

WEEK1

Lists, Links and Images

1a)

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

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

Description:

In HTML, list tags are used to display a group of related items in a structured way. HTML provides different types of lists, each serving a specific purpose.

- `` - Creates an unordered list (bulleted).
- `` - Creates an ordered list (numbered or lettered).
- `` - Defines a list item inside `` or ``.
- `<dl>` - Creates a description list (used for definitions or terms).
- `<dt>` - Defines a term/name in a description list.
- `<dd>` - Defines a description/detail for the term in `<dt>`.

list.html code:

```
<!DOCTYPE html>

<html>

<head>

  <title>HTML Lists </title>

</head>

<body>

  <h1>Working with Lists in HTML</h1>

  <!-- Ordered List -->

  <h2>Ordered List</h2>
```


CSE

AGRIC

MECH

<!-- Unordered List -->

<h2>Unordered List</h2>

BLACK

WHITE

PINK

<!-- Nested Lists -->

<h2>Nested List</h2>

Frontend

 JavaScript

CSS

html

Backend

Node.js

```
        <li>Python</li>
    </ul>
</li>
</ol>
```

```
<!-- Ordered List inside Unordered List -->
<h2>Ordered List inside Unordered List</h2>
<ul>
```

```
    <li>Web Technologies
        <ol>
            <li>HTML</li>
            <li>CSS</li>
            <li>JavaScript</li>
        </ol>
    </li>
    <li>Database
        <ol>
            <li>MySQL</li>
            <li>MongoDB</li>
        </ol>
    </li>
</ul>
```

```
<!-- Definition List -->
<h2>Definition List</h2>
```

<dl>

<dt>HTML</dt>

<dd>HyperText Markup Language</dd>

<dt>CSS</dt>

<dd>Cascading Style Sheets</dd>

<dt>JS</dt>

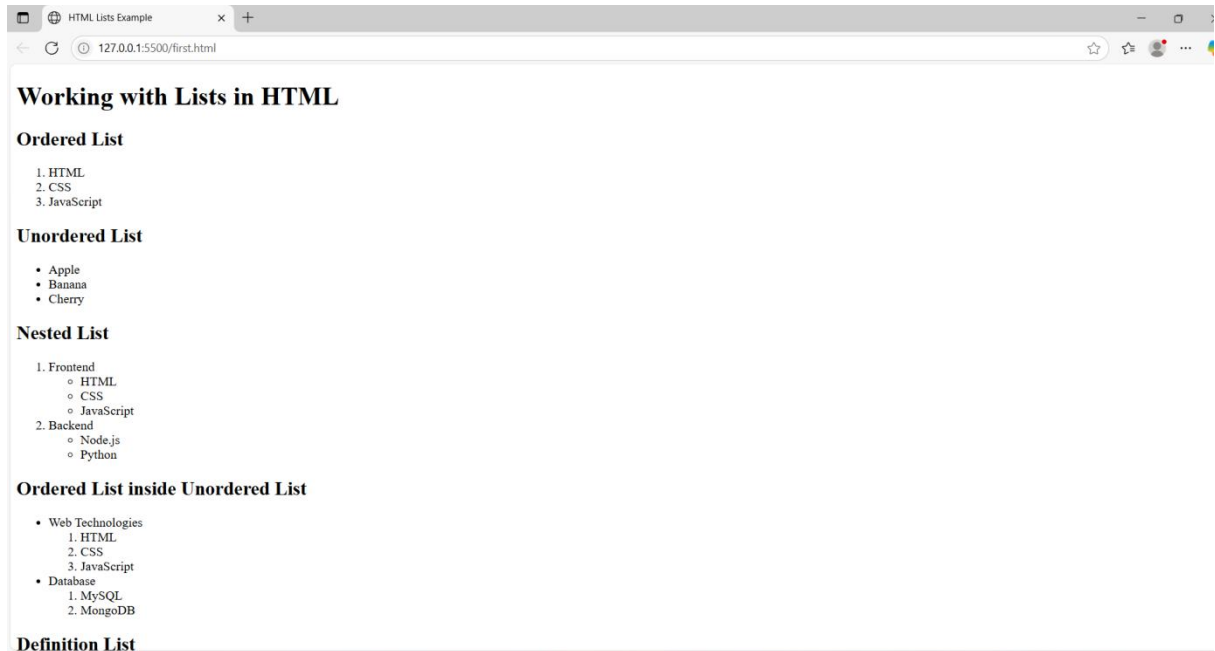
<dd>JavaScript – Scripting language for web pages</dd>

</dl>

</body>

</html>

Output:



1 b)

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

Note: Use text to link <https://www.aec.edu.in/>

Use image to link <https://www.aec.edu.in/?p=Gallery>

Description:

<a> (Anchor Tag):

Used to create hyperlinks to other web pages, email addresses, or files.

href (Hypertext REference):

Specifies the URL or link destination.

target Attribute:

Specifies where to open the linked document.

_blank opens the link in a new tab.

_self (default) opens in the same tab.

** (Image Tag):**

Used to embed images in a web page.

src defines the image path.

alt provides alternative text if the image doesn't load.

width sets the image width in pixels.

Code:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Hyperlink Example</title>
```

```
</head>
```

```
<body>
```

```
  <h1>Working with Hyperlinks in HTML</h1>
```

```
  <!-- Text Link -->
```

```
  <h2>Text Hyperlink</h2>
```

```
  <p>
```

```
    Visit the official website of
```

```
    <a href="https://www.aec.edu.in/" target="_blank">Aditya Engineering College</a>
```

```
    for more information.
```

```
  </p>
```

```
  <!-- Image Link -->
```

```
  <h2>Image Hyperlink</h2>
```

```
  <p>Click the image below to visit the AEC Gallery:</p>
```

```
  <a href="https://www.aec.edu.in/?p=Gallery" target="_blank">
```

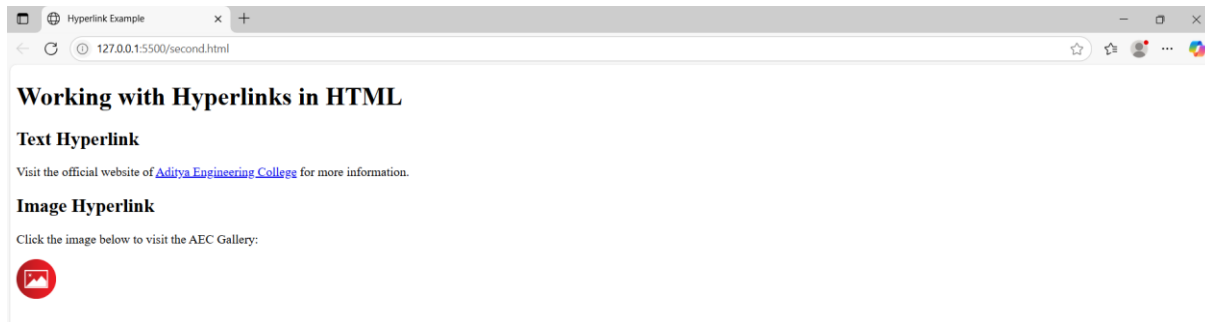
```
    
```

```
  </a>
```

```
</body>
```

```
</html>
```

Output:



1 c)

AIM: Create a HTML document that has your image and your friend's image with a specific height and width. Also when clicked on the images it should navigate to their respective profiles.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Profile Image Links</title>
```

```
</head>
```

```
<body>
```

```
  <h1>My Profile and My Friend's Profile</h1>
```

```
  <!-- Your Image -->
```



```

<a href="https://www.linkedin.com/in/your-profile" target="_blank">

     </a>

<p>Click on my image to visit my profile</p>

<!-- Friend's Image -->

<a href="https://www.linkedin.com/in/friends-profile" target="_blank">

</a>

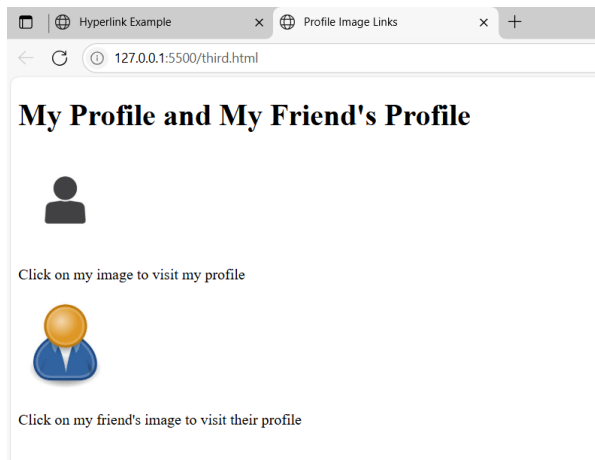
<p>Click on my friend's image to visit their profile</p>

</body>

</html>

```

Output:



1 d)

AIM: 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 thumbnails image is also a link to a full-sized version of the image. Create an image gallery using this technique.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Thumbnail Image Gallery</title>

