Lecture Notes

for

HTML

By

CLARITECH
ACADEMY

LEARNERS TODAY, LEADERS TOMORROW

HTML Introduction

HTML tutorial or HTML 5 tutorial provides basic and advanced concepts of HTML. Our HTML tutorial is developed for beginners and professionals. In our tutorial, every topic is given step-by-step so that you can learn it in a very easy way. If you are new in learning HTML, then you can learn HTML from basic to a professional level and after learning HTML with CSS and JavaScript you will be able to create your own interactive and dynamic website. But Now We will focus on HTML only in this tutorial.

The major points of HTML are given below:

- HTML stands for Hyper Text Markup Language.
- HTML is used to create web pages and web applications.
- HTML is widely used language on the web.
- We can create a static website by HTML only.
- Technically, HTML is a Markup language rather than a programming language.

HTML Example with HTML Editor

In this tutorial, you will get a lot of HTML examples, at least one example for each topic with explanation. You can also edit and run these examples, with our online HTML editor. Learning HTML is fun, and it's very easy to learn.

```
<!DOCTYPE>

<html>

<head>

<title>Web page title</title>

</head>

<body>

<h1>Write Your First Heading</h1>

<p>Write Your First Paragraph.</p>

</body>

</html>
```

What is HTML

HTML is an acronym which stands for Hyper Text Markup Language which is used for creating web pages and web applications. Let's see what is meant by Hypertext Markup Language, and Web page.

Hyper Text: Hyper Text simply means "Text within Text." A text has a link within it, is a hypertext. Whenever you click on a link which brings you to a new webpage, you have clicked on a hypertext. Hyper Text is a way to link two or more web pages (HTML documents) with each other.

Markup language: A markup language is a computer language that is used to apply layout and formatting conventions to a text document. Markup language makes text more interactive and dynamic. It can turn text into images, tables, links, etc.

Web Page: A web page is a document which is commonly written in HTML and translated by a web browser. A web page can be identified by entering an URL. A Web page can be of the static or dynamic type. With the help of HTML only, we can create static web pages.

Hence, HTML is a markup language which is used for creating attractive web pages with the help of styling, and which looks in a nice format on a web browser. An HTML document is made of many HTML tags and each HTML tag contains different content.

Let's see a simple example of HTML.

```
<!DOCTYPE>

<html>

<head>

<title>Web page title</title>

</head>

<body>

<h1>Write Your First Heading</h1>
```

```
<p>Write Your First Paragraph.</p>

</body>

</html>
```

Description of HTML Example

<!DOCTYPE>: It defines the document type or it instruct the browser about the version of HTML.

<html>: This tag informs the browser that it is an HTML document. Text between html tag describes the web document. It is a container for all other elements of HTML except <!DOCTYPE>

<head>: It should be the first element inside the <html> element, which contains the metadata(information about the document). It must be closed before the body tag opens.

<title>: As its name suggested, it is used to add title of that HTML page which appears at the top of the browser window. It must be placed inside the head tag and should close immediately. (Optional)

<body>: Text between body tag describes the body content of the page that is visible to the end user. This tag contains the main content of the HTML document.

<h1>: Text between <h1> tag describes the first level heading of the webpage.

<p>: Text between <p> tag describes the paragraph of the webpage.

Brief History of HTML

In the late 1980's , a physicist, Tim Berners-Lee who was a contractor at CERN, proposed a system for CERN researchers. In 1989, he wrote a memo proposing an internet based hypertext system.

Tim Berners-Lee is known as the father of HTML. The first available description of HTML was a document called "HTML Tags" proposed by Tim in late 1991. The latest version of HTML is HTML5, which we will learn later in this tutorial.

HTML Versions

Since the time HTML was invented there are lots of HTML versions in market, the brief introduction about the HTML version is given below:

HTML 1.0: The first version of HTML was 1.0, which was the barebones version of HTML language, and it was released in1991.

HTML 2.0: This was the next version which was released in 1995, and it was standard language version for website design. HTML 2.0 was able to support extra features such as form-based file upload, form elements such as text box, option button, etc.

HTML 3.2: HTML 3.2 version was published by W3C in early 1997. This version was capable of creating tables and providing support for extra options for form elements. It can also support a web page with complex mathematical equations. It became an official standard for any browser till January 1997. Today it is practically supported by most of the browsers.

HTML 4.01: HTML 4.01 version was released on December 1999, and it is a very stable version of HTML language. This version is the current official standard, and it provides added support for stylesheets (CSS) and scripting ability for various multimedia elements.

HTML5: HTML5 is the newest version of HyperText Markup language. The first draft of this version was announced in January 2008. There are two major organizations one is W3C (World Wide Web Consortium), and another one is WHATWG(Web Hypertext Application Technology Working Group) which are involved in the development of HTML 5 version, and still, it is under development.

Features of HTML

1) It is a very easy and simple language. It can be easily understood and modified.
2) It is very easy to make an effective presentation with HTML because it has a lot of formatting tags.
3) It is a markup language, so it provides a flexible way to design web pages along with the text.
4) It facilitates programmers to add a link on the web pages (by html anchor tag), so it enhances the interest of browsing of the user.
5) It is platform-independent because it can be displayed on any platform like Windows, Linux, and Macintosh, etc.
6) It facilitates the programmer to add Graphics, Videos, and Sound to the web pages which makes it more attractive and interactive.
7) HTML is a case-insensitive language, which means we can use tags either in lower-case or upper-case.

NOTE: It is recommended to write all tags in lower-case for consistency, readability, etc.

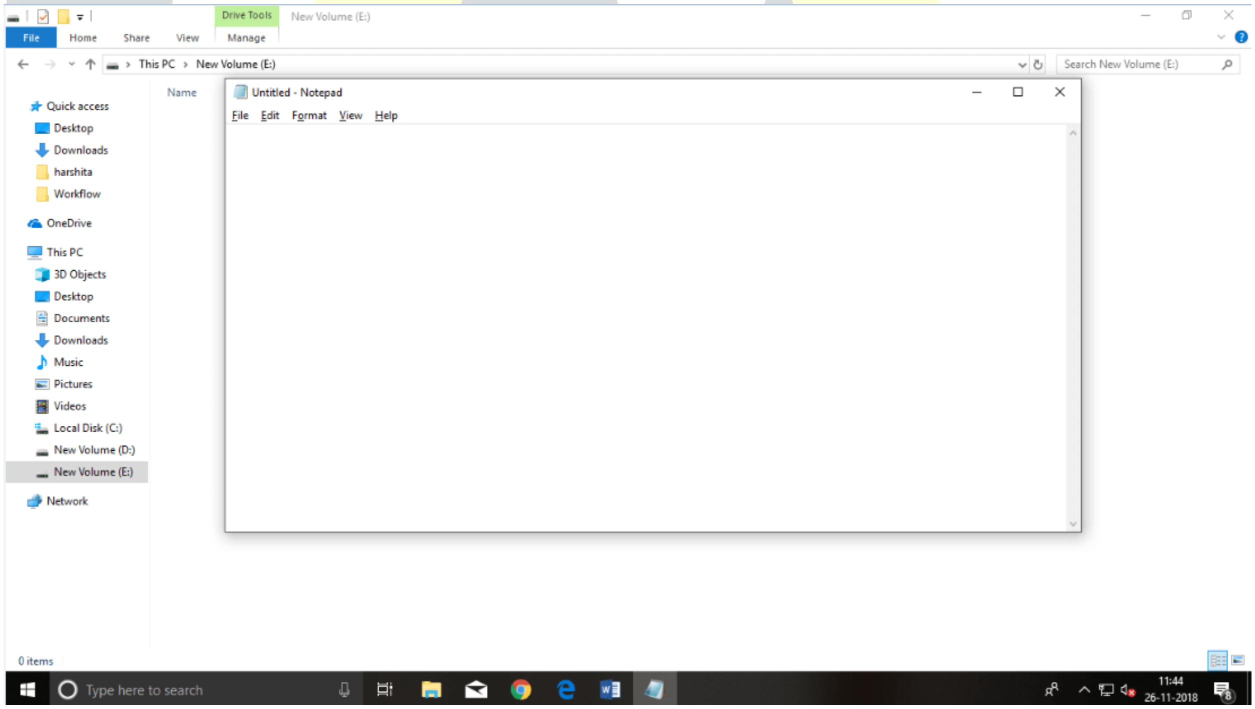
HTML text Editors

- An HTML file is a text file, so to create an HTML file we can use any text editors.
- Text editors are the programs which allow editing in a written text, hence to create a web page we need to write our code in some text editor.
- There are various types of text editors available which you can directly download, but for a beginner, the best text editor is Notepad (Windows) or TextEdit (Mac).
- After learning the basics, you can easily use other professional text editors which are, Notepad++, Sublime Text, Vim, etc.
- In our tutorial, we will use Notepad and sublime text editor. Following are some easy ways to create your first web page with Notepad, and sublime text.

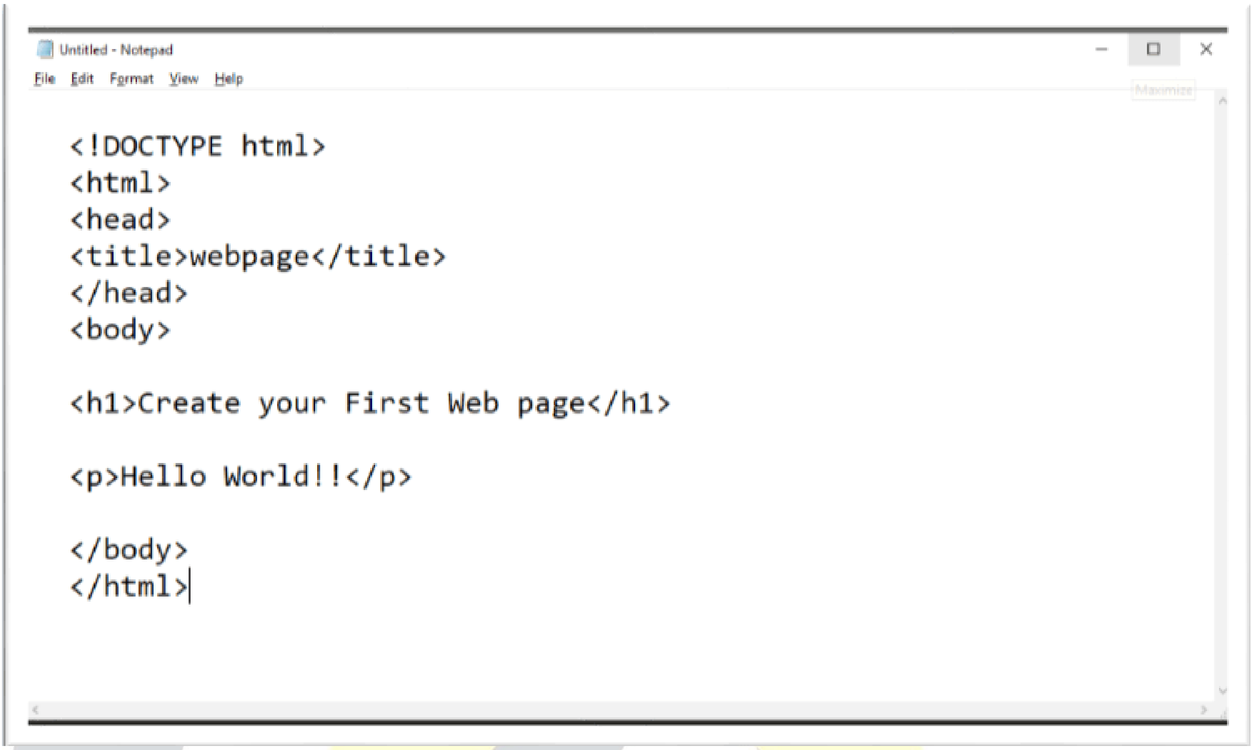
A. HTML code with Notepad. (Recommended for Beginners)

Notepad is a simple text editor and suitable for beginners to learn HTML. It is available in all versions of Windows, from where you easily access it.

Step 1: Open Notepad (Windows)

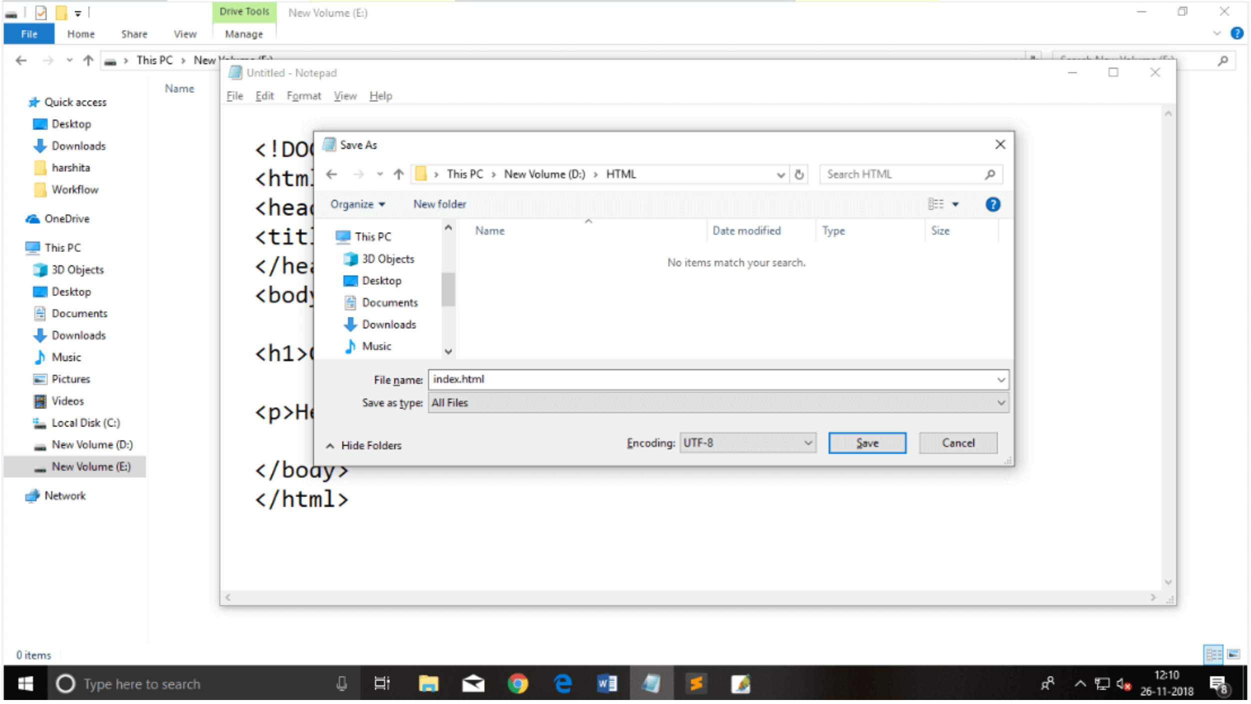


Step 2: Write code in HTML



```
<!DOCTYPE html>
<html>
<head>
<title>webpage</title>
</head>
<body>

<h1>Create your First Web page</h1>

<p>Hello World!!</p>

</body>
</html>
```

Step 3: Save the HTML file with .htm or .html extension.



No items match your search.

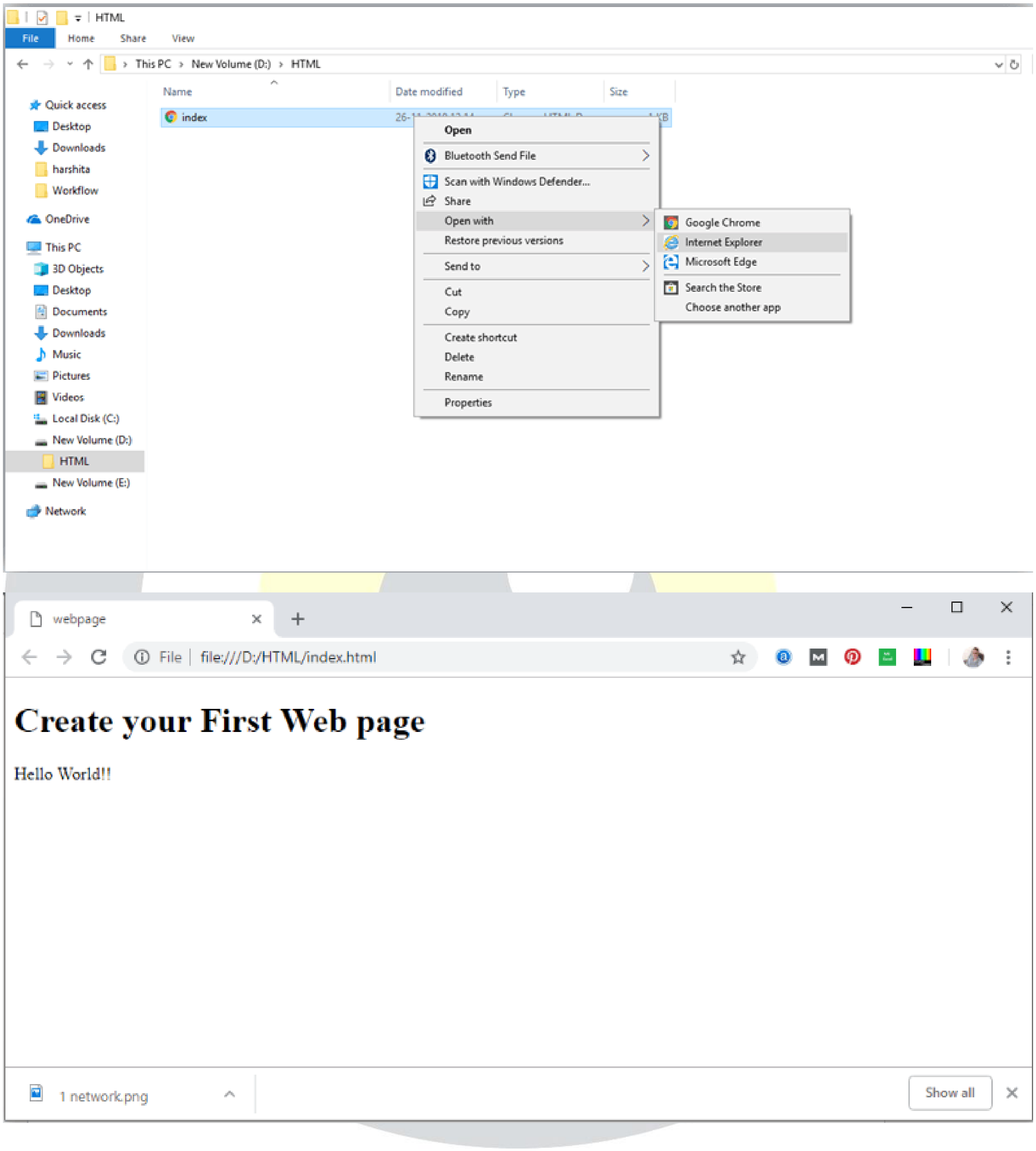
File name: index.html

Save as type: All Files

Encoding: UTF-8

Step 4: Open the HTML page in your web browser.

