HTML – OVERVIEW

HTML stands for **H**yper**t**ext **M**arkup **L**anguage, and it is the most widely used language to write Web Pages.

- **Hypertext** refers to the way in which Web pages (HTML documents) are linked together. Thus, the link available on a webpage is called Hypertext.

- As its name suggests, 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 structureit to display.

Originally, HTML was developed with the intent of defining the structure of documents like headings, paragraphs, lists, and so forth to facilitate the sharing of scientific information between researchers.

Now, HTML is being widely used to format web pages with the help of different tags availablein HTML language.

Basic HTML Document

In its simplest form, following is an example of an HTML document:

```
<!DOCTYPE html>

<html>

<head>

<title>This is document title</title>

</head>

<body>

<h1>This is a heading</h1>

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

</body>

</html>
```

HTML Tags

As told earlier, HTML is a markup language and makes use of various tags to format the content. These tags are enclosed within angle braces **<Tag Name>**.

Except few tags, most of the tags have their corresponding closing tags. For example, **<html>** has its closing tag**</html>** and **<body>** tag has its closing tag **</body>** tag etc.

Above example of HTML document uses the following tags:

Tag	Description
<!DOCTYPE...>	This tag defines the document type and HTML version.
<html>	This tag encloses the complete HTML document and mainly comprises of document header which is represented by <head>...</head> and document body which is represented by <body>...</body> tags.
<head>	This tag represents the document's header which can keep other HTML tags like <title>, <link> etc.
<title>	The <title> tag is used inside the <head> tag to mention the document title.
<body>	This tag represents the document's body which keeps other HTML tags like <h1>, <div>, <p> etc.
<h1>	This tag represents the heading.

<p>	This tag represents a paragraph.

HTML Document Structure

A typical HTML document will have the following structure:

```
Document declaration tag
<html>
    <head>
        Document header related tags
    </head>

    <body>
        Document body related tags
    </body>
</html>
```

The <!DOCTYPE> Declaration

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

```
<!DOCTYPE html>
```

There are many other declaration types which can be used in HTML document depending on what version of HTML is being used. We will see more details on this while discussing <!DOCTYPE...> tag along with other HTML tag

HTML – BASIC TAGS

Heading Tags

Any document starts with a heading. You can use different sizes for your headings. HTML also has six levels of headings, which use the elements **<h1>, <h2>, <h3>, <h4>, <h5>, and <h6>**. While displaying any heading, browser adds one line before and one line after that heading.

Example

```
<!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

Paragraph Tag

The **<p>** tag offers a way to structure your text into different paragraphs. Each paragraph of text should go in between an opening <p> and a closing </p> tag as shown below in the example:

Example

```
<!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>
```

Line Break Tag

Whenever you use the **
** element, anything following it starts from the next line. This tag is an example of an **empty** element, where you do not need opening and closing tags, as there is nothing to go in between them.

The <br /> tag has a space between the characters **br** and the forward slash. If you omit this space, older browsers will have trouble rendering the line break, while if you miss the forward slash character and just use <br> it is not valid in XHTML.

HTML NOTES

Example

```
<!DOCTYPE html>

<html>

<head>

<title>Line Break  Example</title>

</head>

<body>

<p>Hello<br />

You delivered your assignment on time.<br />

Thanks<br />

Mahnaz</p>

</body>

</html>
```

Centering Content

You can use **<center>** tag to put any content in the center of the page or any table cell.

Example

```
<!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>
```

.

Horizontal Lines

Horizontal lines are used to visually break-up sections of a document. The **<hr>** tag creates a line from the current position in the document to the right margin and breaks the line accordingly.

For example, you may want to give a line between two paragraphs as in the given example below:

Example

```
<!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>
```

Again **<hr />** tag is an example of the **empty** element, where you do not need opening and closing tags, as there is nothing to go in between them.

The **<hr />** element has a space between the characters **hr** and the forward slash.

HTML – ELEMENTS

An **HTML element** is defined by a starting tag. If the element contains other content, it ends with a closing tag, where the element name is preceded by a forward slash as shown below with few tags:

Start Tag	Content	End Tag
<p>	This is paragraph content.	</p>
<h1>	This is heading content.	</h1>
<div>	This is division content.	</div>

So here **<p>....</p>** is an HTML element, **<h1>...</h1>** is another HTML element. There are some HTML elements which don't need to be closed, such as **<img.../>**, **<hr />** and **
** elements. These are known as **void elements**.

HTML documents consists of a tree of these elements and they specify how HTML documents should be built, and what kind of content should be placed in what part of an HTML document.

HTML Tag vs. Element

An HTML element is defined by a *starting tag*. If the element contains other content, it ends with a *closing tag*.

For example, **<p>** is starting tag of a paragraph and **</p>** is closing tag of the same paragraph but **<p>This is paragraph</p>** is a paragraph element.

Nested HTML Elements

It is very much allowed to keep one HTML element inside another HTML element:

Example

