>
    <h3>Related Resources</h3>
    <p>Check out these resources for more
information on web development:</p>
    <ul>
      <li><a
href="https://developer.mozilla.org/en-
US/docs/Web/HTML">MDN Web
Docs</a></li>
      <li><a
href="https://www.w3schools.com/">W3Sc
hools</a></li>
      <li><a href="https://www.css-
tricks.com/">CSS-Tricks</a></li>
    </ul>
  </aside>
```

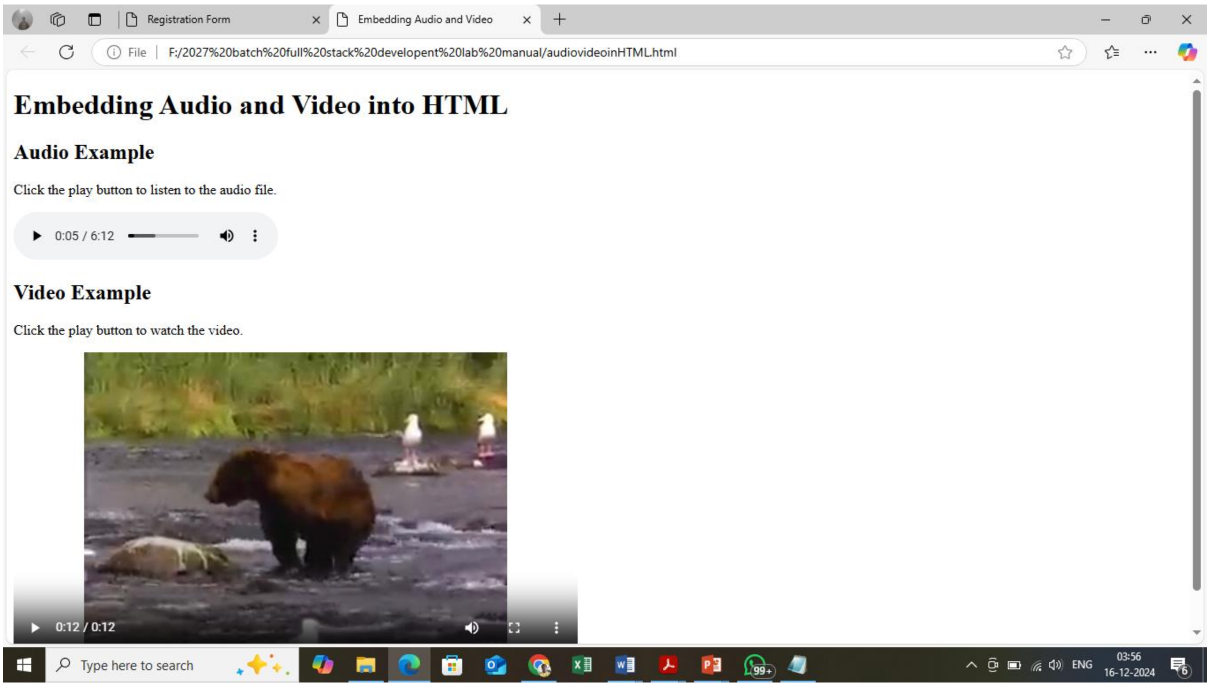
```
    <figure>
 <img src="https://via.placeholder.com/300"
alt="HTML5 Logo">
<figcaption>HTML5 Logo</figcaption>
    </figure>
</main>
<footer>
    <p>&copy; 2024 My Website. All Rights
Reserved.</p>
</footer>
</body>
</html>
```

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

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


    <h1>Embedding Audio and Video into
HTML</h1>
    <!-- Audio Section -->
    <section>
        <h2>Audio Example</h2>
        <p>Click the play button to listen to the audio
file.</p>
        <audio controls>
<source
src="https://www.soundhelix.com/examples/mp3/So
undHelix-Song-1.mp3" type="audio/mp3">Your
browser does not support the audio element.
 </audio>
    </section>
    <!-- Video Section -->
    <section>
        <h2>Video Example</h2>
        <p>Click the play button to watch the video.</p>
<video width="640" height="360" controls>
<source
src="https://www.w3schools.com/html/movie.mp4"
type="video/mp4">
<source
src="https://www.w3schools.com/html/movie.ogg"
type="video/ogg"> Your browser does not support
the video element.
        </video>
    </section>
</body>
```

```
</html>
```



Registration Form
Embedding Audio and Video
File
F:/2027%20batch%20full%20stack%20developent%20lab%20manual/audiovideoinHTML.html
0:05 / 6:12
0:12 / 0:12
Type here to search
ENG
03:56
16-12-2024

Embedding Audio and Video into HTML

Audio Example

Click the play button to listen to the audio file.