To run the HTML page, you need to open the file location, where you have saved the file and then either double-click on file or click on open with option



Create your First Web page

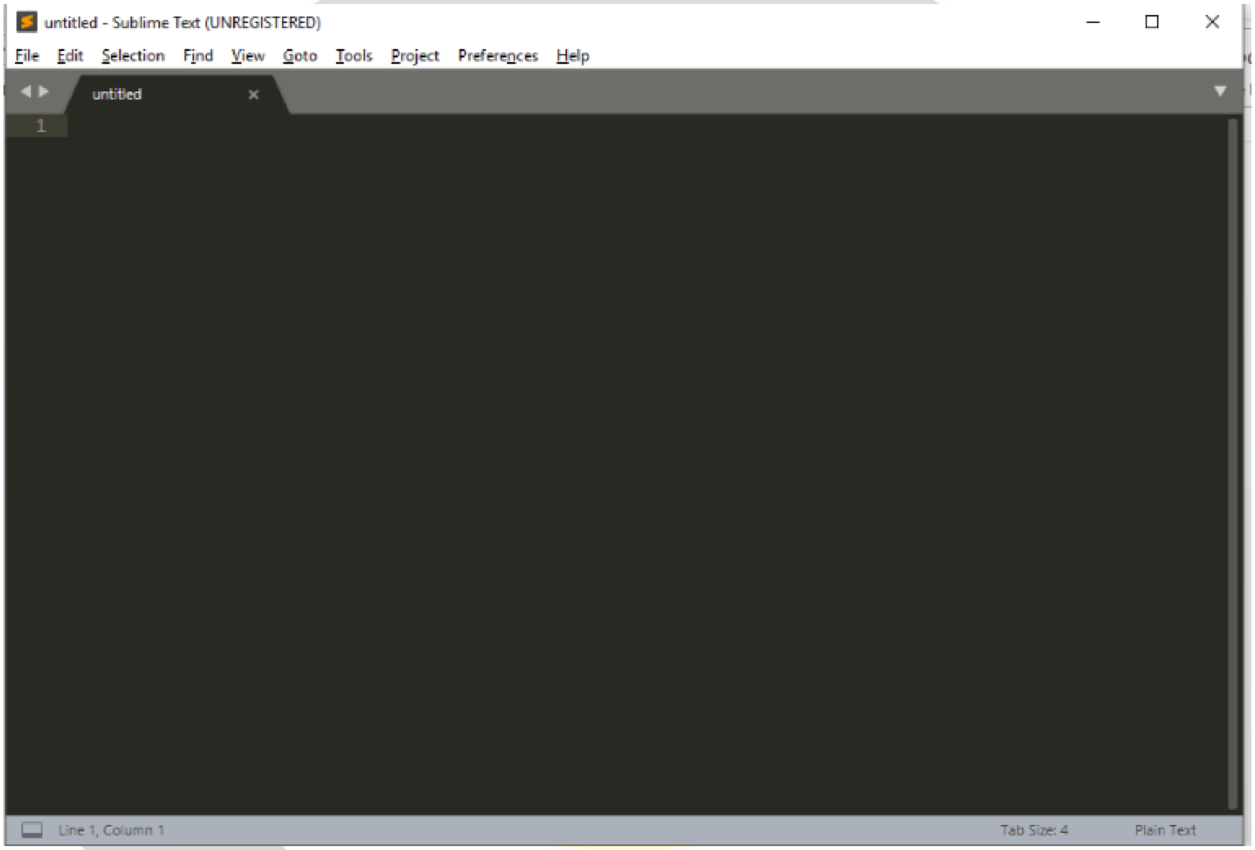
Hello World!!

B. HTML code with Sublime Text-editor.(Recommended after learning basics of HTML)

When you will learn the basics of HTML, then you can use some professional text editors, which will help you to write an efficient and fast code. So to use Sublime Text editors, first it needs to download and install from internet. You can easily download it from this https://www.sublimetext.com/download link and can install in your PC. When installation of Sublime text editor done then you can follow the simple steps to use it:

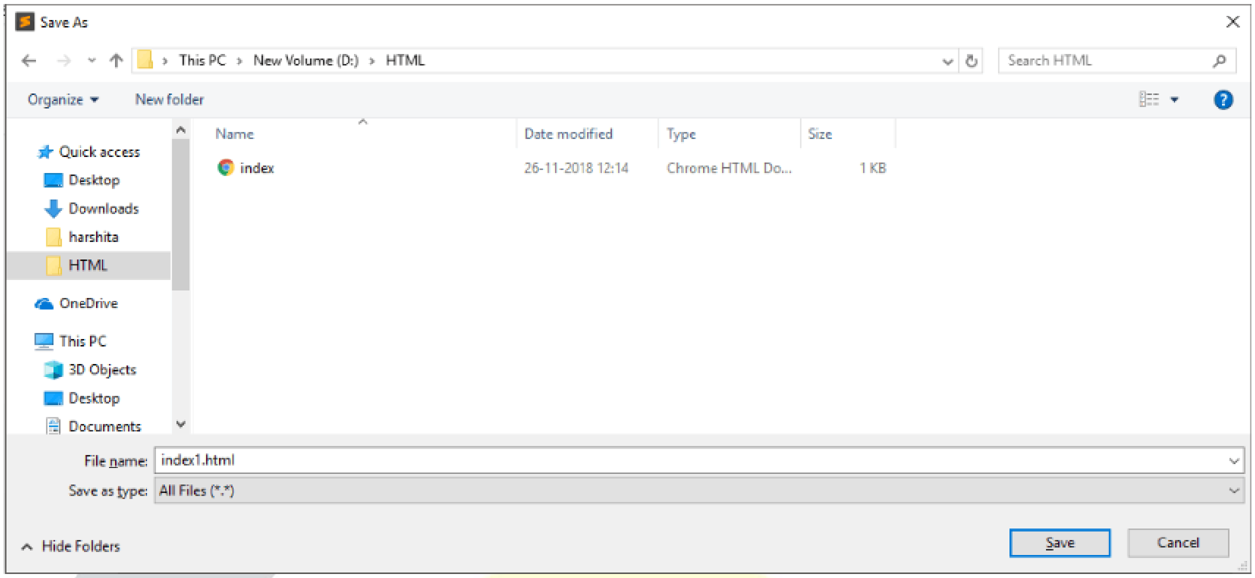
Step 1: Open Sublime Text editor(Windows 8):

To open Sublime Text editor go to Start screen ⇢ type Sublime Text⇢ Open it. To open a new page press CTRL+N.


untitled - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
untitled
1
Line 1, Column 1
Tab Size: 4
Plain Text

Step 2: Save the page before writing any code.

To save your page in Sublime Text press Ctrl+S or go to File option ⇢ save, to save a file use extension .htm or .html. We recommend to save the file first then write the code because after saving the page sublime text editor will give you suggestions to write code.

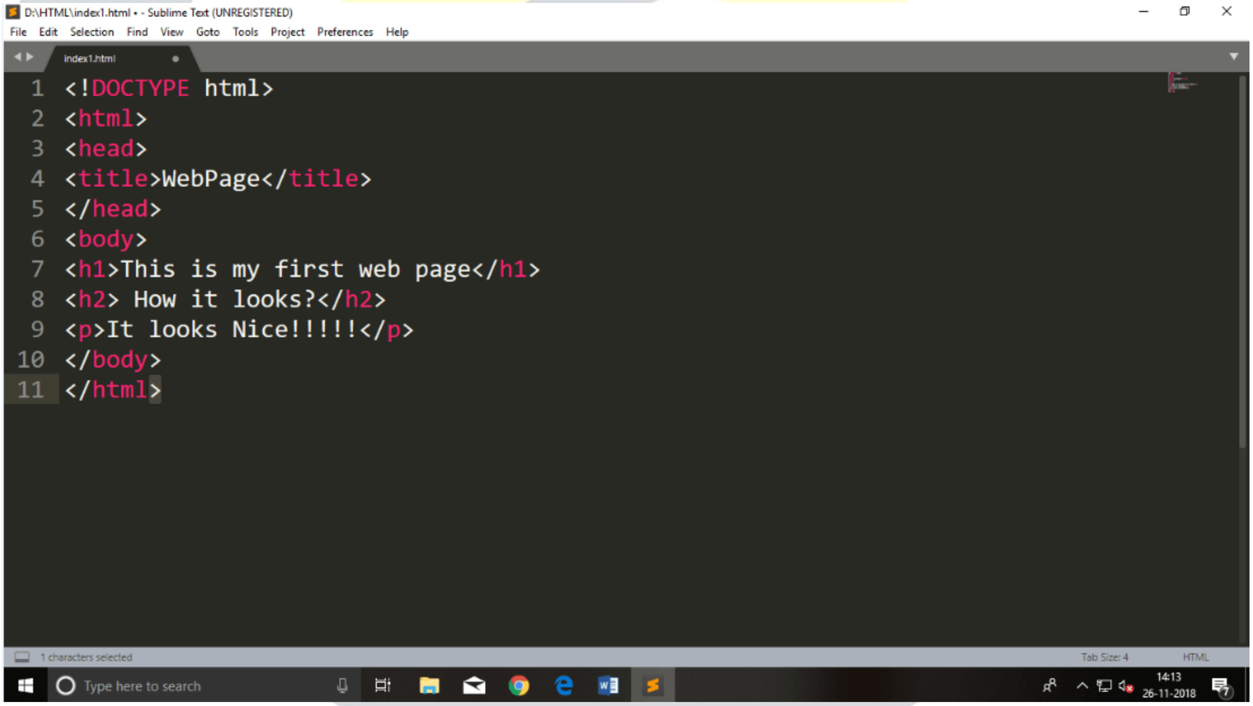


Name	Date modified	Type	Size
index	26-11-2018 12:14	Chrome HTML Do...	1 KB

File name: index1.html

Save as type: All Files (*.*)

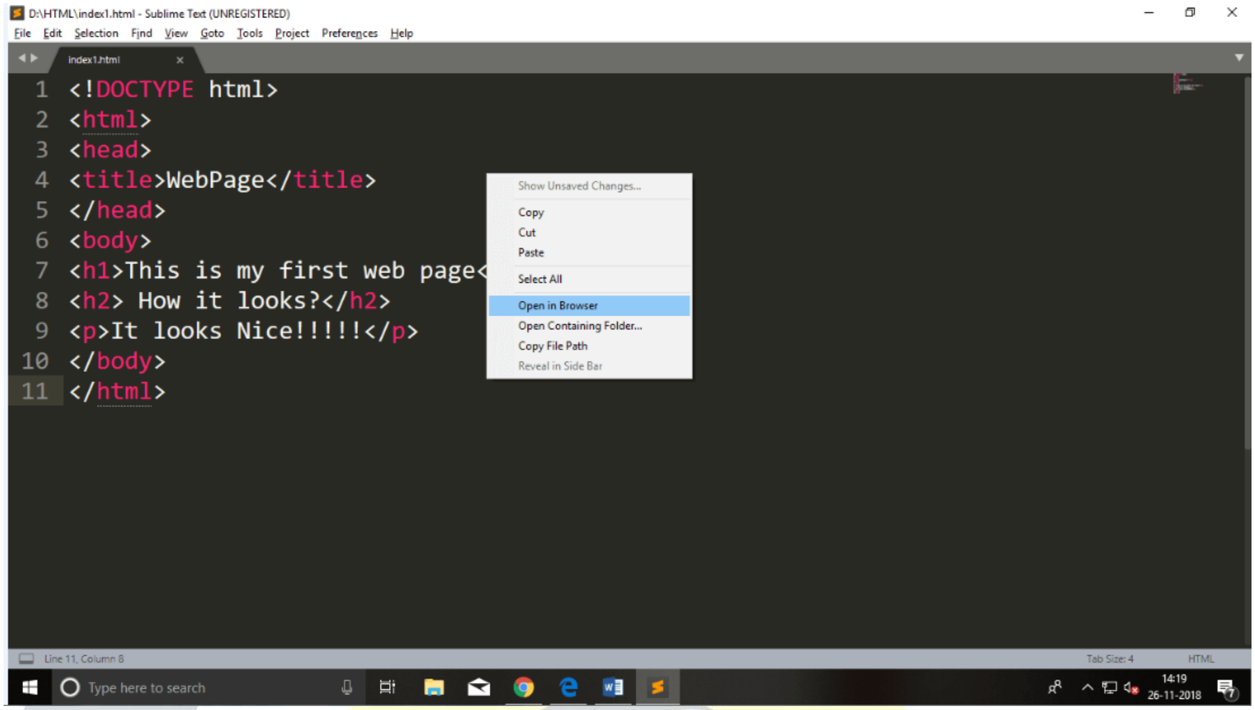
Step 3: Write the code in Sublime Text editor


D:\HTML\index1.html • - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
index1.html
1 characters selected
Tab Size: 4
HTML
Type here to search
14:13
26-11-2018

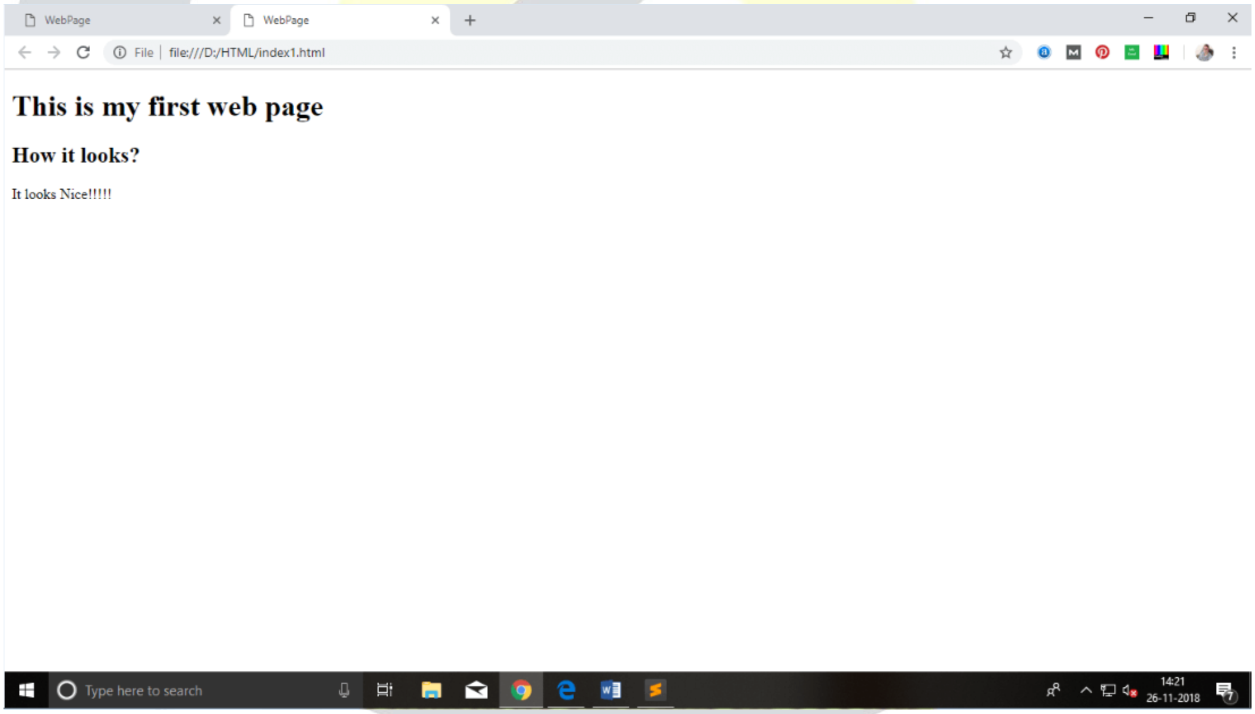
```
<!DOCTYPE html>
<html>
<head>
<title>WebPage</title>
</head>
<body>
<h1>This is my first web page</h1>
<h2> How it looks?</h2>
<p>It looks Nice!!!!!</p>
</body>
</html>
```

Step 4: Open the HTML page in your Browser

To execute or open this page in Web browser just right click by mouse on sublime text page and click on Open in Browser.



```
<!DOCTYPE html>
<html>
<head>
<title>WebPage</title>
</head>
<body>
<h1>This is my first web page<
<h2> How it looks?</h2>
<p>It looks Nice!!!!!</p>
</body>
</html>
```



This is my first web page

How it looks?

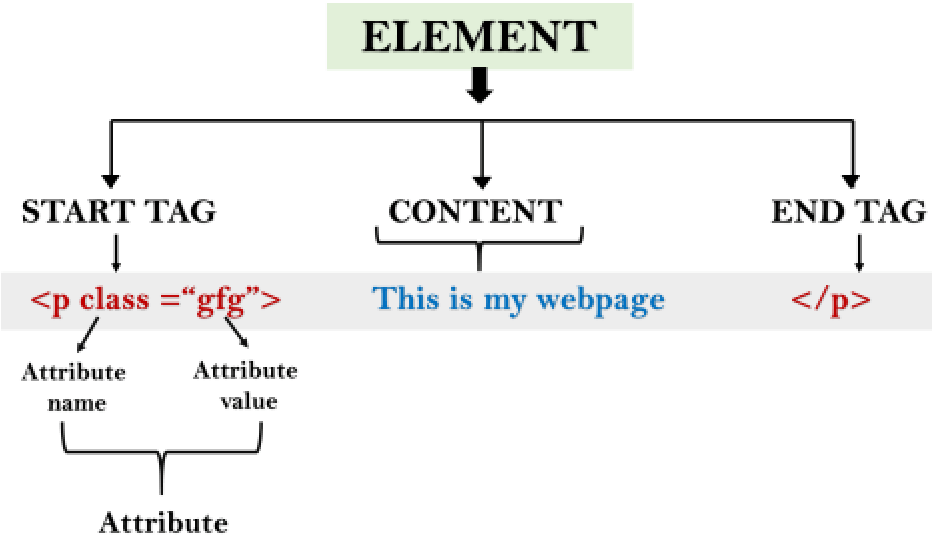
It looks Nice!!!!!