```
<html>

<head>

<title>Nested Elements Example</title>

</head>

<body>

<h1>This is <i>italic</i> heading</h1>

<p>This is <u>underlined</u> paragraph</p>
```

We have seen few HTML tags and their usage like heading tags **<h1>, <h2>,** paragraph tag **<p>** and other tags. We used them so far in their simplest form, but most of the HTML tags can also have attributes, which are extra bits of information.

An attribute is used to define the characteristics of an HTML element and is placed inside the element's opening tag. All attributes are made up of two parts: a **name** and a **value**:

- The **name** is the property you want to set. For example, the paragraph **<p>** element in the example carries an attribute whose name is **align**, which you can use to indicate the alignment of paragraph on the page.
-
- The **value** is what you want the value of the property to be set and always put within quotations. The below example shows three possible values of align attribute: **left, center** and **right**.

Attribute names and attribute values are case-insensitive.

Example

```
<!DOCTYPE html>

<html>

<head>

<title>Align Attribute  Example</title>

</head>

<body>

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

<p align="center">This is center aligned</p>

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

</body>

</html>
```

This will display the following result:

```
This is left aligned

                              This is center aligned
```

Core Attributes

The four core attributes that can be used on the majority of HTML elements (although not all) are:

- Id
- Title
- Class
- Style

The Id Attribute

The **id** attribute of an HTML tag can be used to uniquely identify any element within an HTML page. There are two primary reasons that you might want to use an id attribute on an element:

- If an element carries an id attribute as a unique identifier, it is possible to identify just that element and its content.

- If you have two elements of the same name within a Web page (or style sheet), you can use the id attribute to distinguish between elements that have the same name.

Example

```
<p id="html">This para explains what is HTML</p>

<p id="css">This para explains what is Cascading Style Sheet</p>
```

The title Attribute

The **title** attribute gives a suggested title for the element. They syntax for the **title** attribute is similar as explained for **id** attribute:

The behavior of this attribute will depend upon the element that carries it, although it is often displayed as a tooltip when cursor comes over the element or while the element is loading.

Example

```
<!DOCTYPE html>

<html>

<head>
```

```
<title>The title Attribute Example</title>

</head>

<body>

<h3 title="Hello HTML!">Titled Heading Tag Example</h3>

</body>

</html>
```

This will produce the following result:

```
Titled Heading Tag Example
```

Now try to bring your cursor over "Titled Heading Tag Example" and you will see that whatever title you used in your code is coming out as a tooltip of the cursor.

The class Attribute

The **class** attribute is used to associate an element with a style sheet, and specifies the class of element. You will learn more about the use of the class attribute when you will learn Cascading Style Sheet (CSS). So for now you can avoid it.

The value of the attribute may also be a space-separated list of class names. For example:

```
class="className1 className2 className3"
```

The style Attribute

The style attribute allows you to specify Cascading Style Sheet (CSS) rules within the element.

```
<!DOCTYPE html>

<html>

<head>

<title>The style Attribute</title>

</head>

<body>

<p style="font-family:arial; color:#FF0000;">Some text...</p>

</body>

</html>
```

Generic Attributes

Here's a table of some other attributes that are readily usable with many of the HTML tags.

Attribute	Options	Function
align	right, left, center	Horizontally aligns tags
valign	top, middle, bottom	Vertically aligns tags within an HTML element.
bgcolor	numeric, hexidecimal, RGB values	Places a background color behind an element
background	URL	Places a background image behind an element
width	Numeric Value	Specifies the width of tables, images, or table cells.
height	Numeric Value	Specifies the height of tables, images, or table cells.
title	User Defined	"Pop-up" title of the elements.

HTML – FORMATTING

If you use a word processor, you must be familiar with the ability to make text bold, italicized, or underlined; these are just three of the ten options available to indicate how text can appear in HTML and XHTML.

Bold Text

Anything that appears within **...** element, is displayed in bold as shown below:

Example

```
<!DOCTYPE html>

<html>

<head>

<title>Bold Text Example</title>

</head>

<body>

<p>The following word uses a <b>bold</b> typeface.</p>

</body>

</html>
```

Italic Text

Anything that appears within **<i>...</i>** element is displayed in italicized as shown below:

Example

