LEARN FOR CAUSE

WEB DEVELOPMENT COURSE MATERIAL

HTML (Hyper Text Mark-up Language) Study Material

What is HTML?

HTML, otherwise known as HyperText Markup Language, is the language used to create Web pages
Hypertext" refers to the hyperlinks that an **HTML** page may contain.
"**Markup language**" refers to the way tags are used to define the page layout and elements within the page
Using HTML, you can create a Web page with text, graphics, sound, and videos
HTML elements are the building blocks of HTML pages
HTML elements are represented by tags

HTML along with CSS and JavaScript are the core parts of **web technologies**. **HTML** is used to create & display content of a website. This content can be text, headings, media, list , tables etc on a web browser. It is impossible to build a website without html.

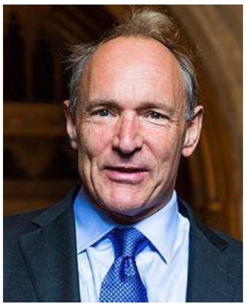
HTML is very easy to learn. HTML contains tags and attributes used to build a webpage. HTML is browser interpreted language and need no compilation.

A Markup Language is a computer language which use tags and attributes to create structure. For ex **HTML** and **XML**.

HTML Facts

1. HTML is not a programming language.
2. HTML can build Websites, emails and Hybrid Apps.
3. World's First Web site was created on 12 Mar 1989 using html only.

HTML History



Tim Berners-Lee

HTML was invented by a CERN scientist **Tim Berners-Lee** in 1989. The primary purpose to invent **HTML** was to share information on web for Research Scientists and Engineers so that other professionals in same domain can have access to your research work. Writing Books and Journals are only limited to some people. To run html, a web browser *WorldWideWeb* was developed, but later on it was renamed to *Nexus*.

W3C is the organisation who build standards for the development of **World Wide Web**. W3C was founded by Tim Berners-Lee in Oct 1994. Tim Berners-Lee is also the current CEO of W3C. To check a webpage standards, we use W3C Validator.

World Wide Web facts

1. There are 1.8 Billions websites on www.
2. Archie was the first search engine.
3. First image on web was published on 1992.
4. China is no 1 on web surfers list.
5. India is second country on most web surfers list.

HTML was first formed in 1991. Till now, **HTML** receive many updates. Here is a **list of HTML versions** with release date.

HTML VERSION	**DATE**
HTML	1991
HTML 2	1995
HTML 3.2	1997
HTML 4	1997
HTML4.01	1999
XHTML 1.0	2000
XHTML 1.1	2001
HTML5	2011

HTML Editors

HTML Editor is the software used to write html code. A web browser is used to view webpage. Some Popular code editors are **notepad++**, **brackets**, **sublime text** , **atom**, and **Visual Studio Code** etc. You can use any one of them.

How to create a website using html

To **build a webpage using html**, use following steps. These steps are based on HTML5 web standards. Just follow these simple steps, and your **first html page** is ready, with *W3C Standards*.

1. Open code editor.
2. Create the **doctype of webpage**. e.g. <!doctype html>
3. Create Parent **html tag** e.g. <html> </html>
4. Create **head tag** inside **html tag**, e.g. <head> </head>
5. Create **body tag** after head tag closing, e.g. <body> </body>
6. Add **title tag** inside head, e.g. <title> </title>
7. Add **meta tag** inside head, e.g. <meta charset="utf-8">
8. Save page as **index.html** file on your system.
9. Double Click the file and your webpage is live on browser.

To edit webpage, right click on html file and select open in editor.

Sample HTML Page

```
<!doctype html>

<html lang="en">

<head>

    <title>LFC BHUBANESWAR </title>

    <meta charset="UTF-8">

</head>

<body>

</body>

</html>
```

In body tag, add some text, for example, *Hello STUDENTS*.

```
<!DOCTYPE html>

<html lang="en">

<head>

    <title>Hello STUDENTS</title>

    <meta charset="UTF-8">

</head>

<body>

    Hello STUDENTS

</body>

</html>
```

HTML Tags

HTML Tags are used to build components in a webpage. Everything in a webpage is create by **HTML Tags**. Here are some popular **HTML Tags** and their use.

Tag Name	Description
<html>	**HTML Tag** is the **parent tag** of a webpage.
<head>	**Head Tag** is first child of html tag and is used to write the information for web browsers and Search engines.
<title>	**Title tag** is used **only once** inside head tag. Title is always displayed in browsers tab..
<meta>	**Meta tag** is used to define the **charset family**, **description**, **keywords**, **Author**, **robots** and **Geo Location** of the website.
<link>	**Link tag** is used to link **external css**, **favicon icon**, **publisher** and **canonical** of the webpage.
<script>	**Script Tag** is used to attach **external javascript**, **jquery**, and **AngularJS scripts** with the webpage.
<body>	**Body tag** is used to create the **webpage structure** which includes **Headings**, **Paragraphs**, **images**, **tables**, **division**.

HTML Attributes

HTML Attributes are used to add extra information to an HTML Element. Attributes are used in start tag. Only a single attribute is allowed once. Attribute repetition is not allowed.

Attribute	Description
class	group single or multiple elements.
	<p class="text">LFC </p>
id	Set unique id of single element.
	<p id="text">LFC</p>
title	Tooltip of an element. Shows title content on mouseover.
	<p title="welcome to LFC">para</p>
type	Specific type of html Elements.
	<input type="text">
src	Source of media elements like images, iframes, Audio, Video.
	<img src="logo.png" alt="">
href	Hypertext Reference of Hyperlinks or Link Tag.
	<a href="about.html">About</a>

An HTML page is always started with doctype first. HTML Doctype is not a tag. Doctype define the html version to browsers, search engines and w3c validator with DTD we are using.

HTML Comments

HTML Comments are used to write messages or notifications for web developers. Comments are started with <!-- and ends with -->. These comments are not visible in browsers, but remain in page source code.

```
<!-- HTML Comment -->
```

Attributes

HTML elements can have **attributes** in opening tag. **Attribute** provides **additional information** about that element. **HTML attributes** are added in opening tag and can have some value. Some popular **Global Attributes** are **class**, **id**, **title**, **style** etc.

HTML attributes are always defined in **the start tag**. Attributes are written in name/value pair like: <tag attributename="attributevalue">. Attribute once used can't be repeat in same element.

Only one attribute is allowed in a single element.

HTML Attributes Type

HTML Attributes are categorized on the basics of their functionality and the way they are written.

Attribute Type	**Meaning**
Global Attribute	Attributes meant for all html elements.
Boolean Attribute	Attribute with Attribute name only, no value. Exp :disabled, hidden, reversed etc.
Presentational Attributes	Attributes used to style HTML Element. example : style, size, color, border, cellspacing.

Class Attribute

Class attribute is used to **define a group of elements** having same css properties and definitions. **Class** can have single or multiple values with whitespace separation.

The name of the class should be descriptive. Like, for top-most div, preferred class name is *header*. For navigation, class name should be *nav*, for main content, class name should be *container* and for last div, class name should be *footer*.