Note: You can execute HTML file in any browser, but there are some tags which are not supported by Some Web browser.

Building blocks of HTML

An HTML document consist of its basic building blocks which are:

- **Tags:** An HTML tag surrounds the content and apply meaning to it. It is written between < and > brackets.
- **Attribute:** An attribute in HTML provides extra information about the element, and it is applied within the start tag. An HTML attribute contains two fields: name & value.
  Syntax
  <tag name attribute_name= " attr_value"> content </ tag name>
- **Elements:** An HTML element is an individual component of an HTML file. In an HTML file, everything written within tags are termed as HTML elements.



Example:

<!DOCTYPE html>

<html>

  <head>

    <title>The basic building blocks of HTML</title>

```
 </head>
  <body>
       <h2>The building blocks</h2>
       <p>This is a paragraph tag</p>
       <p style="color: red">The style is attribute of paragraph tag</p>
       <span>The element contains tag, attribute and content</span>
  </body>
</html>
```

Output:

```
The building blocks
This is a paragraph tag
The style is attribute of paragraph tag
The element contains tag, attribute and content
```

HTML Tags

HTML tags are like keywords which defines that how web browser will format and display the content. With the help of tags, a web browser can distinguish between an HTML content and a simple content. HTML tags contain three main parts: opening tag, content and closing tag. But some HTML tags are unclosed tags.

When a web browser reads an HTML document, browser reads it from top to bottom and left to right. HTML tags are used to create HTML documents and render their properties. Each HTML tags have different properties.

An HTML file must have some essential tags so that web browser can differentiate between a simple text and HTML text. You can use as many tags you want as per your code requirement.

- All HTML tags must enclosed within < > these brackets.
- Every tag in HTML perform different tasks.
- If you have used an open tag <tag>, then you must use a close tag </tag> (except some tags)

Syntax

<tag> content </tag>

HTML Tag Examples

Note: HTML Tags are always written in lowercase letters. The basic HTML tags are given below:

<p> Paragraph Tag </p>

<h2> Heading Tag </h2>

<b> Bold Tag </b>

<i> Italic Tag </i>

<u> Underline Tag</u>

Unclosed HTML Tags

Some HTML tags are not closed, for example br and hr.

<br> Tag: br stands for break line, it breaks the line of the code.

<hr> Tag: hr stands for Horizontal Rule. This tag is used to put a line across the webpage.

HTML Meta Tags

DOCTYPE, title, link, meta and style

HTML Text Tags

<p>, <h1>, <h2>, <h3>, <h4>, <h5>, <h6>, <strong>, <em>, <abbr>, <acronym>, <address>, <bdo>, <blockquote>, <cite>, <q>, <code>, <ins>, <del>, <dfn>, <kbd>, <pre>, <samp>, <var> and <br>

HTML Link Tags

<a> and <base>

HTML Image and Object Tags

<img>, <area>, <map>, <param> and <object>

HTML List Tags

<ul>, <ol>, <li>, <dl>, <dt> and <dd>

HTML Table Tags

table, tr, td, th, tbody, thead, tfoot, col, colgroup and caption

HTML Form Tags

form, input, textarea, select, option, optgroup, button, label, fieldset and legend

HTML Scripting Tags

script and noscript

Note: We will see examples using these tags in later charters.

HTML Tags List

Following is the complete list of HTML tags with the description which are arranged alphabetically.

Note: Here HTML Tags Listrepresents newly added Elements in HTML5.

HTML Tags by Alphabets

Tag name	**Description**
<!-- -->	This tag is used to apply comment in an HTML document.
<!DOCTYPE>	This tag is used to specify the version of HTML

A	
<a>	It is termed as anchor tag and it creates a hyperlink or link.
<abbr>	It defines an abbreviation for a phrase or longer word.
<acronym>	It defines acronym for a word. **(Not supported in HTML5)**
<address>	It defines the author's contact information of the HTML article
<applet>	It defines an embedded Java applet. **(Not supported in HTML5)**
<area>	It defines the area of an image map.
<article>	It defines the self-contained content.

<aside>	It defines content aside from main content. Mainly represented as sidebar.
<audio>	It is used to embed sound content in HTML document.
B	
<b>	It is used to make a text bold.
<base>	This tag defines the base URL for all relative URL within the document.
<basefont>	This tag is used to set default font, size and color for all elements of document. (Not supported in HTML5)
<bdi>	This tag is used to provide isolation for that part of text which may be formatted in different directions from its surrounding text.
<bdo>	It is used to override the current text direction.
<big>	This tag is used to make font size one level larger than its surrounding content. (Not supported in HTML5)
<blockquote>	It is used to define a content which is taken from another source.
<body>	It is used to define the body section of an HTML document.
<br>	It is used to apply single line break.
<button>	It is used to represent a clickable button
C	
<canvas>	It is used to provide a graphics space within a web document.
<caption>	It is used to define a caption for a table.

<center>	It is used to align the content in center. (Not supported in HTML5)
<cite>	It is used to define the title of the work, book, website, etc.
<code>	It is used to display a part of programming code in an HTML document.
<col>	It defines a column within a table which represent common properties of columns and used with the <colgroup> element.
<colgroup>	It is used to define group of columns in a table.
D	
<data>	It is used to link the content with the machine-readable translation.
<datalist>	It is used to provide a predefined list for input option.
<dd>	It is used to provide definition/description of a term in description list.
<del>	It defines a text which has been deleted from the document.
<details>	It defines additional details which user can either view or hide.
<dfn>	It is used to indicate a term which is defined within a sentence/phrase.
<dialog>	It defines a dialog box or other interactive components.
<dir>	It is used as container for directory list of files. (Not supported in HTML5)
<div>	It defines a division or section within HTML document.
<dl>	It is sued to define a description list.

<dt>	It is used to define a term in description list.
E	
<em>	It is used to emphasis the content applied within this element.
<embed>	It is used as embedded container for external file/application/media, etc.
F	
<fieldset>	It is used to group related elements/labels within a web form.
<figcaption>	It is used to add a caption or explanation for the <figure> element.
<figure>	It is used to define the self-contained content, and s mostly refer as single unit.
<font>	It defines the font, size, color, and face for the content. (Not supported in HTML5)
<footer>	It defines the footer section of a webpage.
<form>	It is used to define an HTML form.
<frame>	It defines a particular area of webpage which can contain another HTML file. (Not supported in HTML5)
<frameset>	It defines group of Frames. (Not supported in HTML5)
H	
<h1> to <h6>	It defines headings for an HTML document from level 1 to level 6.
<head>	It defines the head section of an HTML document.

<header>	It defines the header of a section or webpage.
<hr>	It is used to apply thematic break between paragraph-level elements.
<html>	It represents root of an HTML document.
I	
<i>	It is used to represent a text in some different voice.
<iframe>	It defines an inline frame which can embed other content.
<img>	It is used to insert an image within an HTML document.
<input>	It defines an input field within an HTML form.
<ins>	It represent text that has been inserted within an HTML document.
<isindex>	It is used to display search string for current document. (Not supported in HTML5)
K	
<kbd>	It is used to define keyboard input.
L	
<label>	It defines a text label for the input field of form.
<legend>	It defines a caption for content of <fieldset>
<li>	It is used to represent items in list.
<link>	It represents a relationship between current document and an external resource.

M	
<main>	It represents the main content of an HTML document.
<map>	It defines an image map with active areas.
<mark>	It represents a highlighted text.
<marquee>	It is used to insert the scrolling text or an image either horizontally or vertically. (Not supported in HTML5)
<menu>	It is used for creating a menu list of commands.
<meta>	It defines metadata of an HTML document.
<meter>	It defines scalar measurement with known range or fractional value.
N	
<nav>	It represents section of page to represent navigation links.
<noframes>	It provides alternate content to represent in browser which does not support the <frame> elements. (Not supported in HTML5)
<noscript>	It provides an alternative content if a script type is not supported in browser.
O	
<object>	It is used to embed an object in HTML file.
<ol>	It defines an ordered list of items.
<optgroup>	It is used to group the options of a drop-down list.

<option>	It is used to define options or items in a drop-down list.
<output>	It is used as container element which can show result of a calculation.
P	
<p>	It represents a paragraph in an HTML document.
<param>	It defines parameter for an <object> element
<picture>	It defines more than one source element and one image element.
<pre>	It defines preformatted text in an HTML document.
<progress>	It defines the progress of a task within HTML document.
Q	
<q>	It defines short inline quotation.
R	
<rp>	It defines an alternative content if browser does not supports ruby annotations.
<rt>	It defines explanations and pronunciations in ruby annotations.
<ruby>	It is used to represent ruby annotations.
S	
<s>	It render text which is no longer correct or relevant.
<samp>	It is used to represent sample output of a computer program.

<script>	It is used to declare the JavaScript within HTML document.
<section>	It defines a generic section for a document.
<select>	It represents a control which provides a menu of options.
<small>	It is used to make text font one size smaller than document?s base font size.
<source>	It defines multiple media recourses for different media element such as <picture>, <video>, and <audio> element.
<span>	It is used for styling and grouping inline.
<strike>	It is used to render strike through the text. (Not supported in HTML5)
<strong>	It is used to define important text.
<style>	It is used to contain style information for an HTML document.
<sub>	It defines a text which displays as a subscript text.
<summary>	It defines summary which can be used with <details> tag.
<sup>	It defines a text which represent as superscript text.
<svg>	It is used as container of SVG (Scalable Vector Graphics).
T	
<table>	It is used to present data in tabular form or to create a table within HTML document.
<tbody>	It represents the body content of an HTML table and used along with <thead> and <tfoot>.

<td>	It is used to define cells of an HTML table which contains table data
<template>	It is used to contain the client side content which will not display at time of page load and may render later using JavaScript.
<textarea>	It is used to define multiple line input, such as comment, feedback, and review, etc.
<tfoot>	It defines the footer content of an HTML table.
<th>	It defines the head cell of an HTML table.
<thead>	It defines the header of an HTML table. It is used along with <tbody> and <tfoot> tags.
<time>	It is used to define data/time within an HTML document.
<title>	It defines the title or name of an HTML document.
<tr>	It defines the row cells in an HTML table
<track>	It is used to define text tracks for <audio> and <video> elements.
<tt>	It is used to define teletype text. (Not supported in HTML5)
U	
<u>	It is used to render enclosed text with an underline.
<ul>	It defines unordered list of items.
V	
<var>	It defines variable name used in mathematical or programming context.

<video>	It is used to embed a video content with an HTML document
W	
<wbr>	It defines a position within text where break line is possible.

HTML Attributes

- HTML attributes are special words which provide additional information about the elements or attributes are the modifier of the HTML element.
- Each element or tag can have attributes, which defines the behaviour of that element.
- Attributes should always be applied with start tag.
- The Attribute should always be applied with its name and value pair.
- The Attributes name and values are case sensitive, and it is recommended by W3C that it should be written in Lowercase only.
- You can add multiple attributes in one HTML element, but need to give space between two attributes.

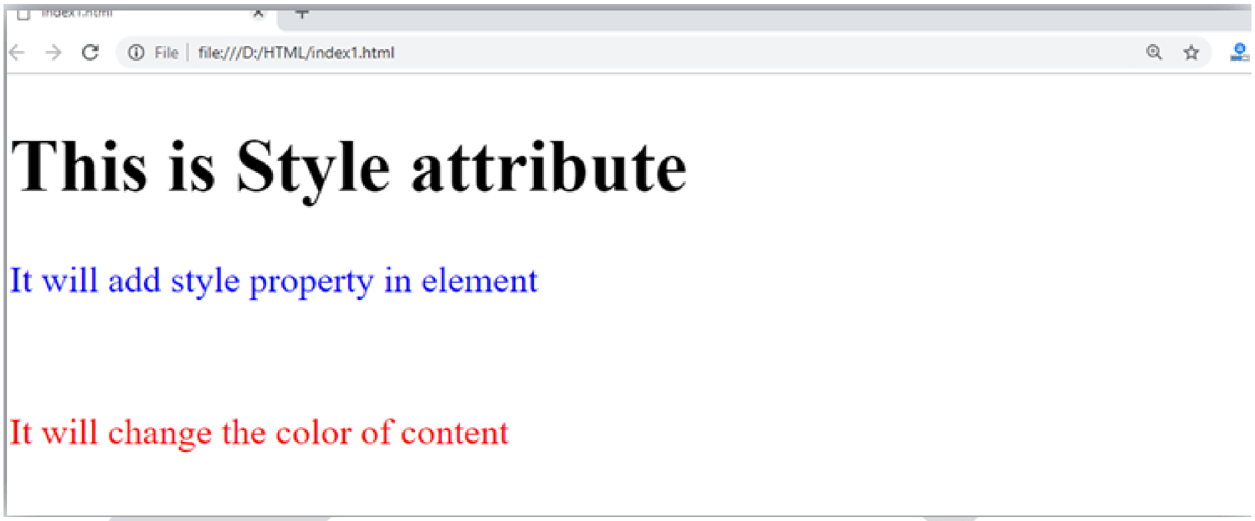
Syntax

```
<element attribute_name="value">content</element>
```

Example

```
 <!DOCTYPE html>
<html>
<head>
</head>
<body>
    <h1> This is Style attribute</h1>
   <p style="height: 50px; color: blue">It will add style property in element</p>
    <p style="color: red">It will change the color of content</p>
</body>
</html>
```

Output:



This is Style attribute

It will add style property in element

It will change the color of content

Explanation of above example:

<p style="height: 50px; color: blue">It will add style property in element</p>

In the above statement, we have used paragraph tags in which we have applied style attribute. This attribute is used for applying CSS property on any HTML element. It provides height to paragraph element of 50px and turns it colour to blue.

<p style="color: red">It will change the color of content</p>

In the above statement we have again used style attribute in paragraph tag, which turns its colour red.

Note: There are some commonly used attributes are given below, and the complete list and explanation of all attributes are given in HTML attributes List.

The title attribute in HTML

Description: The title attribute is used as text tooltip in most of the browsers. It display its text when user move the cursor over a link or any text. You can use it with any text or link to show the description about that link or text. In our example, we are taking this with paragraph tag and heading tag.

Example

With <h1> tag:

```
<h1 title="This is heading tag">Example of title attribute</h1>
```

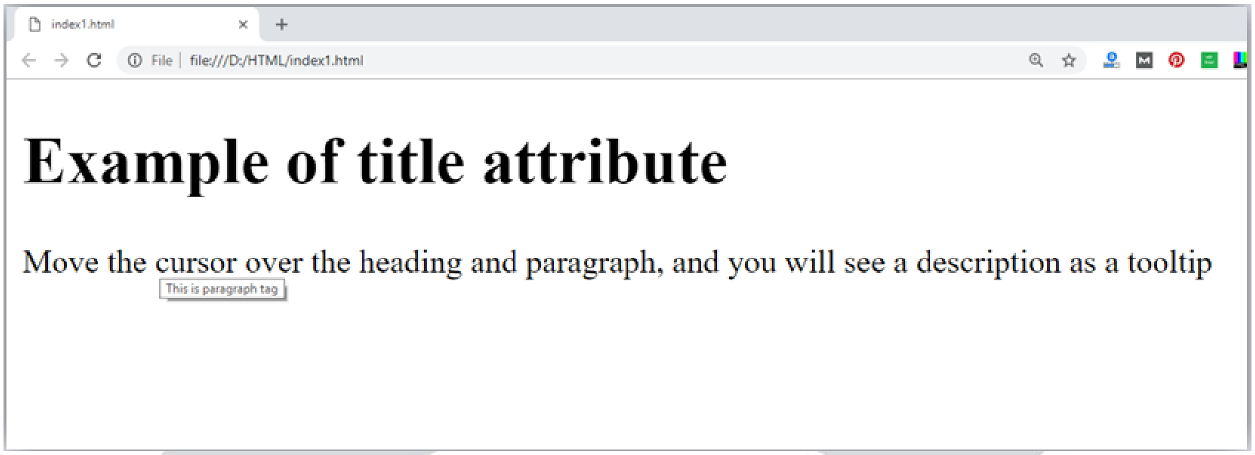
With <p> tag:

```
<p title="This is paragraph tag">Move the cursor over the heading and paragraph, and you will see a description as a tooltip</p>
```

Code:

```
<!DOCTYPE html>
<html>
  <head>
 </head>
<body>
  <h1 title="This is heading tag">Example of title attribute</h1>
  <p title="This is paragraph tag">Move the cursor over the heading and paragraph, and you will see a description as a tooltip</p>
</body>
</html>
```

Output:


index1.html

File | file:///D:/HTML/index1.html

Example of title attribute

Move the cursor over the heading and paragraph, and you will see a description as a tooltip

This is paragraph tag

The href attribute in HTML

Description: The href attribute is the main attribute of <a> anchor tag. This attribute gives the link address which is specified in that link. The href attribute provides the hyperlink, and if it is blank, then it will remain in same page.

