

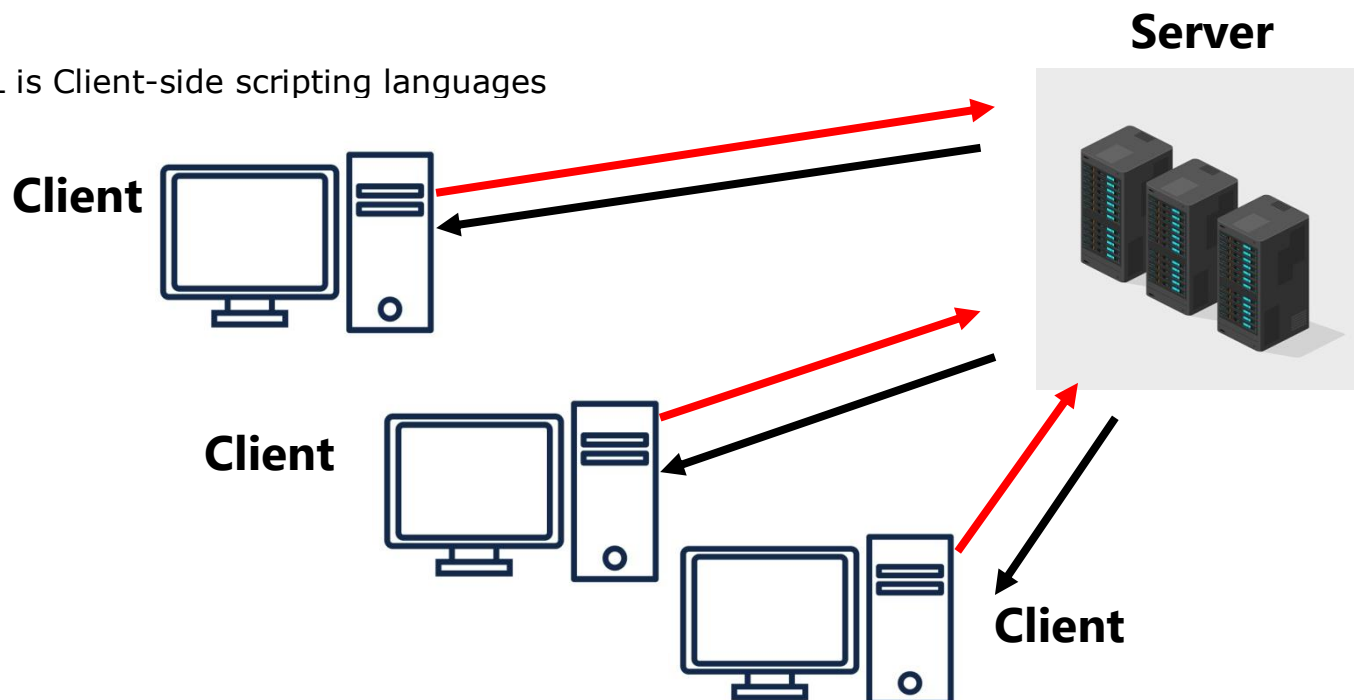
★ HTML ★

What is HTML?

1. HTML stands for Hyper Text Markup Language and **HTML** is the standard markup language for documents designed to be displayed in a web browser
2. HTML is the standard markup
3. HTML consists of a series of elements
4. HTML elements tell the browser how language for creating Web pages
5. HTML describes the structure of a Web page
6. to display the content
7. HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

How will HTML work?

HTML is Client-side scripting languages



HTML code editors

1. Notepad
2. Komodo edit
3. Notepad++
4. Microsoft Visual Studio Code
5. Sublime text
6. Atom

1). HTML Basic Elements , Attributes

A).Simple HTML Format

```
<!DOCTYPE html>

<html>

    <head>

        <title>Page Title</title>

    </head>

    <body>

        <h1>My First Heading</h1>
        <p>Paragraph</p>

    </body>
</html>
```

1. The **<!DOCTYPE html>** declaration defines that this document is an HTML5 document. **//no end tag**
2. The **<html>** element is the root element of an HTML page. **//end tag </html>**
3. The **<head>** element contains meta information about the HTML page. **//end tag </head>**
4. The **<title>** element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab). **//end tag </title>**
5. The **<body>** element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc. **//end tag </body>**
6. The **<h1>** element defines a large heading. **//end tag </h1>**
7. The **<p>** element defines a paragraph. **//end tag </p>**

B).Headings

HTML headings are defined with the **<h1>** to **<h6>** tags.

```
<body>
```

```
    <h1>This is heading 1</h1>
    <h2>This is heading 2</h2>
    <h3>This is heading 3</h3>
    <h4>This is heading 4</h4>
    <h5>This is heading 5</h5>
    <h6>This is heading 6</h6>
```

```
</body>
```

Result:-//

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

C).Paragraphs

HTML paragraphs are defined with the <p> tag:

<pre><body> <p>This is a paragraph. /<p> <p>This is another paragraph.</p> </body></pre>	<p>Result:-//</p> <p>This is a paragraph.</p> <p>This is another paragraph.</p>
---	--

D).Links


HTML links are defined with the <a> tag:

<pre><body> <h2>First Links in HTML</h2> <p>HTML links are defined with the "a" tag:</p> This is a your link </body></pre>	<p>Result:-//</p> <p>First Links in HTML</p> <p>HTML links are defined with the "a" tag:</p> <p>This is a your link</p>
<p><!--The link's destination is specified in the href attribute. --></p> <p><!--The href attribute specifies the URL of the page the link goes to.--></p>	

Href=//Hypertext REference

E).Images

HTML images are defined with the `` tag.

<pre><body> <h2>My HTML Images</h2> <p>HTML images are defined with the img tag:</p> </body></pre>	<p>Result:-//</p> <p>My HTML Images</p> <p>HTML images are defined with the img tag:</p>  <p>Carving your career</p>
<p><!--The source file (src), alternative text (alt)--></p>	

F).Empty HTML Elements

The `
` tag defines a line break, and is an empty element without a closing tag:

<pre><!DOCTYPE html> <html> <body> <p>This is a
 paragraph with a line break.</p> </body> </html></pre>	<p>Result:-//</p> <p>This is a paragraph with a line break.</p>
---	---


a).The href Attribute

The <a> tag defines a hyperlink. The href attribute specifies the URL of the page the link goes to:

<pre><!DOCTYPE html> <html> <body> <h2>The href Attribute</h2> <p>HTML links are defined with the a tag. The link address is specified in the href attribute:</p> Visit weetechinstitute </body> </html></pre>	<p>Result:--</p> <h3>The href Attribute</h3> <p>HTML links are defined with the a tag. The link address is specified in the href attribute:</p> <p>Visit WeeTech institute</p>
---	---

b).src Attribute

The tag is used to embed an image in an HTML page. The src attribute specifies the path to the image to be displayed:

<pre><!DOCTYPE html> <html> <body> <h2>The src Attribute</h2> <p>HTML images are defined with the img tag, and the filename of the image source is specified in the src attribute:</p> </body> </html></pre>	<p>Result:--</p> <h3>The src Attribute</h3> <p>HTML images are defined with the img tag, and the filename of the image source is specified in the src attribute:</p> 
---	---

<---The tag should also contain the width and height attributes, which specifies the width and height of the image (in pixels):-->

<--The required alt attribute for the tag specifies an alternate text for an image, if the image for some reason cannot be displayed. This can be due to slow connection, or an error in the src attribute, or if the user uses a screen reader.-->

c). HTML Formatting Elements

Formatting elements were designed to display special types of text:

- | | |
|------------|--------------------|
| • | - Bold text |
| • | - Important text |
| • <i> | - Italic text |
| • | - Emphasized text |
| • <mark> | - Marked text |
| • <small> | - Smaller text |
| • | - Deleted text |
| • <ins> | - Inserted text |
| • <sub> | - Subscript text |
| • <sup> | - Superscript text |

2).HTML Comment, Colors, Links, Images

a).Comment Tag

With comments you can place notifications and reminders in your HTML code:

Comments can be used to hide content.