</head>

<body>

  <h1>Thumbnail Image Gallery</h1>

  <p>Click on any image to view the full-size version.</p>

  <!-- Gallery Thumbnails -->

  <a href="JBL.jpg" target="_blank">

  </a>

  <a href="https://picsum.photos/id/1015/600/400" target="_blank">

     </a>

  <a href="https://picsum.photos/id/1016/600/400" target="_blank">

  </a>

  <a href="https://picsum.photos/id/1018/600/400" target="_blank">

  </a>

  <a href="https://picsum.photos/id/1020/600/400" target="_blank">

  </a>

  <a href="https://picsum.photos/id/1021/600/400" target="_blank">

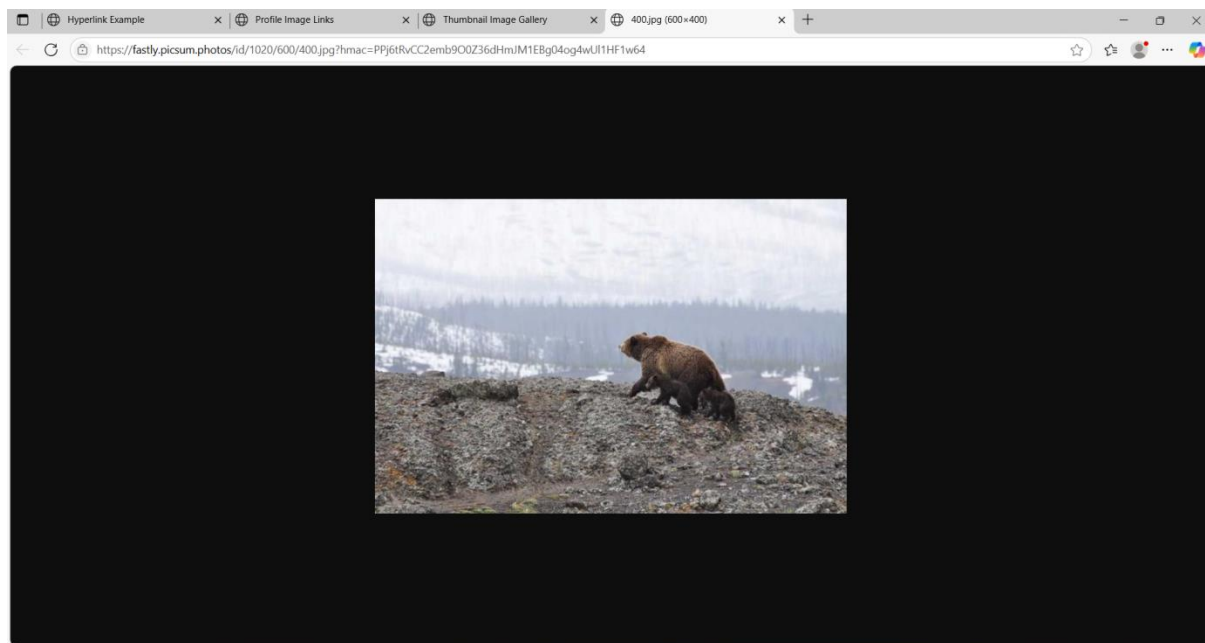
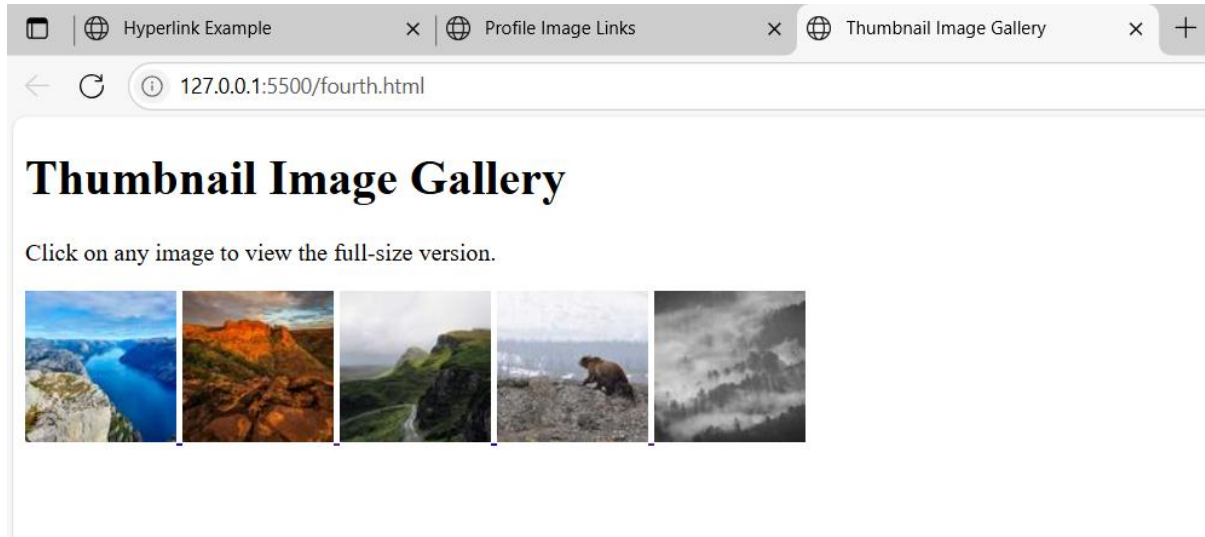
    

  </a>

</body>

</html>
```

Output:



Viva Questions:

1) What is the difference between `` and ``?

Ans) `` creates an ordered list with numbers or letters; `` creates an unordered list with bullets.

2) Which tag is used to define a list item?

Ans) The `` tag is used inside `` or `` to define a list item.

3) How do you create a hyperlink in HTML?

Ans) By using the `<a>` tag. Example: `Click here`.

4) What attribute is required in the `<a>` tag to define the link?

Ans) The `href` attribute specifies the URL of the link.

5) What is the use of the `target="_blank"` attribute in an anchor tag?

Ans) It opens the linked page in a new tab or window.

6) Which tag is used to insert an image in HTML?

Ans) The `` tag is used to embed an image.

7) What are the required attributes of the `` tag?

Ans) `src` (image URL) and `alt` (alternate text).

8) Can we make an image clickable? If yes, how?

Ans) Yes, wrap the `` tag inside an `<a>` tag.

Example: ``

9) What is the use of the `<dl>`, `<dt>`, and `<dd>` tags?

Ans) They create a description list:

- `<dl>`: starts the list
- `<dt>`: defines the term
- `<dd>`: gives the definition

10) How can you display an image of fixed size?

Ans) Use width and height attributes.

Example: ``

Week-2: Tables, Forms and Frames

2 a)

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

Description:

The `<table>` tag in HTML is used to create a table structure to display data in rows and columns.

Tag	Purpose
<code><table></code>	- Starts the table structure.
<code><tr></code>	- Defines a table row.
<code><th></code>	- Defines a header cell (bold and centered by default).
<code><td></code>	- Defines a data cell (normal text).
<code><thead></code>	- Groups the header content. Helps in accessibility and styling.
<code><tbody></code>	- Groups the main body content of the table.
<code><tfoot></code>	- Groups the footer content, usually summary or totals.
<code><caption></code>	- Adds a title to the table (appears above the table).

Attribute	Used With	Description
border	<table>	Sets the border around table and cells
cellpadding	<table>	Adds space inside each cell.
cellspacing	<table>	Adds space between cells.
colspan	<td>, <th>	Merges multiple columns into one cell.
rowspan	<td>, <th>	Merges multiple rows into one cell.
align	<table>, <tr>, <td>, <th>	Aligns content horizontally.
valign	<td>, <th>	Aligns content vertically .
width, height	<table>, <td>	Sets size (not recommended now; use CSS).

Code:

```
<!DOCTYPE html>

<html>

<head>

  <title>Employee Information Table</title>

</head>

<body>  <h1>Company Employee Information</h1>

  <table border="1" cellpadding="10" cellspacing="0">

    <!-- Table Header -->

    <tr>

      <th rowspan="2">Emp ID</th>

      <th rowspan="2">Name</th>
```

```

        <th rowspan="2">Department</th>

        <th rowspan="2">Designation</th>

        <th colspan="2">Contact Info</th>

        <th rowspan="2">Salary</th>

        <th rowspan="2">Address</th>

    </tr>

    <tr>

        <th>Email</th>

        <th>Phone</th>

    </tr>

    <!-- Employee 1 -->

    <tr>

        <td>EMP101</td>

        <td>Alice Johnson</td>

        <td>HR</td>

        <td>Manager</td>

        <td>alice.hr@company.com</td>

        <td>9876543210</td>

        <td>$60,000</td>

        <td>New York, USA</td>

    </tr>

    <!-- Employee 2 -->

    <tr>

        <td>EMP102</td>

        <td>Bob Smith</td>

```

```
<td>IT</td>

<td>Software Engineer</td>

<td>bob.it@company.com</td>

<td>9123456780</td>

<td>$80,000</td>

<td>San Francisco, USA</td>

</tr>

<!-- Employee 3 -->

<tr>

<td>EMP103</td>

<td>Charlie Lee</td>

<td>Finance</td>

<td>Analyst</td>

<td>charlie.finance@company.com</td>

<td>9988776655</td>

<td>$70,000</td>

<td>Chicago, USA</td>

</tr>

</table>

</body>

</html>
```

Output:

Company Employee Information

Emp ID	Name	Department	Designation	Contact Info		Salary	Address
				Email	Phone		
EMP101	Alice Johnson	HR	Manager	alice.hr@company.com	9876543210	\$60,000	New York, USA
EMP102	Bob Smith	IT	Software Engineer	bob.it@company.com	9123456780	\$80,000	San Francisco, USA
EMP103	Charlie Lee	Finance	Analyst	charlie.finance@company.com	9988776655	\$70,000	Chicago, USA

2 b)

AIM: 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.).

Code:

```
<!DOCTYPE html>

<html>

<head>

    <title>B.Tech I Semester Timetable</title>

</head>

<body>

    <h1 align="center">B.Tech I Semester Timetable</h1>

    <table border="2" cellspacing="4" cellpadding="10" align="center">

        <caption><strong>Department of Computer Science & Engineering</strong></caption>

        <tr>

            <th>Day</th>

            <th>9:00 - 10:00</th>

            <th>10:00 - 11:00</th>

            <th>11:00 - 12:00</th>

            <th>12:00 - 1:00</th>

            <th>1:00 - 2:00</th>

            <th>2:00 - 3:00</th>

            <th>3:00 - 4:00</th>
```

</tr>

<!-- Monday -->

<tr>

<td>Monday</td>

<td>Mathematics-I</td>

<td>Physics</td>

<td>English</td>

<td rowspan="5" align="center">Lunch Break</td>

<td>Python Programming</td>

<td>Engineering Graphics</td>

<td>Library</td>

</tr>

<!-- Tuesday -->

<tr>

<td>Tuesday</td>

<td>Engineering Chemistry</td>

<td>English</td>

<td>Workshop</td>

<td>Mathematics-I</td>

<td>Physics</td>

<td>Sports</td>

</tr>

<!-- Wednesday -->

<tr>

<td>Wednesday</td>

