

Unit -1

HTML

Prepared by:
Dr. Susheela

HTML

- Hyper Text Markup Language
- HTML was first created by Tim Berners-Lee, Robert Cailliau, and others starting in 1989
- HTML is the standard markup language for creating Web pages
- HTML elements tell the browser how to display the content
- Hypertext means that the document contains links that allow the reader to jump to other places in the document or to another document altogether.
- A Markup Language is a way that computers speak to each other to control how text is processed and presented. To do this HTML uses two things: tags and attributes.
- The latest version is known as HTML5.

HTML Tags

- HTML tags are used to **mark up the start of an HTML element** and they are usually enclosed in angle brackets. An example of a tag is: <h1>.
- Most tags must be opened <h1> and closed </h1> in order to function.
- When using multiple tags, the tags must be closed in the order in which they were opened. For example:

```
<body>  
<p>This is really important!  
</p>  
</body>
```

HTML Tag's Attributes

- Attributes contain additional pieces of information. Attributes take the form of an opening tag and additional info is placed inside.
- An example of an attribute is:

```

```

HTML Editors

- Notepad++
 - Auto-completion feature
 - No support for Mac.
- TextEdit (Mac)
- Sublime Text 3
 - Easily customizable
 - Beginner-friendly
 - Pleasant color schemes to choose from
 - Available for Mac, Windows, and Linux
 - Can't print documents or code
 - No toolbar or dashboard available.
- Komodo Edit
 - Available for Mac, Windows, and Linux
 - Impressive language support
 - No autocompletion by default

Basic Construction of an HTML Page

- <!DOCTYPE html> — This tag **specifies the language** you will write on the page (HTML 5).
- <html> — This tag signals that from here on we are going to write in HTML code.
 - There is one tag that is always included: <title>
- <head> — This is where all the **metadata for the page** goes — stuff mostly meant for search engines and other computer programs.
- <body> — This is where the **content of the page** goes.
- The HTML <body> is where we add the content which is designed for viewing by human eyes.

```
<!DOCTYPE html>
<html>
<head>
<title> Title of Page </title>
</head>
<body>
<h1>First Heading</h1>
<p>paragraph</p>
</body>
</html>
```

HTML 5.0

- HTML5 is a cooperation between the World Wide Web Consortium (W3C) and the Web Hypertext Application Technology Working Group (WHATWG).
- HTML5 introduces a number of new elements and attributes that can help you in building modern websites.
- HTML5 is designed, as much as possible, to be backward compatible with existing web browsers.
- HTML5 comes with a lot of flexibility, and it supports the following features –
 - Uppercase tag names.
 - Quotes are optional for attributes.
 - Attribute values are optional.
 - Closing empty elements are optional.

Dynamic Page Support

Now a days there is need of dynamic as well as interactive websites rather than static websites. Here are enormous features that provide the dynamic touch to the website :

- time – This helps in adding current time as well as date to the webpage.
- meter – It helps in indicating that how much space in the storage disk is still there.
- progress bar – It helps in knowing the progress of the task that has been assigned for its completion.

New Elements in HTML 5

- Navigation Tag <nav>
- Header Tag <header>
- Footer Tag <footer>
- Section Tag <section>
- Article Tag <article>
- Main Tag <main>
- Figure Tag <figure>
- Figure caption Tag <figcaption>
- Mark Tag <mark>
- Progress Tag <progress>
- Sematic Tag <sem>
- Audio Tag <audio>
- Video Tags <video>
- Media Control
- Scalable Vector Graphics Tag <svg>
- Canvas Tag <canvas>

HTML	HTML5
HTML does not provide native audio and video support.	HTML5 provides native audio and video support.
HTML only supports vector graphics if used in conjunction with different technologies like <u>Flash</u> , <u>VML</u> , or <u>Silverlight</u> .	HTML5 supports SVG (Scalable Vector Graphics), Canvas, and other virtual vector graphics.
HTML doesn't allow users to draw shapes such as circles, triangles, and rectangles.	HTML5 allows users to draw shapes such as circles, triangles, and rectangles.
HTML only uses browser cache and cookies to store data temporarily.	HTML5 uses web SQL databases, local storage, and application cache for storing data temporarily.
JavaScript and browser interface run in the same thread.	JavaScript and browser interface run in separate threads.
Longer <u>document type declaration</u> .	Shorter document type declaration.
Longer character encoding declaration. Uses the ASCII <u>character set</u> .	Shorter <u>character encoding</u> declaration. Uses the UTF-8 character set.
Compatible with almost all browsers.	Only compatible with newer browsers, considering there are many new tags and elements which only some browsers support.
Built based on <u>Standard Generalized Markup Language (SGML)</u> .	HTML5 has improved parsing rules providing enhanced compatibility.
Programmers are unable to use features that determine a user's geolocation..	HTML5 has a JavaScript geolocation API, which can be used to identify any user's location when accessing the website.
It cannot handle images in a single file.	It is possible of handling images in a single file.

Semantic Elements in HTML5

- Semantic elements in HTML refers to the tags that provide meaning to an HTML page rather than just presentation.
- It makes HTML more comprehensible by better defining the different sections and layout of web pages.
- The semantic HTML tags help the search engines and other user devices to determine the importance and context of web pages.
- The pages made with semantic elements are much easier to read.
- It has greater accessibility. It offers a better user experience.



Navigation <nav> Tag

```
<!DOCTYPE html>
<html>
<body>
<nav>
  <a href="/html/">HTML</a>
  <a href="/css/">CSS</a>
  <a href="/js/">JavaScript</a>
  <a href="/jquery/">jQuery</a>
</nav>

</body>
</html>
```

[HTML](#) [CSS](#) [JavaScript](#) [jQuery](#)

Article Tag <article>

- The <article> tag 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 site.
- Potential sources for the <article> element:
 - Forum post
 - Blog post
 - News story

The <article> element does not render as anything special in a browser. However, you can use CSS to style the <article> element (see example below).