Video Example

Click the play button to watch the video.

3.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).

Index.html

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Styling Examples</title>
    <!-- Link to external CSS file -->
    <link rel="stylesheet" href="styles.css">
    <!-- Internal CSS -->
    <style>
        .internal-style {
            color: blue;
            font-size: 20px;
        }
    </style>
</head>
<body>
    <h1 style="color: red; font-size: 24px;">Inline Style</h1>
    <p class="internal-style">This paragraph uses internal styles.</p>
    <div class="external-style">This div uses external styles.</div>
</body>
</html>
```

External css style.css

```
.external-style {
    color: green;
    font-size: 18px;
}
```

4. Selector forms

4.a) Write a program to apply different types of selectors forms.

i) Simple selector (element, id, class, group, universal)

ii) combinatory selector(descendent, child, adjacent sibling, general sibling)

iii) Pseudo class selector

iv) Pseudo element selector

v) Attribute selector.

4.a) Write a program to apply different types of selectors forms.

i) Simple selector (element, id, class, group, universal)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Selectors Examples</title>
    <!-- Internal CSS to demonstrate various selectors -->
    <style>
```

```
/* Element Selector */
p {
    color: blue;
    font-size: 18px;
}
/* ID Selector */
#unique-element {
    background-color: lightgray;
    padding: 10px;
}
/* Class Selector */
.highlight {
    color: red;
    font-weight: bold;
}
/* Group Selector */
h1, h2, h3 {
    color: green;
}
/* Universal Selector */
* {
    margin: 10px;
    padding: 5px;
```

```
        }
    </style>
</head>


<body>
    <!-- Element Selector Example -->
    <p>This paragraph is styled using an element selector.</p>
    <!-- ID Selector Example -->
    <div id="unique-element">This div is styled using an ID selector.</div>
    <!-- Class Selector Example -->
    <span class="highlight">This text is styled using a class selector.</span>
    <!-- Group Selector Example -->
    <h1>This is a heading 1 styled using a group selector.</h1>
    <h2>This is a heading 2 styled using a group selector.</h2>
    <h3>This is a heading 3 styled using a group selector.</h3>
    <!-- Universal Selector Example -->
    <p>All elements have margin and padding set by the universal selector.</p>
    <div>All elements have margin and padding set by the universal selector.</div>
</body>
</html>
```

How to Run

1. Create a file named `index.html` and type the aboveHTML code into it.
2. Open the `index.html` file in a web browser to see the different styles applied to the elements.

4.ii)combinatory selector(descendent, child, adjacent sibling, general sibling)

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Combinatory Selectors Examples</title>
    <!-- Internal CSS to demonstrate various combinatory selectors -->
    <style>
        /* Descendant Selector */
        .container p {
            color: blue;
        }
        /* Child Selector */
        .container > p {
            font-weight: bold;
        }

        /* Adjacent Sibling Selector */
        h2 + p {
            color: green;
        }
        /* General Sibling Selector */
        h2 ~ p {
            font-size: 14px;
        }
    </style>
</head>
<body>
    <div class="container">
        <p>This paragraph is a descendant of .container and is styled using the descendant selector.</p>
        <div>
```