```
<!DOCTYPE html>

<html>

<head>

<title>Italic Text Example</title>

</head>
```

```
<body>

<p>The following word uses a <i>italicized</i> typeface.</p>

</body>

</html>
```

Underlined Text

Anything that appears within **<u>...</u>** element, is displayed with underline as shown below:

Example

```
<!DOCTYPE html>

<html>

<head>

<title>Underlined Text Example</title>

</head>

<body>

<p>The following word uses a <u>underlined</u> typeface.</p>

</body>

</html>
```

Strike Text

Anything that appears within **<strike>...</strike>** element is displayed with strikethrough, which is a thin line through the text as shown below:

Example

```
<!DOCTYPE html>

<html>

<head>
```

```
<title>Strike Text Example</title>

</head>

<body>

<p>The following word uses a <strike>strikethrough</strike> typeface.</p>

</body>

</html>
```

Superscript Text

The content of a **^{...}** element is written in superscript; the font size used is the same size as the characters surrounding it but is displayed half a character's height above the other characters.

Example

```
<!DOCTYPE html>

<html>

<head>

<title>Superscript Text Example</title>

</head>

<body>

<p>The following word uses a <sup>superscript</sup> typeface.</p>

</body>

</html>
```

Subscript Text

The content of a **_{...}** element is written in subscript; the font size used is the same as the characters surrounding it, but is displayed half a character's height beneath the other characters.

Example

```
<!DOCTYPE html>

<html>

<head>

<title>Subscript Text Example</title>

</head>

<body>

<p>The following word uses a <sub>subscript</sub> typeface.</p>

</body>

</html>
```

Inserted Text

Anything that appears within **<ins>...</ins>** element is displayed as inserted text.

Example

```
<!DOCTYPE html>

<html>

<head>

<title>Inserted Text Example</title>

</head>

<body>

<p>I want to drink <del>cola</del> <ins>wine</ins></p>

</body>

</html>
```

Deleted Text

Anything that appears within **~~...~~** element, is displayed as deleted text.

Example

```
<!DOCTYPE html>
<html>
<head>
<title>Deleted Text Example</title>
</head>
<body>
<p>I want to drink <del>cola</del> <ins>wine</ins></p>
</body>
</html>
```

Larger Text

The content of the **<big>...</big>** element is displayed one font size larger than the rest of the text surrounding it as shown below:

Example

```
<!DOCTYPE html>
<html>
<head>
<title>Larger Text Example</title>
</head>
<body>
<p>The following word uses a <big>big</big> typeface.</p>
</body>
</html>
```

Smaller Text

The content of the **<small>...</small>** element is displayed one font size smaller than the rest of the text surrounding it as shown below:

Example

```
<!DOCTYPE html>

<html>

<head>

<title>Smaller Text Example</title>

</head>

<body>

<p>The following word uses a <small>small</small> typeface.</p>

</body>
```

```
</html>
```

Grouping Content

The **<div>** and **<span>** elements allow you to group together several elements to create sections or subsections of a page.

For example, you might want to put all of the footnotes on a page within a <div> element to indicate that all of the elements within that <div> element relate to the footnotes. You might then attach a style to this <div> element so that they appear using a special set of style rules.

Example

```
<!DOCTYPE html>
<html>
<head>
<title>Div Tag Example</title>
</head>
<body>
<div id="menu" align="middle" >
<a href="/index.htm">HOME</a> |
<a href="/about/contact_us.htm">CONTACT</a> |
<a href="/about/index.htm">ABOUT</a>
</div>

<div id="content" align="left" bgcolor="white">
<h5>Content Articles</h5>
<p>Actual content goes here ....</p>
</div>
</body>
</html>
```

Image Tag

Images are very important to beautify as well as to depict many complex concepts in simple way on your web page.

Insert Image

You can insert any image in your web page by using **<img>** tag. Following is the simple syntax to use this tag.

The <img> tag is an empty tag, which means that, it can contain only list of attributes and it has no closing tag.

Example

To try following example, let's keep our HTML file test.htm and image file test.png in the same directory −

```
<html>

  <head>
    <title>Using Image in Webpage</title>
  </head>

  <body>
    <p>Simple Image Insert</p>
    <img src = "/html/images/test.png" alt = "Test Image" />
  </body>

</html>
```

Set Image Width/Height

You can set image width and height based on your requirement using **width** and **height** attributes. You can specify width and height of the image in terms of either pixels or percentage of its actual size.

Example

