

HTML

Hypertext Markup Language

- Ruchita Tailor

Introduction to HTML

- HTML stands for Hypertext Markup Language
- HTML is the basic building block of World Wide Web. It is the most widely used language to write Web Pages.
- HTML was originally developed by Tim Berners-Lee in 1990. He is also known as the father of the web. In 1996, the World Wide Web Consortium (W3C) became the authority to maintain the HTML specifications.

Introduction to HTML (1)

- **Hypertext** refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext.
- Hypertext is text displayed on a computer or other electronic device with references to other text that the user can immediately access, usually by a mouse click or key press.
- Apart from text, hypertext may contain tables, lists, forms, images, and other presentational elements.

Introduction to HTML (2)

- HTML is a **Markup Language** which means you use HTML to simply "mark-up" a text document with tags that tell a Web browser how to structure it to display.
- It is language of writing layout information within documents.
- It is used to define the layout and attributes of a World Wide Web document as well as to create links between web pages.
- Markup languages use sets of markup tags to characterize text elements within a document, which gives instructions to the web browsers on how the document should appear.

What You Can Do with HTML

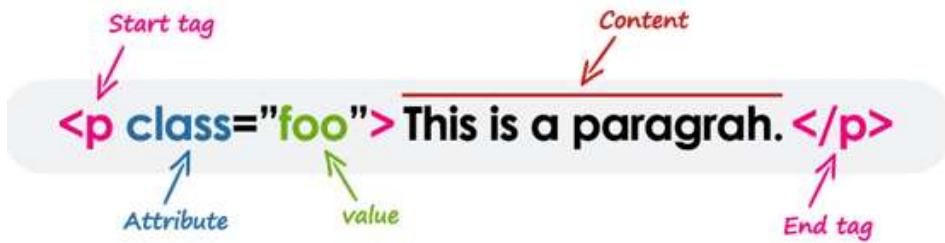
- You can publish documents online with text, images, lists, tables, etc.
- You can access web resources such as images, videos or other HTML document via hyperlinks.
- You can create forms to collect user inputs like name, e-mail address, comments, etc.
- You can include images, videos, sound clips, flash movies, applications and other HTML documents directly inside an HTML document.
- You can store data in the user's web browser and access later on.
- You can find the current location of your website's visitor.

Introduction to HTML (4)

- **Tags** can be defined as the instructions which are being directly embedded in the text of an HTML document.
- The types of tags used in the HTML document are responsible to tell a web browser to do something (follow the instruction) instead of just displaying text.
- 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.
- **Paired tags:**
 - The tag consists of an opening tag and a closing tag as its companion tag.
 - Eg: <p> paragaraph content</p>
 - <i> </i> ,
- **Unpaired tags/ Self-contained:**
 - The tag only has an opening tag and does not have a closing tag or a companion tag.
 - Eg: <hr>

Introduction to HTML (5)

- **Attributes** contain additional pieces of information. Attributes take the form of an opening tag and additional info is placed inside.



- In HTML, tag and attribute names are not case-sensitive.
- An HTML comment begins with `<!--`, and ends with `-->`

Basic HTML Structure

```
<!DOCTYPE html>
<html>

  <head>
    <title> <!-- title bar --> </title>
    <!-- header for the website -->
  </head>

  <body>
    <!-- body section of the website -->
  </body>

</html>
```

- **DOCTYPE:** specifies the document type.
 - Specified by DTD (Document Type Definition)
 - HTML syntax rules are specified by the file xhttp11.dtd file
 - 3 types of HTML DTD's :
 1. HTML 1.0 Strict:
 2. HTML 1.0 Transitional:
 3. HTML 1.0 Frameset:
- **<head>**
 - This section contains information about the page, such as its title, meta tags, and where to locate the CSS file.
 - Metadata is mostly meant for search engines and other computer programs.
- **<body>**
 - This is where the **content of the page** goes.

Formatting and Fonts

- **Heading Tags:**
 - Display the text as some header.
- Eg:

```
<!DOCTYPE html>
<html>

<head>
  <title>Heading Example</title>
</head>

<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>

</html>
```

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

Formatting and Fonts (1)

- **Paragraph Tag <p>**
 - Places a blank line before the line it is on.
 - Way of defining breaks

```
<!DOCTYPE html>
<html>

<head>
  <title>Paragraph Example</title>
</head>

<body>
  <p>Here is a first paragraph of text.</p>
  <p>Here is a second paragraph of text.</p>
  <p>Here is a third paragraph of text.</p>
</body>

</html>
```

Here is a first paragraph of text.

Here is a second paragraph of text.

Here is a third paragraph of text.

Formatting and Fonts (2)

- **Line Break Tag
**

- Whenever you use the
 element, anything following it starts from the next line.
- To insert blank lines
- Don't want line break <nobr> tag can be used.

```
<!DOCTYPE html>
<html>

<head>
    <title>Line Break Example</title>
</head>

<body>
    <p>Hello<br > You delivered your assignment on time.<br />
        Thanks<br >
        Buddy</p>
</body>

</html>
```

Hello
You delivered your assignment on time.
Thanks
Buddy

Formatting and Fonts (3)

- <center>

```
<!DOCTYPE html>
<html>

<head>
    <title>Centring Content
Example</title>
</head>

<body>
    <p>This text is not in the center.</p>

    <center>
        <p>This text is in the center.</p>
    </center>
</body>

</html>
```

This text is not in the center.

This text is in the center.

Formatting and Fonts (4)

- **Horizontal Lines: <hr>**

- The <hr> element is most often displayed as a horizontal rule that is used to separate content (or define a change) in an HTML page.

```
<!DOCTYPE html>
<html>

<head>
  <title>Horizontal Line Example</title>
</head>

<body>
  <p>This is paragraph one and should be on top</p>
  <hr>
  <p>This is paragraph two and should be at bottom</p>
</body>

</html>
```

This is paragraph one and should be on top

This is paragraph two and should be at bottom

Formatting and Fonts (5)

- **<pre>**
 - Preserve the white space and lines in the text.
 - Text in a <pre> element is displayed in a fixed-width font, and the text preserves both spaces and line breaks. The text will be displayed exactly as written in the HTML source code.
- **Output:**

The pre element

Text in a pre element
is displayed in a fixed-width
font, and it preserves
both spaces and
line breaks

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

<h1>The pre element</h1>

<pre>
Text in a pre element
is displayed in a fixed-width
font, and it preserves
both   spaces and
line breaks
</pre>

</body>
</html>
```

Formatting and Fonts (6)

- **<div>**
 - The <div> tag defines a division or a section in an HTML document.
 - The <div> tag is used as a container for HTML elements - which is then styled with CSS or manipulated with JavaScript.

The div element

This is a heading in a div element

This is some text in a div element.

This is some text outside the div element.

```
<!DOCTYPE html>
<html>
<head>
<style>
.myDiv {
border: 5px outset red;
background-color: lightblue;
text-align: center;
}
</style>
</head>
<body>

<h1>The div element</h1>

<div class="myDiv">
<h2>This is a heading in a div element</h2>
<p>This is some text in a div element.</p>
</div>

<p>This is some text outside the div element.</p>

</body>
</html>
```

Formatting and Fonts (7): Adding emphasis

- **.. **
- *<i>...</i>*
- **...**
- ~~<strike>...</strike>~~
- **<center>...</center>**

```
<!DOCTYPE html>
<html>
<body>
<p>This is normal text - <b>and this is bold text</b>.</p>
<p>This is normal text - <i>and this is italics text</i>.</p>
<p>This is normal text - <strong>and this is strong text</strong>.</p>
<p>This is normal text - <strike>and this is striked text</strike>.</p>
<center> This is center text</center>.
</body>
</html>
```

- **Output:**

This is normal text - **and this is bold text.**

This is normal text - *and this is italics text.*

This is normal text - **and this is strong text.**

This is normal text - ~~and this is striked text.~~

This is center text

Formatting and Fonts (8): Adding emphasis(2)

- <tt>..</tt>
- *..*
- <blink>...</blink>
- ^{^{...}}
- _{_{...}}

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

```
<head>
  <title> Example</title>
</head>

<body>
  <tt>Typewriter effect</tt> <br><br>
  <em> Emphasis</em><br><br>
  <blink> Blinking effect</blink><br><br>
  a<sup> 3</sup><br><br>
  H<sub>2</sub>O<br><br>
</body>