Font with class green

Font without any class

```
<style>

    p{ font-family:sans-serif}

    .red{ color:green}

    p.bold{ font-weight:bold}

</style>



<p class="green bold"> I am learning web development</p>

<p class="green”> I am learning web development </p>

<p> BLANK ( NO CLASS NAME ) </p>
```

ID Attribute

Id attribute is used to call a unique element. In a single webpage, an id name is always used once. An html element can have a single ID attribute and single value of id.

ID Attribute Code

```
<style>

    .red{ color:red}

    #blue{ color:blue}

</style>
```

```
<p class="red" > class red</p>

<p class="red" class red</p>

<p id="blue”> id blue</p>
```

HTML Headings

HTML includes Six headings elements. These headings are <h1>, <h2>, <h3>, <h4>, <h5> and <h6>.

Heading elements are used to write **headings** in a webpage, whereas p tag is only for plain text. All Headings are **bold** and **block level elements.**

HTML Headings Example

1. Heading 1 (<h1>)
2. Heading 2 (<h2>)
3. Heading 3 (<h3>)
4. Heading 4 (<h4>)
5. Heading 5 (<h5>)
6. Heading 6 (<h6>)

Headings Code

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

HTML h1 tag <h1>

H1 tag or **<h1>** is the **main heading** or **level one heading** of a webpage. Search engines like *google*, *yahoo*, *bing* etc gives maximum priority to the **h1 element**.

```
<h1> Heading 1 </h1>
```

HTML h2 tag <h2>

H2 tag or **<h2>** is the **sub heading** of **h1 element**. It is recommended to use headings in proper order for better search results.

Heading 2

```
<h2> Heading 2 </h2>
```

HTML Headings Example

In the example below, we are building a webpage with title and h1 **Web Designing**. In the body, we will start with **h1 tag** first, and then other headings.

```
<!doctype html>

<html lang="en">

<head>

<meta charset="UTF-8">

<title>Web Designing</title>

</head>

<body>

    <h1>Web Designing Course</h1>

    <p>Description for web designing</p>
```

```
<h2>HTML</h2>
<p>HTML is beautiful </p>

    <h3>HTML 4</h3>
    <p>HTML 4 is wonderful </p>

    <h3>HTML 5</h3>
    <p>HTML 5 is the best </p>

<h2>CSS</h2>
<p>Cascading Style Sheets </p>

    <h3>CSS 2</h3>
    <p>CSS is beautiful </p>

    <h3>CSS 3</h3>
    <p>CSS makes the website dynamic </p>

<h2>Javascript</h2>
<p>I LOVE JS </p>

    <h3>Javascript</h3>
    <p>JavaScript is cool </p>
```

```
        <h3>Jquery</h3>

        <p>Jquery</p>

</body>

</html>
```

HTML Block and Inline Elements

HTML Elements are classified as **Block Level** and **Inline Level** elements on the basics of their **display**. Some elements display as blocks and some inline.

To check an HTML Element is block or inline, open browser inspect and check display property in user agent stylesheet. If its display: block, it is block level, else it is inline level.

Block Vs Inline level Elements

<p> <div> <h1> < h2> <address> etc are **block level elements**, whereas <b>, <strong>, <i>, <span>, <u> and <s> are **inline level elements**.

HTML Block level elements are elements who behave like blocks, like <p>, <h1>, <div>, <ul>, <ol>, <pre> and <address>. These elements always starts from a new line and occupy full width of parent element. **Block elements** can contain both **inline elements** and **block elements**. Here are some **block elements**.

HTML Block Level Elements List

Element Name	Code	Use
Html Tag	<html> </html>	Driver tag to build html page.
Body Tag	<body> </body>	To group visible content of a webpage.

Para Tag	<p> </p>	Create new **paragraph**
Pre Tag	<pre> </pre>	Create **pre formatted text**.
hr	<hr>	**Thematic Break** or formally known as **Horizontal rule**, used to break with gradient shadow.
Blockquote	<blockquote> </blockquote>	Create a **blockquote** from new line.
Div Tag	<div> </div>	Create new New **Division**
ul Tag	<ul> </ul>	Create new **Unordered List**.
ol Tag	<ol> </ol>	Create new **Ordered List**.
Address	<address> </address>	Create **Postal Address**
Headings	<h1> </h1>, <h2> </h2> till <h6> </h6>	Create **Headings** and sub-headings.
Form Tag	<form> </form>	Used to group **Form controls and send form data.**.
Fieldset	<fieldset> </fieldset>	This is a fieldset, used to group form element.

HTML **Block Level** elements always start from a new line and occupy full width of parent element. They support **width**, **height** and **text-align** as they occupy full width of container.

HTML **inline elements** always start in the same line. Their width is equal to their content. Maximum **inline elements** are presentational, for exp, <b>, <i>, <s>, <u>. Some functional **inline elements** are <strong>, <em>, <del>, <time> etc.

HTML Inline Level Elements List

Element Name	Code	Use
span Tag	<span> </span>	Used to group **inline elements**.
anchor Tag	<a href="">Link</a>	Used to create **hyperlinks**.
b Tag	<b> </b>	Used to give **bold appearance**.
i Tag	<i> </i>	Presentational Element used to **italicize** text.
Strong Tag	<strong> </strong>	Gives **bold appearance** and highlight content in **searching** .
em tag	<em> </em>	**Italicize** text and highlight content in **searching** .
small tag	I am <small>small</small>	**small** print.
u tag	<u> </u>	**underline** text.

s tag	<s> </s>	Shows **struck** text .
del tag	<del> </del>	Shows **deleted** text .
sup tag	<sup> </sup>	Shows **superscript** text.
sub tag	<sub> </sub>	Shows **subscript** text .
abbr tag	<abbr title="Prime Minister">PM</abbr>	Shows full version of **abbreviation** in title tag .
kbd tag	<kbd> </kbd>	Shows **keyboard command** .
code tag	<code> </code>	To show **computer code** .
q tag	<q> </q>	To show **quotes** .
cite tag	I resides in <cite>India </cite>	To show **cited title** of work.
samp tag	<samp> </samp>	To show **sample**.
ins tag	<ins> </ins>	To indicate addition to document
var tag	<var> </var>	to show **variables** in code .

HTML Inline Level elements are elements generally used inside **block level elements**. They doesn't support **width**, **height** and **text-align** as are used inside line. Only **<img>** tag supports width and height.

Div tag

Html div tag is used to create divisions. Div is block level, thus starts from new line and occupy full width of parent. Div can group all **block level** elements**.**

```
<div>
   // your content goes here
</div>
```

Span tag

Html **span tag** is used to create division inside line. Span is **inline level**, thus starts in the same line and occupy width of content. Span is used to group **inline level elements**.

```
<span>
   // your content goes here
</span>
```

HTML Lists

HTML Lists includes three **lists** to show single or multiple **list item**.

There are **three types of list in HTML**, i.e, **unordered lists** , **ordered lists** & **description list.**

Unordered lists and **ordered lists** work the same way, except that the **Unordered** is used for **non-sequential lists** with list items usually preceded by bullets and the letters is for **Ordered List**, which are normally represented by **incremental numbers**.