```

        <td>Python Lab</td>

        <td colspan="2">Python Programming Lab</td>

        <td>English</td>

        <td>Engineering Graphics</td>

        <td>Mentoring Hour</td>

    </tr>

    <!-- Thursday -->

    <tr>

        <td>Thursday</td>

        <td>Physics Lab</td>

        <td colspan="2">Physics Laboratory</td>

        <td>Engineering Chemistry</td>

        <td>Mathematics-I</td>

        <td>Workshop</td>

    </tr>

    <!-- Friday -->

    <tr>

        <td>Friday</td>

        <td>Engineering Graphics</td>

        <td>Physics</td>

        <td>Mathematics-I</td>

        <td>Python Programming</td>

        <td>Library</td>

        <td>Sports</td>

    </tr>

```

```

</table>

</body>

</html>

```

Output:

B.Tech I Semester Timetable

127.0.0.1:5500/Week-2/table2.html

<

2 c)

AIM: 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 ie: submit and reset. Use tables to provide a better view).

Description:

The <form> tag is used to create an HTML form for collecting user input. Forms can be used to send data to a server, such as login information, feedback, survey responses, or file uploads.

Form Element	Purpose	Basic Syntax
<form>	Container for all form elements	<form action="submit.php" method="post">
<input>	Accepts various types of user input (text, email, etc.)	<input type="text" name="username">
<label>	Defines a label for an input element	<label for="email">Email:</label>
<textarea>	Multiline text input (e.g., comments)	<textarea name="comment" rows="4" cols="30"></textarea>
<select>	Dropdown list	<select name="country">...</select>
<option>	Defines an option in a dropdown	<option value="IN">India</option>
<button>	Clickable button (can submit or reset form)	<button type="submit">Submit</button>
<fieldset>	Groups related elements in a form	<fieldset>... </fieldset>
<legend>	Caption for a fieldset	<legend>Login Info</legend>
<input type="radio">	Select one option from multiple choices	<input type="radio" name="gender" value="male">
<input type="checkbox">	Select multiple options	<input type="checkbox" name="hobby" value="reading">

<input type="submit">	Submits the form	<input type="submit" value="Send">
<input type="reset">	Resets the form fields	<input type="reset" value="Clear">
<input type="file">	Allows file upload	<input type="file" name="upload">
<input type="password">	Hides text input (for passwords)	<input type="password" name="pass">
<input type="email">	Validates email format input	<input type="email" name="useremail">
<input type="number">	Accepts only numeric values	<input type="number" name="quantity" min="1" max="10">

Important Attributes in <form>:

Attribute	Purpose	Example
action	URL where form data will be sent	action="submit.php"
method	HTTP method: "get" or "post"	method="post"
name	Name of the form or element	name="loginForm"
id	Unique identifier	id="form1"
value	Default or submitted value of form field	value="Submit"
placeholder	Shows hint inside input field	placeholder="Enter your name"
required	Makes a field mandatory	required
readonly	Field cannot be edited	readonly
disabled	Field is disabled and not submitted	disabled
autocomplete	Enables or disables auto-fill by browser	autocomplete="off"

Code:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Extended Student Registration Form</title>
```

```
</head>
```

```
<body>
```

```
    <h2 align="center">Extended Student Registration Form</h2>
```

```
    <form action="#" method="post" enctype="multipart/form-data">
```

```
        <table border="1" cellpadding="10" cellspacing="0" align="center">
```

```
            <caption><strong>Student Details</strong></caption>
```

```
            <tr>
```

```
                <th>Full Name:</th>
```

```
                <td><input type="text" name="fullname" placeholder="Enter full name"
```

```
required></td>
```

```
            </tr>
```

```
            <tr>
```

```
                <th>Email:</th>
```

```
                <td><input type="email" name="email" placeholder="example@email.com"
```

required></td>

</tr>

<tr>

<th>Password:</th>

<td><input type="password" name="password" required></td>

</tr>

<tr>

<th>Mobile Number:</th>

<td><input type="tel" name="phone" pattern="[0-9]{10}" placeholder="10-digit
number" required></td>

</tr>

<tr>

<th>Age:</th>

<td><input type="number" name="age" min="17" max="30"></td>

</tr>

<tr>

<th>Date of Birth:</th>

<td><input type="date" name="dob"></td>

</tr>

<tr>

<th>Gender:</th>

<td>

<input type="radio" name="gender" value="Male"> Male

<input type="radio" name="gender" value="Female"> Female

<input type="radio" name="gender" value="Other"> Other


```

        </td>
    </tr>
    <tr>
        <th>Courses Interested:</th>
        <td>
            <input type="checkbox" name="course" value="C"> C
            <input type="checkbox" name="course" value="C++"> C++
            <input type="checkbox" name="course" value="Java"> Java
            <input type="checkbox" name="course" value="Python"> Python
            <input type="checkbox" name="course" value="Web"> Web Development
        </td>
    </tr>
    <tr>
        <th>Branch:</th>
        <td>
            <select name="branch">
                <option value="">--Select Branch--</option>
                <option value="CSE">Computer Science</option>
                <option value="ECE">Electronics</option>
                <option value="EEE">Electrical</option>
                <option value="MECH">Mechanical</option>
                <option value="CIVIL">Civil</option>
            </select>
        </td>
    </tr>

```

```
<tr>

  <th>College Website:</th>

  <td><input type="url" name="website"
placeholder="https://www.example.com"></td>

</tr>

<tr>

  <th>Address:</th>

  <td><textarea name="address" rows="4" cols="30" placeholder="Enter your
address"></textarea></td>

</tr>

<tr>

  <th>Upload Resume:</th>

  <td><input type="file" name="resume"></td>

</tr>

<tr>

  <th>Preferred Time Slot:</th>

  <td><input type="time" name="timeslot"></td>

</tr>

<tr>

  <th>Skill Level (0-10):</th>

  <td><input type="range" name="skill" min="0" max="10"></td>

</tr>

<tr>

  <th>Favorite Color:</th>

  <td><input type="color" name="color"></td>
```

```
</tr>

<!-- Hidden Field -->

<tr>

  <td colspan="2">

    <input type="hidden" name="registration_type" value="student">

  </td>

</tr>

<tr>

  <th colspan="2" align="center">

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

    <input type="reset" value="Reset">

  </th>

</tr>

</table>

</form>





</body>

</html>
```

Output:

Student Registration Form

Student Details

Full Name:	<input type="text" value="Enter full name"/>
Email:	<input type="text" value="example@email.com"/>
Password:	<input type="password"/>
Mobile Number:	<input type="text" value="10-digit number"/>
Age:	<input type="text"/>
Date of Birth:	<input type="text" value="dd-mm-yyyy"/> 
Gender:	<input type="radio"/> Male <input type="radio"/> Female <input type="radio"/> Other
Courses Interested:	<input type="checkbox"/> C <input type="checkbox"/> C++ <input type="checkbox"/> Java <input type="checkbox"/> Python <input type="checkbox"/> Web Development
Branch:	<input type="text" value="--Select Branch--"/> 
College Website:	<input type="text" value="https://www.example.com"/>
Address:	<div><input type="text" value="Enter your address"/></div> 
Upload Resume:	<input type="button" value="Choose File"/> No file chosen
Preferred Time Slot:	<input type="text" value="--:--"/> 
Skill Level (0-10):	<div><div></div></div>
Favorite Color:	<div><div></div></div>
<div><input type="button" value="Submit"/> <input type="button" value="Reset"/></div>	

2 d)

AIM: 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).

Description:

The <frameset> tag was used in HTML 4 to divide the browser window into multiple frames, each capable of displaying a different HTML document simultaneously. It replaces the <body> tag and allows developers to create a multi-pane view in one browser window.

Purpose of <frameset>:

- To split the web page into rows or columns.
- To load multiple documents at the same time within one page.
- To enable side-by-side navigation, like menus and content areas.

Code:

image.html:

```
<!DOCTYPE html>

<html>

<head>

  <title>Image Frame</title>

</head>

<body>

  <h2 align="center">Welcome to Our Site</h2>

  <div align="center">

  </div>

</body>
```

`</html>`

paragraph.html:

`<!DOCTYPE html>`

`<html>`

`<head>`

`<title>Paragraph Frame</title>`

`</head>`

`<body>`

`<h3 align="center">About Us</h3>`

`<p align="justify" style="padding: 10px;">`

This website demonstrates the usage of HTML frames by dividing the layout into three horizontal sections.

The first frame shows an image, the second frame provides some descriptive text, and the third frame lists useful links.

Frames are used to present multiple HTML documents within a single browser window.

`</p>`

`</body>`

`</html>`

links.html:

`<!DOCTYPE html>`

`<html>`

`<head>`

`<title>Links Frame</title>`

`</head>`

`<body>`

`<h3 align="center">Useful Links</h3>`

```
<ul style="text-align: center; list-style-type: none;">

  <li><a href="https://www.google.com" target="_blank">Google</a></li>

  <li><a href="https://www.wikipedia.org" target="_blank">Wikipedia</a></li>

  <li><a href="https://www.w3schools.com" target="_blank">W3Schools</a></li>

</ul>

</body>

</html>
```

framesetdemo.html:

```
<!DOCTYPE html>

<html>

<head>

  <title>Frameset Example</title>

</head>

<frameset rows="30%,40%,30%">

  <frame src="image.html" name="frame1">

  <frame src="paragraph.html" name="frame2">

  <frame src="links.html" name="frame3">

</frameset>

<body>

  <p>This page uses frames but your browser does not support them.</p>

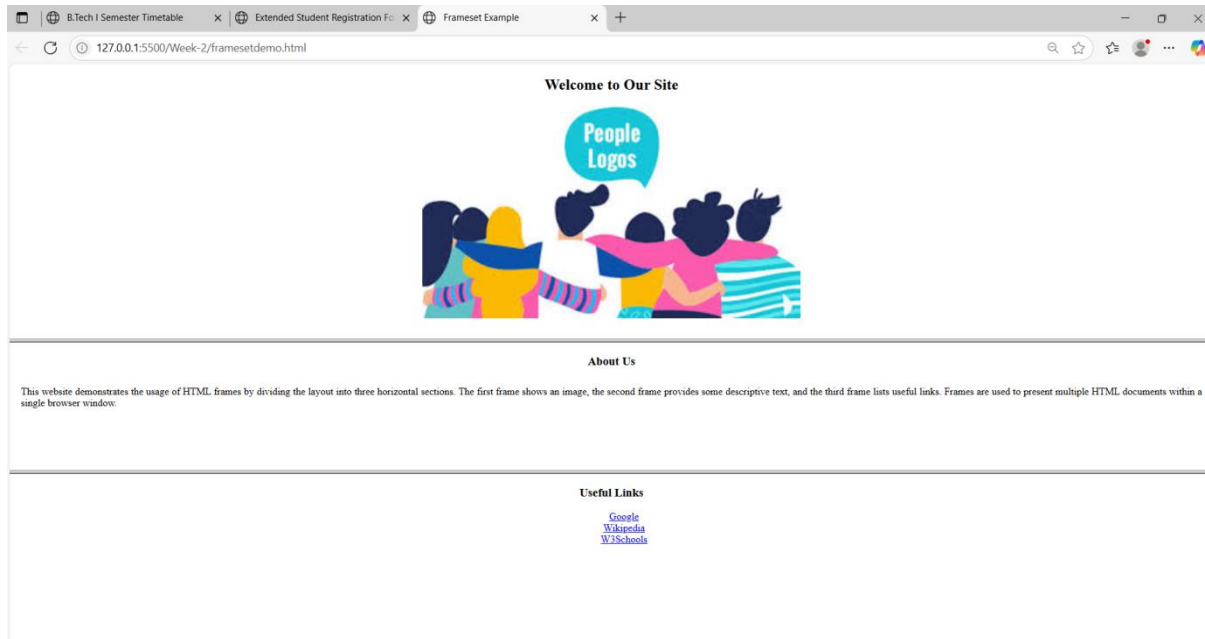
</body>

</frameset>

</frameset>
```

</html>

Output:



Viva Questions:

1) What is the use of the <table> tag?

Ans) It creates a table to display data in rows and columns.

2) What is the difference between <th> and <td>?

Ans) <th> is a header cell (bold), <td> is a normal cell.

3) Which tag is used to give a title to a table?

Ans) <caption> is used to add a title to the table.

4) What is the default alignment of <th> text?

Ans) Center aligned and bold.

5) What is the use of colspan?

Ans) It merges two or more columns into one cell.

6) What is the use of rowspan?

Ans) It merges two or more rows into one cell.

7) What does cellpadding do?

Ans) It adds space inside the table cell.

8) What does cellspacing do?

Ans) It adds space between the table cells.

Working with Forms

9) What is the purpose of the <form> tag in HTML?

Ans) The <form> tag is used to create a form that collects user input and sends it to a server.