Which can be helpful if you hide content temporarily:

You can also hide more than one line, everything between the `<!--` and the `-->` will be hidden from the display.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<!-- This is a comment -->
```

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

```
<!-- Comments are not displayed in the browser -->
```

```
</body>
```

```
</html>
```

Result:--

This is a paragraph.

b).Colors

You can set the background color for HTML elements:

You can set the color of text:

You can set the color of borders:

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1 style="background-color :DodgerBlue;">Hello World</h1>
```

```
<h3 style="color:Tomato;">Hello World</h3>
```

```
<h1 style="border: 2px solid Tomato;">Hello World</h1>
```

```
</body>
```

```
</html>
```

Result:--

Hello WeeTech

Hello WeeTech

Hello WeeTech

<--HTML colors are specified with predefined color names, or with RGB, HEX,values.-->

c).Links

Links are found in nearly all web pages. Links allow users to click their way from page to page.

By default, the linked page will be displayed in the current browser window. To change this, you must specify another target for the link.

The target attribute specifies where to open the linked document.

The target attribute can have one of the following values:

- `_self` - Default. Opens the document in the same window/tab as it was clicked
- `_blank` - Opens the document in a new window or tab
- `_parent` - Opens the document in the parent frame
- `_top` - Opens the document in the full body of the window

Absolute URLs vs. Relative URLs

Both examples above are using an absolute URL (a full web address) in the href attribute.

A local link (a link to a page within the same website) is specified with a relative URL (without the "https://www" part):

```
<!DOCTYPE html>
<html>
<body>

<h2>Absolute URLs</h2>

<a href="https://www.weetechinstitute.com/" target="_blank">
Visit weetechinstitute! </a>

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

<h2>Relative URLs</h2>

<p><a href="/user/admin/desktop/HTML.word">CSS
Tutorial</a></p>

</body>
</html>
```

Result:-//

Absolute URLs

[Visit weetechinstitute!](https://www.weetechinstitute.com/)

[Google](https://www.google.com/)

Relative URLs

[CSS Tutorial](/user/admin/desktop/HTML.word)

d).Images Syntax

The HTML tag is used to embed an image in a web page.

Images are not technically inserted into a web page; images are linked to web pages. The tag creates a holding space for the referenced image.

The tag is empty, it contains attributes only, and does not have a closing tag.

The tag has two required attributes:

src - Specifies the path to the image

alt - Specifies an alternate text for the image

You can use the style attribute to specify the width and height of an image.

Alternatively, you can use the width and height attributes:

The width, height, and style attributes are all valid in HTML.

However, we suggest using the style attribute. It prevents styles sheets from changing the size of images:

```
<!DOCTYPE html>
<html>
<head>
```

```
<style>
```

```
/* This style sets the width of all images to 100%: */
img {
  width: 100%;
}
```

```
</style>
</head>
<body>
```

```
<p>If a browser cannot find the image, it will display the alternate
text:</p>
```

```

```

```
<h2>Image Size</h2>
```

```
<p>Here we use the style attribute to specify the width and height
of an image:</p>
```

```

```

```

```

```
<h2>Images in Another Folder</h2>
```

```
<p>It is common to store images in a sub-folder. You must then
include the folder name in the src attribute:</p>
```

```

```

```
<h2>Images on Another Server</h2>
```

```

```

```
<h2>Animated Images</h2>
```

```
<p>HTML allows moving images:</p>
```

```

```

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

Result:-//

If a browser cannot find the image, it will display the alternate text:

 Flowers in india

Image Size

Here we use the style attribute to specify the width and height of an image:



Images in Another Folder

It is common to store images in a sub-folder. You must then include the folder name in the src attribute:



Images on Another Server



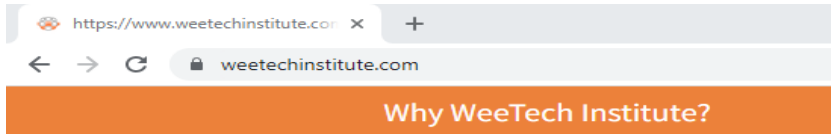
Animated Images

HTML allows moving images:



3).HTML Favicon, Tables

A favicon is a small image displayed next to the page title in the browser tab.



A). How To Add a Favicon in HTML

a `<link>` element to your "index.html" file, after the `<title>` element, like this:

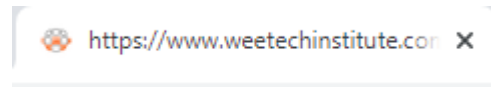
```
<html>
<head>
  <title>My Page Title</title>
  <link rel="shortcut icon"
href="https://www.weetechinstitute.com/assets/images/weetech-
fav.png">

</head>
<body>

<h1>This is a Heading</h1>
<p>This is a paragraph.</p>

</body>
```

Result:-//



B). Tables

Each table cell is defined by a `<td>` and a `</td>` tag.

Everything between `<td>` and `</td>` are the content of the table cell.

Each table row starts with a `<tr>` and end with a `</tr>` tag.

Sometimes you want your cells to be headers, in those cases use the `<th>` tag instead of the `<td>` tag:

td stands for table data.

tr stands for table row.

th Table Headers

```
<!DOCTYPE html>
<html>
<style>
table, th, td {
  border: 1px solid black;
}
</style>
<body>
<h2>TH elements define table headers</h2>
<h2>TR elements define table rows</h2>
<h2>TD elements define table cells</h2>
<table style="width:95%">
  <tr>
    <th>Harsh</th>
    <th>Savan</th>
    <th>Rahul</th>
  </tr>
  <tr>
    <td>Black</td>
    <td>Red</td>
    <td>Pink</td>
  </tr>
  <tr>
    <td>76</td>
    <td>68</td>
    <td>87</td>
  </tr>
</table>
<p>To undestand the example better, we have added borders to the
table.</p>

</body>
</html>
```

Result:-//

TH elements define table headers

TD elements define table cells

TR elements define table rows

To undestand the example better, we have added borders to the table.

Harsh	Savan	Rahul
Black	Red	Pink
76	68	87

C). Table Borders

a). Collapsed Table Borders

set the CSS border-collapse property to collapse.

```
<style>
table, th, td {
  border: 1px solid black;
}
table, th, td {
  border: 1px solid black;
  border-collapse: collapse;
}
</style>
<body>
<h2>Collapsed Borders</h2>
<p>If you want the borders to collapse into one border, add the CSS border-
collapse property.</p>

</body>
```

Result:-//

Collapsed Borders

If you want the borders to collapse into one border, add the CSS border-collapse property.

Harsh	Savan	Rahul
Black	Red	Pink
76	68	87

b).Style Table Borders

```
th, td {

  background-color: #96D4D4;

}
```


c).Dotted Table Borders

- Dotted
- Dashed
- Solid
- Double
- groove
- ridge
- inset
- outset
- none
- hidden

D).Table Sizes

Use the **style** attribute with the **width** or **height** properties to specify the size of a table, row or column.

Ex.// <th style="width: 70%">Firstname</th>

C).Table Headers

HTML tables can have headers for each column or row, or for many columns/rows.

Black	Red	Pink

10:00		
11:00		
12:00		
13:00		
14:00		
15:00		

	MON	TUE	WED	THU	FRI
10:00					
11:00					
12:00					
13:00					
14:00					

DECEMBER		

March		

D).Colspan & Rowspan

Name		

2025		
Food		

3. To create an html file as “table.html”.

Day	Seminar		
	Schedule		Topic
	Begin	End	
Monday	8:00 a.m.	5:00 p.m.	Introduction to XML
			Validity: DTD and Relax NG
Tuesday	8:00 a.m.	11:00 a.m.	XPath
	11:00 a.m.	2:00 p.m.	XSL Transformations
	2:00 p.m.	5:00 p.m.	
Wednesday	8:00 a.m.	12:00 p.m.	XSL Formatting Objects

The value of the **colspan** attribute represents the number of columns to span.

The value of the **rowspan** attribute represents the number of rows to span.

C).Table Styling

D).Table Colgroup

Use CSS to make your tables look better. This part learn in CSS class .

4).HTML Lists,Block and Inline Elements

HTML lists allow web developers to group a set of related items in lists.

A).Unordered List

An unordered list starts with the `` tag. Each list item starts with the `` tag.

The list items will be marked with bullets (small black circles) by default:

<pre><!DOCTYPE html> <html> <body> <h2>unordered HTML list</h2> Labe 1 Labe 2 Waiting Room </body> </html></pre>	<p>Result:-//</p> <h3>unordered HTML list</h3> <ul style="list-style-type: none">• Labe 1• Labe 2• Waiting Room
--	---

B).Ordered HTML List

An ordered list starts with the `` tag. Each list item starts with the `` tag.