```
<!DOCTYPE html>
<html>

  <head>
    <title>Set Image Width and Height</title>
  </head>
```

```
  <body>
    <p>Setting image width and height</p>
    <img src = "/html/images/test.png" alt = "Test Image" width = "150" height = "100"/>
  </body>

</html>
```

Set Image Border

By default, image will have a border around it, you can specify border thickness in terms of pixels using border attribute. A thickness of 0 means, no border around the picture.

Example

```
<!DOCTYPE html>
<html>

  <head>
    <title>Set Image Border</title>
  </head>

  <body>
    <p>Setting image Border</p>
    <img src = "/html/images/test.png" alt = "Test Image" border = "3"/>
  </body>

</html>
```

Set Image Alignment

By default, image will align at the left side of the page, but you can use **align** attribute to set it in the center or right.

Example

```
<!DOCTYPE html>
<html>

  <head>
    <title>Set Image Alignment</title>
  </head>

  <body>
    <p>Setting image Alignment</p>
```

```
   <img src = "/html/images/test.png" alt = "Test Image" border = "3"
align = "right"/>
   </body>

</html>
```

TABLE TAG

The HTML tables allow web authors to arrange data like text, images, links, other tables, etc. into rows and columns of cells.

The HTML tables are created using the **<table>** tag in which the **<tr>** tag is used to create table rows and **<td>** tag is used to create data cells. The elements under <td> are regular and left aligned by default

Example

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Tables</title>
   </head>

   <body>
      <table border = "1">
         <tr>
            <td>Row 1, Column 1</td>
            <td>Row 1, Column 2</td>
         </tr>

         <tr>
            <td>Row 2, Column 1</td>
            <td>Row 2, Column 2</td>
         </tr>
      </table>

   </body>
</html>
```

Here, the **border** is an attribute of <table> tag and it is used to put a border across all the cells. If you do not need a border, then you can use border = "0".

Table Heading

Table heading can be defined using **<th>** tag. This tag will be put to replace <td> tag, which is used to represent actual data cell. Normally you will put your top row as table

heading as shown below, otherwise you can use <th> element in any row. Headings, which are defined in <th> tag are centered and bold by default.

Example

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Table Header</title>
   </head>

   <body>
      <table border = "1">
         <tr>
            <th>Name</th>
            <th>Salary</th>
         </tr>
         <tr>
            <td>Mohit</td>
            <td>5000</td>
         </tr>

         <tr>
            <td>Richa Arora</td>
            <td>7000</td>
         </tr>
      </table>
   </body>

</html>
```

Cellpadding and Cellspacing Attributes

There are two attributes called *cellpadding* and *cellspacing* which you will use to adjust the white space in your table cells. The cellspacing attribute defines space between table cells, while cellpadding represents the distance between cell borders and the content within a cell.

Example

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Table Cellpadding</title>
   </head>
```

```
    <body>
        <table border = "1" cellpadding = "5" cellspacing = "5">
            <tr>
                <th>Name</th>
                <th>Salary</th>
            </tr>
            <tr>
                <td>Mohit</td>
                <td>5000</td>
            </tr>
            <tr>
                <td>Richa Arora</td>
                <td>7000</td>
            </tr>
        </table>
    </body>

</html>
```

Colspan and Rowspan Attributes

You will use **colspan** attribute if you want to merge two or more columns into a single column. Similar way you will use **rowspan** if you want to merge two or more rows.

Example

```
<!DOCTYPE html>
<html>

    <head>
        <title>HTML Table Colspan/Rowspan</title>
    </head>

    <body>
        <table border = "1">
            <tr>
                <th>Column 1</th>
                <th>Column 2</th>
                <th>Column 3</th>
            </tr>
            <tr>
                <td rowspan = "2">Row 1 Cell 1</td>
                <td>Row 1 Cell 2</td>
                <td>Row 1 Cell 3</td>
            </tr>
            <tr>
                <td>Row 2 Cell 2</td>
```

```
            <td>Row 2 Cell 3</td>
        </tr>
        <tr>
            <td colspan = "3">Row 3 Cell 1</td>
        </tr>
    </table>
  </body>

</html>
```

Tables Backgrounds

You can set table background using one of the following two ways −

- **bgcolor** attribute − You can set background color for whole table or just for one cell.

- **background** attribute − You can set background image for whole table or just for one cell.

You can also set border color also using **bordercolor** attribute.

Note − The *bgcolor*, *background*, and *bordercolor* attributes deprecated in HTML5. Do not use these attributes.