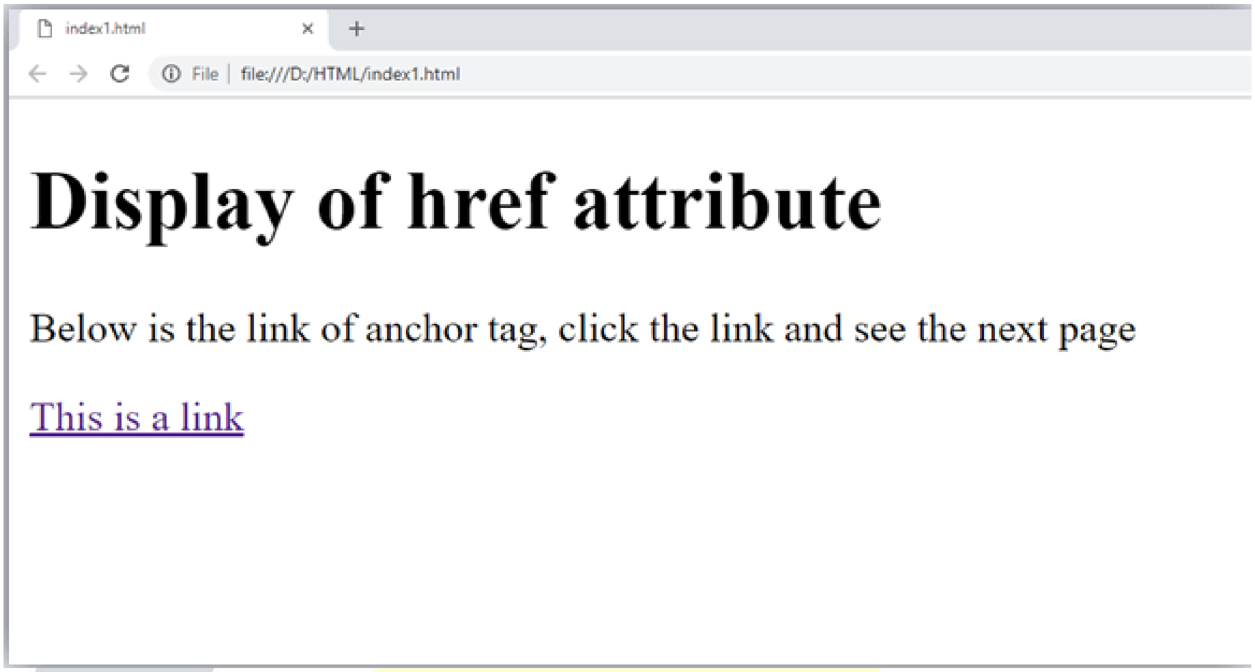
Example

With link address:

<a href="https://www.javatpoint.com/html-anchor">This is a link</a>

Without link address:

<a href="">This is a link</a>



Display of href attribute

Below is the link of anchor tag, click the link and see the next page

This is a link

The src Attribute

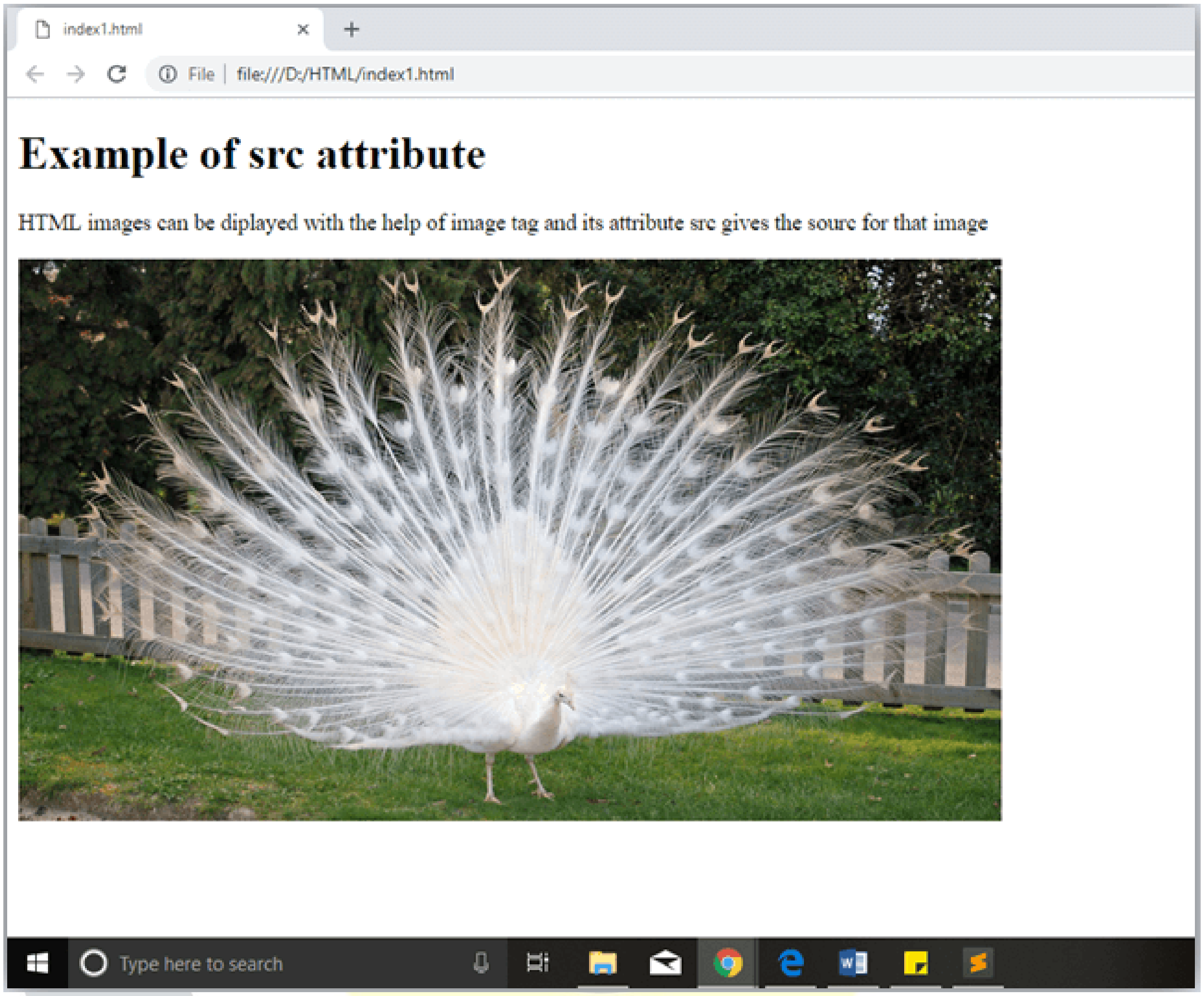
The src attribute is one of the important and required attribute of <img> element. It is source for the image which is required to display on browser. This attribute can contain image in same directory or another directory. The image name or source should be correct else browser will not display the image.

Example

<img src="whitepeacock.jpg" height="400" width="600">

Note: The above example also have height and width attribute, which define the height and width of image on web page.

Output:



file:///D:/HTML/index1.html

Example of src attribute

HTML images can be diplayed with the help of image tag and its attribute src gives the sourc for that image

Quotes: single quotes or double quotes?

In this chapter you have seen that, we have used attribute with double quotes, but some people might use single quotes in HTML. So use of single quotes with HTML attribute, is also allowed. The following both statements are absolutely fine.

```
<a href="https://www.javatpoint.com">A link to HTML.</a>
```

```
<a href='https://www.javatpoint.com'>A link to HTML.</a>
```

IN HTML5, you can also omit use of quotes around attribute values.

```
<a href=https://www.javatpoint.com>A link to HTML.</a>
```

HTML Elements

An HTML file is made of elements. These elements are responsible for creating web pages and define content in that webpage. An element in HTML usually consist of a start tag <tag name>, close tag </tag name> and content inserted between them. Technically, an element is a collection of start tag, attributes, end tag, content between them.

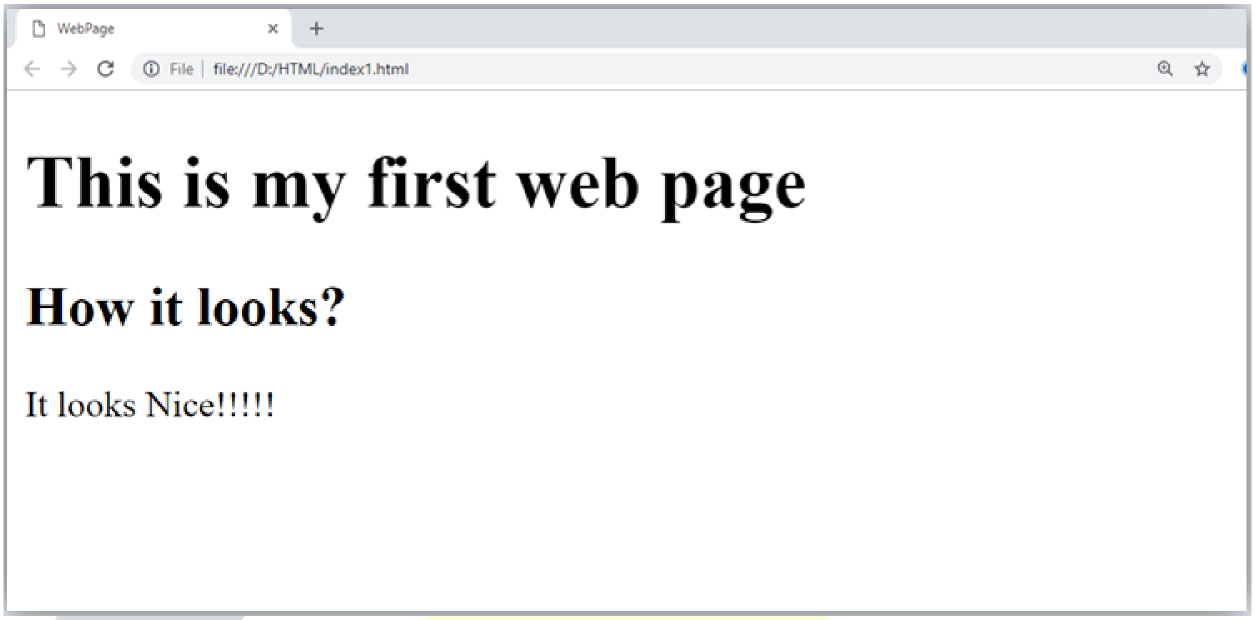
Note: Some elements does not have end tag and content, these elements are termed as empty elements or self-closing element or void elements.

Such as:

```
<p> Hello world!!! </p>
```

Example

```
<!DOCTYPE html>
<html>
<head>
   <title>WebPage</title>
</head>
<body>
  <h1>This is my first web page</h1>
   <h2> How it looks?</h2>
    <p>It looks Nice!!!!!</p>
</body>
</html>
```



This is my first web page

How it looks?

It looks Nice!!!!!

All the content written between body elements are visible on web page.

Void element: All the elements in HTML do not require to have start tag and end tag, some elements does not have content and end tag such elements are known as Void elements or empty elements. These elements are also called as unpaired tag.

Some Void elements are <br> (represents a line break) , <hr>(represents a horizontal line), etc.

Nested HTML Elements: HTML can be nested, which means an element can contain another element.

Block-level and Inline HTML elements

For the default display and styling purpose in HTML, all the elements are divided into two categories:

- Block-level element
- Inline element

Block-level element:

- These are the elements, which structure main part of web page, by dividing a page into coherent blocks.

- A block-level element always start with new line and takes the full width of web page, from left to right.
- These elements can contain block-level as well as inline elements.

Following are the block-level elements in HTML.

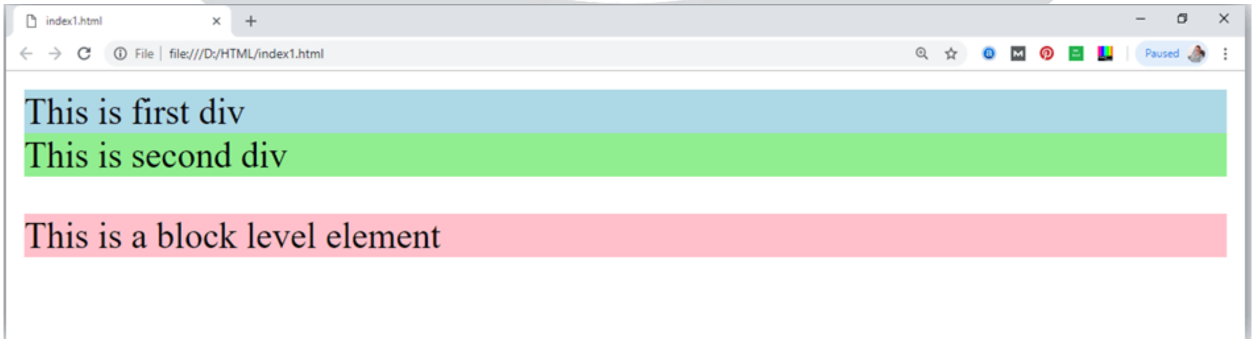
<address>, <article>, <aside>, <blockquote>, <canvas>, <dd>, <div>, <dl>, <dt>, <fieldset>, <figcaption>, <figure>, <footer>, <form>, <h1>-<h6>, <header>, <hr>, <li>, <main>, <nav>, <noscript>, <ol>, <output>, <p>, <pre>, <section>, <table>, <tfoot>, <ul> and <video>.

Note: All these elements are described in later chapters.

Example:

```
<!DOCTYPE html>

<html>

          <head>

   </head>

<body>

   <div style="background-color: lightblue">This is first div</div>

   <div style="background-color: lightgreen">This is second div</div>

   <p style="background-color: pink">This is a block level element</p>

</body>

</html>
```

Output:



```
This is first div
This is second div

This is a block level element
```

In the above example we have used tag, which defines a section in a web page, and takes full width of page.

We have used style attribute which is used to styling the HTML content, and the background color are showing that it's a block level element.

Inline elements:

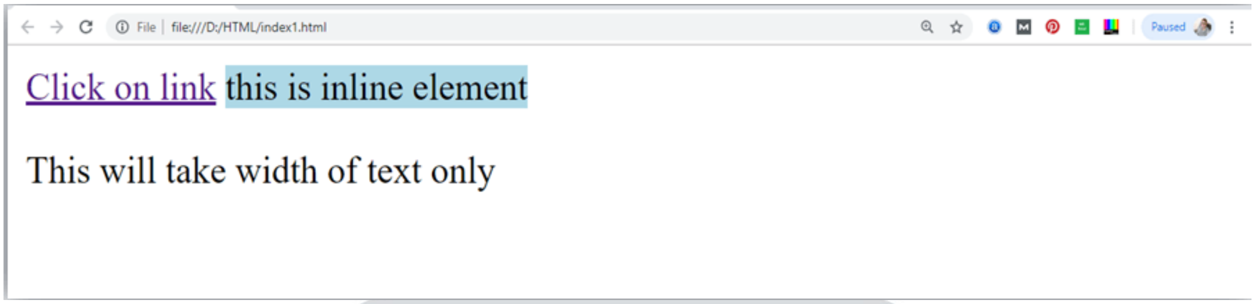
- Inline elements are those elements, which differentiate the part of a given text and provide it a particular function.
- These elements does not start with new line and take width as per requirement.
- The Inline elements are mostly used with other elements.

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

Example:

```
<!DOCTYPE html>

<html>

   <head>

   </head>

<body>

   <a href="https://www.javatpoint.com/html-tutorial">Click on link</a>

   <span style="background-color: lightblue">this is inline element</span>

   <p>This will take width of text only</p>

 </body>

</html>
```

Output:



Click on link this is inline element

This will take width of text only

Following is the list of the some main elements used in HTML:

Start tag	Content	End tag	Description
<h1> ...... <h6>	These are headings of HTML	</h1>??..</h6>	These elements are used to provide the headings of page.
<p>	This is the paragraph	</p>	This element is used to display a content in form of paragraph.
<div>	This is div section	</div>	This element is used to provide a section in web page.
<br>			This element is used to provide a line break. (void element)
<hr>			This element is used to provide a horizontal line. (void element)

HTML Formatting

HTML Formatting is a process of formatting text for better look and feel. HTML provides us ability to format text without using CSS. There are many formatting tags in HTML. These tags are used to make text bold, italicized, or underlined. There are almost 14 options available that how text appears in HTML and XHTML.

In HTML the formatting tags are divided into two categories:

- **Physical tag:** These tags are used to provide the visual appearance to the text.
- **Logical tag:** These tags are used to add some logical or semantic value to the text.

NOTE: There are some physical and logical tags which may give same visual appearance, but they will be different in semantics.

Here, we are going to learn 14 HTML formatting tags. Following is the list of HTML formatting text.

Element name	Description
<b>	This is a physical tag, which is used to bold the text written between it.
<strong>	This is a logical tag, which tells the browser that the text is important.
<i>	This is a physical tag which is used to make text italic.
<em>	This is a logical tag which is used to display content in italic.
<mark>	This tag is used to highlight text.
<u>	This tag is used to underline text written between it.
<tt>	This tag is used to appear a text in teletype. (not supported in HTML5)

<strike>	This tag is used to draw a strikethrough on a section of text. (Not supported in HTML5)
<sup>	It displays the content slightly above the normal line.
<sub>	It displays the content slightly below the normal line.
<del>	This tag is used to display the deleted content.
<ins>	This tag displays the content which is added
<big>	This tag is used to increase the font size by one conventional unit.
<small>	This tag is used to decrease the font size by one unit from base font size.

1) Bold Text

HTML<b> and <strong> formatting elements

The HTML <b> element is a physical tag which display text in bold font, without any logical importance. If you write anything within <b>............</b> element, is shown in bold letters.

See this example:

```
<p> <b>Write Your First Paragraph in bold text.</b></p>
```

Output:

Write Your First Paragraph in bold text.

The HTML <strong> tag is a logical tag, which displays the content in bold font and informs the browser about its logical importance. If you write anything between <strong>???????. </strong>, is shown important text.

See this example:

```
<p><strong>This is an important content</strong>, and this is normal content</p>
```

Output:

This is an important content, and this is normal content

Example

```
<!DOCTYPE html>
<html>
<head>
   <title>formatting elements</title>
</head>
<body>
<h1>Explanation of formatting element</h1>
<p><strong>This is an important content</strong>, and this is normal content</p>
</body>
</html>
```

2) Italic Text

HTML <i> and <em> formatting elements

The HTML <i> element is physical element, which display the enclosed content in italic font, without any added importance. If you write anything within <i>............</i> element, is shown in italic letters.

See this example:

```
<p> <i>Write Your First Paragraph in italic text.</i></p>
```

Output:

Write Your First Paragraph in italic text.

The HTML <em> tag is a logical element, which will display the enclosed content in italic font, with added semantics importance.

See this example:

```
<p><em>This is an important content</em>, which displayed in italic font.</p>
```

Output:

This is an important content, which displayed in italic font.

```
<!DOCTYPE html>
<html>
<head>
   <title>formatting elements</title>
</head>
<body>
<h1>Explanation of italic formatting element</h1>
<p><em>This is an important content</em>, which displayed in italic font.</p>
</body>
</html>
```

3) HTML Marked formatting

If you want to mark or highlight a text, you should write the content within <mark>.........</mark>.

See this example:

```
<h2>  I want to put a <mark> Mark</mark> on your face</h2>
```

Output:

I want to put a Mark on your face

4) Underlined Text

If you write anything within <u>.........</u> element, is shown in underlined text.

See this example:

<p> <u>Write Your First Paragraph in underlined text.</u></p>

Output:

Write Your First Paragraph in underlined text.

5) Strike Text

Anything written within <strike>.......................</strike> element is displayed with strikethrough. It is a thin line which cross the statement.

See this example:

<p> <strike>Write Your First Paragraph with strikethrough</strike>.</p>

Output:

~~Write Your First Paragraph with strikethrough.~~

6) Monospaced Font

If you want that each letter has the same width then you should write the content within <tt>.............</tt> element.