</html>
```

Typewriter effect

Emphasis

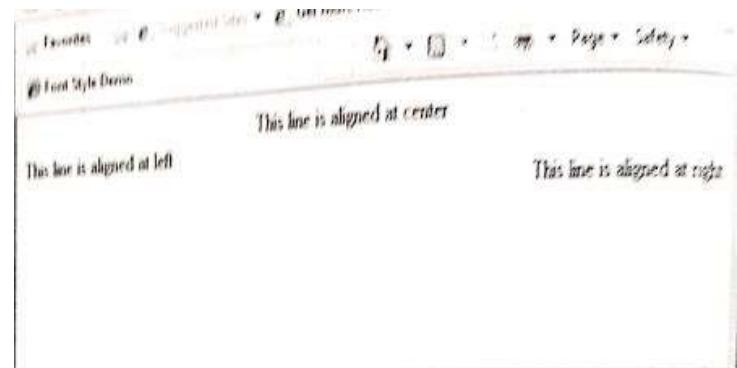
Blinking effect

a³

H₂O

Formatting and Fonts (9): Text Alignment

- Using `<div>` tag:
 - `<div align="center">`
Center`</div>`
 - `<div align="left">`
Left`</div>`
 - `<div align="right">`
Right`</div>`



Formatting and Fonts (10): Setting the Font

- <basefont> - IE only
 - Set font, size and color of the text in the web page.
 - Face
 - Color
 - size
- <basefont face="arial" size="10">

Formatting and Fonts (11): Font sizes

- Ways to change font size:
 -
 - Font size running from 2 to 7
 - <h1> to <h6> tags

Font size	Heading	Point size
7	-	36 pt
6	<h1>	24 pt
5	<h2>	18 pt
	<h3>	12 pt
4	<h4>	12 pt bold
3	body text	12 pt plain
	<h5>	10 pt
	<h6>	7 pt
2	-	9 pt

Table 2-2 Font comparison

Formatting and Fonts (12): Special Characters

- This entity reference is useful for defining the space between two strings and informing browser for not performing the word wrapper between the strings.

Character	Meaning	Entity Reference
<	less than	<
>	greater than	>
&	ampersand	&
"	double quote	"
'	apostrophes	'
©	copyright	©

- non breaking space character

Color

- Add color:
 - Background color of the browser
 - Text
- Background color
 - <body bgcolor= value text=value>
- Change the color of text:
 -

Color (2)

1. You can specify a color by using RGB value
 - For example rgb(255,255,255) indicates white.

2. You can specify a color by using color name.
 - For example white indicates white.

Aqua	Navy
Black	Olive
Blue	Purple
Fuchsia	Red
Gray	Silver
Green	Teal
Lime	White
Maroon	Yellow

Color	Color HEX	Color RGB
Black	#000000	rgb(0,0,0)
Red	#FF0000	rgb(255,0,0)
Green	#00FF00	rgb(0,255,0)
Blue	#0000FF	rgb(0,0,255)
Lime	#FFFF00	rgb(255,255,0)
Teal	#00FFFF	rgb(0,255,255)
White	#FF00FF	rgb(255,0,255)
Yellow	#C0C0C0	rgb(192,192,192)
	#FFFFFF	rgb(255,255,255)

3. You can specify a color by using hex color value.
 - For example #FFFFFF indicates white.
 - Each pair of digits specifies 0 to 255 color values
 - Hexadecimal color coding helps us to specify
 - $256 \times 256 \times 256 = 16,777,216$ colors

R	G	B	Color produced
00	00	00	Black
FF	00	00	Bright red
00	FF	00	Bright green
00	00	FF	Bright blue
80	00	00	Dark red
00	80	00	Dark green
00	00	80	Dark blue
FF	FF	00	Bright yellow
80	80	00	Brown
FF	00	FF	Magenta
80	00	80	Indigo
00	80	80	Turquoise
80	80	80	Grey
FF	FF	FF	White

Table 2-4 Some example colors with codes

Images

-
 - Attributes:
 - src =“filename | path | web_address | gif “
 - alt =“text”
 - align =“bottom(default) | middle | top”
 - border = n
 - hspace = n (pixels)
 - vspace =n (pixels) //vertical white space (top & bottom sides)
 - height = n (pixels) //horizontal white space (left & right sides)
 - width = n (pixels)

Images (1)

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

```
<h2>HTML Image</h2>  

```

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

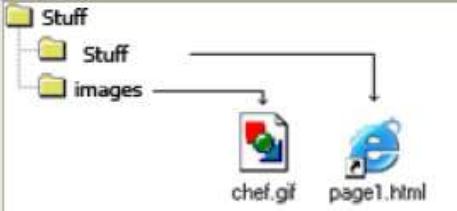
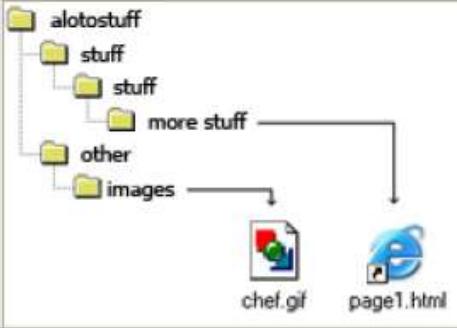
HTML Image



Images (1) -src

<p>A diagram showing a 'My Stuff' folder containing a 'chef.gif' file and a 'page1.html' file. Arrows point from both files to the 'chef.gif' icon.</p>	<p>src="chef.gif" means that the image is in the same folder as the html document calling for it.</p>
<p>A diagram showing a 'My Stuff' folder containing an 'images' folder. The 'images' folder contains a 'chef.gif' file. A 'page1.html' file is shown outside the 'My Stuff' folder. Arrows point from both files to the 'chef.gif' icon.</p>	<p>src="images/chef.gif" means that the image is one folder down from the html document that called for it. This can go on down as many layers as necessary.</p>
<p>A diagram showing an 'images' folder containing a 'My Stuff' folder. The 'My Stuff' folder contains a 'page1.html' file. A 'chef.gif' file is shown outside the 'My Stuff' folder. Arrows point from both files to the 'chef.gif' icon.</p>	<p>src="..../chef.gif" means that the image is in one folder up from the html document that called for it.</p>

Images (2)-src

	<p><code>src=".../.../chef.gif"</code> means that the image is two folders up from the html document that called for it.</p>
	<p><code>src=".../images/chef.gif"</code> means that the image is one folder up and then another folder down in the images directory.</p>
	<p><code>src=".../.../.../other/images/chef.gif"</code> means this goes multiple layers up.</p>

Images (2) : alt

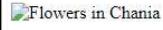
```
<!DOCTYPE html>
<html>
<body>
<p>If a browser cannot find the image, it will display the alternate text:</p>
```

```

```

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

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



Images (3) : Hspace and vspace

-
-

```
<p>

some text some text<br>some text some text<br>some text ...
</p>
```

Output



some text some text
some text some text
some text some text
some text some text

```
<p>

some text some text<br>some text some text<br>some text ...
</p>
```

Output



some text some text
some text some text

Images (4): hspace and vspace

-

```
<p>

some text some text<br>some text some text<br>some text ...
</p>
```

Output



some text some text

some text some text
some text some text
some text some text
some text some text
some text some text
some text some text

Ruchita Tailor- HTML

30

Images (5): align

align=""	vertical alignment	
	top	aligns to the top
	middle	aligns to the middle
	bottom	aligns to the bottom (default)
align=""	floats image	
	left	the image floats to the left (the text wraps to the right of the image)
	right	the image floats to the right (the text wraps to the left of the image)

Images (6): align

-

Aligns to the top

```
<p>

Aligns to the top
<br>
The text aligns to the top of the image.
</p>
```

Output



Aligns to the top

The text aligns to the top of the image.

Images (7): align

-

Aligns to the middle

```
<p>

Aligns to the middle
<br>
The text aligns to the middle of the image.
</p>
```

Output



Aligns to the middle

The text aligns to the middle of the image.

Images (8): align

-

Aligns to the bottom

```
<p>

Aligns to the bottom (Default)
<br>
The text aligns to the bottom of the image.
</p>
```

Output



Aligns to the bottom (Default)
The text aligns to the bottom of the image.

Images (9): align

Aligns to the left

```
• <p>  
  
Aligns to the left  
<br>  
The image floats to the left.  
<br>  
And the text wraps to the right of the image.  
<br clear="left">  
Clears the left  
</p>
```

Output



Aligns to the left
The image floats to the left.
And the text wraps to the right of the image.

Clears the left

Images (10): align

- **Aligns to the right**

```
<p>

Aligns to the right
<br>
The image floats to the right.
<br>
And the text wraps to the left of the image.
<br clear="right">
Clears the right
</p>
```

Output

Aligns to the right
The image floats to the right.
And the text wraps to the left of the image.

Clears the right



Background Images

- `<body background =“ “>`
 - Image smaller than the window will be tiled over the area of the browser screen.

– Eg:

```
<!DOCTYPE html>
```

```
<html>
```

```
    <body background="edureka.png">
```

```
        <h1>Welcome to Edureka</h1>
```

```
    </body>
```

```
</html>
```



Hyperlinks

- The link acts as a pointer to some web page or some resources
- It allows page to link logically with other page.
- **Syntax:**
 - ` Link `
- **Link**
 - `` links to another document
 - `` the target window of the link
- **A named anchor**
 - `` links to a named anchor
 - `` links to a named anchor in another document
- **E-mail sending**
 - `` opens the email client
- **Use of Images as a link:**
` `

List

- List is the collection of items or elements.
- Types of lists:
 - Ordered List
 - Unordered list
 - Definition list
- Nested lists

Lists (1) - Unordered List

- Unordered List

Attribute	Value	Explanation
type=""	disc	• disc styles
	circle	○ circle styles
	square	§ square styles

```
<ul>
<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>
</ul>
```

Output

- First list item
- Second list item
- Third list item

```
<ul>
<li type="disc">Disc styles</li>
<li type="circle">Circle styles</li>
<li type="square">Square styles</li>
</ul>
```

Output

- Disc styles
- Circle styles
- Square styles

Lists (2) - Ordered List

- Ordered List

Attribute	Value	Explanation
type=""	1	arabic numeral
	A	alphabet (capital letter)
	a	alphabet (small letter)
	I	roman numeral (capital letter)
	i	roman numeral (small letter)

```
<ol>
<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>
</ol>
```

Output

1. First list item
2. Second list item
3. Third list item

```
<ol>
<li type="1">Arabic numeral</li>
<li type="A">Alphabet (capital letter)</li>
<li type="a">Alphabet (small letter)</li>
<li type="I">Roman numeral (capital letter)</li>
<li type="i">Roman numeral (small letter)</li>
</ol>
```

Output

1. Arabic numeral
- B. Alphabet (capital letter)
- c. Alphabet (small letter)
- IV. Roman numeral (capital letter)
- v. Roman numeral (small letter)

Ruchita Tailor vHIT

51

Lists (3)- Ordered List (1)

- The start attribute of the OL element specifies the starting number of the first item.

Attribute	Value	Explanation
<code>start=""</code>	number	the number to start on

```
<ol start="5">
<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>
</ol>
```

Output

- 5. First list item
- 6. Second list item
- 7. Third list item

```
<ol type="A" start="5">
<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>
</ol>
```

Output

- E. First list item
- F. Second list item
- G. Third list item

Lists (4) - Ordered List (2)

- The value attribute of the li element specifies the number of the list item.

Attribute	Value	Explanation
value=""	number	the number of the list item

```
<ol>
<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>
<li value="7">Fourth list item</li>
<li>Fifth list item</li>
<li>Sixth list item</li>
</ol>
```

Output

1. First list item
2. Second list item
3. Third list item
7. Fourth list item
8. Fifth list item
9. Sixth list item

```
<ol type="I">
<li>First list item</li>
<li>Second list item</li>
<li>Third list item</li>
<li value="7">Fourth list item</li>
<li>Fifth list item</li>
<li>Sixth list item</li>
</ol>
```

Output

- I. First list item
- II. Second list item
- III. Third list item
- VII. Fourth list item
- VIII. Fifth list item
- IX. Sixth list item

Lists (5) – Definition List

- The DL element defines a definition list, and the DT and DD elements are used to define term and description.
- Syntax:

– <dl>
 <dt>definition term</dt>
 <dd>definition description</dd>
 </dl>

```
<dl>
<dt>HTML</dt>
<dd>Hypertext Markup Language</dd>
<dt>CGI</dt>
<dd>Common Gateway Interface</dd>
</dl>
```

Output

HTML	Hypertext Markup Language
CGI	Common Gateway Interface

Ruchita Tailor- HTML

Lists (6): Nested List

- List inside a list

```
<ul>
<li>Fruit
<ul>
<li>Bananas</li>
<li>Apples
<ul>
<li>Green</li>
<li>Red</li>
</ul>
</li>
<li>Pears</li>
</ul>
</li>
<li>Vegetables</li>
<li>Meat</li>
</ul>
```

- Fruit
 - Bananas
 - Apples
 - Green
 - Red
 - Pears
- Vegetables
- Meat

Table

- Data arranged in a tabular form; i.e. an area is split into columns and rows to show association.
- Syntax:

```
<table>
    <tr>
        <td> content of table </td>
    </tr>
</table>
```

	<td>	<td>	<td>
<tr>	Content	Content	Content
<tr>	Content	Content	Content
<tr>	Content	Content	Content

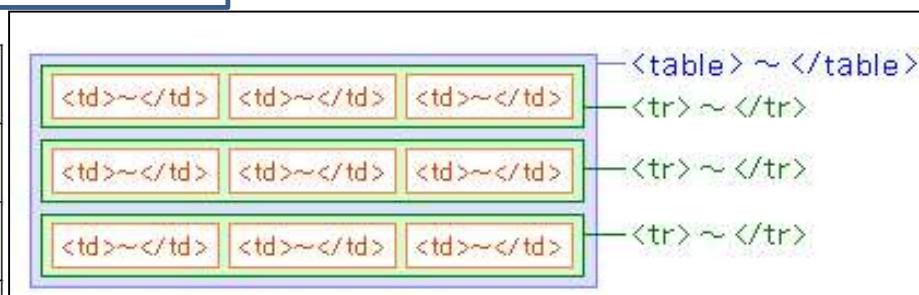
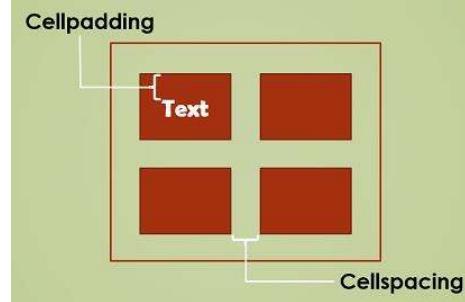


Table (1)

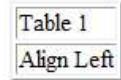
- **Border:** sets the border width in pixels around the table.

<table border=""> table border



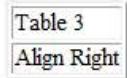
- **Cell spacing:** Sets the spacing in pixels between the inner and outer borders.

<table cellspacing=""> spacing between cells



- **Cell padding:** Distance in pixels between the inner border and the text.

<table cellpadding=""> spacing within cells



- **Width:** set the width of the cell as percentage of the browser window

<table width=""> table width

Horizontal Align (Align)



- **Align:** puts data at the left, middle or right of the cell.

- **Valign:** puts data at the top, middle or bottom of the cell.

Vertical Align (VAlign)



- **<caption>** table caption

<caption align=""> alignment of the caption

Table (2)

A table of one row

```
<table border="3">
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
</table>
```

Output

Row1 - Col1	Row1 - Col2	Row1 - Col3
-------------	-------------	-------------

A table of two rows

```
<table border="3">
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
</table>
```

Output

Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3

Table (3) - th

- **<th>**: defines a table header cell.
- The text within the TH element displayed in bold.

Column heading

```
<table border="1" width="100%">
<tr>
<th>Column heading</th>
<th>Column heading</th>
<th>Column heading</th>
</tr>
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
</table>
```

Output

Column heading	Column heading	Column heading
Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3

Table (4) – th (1)

- <th> with row heading:

Row heading

```
<table border="1" width="100%">
<tr>
<th>Row heading</th>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
</tr>
<tr>
<th>Row heading</th>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
</tr>
<tr>
<th>Row heading</th>
<td>Row3 - Col1</td>
<td>Row3 - Col2</td>
</tr>
</table>
```

Output

Row heading	Row1 - Col1	Row1 - Col2
Row heading	Row2 - Col1	Row2 - Col2
Row heading	Row3 - Col1	Row3 - Col2

Table (5) - border

- **Border = n**
- The value can be omitted. (The border width becomes the same as size 1.)

The border size is 10

```
<table border="10">
<tr>
<td>Cell A</td>
<td>Cell B</td>
<td>Cell C</td>
</tr>
</table>
```

Output

Cell A	Cell B	Cell C
--------	--------	--------

The border size is 0

```
<table border="0">
<tr>
<td>Cell A</td>
<td>Cell B</td>
<td>Cell C</td>
</tr>
</table>
```

Output

Cell A Cell B Cell C

```
<table>
```

```
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>

<tr>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
```

```
</table>
```

Output

Row1 - Col1 Row1 - Col2 Row1 - Col3
Row2 - Col1 Row2 - Col2 Row2 - Col3

The border size is 1