10) What are the two main attributes of the <form> tag?)

Ans) action: URL where the form data is sent

method: HTTP method (GET or POST) used to submit the form

11) What is the use of the <input> tag?)

Ans) The <input> tag is used to create various input fields like textboxes, checkboxes, radio buttons, etc.

12) What is the difference between radio and checkbox inputs?

Ans) radio: allows only one option to be selected from a group and checkbox: allows multiple options to be selected

13) What is the use of the <select> and <option> tags?

Ans) <select> creates a dropdown menu and <option> defines each choice in the dropdown

14) What does the required attribute do in form fields?

Ans) It makes the input field mandatory, so the form can't be submitted without filling it.

Working with Frameset:

15) What is the purpose of the <frameset> tag in HTML?

Ans) The <frameset> tag is used to divide the browser window into multiple sections (frames), where each frame can display a different HTML document.

16) What attribute is used in <frameset> to split the window vertically?

Ans) The cols attribute is used to divide the screen into vertical frames (columns).

Example: <frameset cols="30%,70%">

17) What attribute is used in <frameset> to split the window horizontally?

Ans) The rows attribute is used to divide the screen into horizontal frames (rows).

Example: <frameset rows="50%,50%">

18) Which tag is used inside <frameset> to define each frame?

Ans) The <frame> tag is used to define each frame and load separate documents.

19) How do you load a webpage in a frame?

Ans) Use the src attribute in the <frame> tag.

Example: <frame src="menu.html">

20) What is the use of the noresize attribute in <frame>?

Ans) It prevents the user from resizing the frame manually.

Week-3: HTML5 and CSS

3a)

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

Description:

Tag	Purpose
<code><header></code>	Represents the top section of the page, usually with a title, logo, or introductory content.
<code><nav></code>	Contains navigation links to other parts of the site or page.
<code><main></code>	Wraps the central content that is unique to the page.
<code><article></code>	Represents independent, self-contained content like news articles or blog posts.
<code><section></code>	Divides content into meaningful sections, often with a heading. Used inside articles or main content.
<code><figure></code>	Used to group media like images or diagrams with an optional caption.
<code><figcaption></code>	Caption for the <code><figure></code> element, describing the media content.
<code><aside></code>	Represents related or sidebar content that complements the main content.
<code><footer></code>	Represents the bottom section, usually with copyright, contact, or summary info.
<code><div></code>	A generic container used for styling or grouping elements; has no semantic meaning.
<code></code>	An inline container used to style a part of text (e.g., highlight words).

Code:

```

<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <title>Aside as Sidebar</title>

  <style>

    body {

      margin: 0;

      font-family: Arial, sans-serif;

    }

    header, footer {

```

```
background-color: #333;  
  
color: white;  
  
padding: 10px;  
  
text-align: center;  
}
```

```
nav {  
  
background-color: #eee;  
  
padding: 10px;  
}
```

```
nav a{  
  
margin-right: 10px;  
}
```

```
.container {  
  
display: flex;  
  
width: 100%;  
  
height: 400px;  
}
```

```
main {  
  
flex: 3;  
  
padding: 10px;
```

```
background-color: #f9f9f9;  
}
```

```
aside {  
  /* flex: 1;  
  padding: 20px;  
  background-color: #dfefff; */  
  width: 15%;  
  padding: 15px;  
  margin-left: 25px;  
  float: right;  
  font-style: italic;  
  background-color: #f0f0f0;  
  border-left: 3px solid #0070d0;  
}
```

```
footer {  
  margin-top: 10px;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<header>
```

<h1>Welcome to My College News Portal</h1>

</header>

<nav>

Home

Departments

Events

Contact

</nav>

<div class="container">

<main>

<h2>Main Content</h2>

<p>This is the main content of the page.</p>

<article>

<h2>Annual Tech Fest 2025</h2>

<section>

<p>The Tech Fest 2025 will be held from September 10 to 12. It will include coding contests, robotics, and workshops.</p>

</section>

<figure>

<figcaption>Figure: Tech Fest 2025 Poster</figcaption>

</figure>

</article>

`</main>`

`<aside>`

`<h3>Sidebar Tips</h3>`

`<p>Use semantic tags for clean HTML structure.</p>`

`</aside>`

`</div>`

`<footer>`

`<p>© 2025 My Website</p>`

`</footer>`

`</body>`

`</html>`

Output:

Welcome to My College News Portal

[Home](#) [Departments](#) [Events](#) [Contact](#)

Main Content

This is the main content of the page.

Annual Tech Fest 2025

The Tech Fest 2025 will be held from September 10 to 12. It will include coding contests, robotics, and workshops.



Figure: Tech Fest 2025 Poster

Sidebar Tips

Use semantic tags for clean HTML structure.

3 b)

AIM: Write a HTML program, to embed audio and video into HTML web page

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
    <title>Embedding Audio and Video</title>
```

```
</head>
```

```
<body>
```

```
    <h2>HTML5 Audio and Video Embedding</h2>
```

```
    <!-- Embed Audio -->
```

```
    <section>
```

```
        <h3>Audio Example:</h3>
```

```
        <audio controls>
```

```
            <source src="https://www.soundhelix.com/examples/mp3/SoundHelix-Song-1.mp3"
type="audio/mpeg"> Your browser does not support the audio element.
```

</audio>

</section>

<!-- Embed Video -->

<section>

<h3>Video Example:</h3>

<video width="600" height="340" controls>

<source src="https://www.w3schools.com/html/mov_bbb.mp4" type="video/mp4">

Your browser does not support the video tag.

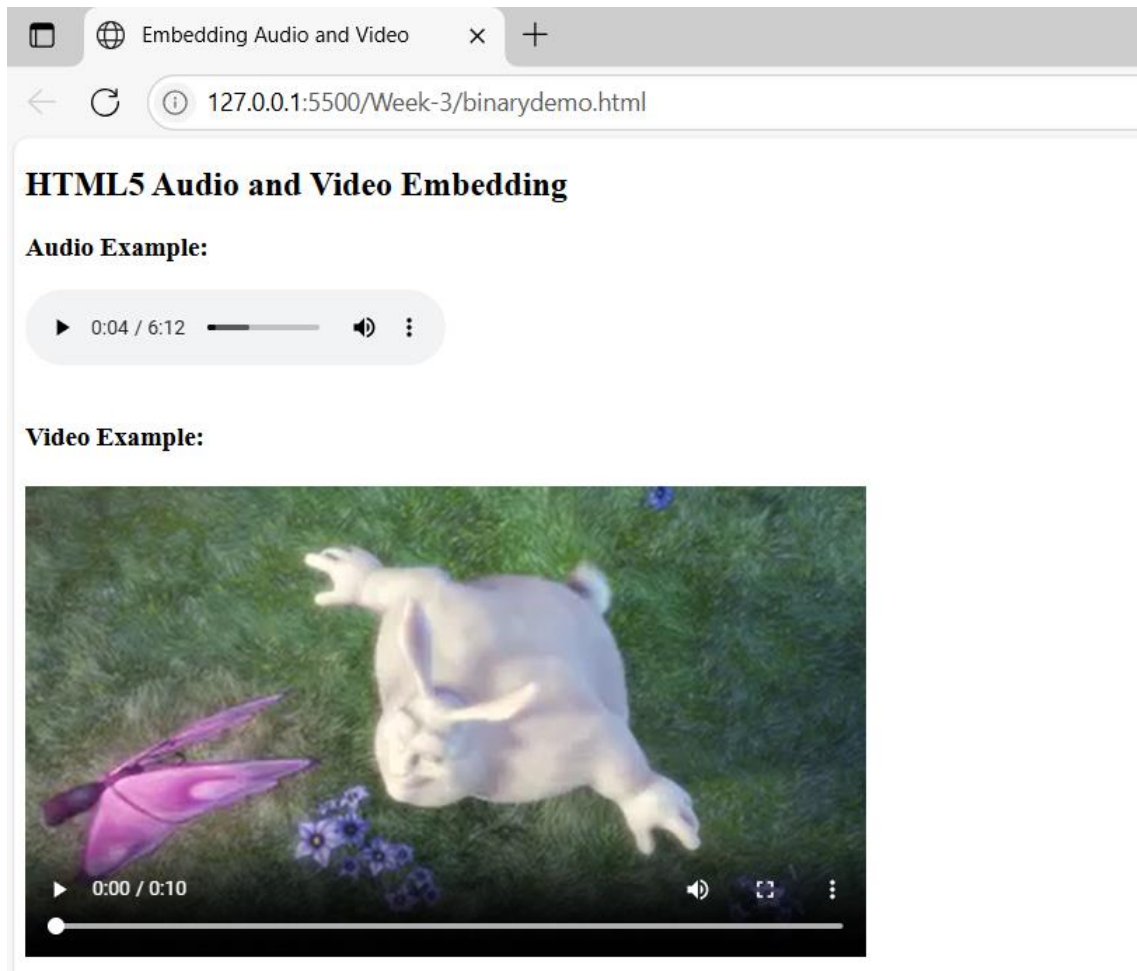
</video>

</section>

</body>

</html>

Output:



3 c)

AIM: 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).

Description:

CSS Style Specification Formats:

- 1. Inline CSS – inside the HTML tag using style attribute**
- 2. Internal CSS – inside a <style> block in the <head>**
- 3. External CSS – in a separate .css file and linked using <link> (simulated here)**

Each CSS rule has:

- Selector – the HTML element(s) to style**
- Property – the style attribute to change**

- Value – the setting for that property

Syntax:

```
selector {  
  
    property: value;  
  
}
```

Example 1 (from *inline* CSS):

```
<h1 style="color: blue;">Hello</h1>
```

- Selector: h1 (implied, though it's inline)
- Property: color
- Value: blue

Example 2 (from *internal* CSS):

```
h2 {  
  
    color: darkgreen;  
  
    font-family: Georgia, serif;  
  
}
```

Selector: h2

- First:
 - Property: color
 - Value: darkgreen
- Second:
 - Property: font-family
 - Value: Georgia, serif

Example 3 (from *external* CSS):

```
.external-paragraph {  
  
    font-size: 18px;  
  
    color: darkred;  
  
}
```

Selector: .external-paragraph (a class selector)

- **First:**
 - **Property: font-size**
 - **Value: 18px**
- **Second:**
 - **Property: color**
 - **Value: darkred**

ID Selector:

Syntax:

```
#idname {  
  
    property: value;  
  
}
```

Ex:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
    #highlight {
```

```
        background-color: yellow;
```

```
border: 2px solid orange;

padding: 10px;
}

</style>

</head>

<body>

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

  <p id="highlight">This paragraph has an ID and special style.</p>

</body>

</html>
```