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

<h1> Article Tag</h1>

<article>
<h2>Google Chrome</h2>
<p>Google Chrome is a web browser developed by Google, released in 2008. Chrome is the world's most popular web browser today!</p>
</article>

<article>
<h2>Mozilla Firefox</h2>
<p>Mozilla Firefox is an open-source web browser developed by Mozilla. Firefox has been the second most popular web browser since January, 2018.</p>
</article>

</body>
</html>
```

Article Tag

Google Chrome

Google Chrome is a web browser developed by Google, released in 2008. Chrome is the world's most popular web browser today!

Mozilla Firefox

Mozilla Firefox is an open-source web browser developed by Mozilla. Firefox has been the second most popular web browser since January, 2018.

Header Tag <header>

- HTML **<header> tag** is used as a container of introductory content or navigation links. Generally a <header> element contains one or more heading elements, logo or icons or author's information.
- You can use several <header> elements in one document, but a <header> element cannot be placed within a <footer>, <address> or another <header> element.

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

<h1> Header Tag</h1>

<header>
  <h1> Heading..... </h1>
  <p>Use of header tag</p>
  <p>Some additional information here</p>
</header>
|
</body>
</html>
```

Header Tag

Heading.....

Use of header tag

Some additional information here

Footer Tag <footer>

- The <footer> tag 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

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

<h1>The Footer Tag</h1>

<footer>
  <address>
    Written by <a href="mailto:susheela.dahiya@gehu.ac.in">
      Dr. Susheela Dahiya</a><br>
    Visit us at:<br>
    Example.com<br>
    Dehradun<br>
    India
  </address>
</footer>

</body>
</html>
```

The Footer Tag

Written by [Dr. Susheela Dahiya](mailto:susheela.dahiya@gehu.ac.in)
Visit us at:
Example.com
Dehradun
India

Main Tag <main>

- HTML <main> tag is used to represent the main content of the <body> tag.
- The <main> tag is written within <body> tag. It is used to accurately describe the primary content of a page.
- The content of the main tag is directly related to the central topic of the document.
- Author should not include more than one <main> tag within a document.
- The <main> element should not be used as a child of an <article>, <aside>, <header>, <footer>, or <nav> element.

Mark Tag <mark>

- HTML <mark> tag is used to highlight the text

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

<p>
<h2>This is an example of <mark> Mark Tag </mark> </h2>
</p>

</body>
</html>
```

This is an example of Mark Tag

Progress Tag

- The `<progress>` tag represents the completion progress of a task.
- Always add the `<label>` tag for best accessibility practices

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

<h1>The Progress Tag</h1>

<label for="file">Downloading progress:</label>
<progress id="file" value="30" max="100"> 30% </progress>

</body>
</html>
```

The Progress Tag

Downloading progress: 

Figure & Figure Caption Tag

1. **HTML <figure> tag** is used to mark up a photo in the document on a web page.
2. **HTML <figcaption> tag** is used to add a caption to a photo.

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

<figure>
  
  <figcaption>Fig.1 - Trulli, Puglia, Italy.</figcaption>
</figure>

</body>
</html>
```



Fig.1 - Trulli, Puglia, Italy.

Figure & Figure Caption Tag

1. **HTML <figure> tag** is used to mark up a photo in the document on a web page.
2. **HTML <figcaption> tag** is used to add a caption to a photo.

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

<figure>
  
  <figcaption>Fig.1 - Trulli, Puglia, Italy.</figcaption>
</figure>

</body>
</html>
```



Fig.1 - Trulli, Puglia, Italy.

<video> tag

- HTML 5 supports only mp4 & ogg audio/video
- The text between the <video> and </video> tags will only be displayed in browsers that do not support the <video> element.
- You can set up different sources, and the first source that fits the preferences is the one being used.
- Attributes:
 - Controls - Adds video controls, like play, pause, and volume.
 - Preload - auto, none
 - width and height - If height and width are not set, the page might flicker while the video loads.
 - Add muted after autoplay to let your video start playing automatically (but muted)

```
<!DOCTYPE html>
<html>
<body>
<h2>Example of video and audio tag</h2>

<video controls preload="none">
  <source src="movie.mp4" type="video/mp4">
  Your browser does not support the video tag.
</video>

</body>
</html>
```

Example of video and audio tag



```
<!DOCTYPE html>
<html>
<body>
<h2>Example of video and audio tag</h2>

<video controls preload="auto">
  <source src="movie.mp4" type="video/mp4">
  Your browser does not support the video tag.
</video>

</body>
</html>
```

Example of video and audio tag



Audio Tag

```
<!DOCTYPE html>
<html>
<body>
<h2>Example of audio tag</h2>
<audio controls autoplay>
  <source src = "/html5/audio.ogg" type = "audio/ogg" />
  <source src = "/html5/audio.wav" type = "audio/wav" />
  Your browser does not support the audio element.
</audio>
</body>
</html>
```

Example of audio tag



```
<!DOCTYPE html>
<html>
<body>
<h2>Example of video and audio tag</h2>

<video controls preload="none">
  <source src="movie.mp4" type="video/mp4">
  Your browser does not support the video tag.
</video>

</body>
</html>
```

Example of video and audio tag



```
<!DOCTYPE html>
<html>
<body>
<h2>Example of video and audio tag</h2>

<video controls preload="auto">
  <source src="movie.mp4" type="video/mp4">
  Your browser does not support the video tag.
</video>

</body>
</html>
```

Example of video and audio tag



Scalable Vector Graphics <svg> Tag

```
<!DOCTYPE html>
<html>
<body>
<h2> Rounded Rectangle </h2>
<svg width="500" height="500">
  <rect width="400" height="100" style="fill:rgb(0,0,255);stroke-width:10;stroke:rgb(0,0,0)" />
  <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>

</body>
</html>
```

Rounded Rectangle



```
<!DOCTYPE html>
<html>
<body>
<h2> Rectangle </h2>
<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>

</body>
</html>
```

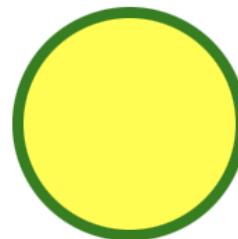
Rectangle



```
<!DOCTYPE html>
<html>
<body>
<h2> Circle </h2>
<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>

</body>
</html>
```

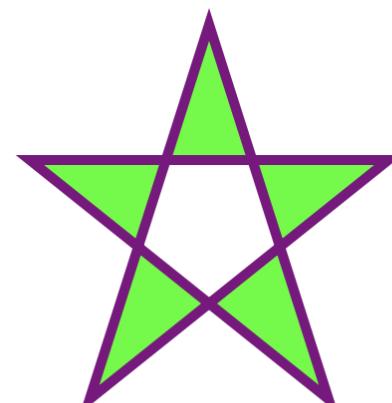
Circle



```
<!DOCTYPE html>
<html>
<body>
<h2> Star </h2>
<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>