```
<table border="1">
<tr>
<td>Cell A</td>
<td>Cell B</td>
<td>Cell C</td>
</tr>
</table>
```

Output

Cell A	Cell B	Cell C
--------	--------	--------

```
<table border>
<tr>
<td>Cell A</td>
<td>Cell B</td>
<td>Cell C</td>
</tr>
</table>
```

Output

Cell A	Cell B	Cell C
--------	--------	--------

Table (6) - caption

- **Caption:** defines a caption for a table.
- Placed immediately after the TABLE start tag.
- Syntax:
 - <caption>table caption</caption>
 - <caption align="top">table caption</caption>

Attribute	Value	Explanation
align=""	top	positions the caption at the top of the table (default)
	bottom	positions the caption at the bottom of the table

Table (7) – caption (1)

- <caption>

```
<table border="1">
<caption>Table caption</caption>
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
</table>
```

Output

Table caption		
Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3

bottom

```
<table border="1">
<caption align="bottom">Table caption</caption>
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
</table>
```

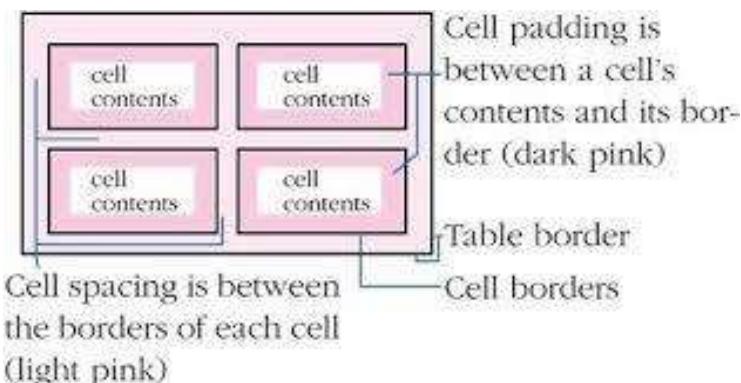
Output

Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3

Table caption

Table (8) - cellspacing

- **Cellspacing:**
 - specifies the space between cells.
- `<table cellspacing="value">`



The cellspacing is 0

```
<table border="1" cellspacing="0">
<tr>
<td>Cell A</td>
<td>Cell B</td>
<td>Cell C</td>
</tr>
</table>
```

Output

Cell A	Cell B	Cell C
--------	--------	--------

The cellspacing is 5

```
<table border="1" cellspacing="5">
<tr>
<td>Cell A</td>
<td>Cell B</td>
<td>Cell C</td>
</tr>
</table>
```

Output

Cell A	Cell B	Cell C
--------	--------	--------

The cellspacing is 15

```
<table border="1" cellspacing="15">
<tr>
<td>Cell A</td>
<td>Cell B</td>
<td>Cell C</td>
</tr>
</table>
```

Output

Cell A	Cell B	Cell C
--------	--------	--------

Table (9) – height and width

- Height and Width:

Table width is 100 pixel
Table width is 100 %

The width is 280 pixels
and the height is 100 pixels

```
<table border="1" width="280" height="100">  
<tr>  
<td>Cell A</td>  
<td>Cell B</td>  
<td>Cell C</td>  
</tr>  
</table>
```

Output

Cell A	Cell B	Cell C
--------	--------	--------

The width is not specified

```
<table border="1">  
<tr>  
<td>Cell A</td>  
<td>Cell B</td>  
<td>Cell C</td>  
</tr>  
</table>
```

Output

Cell A	Cell B	Cell C
--------	--------	--------

The width is 100%

```
<table border="1" width="100%">  
<tr>  
<td>Cell A</td>  
<td>Cell B</td>  
<td>Cell C</td>  
</tr>  
</table>
```

Output

Cell A	Cell B	Cell C
--------	--------	--------

Table (10) – height and width (1)

- Height and width:
 - The width and height attributes of the TD (TH) element specifies the width and height of a cell.

Width (Pixels)

```
<table border="1">
<tr>
<td width="150">150px</td>
<td width="200">200px</td>
<td width="250">250px</td>
</tr>
</table>
```

Output

150px	200px	250px
-------	-------	-------

Table (11) – height and width (2)

Width and Height (Pixels)

```
<table border="1">
<tr>
<td width="150" height="100">150px * 100px</td>
<td width="200">200px * 100px</td>
<td width="250">250px * 100px</td>
</tr>
</table>
```

Output

150px * 100px	200px * 100px	250px * 100px
---------------	---------------	---------------

Table (12) - height and width (3)

Width and Height (Percentage)

```
• <table border="1" width="100%" height="200">
  <tr>
    <td width="30%" height="30%">30% * 30%</td>
    <td width="70%">70% * 30%</td>
  </tr>
  <tr>
    <td height="70%">30% * 70%</td>
    <td>70% * 70%</td>
  </tr>
</table>
```

Output

30% * 30%	70% * 30%
30% * 70%	70% * 70%

Table (13) - Alignment

- **Alignment:**

- The **align** and **valign** attributes of the TR and TD (TH) elements specifies the alignment of cell content.
- <tr align="" valign=""><td align="" valign="">
 - <tr align="center" valign="top"> : Applied to all cells in a row.
 - <td align="center" valign="top"> : Applied to one cell.

Attribute	Value	Explanation
align=" "	horizontal alignment in cell	
	left	aligns to the left
	center	aligns to the center
	right	aligns to the right
valign=" "	vertical alignment in cell	
	top	aligns to the top
	middle	aligns to the middle
	bottom	aligns to the bottom

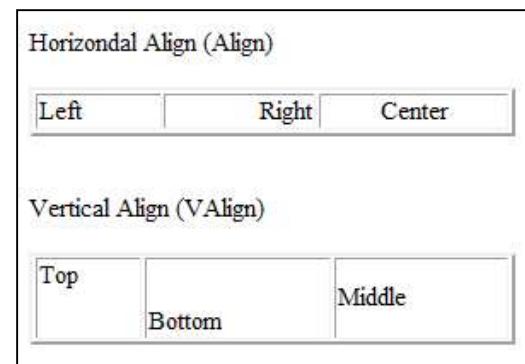


Table (14) – Alignment(1)

- Horizontal Alignment:

Horizontal alignment

```
<table border="1" width="100%" height="100">
<tr>
<td width="25%">Default</td>
<td align="left" width="25%">Left</td>
<td align="center" width="25%">Center</td>
<td align="right" width="25%">Right</td>
</tr>
</table>
```

Output

Default	Left	Center	Right
---------	------	--------	-------

Table (15) – Alignment (2)

- Vertical Alignment:

Vertical alignment

```
<table border="1" width="100%" height="100">
<tr>
<td width="25%">Default</td>
<td valign="top" width="25%">Top</td>
<td valign="middle" width="25%">Middle</td>
<td valign="bottom" width="25%">Bottom</td>
</tr>
</table>
```

Output

Default	Top	Middle	Bottom
---------	-----	--------	--------

Table (16) – Alignment (3)

- Horizontal and Vertical alignment:

Horizontal and Vertical alignment

```
<table border="1" width="100%" height="100">
<tr align="center">
<td width="25%">Center</td>
<td valign="top" width="25%">Center - Top</td>
<td valign="middle" width="25%">Center - Middle</td>
<td valign="bottom" width="25%">Center - Bottom</td>
</tr>
</table>
```

Output

Center	Center - Top	Center - Middle	Center - Bottom
--------	--------------	-----------------	-----------------

Table (17) – Alignment (4)

- The **align** attribute of the TABLE element specifies the alignment of the table.
- <table align="value">

Attribute	Value	Explanation
align=""	left	the table floats to the left (the text wraps to the right of the table)
	right	the table floats to the right (the text wraps to the left of the table)
	center	centers the table

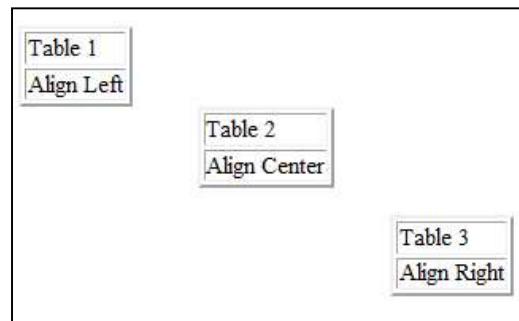


Table (18) – Alignment (5)

Aligns to the left

```
<table border="1" align="left">
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
<tr>
<td>Row3 - Col1</td>
<td>Row3 - Col2</td>
<td>Row3 - Col3</td>
</tr>
</table>

