

HTML & CSS

<https://www.tutorialspoint.com/html/>

HTML is a standard markup language, which stands for HyperText Markup Language. It is a widely used language to create webpages.

HyperText defines the internal links between web pages, and **Markup language** defines the layout and presentation of text and media.

Basic HTML Structure

```
<!DOCTYPE html>

<html>

<head>

<title>Page title</title>

</head>

<body>

<h1>Webpage's Heading</h1>

<p>Content (Your first paragraph).</p>

</body>

</html>
```

zElements of HTML Basic Structure

The following are the basic tags that define the basic HTML structure –

1. <!DOCTYPE html>

This element defines the document type as HTML. This element must be written before writing any HTML document.

2. <html>...</html>

The <html> tag is the parent tag for all HTML elements. Everything related to create an HTML document must be written inside the <html> tag. CSS, JavaScript, and jQuery must also be written inside this tag.

3. <head>...</head>

The <head> tag is a container tag for all those elements that are not directly displayed on the webpage but required for the page functionalities. It contains meta tags (which are used for SEO purposes), title tag, script tags, etc.

4. <title>...</title>

The <title> tag is used to define the title of the webpage that you can see in the browser's tab, bookmarks list, and search engine results. This tag is also very important for SEO purposes to help search engine to understand the content of the webpage.

5. <body>...</body>

The <body> tag is the container tag for all those elements, which represents the main content of a webpage that displays on the browser.

6. <h1>...</h1>

The <h1> tag is one of the heading tags. It is the most important heading tag, which defines the main title or headline of the webpage. Any text written inside <h1> and </h1> is a top-level heading of the content.

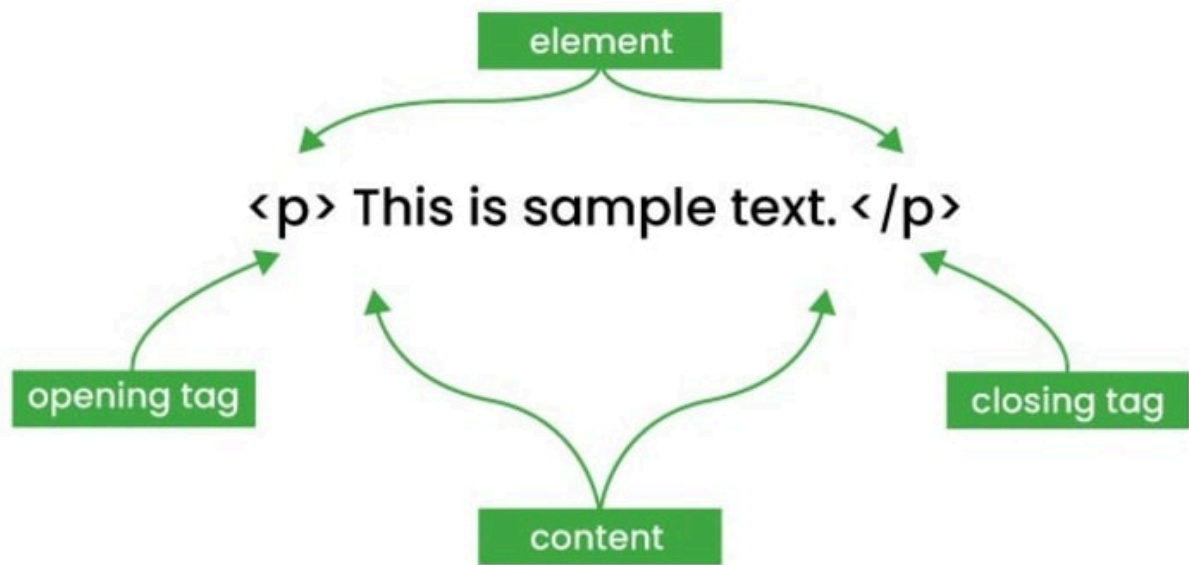
7. <p>...</p>

The <p> tag defines a paragraph, anything written inside <p> and </p> displays as a paragraph on the webpage. Use multiple <p> tags to display text in different paragraphs.

What is an HTML Element?