</body>
</html>
```

Star



HTML Text Formatting Tags

Element	Meaning	Purpose
	Bold	Highlight important information
	Strong	Similarly to bold, to highlight key text
<i>	Italic	To denote text
	Emphasised Text	Usually used as image captions
<mark>	Marked Text	Highlight the background of the text
<small>	Small Text	To shrink the text
<strike>	Striked Out Text	To place a horizontal line across the text
<u>	Underlined Text	Used for links or text highlights
<ins>	Inserted Text	Displayed with an underline to show an inserted text
<sub>	Subscript Text	Typographical stylistic choice
<sup>	Superscript Text	Another typographical presentation style
<u></u>		Defines deleted text

```
<!DOCTYPE html>
<html>
<body>
<p>This text is normal.</p>
<p><b>This text is bold</b></p>
<p><strong>This text is strong</strong></p>
<p><i>This text is italic</i></p>
<p><u>This text is underline</u></p>
<p><s>This text is strike through</s></p>
<p><small>This is some smaller text.</small></p>
<p>Do not forget to buy <mark>fruits</mark> today.</p>
<p>My favorite color is <del>blue</del> red.</p>
<p>My favorite color is <del>blue</del> <ins>red</ins>.</p>
<p>This is <sub> subscript </sub> and <sup>superscript</sup></p>
<p><em>This text is emphasized.</em></p>
</body>
</html>
```

This text is normal.

This text is bold

This text is strong

This text is italic

This text is underline

~~This text is strike through~~

This is some smaller text.

Do not forget to buy fruits today.

My favorite color is blue red.

My favorite color is blue red.

This is subscript and superscript

This text is emphasized.

HTML Elements

The HTML **element** is everything from the start tag to the end tag:

`<tagname>Content goes here...</tagname>`

Nested HTML Elements

- HTML elements can be nested

```
<!DOCTYPE html>
<html>
<head>
<title> Title of Page </title>
</head>
<body>
<h1>First Heading</h1>
<p>paragraph</p>

</body>
</html>
```

HTML Attributes

- All HTML elements can have **attributes**
- Attributes provide **additional information** about elements
- Attributes are always specified in **the start tag**
- Attributes usually come in name/value pairs like: **name="value"**

Style Attribute

- The style attribute is used to add styles to an element, such as color, font, size, etc.
- HTML supports 140 standard color names.

```
<tagname style="property:value;">

<p style="color:red;">This is a red paragraph.</p>
<body style="background-color:powderblue;">
<h1 style="background-color:powderblue;">This is a heading</h1>
<p style="background-color:green;">This is a paragraph.</p>
<h1 style="font-family:verdana;">This is a heading</h1>
<p style="font-family:courier;">This is a paragraph.</p>
<h1 style="font-size:300%;">This is a heading</h1>
<p style="font-size:160%;">This is a paragraph.</p>
<h1 style="text-align:center;">Centered Heading</h1>
<p style="text-align:center;">Centered paragraph.</p>
<h1 style="border:2px solid Red;">Hello World</h1>
```

```
<head>
<style> /* This is internal styling */
h1
{
color: indianred;
}
p
{
color: red;
}
</style>
</head>
```

HTML Headings

- HTML headings are defined with the `<h1>` to `<h6>` tags.
- `<h1>` defines the most important heading. `<h6>` defines the least important heading.
- `<h1>Heading 1</h1>`
`<h2>Heading 2</h2>`
`<h3>Heading 3</h3>`
`<h4>Heading 4</h4>`
`<h5>Heading 5</h5>`
`<h6>Heading 6</h6>`

HTML Line Breaks

- The HTML `
` element defines a line break.
- Use `
` if you want a line break (a new line) without starting a new paragraph:

```
<p>This is<br>a paragraph<br>with line breaks.</p>
```

- The `
` tag is also an empty tag, means that it has no end tag.

This is
a paragraph
with line breaks.

HTML Paragraphs

- A paragraph always starts on a new line and is usually a block of text.
- <p> tag defines a paragraph.
- <p>
This paragraph
contains a lot of lines
in the source code,
but the browser
ignores it.
</p>
- <p>
The number of lines in a paragraph depends on the size of the browser window. If you resize the browser window, the number of lines in this paragraph will change.
</p>

HTML Horizontal Rules

- The `<hr>` tag defines a thematic break in an HTML page and is most often displayed as a horizontal rule.
- The `<hr>` tag is an empty tag means that it has no end tag
- The `<hr>` element is used to separate content (or define a change) in an HTML page:

```
<h1>This is heading 1</h1>
<p>This is some text.</p>
<hr>
<h2>This is heading 2</h2>
<p>This is some other text.</p>
<hr>
<h2>This is heading 2</h2>
<p>This is some other text.</p>
```

This is heading 1

This is some text.

This is heading 2

This is some other text.

This is heading 2

This is some other text.

 Tag

```

```

1. Absolute URL - Links to an external image that is hosted on another website. Example: src="https://www.gehu.ac.in/images/img1.jpg".

- External images might be under copyright. If you do not get permission to use it, you may be in violation of copyright laws. In addition, you cannot control external images; it can suddenly be removed or changed.

2. Relative URL - Links to an image that is hosted within the website. Here, the URL does not include the domain name. If the URL begins without a slash, it will be relative to the current page. Example: src="img_girl.jpg". If the URL begins with a slash, it will be relative to the domain. Example: src="/images/img_girl.jpg".

- It is almost always best to use relative URLs. They will not break if you change domain.

HTML Comment Tag

- HTML Comment Tag can help document the source code.
- Comments can sometimes also used to hide content.

Syntax:

```
<!-- Write your comments here -->
```

Example:

```
<!DOCTYPE html>
<html>
<body>
<!-- This is a comment -->
<p>This is a paragraph.</p>
<!-- Comments are not displayed in the browser -->
</body>
</html>
```

HTML Links - Hyperlinks

- Links allow users to click their way from page to page.

- <a> tag defines a hyperlink.

- Syntax:

```
<a href="url">Link text</a>
```

- By default, links will appear as follows in all browsers:

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

- 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

- Visit GEHU!

HTML Link Colors

```
<!DOCTYPE html>
<html>
<head>
<style>
a:link {
  color: green;
  background-color: transparent;
}

a:visited {
  color: pink;
  background-color: transparent;
}

a:hover {
  color: red;
  background-color: transparent;
}

a:active {
  color: yellow;
  background-color: transparent;
}
</style>
<h2>Link Colors</h2>
<p>You can change the default colors of links</p>
<a href="html_images.asp" target="_blank">HTML Images</a>