Note: We know that most of the fonts are known as variable-width fonts because different letters have different width. (for example: 'w' is wider than 'i'). Monospaced Font provides similar space among every letter.

See this example:

<p>Hello <tt>Write Your First Paragraph in monospaced font.</tt></p>

Output:

Hello Write Your First Paragraph in monospaced font.

7) Superscript Text

If you put the content within <sup>..............</sup> element, is shown in superscript; means it is displayed half a character's height above the other characters.

See this example:

```
<p>Hello <sup>Write Your First Paragraph in superscript.</sup></p>
```

Output:

Hello $^{\text{Write Your First Paragraph in superscript.}}$

8) Subscript Text

If you put the content within <sub>..............</sub> element, is shown in subscript ; means it is displayed half a character's height below the other characters.

See this example:

```
<p>Hello <sub>Write Your First Paragraph in subscript.</sub></p>
```

Output:

Hello $_{\text{Write Your First Paragraph in subscript}}$.

9) Deleted Text

Anything that puts within <del>..........</del> is displayed as deleted text.

See this example:

```
<p>Hello <del>Delete your first paragraph.</del></p>
```

Output:

Hello

10) Inserted Text

Anything that puts within <ins>..........</ins> is displayed as inserted text.

See this example:

<p> <del>Delete your first paragraph.</del><ins>Write another paragraph.</ins></p>

Output:

~~Delete your first paragraph~~. Write another paragraph.

11) Larger Text

If you want to put your font size larger than the rest of the text then put the content within <big>.........</big>. It increase one font size larger than the previous one.

See this example:

<p>Hello <big>Write the paragraph in larger font.</big></p>

Output:

Hello Write the paragraph in larger font.

12) Smaller Text

If you want to put your font size smaller than the rest of the text then put the content within <small>.........</small>tag. It reduces one font size than the previous one.

See this example:

<p>Hello <small>Write the paragraph in smaller font.</small></p>

Output:

```
Hello Write the paragraph in smaller font.
```

HTML Heading

A HTML heading or HTML h tag can be defined as a title or a subtitle which you want to display on the webpage. When you place the text within the heading tags <h1>.........</h1>, it is displayed on the browser in the bold format and size of the text depends on the number of heading.

There are six different HTML headings which are defined with the <h1> to <h6> tags, from highest level h1 (main heading) to the least level h6 (least important heading).

h1 is the largest heading tag and h6 is the smallest one. So h1 is used for most important heading and h6 is used for least important.

Headings in HTML helps the search engine to understand and index the structure of web page.

Note: The main keyword of the whole content of a webpage should be display by h1 heading tag.

See this example:

```
<h1>Heading no. 1</h1>

<h2>Heading no. 2</h2>

<h3>Heading no. 3</h3>

<h4>Heading no. 4</h4>

<h5>Heading no. 5</h5>

<h6>Heading no. 6</h6>
```

Output:

Heading no. 1

Heading no. 2

Heading no. 3

Heading no. 4

Heading no. 5

Heading no. 6

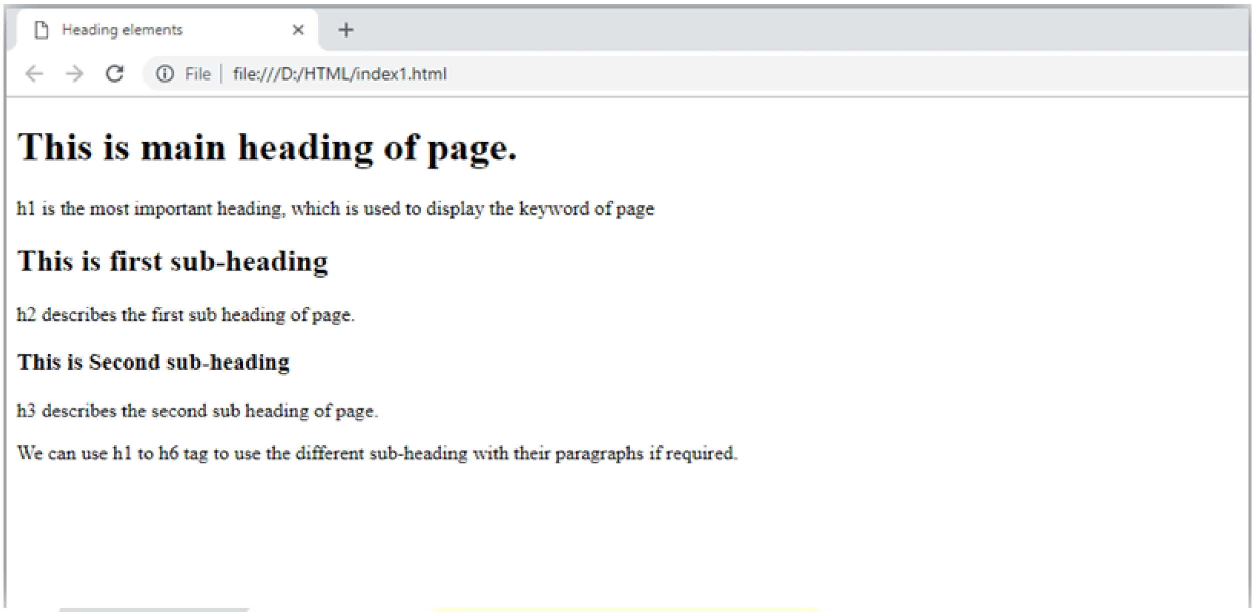
Heading elements (h1....h6) should be used for headings only. They should not be used just to make text bold or big.

HTML headings can also be used with nested elements. Following are different codes to display the way to use heading elements.

Example:

```
<!DOCTYPE html>
<html>
 <head>
    <title>Heading elements</title>
 </head>
 <body>
    <h1>This is main heading of page. </h1>
     <p>h1 is the most important heading, which is used to display the keyword of page </p>
    <h2>This is first sub-heading</h2>
     <p>h2 describes the first sub heading of page. </p>
    <h3>This is Second sub-heading</h3>
     <p>h3 describes the second sub heading of page.</p>
     <p>We can use h1 to h6 tag to use the different sub-heading with their paragraphs if
                required.
            </p>
   </body>
</html>
```

Output:



This is main heading of page.

h1 is the most important heading, which is used to display the keyword of page

This is first sub-heading

h2 describes the first sub heading of page.

This is Second sub-heading

h3 describes the second sub heading of page.

We can use h1 to h6 tag to use the different sub-heading with their paragraphs if required.

HTML Paragraph

HTML paragraph or HTML p tag is used to define a paragraph in a webpage. Let's take a simple example to see how it work. It is a notable point that a browser itself add an empty line before and after a paragraph. An HTML <p> tag indicates starting of new paragraph.

Note: If we are using various <p> tags in one HTML file then browser automatically adds a single blank line between the two paragraphs.

See this example:

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

<p>This is second paragraph.</p>

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

Output:

```
This is first paragraph.

This is second paragraph.

This is third paragraph.
```

Space inside HTML Paragraph

If you put a lot of spaces inside the HTML p tag, browser removes extra spaces and extra line while displaying the page. The browser counts number of spaces and lines as a single one.

```
<p>

I am

going to provide

you a tutorial on HTML

and hope that it will

be very beneficial for you.

</p>
```

```
<p>
Look, I put here a lot
of spaces                    but           I know, Browser will ignore it.
</p>
<p>
You cannot determine the display of HTML</p>
<p>because resized windows may create different result.
</p>
```

Output:

```
I am going to provide you a tutorial on HTML and hope that it will be very beneficial for you.
Look, I put here a lot of spaces but I know, Browser will ignore it.
You cannot determine the display of HTML
because resized windows may create different result.
```

As you can see, all the extra lines and unnecessary spaces are removed by the browser.

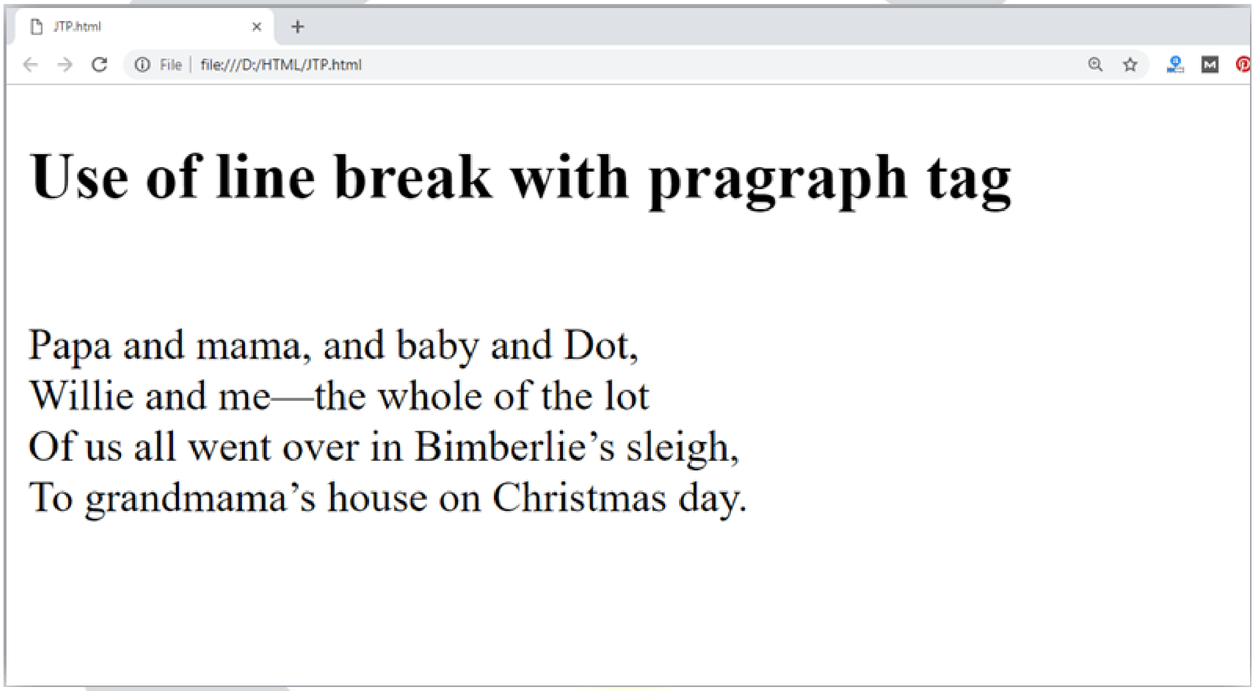
How to Use <br> and <hr> tag with paragraph?

An HTML <br> tag is used for line break and it can be used with paragraph elements. Following is the example to show how to use <br> with <p> element.

Example:

```
<!DOCTYPE html>
<html>
     <head>
    </head>
  <body>
      <h2> Use of line break with pragraph tag</h2>
          <p><br>Papa and mama, and baby and Dot,
     <br>Willie and me?the whole of the lot
           <br>Of us all went over in Bimberlie's sleigh,
             <br>To grandmama's house on Christmas day.
        </p>
   </body>
</html>
```

Output:



Use of line break with pragraph tag

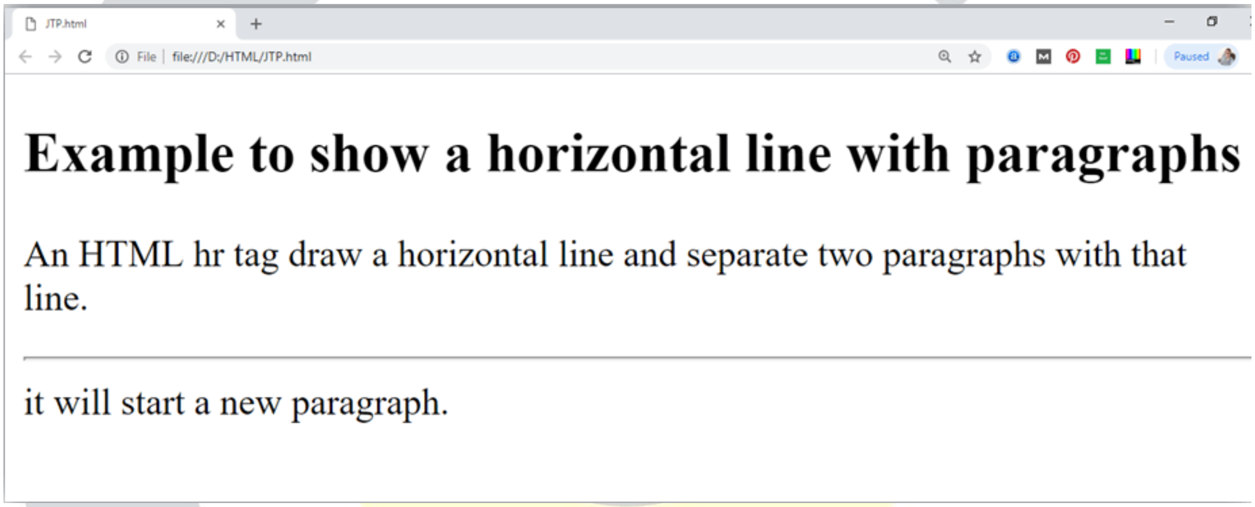
Papa and mama, and baby and Dot,
Willie and me—the whole of the lot
Of us all went over in Bimberlie’s sleigh,
To grandmama’s house on Christmas day.

An HTML <hr> tag is used to apply a horizontal line between two statements or two paragraphs. Following is the example which is showing use of <hr> tag with paragraph.

Example:

```
<!DOCTYPE html>
<html>
 <head>
    </head>
 <body>
   <h2> Example to show a horizontal line with paragraphs</h2>
     <p> An HTML hr tag draw a horizontal line and separate two paragraphs with that line.<hr>
it will start a new paragraph.
     </p>
  </body>
</html>
```

Output:



Example to show a horizontal line with paragraphs

An HTML hr tag draw a horizontal line and separate two paragraphs with that line.

---

it will start a new paragraph.

HTML Phrase tag

The HTML phrase tags are special purpose tags, which defines the structural meaning of a block of text or semantics of text. Following is the list of phrase tags, some of which we have already discussed in HTML formatting.

- Abbreviation tag : <abbr>
- Acronym tag: <acronym> (not supported in HTML5)
- Marked tag: <mark>
- Strong tag: <strong>
- Emphasized tag : <em>
- Definition tag: <dfn>
- Quoting tag: <blockquote>
- Short quote tag : <q>
- Code tag: <code>
- Keyboard tag: <kbd>
- Address tag: <address>

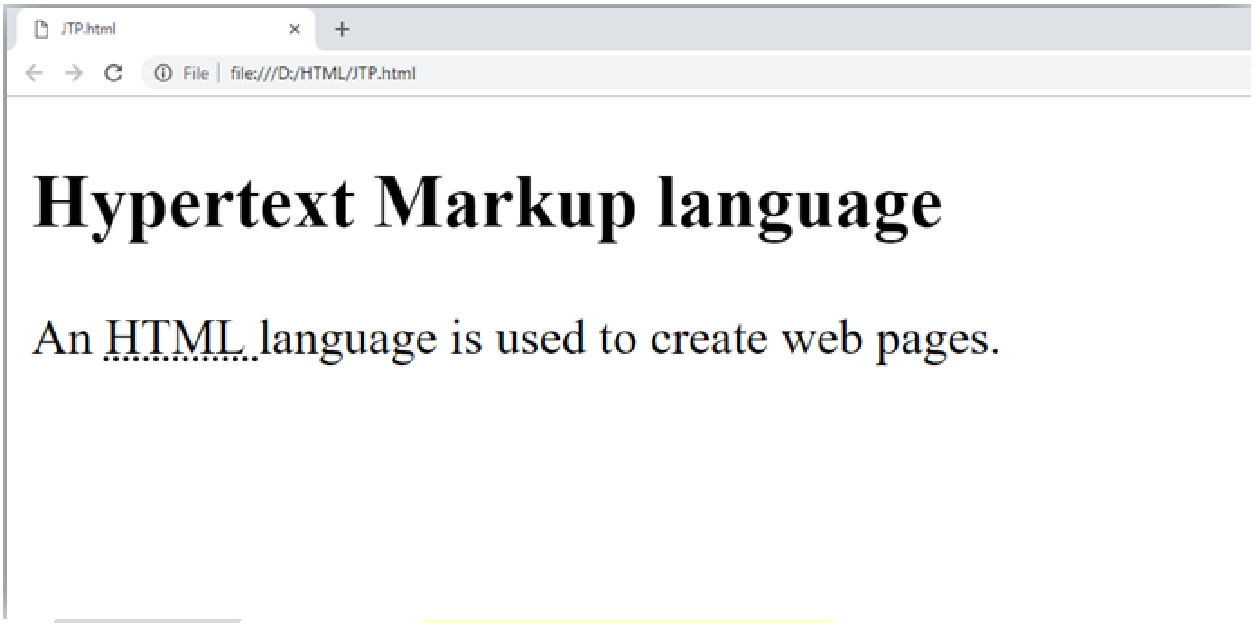
1. Text Abbreviation tag

This tag is used to abbreviate a text. To abbreviate a text, write text between <abbr> and </abbr> tag.

Example

<p>An <abbr title = "Hypertext Markup language">HTML </abbr>language is used to create web pages. </p>

Output:



Hypertext Markup language

An HTML language is used to create web pages.