The table floats to the left.
<br>
And the text wraps to the right of the table.
<br clear="left">
Clears the left
```

Output

Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3
Row3 - Col1	Row3 - Col2	Row3 - Col3

The table floats to the left.
And the text wraps to the right of the table.

Clears the left

Table (19) – Alignment (6)

• Aligns to the right

```
<table border="1" align="right">
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
<tr>
<td>Row3 - Col1</td>
<td>Row3 - Col2</td>
<td>Row3 - Col3</td>
</tr>
</table>
```

The table floats to the right.

And the text wraps to the left of the table.

<br clear="right">

Clears the right

Output

The table floats to the right.

And the text wraps to the left of the table.

Clears the right

Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3
Row3 - Col1	Row3 - Col2	Row3 - Col3

Ruchita Tailor- HTML

Table(20) – Alignment(7)

-

Aligns to the center

```
<table border="1" align="center">
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
<tr>
<td>Row3 - Col1</td>
<td>Row3 - Col2</td>
<td>Row3 - Col3</td>
</tr>
</table>
```

Output

Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3
Row3 - Col1	Row3 - Col2	Row3 - Col3

Table (21) – colspan and rowspan

- The **colspan** attribute of the TD (TH) element specifies the number of columns spanned by the cell. It is the attribute used in table header cells and data cells to merge columns.

1st Column	2nd Column
Mergin Two Columns	

- The **rowspan** attribute specifies the number of rows spanned by the cell. It is the attribute used in table header cells and data cells to merge rows.
- <td colspan="3" rowspan="3">

1st row	Merging
2nd row	Three
3rd row	Rows

Table (22) - colspan and rowspan (1)

Column spanning

```
<table border="1">
<tr>
<td colspan="3">Column spanning</td>
</tr>
<tr>
<td>Col1</td>
<td>Col2</td>
<td>Col3</td>
</tr>
</table>
```

Output

Column spanning		
Col1	Col2	Col3

Row spanning

```
<table border="1">
<tr>
<td rowspan="3">Row spanning</td>
<td>Row1</td>
</tr>
<tr>
<td>Row2</td>
</tr>
<tr>
<td>Row3</td>
</tr>
</table>
```

Output

Row spanning	Row1
	Row2
	Row3

Table (23)- colspan and rowspan (2)

Column and Row spanning

```
<table border="1">
<tr>
<td colspan="3" rowspan="3">Column and Row spanning</td>
<td>Row1 - Col4</td>
</tr>
<tr>
<td>Row2 - Col4</td>
</tr>
<tr>
<td>Row3 - Col4</td>
</tr>
<tr>
<td>Row4 - Col1</td>
<td>Row4 - Col2</td>
<td>Row4 - Col3</td>
<td>Row4 - Col4</td>
</tr>
</table>
```

Output

Column and Row spanning			Row1 - Col4
			Row2 - Col4
			Row3 - Col4
Row4 - Col1	Row4 - Col2	Row4 - Col3	Row4 - Col4

Table(24) – Background color

- Background Color to the table:
- `<table bgcolor="#0080ff">` : Applied to an entire table.
- `<tr bgcolor="#0080ff">` : Applied to all cells in a row.
- `<td bgcolor="#0080ff">` : Applied to one cell.

Table (25) - Background color (1)

Background color of the table

```
<table border="1" bgcolor="#80ffff">
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr>
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
<tr>
<td>Row3 - Col1</td>
<td>Row3 - Col2</td>
<td>Row3 - Col3</td>
</tr>
</table>
```

Output

Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3
Row3 - Col1	Row3 - Col2	Row3 - Col3

Background color of the row

```
<table border="1">
<tr>
<td>Row1 - Col1</td>
<td>Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr bgcolor="#ffff80">
<td>Row2 - Col1</td>
<td>Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
<tr>
<td>Row3 - Col1</td>
<td>Row3 - Col2</td>
<td>Row3 - Col3</td>
</tr>
</table>
```

Output

Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3
Row3 - Col1	Row3 - Col2	Row3 - Col3

Table (26) - Background color(2)

Background color of the cells

```
<table border="1">
<tr>
<td>Row1 - Col1</td>
<td bgcolor="#80ff80">Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr>
<td>Row2 - Col1</td>
<td bgcolor="#80ff80">Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
<tr>
<td>Row3 - Col1</td>
<td bgcolor="#80ff80">Row3 - Col2</td>
<td>Row3 - Col3</td>
</tr>
</table>
```

Output

Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3
Row3 - Col1	Row3 - Col2	Row3 - Col3

Combination

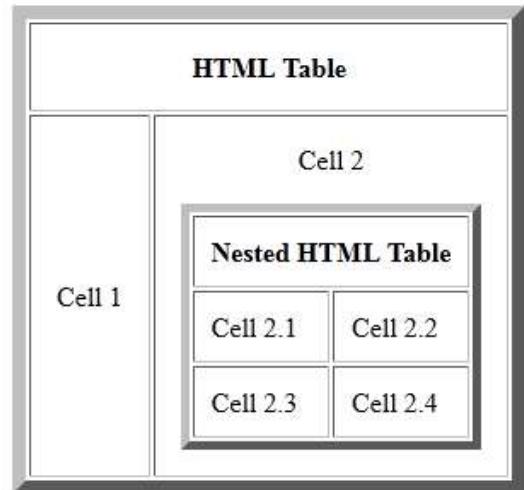
```
<table border="1" bgcolor="#80ffff">
<tr>
<td>Row1 - Col1</td>
<td bgcolor="#80ff80">Row1 - Col2</td>
<td>Row1 - Col3</td>
</tr>
<tr bgcolor="#ffff80">
<td>Row2 - Col1</td>
<td bgcolor="#80ff80">Row2 - Col2</td>
<td>Row2 - Col3</td>
</tr>
<tr>
<td>Row3 - Col1</td>
<td bgcolor="#80ff80">Row3 - Col2</td>
<td>Row3 - Col3</td>
</tr>
</table>
```

Output

Row1 - Col1	Row1 - Col2	Row1 - Col3
Row2 - Col1	Row2 - Col2	Row2 - Col3
Row3 - Col1	Row3 - Col2	Row3 - Col3

Nested Table

- **Nested tables** is placing one table over another table.
- The nested tables or ‘tables within table’ is a concept used while creating bigger and complex tables.



Nested Table: Eg:1

```
<body>
<table border="5px" bordercolor="#8707B0">
<tr>
    <td>Left side of the main table</td>

    <td>
        <h4 align="center">Nested Table</h4>
        <table border="5px" bordercolor="#F35557">
            <tr>
                <td>nested table C1</td>
                <td>nested table C2</td>
            </tr>
            <tr>
                <td>nested table</td>
                <td>nested table</td>
            </tr>
        </table>
    </td>
</tr>
</table>
```

Left side of the main table	Nested Table	
	nested table C1	nested table C2
	nested table	nested table

Nested Table: Eg: 2

```
<body>
<caption title="Container Table"> Container Table </caption>
<table border="5px" bordercolor = "red">
    <tr>
        <td >
            <table>
                <tr>
                    <th colspan="2"> Nested Table 2 </th>
                </tr>
                <tr>
                    <th> Column 1 </th>
                    <th> Column 2 </th>
                </tr>
                <tr>
                    <td> Our First Table </td>
                    <td> Nested Within </td>
                </tr>
            </table>
        </td>
```

Container Table	
Nested Table 2 Column 1 Column 2 Our First Table Nested Within	Nested Table 2 • List Object 1 • List Object 2 • List Object 3
Nested Table 3 <u>Nested Table</u> Demo Continued	Nested Table 4 HTML

Nested Table: Eg: 2 (contd..)

```
<td>  
  <table>  
    <tr>  
      <th> Nested Table 2 </th>  
    </tr>  
    <tr>  
      <td>  
        <ul>  
          <li> List Object 1 </li>  
          <li> List Object 2 </li>  
          <li> List Object 3 </li>  
        </ul>  
      </td>  
    </tr>  
  </table>  
</td>  
</tr>
```

Container Table	
Nested Table 2 Column 1 Column 2 Our First Table Nested Within	Nested Table 2 <ul style="list-style-type: none">• List Object 1• List Object 2• List Object 3
Nested Table 3 <u>Nested Table Demo Continued</u>	Nested Table 4 

Ruchita Tailor- HTML

86

Nested Table: Eg: 2 (contd..)