The **ul tag** is used to define **unordered list** and the **ol tag** is used to define **ordered list**. Inside the list, the **li** tag is used to define each **list item**.

By default, **unordered list** are styled with *disk* (•). Till html4, type attribute was used to change list style. But in HTML5, type attribute of **unordered list** is deprecated. But we can remove or change list style using CSS list-style property..

Use **Unordered List** where sequence is not required

```
<ul>

    <li>List 1</li>

    <li>List 2</li>

    <li>List 3</li>

    <li>List 4</li>

</ul>
```

Unordered List with type "none"

```
<ul type="none">


          <li>List 1</li>


          <li>List 2</li>


          <li>List 3</li>


          <li>List 4</li>
```

```
</ul>
```

Unordered List with type "square"

```
<ul type="square">
        <li>List 1</li>
        <li>List 2</li>
        <li>List 3</li>
        <li>List 4</li>
</ul>
```

Unordered List with type "circle"

```
<ul type="circle">
        <li>List 1</li>
        <li>List 2</li>
        <li>List 3</li>
        <li>List 4</li>
</ul>
```

Nested Unordered List

An **unordered list** can have another descendant unordered list, but only inside list item.
A <*ul*> cannot have another <*ul*> element. Only <*li*> is allowed as chlid element of ul or ol.

- List 1
- List 2
- List 3
    - List 31
    - List 32
- List 4

```
<ul>

    <li>List 1</li>

    <li>List 2</li>

    <li>List 3

        <ul>

            <li>List 31</li>

            <li>List 32</li>

        </ul>

    </li>

    <li>List 4</li>

</ul>
```

Ordered list are Sequential List. **ol** use numbers, alphabets and Roman characters as list style. **Ordered list** are countable. By default, ordered list are styled with numbers.

Type of Ordered List

1. Number
2. Roman
3. Alphabet

Ol type number

Default type of ol is number.

1. List 1
2. List 2
3. List 3
4. List 4

```
<ol>

        <li>List 1</li>

        <li>List 2</li>

        <li>List 3</li>

        <li>List 4</li>

</ol>
```

OL type Roman

to start an **ordered list** with roman characters, use type I or i.

Ordered list with uppercase roman

I. List 1
II. List 2
III. List 3
IV. List 4

```
<ol type="I">

        <li>List 1</li>

        <li>List 2</li>

        <li>List 3</li>

        <li>List 4</li>

</ol>
```

Ordered list with lowercase roman

i. List 1
ii. List 2
iii. List 3
iv. List 4

```
<ol type="i">
```

```
        <li>List 5</li>

        <li>List 6</li>

        <li>List 7</li>

        <li>List 8</li>

</ol>
```

OL type Alphabet

Ordered List can have alphebets, both uppsecase and lowercase.

Ordered list with uppercase

A. List 1
B. List 2
C. List 3
D. List 4

```
<ol type="A">

        <li>List 1</li>

        <li>List 2</li>

        <li>List 3</li>

        <li>List 4</li>

</ol>
```

Ordered list with lowercase

a. List 1
b. List 2
c. List 3
d. List 4

```
<ol type="a">

        <li>List 1</li>

        <li>List 2</li>

        <li>List 3</li>

        <li>List 4</li>

</ol>
```

Start Attribute in Ordered List

start attribute in ordered list can start list from n number, instead of 1, A, a, I or i. value of **start attribute** is always a number.

Ordered List starting from 10

10. List 1
11. List 2
12. List 3
13. List 4

```
<ol type="1" start="10">

        <li>List 1</li>

        <li>List 2</li>

        <li>List 3</li>

        <li>List 4</li>

</ol>
```

Ordered List starting from X

X. List 5
Y. List 6
Z. List 7

```
<ol type="A" start="24">

        <li>List 5</li>

        <li>List 6</li>

        <li>List 7</li>

</ol>
```

Reversed Attribute

Ordered List can also have optional **reversed attribute**. **Reversed attribute** can reverse sequence or order of ordered list.

10. List 1
11. List 2
12. List 3
13. List 4

```
<ol start="10" reversed>

        <li>List 1</li>

        <li>List 2</li>

        <li>List 3</li>

        <li>List 4</li>

</ol>
```

List is a list with **description term** and **description data**.

How to use description list

LFC
BEST TRAINING INSTITUTE IN ODISHA.
COURSES
ANDROID,WEB DEV ETC

```
<dl>

            <dt> LFC

 </dt>
<dd> BEST TRAINING INSTITUTE IN ODISHA.</dd>


</dl>
```

```
<dl>
```

```
                <dt> COURSES
```

```
</dt>
```

```
                <dd> ANDROID,WEB DEV ETC
```

```
 </dd>


</dl>
```

Hyperlink

HTML Hyperlinks are created using <a> tag. Formarly known as anchor tag, **hyperlink** defines a **link** <a href=""> in html document.
A **hyperlink** links a webpage with other webpages and external pages

In HTML5, an **hyperlink** can contain both **block level** and **inline level** elements. That means, we can write **text**, image or a div inside **hyperlink**.

Anchor Tag

Anchor Tag was the previous name of **hyperlinks** till html4/xhtml. But HTML5 renamed anchor tag to **hyperlink**.

```
<a>LFC </a>
```

Hyperlink

Hyperlink means an **a tag** with **href attribute**. **Hyperlink** is used to link webpages.

```
<a href="www.learnforcause.com">LFC </a>
```

Type of Hyperlinks

There are six types **hyperlinks**. **Types of Hyperlinks** are defined on the basis of their value. Here are six type of html hyperlinks with example and usage.

Hyperlinks Types

1. Empty Link
2. Relative Link
3. Absolute Link
4. Internal Link
5. Mailto Link
6. Telephone Link

Empty Link

An hyperlink without href value id blank link. Usually we avoid empty hyperlinks as they refresh or reload webpages.

Empty Link

```
<a href="">Empty Link</a>
```

Prevent a link to reload page

Empty Link

```
<a href="JavaScript:Void()">Empty Link</a>
```

Relative Link

Relative link means a hyperlink with path relative to root directory. If your webpage is saved on desktop, then desktop is your root directory.

```
<a href="index.html">LFC HOME </a>
```

Absolute Link

Absolue Link is hyperlink with **Absolute Path**. This could be *http*, *https*, or *file* protocol based.

```
<a href=" www.learnforcause.com ">LFC </a>
```

Internal Link

Internal Link is hyperlink with **Internal Path**. To call an element with id, **internal link** is used. As id is unique, internal link call that element.

```
<a href="#index">index</a>

<a href="#about">about</a>

<a href="#contactus">contact us</a>


<div id="index">index sub </div>

<div id="about">about division</div>

<div id="contactus">contact division</div>
```

Email Link

Mailto Link is used to create **Click to mail link** or **Email Link** in html. This is very useful when we are showing our mail to user. On click of **mail link**, your mail application, (Outlook, Gmail, mail etc) will open compose tab.

Mail Link

```
<a href="info@LFC.in ">Mail </a>
```

Click To Call Link

```
<a href="tel:+918249250694">Call LFC</a>
```

Hypertext Reference (href="")

href attribute is used in an **anchor tag** to specify the URL(*Uniform Resource Location*) of the page link to go. It can be a **relative link**, **absolute link** or an id.

