*WEB DESIGNING*

# ***HTML:-***

HTML stands for **HyperText Markup Language**. It is the standard markup language used to create and structure content on the World Wide Web.

**Hypertext** refers to text that has links in it. These links can take you to other pieces of information, like jumping from one web page to another by clicking on highlighted words or phrases.

**Markup** means adding special instructions or symbols to a document or text to show how it should be formatted or displayed

Markup - process or result of [correcting](https://www.google.com/search?sca_esv=557369124&q=correcting&si=ACFMAn9-5A9OMKPWcg180I9o9Mndnz_o3Rg5MmKsHGN6_lr-GqB_fvQy6W-3UCn2Gjre15jMkrg-lDH-ywgqUsnmWos3b5tGAA%3D%3D&expnd=1) text in preparation for printing.

In simpler terms, **HTML provides a way to describe the different parts of a webpage**, such as headings, paragraphs, lists, images, and links, so that web browsers can interpret and display them correctly. HTML documents are usually text files with a .html extension.

HTML works in conjunction with other technologies like Cascading Style Sheets (CSS) and JavaScript to create interactive and visually appealing web pages. CSS is used to style the appearance of HTML elements, while JavaScript adds interactivity and dynamic behavior to web pages.

Basically there are two types of pages:

* Static web pages
* Dynamic web pages

## **Static web pages:-**

Static content refers to web pages that remain the same every time they are loaded. In other words, the content does not change based on user interactions or other factors.

## **Dynamic web pages:-**

Dynamic content refers to web pages that can change their content based on various factors such as user input, user preferences, database queries, and other real-time interactions. Dynamic web pages are generated on-the-fly by server-side scripts (often using languages like PHP, Python, Ruby, etc.) or client-side scripting (JavaScript) in response to user actions.

# ***Doctype declaration for different version of HTML:***

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">

<!DOCTYPE html>:-

## It represents the HTML5 version.

Certainly, here are the key features of HTML5 in brief:

**Features of HTML5:-**

1. **Semantic Elements**: Introduces semantic tags like <header>, <nav>, and <footer> for better content structure.

2. **Audio and Video:** Built-in support for embedding audio and video without plugins.

3. **Canvas:** Allows dynamic graphics and animations with JavaScript.

4. **Improved Forms:** New input types and attributes for easier and enhanced form creation.

5. **Web Storage:** Local storage for storing data on user devices.

6. **Geolocation:** Access user's location with permission.

7. **Responsive Design:** Easier creation of websites that adapt to different devices.

8. **Web Workers:** Enables concurrent script execution for improved performance.

9. **Offline Web Apps:** Ability to create apps that work offline.

10. **Drag and Drop:** Native support for dragging and dropping elements.

11. **SVG Support:** Better integration of scalable vector graphics.

12. **New Doctype:** Simplified and standard <!DOCTYPE html> declaration.

13. **Cross-browser Compatibility:** Improved support across different web browsers.

14. **Cleaner Code:** Encourages cleaner, more organized coding practices.

# ***Tags:-***

In HTML, **"tags" are like labels that you put around content to tell web browsers how to display it on a webpage**. They're like instructions that say, "This part is a heading," or "This is a picture," or "This is a link."

For example, think of making a sandwich. If you label one part as "bread," another as "lettuce," and another as "tomato," it helps you know how to put everything together.

## ***Basically there are two types of tags:-***

* Container tags
* Empty tags

### ***Container tags:-***

Container tags consist of an **opening tag and a closing tag that enclose content between them**.Remember that container tags always have an opening tag and a closing tag, like <tag> and </tag>. The content you want to affect with the tag's formatting or behavior goes between these two tags.

1. Paragraph Tag (<p>)

<p>This is a paragraph of text.</p>

2. Heading Tags (<h1> to <h6>)

<h1>This is a main heading</h1>

<h2>This is a subheading</h2>

3. List Tags (<ul>, <ol>, <li>):

<ul>

<li>Item 1</li>

<li>Item 2</li>

</ul>

<ol>

<li>First</li>

<li>Second</li>

</ol>

4. Table Tags (<table>, <tr>, <th>, <td>):

<table>

<tr>

<th>Header 1</th>

<th>Header 2</th>

</tr>

<tr>

<td>Data 1</td>

<td>Data 2</td>

</tr>

</table>

5. Form Tags (<form>, <input>, <button>, <select>, <textarea>):

<form action="/submit" method="post">

<label for="username">Username:</label>

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

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

</form>

6. Semantic Tags (<header>, <article>, <footer>):

<header>

<h1>Welcome to Our Website</h1>

</header>

<article>

<p>This is an interesting article.</p>

</article>

<footer>

<p>Contact us at example@example.com</p>

</footer>

7. Division Tag (<div>):

<div class="container">

<p>This is some content inside a div.</p>

</div>

8. Anchor Tag (<a>):

<p>Visit our <a href="https://www.example.com">website</a> for more information.</p>

9. Formatting Tags (<strong>, <em>, <b>, <i>):

<p>This is <strong>strong</strong> and <em>emphasized</em> text.</p>

### ***Empty tags:-***

"Empty tags" typically **refer to self-closing tags in HTML.** These are tags that don't require a separate closing tag because they **don't contain any content,** they have a slash before the closing angle bracket to indicate that they are self-closing. They are also sometimes referred to as "void elements" or "self-contained elements."