The list items will be marked with numbers by default:

<pre><!DOCTYPE html> <html> <body> <h2>unordered HTML list</h2> Labe 1 Labe 2 Waiting Room </body> </html></pre>	<p>Result:-//</p> <h3>unordered HTML list</h3> <ol style="list-style-type: none">1. Labe 12. Labe 23. Waiting Room
---	--

C).Description Lists

HTML also supports description lists.

A description list is a list of terms, with a description of each term.

The `<dl>` tag defines the description list, the `<dt>` tag defines the term (name), and the `<dd>` tag describes each term:

<pre><!DOCTYPE html> <html> <body> <h2>A Description List</h2> <dl> <dt>Labe 1</dt> <dd>- Only C stundent</dd> <dt>Labe 2</dt> <dd>- All student allow</dd> </dl> </body> </html></pre>	<p>Result:-//</p> <h2>A Description List</h2> <p>Labe 1</p> <ul style="list-style-type: none">- Only C stundent <p>Labe 2</p> <ul style="list-style-type: none">- All student allow
--	---

{ Unordered List }

a). CSS Unordered

The CSS **list-style-type** property is used to define the style of the list item marker. It can have one of the following values:

- `<ul style="list-style-type:circle;">`
- `<ul style="list-style-type:square;">`
- `<ul style="list-style-type:none;">`

<pre><!DOCTYPE html> <html> <body> <h2>Unordered List with Circle Bullets</h2> <h2>Unordered List with Square Bullets</h2> <h2>Unordered List without Bullets</h2> <ul style="list-style-type:circle;"> Labe 1 <ul style="list-style-type:square;"> Labe 2 <ul style="list-style-type:none;"> Waiting Room </body> </html></pre>	<p>Result:-//</p> <p>Unordered List with Circle Bullets</p> <p>Unordered List with Square Bullets</p> <p>Unordered List without Bullets</p> <ul style="list-style-type: none">○ Labe 1▪ Labe 2• Waiting Room
--	---

{ Ordered List}

b).Ordered - The Type Attribute

The type attribute of the `` tag, defines the type of the list item marker:

- `<ol type="1">`
- `<ol type="A">`
- `<ol type="a">`
- `<ol type="I">`
- `<ol type="i">`

<pre><!DOCTYPE html> <html> <body> <h2>Ordered List with Numbers</h2> <ol type="1"> Labe 1 <h2>Ordered List with Letters</h2> <ol type="A"> Labe 2 <h2>Ordered List with Lowercase Letters</h2> <ol type="a"> Waiting Room 1 <h2>Ordered List with Lowercase Roman Numbers</h2> <ol type="i"> Waiting Room 2 </body> </html></pre>	<p>Result:-//</p> <p>Ordered List with Numbers</p> <p>1. Labe 1</p> <p>Ordered List with Letters</p> <p>A. Labe 2</p> <p>Ordered List with Lowercase Letters</p> <p>a. Waiting Room 1</p> <p>Ordered List with Lowercase Roman Numbers</p> <p>i. Waiting Room 2</p>
---	---

c).Control List Counting

```
<!DOCTYPE html>
<html>
<body>

<h2>The start attribute</h2>
<p>By default, an ordered list will start counting from 1. Use the start
attribute to start counting from a specified number:</p>

<ol start="50">
  <li>Labe 1</li>
  <li>Labe 2</li>
  <li>Waiting Room</li>
</ol>

<ol type="I" start="50">
  <li>Labe 1</li>
  <li>Labe 2</li>
  <li>Waiting Room</li>
</ol>

</body>
</html>
```

Result:-//

The start attribute

By default, an ordered list will start counting from 1. Use the start attribute to start counting from a specified number:

50. Labe 1
51. Labe 2
52. Waiting Room

L. Labe 1
LI. Labe 2
LII. Waiting Room

D).Block-level Elements

Two commonly used block elements are: <p> and <div>.
Here are the block-level elements in HTML:

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

C).Inline Elements

Here are the inline elements in HTML:

<time><tt><var>

< a >	< b >	< dfn >	< kbd >	< q >	< span >
< abbr >	< br >	< em >	< label >	< samp >	< strong >
< acronym >	< button >	< i >	< map >	< script >	< sub >
< big >	< cite >	< img >	< object >	< select >	< sup >
< bdo >	< code >	< input >	< output >	< small >	< textarea >

D).<div> Element

The <div> element is often used as a container for other HTML elements.

The <div> element has no required attributes, but style, class and id are common.

When used together with CSS, the <div> element can be used to style blocks of content:

```
<!DOCTYPE html>
<html>
<body>

<div style="background-color:black;color:white;padding:20px;">
  <h2>London</h2>
  <p> is simply dummy text of the printing and typesetting industry. Lorem
Ipsum has</p>
  <p>belief, Lorem Ipsum is not simply random text. It has roots in a piece
of classical Latin literature from 45 BC, making it over 2000 years old.</p>
</div>

</body>
</html>
```

Result:-//

London

is simply dummy text of the printing and typesetting industry. Lorem Ipsum has

belief, Lorem Ipsum is not simply random text. It has roots in a

E). Element

The element is an inline container used to mark up a part of a text, or a part of a document.

The element has no required attributes, but style, class and id are common.

When used together with CSS, the element can be used to style parts of the text:

```
<!DOCTYPE html>
<html>
<body>

<h1>The span element</h1>

<p>My mother has <span style="color:blue;font-weight:bold">blue</span>
eyes and my father has <span style="color:green;font-weight:bold">dark
green</span> eyes.</p>

</body>
</html>
```

Result:-//

The span element

My mother has **blue** eyes and my father has **dark green** eyes.

5).HTML class, id, iframes

A).The **class** Attribute

The class attribute is often used to point to a class name in a style sheet. It can also be used by a JavaScript to access and manipulate elements with the specific class name.

In the following example we have three <div> elements with a class attribute with the value of "city". All of the three <div> elements will be styled equally according to the .city style definition in the head section:

<pre><!DOCTYPE html> <html> <head> <style> .city { background-color: tomato; color: white; border: 2px solid black; margin: 20px; padding: 20px; } .note { font-size: 120%; color: red; } </style> </head> <body> <div class="city"> <h2>London</h2> <p>London is the capital of England.</p> </div> <div class="city"> <h2>Paris</h2> <p>Paris is the capital of France.</p> </div> <div class="city"> <h2>Tokyo</h2> <p>Tokyo is the capital of Japan.</p> </div> <h1>My Important Heading</h1> <p>This is some important text.</p> </body> </html></pre>	<p>Result:-//</p> <div>London London is the capital of England.</div> <div>Paris Paris is the capital of France.</div> <div>Tokyo Tokyo is the capital of Japan.</div> <div>My Important Heading This is some important text.</div>
--	--

B). **id** Attribute

The HTML id attribute is used to specify a unique id for an HTML element.

You cannot have more than one element with the same id in an HTML document.

<pre><!DOCTYPE html> <html> <head> <style> #myHeader { background-color: lightblue; color: black; padding: 40px; text-align: center; } </style> </head> <body> <h2>The id Attribute</h2> <p>Use CSS to style an element with the id "myHeader":</p> <h1 id="myHeader">My Header</h1> </body> </html></pre>	<p>Result:-//</p> <div>The id Attribute</div> <p>Use CSS to style an element with the id "myHeader":</p> <div>My Header</div>
---	---

a).Bookmarks with ID and Links

<pre><!DOCTYPE html> <html> <body> <p>Jump to HarshData</p> <p>Jump to SavanData</p> <h2>Rahul</h2> <p>This chapter explains ba bla bla</p> <h2>Sahista</h2> <p>This chapter explains ba bla bla</p> <h2 id="S3">Harsh</h2> <p>This chapter explains ba bla bla</p> <h2>Sandip</h2> <p>This chapter explains ba bla bla</p> <h2 id="S5">Savan</h2> <p>This chapter explains ba bla bla</p> <h2>Saba</h2> <p>This chapter explains ba bla bla</p> : : : <h2>Saba</h2> <p>This chapter explains ba bla bla</p> </body> </html></pre>	<p>Result:-//</p> <p>Jump to HarshData</p> <p>Jump to SavanData</p> <p>Rahul This chapter explains ba bla bla</p> <p>Sahista This chapter explains ba bla bla</p> <p>Harsh This chapter explains ba bla bla</p> <p>Sandip This chapter explains ba bla bla</p> <p>Savan This chapter explains ba bla bla</p>
--	--