Example

```
<!DOCTYPE html>
<html>

  <head>
      <title>HTML Table Background</title>
  </head>

  <body>
      <table border = "1" bordercolor = "green" bgcolor = "yellow">
        <tr>
            <th>Column 1</th>
            <th>Column 2</th>
            <th>Column 3</th>
        </tr>
        <tr>
            <td rowspan = "2">Row 1 Cell 1</td>
            <td>Row 1 Cell 2</td>
            <td>Row 1 Cell 3</td>
        </tr>
        <tr>
            <td>Row 2 Cell 2</td>
            <td>Row 2 Cell 3</td>
        </tr>
        <tr>
```

```
            <td colspan = "3">Row 3 Cell 1</td>
        </tr>
    </table>
  </body>

</html>
```

Here is an example of using **background** attribute. Here we will use an image available in /images directory.

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Table Background</title>
   </head>

   <body>
      <table border = "1" bordercolor = "green" background =
"/images/test.png">
         <tr>
            <th>Column 1</th>
            <th>Column 2</th>
            <th>Column 3</th>
         </tr>
         <tr>
            <td rowspan = "2">Row 1 Cell 1</td>
            <td>Row 1 Cell 2</td><td>Row 1 Cell 3</td>
         </tr>
         <tr>
            <td>Row 2 Cell 2</td>
            <td>Row 2 Cell 3</td>
         </tr>
         <tr>
            <td colspan = "3">Row 3 Cell 1</td>
         </tr>
      </table>
   </body>

</html>
```

This will produce the following result. Here background image did not apply to table's header.

Table Height and Width

You can set a table width and height using **width** and **height** attributes. You can specify table width or height in terms of pixels or in terms of percentage of available screen area.

Example

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Table Width/Height</title>
   </head>

   <body>
      <table border = "1" width = "400" height = "150">
         <tr>
            <td>Row 1, Column 1</td>
            <td>Row 1, Column 2</td>
         </tr>

         <tr>
            <td>Row 2, Column 1</td>
            <td>Row 2, Column 2</td>
         </tr>
      </table>
   </body>

</html>
```

Table Caption

The **caption** tag will serve as a title or explanation for the table and it shows up at the top of the table. This tag is deprecated in newer version of HTML/XHTML.

Example

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Table Caption</title>
   </head>

   <body>
      <table border = "1" width = "100%">
         <caption>This is the caption</caption>

         <tr>
            <td>row 1, column 1</td><td>row 1, columnn 2</td>
```

28

```
            </tr>

            <tr>
                <td>row 2, column 1</td><td>row 2, columnn 2</td>
            </tr>
        </table>
    </body>

</html>
```

Table Header, Body, and Footer

Tables can be divided into three portions − a header, a body, and a foot. The head and foot are rather similar to headers and footers in a word-processed document that remain the same for every page, while the body is the main content holder of the table.

The three elements for separating the head, body, and foot of a table are −

- **<thead>** − to create a separate table header.
- **<tbody>** − to indicate the main body of the table.
- **<tfoot>** − to create a separate table footer.

A table may contain several <tbody> elements to indicate *different pages* or groups of data. But it is notable that <thead> and <tfoot> tags should appear before <tbody>

Example

```
<!DOCTYPE html>
<html>

    <head>
        <title>HTML Table</title>
    </head>

    <body>
        <table border = "1" width = "100%">
            <thead>
                <tr>
                    <td colspan = "4">This is the head of the table</td>
                </tr>
            </thead>

            <tfoot>
                <tr>
                    <td colspan = "4">This is the foot of the table</td>
                </tr>
            </tfoot>
```

```
            <tbody>
                <tr>
                    <td>Cell 1</td>
                    <td>Cell 2</td>
                    <td>Cell 3</td>
                    <td>Cell 4</td>
                </tr>
            </tbody>

        </table>
    </body>

</html>
```

Nested Tables

You can use one table inside another table. Not only tables you can use almost all the tags inside table data tag <td>.

Example

Following is the example of using another table and other tags inside a table cell.