```
<tr>
```

```
    <td>
```

```
        <table>
```

```
            <tr>
```

```
                <th colspan="2" align="center"> Nested Table 3 </th>
```

```
            </tr>
```

```
            <tr>
```

```
                <td> <a href=""> Nested Table </a> </td>
```

```
                <td> Demo Continued </td>
```

```
            </tr>
```

```
        </table>
```

```
    </td>
```

```
<td>
```

```
    <table>
```

```
        <tr>
```

```
            <th> Nested Table 4 </th>
```

```
        </tr>
```

```
        <tr>
```

```
            <td> 
```

```
        </tr>
```

```
    </table>
```

```
    </td>
```

```
</tr>
```

```
</table>
```

```
</body>
```

Container Table

Nested Table 2		Nested Table 2
Column 1	Column 2	• List Object 1 • List Object 2 • List Object 3
Our First Table Nested Within		Nested Table 4
Nested Table 3 Nested Table Demo Continued		

Table - Summary

- **Whole in table**
 - `<table><tr><td>` creates a table
 - `<table border="">` table border
 - `<table cellspacing="">` spacing between cells
 - `<table cellpadding="">` spacing within cells
 - `<table width="">` table width
 - `<caption>` table caption
 - `<caption align="">` alignment of the caption
- **Cell**
 - `<th>` header cell
 - `<tr align="" valign=""><td align="" valign="">` alignment in cell
 - `<td width="" height="">` width and height of a cell
 - `<td colspan="" rowspan="">` spanning cells
- **Background color, Border color**
 - `<table bgcolor=""><tr bgcolor=""><td bgcolor="">` background color
 - `<table background=""><tr background=""><td background="">` background image
 - `<table border="" bordercolor="">` border color
- **Alignment, Space**
 - `<table align="">` table alignment
 - `<table vspace="" hspace="">` the space of the table
 - `<br clear="">` stopping text wrapping

Forms

- The form is used to send the data inputted by the user.
- The action attribute of the FORM element defines where to send the form data, and the method attribute specifies the HTTP method for sending the form data.

Attribute	Value	Explanation
method=""	POST	sent with the POST method
	GET	sent with the GET method (default)
action=""	URL	the form data is sent to this URL

Input Type: Text Box

- <input type="text" name="name1">
- Default input type
- Accepts characters and numbers

Attribute	Value	Explanation
type=" "	text	the type of input field (the default is "text")
name=" "	field name	a unique name for the field
size=" "	number of characters	the input field width
maxlength=" "	number of characters	the maximum number
value=" "	initial value	the text displayed in the field

Input Type: Text Box: Eg

```
<form method="POST" action="example.cgi">
```

```
<p>Default<br>
```

```
<input type="text" name="example1"></p>
```

```
<p>size="30"<br>
```

```
<input type="text" name="example2" size="30"></p>
```

```
<p>maxlength="10"<br>
```

```
<input type="text" name="example3" size="30" maxlength="10"></p>
```

```
<p>value="Hello!"<br>
```

```
<input type="text" name="example4" size="30" value="Hello!"></p>
```

```
</form>
```

Default	<input type="text"/>
size="30"	<input type="text"/>
maxlength="10"	<input type="text"/>
value="Hello!"	<input type="text"/>
Hello!	<input type="text"/>

Input Type: Password

- The input text is displayed as asterisks or bullets.
- <input type="password" name="pass1">

Attribute	Value	Explanation
type="password"	password	the type of input field
name="pass1"	field name	a unique name for the field
size="10"	number of characters	the input field width
maxlength="10"	number of characters	the maximum number
value="abcde"	initial value	the text displayed in the field

Input Type: Label

- The for attribute of <label> must be equal to the id attribute of the related element to bind them together.
- A label can also be bound to an element by placing the element inside the <label> element.
- <label for="item1">Name : </label>

```
<form action="/action_page.php">
  <label for="male">Male</label>
  <input type="radio" name="gender" id="male" value="male"><br>
  <label for="female">Female</label>
  <input type="radio" name="gender" id="female" value="female">
<br>
  <label for="other">Other</label>
  <input type="radio" name="gender" id="other" value="other"><br>
<br>
  <input type="submit" value="Submit">
</form>
```

Male
Female
Other

Submit

Input Type: Radio Button

- Select only one item
- <input type="radio" name="example" value="yes">Yes
- <input type="radio" name="example" value="no" checked>No

Attribute	Value	Explanation
type=""	radio	the type of input field
name=""	field name	a unique name for the field
value=""	initial value	this value is submitted
checked	checked	that button is checked (HTML : checked XHTML : checked="checked")

Input Type: Radio Button: Eg

```
<form method="POST" action="example.cgi">

<p>
Question 1 : Do you like baseball?
<input type="radio" name="q1" value="yes">Yes
<input type="radio" name="q1" value="no">No
</p>

<p>
Question 2 : Do you like football?
<input type="radio" name="q2" value="yes">Yes
<input type="radio" name="q2" value="no">No
</p>

</form>
```

Output

Question 1 : Do you like baseball? Yes No

Question 2 : Do you like football? Yes No

Example of having selected the "Yes"

```
<form method="POST" action="example.cgi">

<p>
Question 3 : Do you like ice hockey?
<input type="radio" name="q3" value="yes" checked>Yes
<input type="radio" name="q3" value="no">No
</p>

</form>
```

Output

Question 3 : Do you like ice hockey? Yes No

Input Type: Check box

- Select list of items
- <input type="checkbox" name="example" value="check1">Check1
- <input type="checkbox" name="example" value="check2" checked>Check2

Attribute	Value	Explanation
type=""	checkbox	the type of input field
name=""	field name	a unique name for the field
value=""	initial value	this value is submitted
checked	checked	that checkbox is checked (HTML : checked XHTML : checked="checked")

Input Type: Checkbox: Eg

```
<form method="POST" action="example.cgi">  
  
<p>  
Do you agree? :  
<input type="checkbox" name="agree" value="yes">Yes  
</p>  
  
</form>
```

Output

Do you agree? : Yes

Example of having selected the two checkboxes

```
<form method="POST" action="example.cgi">  
  
<p>  
What's your favorite animal? :  
<input type="checkbox" name="animal" value="dog">Dog  
<input type="checkbox" name="animal" value="cat" checked>Cat  
<input type="checkbox" name="animal" value="bird">Bird  
<input type="checkbox" name="animal" value="fish">Fish  
<input type="checkbox" name="animal" value="bear" checked>Bear  
</p>  
  
</form>
```

Output

What's your favorite animal? : Dog Cat Bird Fish Bear

Input Type: File

- Used to select a file on the local machine for upload to a server.
- <input type="file" name="example1"> 
- When the value of the type attribute is "file", the browse button is displayed next to the input box.
- When this button is clicked, the file selection dialog box is displayed.

Attribute	Value	Explanation
type=" "	file	the type of input field
name=" "	field name	a unique name for the field
size=" "	number of chars	the input field width

Input Type: Hidden

- Allows hidden data(not seen by the user) to be passed along with the form.
- <input type="hidden" name="example" value="hidden value">

Attribute	Value	Explanation
type=""	hidden	the type of input field
name=""	field name	a unique name for the field
value=""	initial value	this value is submitted

Input Type: Hidden: Eg

```
<form method="POST" action="example.cgi">

<p>Hidden field
<input type="hidden" name="subject" value="Contact Form"></p>

<p>Name<br>
<input type="text" name="Name" size="30"></p>

<p>Comments<br>
<textarea cols="30" rows="5" name="Comment"></textarea></p>

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

</form>
```

Output

Hidden field

Name

Comments

Submit

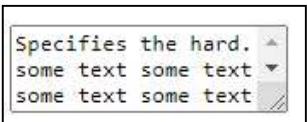
100

Input Type: Text Area

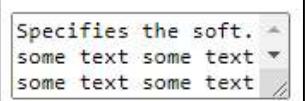
- <textarea name="example1" cols="50" rows="5"></textarea>

Attribute	Value	Explanation
name=""	field name	a unique name for the field
cols=""	number	the number of visible columns
rows=""	number	the number of visible rows
wrap="" (Extension)	hard	the input text is wrapped (and the submitted text contains line breaks)
	soft	the input text is wrapped (but the submitted text doesn't contain line breaks)
	off	the input text is not wrapped

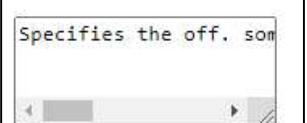
Specifies the hard.
some text some text
some text some text



Specifies the soft.
some text some text
some text some text



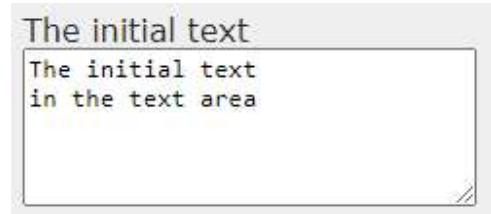
Specifies the off. som



Ruchita Tailor- HTML

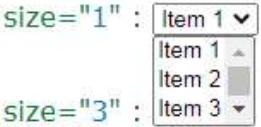
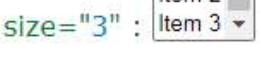
Input Type: Text Area:Eg