Here are some common examples of self-closing tags:

1. **<br>:** Represents a line break, used to create a new line within text.

<p>This is some text.<br>Here is a new line.</p>

2. **<img>:** Embeds images within a web page.

<img src="image.jpg" alt="Image">

3. **<input>:** Used for various types of form input fields.

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

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

4. **<hr>:** Creates a horizontal rule, often used to separate content sections.

<p>Content above<hr>Content below</p>

5. **<meta>:** Provides metadata about the HTML document.

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

6. **<link>:** Links external resources like stylesheets or icon files.

<link rel="stylesheet" href="styles.css">

<link rel="icon" type="image/png" href="favicon.png">

These tags are self-contained and don't have any content between opening and closing tags.

# ***ELEMENTS:-***

**Elements are the building blocks that make up the structure and content of a web page.** An element consists of an opening tag, content, and a closing tag. The opening tag specifies the beginning of the element, the closing tag specifies the end, and the content is the information or media that goes between the tags.



## ***Content:-***

Content refers to the information, text, multimedia, and other elements that make up the actual visible and interactive part of a web page

# ***HTML Tags:-***

The <html> tag is the root element of an HTML document. It encloses all other HTML elements within it and serves as the container for the entire web page's content.

<!DOCTYPE html>

<html>

<head>

<!-- Meta tags, title, linked resources (CSS, JavaScript), etc. -->

</head>

<body>

<!-- Content of the web page (text, images, links, etc.) -->

</body>

</html>

Explanation of the parts:

<!DOCTYPE html>: This declaration specifies the document type and version, indicating that the document is written in HTML5.

<html>: The root element that contains the entire HTML document.

<head>: This section contains meta information, title, linked resources (CSS, JavaScript), and other document-level settings. It's not directly visible on the web page.

<body>: This section contains the visible content of the web page, such as text, images, links, and other HTML elements.

Example of an HTML document using the <html> tag:-

<!DOCTYPE html>

<html>

<head>

<title>My First HTML Page</title>

</head>

<body>

<h1>Hello, World!</h1>

<p>This is my first HTML page.</p>

</body>

</html>

# ***<Head> tags:-***

**The <head> tag is an HTML element used to contain metadata and other non-visible information about a web page.** It's placed within the <html> element, before the <body> element, and it's not directly displayed on the web page itself. The content within the <head> tag provides information that helps browsers, search engines, and other web services understand and process the webpage.

Example of the <head> section of an HTML:-

<!DOCTYPE html>

<html>

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<meta name="description" content="This is an example of the head section in HTML.">

<meta name="keywords" content="HTML, head tag, example">

<meta name="author" content="Your Name">

<meta http-equiv="refresh" content="5"> <!-- Refresh every 5 seconds -->

<link rel="stylesheet" href="styles.css"> <!-- Link to an external CSS stylesheet -->

<link rel="icon" href="favicon.ico" type="image/x-icon"> <!-- Favicon for the website -->

<title>Head Tag Example</title>

<style>

/\* Internal CSS style \*/

body {

font-family: Arial, sans-serif;

background-color: #f0f0f0;

}

</style>

<script src="scripts.js" defer></script> <!-- Link to an external JavaScript file -->

</head>

<body>

<h1>Hello, Head Tags!</h1>

<p>This is an example of the head section in HTML.</p>

</body>

</html>

In this example:-

- <meta charset="UTF-8"> specifies the character encoding for the document.

- <meta name="viewport" content="width=device-width, initial-scale=1.0"> sets up a responsive viewport.

- <meta name="description" content="..."> provides a brief description of the web page's content for search engines.

- <meta name="keywords" content="..."> lists keywords relevant to the page's content for search engines.

- <meta name="author" content="..."> indicates the author of the page.

- <link rel="stylesheet" href="styles.css"> links an external CSS stylesheet.

- <link rel="icon" href="favicon.ico" type="image/x-icon"> specifies the favicon for the website.

- Internal CSS styling is provided using the <style> tag.

- <script src="scripts.js" defer></script> links an external JavaScript file with the defer attribute.

Remember, the elements you include in the <head> section will depend on your project's specific requirements and needs.

# 

# ***Attributes:-***

In HTML, attributes are additional pieces of information that can be added to HTML elements to provide extra context, behavior, or styling.