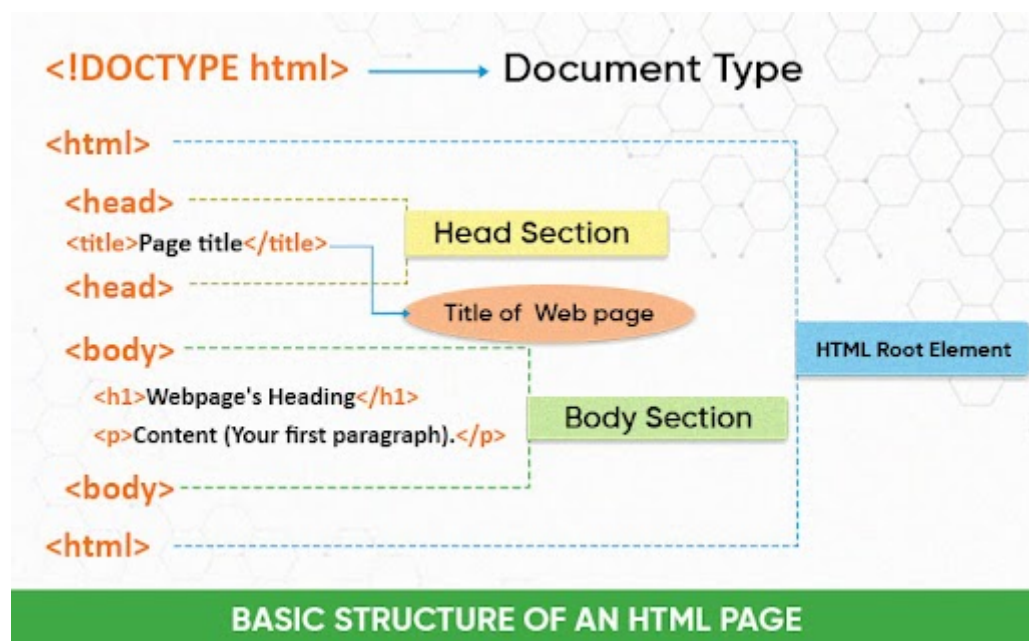
An HTML element is a basic building block to create a webpage, and It is created by a start tag, content, and end tag. In an HTML element, the content is placed between a start and end tag.

The basic syntax of an HTML element is –



`<tag_name>content</tag_name>`

`<h1>It is top-level heading</h1>`



Web Browser Role

The role of a web browser is to read HTML documents from the given path (either from the server or from a local device) and display it on the webpages. All web browsers, such as

Google Chrome, Safari, Firefox, etc., are compatible with reading HTML documents. You can use any of the web browsers to display your HTML document in web format.

The <!DOCTYPE> Declaration

The <!DOCTYPE> declaration tag is used by the web browser to understand the version of the HTML used in the document. The current version of HTML is 5 and it makes use of the following declaration –

```
<!DOCTYPE html>
```

HTML Tags Vs. Elements Vs. Attributes

HTML tags are the keywords that can be used for a specific purpose to display and format the content on the webpage.

HTML elements are the basic building blocks that are made with the help of tags and content. An HTML element is created with a start tag, a content, and an end tag.

And, HTML attributes provide additional information about HTML elements; in order to define or change their behavior. Attributes are used with an opening tag.

HTML Tags Case Sensitivity

HTML tags are **not case-sensitive**. They can be written in **uppercase or in lowercase**. But the World Wide Web Consortium (W3C) recommends using lowercase tags starting from HTML 4.

Popular HTML Editors

Notepad

TextEdit

Notepad++

Sublime

Visual Studio Code

Atom

Brackets

HTML - Basic Tags

HTML tags are the fundamental elements of HTML used for defining the structure of the document. These are letters or words enclosed by angle brackets (< and >).

Heading Tags	<h1>, <h2>, <h3>, <h4>, <h5>, and <h6>
Paragraph Tag	<p>
Line Break Tag	 or
Center Tag	<center>
Horizontal Rule Tag	<hr> or <hr />
Preserve Formatting Tag	<pre>
Non-breaking Space	
Listing Tags	<ul style="list-style-type: none"> and • Unordered Listing: • Ordered Listing: • List Item:
HTML Basic Tags	

HTML Void Elements

HTML void elements are those elements that don't require closing tags. These tags don't have any content model and don't even allow nesting of elements. The void elements are also known as empty or self-closing elements.

Some of the void elements are such as , <hr />, and
 elements. The below table shows a list of void elements –

HTML - Attributes

HTML attributes are special words that provide additional information to an HTML element. Attributes are placed inside the element's opening tag, and they are used to configure or adjust the element's behavior. All attributes are made up of two parts: a name and a value –

```
<p id="html">This is a paragraph.</p>
```

```
Name: The attribute name (id, class, src, etc.).
```

```
Value: The content or value assigned to that attribute  
("html", "example.jpg", etc.).
```

```
<!-- Specifies the source file for an image ->
```

```

```

```
<!-- Defines a hyperlink to another page or resource ->
```

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

```
<!-- Sets an alternative text for an image if the image cannot be displayed ->
```

```

```

```
<!-- Specifies the text for a button ->
```

```
<button type="button">Click Me!</button>
```

```
<!-- Defines the target for a hyperlink (where the link will open) ->
```

```
<a href="https://www.example.com" target="_blank">Open in New Tab</a>
```

```
<!-- Sets a unique identifier for an element ->
```

```
<div id="header">Header Content</div>
```

```
<!-- Specifies the class for an element to apply CSS styles ->
```