</body>
</html>
```

Link Colors

You can change the default colors of links

HTML Images

Link Colors

You can change the default colors of links

HTML Images

- a:link - a normal, unvisited link
- a:visited - a link the user has visited
- a:hover - a link when the user mouses over it
- a:active - a link the moment it is clicked

Lists

- Ordered Lists
- Unordered Lists
- Definition List <dl>

Tag	Description
<u></u>	Defines an unordered list
<u></u>	Defines an ordered list
<u></u>	Defines a list item
<u><dl></u>	Defines a description list
<u><dt></u>	Defines a term in a description list
<u><dd></u>	Describes the term in a description list

Lists

- Ordered Lists
- Unordered Lists
- Definition List <dl>

```
<h2>An ordered HTML list</h2>
```

```
<ol>
  <li>An item </li>
  <li>Another item </li>
  <li>Another goes here </li>
</ol>
```

An ordered HTML list

1. An item
2. Another item
3. Another goes here

```
<h2>An unordered HTML list</h2>
```

```
<ul>
  <li>An item </li>
  <li>Another item </li>
  <li>Another goes here </li>
</ul>
```

An unordered HTML list

- An item
- Another item
- Another goes here

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

<h2>Definition list</h2>

<dl>
  <dt>Item</dt>
  <dd>The definition goes here
</dl>

</body>
</html>
```

Definition list

Item

The definition goes here

Unordered List

Value	Description	
disc	Sets the list item marker to a bullet (default)	<pre><ul type="disc"> Coffee Tea Milk </pre> <ul style="list-style-type: none">Item 1Item 2Item 2
circle	Sets the list item marker to a circle	
square	Sets the list item marker to a square	
none	The list items will not be marked	<pre>SQUARE ■ Item 1 ■ Item 2 ■ Item 2</pre> <pre>NONE Item 1 Item 2 Item 2</pre>

Unordered List with Disc Bullets

```
<ul style="list-style-type:disc;">
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 2</li>
</ul>
```

Unordered List with Disc Bullets

- Item 1
- Item 2
- Item 2

Ordered List

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

<h2>Ordered List with Numbers</h2>

<ol type="1">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>

</body>
</html>
```

Ordered List with Numbers

- 1. Coffee
- 2. Tea
- 3. Milk

Type	Description
type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters
type="a"	The list items will be numbered with lowercase letters
type="I"	The list items will be numbered with uppercase roman numbers
type="i"	The list items will be numbered with lowercase roman numbers

- A. Coffee
- B. Tea
- C. Milk

- I. Coffee
- II. Tea
- III. Milk

```
<ol type="A" start="E">
  <li> Coffee </li>
  <li> Tea </li>
  <li> Milk </li>
</ol>
```

OUTPUT
E. Coffee
F. Tea
G. Milk

Nested List

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

<h2>A Nested List</h2>
<p>Lists can be nested (list inside list):</p>

<ul>
  <li>Coffee</li>
  <li>Tea
    <ul>
      <li>Black tea</li>
      <li>Green tea</li>
    </ul>
  </li>
  <li>Milk</li>
</ul>

</body>
</html>
```

A Nested List

Lists can be nested (list inside list):

- Coffee
- Tea
 - Black tea
 - Green tea
- Milk

Table Tag

Tag	Description
<u><table></u>	Defines a table
<u><th></u>	Defines a header cell in a table
<u><tr></u>	Defines a row in a table
<u><td></u>	Defines a cell in a table
<u><caption></u>	Defines a table caption
<u><colgroup></u>	Specifies a group of one or more columns in a table for formatting
<u><col></u>	Specifies column properties for each column within a <colgroup> element
<u><thead></u>	Groups the header content in a table
<u><tbody></u>	Groups the body content in a table
<u><tfoot></u>	Groups the footer content in a table

Table Tag

Cell spacing	<code><table cellspacing="">...</table></code>
Cell padding	<code><table cellpadding="">...</table></code>
Table border	<code><table border="">...</table></code>
Alignment	<code><table align=center/left/right>...</table></code>
colspan in table	<code><table colspan="">...</table></code>
rowspan in table	<code><table rowspan="">...</table></code>
Cell color	<code><table bgcolor="#\$\$\$\$\$\$">...</table></code>
No linebreaks	<code><table nowrap>...</table></code>

```
<!DOCTYPE html>
<html>
<head> <title> Table Tag </title> </head>
<body>
<table>
<tr>
  <th>Name</th>
  <th>Job</th>
  <th>Working Experience</th>
</tr>
<tr>
  <td>John</td>
  <td>Software Engineer</td>
  <td>5 Years</td>
</tr>
<tr>
  <td>Ale</td>
  <td>Senior Web developer</td>
  <td>2 Year</td>
</tr>
<tr>
  <td>Jack</td>
  <td>Junior Tech Writer</td>
  <td>6 Months</td>
</tr>
</table>
</body>
</html>
```

Name	Job	Working Experience
John	Software Engineer	5 Years
Ale	Senior Web developer	2 Years
Jack	Junior Tech Writer	6 Months

Table Tag

```
<!DOCTYPE html>
<html>
<head> <title> Table Tag </title> </head>
<body>
<table border="1">
  <tr>
    <th>Name</th>
    <th>Job</th>
    <th>Working Experience</th>
  </tr>
  <tr>
    <td>John</td>
    <td>Software Engineer</td>
    <td>5 Years</td>
  </tr>
  <tr>
    <td>Ale</td>
    <td>Senior Web developer</td>
    <td>2 Year</td>
  </tr>
  <tr>
    <td>Jack</td>
    <td>Junior Tech Writer</td>
    <td>6 Months</td>
  </tr>
</table>
```

Name	Job	Working Experience
John	Software Engineer	5 Years
Ale	Senior Web developer	2 Year
Jack	Junior Tech Writer	6 Months

cellspacing Attribute

```
<html>
<head> <title> Table Tag </title> </head>
<body>
<table border="1" cellspacing = "10">
  <tr>
    <th>Name</th>
    <th>Job</th>
    <th>Working Experience</th>
  </tr>
  <tr>
    <td>John</td>
    <td>Software Engineer</td>
    <td>5 Years</td>
  </tr>
  <tr>
    <td>Ale</td>
    <td>Senior Web developer</td>
    <td>2 Year</td>
  </tr>
  <tr>
    <td>Jack</td>
    <td>Junior Tech Writer</td>
    <td>6 Months</td>
  </tr>
</table>
</body>
</html>
```