**attributeName="attributeValue".**

<img src="image.jpg" alt="An example image">

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

<p id="intro">This is an introductory paragraph.</p>

<p style="color:blue; font-size:14px;">This is a styled paragraph.</p>

# ***Comments:-***

Comments are ignored by web browsers and do not affect the display or functionality of the web page. They are only visible in the source code.

HTML comments are enclosed within **<!-- and -->** tags. Anything placed between these tags is treated as a comment.

**Syntax:-**

<!-- This is a comment. It won't be displayed in the browser. -->

# ***Format tags:-***

In HTML, formatting tags are used to control the presentation and styling of text content within a web page. These tags allow you to apply various visual styles such as bold, italic, underlined, headings, and more. Here are some of the most commonly used formatting tags in HTML:

1. **Bold Text (<strong> or <b>):**

Both <strong> and <b> tags are used to make text bold. However, <strong> also carries semantic importance, indicating that the enclosed text is of strong importance or relevance.

<p>This is <strong>important</strong> information.</p>

<p>This is <b>also bold</b> but doesn't imply strong importance.</p>

2. **Italic Text (<em> or <i>):**

Similar to bold text, both <em> and <i> tags can be used to italicize text. However, <em> also adds semantic meaning, indicating emphasis.

<p>This is <em>emphasized</em> text.</p>

<p>This is <i>also italic</i> but doesn't imply emphasis.</p>

3. **Underlined Text (<u>):**

The <u> tag is used to underline text.

<p>This is <u>underlined</u> text.</p>

4. **Strikethrough Text (<s> or <del>):**

Both <s> and <del> tags are used to create strikethrough text, indicating that the text has been deleted or is no longer valid.

<p>This text is <s>strikethrough</s>.</p>

<p>This text is <del>also strikethrough</del>.</p>

5. **Superscript (<sup>) and Subscript (<sub>):**

The <sup> tag is used for superscript text (raised above the baseline), and the <sub> tag is used for subscript text (lowered below the baseline).

<p>E=mc<sup>2</sup></p>

<p>H<sub>2</sub>O</p>

6. **Headings (<h1> to <h6>):**

Headings are used to structure content hierarchically, with <h1> being the highest level and <h6> the lowest.

<h1>This is a heading level 1</h1>

<h2>This is a heading level 2</h2>

<!-- ... -->

<h6>This is a heading level 6</h6>

Remember that while these tags provide visual formatting, it's generally recommended to use CSS (Cascading Style Sheets) for more precise and consistent styling of your web page's content. CSS provides greater flexibility and control over the appearance of your content while keeping the HTML focused on its structure and semantics.

<!DOCTYPE html>

<html>

<head>

<title>Formatting Elements Example</title>

</head>

<body>

<h1>Formatting Elements Example</h1>

<h2>Text Formatting</h2>

<p>This is <b>bold</b> text.</p>

<p>This is <strong>strong</strong> text.</p>

<p>This is <i>italic</i> text.</p>

<p>This is <em>emphasized</em> text.</p>

<p>This is <u>underlined</u> text.</p>

<p>This is <mark>highlighted</mark> text.</p>

<p>This is <del>deleted</del> text.</p>

<p>This is <ins>inserted</ins> text.</p>

<p>This is <sub>subscript</sub> text.</p>

<p>This is <sup>superscript</sup> text.</p>

<h2>Abbreviations</h2>

<p><abbr title="HyperText Markup Language">HTML</abbr> is used to structure web content.</p>

<h2>Superscripts and Subscripts</h2>

<p>H<sub>2</sub>O is water, and E=mc<sup>2</sup> is a famous equation.</p>

<h2>Small Text</h2>

<p><small>This is smaller text.</small></p>

<h2>Strikethrough</h2>

<p><s>This text is strikethrough.</s></p>

</body>

</html>

# ***<br> and <hr> Tags:-***

**<br> tag** is used to insert a line break, which causes the text or content that follows it to start on a new line within the same block element. The <br> tag does not have a closing tag, as it's a self-closing tag.

This is some text.<br>

This text will appear on a new line.

**<hr> tag** is used to create a thematic break or horizontal rule,which is a visual element that separates content into distinct sectionsIt's typically displayed as a horizontal line across the width of its containing element.It is a self-closing tag and doesn't require a closing tag.