<div class="container">This is a container</div>

<!-- Defines a style to apply CSS directly to an element ->>

<p style="color: blue;">This is a blue paragraph.</p>

<!-- Specifies the width of an element ->>

<!-- Specifies the height of an element ->>

<!-- Specifies the type of input field ->>

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

<!-- Indicates that an input field is required to submit a form ->>

<input type="email" required>

<!-- Specifies the action for a form (where data is submitted) ->>

<form action="submit_form.php" method="POST">Form Content</form>

<!-- Sets the language of the document ->>

<html lang="en">

<!-- Specifies the title of the document ->>

<title>My Web Page</title>

<!-- Defines a meta tag for character set ->>

<meta charset="UTF-8">

<!-- Specifies the autocomplete behavior of an input field ->>

<input type="text" autocomplete="on">

<!-- Specifies a placeholder text in a form input ->>

<input type="text" placeholder="Enter your email">

<!-- Defines the size of an input field ->>

<input type="text" size="30">

<!-- Specifies a value to pre-fill an input field ->>

<input type="text" value="John Doe">

<!-- Specifies the disabled state for an element (disables it) ->

<button disabled>Disabled Button</button>

<!-- Specifies a name for form elements to group them ->

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

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

<!-- Specifies the type of the button (submit, reset, etc.) ->

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

<!-- Specifies the language of the document ->

<p lang="en">This is an English text.</p>

<!-- Sets a minimum value for an input element ->

<input type="number" min="1">

<!-- Specifies the maximum value for an input element ->

<input type="number" max="100">

<!-- Sets the autofocus attribute on an element ->

<input type="text" autofocus>

<!-- Specifies the maximum length of an input field ->

<input type="text" maxlength="10">

<!-- Defines a background color for an element ->

<div style="background-color: lightblue;">This is a background-colored div</div>

<!-- Specifies the source file for audio ->

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

<!-- Specifies the source file for video ->

<video src="video.mp4" controls></video>

<!-- Specifies whether an element is visible or not ->

<div hidden>This content is hidden</div>

<!-- Specifies a custom data attribute ->

<div data-user-id="123">User Info</div>

<!-- Specifies the role of an element for accessibility ->

`<div role="button">This is a button</div>`

`<!-- Specifies the type of list item (used for ordered lists) ->`

``

`<li value="10">Item 1`

`Item 2`

``

`<!-- Defines a relationship between the current document and a linked resource ->`

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

`<!-- Specifies that an input field accepts numeric values ->`

`<input type="number" step="0.1" value="3.14">`

`<!-- Specifies the form method (GET or POST) ->`

`<form method="POST">Form Content</form>`

`<!-- Specifies that an input field is of password type ->`

`<input type="password" placeholder="Enter your password">`

`<!-- Defines a list of options for a drop-down menu ->`

`<select>`

`<option value="apple">Apple</option>`

`<option value="orange">Orange</option>`

`</select>`

`<!-- Defines a textarea element for multi-line text input ->`

`<textarea rows="4" cols="50">Enter text here</textarea>`

`<!-- Specifies an element to be focused after page load ->`

`<input type="text" id="input1" autofocus>`

`<!-- Specifies whether the user can resize the textarea ->`

`<textarea resize="both">Resizable Textarea</textarea>`

`<!-- Specifies an inline style to apply a CSS rule to the element ->`

`<p style="color:red;">This text is red.</p>`

`<!DOCTYPE html>`

`<html>`

```

<body>
  <h1>This is Heading 1 (H1 Tag)</h1>
  <h2>This is Heading 2 (H2 Tag)</h2>
  <h3>This is Heading 3 (H3 Tag)</h3>
  <h4>This is Heading 4 (H4 Tag)</h4>
  <h5>This is Heading 5 (H5 Tag)</h5>
  <h6>This is Heading 6 (H6 Tag)</h6>
</body>
</html>

```

The Tag

You can use the `` tag to apply inline styles or classes to specific portions of the text within a heading. This allows for custom styling of text within the heading.

```

<h2>This is a <span style="color: blue;">blue</span>
word.</h2>

```

The and Tags

These tags are used for emphasizing text within headings. The `` tag *italicised* the text, while `` makes it **bold**.

The <sup> and <sub> Tags

In heading, to include **superscript** or **subscript** text within a heading, use `<sup>` and `<sub>`.

```

<h4>The 10<sup>th</sup> floor is at the top.</h4>
<h5>The chemical formula for water is H<sub>2</sub>O.

```

The 10th floor is at the top.

The chemical formula for water is H₂O.

The `<abbr>` Tag for Abbreviations

When you need to include an **abbreviation or acronym** in a heading, use the `<abbr>` tag. It often provides a tooltip with the full meaning.

```
<h2>HTML stands for <abbr title="Hypertext Markup  
Language">HTML</abbr>.</h2>
```

The `
` Tag for Line Breaks like :- “\n” in programming language.

```
<h3>This is the first line.<br>This is the second line.</h3>
```

This is the first line.

This is the second line.

The `<mark>` Tag

Use the `<mark>` tag to **highlight** specific text within a heading. It's often used to indicate search results or selected portions of text.

```
<h1>Search for "<mark>important</mark>" information here.</h1>
```

Search for "**important**" information here.