Relative Link

A relative Link is used to link pages within same domain.

Example of Relative link

```
<a href="contact.html">Link 1</a>
```

Relative link in same directory

```
<a href="blog/blog1.html">Link 2</a>
```

Relative link in child directory

```
<a href="../cart.html">Link 3</a>
```

Relative link in parent directory

Absolute Link

An **Absolute Link** is used to link pages from their Absolute Path. for exp <a href="http://www.domain.com">Link</a> is an Absolute Link as it contains domain name and page.

Example of Absolute link?

```
<a href="http:// www.learnforcause.com

">Link</a> ( Absolute Link)
```

```
<a href="https://www.facebook.com/LFC ">Join us on Facebook</a>
```

(Absolute Link of facebook page)

Target attribute

A target attribute (target="") is used to tell window where to open that particular link.
Various values of target

1. _self (By Default)Open link in same window
2. _blankopen link in new tab

3. name of frame(iframe or frame)Open the link on frame whose name is there

Values of target attribute

```
<iframe name="frame1" ></iframe>




<a href="https://www.LFC.in " target="_blank">_blank</a>

<a href="index.html" target="frame1">frame1</a><
```

Click on the link to see behavior of following target

_blank
frame1

HTML Image Tag

HTML Images are defined within **<img> tag**. **Image tag** or **<img>** is a unpair element. src and alt attributes are used to add properties to image.

HTML includes **image tag** from its 4th version **(HTML4)** ie since 1997. Images enhance the look and feel of a website. We uses images as a **logo**, **banner**, **icons**, **symbols**, **product** etc.

How to insert image in html

```
    <img src="" alt="">
```

HTML Images Formats and Comparision

Html supports four popular images extensions, jpg, png, gif and svg.

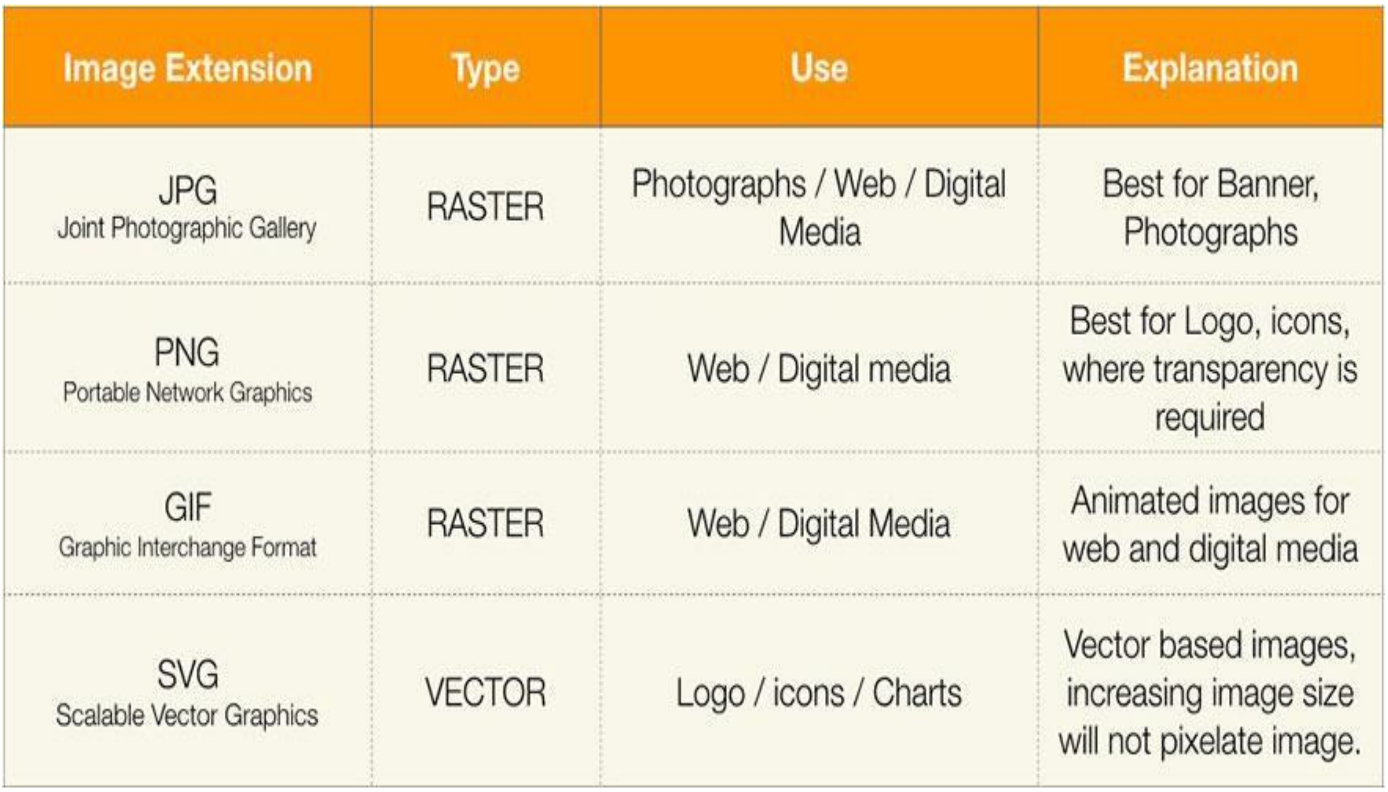


Image Extension	Type	Use	Explanation
JPG Joint Photographic Gallery	RASTER	Photographs / Web / Digital Media	Best for Banner, Photographs
PNG Portable Network Graphics	RASTER	Web / Digital media	Best for Logo, icons, where transparency is required
GIF Graphic Interchange Format	RASTER	Web / Digital Media	Animated images for web and digital media
SVG Scalable Vector Graphics	VECTOR	Logo / icons / Charts	Vector based images, increasing image size will not pixelate image.

Image Extensions comparison, jpg vs png vs gif vs svg

Image extensions for web

- Joint Photographic Gallery (jpg)
- Portable Network Graphics (png)
- Graphics Interchange Format (gif)
- Scalable Vector Graphics (svg)

How to insert image in a webpage

Image is inserted in html using <img> tag. **Img** is an non-pair element. **src** and **alt** attributes are compulsory in an **img tag**. **width** and **height** are optional in img. By increasing width or height, image size increase with same aspect-ratio. Aspect Ratio is ratio of width Vs height. For an image of size 400*300, aspect ratio is 4:3.

HTML image tag example

```
<img src="logo.png" alt="alternate text of image" >
```

Image Attributes

Attribute	Use
src	**source or path of image** (compulsory)
alt	**alternate text** for image (compulsory)
title	show tooltip on mouse over
width	control width of an image
height	control height of an image

Alt Attribute

alt attribute is the compulsory attribute for image tag. It helps google and other search engines to search images in search results. If in any case image fail to upload, alt text hold the space and tell user about image.

What happen if src is invalid in image?

```
  <img src="img/logo" alt="LFC Logo">
```

alt attribute is compulsory, but value of alt can be blank.