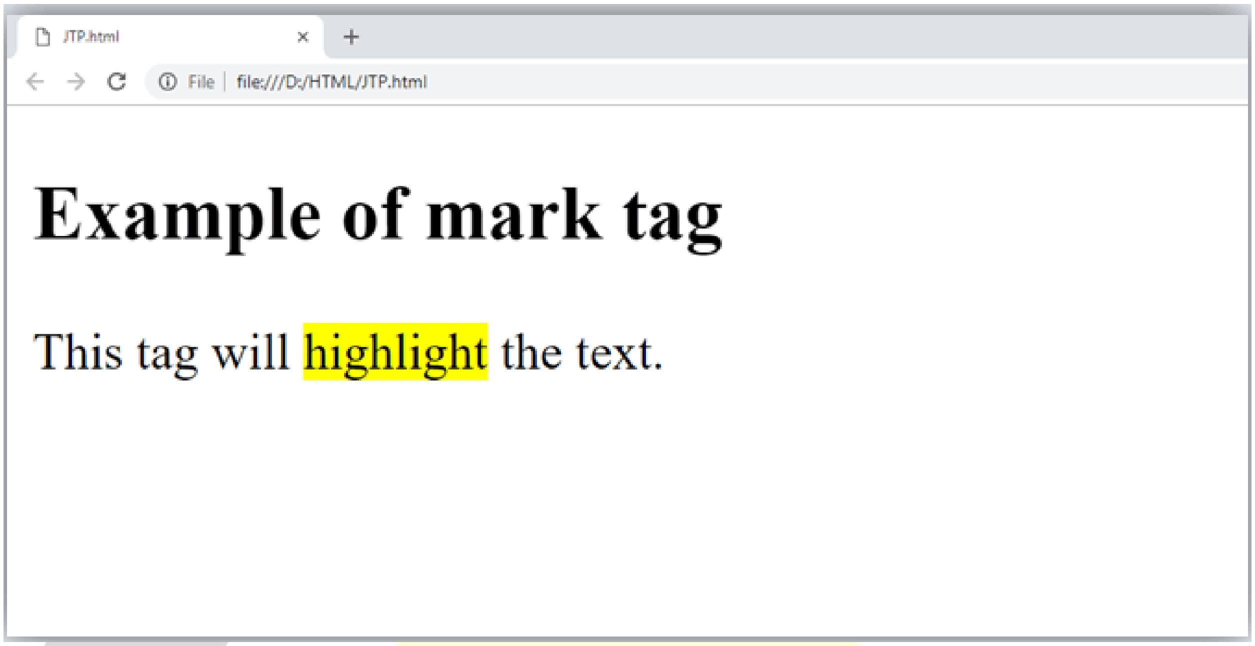
2. Marked tag:

The content written between <mark> and </mark> tag will show as yellow mark on browser. This tag is used to highlight a particular text.

Example

<p>This tag will <mark>highlight</mark> the text.</p>

Output:



Example of mark tag

This tag will highlight the text.

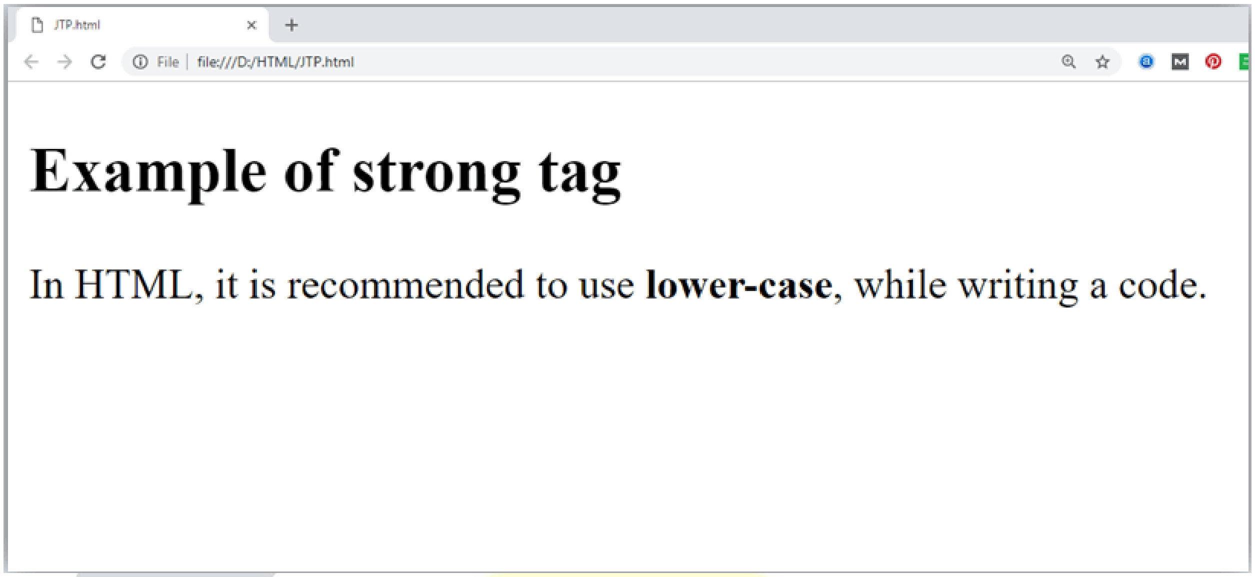
3. Strong text:

This tag is used to display the important text of the content. The text written between <strong> and </strong> will be displayed as important text.

Example

<p>In HTML it is recommended to use <strong>lower-case</strong>, while writing a code. </p>

Output:



Example of strong tag

In HTML, it is recommended to use **lower-case**, while writing a code.

4. Emphasized text

This tag is used to emphasize the text, and displayed the text in italic form. The text written between <em> and </em> tag will italicized the text.

Example

<p>HTML is an <em>easy </em>to learn language.</p>

Output:

HTML Phrase tag

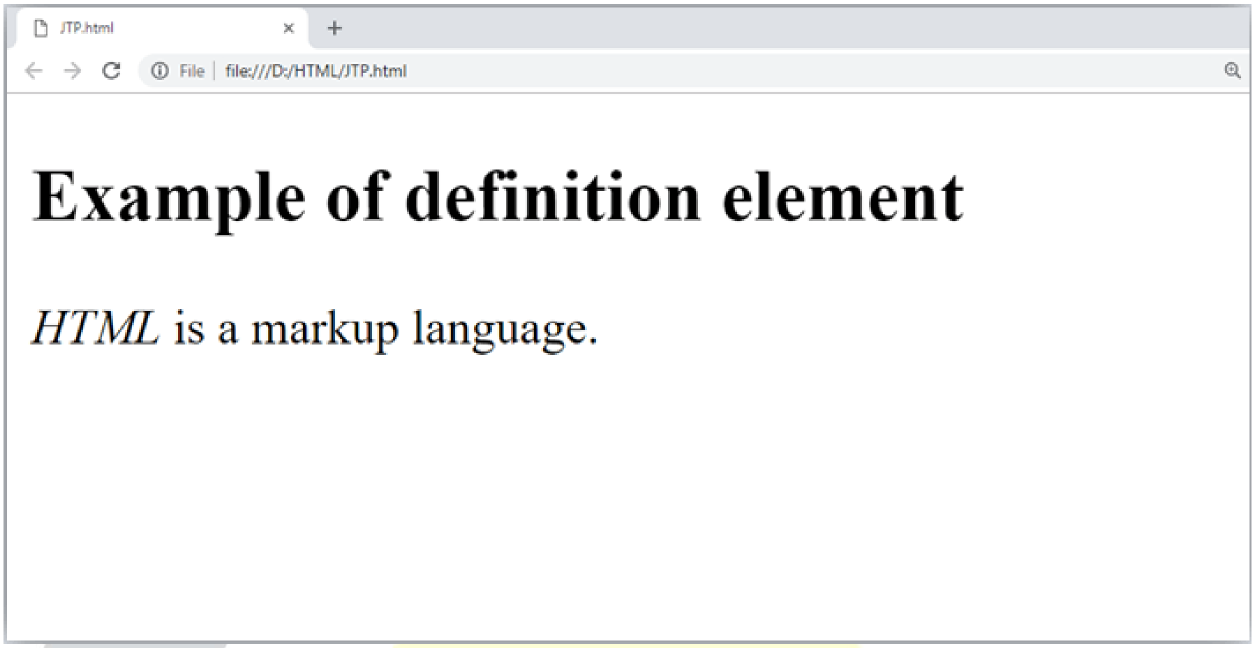
5. Definition tag:

When you use the <dfn> and </dfn> tags, it allow to specify the keyword of the content. Following is the example to show how to definition element.

Example

<p><dfn>HTML </dfn> is a markup language. </p>

Output:



Example of definition element

HTML is a markup language.

6. Quoting text:

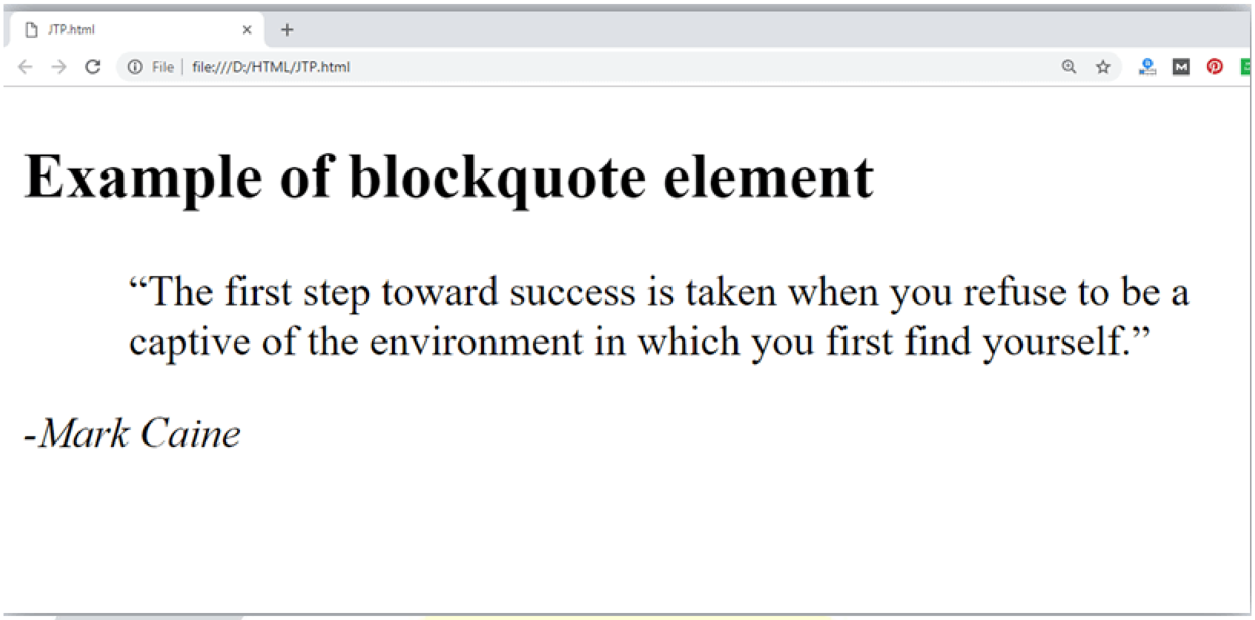
The HTML <blockquote> element shows that the enclosed content is quoted from another source. The Source URL can be given using the cite attribute, and text representation of source can display using <cite> ..... </cite>element.

Example

<blockquote cite="https://www.keepinspiring.me/famous-quotes/"><p>?The first step toward success is taken when you refuse to be a captive of the environment in which you first find yourself.?</p></blockquote>

<cite>-Mark Caine</cite>

Output:



Example of blockquote element

> “The first step toward success is taken when you refuse to be a captive of the environment in which you first find yourself.”

-*Mark Caine*

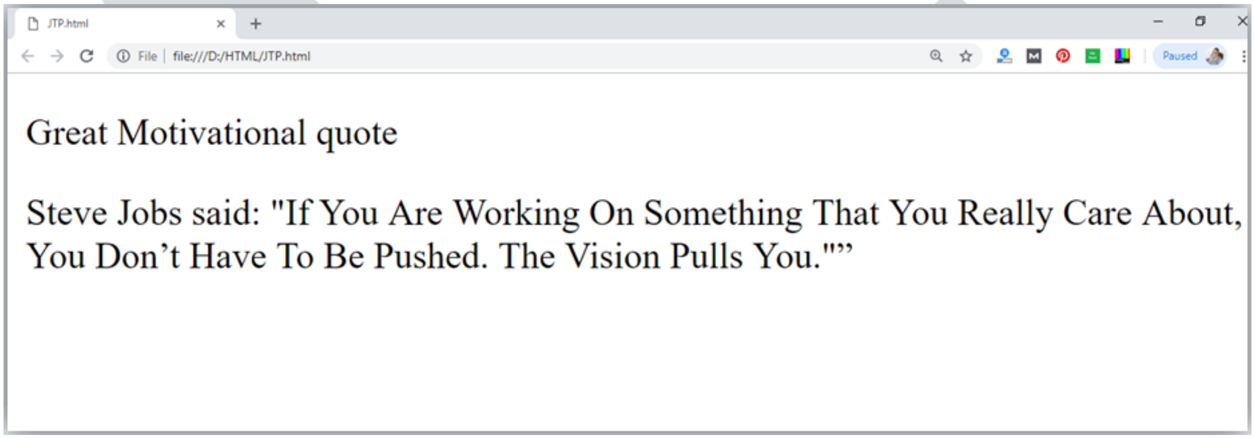
7. Short Quotations:

An HTML <q> .... </q> element defines a short quotation. If you will put any content between <q> .... </q>, then it will enclose the text in double quotes.

Example:

<p>Steve Jobs said: <q>If You Are Working On Something That You Really Care About, You Don?t Have To Be Pushed. The Vision Pulls You.</q>?</p>

Output:



Great Motivational quote

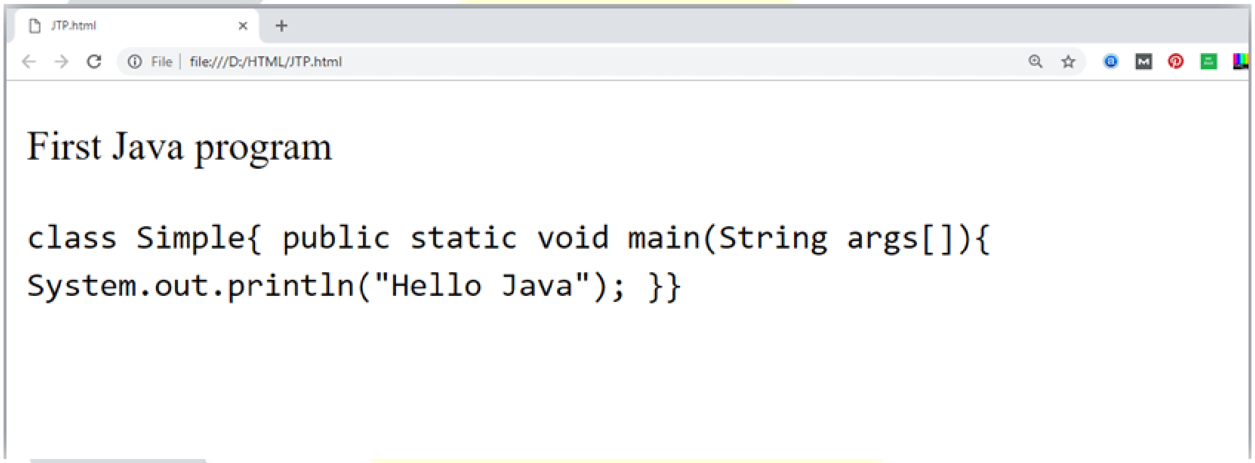
Steve Jobs said: "If You Are Working On Something That You Really Care About, You Don’t Have To Be Pushed. The Vision Pulls You."”

8. Code tags

The HTML <code> </code> element is used to display the part of computer code. It will display the content in monospaced font.

```
<p>First Java program</p>
      <p><code>class Simple{ public static void main(String args[]){
       System.out.println("Hello Java"); }} </code>
     </p>
```

Output:



First Java program

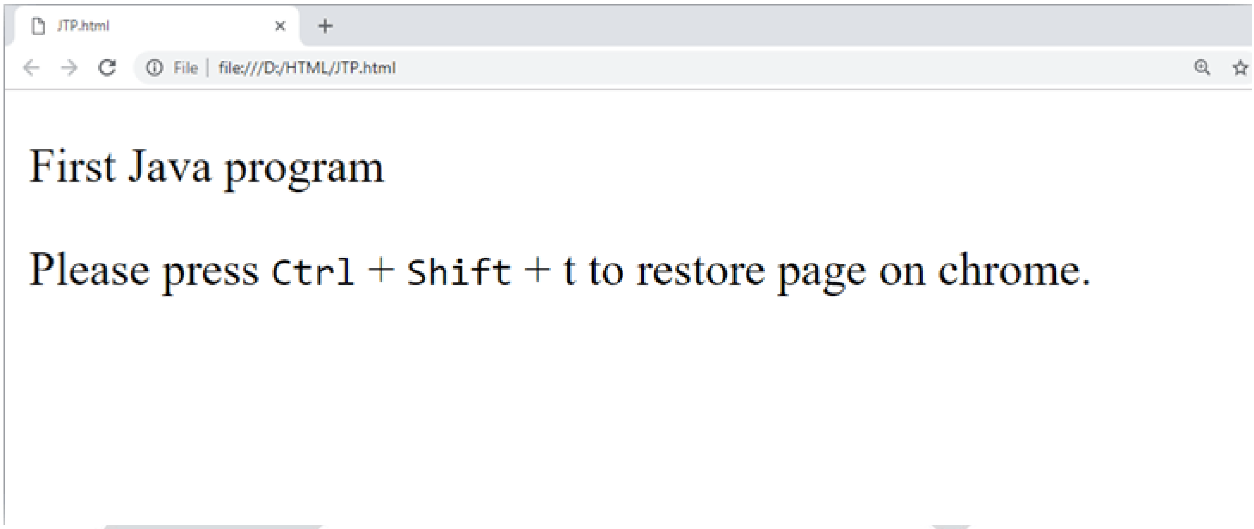
```
class Simple{ public static void main(String args[]){
System.out.println("Hello Java"); }}
```

9. Keyboard Tag

In HTML the keyboard tag, <kbd>, indicates that a section of content is a user input from keyboard.

<p>Please press <kbd>Ctrl</kbd> + <kbd>Shift</kbd> + t<kbd></kbd> to restore page on chrome.</p>

Output:



First Java program

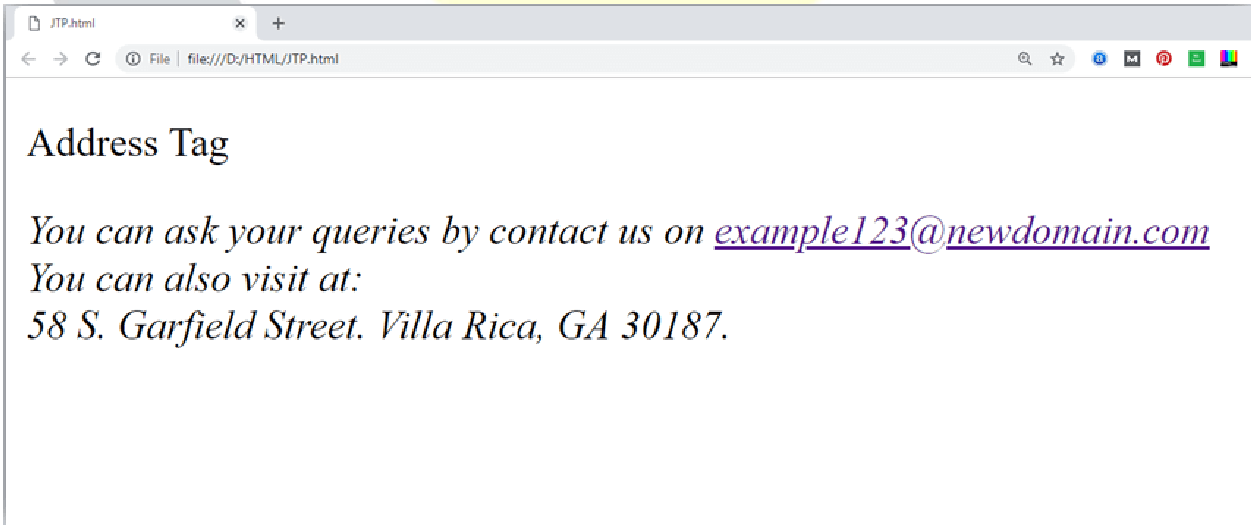
Please press Ctrl + Shift + t to restore page on chrome.