```
<p>The initial text<br>
    <textarea name="ta2" cols="30" rows="5">
        The initial text in the text area
    </textarea>
</p>
```



Input Type: Drop Down Menus

- ```
<select name="example">
<option value="item1">Item 1</option>
</select>
```
- **The SELECT element**

Attribute	Value	Explanation
<code>name=""</code>	field name	a unique name for the field
<code>size=" "</code>	number	the number of visible items (the default is 1)  <code>size="1" :</code>  <code>size="3" :</code> 
<code>multiple</code>	<code>multiple</code>	multiple items can be selected (HTML : <code>multiple</code>   XHTML : <code>multiple="multiple"</code> )  

# Input Type: Drop Down Menus (1)

- The **OPTION** element

Attribute	Value	Explanation
<code>value=""</code>	initial value	<p>this value is submitted</p> <ul style="list-style-type: none"><li>• This value is submitted to the server when selected. <code>&lt;option value="item1"&gt;Item 1&lt;/option&gt;</code></li><li>• If the value attribute is not used, the content of the OPTION element is submitted to the server. <code>&lt;option&gt;Item 1&lt;/option&gt;</code></li></ul>
<code>selected</code>	<code>selected</code>	<p>that item is selected (HTML : <code>selected</code>   XHTML : <code>selected="selected"</code>)</p> 

# Input Type: Drop Down Menus: Eg

```
<form method="POST" action="example.cgi">

<p>
What's your favorite color? :
<select name="color1">
<option value="white">White</option>
<option value="red">Red</option>
<option value="yellow">Yellow</option>
<option value="blue">Blue</option>
<option value="green">Green</option>
</select>
</p>

</form>
```

## Output

What's your favorite color? :

## The third row selected

```
<form method="POST" action="example.cgi">

<p>
The third row selected :
<select name="color2">
<option value="white">White</option>
<option value="red">Red</option>
<option value="yellow" selected>Yellow</option>
<option value="blue">Blue</option>
<option value="green">Green</option>
</select>
</p>

</form>
```

## Output

The third row selected :

# Input Type: Drop Down Menus: Eg(2)

Three visible rows	Select multiple items	Multiple items selected
<pre>&lt;form method="POST" action="example.cgi"&gt;  &lt;p&gt; Three visible rows : &lt;select name="color3" size="3"&gt; &lt;option value="white"&gt;White&lt;/option&gt; &lt;option value="red"&gt;Red&lt;/option&gt; &lt;option value="yellow"&gt;Yellow&lt;/option&gt; &lt;option value="blue"&gt;Blue&lt;/option&gt; &lt;option value="green"&gt;Green&lt;/option&gt; &lt;/select&gt; &lt;/p&gt; &lt;/form&gt;</pre> <p>Output</p>  <p>Three visible rows :</p>	<pre>&lt;form method="POST" action="example.cgi"&gt;  &lt;p&gt; Select multiple items : &lt;select name="color4" size="5" multiple&gt; &lt;option value="white"&gt;White&lt;/option&gt; &lt;option value="red"&gt;Red&lt;/option&gt; &lt;option value="yellow"&gt;Yellow&lt;/option&gt; &lt;option value="blue"&gt;Blue&lt;/option&gt; &lt;option value="green"&gt;Green&lt;/option&gt; &lt;/select&gt; &lt;/p&gt; &lt;/form&gt;</pre> <p>Output</p>  <p>Select multiple items :</p>	<pre>&lt;form method="POST" action="example.cgi"&gt;  &lt;p&gt; Multiple items selected : &lt;select name="color5" size="5" multiple&gt; &lt;option value="white"&gt;White&lt;/option&gt; &lt;option value="red" selected&gt;Red&lt;/option&gt; &lt;option value="yellow"&gt;Yellow&lt;/option&gt; &lt;option value="blue" selected&gt;Blue&lt;/option&gt; &lt;option value="green"&gt;Green&lt;/option&gt; &lt;/select&gt; &lt;/p&gt; &lt;/form&gt;</pre> <p>Output</p>  <p>Multiple items selected :</p>

# Input Type: Submit and reset

- Submit allows a form to be submitted. When pressed, the information will be passed on for processing, usually to a script mentioned in the action attribute option of the form
- Reset will reset the form to its initial state when selected
- <input type="submit" value="Submit">
- <input type="reset" value="Reset">

Attribute	Value	Explanation
type=""	submit	the type of input field submit : creates a submit button
	reset	reset : creates a reset button
name=""	button name	a unique name for the button
value=""	button text	the text displayed on the button

# Input Type: Submit and Reset: Eg

- ```
<form method="POST" action="example.cgi">

<p><input type="text" name="name" size="30"></p>

<p>
<input type="submit" value="Submit Button">
<input type="reset" value="Reset Button">
</p>

</form>
```

Output

Input Type: Image

- When you specify "image" for the type attribute of this element, an image submit button is created.
- This will also submit the form when selected.
- <input type="image" src="button.gif" alt="Submit">

Attribute	Value	Explanation
type=""	image	the type of input field
name=""	button name	a unique name for the button
src=""	URL	the URL of the image to display
alt=""	alternate text	short description
align=""	top, middle, bottom, left, right	image alignment (Deprecated)

Input Type: Image: Eg

```
<form method="GET" action="example.cgi">  
<p><input type="text" name="name" size="30"></p>  
<p><input type="image" src="image/button.gif" alt="Submit"></p>  
</form>
```

Output

Submit

Input Type: Button

- The **BUTTON** element defines a submit button, reset button, or push button.
- `<button type="button">button content</button>`

Attribute	Value	Explanation
<code>type=""</code>	<code>submit</code>	creates a submit button
	<code>reset</code>	creates a reset button
	<code>button</code>	creates a push button
<code>name=""</code>	button name	a unique name for the button
<code>value=""</code>	initial value	this value is submitted

```
<button type="button">Button Content</button>
```

Button Content

```
<button type="button"><strong>Button Content</strong></button>
```

Button Content

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



Fieldset

- The **FIELDSET** element defines a group of related form fields.
- <fieldset> - </fieldset>
- Form is broken into several sections
- Draws rectangles around the areas

Fieldset and Legend

```
<fieldset>  
  
<p>Name :<br>  
<input type="text" name="name" size="30"></p>  
<p>Telephone number :<br>  
<input type="text" name="telephone" size="30"></p>  
<p>E-mail address :<br>  
<input type="text" name="email" size="30"></p>  
  
</fieldset>
```

```
<fieldset>  
<legend>Your Contact Information</legend>  
  
<p>Name :<br>  
<input type="text" name="name" size="30"></p>  
<p>Telephone number :<br>  
<input type="text" name="telephone" size="30"></p>  
<p>E-mail address :<br>  
<input type="text" name="email" size="30"></p>  
  