```
            <p>This paragraph is a descendant of .container but not a direct child, so only the color is applied.</p>
        </div>
        <p>This paragraph is a direct child of .container and is styled using both the descendant and child selectors.</p>
    </div>
    <h2>This heading is an adjacent sibling example.</h2>
    <p>This paragraph is an adjacent sibling to the h2 and is styled using the adjacent sibling selector.</p>
    <p>This paragraph is also a sibling of the h2 but is not immediately adjacent, so only the general sibling selector applies.</p>
</body>
</html>
```

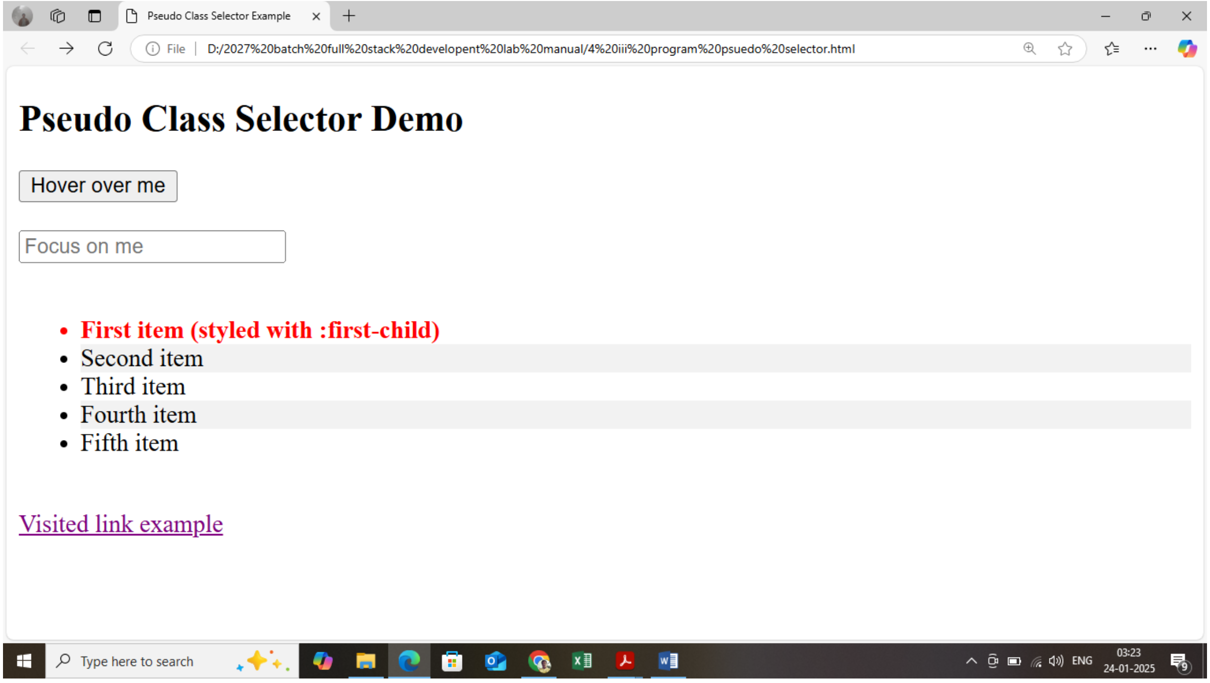
4. iii) Pseudo class selector

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

```
/* Change background color when hovering over a button */
        button:hover {
          background-color: lightblue;
          color: white;
        }

/* Change border color when an input field is focused */
        input:focus {
          border: 2px solid green;
        }
        /* Style the first child of a list */
        ul li:first-child {
          font-weight: bold;
          color: red;
        }
        /* Style every even child in the list */
        ul li:nth-child(even) {
          background-color: #f2f2f2;
        }
        /* Change color when a link is visited */
        a:visited {
          color: purple;
        }
```