HTML - Block and Inline Elements

HTML block elements are used to create the logical and semantic layout of a web page. They help to organize the content into meaningful sections and make it easier for browsers, search engines, and site visitors to understand the structure and meaning of different parts

of the web page. Inline elements are used to make useful block elements, like adding anchor links.

| HTML Block Elements | | | | |
|---------------------------|------------------------|-----------------------|---------------------------|------------------------------------|
| <address> | <article> | <aside> | <blockquote> | <canvas> |
| <dd> | <div> | <dl> | <dt> | <fieldset> |
| <figcaption> | <figure> | <footer> | <form> | <h1> -
<h6> |
| <header> | <hr> | | <main> | <nav> |
| <noscript> | | <p> | <pre> | <section> |
| <table> | <tfoot> | | <video> | |

HTML Inline Elements

Inline elements can appear within the same line and do not start a new line on their own.

| HTML Inline Elements | | | | |
|----------------------|-----------------------|------------------------|-----------------------|-------------------------|
| <a> | <abbr> | <acronym> | | <bdo> |
| <big> |
 | <button> | <cite> | <code> |
| <dfn> | | <i> | | <input> |
| <kbd> | <label> | <map> | <object> | <output> |
| <q> | <samp> | <script> | <select> | <small> |
| | | <sub> | <sup> | <textarea> |
| <time> | <tt> | <var> | | |

<!-- Outer div element (Block-level) -->

<div class="outer-div">

<!-- Content inside the outer div (Block-level) -->

<h1>Welcome to the Outer Div</h1> <!-- Block-level: Takes full width and starts on a new line -->

<p>This is some text inside the outer div.</p> <!-- Block-level: Paragraph that takes full width and creates space above and below -->

<!-- Inner div inside the outer div (Block-level) -->

<div class="inner-div">

<!-- Content inside the inner div (Block-level) -->

<h2>Inner Div</h2> <!-- Block-level: Another heading, similar to h1 -->

<p>This is some text inside the inner div. It is nested inside the outer div.</p> <!-- Block-level: Paragraph inside the inner div -->

<!-- Example of Inline Elements inside the inner div -->

<p>This is an inline example:

This text is inline and appears within the paragraph. <!-- Inline element: The span element only takes as much width as needed and doesn't break the flow of text. -->

</p>

</div>

</div>

HTML - Formatting

HTML formatting defines the way of **content representation** on the webpage to improve the **readability**, to give the semantic meaning, and to improve **visual styling**.

Tag	Description	Category
	This tag is used to make the text bold .	Physical Tag
<i>	This tag is used to make the text <i>italic</i> .	Physical Tag
<big>	This tag is used to make the text bigger. It is not supported in HTML5 .	Physical Tag
<small>	This tag is used to make the text smaller.	Physical Tag
<u>	This tag is used to underline text.	Physical Tag
<strike>	This tag is used to strike through text. It is not supported in HTML5 .	Physical Tag
<tt>	This tag is used to make text appear in teletype (monospace font). It is not supported in HTML5 .	Physical Tag
	This tag is used to bold text and give it semantic importance .	Logical Tag
	This tag is used to italicize text and give it <i>semantic emphasis</i> .	Logical Tag
<sup>	This tag is used to make superscript text (slightly above the normal line).	Other Tag
<sub>	This tag is used to make subscript text (slightly below the normal line).	Other Tag
<ins>	This tag is used to indicate that content has been added (typically underlined).	Other Tag
	This tag is used to indicate that content has been deleted (typically struck through).	Other Tag

<mark>	This tag is used to highlight text with a yellow background.	Other Tag
---------------------	--	-----------

HTML - Quotations

HTML quotations allow you to include and format quoted text within your web content. HTML provides tags such as `<blockquote>`, `<q>`, `<cite>`, `<address>`, `<bdo>`, and `<abbr>` to structure and style quotes.

Tag	Description
<q>	Defines a short inline quotation.
<blockquote>	Defines a block-level indented quotation.
<cite>	Specifies a reference to the title of a creative work, such as books or articles.
<address>	Defines contact information.
<bdo>	Overrides text direction.
<abbr>	Defines an abbreviation or acronym.

HTML Element

```

```

**Image
Source**

**Alternative
Text**

HTML - Iframes

HTML iframe is an inline frame that allows you to embed another document within the current HTML document. Whenever you want to display another webpage within the webpage, you can use an iframe.

```
<iframe src="/html/m.htm"> Sorry your browser does not  
support inline frames. </iframe>
```

HTML <kbd> Element

The <kbd> tag is used to define the keyboard input. Use this when you want the user to type on their keyboard, for example, shortcut keys Ctrl+C for copy, Esc for exit, etc.

HTML Computer Code Elements

Here is the list of the computer code tags along with their descriptions used for defining user input and computer code:

Tag	Description
<code><kbd></code>	Defines keyboard input.
<code><pre></code>	Displays preformatted text with preserved spaces and line breaks.
<code><code></code>	Defines a piece of computer code.
<code><var></code>	Represents a variable in programming or math expressions.
<code><samp></code>	Defines sample output from a program or device.

HTML `<meta>` tag lets us specify metadata, which is additional important information about a document, in a variety of ways. The META elements can be used to include name and content pairs describing properties of the HTML document, such as author, expiry date, a list of keywords, document author, etc.

Setting Cookies

Cookies are data stored in small text files on your computer, and it is exchanged between a web browser and a web server to keep track of various information based on your web application needs.

You can use the `<meta>` tag to store cookies on the client side, and later this information can be used by the Web server to track a site visitor. If you do not include the expiration date and time, the cookie is considered a session cookie and will be deleted when the user exits the browser.