<p>This is the first paragraph.</p>

<hr>

<p>This is the second paragraph.</p>

# ***Align Attribute:-***

**align** attribute is used to control the alignment of various elements, including headings, paragraphs, and images.

<!DOCTYPE html>

<html>

<head>

<title>Alignment Example</title>

</head>

<body>

<h1 align="center">Center Aligned Heading</h1>

<p align="left">This paragraph is left aligned.</p>

<p align="right">This paragraph is right aligned.</p>

<div align="center">

<p>This paragraph is centered within a div.</p>

</div>

<img src="example.jpg" alt="Example Image" align="left">

<p>This paragraph wraps around the left-aligned image.</p>

<img src="example.jpg" alt="Example Image" align="right">

<p>This paragraph wraps around the right-aligned image.</p>

</body>

</html>

# 

# ***<blockquote> and <q>:-***

**<blockquote>** element is used to indicate a block of quoted text from another source. It is commonly used to highlight and visually separate quoted content within a document. Additionally, **<cite>** element is to indicate the source in a structured way.

The **<q>** element, on the other hand, is used to mark inline quotations or short pieces of text that are part of the surrounding content.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<h1>Quoting and Blockquoting Example</h1>

<blockquote cite ="https://www.sciencenews.org/article/frogs-glow-biofluorescence-language">

<h1>Many frogs glow in blue light, and it may be a secret, eerie language</h1>

<p>A survey of hundreds of frogs in South America shows that far more frogs are

biofluorescent than previously thought, researchers report in a preprint posted

July 28 at bioRxiv.org. <q>ghostly colors may have a role in the frogs</q> communication

with members of the same species, the scientists say.</p>

</blockquote>

<p>This is some regular text outside the blockquote.</p>

</body>

</html>

# ***<address> Tag:-***

**<address>** element in HTML is used to provide contact information for the author or owner of a document. It can also be used to enclose contact information within other elements, such as within a **<footer> or <article>** element. The content inside the **<address>** element is typically **styled in italics** by default in most browsers.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<h1>Contact Information</h1>

<address>

<p>Written by: John Doe</p>

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

<p>Website: <a href="http://www.johndoe.com">www.johndoe.com</a></p>

</address>

<p>For inquiries, please reach out to the author using the provided contact information.</p>

</body>

</html>

# 

# ***<pre> Tag:-***

**<pre>** element in HTML is used to define preformatted text. Preformatted text is text that is **displayed exactly as it's written in the HTML code,** including whitespace and line breaks. This is useful for displaying code, ASCII art, or any text that should maintain its formatting and spacing.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<h1>Preformatted Text Example</h1>

<pre>

This is a line of preformatted text.

It maintains spaces and line breaks exactly as written.

It's often used to display code:

def Hello():

print ("Hello, world!")

</pre>

<p>This is regular text outside the preformatted block.</p>

</body>

</html>

# ***List Tags:-***

HTML provides several elements for creating lists, both ordered and unordered. Lists are used to organize and present content in a structured manner.

The main list elements are:

* <ul> (unordered list)
* <ol> (ordered list)
* <li> (list item)

Here's how these list elements are used:

## 1. **Unordered List (<ul>):** This element is used to create a bulleted or unordered list.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Unorder list</title>

</head>

<body>

<h1>Favorite Fruits</h1>

<ul>

<li>Apples</li>

<li>Bananas</li>

<li>Oranges</li>

<li>Grapes</li>

</ul>

<p>These are some of my favorite fruits.</p>

</body>

</html>

### **Unordered Nested List:-**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Nested Unordered List</title>

</head>

<body>

<!DOCTYPE html>

<html>

<head>

<title>Nested Unordered List Example</title>

</head>

<body>

<h1>Shopping List</h1>

<ul>

<li>Fruits

<ul>

<li>Apples</li>

<li>Bananas</li>

<li>Oranges</li>

</ul>

</li>

<li>Vegetables

<ul>

<li>Carrots</li>

<li>Spinach</li>

<li>Tomatoes</li>

</ul>

</li>

<li>Dairy

<ul>

<li>Milk</li>

<li>Yogurt</li>

<li>Cheese</li>

</ul>

</li>

</ul>

<p>Remember to pick up these items from the grocery store.</p>

</body>

</html>

## 2.**Ordered List (<ol>):** This element is used to create a numbered or ordered list.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<h1>Steps to Make a Sandwich</h1>

<ol>

<li>Get two slices of bread.</li>

<li>Spread your favorite condiments on one or both slices.</li>

<li>Add your choice of fillings, such as lettuce, tomatoes, and cheese.</li>

<li>Place the slices of bread together to form the sandwich.</li>

<li>Cut the sandwich in half if desired.</li>

</ol>

<p>Enjoy your delicious sandwich!</p>

</body>

</html>

### **Types of order list:-**

An ordered list <ol> can have different types of numbering styles. The default numbering style is decimal (1, 2, 3, ...), but you can change it to different types using the type attribute.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<h1>Ordered List Types Example</h1>