```
<!DOCTYPE html>
<html>

    <head>
        <title>HTML Table</title>
    </head>

    <body>
        <table border = "1" width = "100%">

            <tr>
                <td>
                    <table border = "1" width = "100%">
                        <tr>
                            <th>Name</th>
                            <th>Salary</th>
                        </tr>
                        <tr>
                            <td>Ramesh Raman</td>
                            <td>5000</td>
                        </tr>
                        <tr>
                            <td>Richa Arora</td>
                            <td>7000</td>
                        </tr>
```

```
                    </table>
            </td>
        </tr>

    </table>
    </body>

</html>
```

LIST

HTML offers web authors three ways for specifying lists of information. All lists must contain one or more list elements. Lists may contain −

- **<ul>** − An unordered list. This will list items using plain bullets.

- **<ol>** − An ordered list. This will use different schemes of numbers to list your items.

- **<dl>** − A definition list. This arranges your items in the same way as they are arranged in a dictionary.

HTML Unordered Lists

An unordered list is a collection of related items that have no special order or sequence. This list is created by using HTML **<ul>** tag. Each item in the list is marked with a bullet.

Example

```
<!DOCTYPE html>
<html>

    <head>
        <title>HTML Unordered List</title>
    </head>

    <body>
        <ul>
            <li>Beetroot</li>
            <li>Ginger</li>
            <li>Potato</li>
            <li>Radish</li>
        </ul>
    </body>

</html>
```

The type Attribute

You can use **type** attribute for <ul> tag to specify the type of bullet you like. By default, it is a disc. Following are the possible options −

```
<ul type = "square">
<ul type = "disc">
<ul type = "circle">
```

Example

Following is an example where we used <ul type = "square">

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Unordered List</title>
   </head>

   <body>
      <ul type = "square">
         <li>Beetroot</li>
         <li>Ginger</li>
         <li>Potato</li>
         <li>Radish</li>
      </ul>
   </body>

</html>
```

Example

Following is an example where we used <ul type = "disc"> −

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Unordered List</title>
   </head>

   <body>
      <ul type = "disc">
         <li>Beetroot</li>
         <li>Ginger</li>
```

```
        <li>Potato</li>
        <li>Radish</li>
    </ul>
</body>

</html>
```

Example

Following is an example where we used <ul type = "circle"> −

```
<!DOCTYPE html>
<html>

    <head>
        <title>HTML Unordered List</title>
    </head>

    <body>
        <ul type = "circle">
            <li>Beetroot</li>
            <li>Ginger</li>
            <li>Potato</li>
            <li>Radish</li>
        </ul>
    </body>

</html>
```

HTML Ordered Lists

If you are required to put your items in a numbered list instead of bulleted, then HTML ordered list will be used. This list is created by using **<ol>** tag. The numbering starts at one and is incremented by one for each successive ordered list element tagged with <li>.

Example

```
<!DOCTYPE html>
<html>

    <head>
        <title>HTML Ordered List</title>
    </head>

    <body>
        <ol>
```

```
            <li>Beetroot</li>
            <li>Ginger</li>
            <li>Potato</li>
            <li>Radish</li>
        </ol>
    </body>

</html>
```

The type Attribute

You can use **type** attribute for <ol> tag to specify the type of numbering you like. By default, it is a number. Following are the possible options −

```
<ol type = "1"> - Default-Case Numerals.
<ol type = "I"> - Upper-Case Numerals.
<ol type = "i"> - Lower-Case Numerals.
<ol type = "A"> - Upper-Case Letters.
<ol type = "a"> - Lower-Case Letters.
```

Example

Following is an example where we used <ol type = "1">

```
<!DOCTYPE html>
<html>

    <head>
        <title>HTML Ordered List</title>
    </head>

    <body>
        <ol type = "1">
            <li>Beetroot</li>
            <li>Ginger</li>
            <li>Potato</li>
            <li>Radish</li>
        </ol>
    </body>

</html>
```

Example

Following is an example where we used <ol type = "I">

```
<!DOCTYPE html>
```

```
<html>

    <head>
        <title>HTML Ordered List</title>
    </head>

    <body>
        <ol type = "I">
            <li>Beetroot</li>
            <li>Ginger</li>
            <li>Potato</li>
            <li>Radish</li>
        </ol>
    </body>

</html>
```

Example

Following is an example where we used <ol type = "i">

```
<!DOCTYPE html>
<html>

    <head>
        <title>HTML Ordered List</title>
    </head>

    <body>
        <ol type = "i">
            <li>Beetroot</li>
            <li>Ginger</li>
            <li>Potato</li>
            <li>Radish</li>
        </ol>
    </body>

</html>
```

Example

Following is an example where we used <ol type = "A" >

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

```
    <head>
        <title>HTML Ordered List</title>
    </head>

    <body>
        <ol type = "A">
            <li>Beetroot</li>
            <li>Ginger</li>
            <li>Potato</li>
            <li>Radish</li>
        </ol>
    </body>