```
            </style>
        </head>
        <body>
            <h2>Pseudo Class Selector Demo</h2>
            <button>Hover over me</button>
            <br><br>
<input type="text" placeholder="Focus on me">
            <br><br>
             <ul>
                <li>First item (styled with :first-child)</li>
                <li>Second item</li>
                <li>Third item</li>
                <li>Fourth item</li>
                <li>Fifth item</li>
            </ul>
            <br>
<a href="https://google.com">Visited link example</a>
        </body>
        </html>
```


Pseudo Class Selector Example
File
D:/2027%20batch%20full%20stack%20developent%20lab%20manual/4%20iii%20program%20psuedo%20selector.html
Type here to search
ENG
03:23
24-01-2025

Pseudo Class Selector Demo

Hover over me

Focus on me

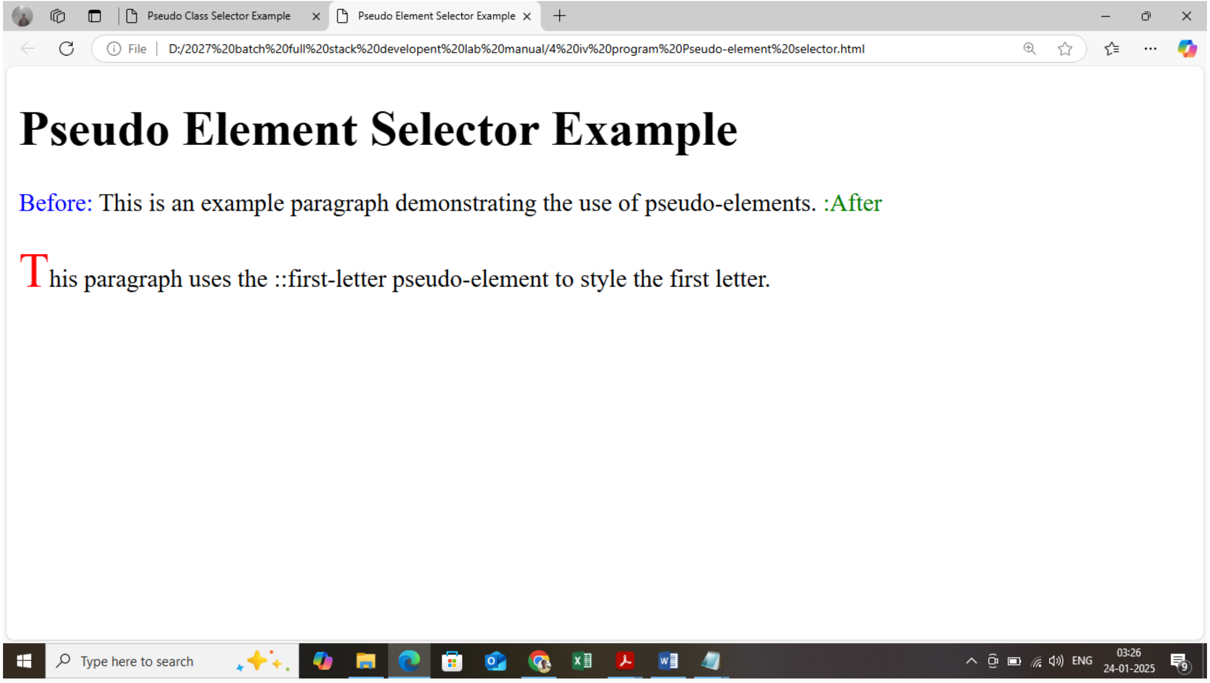
- First item (styled with :first-child)
- Second item
- Third item
- Fourth item
- Fifth item

Visited link example

4.iv) Pseudo element selector

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Pseudo Element Selector Example</title>
    <style>
        .example::before {
            content: "Before: ";
            color: blue;
        }
        .example::after {
            content: " :After";
            color: green;
        }
        .highlight::first-letter {
            font-size: 2em;
            color: red;
        }
    </style>
</head>
<body>
    <h1>Pseudo Element Selector Example</h1>
    <p class="example">This is an example paragraph demonstrating the use of pseudo-elements.</p>
    <p class="highlight">This paragraph uses the ::first-letter pseudo-element to style the first letter.</p>
</body>
</html>
```

- The `::before` pseudo-element is used to insert content before the content of the selected element. In this case, it adds "Before: " before the paragraph with the class `example`.
- The `::after` pseudo-element is used to insert content after the content of the selected element. In this case, it adds " :After" after the paragraph with the class `example`.
- The `::first-letter` pseudo-element is used to style the first letter of the selected element. Here, it enlarges and changes the color of the first letter of the paragraph with the class `highlight`.


Pseudo Class Selector Example
Pseudo Element Selector Example
File
D:/2027%20batch%20full%20stack%20developent%20lab%20manual/4%20iv%20program%20Pseudo-element%20selector.html
Type here to search
ENG
03:26
24-01-2025

Pseudo Element Selector Example

Before: This is an example paragraph demonstrating the use of pseudo-elements. :After

This paragraph uses the ::first-letter pseudo-element to style the first letter.

4.v) Attribute selector.