Code:

index.html:

```
<!DOCTYPE html>

<html lang="en">

<head>

  <meta charset="UTF-8">

  <title>CSS Styles Demo</title>

  <!-- Internal CSS -->

  <style>

    h2 {

      color: darkgreen;

      font-family: Georgia, serif;

    }

    .internal-box {
```

```
background-color: lightyellow;

border: 2px solid orange;

padding: 10px;
}
</style>

<!-- External CSS -->

<link rel="stylesheet" type="text/css" href="styles.css">

</head>

<body>


<h1 style="color: blue; font-size: 28px;">Inline Styled Heading (h1)</h1>

<h2>Internal Styled Heading (h2)</h2>

<div class="internal-box">

    This box uses <strong>internal CSS</strong> for styling.

</div>

<br>

<p class="external-paragraph">

    This paragraph is styled using <strong>external CSS</strong>.

</p>

</body>

</html>
```

styles.css:

```
/* External CSS styles */

.external-paragraph {
```



```
font-size: 18px;

color: darkred;

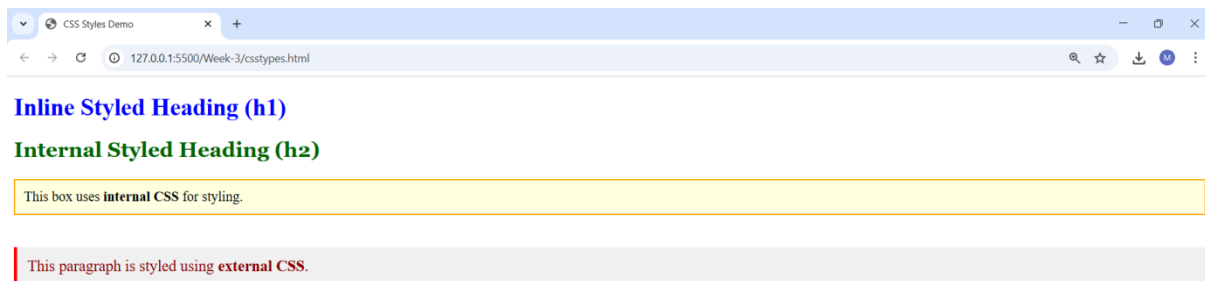
background-color: #f0f0f0;

padding: 12px;

border-left: 4px solid red;

}
```

Output:



Viva Questions:

1. What is the purpose of the <article> tag in HTML?

Ans) The <article> tag is used to represent independent, self-contained content such as blog posts, news articles, or forum posts.

2. What does the <aside> tag represent in HTML?

Ans) The <aside> tag is used for content that is tangentially related to the main content, like sidebars, tips, or advertisements.

3. What is the <figure> tag used for?

Ans) The <figure> tag is used to enclose media or illustrations, such as images, diagrams, or code snippets.

4. What is the role of <figcaption> in HTML?

Ans) The <figcaption> tag provides a caption or description for the content inside the <figure> tag.

5. What does the <footer> tag define?

Ans) The <footer> tag represents the footer of a document or a section, typically containing copyright, contact info, or related links.

6. What is the <header> tag used for?

Ans) The <header> tag defines the introductory content or navigation links for a section or entire webpage.

7. What is the use of the <main> tag in HTML5?

Ans) The <main> tag specifies the central content of a webpage that is unique and not repeated across pages like headers or footers.

8. What is the purpose of the <nav> tag?

Ans) The <nav> tag is used to define a section that contains navigation links like menus, tables of contents, etc.

9. How is the <section> tag different from <div>?

Ans) <section> is a semantic tag used to group related content with a heading, while <div> is a generic container without any semantic meaning.

10. What is the difference between <div> and tags?

Ans) <div> is a block-level container used for structuring page layout, while is an inline container used for styling small portions of text.

Week-4: Selector Forms

4 a)

AIM: Write a program to apply different types of selector forms

- i. Simple selector (element, id, class, group, universal)
- ii. Combinator selector (descendant, child, adjacent sibling, general sibling)
- iii. Pseudo-class selector
- iv. Pseudo-element selector
- v. Attribute selector

Description:

Simple Selector Type Example		Description
Element	h1	Selects all <h1>
ID	#main-title	Selects element with that ID
Class	.highlight	Selects elements with that class
Group	h2, p, li	Applies same rule to multiple elements
Universal	*	Applies to all elements
Combinator	Example Matches...	
Descendant	div p	All <p> inside <div>
Child	ul > li	Direct children only

Simple Selector Type	Example	Description
----------------------	---------	-------------

Adjacent	<code>h3 + p</code>	First <code><p></code> immediately after <code><h3></code>
----------	---------------------	--

General	<code>h3 ~ p</code>	All <code><p></code> after <code><h3></code> siblings
---------	---------------------	---

Pseudo-class	Example
--------------	---------

<code>:hover</code>	Link effects
---------------------	--------------

<code>:first-child</code> , <code>:nth-child(n)</code>	Position-based selection
--	--------------------------

Pseudo-element	Example
----------------	---------

<code>::first-line</code>	Style first line of text
---------------------------	--------------------------

<code>::before</code> , <code>::after</code>	Inject content before/after element
--	-------------------------------------

Explanation Summary:

Attribute Selector	Example
--------------------	---------

<code>[type="text"]</code>	Exact attribute value
----------------------------	-----------------------

<code>[href^="https"]</code>	Starts with https
------------------------------	-------------------

<code>[placeholder*="name"]</code>	Contains "name"
------------------------------------	-----------------

4 a) Simple Selectors

Code:

```
<!DOCTYPE html>

<html>

<head>

  <title>Simple Selectors Demo</title>

  <style>

    /* Universal Selector */

    * {

      font-family: Arial, sans-serif;

    }


    /* Type Selector */

    h1 {

      color: darkblue;

    }


    /* Class Selector */

    .highlight {

      background-color: yellow;

      font-weight: bold;

    }


    /* ID Selector */

    #main-para {

      color: green;

      font-size: 18px;
```

```
}

/* Group Selector */

h2, p {
    margin-bottom: 10px;
}

</style>

</head>

<body>

<h1>Welcome to the Selector Demo</h1>

<h2>This is a subheading</h2>

<p>This paragraph is affected by the group selector.</p>

<p id="main-para">This paragraph is styled using an ID selector.</p>

<p class="highlight">This paragraph uses a class selector for highlighting.</p>

</body>

</html>
```

Output:



4 b) Combinator Selectors:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
.university{
```

```
width: 50%;
```

```
border: 8px solid gray;
```

```
border-radius: 5px;
```

```
margin: 50px auto;
```

```
}
```

```
div p{
```

```
background-color: aquamarine;
```

```
}
```

```
/*div > p{
```

```
background-color: blanchedalmond;
```

```
}*/
```



```
/* h2+p{
    background-color: lightblue;
} */

/* h2~p{
    background-color: lightgray;
} */

</style>

</head>

<body>

<h1>Aditya University Information</h1>

<div class="university">

    <div class="department">

        <h2>CSE</h2>

        <p>Head: Ms. Sudha Rani T</p>

        <p>Associate Profissor</p>

        <div class="faculty">

            <h2>Faculty</h2>

            <p>Prof. Ramesh</p>

            <p>Prof. Sivaram</p>

        </div>

    </div>

    <div class="department">

        <h2>IT</h2>

        <p>Head: Ms.A. Rajababu</p>

        <p>Associate Profissor</p>
```

```
<div class="faculty">

<h2>Faculty</h2>

<p>Prof. Balaram</p>

<p>Prof. Kalyan</p>

</div>

</div>

<p>This is testing University styles</p>

</div>

<p>This is for testing paragraph styles</p>

</body>

</html>
```

Output:

127.0.0.1:5500/Week2/4b.html

Aditya University Information

CSE

Head: Ms. Sudha Rani T

Associate Professsor

Faculty

Prof. Ramesh

Prof. Sivaram

IT

Head: Ms.A. Rajababu

Associate Professsor

Faculty

Prof. Balaram

Prof. Kalyan

This is testing University styles

This is for testing paragraph styles

4 c) Pseudo-class selector

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Login Form with Pseudo-Classes</title>
```

```
<style>
```

```
body {
```

```
    font-family: Arial, sans-serif;
```

```
    padding: 20px;
```

```
}
```

```
.form {  
  
    width: 300px;  
  
    margin: auto;  
  
    padding: 20px;  
  
    border: 2px solid #ccc;  
  
    border-radius: 8px;  
  
}
```

```
input[type="text"],  
input[type="password"] {  
  
    width: 90%;  
  
    padding: 10px;  
  
    margin-bottom: 15px;  
  
    border: 1px solid #ccc;  
  
    border-radius: 4px;  
  
}
```

```
/* Pseudo-class: when input is focused */  
  
input:focus {  
  
    border-color: dodgerblue;  
  
    background-color: #f0f8ff;  
  
}
```

```
/* Pseudo-class: hover effect on button */
```

```
button:hover {  
    background-color: dodgerblue;  
    color: white;  
    cursor: pointer;  
}
```

```
/* Pseudo-class: visited link */
```

```
a:visited {  
    color: purple;  
}
```

```
/* Pseudo-class: invalid input */
```

```
input:invalid {  
    border-color: red;  
}
```

```
button {  
    width: 100%;  
    padding: 10px;  
    background-color: lightgray;  
    border: none;  
    border-radius: 4px;  
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<div class="form">
```

```
<h2>Login Form</h2>
```

```
<form>

  <label>Username:</label>

  <input type="text" required placeholder="Enter username"

  <label>Password:</label>

  <input type="password" required placeholder="Enter password">

  <button type="submit">Login</button>

</form>

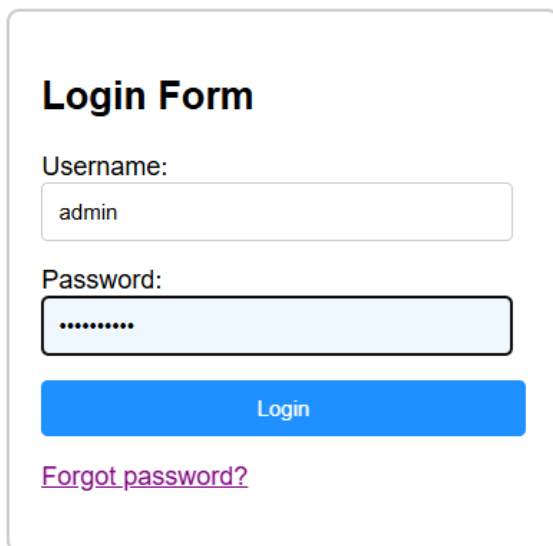
<p><a href="#">Forgot password?</a></p>

</div>

</body>

</html>
```

Output:



Login Form

Username:

Password:

Login

[Forgot password?](#)

4 d) pseudo element selector:

```
<!DOCTYPE html>
```

```
<html>

<head>

  <title>Attribute Selector Demo</title>

  <style>

    input[type="text"] {

      border: 2px solid blue;

    }

    a[href^="https"] {

      font-style: italic;

    }

    input[placeholder*="name"] {

      background-color: #f0f8ff;

    }

  </style>

</head>

<body>

  <h2>Form Example</h2>

  <input type="text" placeholder="Enter your name"><br><br>

  <input type="password" placeholder="Enter password"><br><br>

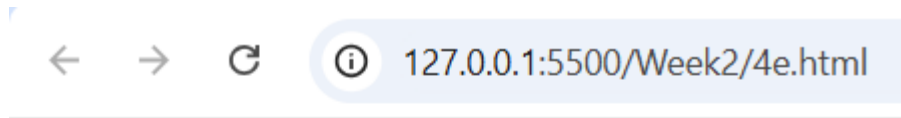
  <a href="https://example.com">Visit Example Site</a><br>
```

`Visit Another Site`

`</body>`

`</html>`

Output:



Form Example

[*Visit Example Site*](#)

[*Visit Another Site*](#)

Viva Questions:

1. What is an element selector in CSS?

Ans) It selects HTML elements by their tag name.