What is Iframe

Iframe is a inline element used to embed other webpages within a page. **Iframe** can also be used to embedded **Youtube video**, **show google map** to a webpage.

```
<iframe src="https://www.LFC.in "></iframe>
```

Target Iframe on Hyperlink

A hyperlink can open a webpage in iframe. To do this, name your irfame, and set target to hyperlink as iframe's name. If iframe name is *frame1*, hyperlink target should be frame1.

```
    <iframe name="frame1" ></iframe>

  <a href="img/img1.jpg" target="frame1">Link 1</a>

  <a href="img/img2.jpg" target="frame1">Link 2</a>

  <a href="img/img3.jpg" target="frame1">Link 3</a>

  <a href="img/img4.jpg" target="frame1">Link 4</a>
```

Embed Google Map in iframe

Iframes are also used to **embed google map** on a webpage. This is a free service by Google Maps. To **embed goggle maps in iframe**, follow these steps

How to embed Google Map

1. Open Google Maps.
2. Click Left menu.
3. Choose share or embed map
4. In popup window, choose Embed a map.
5. Choose the size of iframe.
6. Copy iframe and paste on your webpage.
7. Remove frameborder="0" from copied iframe.
8. Save the code and open in browser.

9.
   ```
   <iframe src="https://www.google.com/maps/embed?pb=!1m16!1m12!1m3!1d3503.575022750921!2d77.31301609999998!3d28.58252155000001!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!2m1!1sTech+Altum%2C+Naya+Bans+Village%2C+Sector+15%2C+Noida%2C+Uttar+Pradesh%2C+India!5e0!3m2!1sen!2s!4v1391062749471" style="width:100%; border:0" height="300"></iframe>
   ```

Embed youtube video in iframe

Like google map, a **youtube video** can also be embedded into iframe and publish on website. To **embed a youtube video**, follow these steps.

1. Open youtube website.
2. Click share button.
3. Choose embed.
4. Copy embed video code.
5. Remove frameborder="0" from iframe.

```
<iframe src="https://www.youtube.com/embed/KAlTUVH576YZ" allow="accelerometer; autoplay; encrypted-media; gyroscope; picture-in-picture" allowfullscreen></iframe>
```

HTML Entities

HTML Entities are used to display reserved characters and special characters on a webpage. Html can not show all characters. Some characters are reserved by html.

To write **special characters** in HTML, we use **HTML Entities**. We can write an HTML entity by **Entity No** or **Entity name**.

We can also use EMOJIS in HTML using Entities. HTML Emojis List.

HTML Reserved Characters

Reserved Characters in HTML.

1. <
2. >
3. &

HTML Entities List

HTML Entities can be written using **Entity name** or **Entity Number** (ASCII Code). To use **Entity Name**, type &entityname; and to use **Entity Number**, type &#ascii;. To write Entity Number in Hexadecimal, use &#x*hexecode*;. See examples

Symbol	Description	Entity Name	Entity Number
	Space		&#32;
!	Exclamation Mark		&#33;
"	Quotation Mark		&#34;
#	Hash		&#35;

$	Dollar Sign	&dollar;	&#36;
%	Percent Sign		&#37;
&	Ampersand Sign	&amp;	&#38;
'	Apostrope		&#39;
(	Opening Brackets		&#40;
)	Closing Bracket		&#41;
*	Asterisk		&#42;
+	Plus		&#43;
,	Comma		&#44;
-	Hyphen		&#45;
.	Period		&#46;
/	Forward Slash		&#47;
0	Zero		&#48;
1	One		&#49;
2	Two		&#50;
3	Three		&#51;
4	Four		&#52;

5	Five		&#53;
6	Six		&#54;
7	Seven		&#55;
8	Eight		&#56;
9	Nine		&#57;
:	Colon		&#58;
;	Semi Colon		&#59;
<	Less Then	&lt;	&#60;
=	Equal To		&#61;
>	Greater Then	&gt;	&#62;
?	Question Mark		&#63;
@	"At" Symbol		&#64;
A	Upper case A		&#65;
B	Upper case B		&#66;
C	Upper case C		&#67;
D	Upper case D		&#68;
E	Upper case E		&#69;
F	Upper case F		&#70;
G	Upper case G		&#71;
H	Upper case H		&#72;

I	Upper case I		&#73;
J	Upper case J		&#74;
K	Upper case K		&#75;
L	Upper case L		&#76;
M	Upper case M		&#77;
N	Upper case N		&#78;
O	Upper case O		&#79;
P	Upper case P		&#80;
Q	Upper case Q		&#81;
R	Upper case R		&#82;
S	Upper case S		&#83;
T	Upper case T		&#84;
U	Upper case U		&#85;
V	Upper case V		&#86;
W	Upper case W		&#87;
X	Upper case X		&#88;
Y	Upper case Y		&#89;
Z	Upper case Z		&#90;
[	Brackets Open		&#91;
\	Back Slash		&#92;
]	Brackets Close		&#93;
^	Caret		&#94;
_	Underscore		&#95;
`	Single Quote		&#96;
a	Lower case a		&#97;
b	Lower case b		&#98;

c	Lower case c		&#99;
d	Lower case d		&#100;
e	Lower case e		&#101;
f	Lower case f		&#102;
g	Lower case g		&#103;
h	Lower case h		&#104;
i	Lower case i		&#105;
j	Lower case j		&#106;
k	Lower case k		&#107;
l	Lower case l		&#108;
m	Lower case m		&#109;
n	Lower case n		&#110;
o	Lower case o		&#111;
p	Lower case p		&#112;
q	Lower case q		&#113;
r	Lower case r		&#114;
s	Lower case s		&#115;
t	Lower case t		&#116;
u	Lower case u		&#117;
v	Lower case v		&#118;
w	Lower case w		&#119;
x	Lower case x		&#120;
y	Lower case y		&#121;
z	Lower case z		&#122;
{	Opening Curly Brace		&#123;

|	Vertical Line		&#124;
}	Closing Curly Brace		&#125;
~	Tilde		&#126;
•	Delete		&#127;
€	Euro	&euro;	&#128;
,	comma		&#130;
ƒ	f with hook		&#131;
„	Quotation mark		&#132;
…	ellipsis	&hellip;	&#133;
†	Dagger	&dagger;	&#134;
‡	Double dagger		&#135;
ˆ	Circumflex	&circ;	&#136;
‰	Per mile	&permil;	&#137;
Š	grapheme, s with caron	&scaron;	&#138;
‹	Guillemet		&#139;
Œ			&#140;
Ž	The grapheme		&#142;
‘	Open Single quote		&#145;
’	Close Single Quote		&#146;
“	Open Double Quote	&quot;	&#147;
”	Close Double Quote	&quot;	&#148;
•	Bullet	•	&#149;
–	Hypen		&#150;
—	Dash	&dash;	&#151;