<h2>Decimal (Default)</h2>

<ol>

<li>First item</li>

<li>Second item</li>

<li>Third item</li>

</ol>

<h2>Lowercase Roman Numerals</h2>

<ol type="i">

<li>First item</li>

<li>Second item</li>

<li>Third item</li>

</ol>

<h2>Uppercase Roman Numerals</h2>

<ol type="I">

<li>First item</li>

<li>Second item</li>

<li>Third item</li>

</ol>

<h2>Lowercase Letters</h2>

<ol type="a">

<li>First item</li>

<li>Second item</li>

<li>Third item</li>

</ol>

<h2>Uppercase Letters</h2>

<ol type="A">

<li>First item</li>

<li>Second item</li>

<li>Third item</li>

</ol>

<h2>Custom Start Value</h2>

<ol start="10">

<li>Tenth item</li>

<li>Eleventh item</li>

<li>Twelfth item</li>

</ol>

</body>

</html>

### **Ordered Nested List:-**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Nested Ordered List Example</title>

</head>

<body>

<h1>Nested Ordered List Example</h1>

<ol>

<li>First item</li>

<li>Second item

<ol type="i">

<li>Nested item</li>

<li>Nested item</li>

<li>Nested item</li>

</ol>

</li>

<li>Third item

<ol type="A">

<li>Nested item</li>

<li>Nested item</li>

<li>Nested item</li>

</ol>

</li>

<li>Fourth item

<ol type="a">

<li>Nested item</li>

<li>Nested item</li>

<li>Nested item</li>

</ol>

</li>

</ol>

<p>This is a nested ordered list example.</p>

</body>

</html>

3. **List Item (<li>):** This element is used within <ul> or <ol> elements to define individual list items.

# ***Definition list:-***

**<dl>** element creates a definition list. Within the definition list, **each term is defined using a <dt>** element, and its corresponding **definition is provided using a <dd>** element.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Definition List</title>

</head>

<body>

<h1>Glossary</h1>

<dl>

<dt>HTML</dt>

<dd>HyperText Markup Language - the standard language for creating web pages.</dd>

<dt>CSS</dt>

<dd>Cascading Style Sheets - used for describing the look and formatting of a document written in HTML.</dd>

<dt>JavaScript</dt>

<dd>A programming language that enables interactive web pages.</dd>

</dl>

<p>This is a basic glossary using definition lists.</p>

</body>

</html>

# ***<table>:-***

**<table>** tag is used to create tables that organize and display data in rows and columns. <table> tag is typically **used in conjunction with other related tags** to define the structure of the table, such as **<tr>** for table rows, **<th>** for table headers, and **<td>** for table data cells.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<table border="1">

<tr>

<th>Header 1</th>

<th>Header 2</th>

<th>Header 3</th>

</tr>

<tr>

<td>Data 1,1</td>

<td>Data 1,2</td>

<td>Data 1,3</td>

</tr>

<tr>

<td>Data 2,1</td>

<td>Data 2,2</td>

<td>Data 2,3</td>

</tr>

</table>

</body>

</html>

## **Align table and table data:-**

In the table by default the heading is at the center of the cell and data is on the left side of the cell.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<table border="1" align="center">

<tr>

<th align="center">Header 1</th>

<th align="center">Header 2</th>

<th align="center">Header 3</th>

</tr>

<tr>

<td align="center">Data 1,1</td>

<td align="right">Data 1,2</td>

<td align="left">Data 1,3</td>

</tr>

<tr>

<td align="center">Data 2,1</td>

<td align="right">Data 2,2</td>

<td align="left">Data 2,3</td>

</tr>

</table>

</body>

</html>

## **Set width of table and column:-**

We can specify the width in pixels (px) or percentages (%), but it's better to use **pixel with image** and **percentage with text**.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<table border="1" align="center" width="50%">

<tr>

<th width="50%">Header 1</th>

<th>Header 2</th>

<th>Header 3</th>

</tr>

<tr>

<td>Data 1,1</td>

<td>Data 1,2</td>

<td>Data 1,3</td>

</tr>

<tr>

<td>Data 2,1</td>

<td>Data 2,2</td>

<td>Data 2,3</td>

</tr>

</table>

</body>

</html>

## **Set color of table, row and cell:-**

We can set color in the table using **bgcolor** attribute with specific value.

Example:- **bgcolor=”red”**

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<table border="1" >

<tr>

<th>Header 1</th>

<th>Header 2</th>

<th>Header 3</th>

</tr>

<tr>

<td>Data 1,1</td>

<td>Data 1,2</td>

<td>Data 1,3</td>

</tr>

<tr>

<td>Data 2,1</td>

<td>Data 2,2</td>

<td>Data 2,3</td>

</tr>

</table>

</body>

</html>

## **Cell padding and Cell spacing:-**

Cell padding in HTML allows you to control the space between the content of a cell (table data or table header) and the cell's borders. You can set cell padding using the cellpadding attribute directly within the <table>.