Example: `p { color: blue; }` selects all `<p>` tags.

2. What is an ID selector?

Ans) It selects an element with a specific ID using `#`.

Example: `#header { background: yellow; }`

3. What is a class selector?

Ans) It selects all elements with a specific class using `.`

Example: `.box { border: 1px solid black; }`

4. What is a group selector?

Ans) It groups multiple selectors to apply the same styles.

Example: `h1, h2, h3 { font-weight: bold; }`

5. What is the universal selector?

Ans) It selects all elements on the page using `*`.

Example: `* { margin: 0; padding: 0; }`

6. What is a descendant selector?

Ans) It selects all elements inside a specified element.

Example: `div p { color: green; }` selects all `<p>` inside `<div>`.

7. What is a child selector?

Ans) It selects direct child elements only using `>`.

Example: `ul > li { list-style: none; }`

8. What is an adjacent sibling selector?

Ans) It selects the next sibling immediately after a specified element using `+`.

Example: `h1 + p { color: red; }`

9. What is a general sibling selector?

Ans) It selects all siblings after the specified element using `~`.

Example: `h1 ~ p { font-style: italic; }`

10. What is a pseudo-class in CSS?

Ans) It defines a special state of an element using `::`.

Example: `a:hover { color: orange; }`

11. What does `:first-child` do?

Ans) Selects the first child of a parent.

Example: `li:first-child { color: red; }`

12. What does `:nth-child(n)` do?

Ans) Selects the nth child of a parent.

Example: `li:nth-child(2) { color: blue; }`

13. What is a pseudo-element in CSS?

Ans) It styles specific parts of an element using `::`.

Example: `p::first-line { font-weight: bold; }`

14. What does `::before` do?

Ans) Inserts content before an element.

Example: `h1::before { content: "★ "; }`

15. What is an attribute selector in CSS?

Ans) It selects elements based on the presence or value of an attribute.

Example: `input[type="text"] { border: 1px solid gray; }`

Week-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>
```

```
<head>
```

```
<title>CSS Color Referencing Demo</title>
```

```
<style>
```

```
/* 1. Named Color */
```

```
.named-color {  
    color: red;  
    background-color: lightyellow;  
}  
  
/* 2. Hexadecimal */  
.hex-color {  
    color: #0000FF; /* Blue */  
    background-color: #F0F0F0;  
}  
  
/* 3. RGB */  
.rgb-color {  
    color: rgb(0, 128, 0); /* Green */  
    background-color: rgb(230, 230, 250);  
}  
  
/* 4. RGBA */  
.rgba-color {  
    color: rgba(255, 165, 0, 0.9); /* Orange with slight transparency */  
    background-color: rgba(255, 255, 255, 0.5);  
}  
  
/* 5. HSL */  
.hsl-color {  
    color: hsl(240, 100%, 50%); /* Blue */  
    background-color: hsl(60, 100%, 90%);  
}  
  
/* 6. HSLA */
```

```

.hsla-color {
    color: hsla(300, 100%, 50%, 0.8); /* Purple with transparency */
    background-color: hsla(0, 0%, 100%, 0.5);
}

</style>

</head>

<body>

    <h2>Different Ways to Apply Colors in CSS</h2>

    <p class="named-color">This text uses a named color: <strong>red</strong>.</p>

    <p class="hex-color">This text uses a hex code: <strong>#0000FF</strong>.</p>

    <p class="rgb-color">This text uses RGB format: <strong>rgb(0, 128, 0)</strong>.</p>

    <p class="rgba-color">This text uses RGBA: <strong>rgba(255, 165, 0, 0.9)</strong>.</p>

    <p class="hsl-color">This text uses HSL: <strong>hsl(240, 100%, 50%)</strong>.</p>

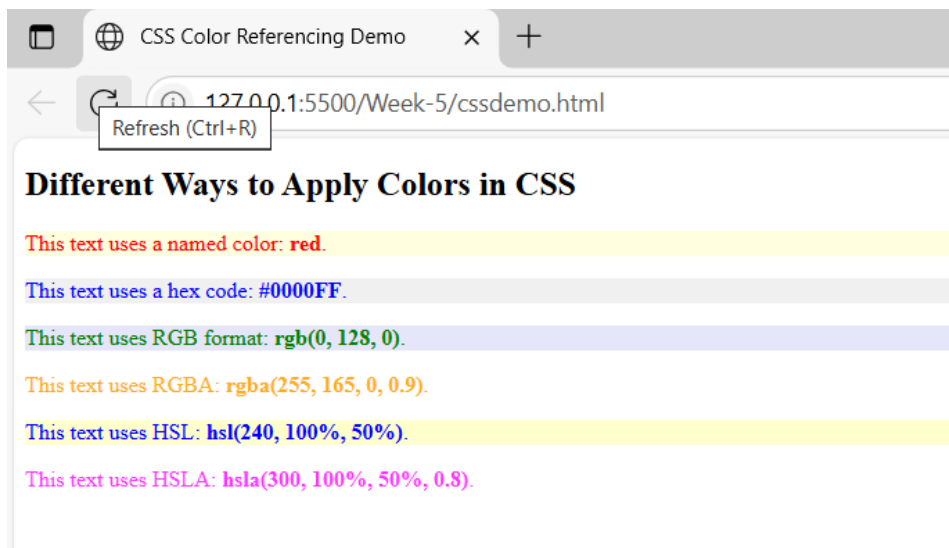
    <p class="hsla-color">This text uses HSLA: <strong>hsla(300, 100%, 50%, 0.8)</strong>.</p>

</body>

</html>

```

Output:



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

Description:

Property	Purpose
<code>background-image</code>	Sets the background image
<code>background-repeat: no-repeat</code>	Prevents the image from repeating
<code>background-attachment: fixed</code>	Keeps the image fixed during scrolling
<code>background-position: center 50%</code>	Places the image halfway down, centered horizontally
<code>background-size: cover</code>	Ensures the image covers the screen
<code>transform: scaleX(-1)</code>	Flips (tilts) the image horizontally

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>Fixed Flipped Background Example</title>
```

```
<style>
```

```
body {
```

```
margin: 0;
```

```
height: 200vh; /* Makes the page scrollable */
```

```
}
```

```
.background {
```

```
position: fixed;
```

```
top: 20%; /* Halfway down the page */
```

```
left: 0;
```

```
width: 100%;
```

```
height: 200px; /* Adjust as needed */
```

```
background-image:url('https://picsum.photos/id/1015/600/400');

background-repeat: no-repeat;

background-size: cover;

background-position: center;

transform: rotate(-15deg); /* Flip horizontally */

z-index: -1; /* Place behind content */

}

.content {

padding: 50px;

font-size: 20px;

}

</style>

</head>

<body>

<div class="background"></div>

<div class="content">

<h1>Scroll Down</h1>

<p>This is an example page with a background image fixed halfway down and flipped horizontally.</p>

<p>Keep scrolling to see the background image stay in place.</p>

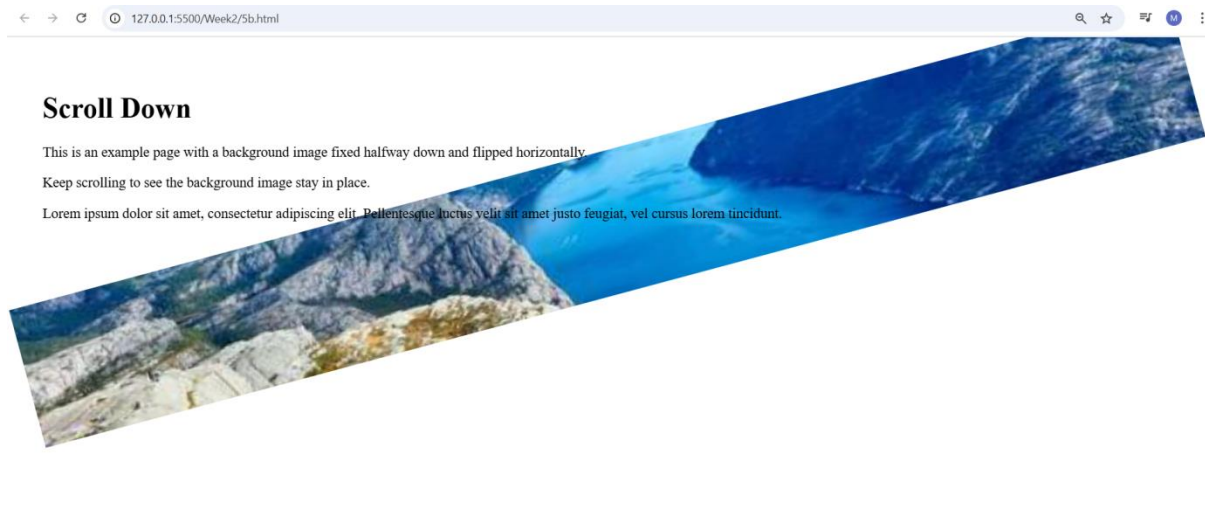
<p>Lorem ipsum dolor sit amet, consectetur adipiscing elit. Pellentesque luctus velit sit amet justo feugiat, vel cursus lorem tincidunt.</p>

</div>

</body>

</html>
```

Output:



5 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

Code:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<title>CSS Font and Text Properties</title>
```

```
<style>
```

```
.font-size {
```

```
font-size: 24px; /* Increases text size */
```

```
}
```

```
.font-weight {
```

```
font-weight: bold; /* Makes text bold */
```

```
}
```

```
.font-style {
```

```
font-style: italic; /* Makes text italic */
```



```
}

.text-decoration {

    text-decoration: underline; /* Underlines the text */

}

.text-transform {

    text-transform: uppercase; /* Converts text to uppercase */

}

.text-align {

    text-align: center; /* Centers the text */

}

</style>

</head>


<body>

<h2>CSS Font and Text Formatting Examples</h2>

<p class="font-size">This text uses <strong>font-size: 24px</strong>.</p>

<p class="font-weight">This text uses <strong>font-weight: bold</strong>.</p>

<p class="font-style">This text uses <strong>font-style: italic</strong>.</p>

<p class="text-decoration">This text uses <strong>text-decoration: underline</strong>.</p>