˜	Tild	&tilde;	&#152;
™	Trade Mark	&trade;	&#153;
š	grapheme, S with caron		&#154;
›	Guillemet		&#155;
œ			&#156;
ž	grapheme		&#158;
Ÿ			&#159;
¡	exclamation mark		&#161;
¢	Cent	&cent;	&#162;
£	Pound	&pound;	&#163;
¤	Currency Sign	&curren;	&#164;
¥	Yean	&yen;	&#165;
¦	broken bar	&brvbar;	&#166;
§	Section	&sect;	&#167;
¨	diaeresis		&#168;
©	Copyright	&copy;	&#169;
ª	Ordinal indicator, feminine		&#170;
«	guillemets	&laquo;	&#171;
¬	Negation		&#172;
®	Registered	&reg;	&#174;
¯	Macron	&macr;	&#175;
°	Degree	&deg;	&#176;
±	Plus Minus	&plusmn;	&#177;
²	Power 2		&#178;
³	Power 3		&#179;

´	Acute accent	&acute;	&#180;
µ	Micro Symbol	&micro;	&#181;
¶	Paragraph symbol	&para;	&#182;
·			&#183;
¸			&#184;
¹			&#185;
º			&#186;
»		&raquo;	&#187;
¼	One Fourth		&#188;
½	One Half		&#189;
¾	Three Fourth		&#190;
¿	Inverted Question Mark		&#191;
À	agrave	&agrave;	&#192;
Á	aacute	&aacute;	&#193;
Â	A-circumflex	&Acirc;	&#194;
Ã	A with tilde	&Atilde;	&#195;
Ä	A with umlaut	&Auml;	&#196;
Å	A with ring	&Aring;	&#197;
Æ	A with		&#198;
Ç	C cedilla	&Ccedil;	&#199;
È	E Grave	&Egrave;	&#200;
É	E Acute	&Eacute;	&#201;
Ê	E Circumflex	&Ecirc;	&#202;
Ë	E Umlaut	&Euml;	&#203;
Ì	I acute	&Iacute;	&#204;

Í	I grave	&Igrave;	&#205;
Î	I acute	&Icirc;	&#206;
Ï	I Umlaut	&Iuml;	&#207;
Ð	E th		&#208;
Ñ			&#209;
Ò	O grave	&Ograve;	&#210;
Ó	O acute	&Oacute;	&#211;
Ô	O with	&Ocirc;	&#212;
Õ	O with	&Otilde;	&#213;
Ö	O with	&Ouml;	&#214;
×	Multiplication Sign		&#215;
Ø		&Oslash;	&#216;
Ù	U with	&Ugrave;	&#217;
Ú	U with	&Uacute;	&#218;
Û	U with	&Ucirc;	&#219;
Ü	U with	&Uuml;	&#220;
Ý	Y with		&#221;
Þ			&#222;
ß	Beta	&beta;	&#223;
à	a with		&#224;
á	a with		&#225;
â	a with		&#226;
ã	a with		&#227;
ä	a with		&#228;
å	a with		&#229;
æ	a with		&#230;

ç			&#231;
è	e with		&#232;
é	e with		&#233;
ê	e with		&#234;
ë	e with		&#235;
ì	i with		&#236;
í	i with		&#237;
î	i with		&#238;
ï	i with		&#239;
ð	i with		&#240;
ñ	n with		&#241;
ò	o with		&#242;
ó	o with		&#243;
ô	o with		&#244;
õ	o with		&#245;
ö	o with		&#246;
÷	Divide		&#247;
ø	oslash	&oslash;	&#242;
ù	ugrave	&ugrave;	&#249;
ú	uacute	&uacute;	&#250;
û	ucirc	&ucirc;	&#251;
ü	uuml	&uuml;	&#252;
ý			&#253;
þ			&#254;
ÿ	yuml	&yuml	&#255;
€	euro	&euro;	&#8364;

₹	**Indian Rupee**	....	&#8377;
™	trademark	&trade;	&#8482;
→	Right Arrow	&rarr;	....
←	Left Arrow	&larr;	....
“	Open Curly Double Quote	&ldquo;	&#8220;
”	Close Curly Double Quote	&rdquo;	&#8221;
♠	Black Spade Suit	&spades;	....
♣	Black Club Suit	&clubs;	....
♥	Black Heart Suit	&hearts;	....
♦	Black Diamond Suit	&diams;	....

Emoji unicode or entities

HTML Entities can also be used to display **HTML Emojies**. **Emojies** are written in **hexadecimal**.

Emoji	Emoji Entity
□	&#x1F601;
□	&#x1F602;
□	&#x1F603;
□	&#x1F604;
□	&#x1F605;

□	&#x1F606;
□	&#x1F607;
□	&#x1F608;
□	&#x1F609;
□	&#x1F610;
□	&#x1F611;
□	&#x1F612;
□	&#x1F613;
□	&#x1F614;
□	&#x1F615;
□	&#x1F616;
□	&#x1F617;
□	&#x1F618;
□	&#x1F619;
□	&#x1F620;

□	&#x1F621;
□	&#x1F622;
□	&#x1F623;
□	&#x1F624;
□	&#x1F625;
□	&#x1F626;
□	&#x1F627;
□	&#x1F628;
□	&#x1F629;
□	&#x1F630;
□	&#x1F631;
□	&#x1F632;
□	&#x1F633;
□	&#x1F634;
□	&#x1F635;

□	&#x1F636;
□	&#x1F637;
□	&#x1F638;
□	&#x1F639;
□	&#x1F640;
□	&#x1F641;
□	&#x1F642;
□	&#x1F643;
□	&#x1F644;
□	&#x1F645;
□	&#x1F646;
□	&#x1F647;
□	&#x1F648;
□	&#x1F649;

Meta Tag

HTML meta tags or **metadata** are used to add metadata to a webpage. Usually **meta tags** are defined in **<head> tag** as these data are for search engines. **Meta Tags** were earlier known as **meta tags**, but in HTML5, the new name is **metadata**.

Meta tags are used in head to add **meta charset**, **meta description**, **meta viewport**, **meta open graph** etc.

Meta Tag example

```
<meta>
```

Meta Charset

meta charset is used to specify the **character encoding** of webpage. It is compulsory to declare meta charset in a webpage.

For Windows OS, the default value for **meta charset** is **windows-1252**.

UTF-8 or **Unicode Transformation Format** is the the most popular charset. **UTF-8** can render almost all characters like alphabets, numbers, greek, devnagri, spanish, french, chinese etc. **UTF-8** is also the default charset of **Chrome Browser**.

Meta charset

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

Meta Description

Meta description is the metadata used to declare **description for search engines**. Almost all search engines used **meta description** to know **description of a webpage**.

It is recommended to used **meta description** in head for search engines. The recommended character length for **meta description** is 160 characters.

Meta Description example

```
<meta name="description" content="type your description here">
```

Meta description on google search engine

Meta Viewport

Meta Viewport is used in head. **Viewport** means the visible part of screen on any device.

Meta viewport is used in Responsive Web Design to control the width, scaling and user scablility.

Without viewport, browser display a webpage as per the actual screen size. But **meta viewport** adjust the screen fonts as per the device pixel ratio. This means, we can see same desktop font on mobile device without zooming.