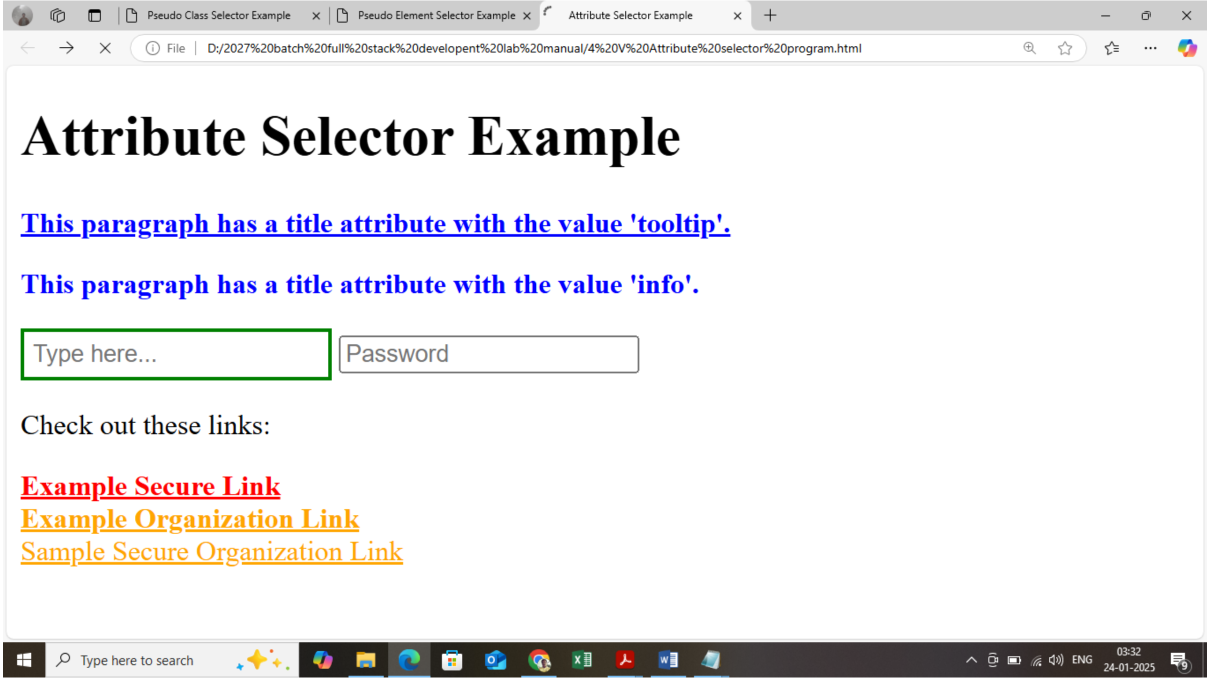
```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Attribute Selector Example</title>
    <style>
        /* Selects all elements with a title attribute */
        [title] {
            color: blue;
            font-weight: bold;
        }
```

```
/* Selects all elements with a title attribute equal to 'tooltip' */
[title="tooltip"] {
    text-decoration: underline;
}
/* Selects all input elements with a type attribute equal to 'text' */
input[type="text"] {
    border: 2px solid green;
    padding: 5px;
}
/* Selects all links (a elements) where the href attribute starts with 'https' */
a[href^="https"] {
    color: red;
}
/* Selects all links (a elements) where the href attribute ends with '.org' */
a[href$=".org"] {
    color: orange;
}
/* Selects all links (a elements) where the href attribute contains 'example' */
a[href*="example"] {
    font-weight: bold;
```

```
            }
        </style>
    </head>
    <body>
        <h1>Attribute Selector Example</h1>
        <p title="tooltip">This paragraph has a title attribute with the value 'tooltip'.</p>
        <p title="info">This paragraph has a title attribute with the value 'info'.</p>
<input type="text" placeholder="Type here...">
<input type="password" placeholder="Password">
        <p>Check out these links:</p>
        <a href="https://www.example.com">Example Secure Link</a><br>
        <a href="http://www.example.org">Example Organization Link</a><br>
        <a href="https://www.sample.org">Sample Secure Organization Link</a>
    </body>
    </html>
```

- The `[title]` selector styles all elements that have a `title` attribute, making their text color blue and bold.
- The `[title="tooltip"]` selector styles elements with a `title` attribute exactly equal to `tooltip`, adding an underline.
- The `input[type="text"]` selector styles input elements with a `type` attribute equal to `text`, adding a green border and padding.
- The `a[href^="https"]` selector styles links (`a` elements) where the `href` attribute starts with `https`, changing their text color to red.

- The `a[href$=".org"]` selector styles links where the `href` attribute ends with `.org`, changing their text color to orange.
- The `a[href*="example"]` selector styles links where the `href` attribute contains the word `example`, making their text bold.


Pseudo Class Selector Example
Pseudo Element Selector Example
Attribute Selector Example
File
D:/2027%20batch%20full%20stack%20developent%20lab%20manual/4%20V%20Attribute%20selector%20program.html
Type here to search
ENG
03:32
24-01-2025

Attribute Selector Example

This paragraph has a title attribute with the value 'tooltip'.

This paragraph has a title attribute with the value 'info'.