C).Iframes

An HTML iframe is used to display a web page within a web page.

a).Iframe Syntax

he HTML <iframe> tag specifies an inline frame.

```
<iframe src="url" title="description"></iframe>
```

b).Set Height and Width

<pre><!DOCTYPE html> <html> <body> <h2>HTML Iframes</h2> <p>You can use the height and width attributes to specify the size of the iframe:</p> <iframe src="demo_iframe.htm" height="200" width="300" title="Iframe Example"></iframe> </body> </html></pre>	<p>Result:-//</p> <h2>HTML Iframes</h2> <p>You can use the height and width attributes to specify the size of the iframe:</p> <div><p>This page is displayed in an iframe</p></div>
---	---

C).Target for a Link

```
<!DOCTYPE html>
<html>
<body>

<h2>Iframe - Target for a Link</h2>

<iframe src="demo_iframe.htm" name="iframe_a" height="300px"
width="100%" title="Iframe Example"></iframe>

<p><a href="https://www.weetechinstitute.com"
target="iframe_a">weetechinstitute.com</a></p>

<p>When the target attribute of a link matches the name of an iframe, the
link will open in the iframe.</p>

</body>
</html>
```

Result:-//

Iframe - Target for a Link

**This page is
displayed in an
iframe**

[weetech](https://www.weetechinstitute.com)

When the target attribute of a link matches the name of an iframe, the link will open in the iframe.

6).HTML JavaScript,Head Element, Layout

A).<script> Tag

The HTML <script> tag is used to define a client-side script (JavaScript).

The <script> element either contains script statements, or it points to an external script file through the src attribute.

Common uses for JavaScript are image manipulation, form validation, and dynamic changes of content.

To select an HTML element, JavaScript most often uses the document.getElementById() method.

This JavaScript example writes "Hello JavaScript!" into an HTML element with id="demo":

```
<!DOCTYPE html>
<html>
<body>

<h2>Use JavaScript to Change Text</h2>
<p>This example writes "Hello JavaScript!" into an HTML element with
id="demo":</p>

<p id="demo"></p>

<script>
document.getElementById("demo").innerHTML = "Hello JavaScript!";
</script>

</body>
</html>
```

Result:-//

Use JavaScript to Change Text

This example writes "Hello JavaScript!" into an HTML element with id="demo":

Hello JavaScript!

B).A Taste of JavaScript

```
<!DOCTYPE html>
<html>
<body>

<p>JavaScript can change the content of an HTML element:</p>
<button type="button" onclick="myFunction()">Click Me!</button>

<p id="demo">This is a demonstration.</p>

<script>
function myFunction() {
  document.getElementById("demo").innerHTML = "Hello JavaScript!";
}
</script>

<h1>My First JavaScript</h1>

<p id="demo">JavaScript can change the style of an HTML element.</p>

<script>
function myFunction() {
  document.getElementById("demo").style.fontSize = "25px";
  document.getElementById("demo").style.color = "red";
  document.getElementById("demo").style.backgroundColor = "yellow";
}
</script>
<button type="button" onclick="myFunction()">Click Me!</button>

</body>
</html>
```

Result:-//

My First JavaScript

JavaScript can change the content of an HTML element:

Click Me!

This is a demonstration.

My First JavaScript

JavaScript can change the style of an HTML element.

Click Me!

C).Head Element

The HTML **<head>** element is a container for the following elements: **<title>**, **<style>**, **<meta>**, **<link>**, **<script>**, and **<base>**.

.

a).<meta> Element

Define the character set used:

<meta charset="UTF-8">

Define keywords for search engines:

<meta name="keywords" content="HTML, CSS, JavaScript">

Define a description of your web page:

<meta name="description" content="Free Web tutorials">

Define the author of a page:

<meta name="author" content="Harsh Savani">

Refresh document every 30 seconds:

<meta http-equiv="refresh" content="25">

Setting the viewport to make your website look good on all devices:

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

b).<script> Element

The **<script>** element is used to define client-side JavaScripts.

```
<!DOCTYPE html>
<html>
<head>
  <title>Page Title</title>
  <script>
    function myFunction() {
      document.getElementById("demo").innerHTML = "Hello JavaScript!";
    }
  </script>
</head>
<body>

<h1>My Web Page</h1>
<p id="demo">A Paragraph</p>
<button type="button" onclick="myFunction()">Try it</button>

</body>
</html>
```

Result:-//

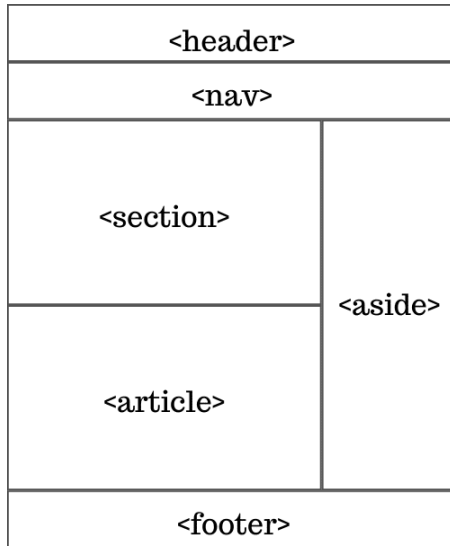
My Web Page

A Paragraph

Try it

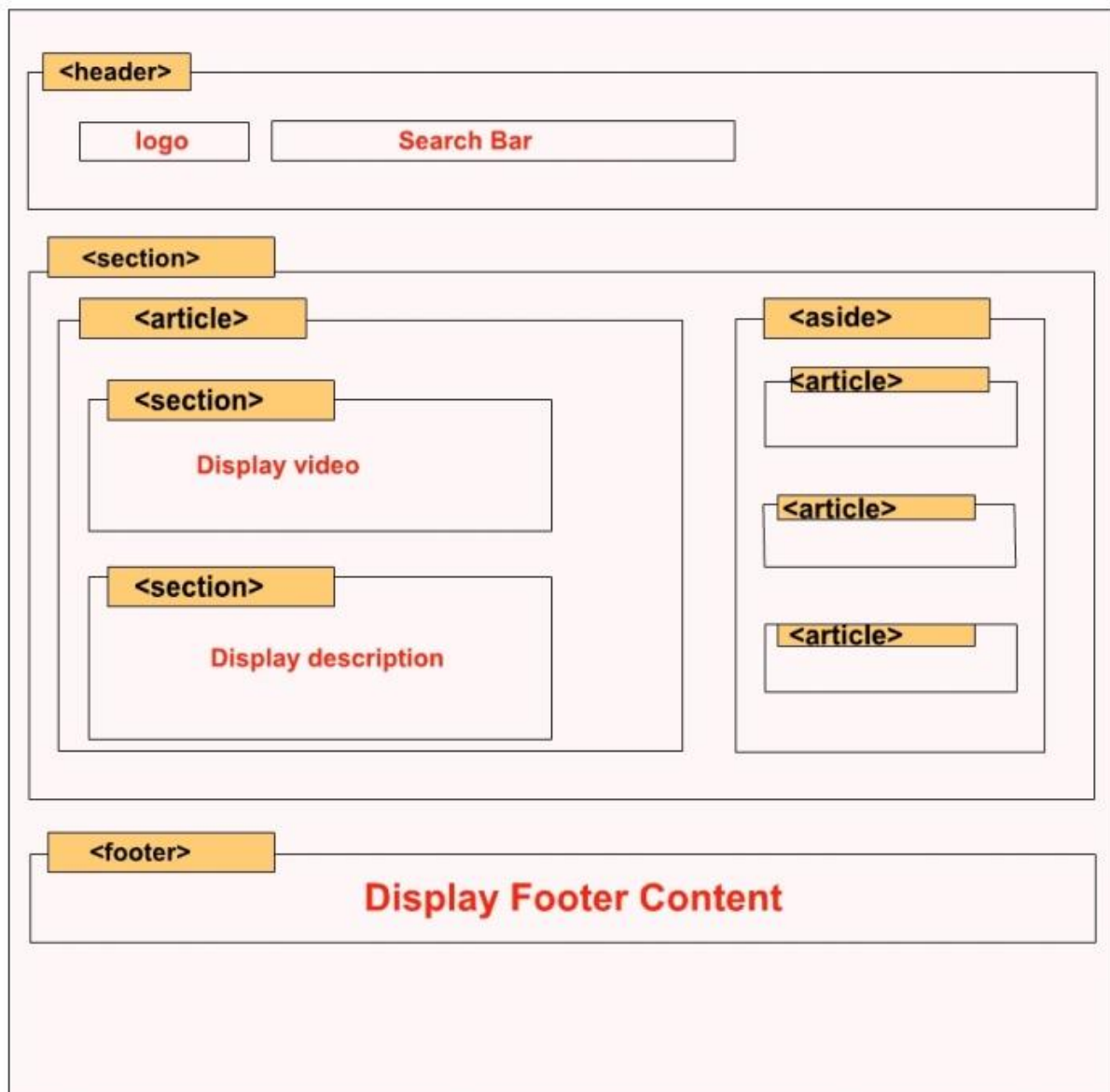
D).Layout Elements

HTML has several semantic elements that define the different parts of a web page:



- <header> - Defines a header for a document or a section
- <nav> - Defines a set of navigation links
- <section> - Defines a section in a document
- <article> - Defines an independent, self-contained content
- <aside> - Defines content aside from the content (like a sidebar)
- <footer> - Defines a footer for a document or a section
- <details> - Defines additional details that the user can open and close on demand
- <summary> - Defines a heading for the <details> element

You can read more about semantic elements in our [HTML Semantics chapter](#).