Meta Viewport Examaple

Viewport for most of the sites

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

Viewport for the sites with no zoom

```
<meta name="viewport" content="width=device-width, initial-scale=1,
maximum-scale=1, user-scalable=no">
```

Viewport Width Vs Actual Width

Viewport Width Vs Actual Width		
Device	Viewport Size	Actual Size
Iphone X	375*812	1125*2436
Iphone 8 Plus	414*736	1080*1920
Iphone 8	375*667	750*1334
Ipad (9.7 inch)	768*1024	1536*2048
Samsung Galaxy S8, S9	360*740	1440*2960
Macbook Pro (13 inch)	1280*800	2560*1600

HTML Table Tag

HTML Tables are used to show **Tabular data**. **HTMl Table** is defined with the <table> tag and then table row <tr> and
cells <td> or <th>**.** **Table** tag is the parent of table. **Table** can have rows, data, captions, colgroup, cols, etc.

Till 2005, whole website was build using **table tag**, but later or Div Based Layouts become popular as div based layouts are easy to build and maintain than table based layout. Also table is not scalable. Using a large table can block content loading.

HTML Table Example

row 1 - cell 1, row 1 - cell 2,
row 2 - cell 1, row 2 - cell 2,

```
<table>

    <tr>

        <td>row 1, cell 1</td>

        <td>row 1, cell 2</td>

    </tr>

    <tr>

        <td>row 2, cell 1</td>

        <td>row 2, cell 2</td>

    </tr>

</table>
```

Table Tags

Here is a list of Tags used in table. Table is started with <table> tag. Inside table tag, we can have rows <tr> and columns <td>. Here s a list of tags used in table.

Table Tags

Tag Name	**Description**
<table>	Defines a table Element
<tr>	Defines a table row
<td>	Defines a table cell or table data
<th>	Defines a table header cell

Table Tags

Tag Name	Description
<caption>	Defines a table caption
<colgroup>	Defines a group of columns in a table, for formatting
<col>	Defines attribute values for one or more columns in a table
<thead>	Groups the header content in a table
<tbody>	Groups the body content in a table
<tfoot>	Groups the footer content in a table

HTML Forms are used to **send data** across the web and are often used as **contact form** to convert information input by a user into Leads.**HTML forms** are used to pass data to the server.

The elements used in **HTML form** are **form tag** as parent, **input**, **textarea,**, **select**, **button** and **label**.

HTML Form Tag

Form Tag defines the **form** and within this tag, there is **action attribute** which tells the form where its contents will be sent when it is submitted.

An **HTML form** can have **input elements**, **checkbox**, **radio buttons**, **submit button** and more.A form can also contain **select dropdown**, **textarea**, **fieldset**, **legend**, and **label** elements.

Create HTML Form

form is build inside <form> tag. See the code below

```
<form action="" method="get" name="formname">

        /* Content */

</form>
```

Form Attributes

HTML Form Attributes

Attribute	Values	Use
method	get or post	http get method submit form data but is visible in url. post includes data in body. more secure as data is not visible to user in url
action	path	the backend file collecting form data
name	any name	name of form control

Input Element

The most important form element is the **input element**.The **input element** is used to get user information. An input element can be of type **text**, **password**, **checkbox**, **radio button**, **submit button** and more.

Attributes in Input element

Attribute Name	values	Use
type	text, password, file, radio, checkbox, button, submit, and reset	type defines **type of input** control.
size	default value is 20	change size of input control
tabindex	any numeric value	used to define a sequence followed by user when he navigate using Tab key
value	any possible value	set a default value of input control
maxlength	n digits	set maximum characters limit
disabled		disabled input control, or fieldset tag
checked		Check checkbox or radio button
multiple		Used in input type file for multiple files upload

Input Type Text

Input type text <input type="text"> is the common **input element** for name, surname, country, numbers and symbols. Default input type is text. Even if we skip type attribute in input element, default input type is text. But still it is recommended to define type attribute in input element.

HTML Input type text

First Name: Last Name:

```
First Name: <input type="text">

Last Name: <input type="text">
```

Label

Label tag is used to write the content just before text field. To Specify particular label, place input inside label or the value of for attribute inside label should match the id of input control.

Input in Label

First Name:

```
<label>First Name: <input type="text"></label>
```

Input Outside Label

Last Name:

```
<label for="lname">Last Name:</label>

<input type="text" id="lname">
```

Input Type Text with value

value attribute can also be used inside input or textarea. Usually we ask user to fill values, but if value is fixed, use value attribute.

First Name:

Last Name:

```
<form>

    <label for="fname">First Name:</label>

    <input type="text" value="First Name" id="fname" >



    <label for="lname">Last Name:</label>

    <input type="text" value="Last Name" id="lname" >

</form>
```

Maxlength

maxlength attribute is used to restrict no of characters in a text field. **maxlength** value is a number. Maxlength attribute is useful for form validations.

Input with maxlength

First Name:

Age:

```
<form>

<label for="fname">First Name:</label>

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



<label for="age">Age:</label>

<input type="text" id="age" maxlength="3">

</form>
```

Input type Password

The **input type password** is used to write passwords. The password value is written in **encrypt form**. i.e. a user cannot see, copy or cut password data from input type password.

Password:

```
<form>

    <label for="pwd">Password:</label>

    <input type="password" id="pwd" >

</form>
```

Input type File

Input type file let user to choose file from his system. This can be used to uplaod a picture, upload resume, upload a video or audio etc.

Default value of input type file is "No file chosen". Once the file is uploaded, the file name replace this text followed by extension.

Resume:

```
<form>

    <label for="resume">Resume:</label>

    <input type="file" id="resume">

</form>
```

Input type file with multiple

Resume:

```
<form>

    <label for="resume">Resume:</label>

    <input type="file" id="resume" multiple>

</form>
```

Radio Buttons

Radio Buttons are used to choose a **single element** among a group. To allow window to choose a single radio, use **name attribute** with **same value** on both radio inputs.

```
<input type="radio" name="gender" id="male">

<label for="m">Male</label>

<input type="radio" name="gender" id="female">
```

<label for="f">Female</label>

Checkbox

Checkbox are used to select multiple selections unlike radio button. They can be checked and unchecked. We can use checkbox for hobbies, interests, terms & conditions, etc.

```
<label> <input type="checkbox"> :Bike</label>

<label> <input type="checkbox"> :Car</label>
```

Checkbox with disabled

Checbox can also have **disabled** attribute. A **disabled** checkbox can't be checked, means checked will remain checked, and unchecked will remain unchecked. See example

HTML Checkbox with disable

```
<<input type="checkbox" disabled>>
```

Checkbox with checked

Default checkbox state is unchecked. But we can change default state to **checked** by using **checked attribute** in input type checkbox. See example

```
<label><input type="checkbox" checked>: I Agree</label>

<label><input type="checkbox" disabled checked>: I Agree </label>
```

Select Dropdown

select or **select dropdown** is used to fetch single or multiple options in dropdown list. **Select** options are fixed, thus used can choose only given option or options. To select city, country, date, month, year etc, **Select Dropdown** is used.

```
<select>

<option selected disabled>--Select City--</option>
```

```
        <option>New York</option>

        <option>Chicago</option>

        <option>Los Angeles</option>

        <option>Washington DC</option>

    </select>
```