Name	Job	Working Experience
John	Software Engineer	5 Years
Ale	Senior Web developer	2 Year
Jack	Junior Tech Writer	6 Months

```
<html>
<head> <title> Table Tag </title> </head>
<body>
<table border = 5>
  <tr>
    <th>Name</th>
    <th colspan="2">Jobs</th>
    <th>Working Experience</th>
  </tr>
  <tr>
    <td>John</td>
    <td>Software Engineer</td>
    <td>Data Analyst</td>
    <td rowspan="2">5 Years</td>
  </tr>
  <tr>
    <td>Ale</td>
    <td colspan="2">Senior Web developer</td>
  </tr>
  <tr>
    <td>Jack</td>
    <td>Junior Tech Writer</td>
    <td>Blogger</td>
    <td>6 Months</td>
  </tr>
</table>
</body>
</html>
```

Name	Jobs	Working Experience
John	Software Engineer	Data Analyst
Ale	Senior Web developer	5 Years
Jack	Junior Tech Writer	Blogger
		6 Months

Screensh

```
<body>
<table border = 1>
  <tr>
    <th>Name</th>
    <th colspan="2">Jobs</th>
    <th>Working Experience</th>
  </tr>
  <tr>
    <td>John</td>
    <td>Software Engineer</td>
    <td>Data Analyst</td>
    <td>5 Years</td>
  </tr>
  <tr>
    <td>Ale</td>
    <td colspan="2">Senior Web developer</td>
    <td>2 Year</td>
  </tr>
  <tr>
    <td>Jack</td>
    <td>Junior Tech Writer</td>
    <td>Blogger</td>
    <td>6 Months</td>
  </tr>
</table>
</body>
```

Name	Jobs	Working Experience
John	Software Engineer	Data Analyst
Ale	Senior Web developer	2 Year
Jack	Junior Tech Writer	Blogger

```
<body>





```

New Employees Records

Name	Jobs	Working Experience
John	Software Engineer	5 Years
Ale	Senior Web developer	
Jack	Junior Tech Writer	Blogger
		6 Months

```

<body>
<table border = "10">
  <caption>
    <b>New Employees Records</b>
  </caption>
  <tr>
    <th>Name</th>
    <th colspan="2">Jobs</th>
    <th>Working Experience</th>
  </tr>
  <tr>
    <td bgcolor = "Green">John</td>
    <td>Software Engineer</td>
    <td>Data Analyst</td>
    <td rowspan="2">5 Years</td>
  </tr>
  <tr>
    <td>Ale</td>
    <td colspan="2" bgcolor = "Red">Senior Web developer</td>
  </tr>
  <tr>
    <td>Jack</td>
    <td>Junior Tech Writer</td>
    <td>Blogger</td>
    <td>6 Months</td>
  </tr>
</table>
</body>

```

Name	Jobs	Working Experience
John	Software Engineer	Data Analyst
Ale	Senior Web developer	5 Years
Jack	Junior Tech Writer	Blogger
		6 Months

```

<body>
<table border = "10">
  <caption>
    <b>New Employees Records</b>
  </caption>
  <tr>
    <th>Name</th>
    <th colspan="2">Jobs</th>
    <th>Working Experience</th>
  </tr>
  <tr id="eligible" bgcolor="#0bb31e">
    <td>John</td>
    <td>Software Engineer</td>
    <td>Data Analyst</td>
    <td rowspan="2">5 Years</td>
  </tr>
  <tr>
    <td>Ale</td>
    <td colspan="2" bgcolor = "Red">Senior Web developer</td>
  </tr>
  <tr>
    <td>Jack</td>
    <td>Junior Tech Writer</td>
    <td>Blogger</td>
    <td>6 Months</td>
  </tr>
</table>
</body>

```

New Employees Records

Name	Jobs	Working Experience
John	Software Engineer	Data Analyst
Ale	Senior Web developer	5 Years
Jack	Junior Tech Writer	Blogger
		6 Months

```

<table border = "10">
  <caption>
    <b>New Employees Records</b>
  </caption>
  <tr>
    <th>Name</th>
    <th colspan="2">Jobs</th>
    <th>Working Experience</th>
  </tr>
  <tr id="eligible" bgcolor="#0bb31e">
    <td>John</td>
    <td>Software Engineer</td>
    <td>Data Analyst</td>
    <td>5 Years</td>
  </tr>
  <tr>
    <td>Ale</td>
    <td colspan="2" bgcolor = "Red">Senior Web developer</td>
    <td>7 Years</td>
  </tr>
  <tr>
    <td>Jack</td>
    <td>Junior Tech Writer</td>
    <td>Blogger</td>
    <td align = "center">6 Months</td>
  </tr>
</table>

```

Align =
left, right, center

New Employees Records

Name	Jobs	Working Experience
John	Software Engineer	Data Analyst
Ale	Senior Web developer	7 Years
Jack	Junior Tech Writer	Blogger

```





```

New Employees Records

Name	Jobs		Working Experience
John	Software Engineer	Data Analyst	5 Years
Ale	Senior Web developer		7 Years
Jack	Junior Tech Writer	Blogger	6 Months

New Employees Records

Name	Jobs		Working Experience
John	Software Engineer	Data Analyst	5 Years
Ale	Senior Web developer		7 Years
Jack	Junior Tech Writer	Blogger	6 Months

```
<table>
  <caption>Nested Tables</caption>
  <tr>
    <th>Header of Table 1</th>
    <th>Header of Table 2</th>
  </tr>
  <tr>
    <td>
      <table>
        <tr>
          <th>1st Header of nested table 1</th>
          <th>2nd Header of nested table 1</th>
        </tr>
        <tr>
          <td>1st cell of nested table</td>
          <td>2nd cell of nested table</td>
        </tr>
        <tr>
          <td>3rd cell of nested table</td>
          <td>4th cell of nested table</td>
        </tr>
      </table>
    </td>
  <td>
    <table>
      <tr>
        <th>1st Header of nested table 2</th>
        <th>2nd Header of nested table 2</th>
      </tr>
      <tr>
        <td>1st cell of nested table</td>
        <td>2nd cell of nested table</td>
      </tr>
      <tr>
        <td>3rd cell of nested table</td>
        <td>4th cell of nested table</td>
      </tr>
    </table>
  </td>
  </tr>
</table>
```

Nested Tables

Header of Table 1

1st Header of nested table 1 **2nd Header of nested table 1**

1st cell of nested table	2nd cell of nested table
3rd cell of nested table	4th cell of nested table

Header of Table 2

1st Header of nested table 2 **2nd Header of nested table 2**

1st cell of nested table	2nd cell of nested table
3rd cell of nested table	4th cell of nested table

HTML <form>

- <form> is a HTML tag to collect input data with interactive controls.
- It provides facilities to input text, number, values, email, password, and control fields such as checkboxes, radio buttons, submit buttons, etc., or in other words, form is a container that contains input elements like text, email, number, radio buttons, checkboxes, submit buttons, etc.
- Forms are generally used when you want to collect data from the user.
- For example, a user wants to register for an event, so he/she has to first enter name, mail id, contact number etc. in the registration form to get registered.