Example

Following is an example of redirecting the current page to another page after 5 seconds. If you want to redirect the page immediately, then do not specify the **content** attribute.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
  <title>Meta Tags Example</title>
```

```
  <!-- Meta tag for specifying keywords relevant to the page -->
```

```

<meta name="keywords" content="HTML, Meta Tags, Metadata" />

<!-- Meta tag for providing a short description of the page -->

<meta name="description" content="Learning about Meta Tags." />

<!-- Meta tag indicating when the page was last revised -->

<meta name="revised" content="Tutorialspoint, 3/7/2014" />

<!-- Meta tag for setting a cookie with a specific expiry date -->

<meta http-equiv="cookie" content="userid=xyz; expires=Wednesday, 08-Aug-15 23:59:59
GMT;" />

</head>

<body>

    <p>Hello HTML5!</p> <!-- A simple paragraph displaying text -->

</body>

</html>

```

HTML - Classes

The class is an important keyword in HTML. It is an attribute that can be applied to one or more elements and is used to style and categorize elements based on common characteristics or purpose. Classes allows multiple elements to share the same styling rules. By assigning the same class to multiple elements, you can apply CSS styles or JavaScript functionality to all of them simultaneously. This promotes consistency in design and layout, making it easier to manage and update a website.

```
<element class="highlight">...</element>
```

```
/* CSS using class Attribute Selector */
```

```
.highlight {
```

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

```
color: black;

font-weight: bold;

}
```

```
document.getElementsByClassName('highlight')
```

HTML - id

The id is an important keyword in HTML. HTML "id" is an attribute used to uniquely identify an element within a web page. It serves as a label for that element and enables JavaScript and CSS to target it specifically.

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<script>
```

```
function showContent() {

var element = document.getElementById('content');

if (element.style.display === 'none') {

element.style.display = 'block';

} else {

element.style.display = 'none';

}

}
```

```
</script>
```

```
<style>
```

```
.interactive-button {

background-color: #007bff;
```

```
color: #fff;

padding: 10px 20px;

border: none;

cursor: pointer;

}

</style>

</head>

<body>

<button class="interactive-button"

onclick="showContent()">Click Me</button>

<p class="content" style="display: none;">

This content can be toggled by clicking the button.

</p>

</body>

</html>
```

Difference between id and class in HTML

In HTML, the id attribute uniquely identifies a single element on a page, making it useful for targeting with CSS and JavaScript, and it must be unique within the document. The class attribute, on the other hand, can be applied to multiple elements, allowing for the grouping of elements that share common styles or behaviors.

Within the document every id must be unique.

HTML - Tables

HTML <table> Tag: This tag is used to create the table that wrap the rows and columns within it.

HTML <tr> Tag: Stands for "table row" and is used to create a row within the table.

HTML <td> Tag: Represents "table data" and is used to create standard cells within a row.

HTML <th> Tag: Represents "table header" and is used to create header cells within a row.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<table border="1">
```

```
<tr>
```

```
<th>Product</th>
```

```
<th>Category</th>
```

```
<th>Price</th>
```

```
</tr>
```

```
<tr>
```

```
<td>Laptop</td>
```

```
<td>Electronics</td>
```

```
<td>$800</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Bookshelf</td>
```

```
<td>Furniture</td>
```

```
<td>$150</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Coffee Maker</td>
```

```
<td>Appliances</td>
```

```

<td>$50</td>

</tr>

</table>

</body>

</html>

```

HTML Table Header and Caption

Employee Record		
Name	Age	City
Kelly Hu	21	Honolulu, Hawaii
Julia Roberts	23	Smyrna, Georgia

```
<caption>Description of table</caption>
```

HTML - Lists

HTML lists are groups or collections of items. These items can be both organized and unorganized depending on the requirement. They help in organizing, structuring, and presenting information to make it more user-friendly, readable, and accessible.

An Organized List	Numerically Organized List
<ul style="list-style-type: none"> ▪ Rice ▪ Wheat Flour ▪ Vegetables ▪ Pulses ▪ Milk 	<ol style="list-style-type: none"> 1. Aqua Man 2. Bat Man 3. Captain America 4. Hulk 5. Iron Man

Unordered lists

Unordered lists display lists of items that are not in a specific order. The unordered lists are marked with bullet points. To create an unordered list, the `` tag is used along with the `` tag. Here, the `` tag specifies the list items.

```
<ul>

<li>HTML</li>

<li>CSS</li>

<li>JavaScript</li>

<li>Java</li>

<li>JavaFX</li>

</ul>
```

Example of HTML List

- HTML
- CSS
- JavaScript
- Java
- JavaFX

Ordered Lists

Ordered lists are lists of items that are in a specific order. The ordered lists are marked with numbers by default; you can change the numbers into alphabets, roman numbers, etc. by using the `type` attribute or the `CSS list-style-type` property.

```
<ol>

<li>HTML</li>

<li>CSS</li>

<li>JavaScript</li>

<li>Java</li>

<li>JavaFX</li>

</ol>
```

Example of HTML List

1. HTML
2. CSS
3. JavaScript
4. Java
5. JavaFX

HTML - Text Links

HTML Links (Hyperlinks) are words or buttons having a link to another page that take the user to that linked page when clicked.

A hyperlink is a specific type of link that allows users to navigate from one web page or resource to another by clicking on it. You can create hyperlinks using text or images available on a webpage. A hyperlink is created using the HTML Anchor Tag (<a>).