</fieldset>
```

Name :

Telephone number :

E-mail address :

Your Contact Information

Name :

Telephone number :

E-mail address :

Forms

- **Whole in form**
 - `<form>` creates a form
 - `<form method="" action="">` method and action URL
 - `<form method="" action="" target="">` the target window of the result
 - `<form method="POST" action="" enctype="">` the MIME type
 - `<label>` label for a form field
 - `<fieldset>` group of related form fields
 - `<legend>` fieldset legend
- **Form item**
 - `<input type="text">` text box
 - `<input type="password">` password text box
 - `<input type="radio">` radio button
 - `<input type="checkbox">` checkbox
 - `<input type="file">` file selection box
 - `<input type="hidden">` hidden input field
 - `<textarea>` multi-line text area
 - `<select><option>` selectable list
- **Button**
 - `<input type="submit"><input type="reset">` submit button and reset button
 - `<input type="image">` image submit button
 - `<input type="button">` general purpose push button
 - `<button type="">` push button

Form: Example

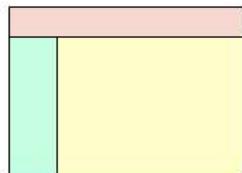
```
<html>
  <head><title>My Form</title></head>
  <form action="processor.php" method="post">
    > <fieldset><legend>Log in now</legend>
      <b>Please enter your user name</b>
      <input name="user" type="text" size=30><br>
      <b>Your password is </b>
      <input name="pass" type="password" size=10><br><br>
    </fieldset>
    <fieldset><legend>Choices!</legend>
      <b>Favourite color</b>
      <br>
      <input name="color" type="text" value="green"
             size=14><br><br>
      My favourite fruit is:
      <select>
        <option value="selected1">Apple</option>
        <option value="selected2">Banana</option>
        <option value="selected3">Orange</option>
      </select>
      <br><br>
      I live in a:
      <input type="radio" name="accom" value="house"
             checked="checked"/>House
      <input type="radio" name="accom" value="flat"/>Flat
      <input type="radio" name="accom" value="bedsit"/>Bedsit
      <input type="radio" name="accom"
             value="caravan"/>Caravan
      <br><br>
      My upload file is:
      <input type="file" name="upload" size=40><br><br>
    </fieldset>
    <fieldset><legend>Leaving?</legend>
      <b>You can leave a message for me here:</b>
      <textarea name="comments" rows=10 cols=70></textarea>
      <br><br>
      <input type="submit" value="Go!">
      <input type="reset" value="Start again">
    </fieldset>
  </form>
</html>
```

The screenshot shows a web page with a form. At the top, there's a legend "Log in now". Below it is a text input field for "Please enter your user name" and a password input field for "Your password is". Under the "Choices!" legend, there's a text input field for "Favourite colour" containing "green", a dropdown menu showing "Apple" as selected, and a list of radio buttons for "House", "Flat", "Bedsit", and "Caravan", with "House" being checked. A text input field for "My upload file is" is empty. At the bottom, there's a legend "Leaving?", a text area for "You can leave a message for me here:", and two buttons: "GO!" and "START AGAIN".

iframe

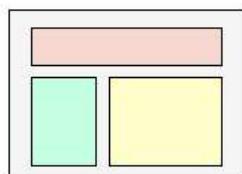
- The HTML <iframe> tag specifies an inline frame.
- The <iframe> tag is not somehow related to <frameset> tag, instead, it can appear anywhere in your document.
- The <iframe> tag defines a rectangular region within the document in which the browser can display a separate document, including scrollbars and borders.
- An inline frame is used to embed another document within the current HTML document.
- The iframe creates an inline frame within the document.
`<iframe src="example.html">- </iframe>`
- The content of the **IFRAME** element is displayed by the browsers that do not support frames.

The difference between FRAMESET and IFRAME



FRAMESET

Divides a window into two or more windows.



IFRAME

Creates frames within the body of the document.

Iframe Attributes

Attribute	Description
src	This attribute is used to give the file name that should be loaded in the frame. Its value can be any URL. For example, src = "/html/top_frame.htm" will load an HTML file available in html directory.
name	This attribute allows you to give a name to a frame. It is used to indicate which frame a document should be loaded into. This is especially important when you want to create links in one frame that load pages into another frame, in which case the second frame needs a name to identify itself as the target of the link.
frameborder	This attribute specifies whether or not the borders of that frame are shown; it overrides the value given in the frameborder attribute on the <frameset> tag if one is given, and this can take values either 1 (yes) or 0 (no).
marginwidth	This attribute allows you to specify the width of the space between the left and right of the frame's borders and the frame's content. The value is given in pixels. For example marginwidth = "10".
marginheight	This attribute allows you to specify the height of the space between the top and bottom of the frame's borders and its contents. The value is given in pixels. For example marginheight = "10".
height	This attribute specifies the height of <iframe>.
scrolling	This attribute controls the appearance of the scrollbars that appear on the frame. This takes values either "yes", "no" or "auto". For example scrolling = "no" means it should not have scroll bars.
width	This attribute specifies the width of <iframe>.

Uses of iframe

- embedding third-party media
- embedding your own media in a document-agnostic way
- embedding code examples (we do it on this site)
- embedding third party “applets” like payment forms

Basically — if you need to embed some independant, already-existing HTML document into the current document, use an `<iframe>`

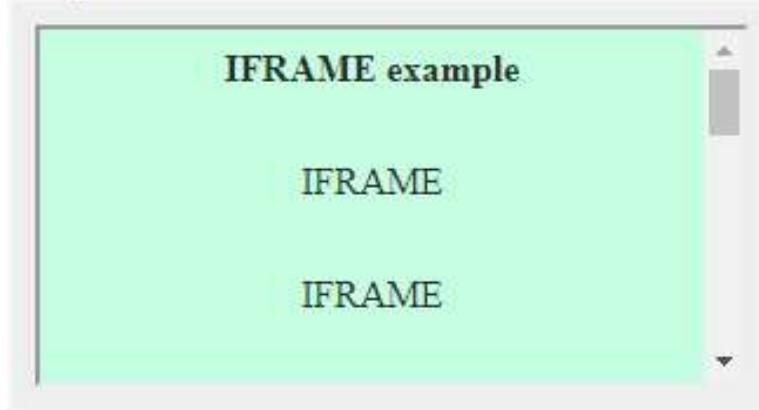
Iframe: Eg

- <iframe src="example.html">
Alternate content
</iframe>

- Eg:

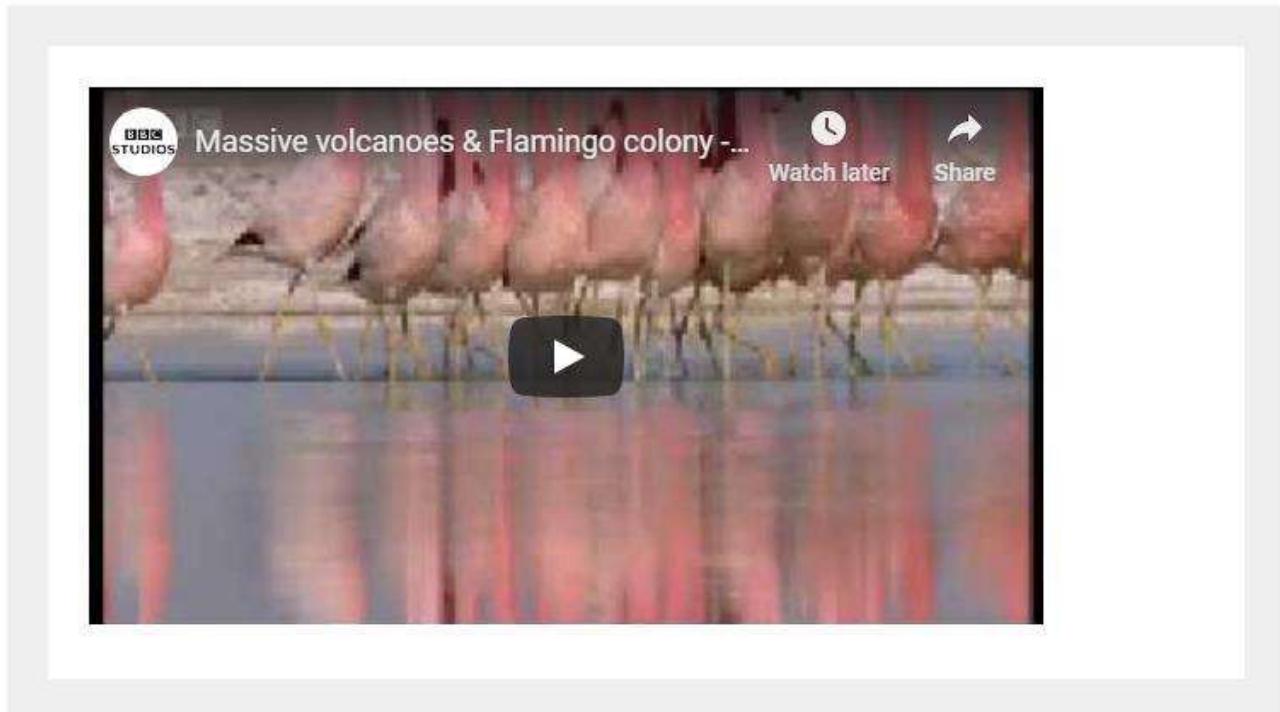
```
<iframe src="iexample_b.html">
<a href="iexample_b.html">Example page</a>
</iframe>
```

Output



Iframe : eg

- ```
<iframe width="560" height="315" src="https://www.youtube.com/embed/owsfdh4gxyc" frameborder="0" allowfullscreen></iframe>
```



INACHTA TANORI - TITIVIL

150

# Iframe: eg

```
<!DOCTYPE html>
<html>

<head>
<title>HTML Iframes</title>
</head>

<body>
<p>Document content goes here...</p>

<iframe src = "C:/Users/Home/iframe_eg.html" width = "555" height = "200">
 Sorry your browser does not support inline frames.
</iframe>

<p>Document content also go here...</p>
</body>

</html>
```