Select with Optgroup

Optgroup element is used to group multiple options in select dropdown. The name of optgroup is set using *label attribute* in optgroup.

```
<select>

    <option selected disabled>--Select City--</option>

    <optgroup label="Metros">

        <option>New Delhi</option>

        <option>Kolkata</option>

        <option>Mumbai</option>

        <option>Chennai</option>

    </optgroup>

    <optgroup label="Others">

        <option>Noida</option>

        <option>Gurgram</option>

        <option>Faridabad</option>

        <option>Gaziabad</option>

    </optgroup>
```

```
</select>
```

Multiple Attribute in Select

Multiple attribute is used in select dropdown to select more than one options by using Ctrl or Cmd key.

```
<select multiple>
    <option selected disabled>--Select City--</option>
    <optgroup label="Metros">
        <option>New Delhi</option>
        <option>Kolkata</option>
        <option>Mumbai</option>
        <option>Chennai</option>
<option>Bhubaneswar </option>

    </optgroup>
    <optgroup label="Others">
        <option>Noida</option>
        <option>Gurgram</option>
        <option>Faridabad</option>
        <option>Gaziabad</option>
    </optgroup>
</select>
```

Textarea

Textarea is used to write multiple line. Like post, query, address, comments, reviews etc. Textarea can have **row** and **col** attributes. Default rows are 2, and default columnns are 20.

Submit Button

Submit Button or **input type submit** is used to send information from user to web server. Submit button can be used only once in a form tag.

Submit Button example

```
    <input type="submit">
```

Reset Button

Reset Button or **input type reset** is used to reload form data, without refreshing webpage. Reset is also used once in a form tag.

Create reset Button

```
    <input type="reset">
```

Fieldset

Form or form controls can also be placed inside **fieldset tag**. **Fieldset tag** is used to group form or multiple input controls. **Fieldset** group form controls in bordered area. We can also use **legend tag** inside fieldset.

The main functionality of **Fieldset** is to disable multiple form controls.

Fieldset with Legend

```
<fieldset>

    <legend>Enquiry Form</legend>

        <form onsubmit="return false">

            <label>Name:<input type="text"></label>
            <input type="submit">

            <input type="reset"> </form>

</fieldset>
```

Fieldset with disabled

Fieldset also supports *disabled* attribute. By adding disables attribute, all form controls including submit button are disabled. Thus user cannot fill and submit form data.

```
<fieldset disabled>

    <legend>Enquiry Form</legend>

        <form onsubmit="return false">

            <label>Name:<input type="text"></label>

            <input type="submit">

            <input type="reset"> </form>

</fieldset>
```

Contact Form Example

A complete **HTML Form** with all inputs, select dropdown, radio buttons, checkbox, textarea, submit and reset buttons.

```
<form>

    <table>

        <tr>

            <td>

                <label for="uname">Name</label>

            </td>

            <td>
```

```
                <input type="text" id="uname" name="username">
            </td>
        </tr>
        <tr>
            <td>
                <label for="uemail">Email</label>
            </td>
            <td>
                <input type="text" id="uemail" name="usermail">
                <button type="button">Check</button>
            </td>
        </tr>
        <tr>
            <td>
                <label for="age">Age</label>
            </td>
            <td>
                <input type="text" name="userage" id="age" size="2"
maxlength="2">
            </td>
        </tr>
        <tr>
            <td>
                <label>Country</label>
            </td>
            <td>
                <input type="text" value="India" name="country" disabled>
            </td>
        </tr>
        <tr>
            <td>
                    <label for="pass">Password</label>
            </td>
            <td>
                    <input type="password" id="pass">
            </td>
        </tr>
        <tr>
            <td>
                <label for="res">Resume</label>
            </td>
            <td>
                <input type="file" id="res">
            </td>
        </tr>
        <tr>
```

```
            <td>
                <label>Hobbies</label>
            </td>
            <td>
                <label>
                    <input type="checkbox" checked> Cricket
                </label>
                <label>
                    <input type="checkbox"> Football
                </label>
            </td>
        </tr>
        <tr>
            <td>
                <label>Gender</label>
            </td>
            <td>
                <label>
                    <input type="radio" value="f" name="gender">
Female</label>
                    <label>
                    <input value="m" type="radio" name="gender"> Male</label>
            </td>
```

```
    </tr>

    <tr>

      <td>

          <label for="city">City</label>

      </td>

      <td>

          <select id="city" name="city">

            <option disabled selected>--Choose City--</option>

              <optgroup label="Metros">

                <option>New Delhi</option>

                <option>Mumbai</option>

                <option>Channai</option>

                <option>Kolkata</option>

              </optgroup>

              <optgroup label="Others">

                <option>Noida</option>

                <option>Gurgram</option>

                <option>Faridabad</option>

                <option>Gaziabad</option>

              </optgroup>

          </select>

      </td>

    </tr>
```

```
        <tr>
            <td>
                <label>Address</label>
            </td>
            <td>
                <textarea rows="4" cols="40"></textarea>
            </td>
        </tr>
        <tr>
            <td></td>
            <td>
                <input type="submit" value="Submit">
                <input type="reset">
            </td>
        </tr>
    </table>
</form>
```

Inside form element, button will work as submit button, but outside form, button will work as button.

Marquee Tag

HTML Marquee Tag is a non standard HTML element used to **scroll text**, image or any content from left to right, right to left, bottom to top and top to bottom directions. We can also control **marquee speed**, **marquee**

direction , **marquee scrolldelay** and **Stop or start marquee on mouseover** and **mouseout** using marquee attributes.

To use marquee in Html5, use CSS3 Animations

Marquee Tag Example

```
<marquee> This is a marquee </marquee>
```

Extensible Hypertext Markup Language

XHTML is the **cleaner** and **stricter** version of HTML. It was started in 2000 with version **XHTML1.0 . XHTML 1.1** comes into existence in 2001. XHTML is very much similar to **HTML 4.0**.

XHTML Versions

Version	**Releasing Date**
XHTML 1.0	2000
XHTML 1.1	2001

XHTML Vs HTML

XHTML is different from HTML 4.0 in many ways. Like,

- DOCTYPE declaration is must in an XHTML page which is written at the top of the document.
- XHTML tags and attributes should be written in lower case only.
- Closing tags are compulsory.
- Values of attributes must be quoted.
- All the tags should be properly nested.
- All Non-pair tags should be closed with slash. e.g(< img src=" " /> , < br />)
- Use Id attribute instead on name in XHTML.
- Value of attribute is must.

HTML4 Vs XHTML Vs HTML5

HTML5 is latest version of HTML, whereas, xhtml was strict and cleaner version of HTML4. Here is a quick comparision between html4, xhtml and html5.

HTML4	**XHTML**	**HTML5**
<DIV></DIV>	<div></div>	Both allowed, but lowercase preffered
<img src="" alt="" >	<img src="" alt="" />	Both allowed, but <im src="" alt=""> is preffered
<br>	<br />	<br> is preffered
<img src="" alt="" width=200 >	<img src="" alt="" width="200" />	<img src="" alt="" width="200">
<input type="text" disabled >	<input type="text" disabled="disabled" />	<input type="text" disabled>

THANK YOU