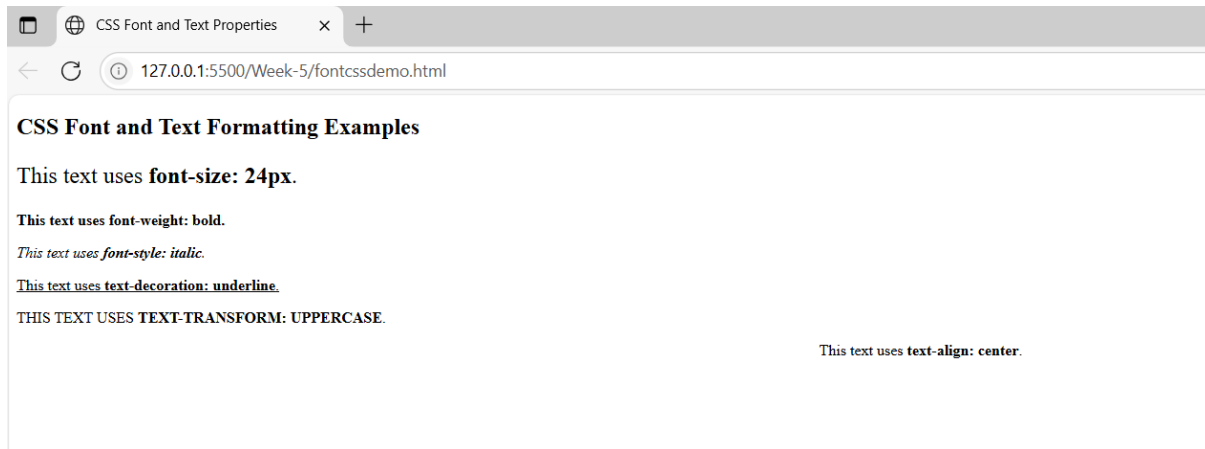
<p class="text-transform">This text uses <strong>text-transform: uppercase</strong>.</p>

<p class="text-align">This text uses <strong>text-align: center</strong>.</p>

</body>

</html>
```

Output:

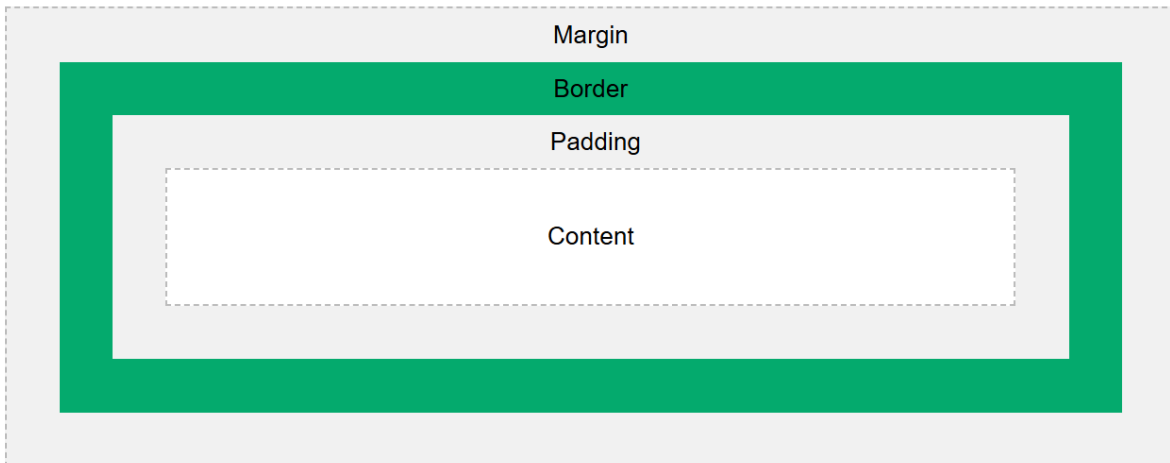


5 d) Write a program, to explain the importance of CSS Box model using

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

Description:

The CSS box model is essentially a box that wraps around every HTML element. It consists of content, padding, borders and margins.



Code:

```
<!DOCTYPE html>

<html>

<head>

<title>CSS Box Model Demo</title>

<style>

.box {

width: 300px;

height: 100px;

background-color: lightyellow; /* Content background */

padding: 20px;                /* Space inside border */

border: 5px solid orange;      /* Border around content and padding */

margin: 30px auto;            /* Space outside the border */

text-align: center;

font-size: 18px;

box-sizing: content-box;      /* Default box model behavior */

}

body {
```

```
background-color: #f2f2f2;

font-family: Arial, sans-serif;
}

</style>

</head>

<body>

<h2 style="text-align:center;">CSS Box Model Explanation</h2>


<div class="box">

  <strong>Content:</strong> This is the main content.<br>

  <strong>Padding:</strong> Adds space inside the box.<br>

  <strong>Border:</strong> Adds a visible edge around it.<br>

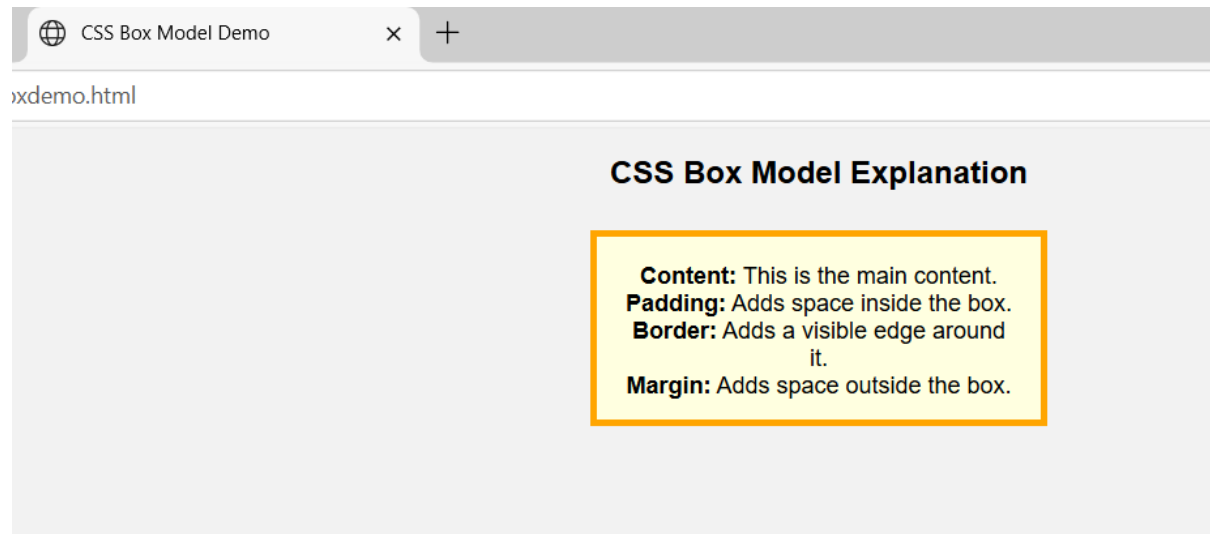
  <strong>Margin:</strong> Adds space outside the box.

</div>

</body>

</html>
```

Output:



Viva Questions:

1) What is the difference between color and background-color in CSS?

Ans) color sets the text (foreground) color.

background-color sets the background color of an element.

2) What are the different ways to define colors in CSS?

Ans)

- Color names (e.g., red, blue)
- Hex codes (e.g., #ff0000)
- RGB values (e.g., rgb(255, 0, 0))
- RGBA (e.g., rgba(255, 0, 0, 0.5))
- HSL (e.g., hsl(0, 100%, 50%))
- HSLA (e.g., hsla(0, 100%, 50%, 0.5))

3) What does the a stand for in rgba or hsla?

Ans) It stands for alpha, which controls the transparency of the color (0 = fully transparent, 1 = fully opaque).

4) What is the Box Model in CSS?

Ans) The CSS Box Model is a layout concept that wraps every HTML element in four parts:

- Content → Actual text or image

- **Padding** → Space between content and border
- **Border** → Line around padding and content
- **Margin** → Space outside the border

5) What is the difference between padding and margin?

Ans)

- **padding:** Space inside the element, between content and border
- **margin:** Space outside the element, between border and neighboring elements

WEEK 6,7,8,9

AIM:Write a program to take three integers → Display the largest or print Equal Numbers.

Description

This program takes three integers from the user and checks:

1. If all three are equal → prints that they are equal.
2. Otherwise → finds the largest number using nested if-else.
The comparison uses \geq so that ties between two largest numbers are also handled correctly.

Answer:

html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Find Largest Number</title>
```

```
</head>
```

```
<body>
```

```
  <script>
```

```
    // Program to find the largest of three numbers or print if they are equal
```

```
// Taking input (For demonstration, values are hardcoded. Use prompt() for real input)

let num1 = parseInt(prompt("Enter the first integer:", 15)); // Assume user entered 15, 9, 22

let num2 = parseInt(prompt("Enter the second integer:", 9));

let num3 = parseInt(prompt("Enter the third integer:", 22));


// Logic to find the largest number or check for equality

if (num1 === num2 && num2 === num3) {

    document.write(`All three numbers are equal: ${num1}`);

} else {

    // Find the largest using nested if-else

    let largest;

    if (num1 >= num2 && num1 >= num3) {

        largest = num1;

    } else if (num2 >= num1 && num2 >= num3) {

        largest = num2;

    } else {

        largest = num3;

    }

    document.write(`The largest number among ${num1}, ${num2}, and ${num3} is: ${largest}`);

}

</script>

</body>

</html>
```

Explanation (for full marks):

- **Input:** The `prompt()` function is used to take input from the user. `parseInt()` converts the input string to an integer.
- **Condition Check:** The first if condition checks if all three numbers are equal using the logical AND (`&&`) operator.
- **Finding Largest:** If they are not all equal, a series of if-else if statements compare the numbers to find the largest one. The condition `>=` is used to handle cases where two numbers might be equal and larger than the third.

Sample Input (via `prompt`)

15, 9, 22

Sample Output

The largest number among 15, 9, and 22 is: 22

AIM: Write a program to display weekdays using switch case.

Description

This program takes a number between **1 and 7** as input.

- Each case in the `switch` corresponds to a weekday.
- If the number is outside 1–7, the `default` case prints an error message.