7).HTML Responsive Web Design, Computer

, Semantic Elements

A).Viewport, Images

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

- If the CSS width property is set to 100%, the image will be responsive and scale up and down:

```

```

- If the max-width property is set to 100%, the image will scale down if it has to, but never scale up to be larger than its original size:

```

```

- The HTML <picture> element allows you to define different images for different browser window sizes.
- Resize the browser window to see how the image below change depending on the width:

```
<picture>
  <source srcset="img_smallflower.jpg" media="(max-width: 600px)">
  <source srcset="img_flowers.jpg" media="(max-width: 1500px)">
  <source srcset="flowers.jpg">
  
</picture>
```

a).Text Size

- The text size can be set with a "vw" unit, which means the "viewport width".

```
<body>
  <h1 style="font-size:10vw;">Responsive Text</h1>
  <p style="font-size:5vw;">Resize the browser window to see how the text size scales.</p>
  <p style="font-size:5vw;">Use the "vw" unit when sizing the text. 10vw will set the size to 10% of the viewport width.</p>
  <p>Viewport is the browser window size. 1vw = 1% of viewport width. If the viewport is 50cm wide, 1vw is 0.5cm.</p>
</body>
```

b).Media Queries

In addition to resize text and images, it is also common to use media queries in responsive web pages. With media queries you can define completely different styles for different browser sizes.

```
<body>
  <h2>Media Queries</h2>
  <p>Resize the browser window.</p>

  <p>Make sure you reach the breakpoint at 800px when resizing this frame.</p>

  <div class="left">
    <p>Left Menu</p>
  </div>

  <div class="main">
    <p>Main Content</p>
  </div>

  <div class="right">
    <p>Right Content</p>
  </div>

</body>
```

Result:-//

Media Queries

Resize the browser window.

Make sure you reach the breakpoint at 800px when resizing this frame.

Left Menu

Main Content

Right Content

```
* {
  box-sizing: border-box;
}

.left {
  background-color: #2196F3;
  padding: 20px;
  float: left;
  width: 20%; /* The width is 20%, by default */
}

.main {
  background-color: #f1f1f1;
  padding: 20px;
  float: left;
  width: 60%; /* The width is 60%, by default */
}
```

```
right {
  background-color: #04AA6D;
  padding: 20px;
  float: left;
  width: 20%; /* The width is 20%, by default */
}

/* Use a media query to add a break point at 800px: */
@media screen and (max-width: 800px) {
  .left, .main, .right {
    width: 100%; /* The width is 100%, when the viewport is 800px or
smaller */
  }
}
</style>
```

A).Computer

a).Code Elements

HTML contains several elements for defining user input and computer code.

```
<!DOCTYPE html>
<html>
<body>

<h2>Computer Code</h2>
<p>Some programming code:</p>

<code>
x = 5;
y = 6;
z = x + y;
</code>

</body>
</html>
```

Computer Code

Some programming code:

```
x = 5; y = 6; z = x + y;
```

b).<kbd> For Keyboard Input

The HTML <kbd> element is used to define keyboard input. The content inside is displayed in the browser's default monospace font.

```
<!DOCTYPE html>
<html>
<body>

<h2>The kbd Element</h2>

<p>The kbd element is used to define keyboard input:</p>

<p>Save the document by pressing <kbd>Ctrl + S</kbd></p>

</body>
</html>
```

The kbd Element

The kbd element is used to define keyboard input:

Save the document by pressing `Ctrl + S`

c).<samp> For Program Output

The HTML <samp> element is used to define sample output from a computer program. The content inside is displayed in the browser's default monospace font.

```
<p>Message from my computer:</p>
<p><samp>File not found.<br>Press F1 to continue</samp></p>
```

Message from my computer:

```
File not found.
Press F1 to continue
```

d).<var> For Variables

The HTML <var> element is used to define a variable in programming or in a mathematical expression. The content inside is typically displayed in *italic*.

<p>The area of a triangle is: 1/2 x <var>b</var> x <var>h</var>, where <var>b</var> is the base, and <var>h</var> is the vertical height.</p>

The area of a triangle is: 1/2 x *b* x *h*, where *b* is the base, and *h* is the vertical height.

B).Semantic

A semantic element clearly describes its meaning to both the browser and the developer.

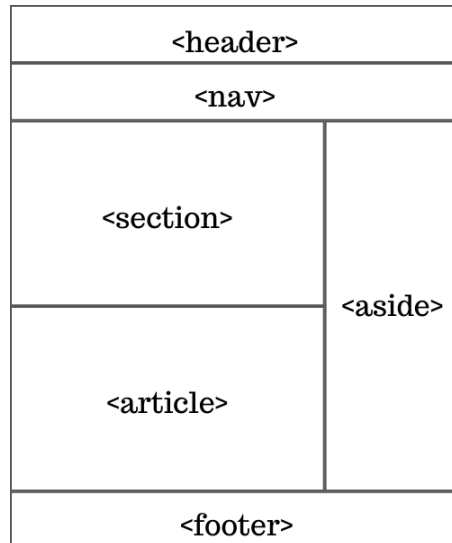
Examples of **non-semantic** elements: <div> and - Tells nothing about its content.

Examples of **semantic** elements: <form>, <table>, and <article> - Clearly defines its content.

Many web sites contain HTML code like: <div id="nav"> <div class="header"> <div id="footer"> to indicate navigation, header, and footer.

In HTML there are some semantic elements that can be used to define different parts of a web page:

- <article>
- <aside>
- <details>
- <figcaption>
- <figure>
- <footer>
- <header>
- <main>
- <mark>
- <nav>
- <section>
- <summary>
- <time>



a).<section> Element

The <section> element defines a section in a document.

According to W3C's HTML documentation: "A section is a thematic grouping of content, typically with a heading."

Examples of where a <section> element can be used:

- Chapters
- Introduction
- News items
- Contact information

A web page could normally be split into sections for introduction, content, and contact information.

<section>

<h1>WWF</h1>

<p>The World Wide Fund for Nature (WWF) is an international organization working on issues regarding the conservation, research and restoration of the environment, formerly named the World Wildlife Fund. WWF was founded in 1961.</p>

</section>

WeeTech

The World Wide Fund for Nature (WWF) is an international organization working on issues regarding the conservation, research and restoration of the environment, formerly named the World Wildlife Fund. WWF was founded in 1961.

b).<article> Element

The <article> element specifies independent, self-contained content.

An article should make sense on its own, and it should be possible to distribute it independently from the rest of the web site.

Examples of where the <article> element can be used:

- Forum posts
- Blog posts
- User comments
- Product cards
- Newspaper articles

<pre><article> <h2>Microsoft Edge</h2> <p>Microsoft Edge is a web browser developed by Microsoft, released in 2015. Microsoft Edge replaced Internet Explorer.</p> </article></pre>	<h2>Microsoft Edge</h2> <p>Microsoft Edge is a web browser developed by Microsoft, released in 2015. Microsoft Edge replaced Internet Explorer.</p>
--	---

c).<header> Element

The <header> element represents a container for introductory content or a set of navigational links.

A <header> element typically contains:

- one or more heading elements (<h1> - <h6>)
- logo or icon
- authorship information

Note: You can have several <header> elements in one HTML document. However, <header> cannot be placed within a <footer>, <address> or another <header> element.