Cell spacing in HTML refers to the space between cells in a table. Similar to cell padding, you can control cell spacing using the cellspacing attribute directly within the <table>

**Cell padding and Cell spacing takes values in pixels.**

## **Row spanning:-**

Row spanning in HTML tables **allows a single cell to span multiple rows vertically**, effectively merging cells from different rows into a single cell. This can be achieved using the rowspan attribute on a <td> (table data) or <th> (table header) element.

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta http-equiv="X-UA-Compatible" content="IE=edge">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body>

<table border="1" >

<tr>

<th>Header 1</th>

<th>Header 2</th>

<th>Header 3</th>

</tr>

<tr>

<td>Data 1,1</td>

<td>Data 1,2</td>

<td>Data 1,3</td>

</tr>

<tr>

<td>Data 2,1</td>

<td>Data 2,2</td>

<td>Data 2,3</td>

</tr>

</table>

<br><br><br>

<table border="1" >

<tr>

<th rowspan="3">Header 1

Header 2

Header 3</th>

</tr>

<tr>

<td>Data 1,1</td>

<td>Data 1,2</td>

<td>Data 1,3</td>

</tr>

<tr>

<td>Data 2,1</td>

<td>Data 2,2</td>

<td>Data 2,3</td>

</tr>

</table>

<br><br><br>

<table border="1" >

<tr>

<th>Header 1</th>

<th>Header 2</th>

<th>Header 3</th>

</tr>

<tr>

<td rowspan="2">Data 1,1</td>

<td>Data 1,2</td>

<td>Data 1,3</td>

</tr>

<tr>

<td>Data 2,2</td>

<td>Data 2,3</td>

</tr>

</table>

</body>

</html>

## **Column spanning:-**

Column spanning in HTML tables **allows a single cell to span multiple columns horizontally**, effectively merging cells from different columns into a single cell. This can be achieved using the colspan attribute on a <td> (table data) or <th> (table header) element.

# ***Anchor Tags:-***

In HTML, the <a> tag, commonly referred to as the "anchor" tag, is used to create hyperlinks, allowing users to navigate to other web pages or resources.

# **<a href="URL"></a>:-**

## **1.Absolute link:-**

<a href="<https://www.google.com/>">google</a>

## **2.Relative link:-**

<a href="/pre tag.html">pretag</a> <!-- Links to a relative path on the same website →

## **3.Email link:-**

<a href="mailto:info@example.com">Email Us</a>

## **4. Id attribute:-**

<p><a href="#section2">Jump to Section 2</a></p>

<h2 id="section1">Section 1</h2>

<p>This is the content of section 1.</p>

<h2 id="section2">Section 2</h2>

<p>This is the content of section 2.

<a href="#section1">Go back to Section 1</a></p>

</body>

## **5. Download:-**

href attribute can also be used in combination with the download attribute to create a link that prompts the user to download a file when clicked.

<p>Click the link below to download a sample image file:</p>

<a href="car.jpg" download="car.jpg">Download image</a>

<p>Click the link below to download a sample mp3 file:</p>

<a href="Ocean.mp3" download>Download mp3</a>

## **6. Target:-**

target attribute is used to specify where a linked resource should be displayed when clicked.

* **\_self:** The default behavior. The linked content opens in the same frame or window where the link was clicked.
* **\_blank:** The linked content opens in a new browser tab or window.
* **\_parent:** The linked content opens in the parent frame, useful in the context of framesets.
* **\_top:** The linked content opens in the full body of the window, breaking out of any frames.

<p>If you set the target attribute to "\_blank",

the link will open in a new browser window or tab.

</p>

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

Google

</a>

<p>If you set the target attribute to "\_self",

the link will open in the same window or tab.

</p>

<a href="https://www.google.com" target="\_self">

Google

</a>

<p>If you set the target attribute to "\_top",

the link will open in the full body of the window.

</p>

<a href="https://www.google.com" target="\_top">

Google

</a>

<p>If you set the target attribute to "\_parent",

the link will open in the parent frame.

</p>

<a href="https://www.google.com" target="\_parent">

Google

</a>

## **7. Color:-**

By default, a link color will appear in the browser.

* An unvisited link is underlined and blue
* A visited link is underlined and purple
* An active link is underlined and red

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Document</title>

</head>

<body link="green" vlink="cyan" alink="red">

<h2>This is a color link</h2>

<p> <a href="https://www.google.com"> <h3>click</h3> </a></p>

<p> <a href="https://www.google.com"> <h3>click</h3> </a></p>

<p> <a href="https://www.google.com"> <h3>click</h3> </a></p>

</body>

</html>

## **8. Link title:-**

The title attribute provides additional information about the link when the user hovers over it with their cursor.

<body>

<h1> link title</h1>

<p> <a href="https://www.google.com" title="Visit google"> <h3>click</h3> </a></p>

</body>

## 9. Image link:-

This allows you to create a clickable image that links to another page or resource.