Type here... Password

Check out these links:

Example Secure Link
Example Organization Link
Sample Secure Organization Link

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

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

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Color Reference</title>
    <style>
        body {
```

```
    font-family: Arial, sans-serif;
    margin: 20px;
}
.color-box {
    width: 150px;
    height: 150px;
    margin: 10px;
    display: inline-block;
    color: white;
    text-align: center;
    line-height: 150px;
    font-weight: bold;
    border-radius: 8px;
}
.color-name {
    background-color: red;
}
.hex {
    background-color: #00FF00;
}
.rgb {
    background-color: rgb(0, 0, 255);
}
```

```
        .rgba {
            background-color: rgba(255, 255, 0, 0.5);
            color: black;
        }
        .hsl {
            background-color: hsl(240, 100%, 50%);
        }
        .hsla {
            background-color: hsla(120, 100%, 50%,
0.3);
        }
    </style>
</head>
<body>
    <h1>CSS Color Reference Demonstration</h1>
    <div class="color-box color-name">red</div>
    <div class="color-box hex">#00FF00</div>
    <div class="color-box rgb">rgb(0, 0, 255)</div>
    <div class="color-box rgba">rgba(255, 255, 0,
0.5)</div>
    <div class="color-box hsl">hsl(240, 100%,
50%)</div>
    <div class="color-box hsla">hsla(120, 100%,
50%, 0.3)</div>
```

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

Explanation:

- **Color Name (`red`)**: This is a predefined color name that browsers recognize.
- **Hex (`#00FF00`)**: This is a hexadecimal color code.
- **RGB (`rgb(0, 0, 255)`)**: This uses the RGB color model where each parameter (red, green, blue) defines the intensity of the color as an integer between 0 and 255.
- **RGBA (`rgba(255, 255, 0, 0.5)`)**: This is similar to RGB but includes an alpha parameter for transparency, ranging from 0 (fully transparent) to 1 (fully opaque).
- **HSL (`hsl(240, 100%, 50%)`)**: This uses the HSL color model which stands for hue, saturation, and lightness.
- **HSLA (`hsla(120, 100%, 50%, 0.3)`)**: This is similar to HSL but includes an alpha parameter for transparency.

When you open this HTML file in a browser, you will see a demonstration of different ways to reference colors in CSS. Each colored box represents a different method of specifying colors.

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

```
body {
    /* Set the background image */
    background-image: url('path/to/your/image.jpg');
/* Place the image halfway down the page and tiling it horizontally */
    background-position: center 50%;
/* Ensure the image remains in place when scrolling */
    background-attachment: fixed;
 /* Tile the image horizontally */
    background-repeat: repeat-x;
 /* Optionally, set a background color as a fallback */
```

```
    background-color: #f0f0f0;
}
```

Explanation:

- `background-image: url('path/to/your/image.jpg');`: Specifies the background image to be used.
- `background-position: center 50%;`: Positions the background image in the center horizontally and halfway down the page vertically.
- `background-attachment: fixed;`: Fixes the background image in place so that it doesn't move when the user scrolls.
- `background-repeat: repeat-x;`: Tiles the background image horizontally.
- `background-color: #f0f0f0;`: Sets a fallback background color in case the image is not available.

You can add this CSS rule to your existing CSS file or within a `<style>` block in your HTML document. Just make sure to replace `'path/to/your/image.jpg'` with the actual path to your background image.

5.c) Write a program using the following ites related to CSS font and text

i) font-size

ii) font-weight

iii) font-style

iv) text-decoration

v) text-transformation

vi) text – alignment

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Font and Text Properties</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 20px;
        }
        .example {
            margin: 20px 0;
        }
        .font-size {
            font-size: 24px; /* Adjust the font size */
        }
        .font-weight {
            font-weight: bold; /* Set the font weight */
        }
        .font-style {
            font-style: italic; /* Set the font style */
        }
        .text-decoration {
            text-decoration: underline; /* Set the text decoration */
        }
        .text-transform {
            text-transform: uppercase; /* Transform the text to uppercase */
        }
        .text-align {
            text-align: center; /* Align the text to the center */
        }
    </style>
</head>
<body>
    <h1>CSS Font and Text Properties Demonstration</h1>
    <div class="example font-size">This text has a font size of 24px.</div>
    <div class="example font-weight">This text is bold.</div>
    <div class="example font-style">This text is italic.</div>
    <div class="example text-decoration">This text is underlined.</div>
    <div class="example text-transform">This text is uppercase.</div>
    <div class="example text-align">This text is centered.</div>
</body>
</html>
```

Explanation:

- **`font-size`**: Sets the size of the font. In the example, it's set to `24px`.
- **`font-weight`**: Sets the weight (or boldness) of the font. In the example, it's set to `bold`.
- **`font-style`**: Sets the style of the font, such as `italic` or `normal`. In the example, it's set to `italic`.
- **`text-decoration`**: Adds decorations to the text, such as `underline`, `overline`, or `line-through`. In the example, it's set to `underline`.
- **`text-transform`**: Controls the capitalization of the text. It can be set to `uppercase`, `lowercase`, or `capitalize`. In the example, it's set to `uppercase`.
- **`text-align`**: Sets the horizontal alignment of the text. It can be `left`, `right`, `center`, or `justify`. In the example, it's set to `center`.