Form Tags

Tag	Description
<form>	It defines an HTML form to enter inputs by the user side.
<input>	It defines an input control.
<textarea>	It defines a multi-line input control.
<label>	It defines a label for an input element.
<fieldset>	It groups the related elements in a form.
<legend>	It defines a caption for a <fieldset> element.
<select>	It defines a drop-down list.
<optgroup>	It defines a group of related options in a drop-down list.
<option>	It defines an option in a drop-down list.
<button>	It defines a clickable button.
<datalist>	It specifies a list of pre-defined options for input control.
<keygen>	It defines a key-pair generator field for forms.
<output>	It defines the result of a calculation.

<input> tag Attributes

1	type Indicates the type of input control and for text input control it will be set to text .
2	name Used to give a name to the control which is sent to the server to be recognized and get the value.
3	value This can be used to provide an initial value inside the control.
4	size Allows to specify the width of the text-input control in terms of characters.
5	maxlength Allows to specify the maximum number of characters a user can enter into the text box.

<input> tag type Attribute values

- Text

```
<input type="text" name = "sname" size = "50" required>
```

- Radio

```
<input type = "radio" name = "Gender" value = "Male">
```

- Checkbox

```
<input type="checkbox" name="color" value="red" />Red</li>
```

- Textarea

```
<textarea rows = "5" cols = "50" name = "description">
```

Enter description here...

```
</textarea>
```

- Submit

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

Text Box

- **Single-line text input controls** – This control is used for items that require only one line of user input, such as search boxes or names. They are created using HTML **<input>** tag.
- **Password input controls** – This is also a single-line text input but it masks the character as soon as a user enters it. They are also created using HTML **<input>** tag.
- **Multi-line text input controls** – This is used when the user is required to give details that may be longer than a single sentence. Multi-line input controls are created using HTML **<textarea>** tag.

```
<label>Username : <input type="text" name = "sname" size = "50" required>
</label>
<label>Password : <input type="password" name = "pass" /></label>
```

<input> tag

<label>Username :

 <input type="text" name = "sname" size = "50" required>

</label>

<label>Password :

 <input type="password" name = "pass" maxlength = "10" />

</label>

Username :

Password :

textarea

```
<label>Description : </label><br>
<textarea rows = "5" cols = "50" name = "description">
    Enter description here...
</textarea>
```

Output:

Description :

Sr.No	Attribute & Description
1	name Used to give a name to the control which is sent to the server to be recognized and get the value.
2	rows Indicates the number of rows of text area box.
3	cols Indicates the number of columns of text area box

Radio button

```
<label> Gender: <input type = "radio" name = "Gender" value = "Male"> Male  
<input type = "radio" name = "Gender" value = "Female"> Female
```

Output:

Gender: Male Female

Sr.No	Attribute & Description
1	type Indicates the type of input control and for checkbox input control it will be set to radio.
2	name Used to give a name to the control which is sent to the server to be recognized and get the value.
3	value The value that will be used if the radio box is selected.
4	checked Set to <i>checked</i> if you want to select it by default.

checkbox

Choose Colors:

 <input type="checkbox" name="color" value="red">Red
 <input type="checkbox" name="color" value="blue">Blue
 <input type="checkbox" name="color" value="green" >Green

Output:

Choose Colors:

1. Red
2. Blue
3. Green

Sr.No	Attribute & Description
1	type Indicates the type of input control and for checkbox input control it will be set to checkbox ..
2	name Used to give a name to the control which is sent to the server to be recognized and get the value.
3	value The value that will be used if the checkbox is selected.
4	checked Set to <i>checked</i> if you want to select it by default.

Select Box (Drop Down) Control

A select box, also called drop down box which provides option to list down various options in the form of drop down list, from where a user can select one or more options.

```
<select name = "dropdown">  
    <option value = "Maths" selected>Maths</option>  
    <option value = "Physics">Physics</option>  
</select>
```

Output:

Maths 

Following is the list of important attributes of `<option>` tag –

Sr.No	Attribute & Description
1	value The value that will be used if an option in the select box box is selected.
2	selected Specifies that this option should be the initially selected value when the page loads.
3	label An alternative way of labeling options

Sr.No	Attribute & Description
1	name Used to give a name to the control which is sent to the server to be recognized and get the value.
2	size This can be used to present a scrolling list box.
3	multiple If set to "multiple" then allows a user to select multiple items from the menu.

```
<select name = "dropdown" size = "2">  
    <option value = "Maths" selected>Maths</option>  
    <option value = "Physics">Physics</option>  
</select>
```

Output:

Maths
Physics

```
<html>
<head><title> Student Registration </title></head>
<body>
<h1> Student Registration </h1>
<form>
    <label>Username : <input type="text" name = "sname" size = "50" maxlength = "10" placeholder = "Enter your name"></label> <br><br>
    <label>Password : <input type="password" name = "pass" maxlength = "10"></label> <br> <br>
    <label> Gender: <input type = "radio" name = "Gender" value = "Male"> Male
    <input type = "radio" name = "Gender" value = "Female"> Female <br> <br>
    Choose programming languages you know:<br>
    <ol>
        <li> <input type="checkbox" name="C" value="yes" checked>C</li>
        <li> <input type="checkbox" name="C++" value="yes" />C++</li>
        <li> <input type="checkbox" name="Java" value="yes" checked>Java</li>
        <li> <input type="checkbox" name="PHP" value="yes" /> PHP </li>
        <li> <input type="checkbox" name="Python" value="yes" />Python</li>
    </ol> <br> <br>
    Choose your Favourite Subject: <br>
    <select name = "dropdown" size = "2" multiple>
        <option value = "OS">OS</option> <option value = "DBMS" selected>DBMS</option>
        <option value = "CSA">CSA</option> <option value = "Data Structure">Data Structure</option>
    </select> <br><br>
    <label>Description : </label><br> <textarea rows = "5" cols = "50" name = "description"> Enter description here... </textarea> <br><br>
    Do you want to receive notifications: <br>
    <input type="radio" name="agree" value="yes" checked>Yes <br>
    <input type="radio" name="agree" value="no">No <br> <br><br>
    <button type="submit">Submit</button>
</form>
</body>
</html>
```

Student Registration

Username :

Password :

Gender: Male Female

Choose programming languages you know:

1. C
2. C++
3. Java
4. PHP
5. PHP

Choose your Favourite Subject:

OS
DBMS
CSA
Data Structure

Description :

Enter description here...