</html>
```

Example

Following is an example where we used <ol type = "a">

```
<!DOCTYPE html>
<html>

    <head>
        <title>HTML Ordered List</title>
    </head>

    <body>
        <ol type = "a">
            <li>Beetroot</li>
            <li>Ginger</li>
            <li>Potato</li>
            <li>Radish</li>
        </ol>
    </body>

</html>
```

The start Attribute

You can use **start** attribute for <ol> tag to specify the starting point of numbering you need.
Following are the possible options −

```
<ol type = "1" start = "4">    - Numerals starts with 4.
<ol type = "I" start = "4">    - Numerals starts with IV.
```

```
<ol type = "i" start = "4">     - Numerals starts with iv.
<ol type = "a" start = "4">     - Letters starts with d.
<ol type = "A" start = "4">     - Letters starts with D.
```

Example

Following is an example where we used <ol type = "i" start = "4" >

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Ordered List</title>
   </head>

   <body>
      <ol type = "i" start = "4">
         <li>Beetroot</li>
         <li>Ginger</li>
         <li>Potato</li>
         <li>Radish</li>
      </ol>
   </body>

</html>
```

HTML Definition Lists

HTML and XHTML supports a list style which is called **definition lists** where entries are listed like in a dictionary or encyclopedia. The definition list is the ideal way to present a glossary, list of terms, or other name/value list.

Definition List makes use of following three tags.

- <dl> − Defines the start of the list
- <dt> − A term
- <dd> − Term definition
- </dl> − Defines the end of the list

Example

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Definition List</title>
   </head>
```

```
    <body>
      <dl>
          <dt><b>HTML</b></dt>
          <dd>This stands for Hyper Text Markup Language</dd>
          <dt><b>HTTP</b></dt>
          <dd>This stands for Hyper Text Transfer Protocol</dd>
      </dl>
    </body>

</html>
```

Creating Frames

To use frames on a page we use <frameset> tag instead of <body> tag. The <frameset> tag defines, how to divide the window into frames. The **rows** attribute of <frameset> tag defines horizontal frames and **cols** attribute defines vertical frames. Each frame is indicated by <frame> tag and it defines which HTML document shall open into the frame.

Note − The <frame> tag deprecated in HTML5. Do not use this element.

Example

Following is the example to create three horizontal frames −

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Frames</title>
   </head>

   <frameset rows = "10%,80%,10%">
      <frame name = "top" src = "/html/top_frame.htm" />
      <frame name = "main" src = "/html/main_frame.htm" />
      <frame name = "bottom" src = "/html/bottom_frame.htm" />

      <noframes>
         <body>Your browser does not support frames.</body>
      </noframes>

   </frameset>

</html>
```

Example

Let's put the above example as follows, here we replaced rows attribute by cols and changed their width. This will create all the three frames vertically −

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Frames</title>
   </head>

   <frameset cols = "25%,50%,25%">
      <frame name = "left" src = "/html/top_frame.htm" />
      <frame name = "center" src = "/html/main_frame.htm" />
      <frame name = "right" src = "/html/bottom_frame.htm" />

      <noframes>
         <body>Your browser does not support frames.</body>
      </noframes>
   </frameset>

</html>
```

The <frameset> Tag Attributes

Following are important attributes of the <frameset> tag −

Sr.No	Attribute & Description
1	**cols** Specifies how many columns are contained in the frameset and the size of each column. You can specify the width of each column in one of the four ways − Absolute values in pixels. For example, to create three vertical frames, use *cols = "100, 500, 100"*. A percentage of the browser window. For example, to create three vertical frames, use *cols = "10%, 80%, 10%"*. Using a wildcard symbol. For example, to create three vertical frames, use *cols = "10%, *, 10%"*. In this case wildcard takes remainder of the window. As relative widths of the browser window. For example, to create three vertical frames, use *cols = "3*, 2*, 1*"*. This is an alternative to percentages. You can use relative widths of the browser window. Here the window is divided into sixths: the

	first column takes up half of the window, the second takes one third, and the third takes one sixth.
2	**rows** This attribute works just like the cols attribute and takes the same values, but it is used to specify the rows in the frameset. For example, to create two horizontal frames, use *rows = "10%, 90%"*. You can specify the height of each row in the same way as explained above for columns.
3	**border** This attribute specifies the width of the border of each frame in pixels. For example, border = "5". A value of zero means no border.
4	**frameborder** This attribute specifies whether a three-dimensional border should be displayed between frames. This attribute takes value either 1 (yes) or 0 (no). For example frameborder = "0" specifies no border.
5	**framespacing** This attribute specifies the amount of space between frames in a frameset. This can take any integer value. For example framespacing = "10" means there should be 10 pixels spacing between each frames.

The <frame> Tag Attributes

Following are the important attributes of <frame> tag −

Sr.No	Attribute & Description
1	**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.
2	**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 an another frame, in which case the second frame needs a name to identify itself as the target of the link.
3	**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).
4	**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".
5	**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".
6	**noresize** By default, you can resize any frame by clicking and dragging on the borders of a frame. The noresize attribute prevents a user from being able to resize the frame. For example noresize = "noresize".
7	**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.
8	**longdesc** This attribute allows you to provide a link to another page containing a long description of the contents of the frame. For example longdesc = "framedescription.htm"

Browser Support for Frames

If a user is using any old browser or any browser, which does not support frames then <noframes> element should be displayed to the user.

So you must place a <body> element inside the <noframes> element because the <frameset> element is supposed to replace the <body> element, but if a browser does not understand <frameset> element then it should understand what is inside the <body> element which is contained in a <noframes> element.

You can put some nice message for your user having old browsers. For example, *Sorry!! your browser does not support frames.* as shown in the above example.

Frame's name and target attributes

One of the most popular uses of frames is to place navigation bars in one frame and then load main pages into a separate frame.

Let's see following example where a test.htm file has following code −