This HTML and CSS code demonstrates how to apply these properties to different text elements on a web page. You can open this HTML file in a web browser to see the effects of each CSS property.

5.d) write a program to explain the importance of CSS Box model using

i. content ii.border iii. Margin iv. Padding

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Box Model Demonstration</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            margin: 20px;
        }
        .box {
            width: 200px; /* Width of the content */
            margin: 20px; /* Margin around the element */
            padding: 15px; /* Padding inside the element */
            border: 5px solid #333; /* Border around the element */
            background-color: #f0f0f0; /* Background color for visibility */
        }
        .box .content {
            background-color: #fff; /* Background color for the content area */
            padding: 10px; /* Padding inside the content area */
        }
    </style>
</head>
<body>
    <h1>CSS Box Model Demonstration</h1>
    <div class="box">
        <div class="content">
            <p>This is the content area.</p>
            <p>The CSS Box Model includes:</p>
            <ul>
```

```
                <li><strong>Content</strong>: The actual content of the element (text, images, etc.)</li>
                <li><strong>Padding</strong>: Space between the content and the border</li>
                <li><strong>Border</strong>: The border surrounding the padding (and content)</li>
                <li><strong>Margin</strong>: Space outside the border, separating the element from others</li>
            </ul>
        </div>
    </div>
</body>
</html>
```

Explanation:

1. **Content**: The content is the innermost part of the box, which holds the actual text, images, or other media. In the example, the content is contained within the `<div class="content">` element.
2. **Padding**: Padding is the space between the content and the border. It creates inner space within the element. In the example, `padding: 15px;` creates space between the content and the border of the `.box` element, and `padding: 10px;` creates additional space inside the content area.
3. **Border**: The border surrounds the padding and content. It can have different styles, widths, and colors. In the example, `border: 5px solid #333;` creates a solid black border around the `.box` element.
4. **Margin**: The margin is the outermost part of the box, creating space between the element and other elements. In the example, `margin: 20px;` creates space outside the border of the `.box` element, separating it from other elements on the page.

How to view:

To see the CSS Box Model in action, open this HTML file in a web browser. You will see a box with clearly defined content, padding, border, and margin, illustrating how each part of the box model affects the layout and spacing of the element.

6.Applying JavaScript – Internal and external I/O type conversion.

6.a) write a program to embed internal and external javascript in a web page.

script.js

```
// script.js
function showAlertExternal() {
    alert('This is an external JavaScript alert!');
}
```

HTML FILE

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Embedding Internal and External JavaScript</title>
    <script src="script.js"></script> <!-- Link to external JavaScript file -->
    <script>
        // Internal JavaScript
        function showAlertInternal() {
            alert('This is an internal JavaScript alert!');
        }
```

```
  </script>
</head>
<body>
  <h1>Embedding Internal and External JavaScript</h1>
  <button onclick="showAlertInternal()">Internal JavaScript Alert</button>
  <button onclick="showAlertExternal()">External JavaScript Alert</button>
</body>
</html>
```

Explanation:

1. **External JavaScript**:
   - The external JavaScript file `script.js` contains a function `showAlertExternal` that displays an alert when called.
   - The HTML file links to this external JavaScript file using the `<script src="script.js"></script>` tag in the `<head>` section.
2. **Internal JavaScript**:
   - The internal JavaScript is included directly within the HTML file inside a `<script>` tag in the `<head>` section.
   - The internal JavaScript contains a function `showAlertInternal` that displays an alert when called.
3. **HTML Content**:
   - The HTML body contains two buttons. Each button has an `onclick` attribute that calls one of the JavaScript functions.
   - The first button calls the `showAlertInternal` function, demonstrating the use of internal JavaScript.
   - The second button calls the `showAlertExternal` function, demonstrating the use of external JavaScript.

How to Run:

1. Save the external JavaScript code in a file named `script.js`.
2. Save the HTML code in a file named `index.html`.
3. Open `index.html` in a web browser.
4. Click the buttons to see alerts triggered by the internal and external JavaScript functions.

