HTML ASSIGNMENT

1.Explain the Table and its properties.

In HTML, the <table> element is used to create tables, which allow you to organize and display data in rows and columns. Tables are composed of several elements that define their structure and appearance. Here are the key properties and elements associated with HTML tables:

1. <table> Element:

The <table> element is the container for the entire table.

<table>

<!-- Table content (rows, headers, data cells, etc.) goes here -->

</table>

1. <tr> Element:

The <tr> element represents a table row. It contains one or more <td> (table data) or <th> (table header) elements.

<tr>

<!-- Table data cells or header cells go here -->

</tr>

1. <th> Element:

The <th> element represents a table header cell. It is used to define header cells for a row or a column.

<tr>

<th>Header 1</th>

<th>Header 2</th>

</tr>

1. <td> Element:

The <td> element represents a table data cell. It contains the actual data for a specific row and column.

<tr>

<td>Data 1</td>

<td>Data 2</td>

</tr>

1. Row and Column Spanning:

The rowspan and colspan attributes can be used to make a cell span multiple rows or columns.

<tr>

<td rowspan="2">Spanned Cell</td>

<td>Data 1</td>

</tr>

<tr>

<td>Data 2</td>

</tr>

1. Table Head, Body, and Footer:

For better organization, tables can be divided into sections using the <thead>, <tbody>, and <tfoot> elements.

<table>

<thead>

<!-- Table header content goes here -->

</thead>

<tbody>

<!-- Table body content goes here -->

</tbody>

<tfoot>

<!-- Table footer content goes here -->

</tfoot>

</table>

1. Border Attribute:

The border attribute can be used to define the border width of the table. However, it's more common to use CSS for styling.

<table border="1">

<!-- Table content goes here -->

</table>

1. Styling with CSS:

CSS is commonly used to style tables, providing control over borders, spacing, colors, and other visual aspects.

table {

border-collapse: collapse;

width: 100%;

}

th, td {

border: 1px solid #ddd;

padding: 8px;

text-align: left;

}

1. Caption Element:

The <caption> element can be used to add a title or caption to a table.

<table>

<caption>Table Caption</caption>

<!-- Table content goes here -->

</table>

2. Explain all Media elements and how they work.

HTML provides several media elements that allow you to embed and control various types of multimedia content, such as audio and video, directly within web pages. Here are the main media elements and how they work:

1. <audio> Element:

The <audio> element is used to embed audio content in a web page. It supports various audio formats, and you can specify multiple source elements to provide fallback options for different browsers.

<audio controls>

<source src="audio.mp3" type="audio/mp3">

<source src="audio.ogg" type="audio/ogg">

Your browser does not support the audio tag.

</audio>

Attributes:

controls: Adds audio controls (play, pause, volume) to the audio player.

src: Specifies the source URL of the audio file.

type: Defines the MIME type of the audio file.

1. <video> Element:

The <video> element is used to embed video content in a web page. Similar to <audio>, it supports multiple source elements for different formats.

<video width="640" height="360" controls>

<source src="video.mp4" type="video/mp4">

<source src="video.webm" type="video/webm">

Your browser does not support the video tag.

</video>

Attributes:

controls: Adds video controls (play, pause, volume, etc.) to the video player.

width and height: Define the dimensions of the video player.

src: Specifies the source URL of the video file.

type: Defines the MIME type of the video file.

1. <img> Element (for Images):

While not specifically a media element, the <img> element is used to embed images in web pages.

<img src="image.jpg" alt="Description of the image">

Attributes:

src: Specifies the source URL of the image.

alt: Provides alternative text for the image (useful for accessibility).

1. <source> Element:

The <source> element is used within <audio> and <video> elements to specify alternative media resources. This allows browsers to choose the most suitable format based on compatibility.

<video width="640" height="360" controls>

<source src="video.mp4" type="video/mp4">

<source src="video.webm" type="video/webm">

Your browser does not support the video tag.

</video>

1. What is Get and Post in Form?

In HTML forms, the GET and POST methods are two different HTTP methods that can be used to submit form data to a server. These methods define how the form data is sent from the browser to the server, and they are specified in the method attribute of the <form> element.

1. GET Method:

The GET method is used to request data from a specified resource. When a form is submitted using the GET method, the form data is appended to the URL in the form of query parameters.

<form action="/example" method="GET">

<!-- Form fields go here -->

<input type="text" name="username" />

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

</form>

In this example, if a user enters "john" in the username field and submits the form, the URL might look like: /example?username=john. The form data is visible in the URL, making it suitable for relatively small amounts of data and situations where data visibility is not a concern.

Characteristics:

Appends data to the URL.

Limited data capacity (as URLs have length limitations).

Data is visible in the URL (may pose security concerns).

Suitable for simple queries and non-sensitive data.

1. POST Method:

The POST method is used to submit data to be processed to a specified resource. When a form is submitted using the POST method, the form data is sent in the body of the HTTP request, not in the URL.

<form action="/example" method="POST">

<!-- Form fields go here -->

<input type="text" name="username" />

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

</form>

In this example, if a user enters "john" in the username field and submits the form, the form data is sent in the body of the HTTP request, and it is not visible in the URL.

Characteristics:

Sends data in the request body.

No data limitations imposed by the URL.

Data is not visible in the URL.

Suitable for sensitive information and larger amounts of data.

1. List all the input types we have in forms.

Text Input (text):

<input type="text" name="username" />

Password Input (password):

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

Email Input (email):

<input type="email" name="user\_email" />

Number Input (number):

<input type="number" name="quantity" />

Checkbox (checkbox):

<input type="checkbox" name="subscribe" />

Radio Button (radio):

<input type="radio" name="gender" value="male" />

<input type="radio" name="gender" value="female" />

Date Input (date):

<input type="date" name="birthdate" />

Time Input (time):

<input type="time" name="appointment\_time" />

URL Input (url):

<input type="url" name="website" />

File Input (file):

<input type="file" name="file\_upload" />

Hidden Input (hidden):

<input type="hidden" name="secret\_id" value="12345" />

Submit Button (submit):

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

Reset Button (reset):

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

Button (button):

<input type="button" value="Click me" />

Color Input (color):

<input type="color" name="favorite\_color" />

Range Input (range):

<input type="range" name="volume" min="0" max="100" />

Search Input (search):

<input type="search" name="search\_query" />

Tel Input (tel):

<input type="tel" name="phone\_number" />

Textarea (textarea):

<textarea name="message"></textarea>