```
<!DOCTYPE html>
<html>

   <head>
      <title>HTML Target Frames</title>
   </head>

   <frameset cols = "200, *">
      <frame src = "/html/menu.htm" name = "menu_page" />
      <frame src = "/html/main.htm" name = "main_page" />

      <noframes>
         <body>Your browser does not support frames.</body>
      </noframes>
   </frameset>

</html>
```

Here, we have created two columns to fill with two frames. The first frame is 200 pixels wide and will contain the navigation menu bar implemented by **menu.htm** file. The second column fills in remaining space and will contain the main part of the page and it is implemented by **main.htm** file. For all the three links available in menu bar, we have mentioned target frame as **main_page**, so whenever you click any of the links in menu bar, available link will open in main page.

Following is the content of menu.htm file

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

```
    <body bgcolor = "#4a7d49">
        <a href = "http://www.google.com" target =
"main_page">Google</a>
        <br />
        <br />

        <a href = "http://www.microsoft.com" target =
"main_page">Microsoft</a>
        <br />
        <br />

        <a href = "http://news.bbc.co.uk" target = "main_page">BBC
News</a>
    </body>

</html>
```

Following is the content of main.htm file −

```
<!DOCTYPE html>
<html>

    <body bgcolor = "#b5dcb3">
        <h3>This is main page and content from any link will be
displayed here.</h3>
        <p>So now click any link and see the result.</p>
    </body>

</html>
```

When we load **test.htm** file, it produces following result −

Now you can try to click links available in the left panel and see the result. The *targetattribute* can also take one of the following values −

Sr.No	Option & Description
1	**_self** Loads the page into the current frame.
2	**_blank** Loads a page into a new browser window. Opening a new window.

3	**_parent** Loads the page into the parent window, which in the case of a single frameset is the main browser window.	
4	**_top** Loads the page into the browser window, replacing any current frames.	
5	**targetframe** Loads the page into a named targetframe.	

You can define an inline frame with HTML tag **<iframe>**. 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 **src** attribute is used to specify the URL of the document that occupies the inline frame.

Example

Following is the example to show how to use the <iframe> −

Live Demo

```
<!DOCTYPE html>
<html>

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

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

      <iframe src = "/html/menu.htm" width = "555" height = "200">
         Sorry your browser does not support inline frames.
      </iframe>

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

</html>
```

The <Iframe> Tag Attributes

Most of the attributes of the <iframe> tag, including *name, class, frameborder, id, longdesc, marginheight, marginwidth, name, scrolling, style,* and *title* behave exactly like the corresponding attributes for the <frame> tag.

Note −The *frameborder*, *marginwidth*, *longdesc*, *scrolling*, *marginheight* attributes deprecated in HTML5. Do not use these attributes.

Sr.No	Attribute & Description
1	**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.
2	**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 an another frame, in which case the second frame needs a name to identify itself as the target of the link.
3	**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).
4	**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".
5	**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".
6	**height**

	This attribute specifies the height of <iframe>.
7	**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.
8	**longdesc** This attribute allows you to provide a link to another page containing a long description of the contents of the frame. For example longdesc = "framedescription.htm"
9	**width** This attribute specifies the width of <iframe>.