6.b) write a program to explain the different ways for displaying output.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>JavaScript Output Methods</title>
    <style>
        #output {
            border: 1px solid #000;
            padding: 10px;
            margin-top: 10px;
            width: 300px;
        }
    </style>
</head>
<body>
    <h1>JavaScript Output Methods</h1>

    <!-- Buttons to trigger different output methods -->
    <button onclick="displayAlert()">Alert</button>
    <button onclick="writeToDocument()">Document Write</button>
    <button onclick="writeToElement()">InnerHTML</button>
    <button onclick="writeToConsole()">Console Log</button>

    <!-- Div for innerHTML output -->
    <div id="output">InnerHTML Output will app
```

6.c) write a program to explain the different ways for taking input.

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>JavaScript Output Methods</title>
    <style>
        #output {
            border: 1px solid #000;
            padding: 10px;
            margin-top: 10px;
            width: 300px;
        }
    </style>
</head>
<body>
    <h1>JavaScript Output Methods</h1>

    <!-- Buttons to trigger different output methods -->
    <button onclick="displayAlert()">Alert</button>
    <button onclick="writeToDocument()">Document Write</button>
    <button onclick="writeToElement()">InnerHTML</button>
    <button onclick="writeToConsole()">Console Log</button>

    <!-- Div for innerHTML output -->
    <div id="output">InnerHTML Output will appear here.</div>
    <script>
        // Method 1: Using alert()
        function displayAlert() {
            alert('This is an alert message!');
        }

        // Method 2: Using document.write()
        function writeToDocument() {
            document.write('<p>This text was written
using document.write().</p>');
            document.write('<p style="color:
red;">Note: This will overwrite the entire document
content if called after the document is fully
loaded.</p>');
        }

        // Method 3: Using innerHTML
        function writeToElement() {

document.getElementById('output').innerHTML =
'This text was written using innerHTML.';
        }

        // Method 4: Using console.log()
        function writeToConsole() {
            console.log('This message is logged to the
console.');
```

```
      }
    </script>
</body>
</html>
```

Explanation:

1. **Alert**: The `alert()` method displays a message in an alert dialog box.
    - `function displayAlert()` shows a simple alert with a message.
2. **Document Write**: The `document.write()` method writes directly to the HTML document.
    - `function writeToDocument()` writes text to the document. Note that calling `document.write()` after the document has fully loaded will overwrite the entire content of the document.
3. **InnerHTML**: The `innerHTML` property allows you to write or replace HTML content within a specific HTML element.
    - `function writeToElement()` sets the inner HTML of a `<div>` with the ID `output`.
4. **Console Log**: The `console.log()` method outputs a message to the web console, which is useful for debugging purposes.
    - `function writeToConsole()` logs a message to the console.

How to Run:

1. Save the HTML code in a file named `index.html`.
2. Open `index.html` in a web browser.
3. Click the buttons to see the different methods of displaying output. Check the console (usually accessible via developer tools in the browser) for the console log output.

6.d) create a webpage which uses prompt dialogue box to ask a voter for his name and age . display the information in table format along with either the voter can vote or not.

```
<!DOCTYPE html>
<html>
<head>
  <title>Voter Information</title>
  <style>
    table {
      border-collapse: collapse;
    }
    th, td {
      border: 1px solid black;
      padding: 10px;
    }
  </style>
</head>
<body>
  <h2>Voter Information</h2>
  <button onclick="getVoterInfo()">Get Voter Info</button>
  <div id="voterInfo"></div>

  <script>
    function getVoterInfo() {
      let name = prompt("Please enter your name:");
      let age = parseInt(prompt("Please enter your age:"));
      let votingEligibility = (age >= 18) ? "Eligible" : "Not Eligible";
      let voterInfoHtml = `
        <table>
          <tr>
            <th>Name</th>
            <th>Age</th>
            <th>Voting Eligibility</th>
          </tr>
          <tr>
            <td>${name}</td>
            <td>${age}</td>
            <td>${votingEligibility}</td>
          </tr>
        </table>
      `;

document.getElementById("voterInfo").innerHTML = voterInfoHtml;
    }
  </script>
</body>
</html>
```

7. JavaScript Pre-defined and User-defined objects

7 a) Write a program using document object properties and methods

7 b) write a program using window object properties and methods

7 c) write a program using array object properties and methods

7 d) write a program using math object properties and methods

7 e) write a program using string object properties and methods

7 f) write a program using regex object properties and methods

7 g) write a program using date object properties and methods

7 h) write a program to explain user defined object by using properties, methods, accessors, constructors and disply