10. Address tag

An HTML <address> tag defines the contact information about the author of the content. The content written between <address> and </address> tag, then it will be displayed in italic font.

```
<address> You can ask your queries by contact us on <a
href="">example123@newdomain.com</a>
  <br> You can also visit at: <br>58 S. Garfield Street. Villa Rica, GA 30187.
 </address>
```

Output:



Address Tag

You can ask your queries by contact us on example123@newdomain.com
You can also visit at:
58 S. Garfield Street. Villa Rica, GA 30187.

HTML Anchor

The HTML anchor tag defines a hyperlink that links one page to another page. It can create hyperlink to other web page as well as files, location, or any URL. The "href" attribute is the most important attribute of the HTML a tag. and which links to destination page or URL.

href attribute of HTML anchor tag

The href attribute is used to define the address of the file to be linked. In other words, it points out the destination page.

The syntax of HTML anchor tag is given below.

```
<a href = "..........."> Link Text </a>
```

Let's see an example of HTML anchor tag.

```
<a href="second.html">Click for Second Page</a>
```

Specify a location for Link using target attribute

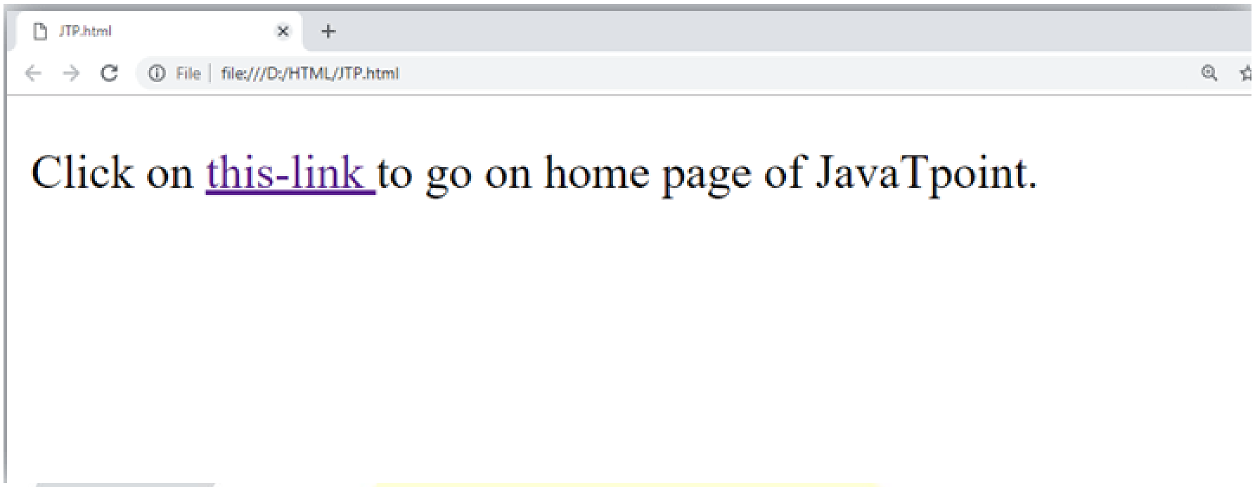
If we want to open that link to another page then we can use target attribute of <a> tag. With the help of this link will be open in next page.

Example:

```
<!DOCTYPE html>
<html>
<head>
   <title></title>
</head>
<body>
<p>Click on <a href="https://www.javatpoint.com/" target="_blank"> this-link </a>to go on home page of JavaTpoint.</p>
</body>
```

</html>

Output:



Click on this-link to go on home page of JavaTpoint.

Note:

- The target attribute can only use with href attribute in anchor tag.
- If we will not use target attribute then link will open in same page.

Appearance of HTML anchor tag

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

HTML Image

HTML img tag is used to display image on the web page. HTML img tag is an empty tag that contains attributes only, closing tags are not used in HTML image element.

Let's see an example of HTML image.

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

<img src="good_morning.jpg" alt="Good Morning Friends"/>
```

Output:

Attributes of HTML img tag

The src and alt are important attributes of HTML img tag. All attributes of HTML image tag are given below.

1) src

It is a necessary attribute that describes the source or path of the image. It instructs the browser where to look for the image on the server.

The location of image may be on the same directory or another server.

2) alt

The alt attribute defines an alternate text for the image, if it can't be displayed. The value of the alt attribute describe the image in words. The alt attribute is considered good for SEO prospective.

3) width

It is an optional attribute which is used to specify the width to display the image. It is not recommended now. You should apply CSS in place of width attribute.

4) height

It h3 the height of the image. The HTML height attribute also supports iframe, image and object elements. It is not recommended now. You should apply CSS in place of height attribute.

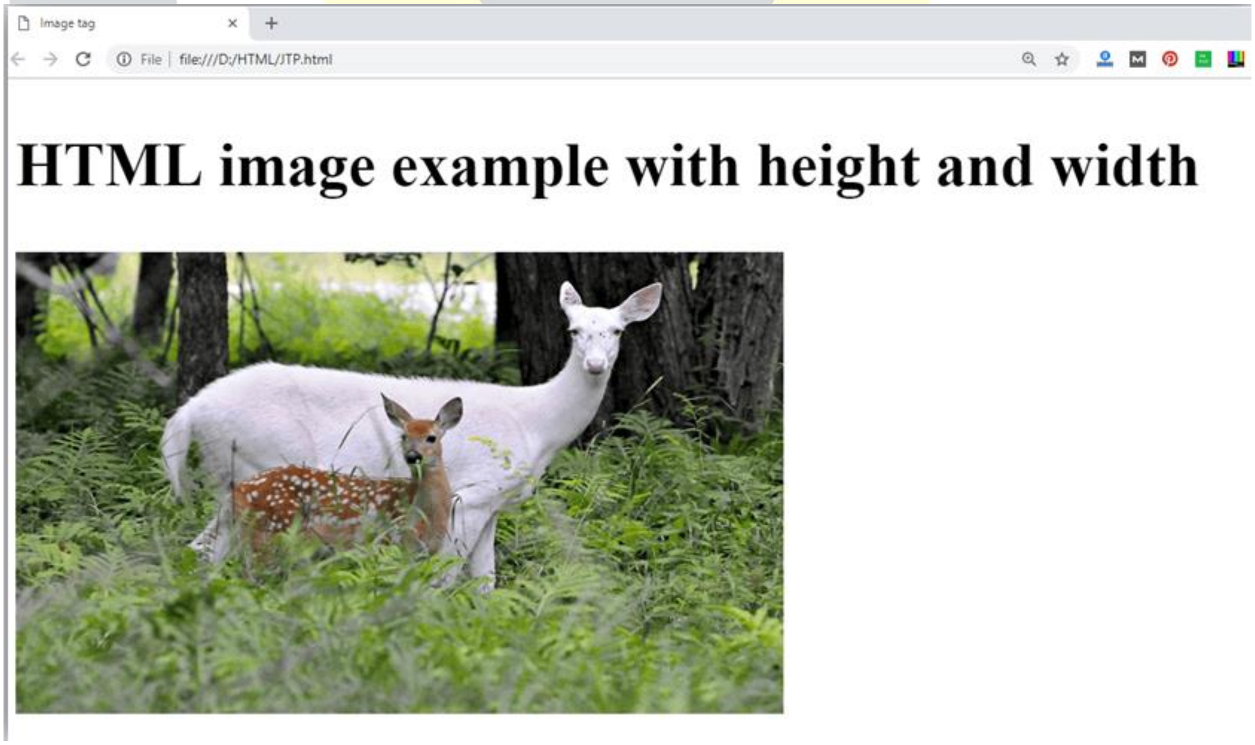
Use of height and width attribute with img tag

You have learnt about how to insert an image in your web page, now if we want to give some height and width to display image according to our requirement, then we can set it with height and width attributes of image.

Example:

```
<img src="animal.jpg" height="180" width="300" alt="animal image">
```

Output:



HTML image example with height and width

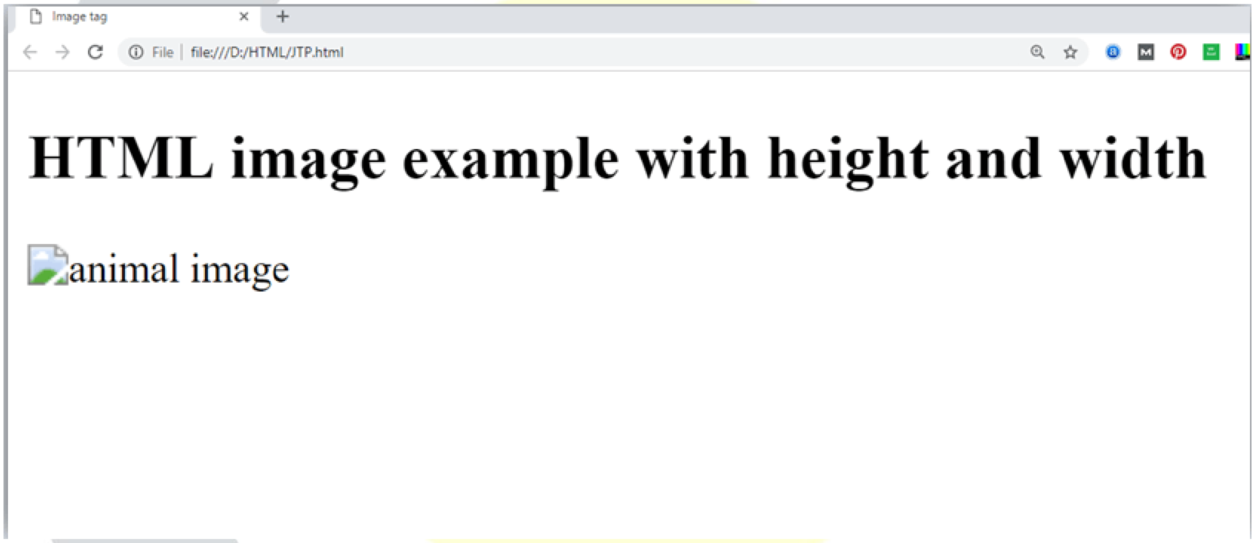
Note: Always try to insert the image with height and width, else it may flicker while displaying on webpage.

Use of alt attribute

We can use alt attribute with tag. It will display an alternative text in case if image cannot be displayed on browser. Following is the example for alt attribute:

<img src="animal.png" height="180" width="300" alt="animal image">

Output:



HTML image example with height and width

animal image

How to get image from another directory/folder?

To insert an image in your web, that image must be present in your same folder where you have put the HTML file. But if in some case image is available in some other directory then you can access the image like this:

<img src="E:/images/animal.png" height="180" width="300" alt="animal image">

In above statement we have put image in local disk E------>images folder------>animal.png.

Note: If src URL will be incorrect or misspell then it will not display your image on web page, so try to put correct URL.

Use <img> tag as a link

We can also link an image with other page or we can use an image as a link. To do this, put <img> tag inside the <a> tag.

Example:

```
<a href="https://www.javatpoint.com/what-is-robotics"><img src="robot.jpg" height="100"
width="100"></a>
```

Output:

HTML Table

HTML table tag is used to display data in tabular form (row * column). There can be many columns in a row.

We can create a table to display data in tabular form, using <table> element, with the help of <tr> , <td>, and <th> elements.

In Each table, table row is defined by <tr> tag, table header is defined by <th>, and table data is defined by <td> tags.

HTML tables are used to manage the layout of the page e.g. header section, navigation bar, body content, footer section etc. But it is recommended to use div tag over table to manage the layout of the page .

HTML Table Tags

Tag	Description
<table>	It defines a table.
<tr>	It defines a row in a table.
<th>	It defines a header cell in a table.
<td>	It defines a cell in a table.
<caption>	It defines the table caption.
<colgroup>	It specifies a group of one or more columns in a table for formatting.
<col>	It is used with <colgroup> element to specify column properties for each column.

<tbody>	It is used to group the body content in a table.
<thead>	It is used to group the header content in a table.
<tfoot>	It is used to group the footer content in a table.

HTML Table Example

Let's see the example of HTML table tag. It output is shown above.

```
<table>
<tr><th>First_Name</th><th>Last_Name</th><th>Marks</th></tr>
<tr><td>Sonoo</td><td>Jaiswal</td><td>60</td></tr>
<tr><td>James</td><td>William</td><td>80</td></tr>
<tr><td>Swati</td><td>Sironi</td><td>82</td></tr>
<tr><td>Chetna</td><td>Singh</td><td>72</td></tr>
</table>
```

In the above html table, there are 5 rows and 3 columns = 5 * 3 = 15 values.

HTML Table with Border

There are two ways to specify border for HTML tables.

- By border attribute of table in HTML
- By border property in CSS

1) HTML Border attribute

You can use border attribute of table tag in HTML to specify border. But it is not recommended now.

```
<table border="1">
```

```
<tr><th>First_Name</th><th>Last_Name</th><th>Marks</th></tr>
<tr><td>Sonoo</td><td>Jaiswal</td><td>60</td></tr>
<tr><td>James</td><td>William</td><td>80</td></tr>
<tr><td>Swati</td><td>Sironi</td><td>82</td></tr>
<tr><td>Chetna</td><td>Singh</td><td>72</td></tr>
</table>
```

2) CSS Border property

It is now recommended to use border property of CSS to specify border in table.

```
<style>
table, th, td {
  border: 1px solid black;
}
</style>
```

You can collapse all the borders in one border by border-collapse property. It will collapse the border into one.

```
<style>
table, th, td {
  border: 2px solid black;
  border-collapse: collapse;
}
</style>
```

HTML Table with cell padding

You can specify padding for table header and table data by two ways:

- By cellpadding attribute of table in HTML

- By padding property in CSS

The cellpadding attribute of HTML table tag is obselete now. It is recommended to use CSS. So let's see the code of CSS.

```
<style>
table, th, td {
  border: 1px solid pink;
  border-collapse: collapse;
}
th, td {
  padding: 10px;
}
</style>
```

HTML Table width:

We can specify the HTML table width using the CSS width property. It can be specify in pixels or percentage.

We can adjust our table width as per our requirement. Following is the example to display table with width.

```
table{
     width: 100%;
    }
```

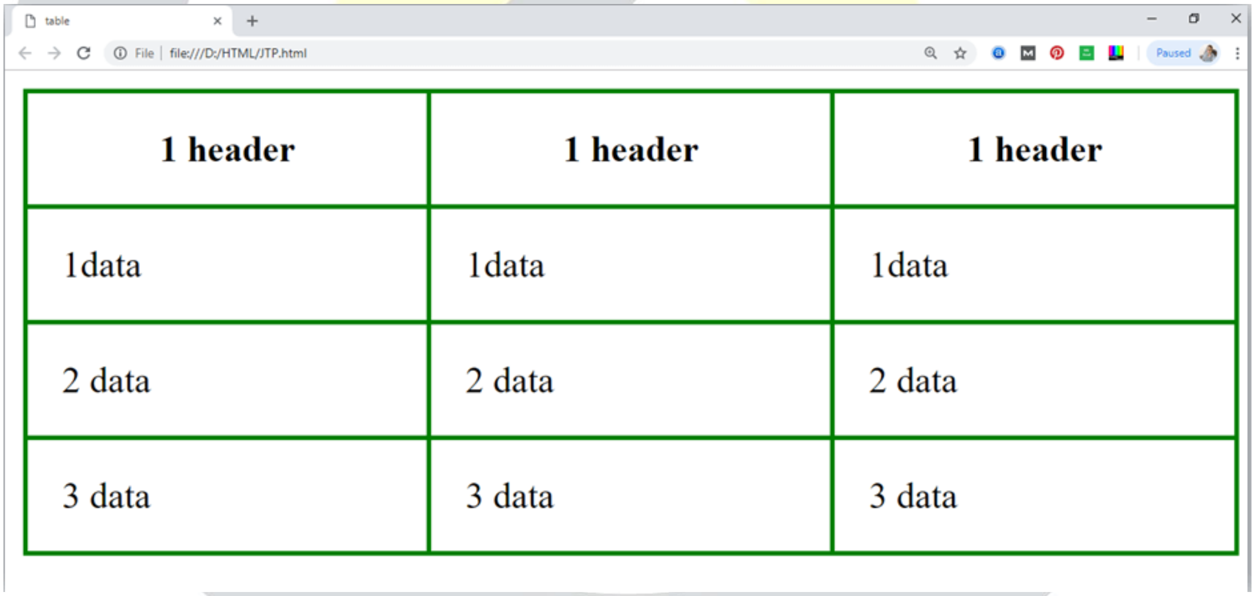
Example:

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

```
  <title>table</title>
  <style>
    table{
      border-collapse: collapse;
      width: 100%;
    }
  th,td{
    border: 2px solid green;
    padding: 15px;
  }

  </style>
 </head>
<body>
 <table>
  <tr>
    <th>1 header</th>
    <th>1 header</th>
    <th>1 header</th>
  </tr>
  <tr>
    <td>1data</td>
    <td>1data</td>
    <td>1data</td>
  </tr>
  <tr>
    <td>2 data</td>
    <td>2 data</td>
```

```
        <td>2 data</td>
    </tr>
    <tr>
        <td>3 data</td>
        <td>3 data</td>
        <td>3 data</td>
    </tr>
</table>
</body>
</html>
```

Output:



1 header	1 header	1 header
1data	1data	1data
2 data	2 data	2 data
3 data	3 data	3 data

HTML Table with colspan

If you want to make a cell span more than one column, you can use the colspan attribute.

It will divide one cell/row into multiple columns, and the number of columns depend on the value of colspan attribute.

Let's see the example that span two columns.

CSS code:

```
<style>
table, th, td {
  border: 1px solid black;
  border-collapse: collapse;
}
th, td {
  padding: 5px;
}
</style>
```

HTML code:

```
<table style="width:100%">
  <tr>
  <th>Name</th>
  <th colspan="2">Mobile No.</th>
  </tr>
  <tr>
  <td>Ajeet Maurya</td>
  <td>7503520801</td>
  <td>9555879135</td>
  </tr>
</table>
```

HTML Table with rowspan

If you want to make a cell span more than one row, you can use the rowspan attribute.

It will divide a cell into multiple rows. The number of divided rows will depend on rowspan values.

Let's see the example that span two rows.

CSS code:

```
<style>
table, th, td {
  border: 1px solid black;
  border-collapse: collapse;
}
th, td {
  padding: 10px;
}
</style>
```

HTML code:

```
<table>
<tr><th>Name</th><td>Ajeet Maurya</td></tr>
<tr><th rowspan="2">Mobile No.</th><td>7503520801</td></tr>
<tr><td>9555879135</td></tr>
</table>
```

HTML table with caption

HTML caption is diplayed above the table. It must be used after table tag only.