<body>

<h1> link image</h1>

<a href="https://www.lamborghini.com/en-en" > <img src="car.jpg" alt="car" width="200" height="150"></a>

</body>

# ***<iframe>:-***

The <iframe> tag in HTML stands for "inline frame." It is used to embed another HTML document or web page within the current document such as videos, maps, interactive elements, or even an entire webpage.

Syntax of the <iframe> tag:

<iframe src="URL"></iframe>

**src**: Specifies the URL of the content you want to embed. It can be a relative or absolute URL.

Example:

<iframe src="https://www.youtube.com/></iframe>

## **Set width and height:-**

<body>

<h1> iframe Tag</h1>

<iframe src="http://www.wikipedia.org/"></iframe>

<iframe src="12.6 link image.html" width=”500 height=”600”></iframe>

</body>

## **Framename:-**

**target** attribute is used to specify where a linked resource should be displayed when clicked. If you want to open a link in a specific frame, you can use the target attribute along with the name attribute of the **<iframe>** element to define the target frame.

<a href="target\_page.html" target="framename">Open in Frame</a>

## **Sandbox:-**

The sandbox attribute is used in HTML to create a **restricted and isolated environment** for embedded content within an <iframe> element.It allows you to specify which actions and capabilities the embedded content is allowed to perform, thereby **preventing potentially malicious or unwanted behavior**.

### **allow-same-origin:**

Allows the embedded content to interact with its parent document if they share the same origin (domain, protocol, and port). This is the default behavior if the sandbox attribute is used without any values.

### **allow-scripts**:

Enables JavaScript execution within the iframe. Without this value, JavaScript execution will be disabled in the iframe.

### 

### **allow-forms**:

Permits the embedded content to submit forms.

### **allow-popups**:

Allows the embedded content to open new browser windows or pop-up windows.

### **allow-pointer-lock**:

Allows the embedded content to use the Pointer Lock API to capture the mouse cursor.

### **allow-top-navigation**:

Permits the embedded content to navigate the top-level window (the parent window). This is disabled by default.

# ***<img>:-***

The **<img>** tag to display images on a web page. The **<img>** tag does not have a closing tag because it's a self-closing element.

<img src="image-url" alt="alternative-text">

Here's what the attributes mean:

**src**: This attribute specifies the URL or path to the image file you want to display. It can be a relative or an absolute path.

**alt**: This attribute provides alternative text for the image. It's important for accessibility and SEO. If the image cannot be displayed, the alternative text will be shown instead. Make sure the alternative text describes the content of the image.

<img src="example.jpg" alt="A beautiful sunset" width="300" height="200" border="1">

ismap attribute:- using ismap attribute image coordinates are sent to the server. ismap attributes are used in anchor tags.

<a href=""><img src="car.jpg" alt="car.jpg" width="800" height="500" ismap="ismap" border="10" ></a>

## **<map>:-**

The <map> tag in HTML is used in conjunction with the <img> tag to define clickable areas on an image. These clickable areas are known as image maps. Each <map> tag contains one or more **<area> tags that define specific regions of the image to be clickable.**

Each <area> tag has attributes:

**shape**: Specifies the shape of the clickable area (rect, circle, or poly).

**coords**: Defines the coordinates of the area based on the chosen shape.

**href:** Specifies the URL that the clickable area should link to.

**alt**: Provides alternative text for the area (similar to the alt attribute for images).

# **<audio> and <video>:-**

Both <audio> and <video> almost have the same attributes, <video> have extra attributes.

**<poster=”image”>:-** Set thumbnail of the video

**1.src:** Specifies the URL of the audio file to be played.

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

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

<audio src="audio.mp3" controls></audio>

**3. autoplay:** Makes the audio play automatically when the page loads.

<audio src="audio.mp3" autoplay></audio>

**4.loop:** Makes the audio repeat continuously.

<audio src="audio.mp3" loop></audio>

**5.preload:** Specifies whether and how the audio file should be preloaded. It can take values "none", "metadata", or "auto".

**auto:- will load entire audio file**

**Metadata:- only metadata load when page loads**

**None:- should not load audio file when page loads**

<audio src="audio.mp3" preload="auto"></audio>

**6. muted:** Specifies whether the audio should be muted by default.

<audio src="audio.mp3" muted></audio>

# <object>:-

The <object> tag in HTML is **used to embed external content within a web page**. It is a versatile element that can be used to embed various types of content, including multimedia (such as **audio and video**), interactive elements (like **Flash animations**), and other external resources.

Here is the basic structure of the <object> tag:

<object data="URL or path to content" type="MIME type">

<!-- Alternative content goes here -->

</object>

**data**: Specifies the URL or local path to the external content you want to embed.

**type**: Specifies the MIME (Multipurpose Internet Mail Extensions) type of the content. This attribute helps the browser understand how to handle the content.