<pre><article> <header> <h1>What Does WWF Do?</h1> <p>WWF's mission:</p> </header> <p>WWF's mission is to stop the degradation of our planet's natural environment, and build a future in which humans live in harmony with nature.</p> </article></pre>	<h2>What Does WWF Do?</h2> <p>WWF's mission:</p> <p>WWF's mission is to stop the degradation of our planet's natural environment, and build a future in which humans live in harm</p>
--	---

d).<footer> Element

The <footer> element defines a footer for a document or section.

A <footer> element typically contains:

- authorship information
- copyright information
- contact information
- sitemap
- back to top links
- related documents

You can have several <footer> elements in one document.

e).<nav> Element

The <nav> element defines a set of navigation links.

<pre><nav> HTML CSS JavaScript jQuery </nav></pre>	HTML CSS JavaScript jQuery
---	--

f).<aside> Element

he <aside> element defines some content aside from the content it is placed in (like a sidebar).
The <aside> content should be indirectly related to the surrounding content.

<pre><aside> <p>The Epcot center is a theme park at Walt Disney World Resort featuring exciting attractions, international pavilions, award-winning fireworks and seasonal special events.</p> </aside></pre>	<h2>Microsoft Edge</h2> <p>Microsoft Edge is a web browser developed by Microsoft, released in 2015. Microsoft Edge replaced Internet Explorer.</p>
---	---

g).<figure> and <figcaption> Elements

The <figure> tag specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.

The <figcaption> tag defines a caption for a <figure> element. The <figcaption> element can be placed as the first or as the last child of a <figure> element.

The element defines the actual image/illustration.

<pre><!DOCTYPE html> <html> <body> <h2>Places to Visit</h2> <p>Puglia's most famous sight is the unique conical houses (Trulli) found in the area around Alberobello, a declared UNESCO World Heritage Site.</p> <figure> <figcaption>Fig.1 - Trulli, Puglia, Italy.</figcaption> </figure> </body> </html></pre>	<h2>Places to Visit</h2> <p>Puglia's most famous sight is the unique conical houses (Trulli) found in the area around Alberobello, a declared UNESCO World Heritage Site.</p> <p> Fig.1 - Trulli, Puglia, Italy.</p>
--	--

8).HTML Style Guide

1.Always Declare Document Type

2.Use Lowercase Element Names

HTML allows mixing uppercase and lowercase letters in element names. However, we recommend using lowercase element names, because:

- Mixing uppercase and lowercase names looks bad
- Developers normally use lowercase names
- Lowercase looks cleaner
- Lowercase is easier to write

3.Close All HTML Elements

In HTML, you do not have to close all elements (for example the `<p>` element).

4.Lowercase Attribute Names

5.Quote Attribute Values

HTML allows attribute values without quotes.

However, we recommend quoting attribute values, because:

- Developers normally quote attribute values
- Quoted values are easier to read
- You MUST use quotes if the value contains spaces

6.Specify alt, width, and height for Images

Always specify the alt attribute for images. This attribute is important if the image for some reason cannot be displayed.

Also, always define the width and height of images. This reduces flickering, because the browser can reserve space for the image before loading.

7.Spaces and Equal Signs

HTML allows spaces around equal signs. But space-less is easier to read and groups entities better together.

8.Avoid Long Code Lines

When using an HTML editor, it is NOT convenient to scroll right and left to read the HTML code.

9.Blank Lines and Indentation

Do not add blank lines, spaces, or indentations without a reason.

For readability, add blank lines to separate large or logical code blocks.

For readability, add two spaces of indentation. Do not use the tab key.

10.Never Skip the <title> Element

The `<title>` element is required in HTML.

The contents of a page title is very important for search engine optimization (SEO)! The page title is used by search engine algorithms to decide the order when listing pages in search results.

The `<title>` element:

- defines a title in the browser toolbar
- provides a title for the page when it is added to favorites
- displays a title for the page in search-engine results

11.Add the lang Attribute

You should always include the lang attribute inside the `<html>` tag, to declare the language of the Web page. This is meant to assist search engines and browsers.

12.The Viewport

The viewport is the user's visible area of a web page. It varies with the device - it will be smaller on a mobile phone than on a computer screen.

You should include the following <meta> element in all your web pages:

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

This gives the browser instructions on how to control the page's dimensions and scaling.

The width=device-width part sets the width of the page to follow the screen-width of the device (which will vary depending on the device).

The initial-scale=1.0 part sets the initial zoom level when the page is first loaded by the browser.

13.Comments

Short comments should be written on one line, like this:

14.Using Style Sheets

Use simple syntax for linking to style sheets (the type attribute is not necessary):

- Place the opening bracket on the same line as the selector
- Use one space before the opening bracket
- Use two spaces of indentation
- Use semicolon after each property-value pair, including the last
- Only use quotes around values if the value contains spaces
- Place the closing bracket on a new line, without leading spaces

15.Use Lower Case File Names

16.File Extensions

HTML files should have a .html extension (.htm is allowed).

CSS files should have a .css extension.

JavaScript files should have a .js extension.

17.Differences Between .htm and .html?

There is no difference between the .htm and .html file extensions!

Both will be treated as HTML by any web browser and web server.

18.Default Filenames

"index.html", "index.htm", "default.html", or "default.htm".

9).HTML Character Entities, Symbol Entities, Emojis, URL Encode

A).Character Entities

a).Character Entities

Result	Description	Entity Name	Entity Number
	non-breaking space	 	
>	greater than	>	>
&	ampersand	&	&
"	double quotation mark	"	"
<	less than	<	<
'	single quotation mark (apostrophe)	'	'
¢	cent	¢	¢
£	pound	£	£
€	euro	€	€
©	copyright	©	©
¥	yen	¥	¥
®	registered trademark	®	®

[Click](#)
Reserved characters in HTML must be replaced with character entities.

b).Diacritical Marks

Mark	Character	Construct	Result
`	a	à	à
'	a	á	á
^	a	â	â
~	a	ã	ã
`	O	Ò	Ò
'	O	Ó	Ó
^	O	Ô	Ô
~	O	Õ	Õ

[click](#)

A diacritical mark is a "glyph" added to a letter.
Some diacritical marks, like grave (`) and acute (') are called accents.

B).Symbols

Symbols that are not present on your keyboard can also be added by using entities.

a).Symbol Entities

HTML entities were described in the previous chapter.

Many mathematical, technical, and currency symbols, are not present on a normal keyboard.

To add such symbols to an HTML page, you can use the entity name or the entity number (a decimal or a hexadecimal reference) for the symbol.

<pre><p>This is result &euro;</p> <p>This is result &#8364;</p> <p>This is result &#x20AC;</p></pre>	Thia is result € Thia is result € Thia is result €
--	--

b).Symbols Supported by HTML

Char	Number	Entity	Description
∂	∂	∂	PARTIAL DIFFERENTIAL
∃	∃	∃	THERE EXISTS
∇	∇	∇	NABLA
∈	∈	∈	ELEMENT OF
∉	∉	∉	NOT AN ELEMENT OF
⊃	∋	∋	CONTAINS AS MEMBER
∏	∏	∏	N-ARY PRODUCT
∀	∀	∀	FOR ALL
Σ	∑	∑	N-ARY SUMMATION
∅	∅	∅	EMPTY SETS

[click](#)

c).Letters Supported by HTML

Char	Number	Entity	Description
Α	Α	Α	GREEK CAPITAL LETTER ALPHA
Ζ	Ζ	Ζ	GREEK CAPITAL LETTER ZETA
Γ	Γ	Γ	GREEK CAPITAL LETTER GAMMA
Δ	Δ	Δ	GREEK CAPITAL LETTER DELTA
Β	Β	Β	GREEK CAPITAL LETTER BETA
Ε	Ε	Ε	GREEK CAPITAL LETTER EPSILON

[Click](#)

d).Entities Supported by HTML

Char	Number	Entity	Description
♦	♦	♦	BLACK DIAMOND SUIT
©	©	©	COPYRIGHT SIGN
€	€	€	EURO SIGN
←	←	←	LEFTWARDS ARROW
®	®	®	REGISTERED SIGN
™	™	™	TRADEMARK
↑	↑	↑	UPWARDS ARROW
↓	↓	↓	DOWNWARDS ARROW
♠	♠	♠	BLACK SPADE SUIT
♣	♣	♣	BLACK CLUB SUIT
→	→	→	RIGHTWARDS ARROW
♥	♥	♥	BLACK HEART SUIT

[Click](#)

C).Emojis in HTML





Emojis are characters from the UTF-8 character set
emojis look like images, or icons, but they are not.
They are letters (characters) from the UTF-8 (Unicode) character set.

a). Emoji Characters

Emojis are also characters from the UTF-8 alphabet:

- 😊 is 128516
- 😍 is 128525
- ❤️ is 128151

<pre><!DOCTYPE html> <html> <head> <meta charset="UTF-8"> </head> <body> <h1>My First Emoji</h1> <p>&#128512;</p> </body> </html></pre>	<h2>My First Emoji</h2> 
--	--

<pre><!DOCTYPE html> <html> <head> <meta charset="UTF-8"> </head> <body> <h1>Sized Emojis</h1> <p style="font-size:48px"> &#128512; &#128516; &#128525; &#128151; </p> </body> </html></pre>	<h2>Sized Emojis</h2>    
---	---

[Click to More Emojis](#)

D).Uniform Resource Locators

A URL is another word for a web address.