Do you want to receive notifications?

- Yes
 No

The 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.
- Example
 - `<form action="/action_page.php" method="get">`
 - `<form action="/action_page.php" method="post">`
- **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
- **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

Data List Tag

- The `<datalist>` tag specifies a list of pre-defined options for an `<input>` element.
- The `<datalist>` tag is used to provide autocomplete feature in the HTML files. It can be used with an input tag so that users can easily fill the data in the forms using select the data.
- Syntax: `<datalist> ... </datalist>`
- Once the user starts typing the input element of `<datalist>` tag, the user will see the pre-defined options starting with the letter or word typed by the user.
- To use the `<datalist>` tag, the id of the tag must be the same as of the `<input>` element attribute.
- Both `<datalist>` and `<select>` tags are used for choosing an option from the given list. But the main difference between both is that in the `<datalist>` tag the user can enter its own input and add that as an option with the help of the `<input>` element whereas the `<select>` tag doesn't provide this feature.

```

<!DOCTYPE html>
<html>
<body>

<h1>The datalist element</h1>

<form action="/action_page.php" method="get">
  <label for="browser">Choose your browser from the list:</label>
  <input list="browsers" name="browser" id="browser">
  <datalist id="browsers">
    <option value="Edge">
    <option value="Firefox">
    <option value="Chrome">
    <option value="Opera">
    <option value="Safari">
  </datalist>
  <input type="submit">
</form>

</body>
</html>

```

The datalist element

Choose your browser from the list:

▼

Edge
Firefox
Chrome
Opera
Safari

The datalist element

Choose your browser from the list:

▼

Submitted Form Data

Your input was received as:

browser=Internet Explorer

The server has processed your input and returned this answer.

Some More Tags:

- <fieldset> tag
 - The <fieldset> tag is used to group related elements in a form.
 - The <fieldset> tag draws a box around the related elements.
 - The <legend> tag is used to define a caption for the <fieldset> element.
- <map> tag
 - The <map> tag is used to define an image map. An image map is an image with clickable areas (hotspots).
 - Required attribute is “name” attribute which is associated with the tag’s usemap attribute and creates a relationship between the image and the map.
 - The <map> element contains a number of <area> elements, that defines the clickable areas in the image map.
- <bdi> Bi-Directional Isolation Tag
 - The <bdi> tag isolates a part of text that might be formatted in a different direction from other text outside it.
 - This element is useful when embedding user-generated content with an unknown text direction.
- <embed> Embedded Tag
 - The <embed> tag defines a container for an external resource, such as a web page, a picture, a media player, or a plug-in application.
 - <embed type="image/jpg" src="pic_trulli.jpg" width="300" height="200">
 - <embed type="text/html" src="snippet.html" width="500" height="200">
 - <embed type="video/webm" src="video.mp4" width="400" height="300">
 - Note: It is better to use , <iframe>, <video>, <audio> tag instead of <embed> tag

Some More Tags:

- <output> Tag
 - The <output> tag is used to represent the result of a calculation

Attribute	Value	Description
<u>for</u>	<i>element_id</i>	Specifies the relationship between the result of the calculation, and the elements used in the calculation
<u>form</u>	<i>form_id</i>	Specifies which form the output element belongs to
<u>name</u>	<i>name</i>	Specifies a name for the output element

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

<h1>The output element</h1>

<form oninput="x.value=parseInt(a.value)+parseInt(b.value)">
<input type="range" id="a" value="50">
+<input type="number" id="b" value="25">
=<output name="x" for="a b"></output>
</form>

<p><strong>Note:</strong> The output element is not supported in Edge 12 (or earlier).</p>

</body>
</html>
```

The output element



Note: The output element is not supported in Edge 12 (or earlier).

HTML <meta> Tag

- The <meta> tag defines metadata about an HTML document.
- Metadata is data (information) about data.
- <meta> tags always go inside the <head> element, and are typically used to specify character set, page description, keywords, author of the document, and viewport settings.
- Metadata will not be displayed on the page, but is machine parsable.
- Metadata is used by browsers (how to display content or reload page), search engines (keywords), and other web services.

Attributes of <meta> tag

Attribute	Value	Description
<u>charset</u>	<i>character_set</i>	Specifies the character encoding for the HTML document
<u>content</u>	<i>text</i>	Specifies the value associated with the http-equiv or name attribute
<u>http-equiv</u>	content-security-policy content-type default-style refresh	Provides an HTTP header for the information/value of the content attribute
<u>name</u>	application-name author description generator keywords viewport	Specifies a name for the metadata

HTML <meta> Tag

- 1. Keywords** - Define keywords for search engines
- 2. Refresh** - Refresh document every 30 seconds:
- 3. Viewport** - 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.
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.