```
<a href="URL" target="target_type">Link Text</a>
```

S.No.	Option & Description
1	<u>blank</u> Opens the linked document in a new window or tab.
2	<u>self</u> Opens the linked document in the same frame.
3	<u>parent</u> Opens the linked document in the parent frame.
4	<u>top</u> Opens the linked document in the full body of the window.
5	<u>targetframe</u> Opens the linked document in a named targetframe.

Linking to a Page Section

Linking to a section on the same page allows users to navigate directly to that section. You can create a link in the same to a specific section by using the href attribute with a **#id value**, where the **#id** targets an element on the page with a corresponding id attribute.

Downloadable Links

HTML allows you to create downloadable links where you can create links to make your PDF, DOC, or ZIP files downloadable. To create any link downloadable, you can use the download attribute with the <a> tag and specify the downloadable file path in the href attribute.

```
<a href="/html/src/sample.txt" download>Download File</a>
```

HTML - Image Links

Images can also be used as links in HTML, which means by clicking the images we can navigate to other web pages or resources. HTML image links are very useful in creating websites like photo galleries, portfolios, online stores, and so on. In this article, we will learn how to use images to create hyperlinks. It is similar to the HTML - Text Link.

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

```

```

```
</a>
```

Image Link with Tooltip

You can also define a tooltip for an image link; when someone moves the mouse over the linked image, it will display a tooltip. To set the tooltip, you can set the title attribute of the <a> tag.

```
<a href="https://www.tutorialspoint.com" title="Go to  
TutorialsPoint">
```

```

```

```
</a>
```

Mouse-Sensitive Images

The HTML and XHTML standards provide a feature that lets us embed several different links inside a single image. We can create different links on the single image based on different coordinates available on the image.

HTML - Email Links

HTML email links allow users to click on a link and automatically open their default email client with a new message composed to the specified email address.

```
<a href= "mailto:name@email.com">name@email.com</a>
```

HTML Forms

HTML forms are collections of interactive controls and various input types, such as text, numbers, email, password, radio buttons, checkboxes, buttons, etc., that collect user information. HTML forms are created by using the HTML <form> tag. All user input-related tags are placed inside the <form> tag.

Attribute	Description
action	It is used to specify a URL that processes the form submission.
method	It is used to define which HTTP method to use when submitting the form.
target	It is used to specify where to open the linked document.
autocomplete	It allows you to set whether the autocomplete for the form should be on or off.
enctype	It is used to specify how the form input data should be encoded before sending it to the server.

novalidate

It defines that while submitting the form, the form data should not be validated in an HTML document.

GET

It is the default method for form submission, which means if we don't specify the method name explicitly, the form will use the GET method to send data.

POST

It is used to send form data inside HTTP request body. It is safer than GET method.

HTML - Form Controls

<!-- Text Input Control: A single-line input field for entering text. -->

```
<input type="text" name="username" placeholder="Enter your username">
```

<!-- Checkboxes Control: A control for selecting multiple options (can be checked or unchecked). -->

```
<input type="checkbox" name="subscribe" value="newsletter"> Subscribe to newsletter
```

<!-- Radio Buttons Control: A control for selecting a single option from a list of choices. -->

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

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

<!-- Select Box Control: A dropdown list to select one or more options. -->

```
<select name="country">
```

```
  <option value="usa">USA</option>
```

```
  <option value="uk">UK</option>
```

```
</select>
```

<!-- File Select Box: A control for selecting a file from the user's device. -->

```
<input type="file" name="upload">
```

<!-- Button Control: A clickable button to perform an action. -->

```
<button type="button">Click Me</button>
```

<!-- Hidden Form Control: A hidden input field used to store data that is not visible to the user. -->

```
<input type="hidden" name="userId" value="12345">
```

<!-- Datetime Controls: A control to input a date and time (combined). -->

```
<input type="datetime-local" name="meeting-time">
```

<!-- Date Control: A control for selecting a date. -->

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

<!-- Month Control: A control for selecting a month and year. -->

```
<input type="month" name="month">
```

<!-- Week Control: A control for selecting a week (with the year). -->

```
<input type="week" name="week">
```

<!-- Time Control: A control for selecting a time (without date). -->

```
<input type="time" name="alarm-time">
```

<!-- Number Control: A control for entering a numeric value. -->

```
<input type="number" name="age" min="18" max="100">
```

<!-- Range Control: A control for selecting a value within a specified range. -->

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

<!-- Email Control: A control for entering an email address (validates email format). -->

```
<input type="email" name="email" placeholder="Enter your email">
```

<!-- URL Control: A control for entering a URL (validates URL format). -->

```
<input type="url" name="website" placeholder="Enter your website URL">
```

HTML - Input Attributes

The HTML input attributes define the characteristics and behavior of the <input> element. These input attributes are used with the different types of input fields, such as text, email, password, date, number, and so forth. Note that the input element is used to create interactive controls for the web-based forms so that it can accept data from the user.

The <input> element requires only an opening tag, and it will work only if we add it in between the <form> tags. In this tutorial, we are going to explore the attributes that are used with the <input> element.

<!-- type: Specifies the type of input control. -->