Web browsers request pages from web servers by using a URL.

A Uniform Resource Locator (URL) is used to address a document (or other data) on the web.

A web address like <https://www.weetechinstitute.com/html/#> follows these syntax rules:

- scheme - defines the type of Internet service (most common is http or https)
- prefix - defines a domain prefix (default for http is www)
- domain - defines the Internet domain name (like weetechinstitute.com)
- port - defines the port number at the host (default for http is 80)
- path - defines a path at the server (If omitted: the root directory of the site)
- filename - defines the name of a document or resource

Scheme	Short for	Used for
http	HyperText Transfer Protocol	Common web pages. Not encrypted
https	Secure HyperText Transfer Protocol	Secure web pages. Encrypted
ftp	File Transfer Protocol	Downloading or uploading files
file		A file on your computer

ADVANCE HTML

1).HTML Forms, Form Attributes ,Form Elements

An HTML form is used to collect user input. The user input is most often sent to a server for processing.

a). <form> Element

The HTML <form> element is used to create an HTML form for user input

The <form> element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc.

b).<input> Element

The HTML <input> element is the most used form element.

An <input> element can be displayed in many ways, depending on the type attribute.

<input type="text">	Displays a single-line text input field
<input type="radio">	Displays a radio button (for selecting one of many choices)
<input type="checkbox">	Displays a checkbox (for selecting zero or more of many choices)
<input type="submit">	Displays a submit button (for submitting the form)
<input type="button">	Displays a clickable button

B). Form Elements

a).<form> Elements

- <input>
- <label>
- <select>
- <textarea>
- <button>
- <fieldset>
- <legend>
- <datalist>
- <output>
- <option>
- <optgroup>

1). <input> Element

One of the most used form element is the <input> element.

The <input> element can be displayed in several ways, depending on the type attribute.

2). <label> Element

The <label> element defines a label for several form elements.

The <label> element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.

The <label> element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the <label> element, it toggles the radio button/checkbox.

The for attribute of the <label> tag should be equal to the id attribute of the <input> element to bind them together.

3). <select> Element

```
<form action="">
<label for=" Bikes ">Choose a Bikes :</label>
<select id="Bikes" name=" Bikes ">
  <option value="KTM">KTM</option>
  <option value="HONDA">HONDA</option>
  <option value="OLA">OLA</option>
</select>
<input type="submit">
</form>
```

Choose a Bikes :

- The <option> elements defines an option that can be selected.
- By default, the first item in the drop-down list is selected.
- To define a pre-selected option, add the selected attribute to the option:

size="3"

multiple

4). <textarea> Element

The <textarea> element defines a multi-line input field (a text area):

```
<form action="">
<textarea name="message" rows="10" cols="30">The cat was playing
in the garden.</textarea>
<br><br>
<input type="submit">
</form>
```

USE. VS CORD

<textarea name="message" style="width:200px; height:600px;">

5). <button> Element

The <button> element defines a clickable button:

```
<h2>The button Element</h2>

<button type="button" onclick="alert('WeeTech ')">Click Me!</button>
```

The button Element

Click Me!

6). <fieldset> and <legend> Elements

The <fieldset> element is used to group related data in a form.

The <legend> element defines a caption for the <fieldset> element.

```
<form action="">
<fieldset>
  <legend>Personalia:</legend>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname"
value="Doe"><br><br>
  <input type="submit" value="Submit">
</fieldset>
</form>
```

USE. VS CORD

7). <datalist> Element

The <datalist> element specifies a list of pre-defined options for an <input> element.

Users will see a drop-down list of the pre-defined options as they input data.

The list attribute of the <input> element, must refer to the id attribute of the <datalist> element.

```
<form action="">
  <input list="student" name="student">
  <datalist id="student">
    <option value="HarshSavani">
    <option value="RahulSolanki">
    <option value="BhavtickSonaviya">
    <option value="ShehkSaba">
    <option value="SahistaPatel">
  </datalist>
  <input type="submit">
</form>
```

USE. VS CORD

8). <output> Element

The <output> element represents the result of a calculation (like one performed by a script).

```
<form action="/action_page.php"
oninput="x.value=parseInt(a.value)+parseInt(b.value)">
  0
  <input type="range" id="a" name="a" value="50">
  100 +
  <input type="number" id="b" name="b" value="50">
  =
  <output name="x" for="a b"></output>
  <br><br>
  <input type="submit">
</form>
```

USE. VS CORD

c). <label>Text Element

- Notice the use of the <label> element in the example above.
- The <label> tag defines a label for many form elements.
- The <label> element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.
- The <label> element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the <label> element, it toggles the radio button/checkbox.
- The for attribute of the <label> tag should be equal to the id attribute of the <input> element to bind them together..

```
<form>
  <label for="fname">First name:</label> <br>
  <input type="text" id="fname" name="fname" value="John"> <br>
  <label for="lname">Last name:</label> <br>
  <input type="text" id="lname" name="lname" value="Doe">
</form>
```

<p>Note that the form itself is not visible.</p>

<p>Also note that the default width of text input fields is 20 characters.</p>

First name:

John

Last name:

Doe

Note that the form itself is not visible.
Also note that the default width of text input fields is 20 characters.

d).Radio Buttons

The <input type="radio"> defines a radio button.

```
<form>
  <input type="radio" id="yes" name="yes_vv" value="yes">
  <label for="yes">Yes</label> <br>
  <input type="radio" id="no" name="yes_vv" value="no">
  <label for="no">No</label> <br>
</form>
```



Yes



No

e). Checkboxes

```
<form action="#">
  <input type="checkbox" id="Adharcad" name="adharcad"
value="address">
  <label for="Adharcad"> Adhar Card</label><br>
  <input type="checkbox" id="pancard" name="pancard"
value="address">
  <label for="pancard"> Pancard</label><br>
  <input type="submit" value="Submit">
</form>
```

| | |
|---------------------------------------|------------|
| <input type="checkbox"/> | Adhar Card |
| <input type="checkbox"/> | Pancard |
| <input type="submit" value="Submit"/> | |

f). Submit Button

- The `<input type="submit">` defines a button for submitting the form data to a form-handler.
- The form-handler is typically a file on the server with a script for processing input data.
- The form-handler is specified in the form's action attribute.

A). Form Attributes

a). Action Attribute

The action attribute defines the action to be performed when the form is submitted.

Usually, the form data is sent to a file on the server when the user clicks on the submit button.

In the example below, the form data is sent to a file called "action_page.php". This file contains a server-side script that handles the form data:

```
<form action="/action_page.php">
```

b). Target Attribute

```
<form action="/action_page.php" target="_blank">
```

c). Method Attribute

The method attribute specifies the HTTP method to be used when submitting the form data.

The form-data can be sent as URL variables (with `method="get"`) or as HTTP post transaction (with `method="post"`).