```
<!DOCTYPE html>
<html>
<head>
  <meta · charset="UTF-8">
  <meta · name="keywords" · content="CSE, · BTech, · HTML, · CSS">
  <meta · name="description" · content="FSWD · Course">
    ...
  <meta · name="author" · content="Dr. · Susheela">
    ...
  <meta · name="viewport" · content="width=device-width, · initial-scale=1.0">
  <meta · http-equiv="refresh" · content="30">
</head>
<body>
```

<p>All meta information goes in the head section...</p>

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

Character entities in HTML

- Reserved characters or the characters that creates ambiguity in code must be replaced with character entities.
- Ex: less than (<) or greater than (>) signs
- Entity name or Entity number both can be used to display a special character

Copyright Mark - Decimal - ©, Hexadecimal - &#A9

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

<h2>HTML Entity Example</h2>

<h3>The less-than sign: &lt; &#60;</h3>
<h3>The greater-than sign: &gt; &#62;</h3>
<h3>The ampersand (and) sign: &amp;lt;&gt;</h3>
<h3>The check sign: &check; &#9745; &#10004;</h3>
<h3>The registered trade mark sign: &reg; &#174;</h3>
<h3>The copyright mark sign: &copy; &#174;</h3>

</body>
</html>
```

Registered trademark

HTML Entity Example

The less-than sign: <<

The greater-than sign: >>

The ampersand (and) sign: &

The check sign: ✓ ☑ ✓

The registered trade mark sign: ®

The copyright mark sign: © ®

HTML | <frame> Tag

- HTML Frames are used to divide the web browser window into multiple sections where each section can be loaded separately.
- A frameset tag is the collection of frames in the browser window.
- The `<frame>` tag was used in HTML 4 to define one particular window (frame) within a `<frameset>`
- Not Supported in HTML5.
- Use the `<iframe>` tag instead of `<frame>` tag in HTML5

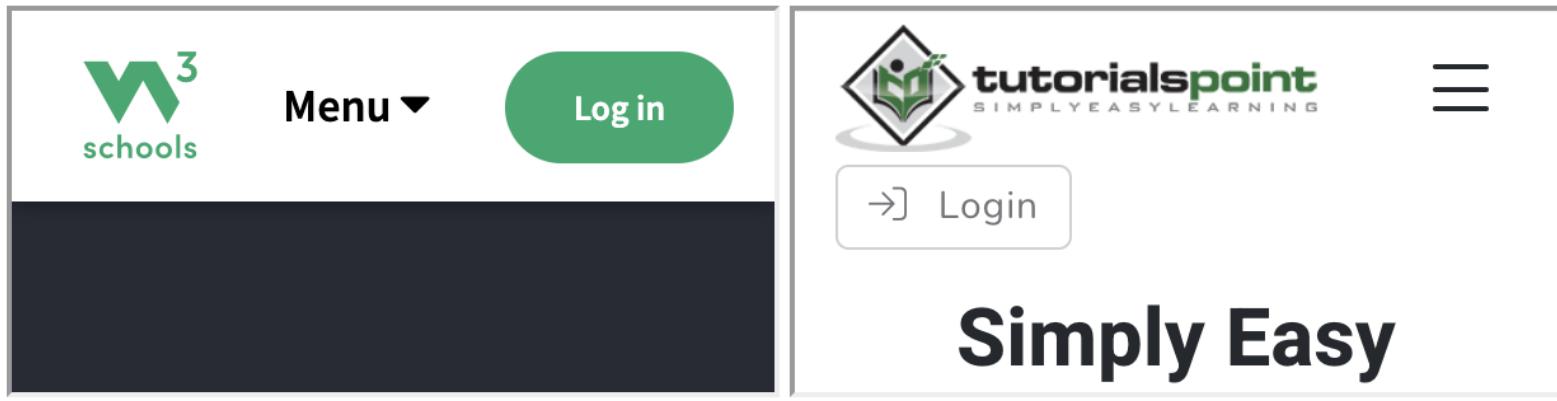
```
<!DOCTYPE html>
<html>
<body>

<h1>The iframe element</h1>

<iframe src="https://www.w3schools.com/" title="W3 School">
</iframe>
<iframe src="https://www.tutorialspoint.com/" title="Tutorialspoint">
</iframe>

</body>
</html>
```

The iframe element



The W3C Markup Validation Service

- Most pages on the World Wide Web are written in computer languages (such as HTML) that allow Web authors to structure text, add multimedia content, and specify what appearance, or style, the result should have.
- As for every language, these have their own grammar, vocabulary and syntax, and every document written with these computer languages are supposed to follow these rules. The (X)HTML languages, for all versions up to XHTML 1.1, are using machine-readable grammars called document type definition (DTDs), a mechanism inherited from SGML.
- However, Just as texts in a natural language can include spelling or grammar errors, documents using Markup languages may (for various reasons) not be following these rules. The process of verifying whether a document actually follows the rules for the language(s) it uses is called validation, and the tool used for that is a validator. A document that passes this process with success is called valid.
- "Markup Validation" is the process of checking a Web document against the grammar (generally a DTD) it claims to be using.
- The W3C (World Wide Web Consortium) markup validation service is used to checks the markup validity of Web documents in HTML, XHTML, SMIL, MathML, etc.
- The W3C Markup Validator provides Perl/CGI/SGML/XML/DTD-based validation of a variety of document types.

The W3C Markup Validation Service

- The Markup Validator is a free tool and service that validates markup: in other words, it checks the syntax of Web documents, written in formats such as (X)HTML.
- The Validator compares your HTML document to the defined syntax of HTML and reports any discrepancies.
- The Markup Validator is maintained at W3C by W3C staff and benevolent collaborators, who receive a lot of help from contributors (read the full credits).
- Why Web professionals should choose to validate:
 - WYSINWOG - What You See Is Not What Others Get
 - Validation as a debugging tool
 - Validation as a future-proof quality check
 - Validation eases maintenance
 - Validation helps teach good practices
 - Validation is a sign of professionalism



Markup Validation Service

Check the markup (HTML, XHTML, ...) of Web documents

Validate by URI

Validate by File Upload

Validate by Direct Input

Validate by URI

Validate a document online:

Address:

▼ More Options

Character Encoding

(detect automatically)



Only if missing

Document Type

(detect automatically)



Only if missing

List Messages Sequentially Group Error Messages by Type

Show Source

Clean up Markup with HTML-Tidy

Show Outline

Validate error pages

Verbose Output

Check

References

- <https://www.geeksforgeeks.org/>
- <https://www.w3schools.com/>
- <https://www.tutorialspoint.com/>
- <https://data-flair.training/blogs/web-browsers/>
- <https://dev.to/mahmoudessam/the-architecture-of-web-browsers-1o1k>
- <https://xd.adobe.com/ideas/process/information-architecture/different-types-of-website-structures/>
- <https://www.infidigit.com/blog/website-structure/>
- <https://www.uxpin.com/studio/blog/web-structures-explained/>
- <https://www.naukri.com/learning/articles/top-features-of-html5-with-examples/>
- <https://www.admecindia.co.in/web-design/10-new-features-html5-examples/>
- https://validator.w3.org/docs/why.html#why_pros