```
<input type="text" name="username" placeholder="Enter your username">
```

<!-- name: Specifies the name of the input element, which is used to reference form data after submission. -->

```
<input type="text" name="email" placeholder="Enter your email">
```

<!-- value: Specifies the default value of an input element. -->

```
<input type="text" name="username" value="JohnDoe">
```

<!-- size: Specifies the visible width of the input element (only for text-based inputs). -->

```
<input type="text" name="username" size="30" placeholder="Enter username">
```

<!-- maxlength: Specifies the maximum number of characters allowed in an input field. -->

```
<input type="text" name="username" maxlength="15" placeholder="Max 15 characters">
```

<!-- readonly: Specifies that an input field is read-only, preventing the user from changing the value. -->

```
<input type="text" name="username" value="ReadOnly" readonly>
```

<!-- disabled: Disables the input field, preventing user interaction. -->

```
<input type="text" name="username" value="Disabled" disabled>
```

<!-- min and max: Specifies the minimum and maximum values allowed for a number or date input. -->

```
<input type="number" name="age" min="18" max="100">
```

<!-- accept and multiple: Used in file input to specify acceptable file types and allow multiple file selection. -->

```
<input type="file" name="file" accept="image/*" multiple>
```

<!-- placeholder: Provides a short hint within the input field before the user enters data. -->

```
<input type="text" name="search" placeholder="Search here...">
```

<!-- required: Specifies that an input field must be filled out before submitting the form. -->

```
<input type="text" name="username" required placeholder="Enter your username">
```

<!-- autofocus: Automatically focuses the input field when the page is loaded. -->

```
<input type="text" name="username" autofocus placeholder="Autofocus on this field">
```

<!-- list: Specifies a list of predefined options for an input field (works with <datalist>). -->

```
<input type="text" name="city" list="cities">
```

```
<datalist id="cities">
```

```
  <option value="New York">
```

```
  <option value="Los Angeles">
```

```
  <option value="Chicago">
```

```
</datalist>
```

HTML Video

The <video> element is used to enable video playback support within a web page. It works very similarly to the element, as it also requires adding the path or URL of the video within the src attribute. The HTML supports only MP4, WebM, and Ogg video formats. The <video> element also supports audio; however, the <audio> element is more suitable for that purpose.

<!-- src: Specifies the path to the video file. -->

```
<video src="movie.mp4" controls></video>
```

<!-- controls: Adds video controls (play, pause, volume, etc.) to the video player. -->

```
<video src="movie.mp4" controls></video>
```

<!-- autoplay: Specifies that the video should automatically start playing as soon as it is loaded. -->

```
<video src="movie.mp4" autoplay controls></video>
```

<!-- loop: Specifies that the video should start over again once it finishes. -->

```
<video src="movie.mp4" loop controls></video>
```

<!-- muted: Specifies that the video should be played without sound. -->

```
<video src="movie.mp4" muted controls></video>
```

<!-- poster: Specifies an image to be shown as a placeholder before the video starts playing. -->

```
<video src="movie.mp4" poster="image.jpg" controls></video>
```

<!-- width and height: Specifies the width and height of the video player in pixels. -->

```
<video src="movie.mp4" width="600" height="400" controls></video>
```

<!-- preload: Specifies if and how the video should be loaded when the page is loaded. -->

<!-- "auto" (default) starts downloading the video as soon as the page is loaded. -->

<!-- "metadata" downloads only metadata (like duration), and "none" doesn't load anything until the user hits play. -->

```
<video src="movie.mp4" preload="auto" controls></video>
```

<!-- type: Specifies the MIME type of the video file. -->

```
<video controls>
```

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

```
  <source src="movie.ogv" type="video/ogg">
```

Your browser does not support the video tag.

```
</video>
```

<!-- poster (again): Specifies a placeholder image that is shown before the video starts. -->

```
<video src="movie.mp4" poster="image.jpg" controls></video>
```

<!-- autoplay and loop together: The video will start playing as soon as it's loaded, and then loop indefinitely. -->

```
<video src="movie.mp4" autoplay loop controls></video>
```


<!-- crossorigin: Allows cross-origin resource sharing (useful when videos are hosted on other domains). -->

```
<video src="movie.mp4" crossorigin="anonymous" controls></video>
```

<!-- height and width (alternative): Another way to set the size of the video player. -->

```
<video src="movie.mp4" height="400" width="600" controls></video>
```

HTML Audio

The <audio> element is used to enable the support of audio files within a web page. We can include multiple sources of audio; however, the browser will choose the most appropriate file automatically. Most of the attributes of <video> element is also compatible with the <audio> element. The most frequently used attributes of the HTML audio element are controls, autoplay, loop, muted, and src.

<!-- src: Specifies the path to the audio file. -->

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

<!-- controls: Adds built-in audio controls (play, pause, volume, etc.) to the audio player. -->

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

<!-- autoplay: Specifies that the audio should start automatically once it is loaded. -->

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

<!-- loop: Specifies that the audio should start over once it finishes. -->

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

<!-- muted: Specifies that the audio should be played without sound. -->

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

<!-- preload: Specifies if and how the audio should be loaded when the page is loaded. -->

<!-- "auto" (default) starts downloading the audio as soon as the page is loaded. -->

<!-- "metadata" downloads only metadata (like duration), and "none" doesn't load until user plays. -->

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

<!-- type: Specifies the MIME type of the audio file. -->