PROGRAM:

html

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Weekdays using Switch Case</title>
```

```
</head>
```

```
<body>
```

```
  <script>
```



```
// Program to display weekday name based on day number
```

```
// Taking input (For demonstration, value is hardcoded. Use prompt() for real input)
```

```
let dayNumber = parseInt(prompt("Enter a number between 1 and 7:", 3)); // Assume user entered
```

3

```
let weekdayName;
```

```
// Using switch case to assign the weekday name
```

```
switch (dayNumber) {
```

```
  case 1:
```

```
    weekdayName = "Sunday";
```

```
    break;
```

```
  case 2:
```

```
    weekdayName = "Monday";
```

```
    break;
```

```
  case 3:
```

```
    weekdayName = "Tuesday";
```

```
    break;
```

```
  case 4:
```

```
    weekdayName = "Wednesday";
```

```
    break;
```

```
  case 5:
```

```
    weekdayName = "Thursday";
```

```
    break;
```

```
  case 6:
```

```

        weekdayName = "Friday";

        break;

    case 7:

        weekdayName = "Saturday";

        break;

    default:

        weekdayName = "Invalid Number! Please enter a number between 1 and 7.";

    }

    document.write(`Day ${dayNumber} is: ${weekdayName}`);

</script>

</body>

</html>

```

Explanation (for full marks):

- **Switch Statement:** The switch statement evaluates an expression (dayNumber), matching the expression's value to a case clause.
- **Break Keyword:** The break statement is crucial. It terminates the switch block, preventing JavaScript from executing code in the next case (a behavior called "fall-through").
- **Default Case:** The default clause is executed if no matching case is found. It is essential for handling invalid input gracefully.

Sample Input

3

Sample Output

Day 3 is: Tuesday

•

AIM: Write a program to print 1–10 numbers using for, while, do-while.

Description

This program demonstrates the **three looping constructs** in JavaScript:

1. **for loop** → Best when the number of iterations is known.
2. **while loop** → Runs as long as the condition remains true.
3. **do...while loop** → Executes at least once, condition is checked later.

Each loop prints numbers from 1 to 10.

PROGRAM:

```
<!DOCTYPE html>

<html>

<head>

  <title>Loops in JavaScript</title>

</head>

<body>

  <script>

    // Program to print numbers from 1 to 10 using different loops


    document.write("<b>Using for loop:</b><br>");

    // 1. for loop: Ideal when the number of iterations is known beforehand.

    for (let i = 1; i <= 10; i++) {

      document.write(i + " ");

    }

    document.write("<br><br>");


    document.write("<b>Using while loop:</b><br>");

    // 2. while loop: Repeats as long as a specified condition is true.

    let j = 1;
```

```
while (j <= 10) {  
    document.write(j + " ");  
    j++;  
}  
document.write("<br><br>");  
  
document.write("<b>Using do...while loop:</b><br>");  
  
// 3. do...while loop: Executes the code block once, and then repeats as long as the condition is  
true.  
  
let k = 1;  
  
do {  
    document.write(k + " ");  
    k++;  
} while (k <= 10);  
  
</script>  
  
</body>  
  
</html>
```

Explanation (for full marks):

- for **loop**: Consists of three parts: initialization (let i = 1), condition (i <= 10), and final expression (i++). It's the most concise way for this task.
- while **loop**: Only has the condition. The loop variable (j) must be initialized before the loop and incremented inside the loop body.
- do...while **loop**: The condition is checked *after* the code block has executed. This guarantees the code runs **at least once**, even if the condition is initially false.

Sample Output

Using for loop:
1 2 3 4 5 6 7 8 9 10

Using while loop:

```
1 2 3 4 5 6 7 8 9 10
```

Using do...while loop:

```
1 2 3 4 5 6 7 8 9 10
```

Part 6: Applying JavaScript – internal and external, I/O, Type Conversion

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

Aim:

To demonstrate embedding **internal** and **external** JavaScript in a web page.

Description:

JavaScript can be added inside HTML in two ways:

1. **Internal** – using `<script>` inside the HTML file.
2. **External** – using a separate `.js` file and linking it with `<script src="">`.

Program:

```
<!-- index.html -->
<!DOCTYPE html>
<html>
<head>
    <title>Internal and External JavaScript</title>
    <script>
        // Internal JavaScript
        function internalMsg() {
            alert("Hello from Internal JavaScript!");
        }
    </script>
    <script src="external.js"></script>
</head>
<body>
    <h2>Internal and External JS Example</h2>
    <button onclick="internalMsg()">Click Internal</button>
    <button onclick="externalMsg()">Click External</button>
</body>
</html>
// external.js
function externalMsg() {
    alert("Hello from External JavaScript!");
}
```

Sample Output:

- Clicking **Internal** → shows Hello from Internal JavaScript!
 - Clicking **External** → shows Hello from External JavaScript!
-

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

Aim:

To show multiple ways of displaying output in JavaScript.

Description:

JavaScript provides different methods:

- `alert()`
- `document.write()`
- `console.log()`
- `innerHTML`

Program:

```
<!DOCTYPE html>
<html>
<head>
  <title>Output Methods</title>
</head>
<body>
  <h2 id="demo"></h2>

  <script>
    // Alert
    alert("This is Alert Output");

    // Document Write
    document.write("This is Document Write Output<br>");

    // Console Log
    console.log("This is Console Log Output");

    // Inner HTML
    document.getElementById("demo").innerHTML = "This is InnerHTML
Output";
  </script>
</body>
</html>
```

Sample Output:

- **Alert Box:** "This is Alert Output"
- **Web Page:** This is Document Write Output + "This is InnerHTML Output" in <h2>
- **Console:** "This is Console Log Output"

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

Aim:

To demonstrate how to take input in JavaScript.

Description:

JavaScript provides different input methods:

- `prompt()`
- HTML `<input>` fields
- Form submission

Program:

```
<!DOCTYPE html>
<html>
<head>
  <title>Input Methods</title>
</head>
<body>
  <script>
    // Using prompt
    let name = prompt("Enter your name:");
    document.write("Hello " + name + "<br>");
  </script>

  <!-- Using input field -->
  <input type="text" id="txtName" placeholder="Enter Name">
  <button onclick="showName()">Submit</button>
  <p id="output"></p>

  <script>
    function showName() {
      let val = document.getElementById("txtName").value;
      document.getElementById("output").innerHTML = "You entered: " +
val;
    }
  </script>
</body>
</html>
```

Sample Output:

- **Prompt:** "Enter your name" → Displays "Hello John"
 - **Input Field:** After entering "John", output shows "You entered: John"
-

(d) Create a webpage using prompt dialogue box to ask a voter for his name and age. Display in table format whether the voter can vote or not.

Aim:

To collect name and age of a voter using `prompt()` and display result in table.

Description:

If age $\geq 18 \rightarrow$ eligible to vote, else not eligible.

Program:

```
<!DOCTYPE html>
<html>
<head>
    <title>Voter Eligibility</title>
</head>
<body>
    <script>
        let name = prompt("Enter your Name:");
        let age = parseInt(prompt("Enter your Age:"));
        let status = (age >= 18) ? "Eligible to Vote" : "Not Eligible to
Vote";

        document.write("<table border='1' cellpadding='5'>");
        document.write("<tr><th>Name</th><th>Age</th><th>Status</th></tr>");
        document.write("<tr><td>" + name + "</td><td>" + age + "</td><td>" +
status + "</td></tr>");
        document.write("</table>");
    </script>
</body>
</html>
```

Sample Output:

Name	Age	Status
Ramesh	20	Eligible to Vote

Part 9: JavaScript Functions and Events

(a) Design a function to display factorial and Fibonacci series.

Aim:

To implement factorial and Fibonacci using functions in JavaScript.

Description:

- Factorial $\rightarrow n! = n * (n-1) * \dots * 1$
- Fibonacci \rightarrow sequence 0, 1, 1, 2, 3, 5, ... up to n

Program:

```
<!DOCTYPE html>
<html>
<head>
  <title>Factorial and Fibonacci</title>
</head>
<body>
  <script>
    function factorial(n) {
      let fact = 1;
      for (let i = 1; i <= n; i++) fact *= i;
      return fact;
    }

    function fibonacci(n) {
      let a = 0, b = 1, series = [a, b];
      for (let i = 2; i < n; i++) {
        series.push(a + b);
        [a, b] = [b, a + b];
      }
      return series.join(", ");
    }

    document.write("Factorial of 5: " + factorial(5) + "<br>");
    document.write("Fibonacci Series (10 terms): " + fibonacci(10));
  </script>
</body>
</html>
```

Sample Output:

- Factorial of 5 \rightarrow 120
 - Fibonacci (10 terms) \rightarrow 0, 1, 1, 2, 3, 5, 8, 13, 21, 34
-

(b) Design HTML page with textbox and four buttons for Factorial & Fibonacci.

Aim:

To create HTML page with buttons that call functions for factorial and Fibonacci.

Description:

- Input taken from textbox.
- Button click triggers function.

Program:

```
<!DOCTYPE html>
<html>
<head>
  <title>Factorial & Fibonacci</title>
</head>
<body>
  <h3>Enter a Number:</h3>
  <input type="number" id="num">
  <button onclick="showFactorial()">Factorial</button>
  <button onclick="showFibonacci()">Fibonacci</button>
  <p id="result"></p>

  <script>
    function showFactorial() {
      let n = parseInt(document.getElementById("num").value);
      let fact = 1;
      for (let i = 1; i <= n; i++) fact *= i;
      document.getElementById("result").innerHTML = "Factorial: " +
fact;
    }

    function showFibonacci() {
      let n = parseInt(document.getElementById("num").value);
      let a = 0, b = 1, series = [a, b];
      for (let i = 2; i < n; i++) {
        series.push(a + b);
        [a, b] = [b, a + b];
      }
      document.getElementById("result").innerHTML = "Fibonacci: " +
series.join(", ");
    }
  </script>
</body>
</html>
```

Sample Output:

- Input 5 → Button *Factorial* → Factorial: 120
 - Input 6 → Button *Fibonacci* → 0, 1, 1, 2, 3, 5
-

(c) Write a program to validate registration form.

Aim:

To validate form fields in JavaScript (Name, Mobile, Email).

Description:

- Name must start with alphabet and min 6 characters.

- Mobile must be 10 digits.
- Email must follow correct format.

Program:

```
<!DOCTYPE html>
<html>
<head>
  <title>Form Validation</title>
</head>
<body>
  <h2>Registration Form</h2>
  <form onsubmit="return validateForm()">
    Name: <input type="text" id="name"><br><br>
    Mobile: <input type="text" id="mobile"><br><br>
    Email: <input type="text" id="email"><br><br>
    <input type="submit" value="Register">
  </form>
  <p id="msg"></p>

  <script>
    function validateForm() {
      let name = document.getElementById("name").value;
      let mobile = document.getElementById("mobile").value;
      let email = document.getElementById("email").value;

      let namePattern = /^[A-Za-z][A-Za-z0-9]{5,}$/;
      let mobilePattern = /^[0-9]{10}$/;
      let emailPattern = /^[^ ]+@[^ ]+\.[a-z]{2,3}$/;

      if (!name.match(namePattern)) {
        document.getElementById("msg").innerHTML = "Invalid Name";
        return false;
      }
      if (!mobile.match(mobilePattern)) {
        document.getElementById("msg").innerHTML = "Invalid Mobile
Number";
        return false;
      }
      if (!email.match(emailPattern)) {
        document.getElementById("msg").innerHTML = "Invalid Email";
        return false;
      }

      document.getElementById("msg").innerHTML = "Registration
Successful!";
      return false;
    }
  </script>
</body>
</html>
```

Sample Output:

- Enter invalid details → "Invalid Name" / "Invalid Mobile" / "Invalid Email"

- Enter valid details → "Registration Successful!"