The default HTTP method when submitting form data is GET.

```
<form action="/action_page.php" method="get">
```

```
<form action="/action_page.php" method="post">
```

Notes on GET:

- Appends the form data to the URL, in name/value pairs
- NEVER use GET to send sensitive data! (the submitted form data is visible in the URL!)
- The length of a URL is limited (2048 characters)
- Useful for form submissions where a user wants to bookmark the result
- GET is good for non-secure data, like query strings in Google

Notes on POST:

- Appends the form data inside the body of the HTTP request (the submitted form data is not shown in the URL)
- POST has no size limitations, and can be used to send large amounts of data.
- Form submissions with POST cannot be bookmarked

d).Autocomplete Attribute

The autocomplete attribute specifies whether a form should have autocomplete on or off.
When autocomplete is on, the browser automatically complete values based on values that the user has entered before.

```
<form action="/action_page.php" autocomplete="on">
```

e). Novalidate Attribute

The novalidate attribute is a boolean attribute.
When present, it specifies that the form-data (input) should not be validated when submitted.

```
<form action="/action_page.php" novalidate>
```

B).<form> Elements

The HTML <form> element can contain one or more of the following form elements:

- <input>
- <label>
- <select>
- <textarea>
- <button>
- <fieldset>
- <legend>
- <datalist>
- <output>
- <option>
- <optgroup>

a). <input> Element

One of the most used form element is the <input> element.
The <input> element can be displayed in several ways, depending on the type attribute.

```
<input type="text" id="fname" name="fname">
```

b). <label> Element

The <label> element defines a label for several form elements.
The <label> element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.
The <label> element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the <label> element, it toggles the radio button/checkbox.
The for attribute of the <label> tag should be equal to the id attribute of the <input> element to bind them together.

c).<select> Element

The <select> element defines a drop-down list:

```
<select id="cars" name="cars">  
  <option value="volvo">Volvo</option>
```

d). <textarea> Element

The rows attribute specifies the visible number of lines in a text area.

The cols attribute specifies the visible width of a text area.

e).<button> Element

```
<button type="button" onclick="alert('Hello World!')">Click Me</button>
```

2).HTML Input Types, Input Attributes ,form* Attributes

A) Input Types

Here are the different input types you can use in HTML:

- `<input type="button">`
- `<input type="checkbox">`
- `<input type="color">`
- `<input type="date">`
- `<input type="datetime-local">`
- `<input type="email">`
- `<input type="file">`
- `<input type="hidden">`
- `<input type="image">`
- `<input type="month">`
- `<input type="number">`
- `<input type="password">`
- `<input type="radio">`
- `<input type="range">`
- `<input type="reset">`
- `<input type="search">`
- `<input type="submit">`
- `<input type="tel">`
- `<input type="text">`
- `<input type="time">`
- `<input type="url">`
- `<input type="week">`

B). Input Attributes

a). readonly Attribute

The input readonly attribute specifies that an input field is read-only.

A read-only input field cannot be modified (however, a user can tab to it, highlight it, and copy the text from it).

The value of a read-only input field will be sent when submitting the form!

```
<input type="text" id="fname" name="fname" value="John" readonly><br>
```

b). disabled Attribute

The input disabled attribute specifies that an input field should be disabled.

A disabled input field is unusable and un-clickable.

The value of a disabled input field will not be sent when submitting the form!

```
<input type="text" id="fname" name="fname" value="John" disabled><br>
```

c). size Attribute

The input size attribute specifies the visible width, in characters, of an input field.

The default value for size is 20.

Note: The size attribute works with the following input types: text, search, tel, url, email, and password.

```
<input type="text" id="fname" name="fname" size="50"><br>
```

d). maxlength Attribute

The input maxlength attribute specifies the maximum number of characters allowed in an input field.

Note: When a maxlength is set, the input field will not accept more than the specified number of characters. However, this attribute does not provide any feedback. So, if you want to alert the user, you must write JavaScript code.

```
<input type="text" id="pin" name="pin" maxlength="4" size="4">
```

e).min and max Attributes

The input min and max attributes specify the minimum and maximum values for an input field.

The min and max attributes work with the following input types: number, range, date, datetime-local, month, time and week.

Tip: Use the max and min attributes together to create a range of legal values.

f). multiple Attribute

The input multiple attribute specifies that the user is allowed to enter more than one value in an input field.

The multiple attribute works with the following input types: email, and file.

```
<input type="file" id="files" name="files" multiple>
```

g). placeholder Attribute

The input placeholder attribute specifies a short hint that describes the expected value of an input field (a sample value or a short description of the expected format).

The short hint is displayed in the input field before the user enters a value.

The placeholder attribute works with the following input types: text, search, url, tel, email, and password.

```
<input type="tel" id="phone" name="phone" placeholder="123-45-678">
```

h). required Attribute

The input required attribute specifies that an input field must be filled out before submitting the form.

The required attribute works with the following input types: text, search, url, tel, email, password, date pickers, number, checkbox, radio, and file.

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

i). step Attribute

The input step attribute specifies the legal number intervals for an input field.

Example: if step="3", legal numbers could be -3, 0, 3, 6, etc.

Tip: This attribute can be used together with the max and min attributes to create a range of legal values.

The step attribute works with the following input types: number, range, date, datetime-local, month, time and week.

```
<input type="number" id="points" name="points" step="3">
```

J). autofocus Attribute

```
<input type="text" id="fname" name="fname" autofocus><br>
```

3).HTML Graphics,SVG Graphics

A)Canvas Graphich

The HTML <canvas> element is used to draw graphics, on the fly, via JavaScript.
The <canvas> element is only a container for graphics. You must use JavaScript to actually draw the graphics.
Canvas has several methods for drawing paths, boxes, circles, text, and adding images.

A canvas is a rectangular area on an HTML page. By default, a canvas has no border and no content.
The markup looks like this:

```
<canvas id="myCanvas" width="200" height="100"></canvas>
```

Note: Always specify an id attribute (to be referred to in a script), and a width and height attribute to define the size of the canvas. To add a border, use the style attribute.

Here is an example of a basic, empty canvas:

B). SVG Graphics

- SVG stands for Scalable Vector Graphics
- SVG is used to define graphics for the Web
- SVG is a W3C recommendation

a)<svg> Element

The HTML <svg> element is a container for SVG graphics.
SVG has several methods for drawing paths, boxes, circles, text, and graphic images.

Circle

```
<svg width="100" height="100">  
  <circle cx="50" cy="50" r="40"  
    stroke="green" stroke-width="4" fill="yellow" />  
Sorry, your browser does not support inline SVG.  
</svg>
```

Us VS Code

Rectangle

```
<svg width="400" height="100">  
  <rect width="400" height="100"  
    style="fill:rgb(0,0,255);stroke-width:10;stroke:rgb(0,0,0)" />  
Sorry, your browser does not support inline SVG.  
</svg>
```

Us VS Code

Rounded Rectangle

```
<svg width="400" height="180">  
  <rect x="50" y="20" rx="20" ry="20" width="150"  
  height="150"  
  style="fill:red;stroke:black;stroke-width:5;opacity:0.5" />  
Sorry, your browser does not support inline SVG.  
</svg>
```

Us VS Code

Star

```
<svg width="300" height="200">  
  <polygon points="100,10 40,198 190,78 10,78 160,198"  
  style="fill:lime;stroke:purple;stroke-width:5;fill-  
  rule:evenodd;" />  
Sorry, your browser does not support inline SVG.  
</svg>
```

Us VS Code

4).HTML Media

A).Video

```
<video width="320" height="240" controls>
  <source src="movie.mp4" type="video/mp4">
  Your browser does not support the video tag.
</video>
```

Us VS Code

b). <video> Autoplay

```
<video width="320" height="240" autoplay muted>
<video width="320" height="240" autoplay>
```

B). Video - Methods, Properties, and Events

```
<!DOCTYPE html>
<html>
<body>

<div style="text-align:center">
  <button onclick="playPause()">Play/Pause</button>
  <button onclick="Big()">Big</button>
  <button onclick="Small()">Small</button>
  <button onclick="Normal()">Normal</button>
  <br><br>
  <video id="video1" width="420">
    <source src="_____" type="video/mp4">
    <source src="_____" type="video/ogg">
    Your browser does not support HTML video.
  </video>
</div>

<script>
var myVideo = document.getElementById("video1");

function playPause() {
  if (myVideo.paused)
    myVideo.play();
  else
    myVideo.pause();
}

function Big() {
  myVideo.width = 560;
}

function Small() {
  myVideo.width = 320;
```

Us VS Code

| | |
|---|--|
| <pre> } function Normal() { myVideo.width = 420; } </script> <p>Video courtesy of Big Buck Bunny.</p> </body> </html> </pre> | |
|---|--|

C).<audio> Element

<pre> <audio controls> <source src="" type="audio/ogg"> <source src="" type="audio/mpeg"> Your browser does not support the audio element. </audio> </pre>	Us VS Code
--	------------

This is only HTML End Now Start the CSS