```
<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>
```

<!-- autoplay and loop together: The audio will start playing automatically and loop indefinitely. -->

```
<audio src="audio.mp3" autoplay loop controls></audio>
```

<!-- crossorigin: Allows cross-origin resource sharing (useful when audio is hosted on other domains). -->

```
<audio src="audio.mp3" crossorigin="anonymous" controls></audio>
```

<!-- volume: Specifies the initial volume level (from 0.0 to 1.0, where 1.0 is 100%). -->

```
<audio src="audio.mp3" controls volume="0.5"></audio>
```

Embedding multimedia in an HTML document refers to the inclusion of content such as images, audio, video, animations, or other interactive media directly within a web page. These multimedia elements can enhance the user experience by making the content more engaging and interactive.

HTML provides several elements to embed different types of multimedia. Here's a breakdown of the most commonly used HTML elements for embedding multimedia content:

```
<head>
```

```
<title>Page Title</title>
```

```
<meta charset="UTF-8">
```

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

```
<meta name="description" content="Meta Description of the  
page">
```

```
<meta name="keywords" content="Keyword 1, Keyword 2, ...">
```

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

```
<script src="script.js" defer></script>
```

```
<link rel="icon" href="favicon.ico">
```

```
</head>
```

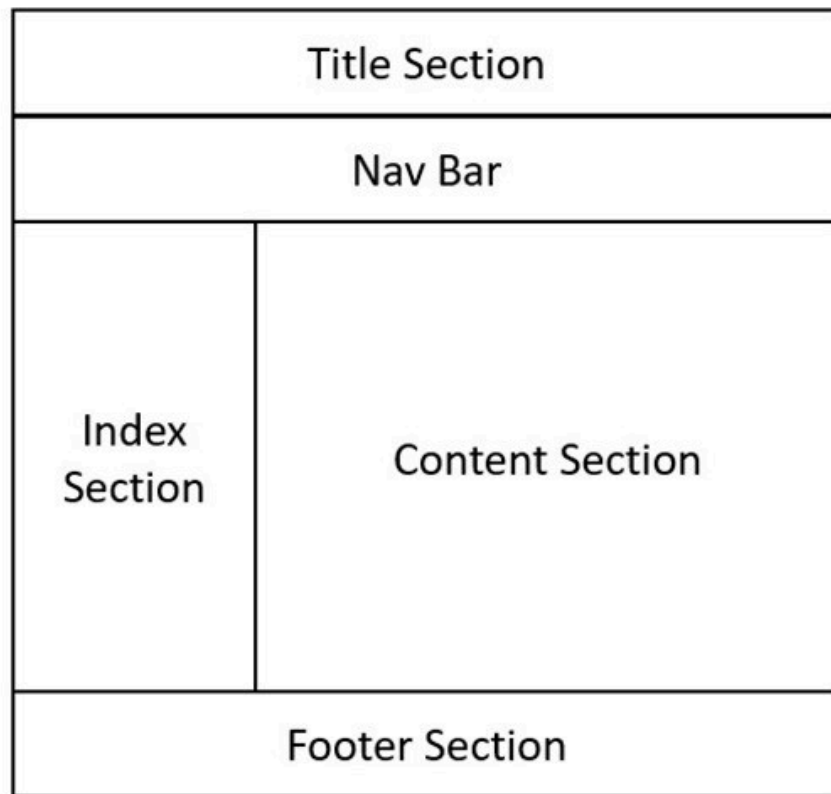
What is a HTML Favicon?

A favicon is a small image that represents your website and helps users identify it among multiple tabs, bookmarks, and search results. It can be in various formats, such as ICO, PNG, GIF, JPEG, or SVG, but ICO is the most widely supported format. If you have ever visited a website and noticed a small icon next to the page title in your browser's tab, you have seen a favicon.



```
<link rel = "icon" type = "image/png" href =  
"images/faviconTP.png">
```

HTML - Layout Elements



HTML `<header>` Element: Represents the header section of a document or a section, typically containing navigation links or introductory content.

HTML `<nav>` Element: Defines a navigation section with links to help users navigate through the site.

HTML `<section>` Element: Represents a thematic grouping of content, usually with a heading, to divide a page into sections.

HTML `<footer>` Element: Represents the footer section of a document or section, typically containing copyright information or contact links.

HTML `<article>` Element: Represents a self-contained piece of content that could be independently distributed, like a blog post or news article.

HTML `<aside>` Element: Defines content that is tangentially related to the surrounding content, such as sidebars or advertisements.

HTML `<div>` Element: A generic container element used to group content together for styling or layout purposes, with no inherent meaning.

Use `<div>` for generic grouping of content or layout purposes.

Use `<section>` when you want to group content that forms a distinct section of the document and carries a meaningful role.