The content inside the <object> element, typically wrapped in a <p> or other HTML tags, serves as alternative content. This **content will be displayed if the browser doesn't support the embedded content or if the external resource cannot be loaded.**

# **<form>:-**

The <form> tag in HTML is **used to create a form on a web page** that allows users to input data, which can then be submitted to a server for processing. Forms are a fundamental part of web development and are used for a wide range of purposes, including user registration, login, search, contact forms, and more.

**Basically if we want to collect data from a user we use from tag.**

## **<input>:-**

Input tag is used for collecting user data from a web page.

The <input> tag in HTML supports various **type** attribute values for different types of user input.

Commonly used **type** attribute :

**1.** **text:** A single-line text input.

**2.** **password:** A text input for passwords (characters are masked).

**3. button:** A generic button with no default behavior (used for custom JavaScript actions).

**4.** **email:** A text input for entering email addresses.

**5.** **checkbox:** A checkbox for binary choices.

**6. radio:** Radio buttons for selecting a single option from a group.

**7. url:** A text input for entering URLs (web addresses).

**8. search:** A text input for search queries.

**9. tel:** A text input for telephone numbers.

**10 range:** A slider control for selecting a value within a specified range.

**11. number:** A text input for numeric values.

**12. file:** A file upload input for uploading files to the server.

**13. image:** An image-based submit button (uses an image as the button).

**14. hidden:** A hidden input for storing data on the client side (not visible to the user).

**15. date:** A date input for selecting dates.

**16. time:** A time input for selecting times.

**17. datetime:**

**18. datetime-local:** A date and time input, including the local time zone.

**19. month:** A month and year input.

**20. week:** A week and year input.

**21. submit:** A button to submit form data to the server.

**22. reset:** A button to reset form fields to their initial values.

**23. color:** A color picker input.

# **autocomplete:**

By default, the autocomplete attribute in HTML is typically set to "on" for most form elements.

If you want to disable autocomplete for a specific input field, you can set the autocomplete attribute to "off" like this:

<input type="text" name="username" autocomplete="off">

If you want to use autocomplete for all input, you can declare it in the form tag.

<form action="" autocomplete="on">

First Name: <input type="text" name="fname"><br><br>

Last Name: <input type="text" name="lname"><br><br>

city: <input type="text" name="city" autocomplete="off"><br><br>

<input type="submit">

</form>

# **Method Attribute:**

In HTML forms, the "GET" and "POST" methods are used to send data from a web page to a server for processing.

## **1.GET Method:**

**Data in URL:** The form data is appended to the URL as query parameters in name=value pair.

**Data Limit:** Limited amount of data that can be sent in the URL up to 2048 characters.

**Caching:** GET requests are often cached by browsers.

**Bookmarkable and Shareable:** GET requests can be bookmarked and shared with others since all the data is included in the URL.

**Example:** Used for search queries or navigating to different pages on a website.

## **2.POST Method:**

**Data in Request Body:** The "POST" method sends the form data in the HTTP request body, rather than in the URL. This means that the data is not visible in the URL bar.

**Data Size:** Sends larger amounts of data and is not limited by URL length constraints. It is commonly used for file uploads and transmitting form data with many fields.

**Not Cached:** POST requests are not cached by browsers, which makes them more appropriate for operations that should not be repeated or cached, such as submitting a payment form.

**Not Bookmarkable or Shareable:** POST requests cannot be easily bookmarked or shared because the data is not included in the URL.

Example: Used for submitting forms that involve sensitive information, making payments, or any action that modifies data on the server.

# **Enctype attribute:**

The enctype attribute is used in HTML forms to specify how the data submitted via the form should be encoded before being sent to the server. It is typically used with the "POST" method, which is used for submitting form data that may include file uploads or binary data.

**1. application/x-www-form-urlencoded (Default):** This is the default encoding type. It URL-encodes the form data, converting spaces into ”+” and special characters into ASCII HEX values that can be transmitted via HTTP. This is suitable for standard textual data.

<form method="POST" enctype="application/x-www-form-urlencoded">

<!-- Form controls go here -->

</form>

**2. multipart/form-data:** This encoding type is used when you have file uploads in your form. It is designed to handle binary data and is commonly used with file input fields (<input type="file">). It allows for the transmission of files and textual data together.

<form method="POST" enctype="multipart/form-data">

<!-- File upload input and other form controls go here -->

</form>

**3. text/plain:** This encoding type sends the data as plain text with no special formatting or encoding. It is rarely used but can be useful in specific cases where data should be transmitted as-is.

<form method="POST" enctype="text/plain">

<!-- Form controls go here -->

</form>

* **application/x-www-form-urlencoded:** for standard text data.
* **multipart/form-data:** for file uploads.
* **text/plain:** only in specialized situations where plain text is required.