```
<table>
<caption>Student Records</caption>
<tr><th>First_Name</th><th>Last_Name</th><th>Marks</th></tr>
<tr><td>Vimal</td><td>Jaiswal</td><td>70</td></tr>
<tr><td>Mike</td><td>Warn</td><td>60</td></tr>
<tr><td>Shane</td><td>Warn</td><td>42</td></tr>
```

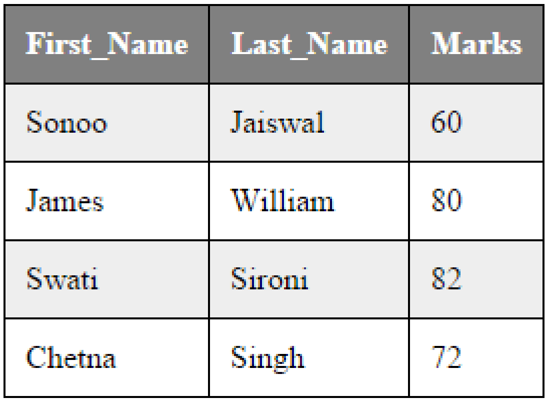
```
<tr><td>Jai</td><td>Malhotra</td><td>62</td></tr>
</table>
```

Styling HTML table even and odd cells

CSS code:

```
<style>
table, th, td {
  border: 1px solid black;
  border-collapse: collapse;
}
th, td {
  padding: 10px;
}
table#alter tr:nth-child(even) {
  background-color: #eee;
}
table#alter tr:nth-child(odd) {
  background-color: #fff;
}
table#alter th {
  color: white;
  background-color: gray;
}
</style>
```

Output:



First_Name	Last_Name	Marks
Sonoo	Jaiswal	60
James	William	80
Swati	Sironi	82
Chetna	Singh	72

NOTE: You can also create various types of tables using different CSS properties in your table.

HTML Lists

HTML Lists are used to specify lists of information. All lists may contain one or more list elements. There are three different types of HTML lists:

- Ordered List or Numbered List (ol)
- Unordered List or Bulleted List (ul)
- Description List or Definition List (dl)

Note: We can create a list inside another list, which will be termed as nested List.

HTML Ordered List or Numbered List

In the ordered HTML lists, all the list items are marked with numbers by default. It is known as numbered list also. The ordered list starts with <ol> tag and the list items start with <li> tag.

```
<ol>

 <li>Aries</li>

 <li>Bingo</li>

 <li>Leo</li>

 <li>Oracle</li>

</ol>
```

Output:

1. Aries
2. Bingo
3. Leo
4. Oracle

HTML Unordered List or Bulleted List

In HTML Unordered list, all the list items are marked with bullets. It is also known as bulleted list also. The Unordered list starts with <ul> tag and list items start with the <li> tag.

```
<ul>
 <li>Aries</li>
 <li>Bingo</li>
 <li>Leo</li>
 <li>Oracle</li>
</ul>
```

Output:

- Aries
- Bingo
- Leo
- Oracle

HTML Description List or Definition List

HTML Description list is also a list style which is supported by HTML and XHTML. It is also known as definition list where entries are listed like a dictionary or encyclopedia.

The definition list is very appropriate when you want to present glossary, list of terms or other name-value list.

The HTML definition list contains following three tags:

- <dl> tag defines the start of the list.
- <dt> tag defines a term.
- <dd> tag defines the term definition (description).

```
<dl>
  <dt>Aries</dt>
  <dd>-One of the 12 horoscope sign.</dd>
  <dt>Bingo</dt>
  <dd>-One of my evening snacks</dd>
```

```
 <dt>Leo</dt>
 <dd>-It is also an one of the 12 horoscope sign.</dd>
  <dt>Oracle</dt>
  <dd>-It is a multinational technology corporation.</dd>
</dl>
```

Output:

```
Aries
        -One of the 12 horoscope sign.
Bingo
        -One of my evening snacks
Leo
        -It is also an one of the 12 horoscope sign.
Oracle
        -It is a multinational technology corporation.
```

HTML Nested List

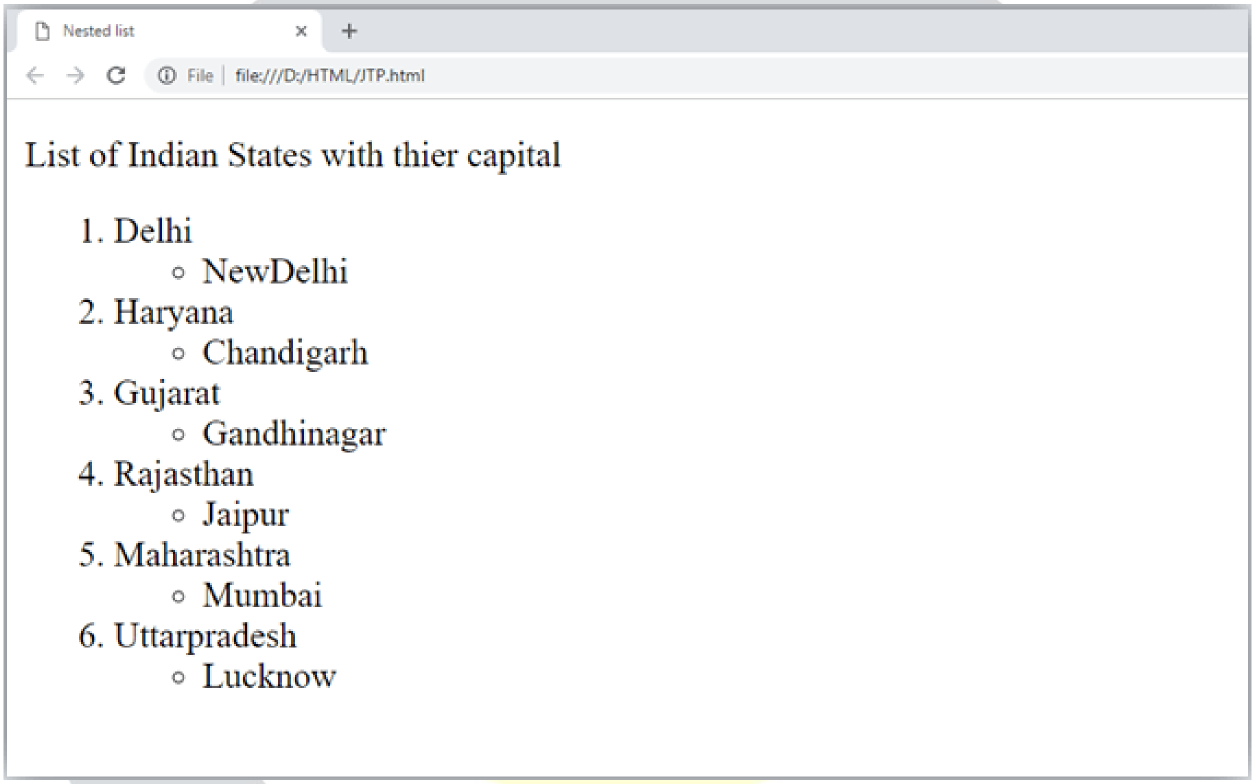
A list within another list is termed as nested list. If you want a bullet list inside a numbered list then such type of list will called as nested list.

Code:

```
<!DOCTYPE html>
<html>
<head>
   <title>Nested list</title>
</head>
<body>
   <p>List of Indian States with thier capital</p>
<ol>
```

```
<li>Delhi
    <ul>
        <li>NewDelhi</li>
    </ul>
</li>
<li>Haryana
    <ul>
        <li>Chandigarh</li>
    </ul>
</li>
<li>Gujarat
    <ul>
        <li>Gandhinagar</li>
    </ul>
</li>
<li>Rajasthan
    <ul>
        <li>Jaipur</li>
    </ul>
</li>
<li>Maharashtra
    <ul>
        <li>Mumbai</li>
    </ul>
</li>
<li>Uttarpradesh
    <ul>
        <li>Lucknow</li></ul>
    </li>
</ol>
</body>
</html>
```

Output:



List of Indian States with thier capital

```
    1. Delhi
          ◦ NewDelhi
    2. Haryana
          ◦ Chandigarh
    3. Gujarat
          ◦ Gandhinagar
    4. Rajasthan
          ◦ Jaipur
    5. Maharashtra
          ◦ Mumbai
    6. Uttarpradesh
          ◦ Lucknow
```

HTML Form

An HTML form is a section of a document which contains controls such as text fields, password fields, checkboxes, radio buttons, submit button, menus etc.

An HTML form facilitates the user to enter data that is to be sent to the server for processing such as name, email address, password, phone number, etc. .

Why use HTML Form

HTML forms are required if you want to collect some data from of the site visitor.

For example: If a user want to purchase some items on internet, he/she must fill the form such as shipping address and credit/debit card details so that item can be sent to the given address.

HTML Form Syntax

```
<form action="server url" method="get|post">
  //input controls e.g. textfield, textarea, radiobutton, button
</form>
```

HTML Form Tags

Let's see the list of HTML 5 form tags.

Tag	Description
<form>	It defines an HTML form to enter inputs by the used side.
<input>	It defines an input control.
<textarea>	It defines a multi-line input control.
<label>	It defines a label for an input element.
<fieldset>	It groups the related element in a form.

<legend>	It defines a caption for a <fieldset> element.
<select>	It defines a drop-down list.
<optgroup>	It defines a group of related options in a drop-down list.
<option>	It defines an option in a drop-down list.
<button>	It defines a clickable button.

HTML 5 Form Tags

Let's see the list of HTML 5 form tags.

Tag	Description
<datalist>	It specifies a list of pre-defined options for input control.
<keygen>	It defines a key-pair generator field for forms.
<output>	It defines the result of a calculation.

HTML <form> element

The HTML <form> element provide a document section to take input from user. It provides various interactive controls for submitting information to web server such as text field, text area, password field, etc.

Note: The <form> element does not itself create a form but it is container to contain all required form elements, such as <input>, <label>, etc.

Syntax:

```
<form>
//Form elements
</form>
```

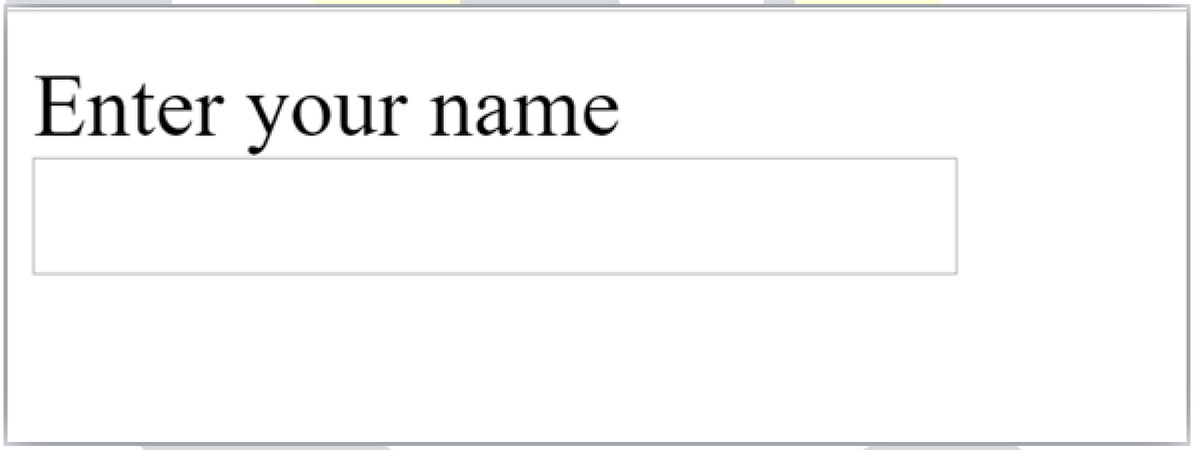
HTML <input> element

The HTML <input> element is fundamental form element. It is used to create form fields, to take input from user. We can apply different input filed to gather different information form user. Following is the example to show the simple text input.

Example:

```
<body>
  <form>
      Enter your name  <br>
     <input type="text" name="username">
  </form>
</body>
```

Output:



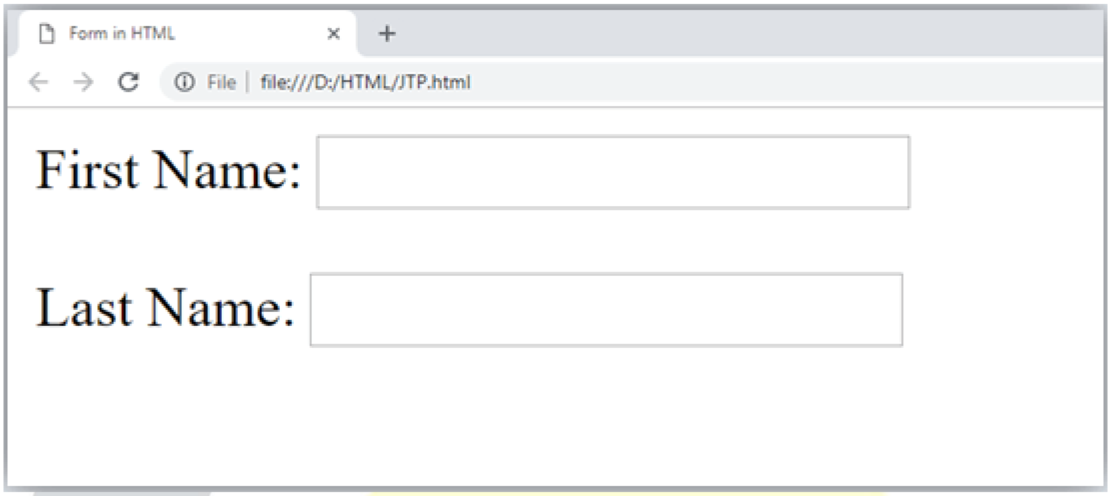
Enter your name

HTML TextField Control

The type="text" attribute of input tag creates textfield control also known as single line textfield control. The name attribute is optional, but it is required for the server side component such as JSP, ASP, PHP etc.

```
<form>
    First Name: <input type="text" name="firstname"/> <br/>
    Last Name:  <input type="text" name="lastname"/> <br/>
 </form>
```

Output:



First Name:

Last Name:

Note: If you will omit 'name' attribute then the text filed input will not be submitted to server.

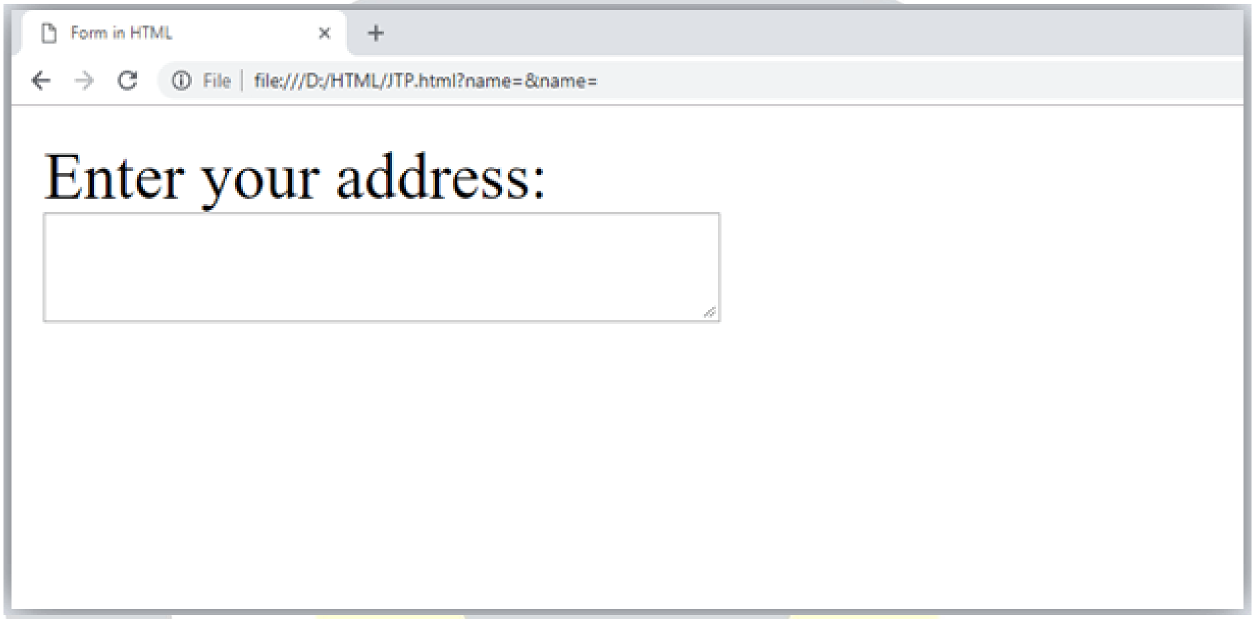
HTML <textarea> tag in form

The <textarea> tag in HTML is used to insert multiple-line text in a form. The size of <textarea> can be specify either using "rows" or "cols" attribute or by CSS.

Example:

```
<!DOCTYPE html>
<html>
<head>
    <title>Form in HTML</title>
</head>
<body>
  <form>
        Enter your address:<br>
      <textarea rows="2" cols="20"></textarea>
  </form>
</body>
</html>
```

Output:



Enter your address:

Label Tag in Form

It is considered better to have label in form. As it makes the code parser/browser/user friendly.

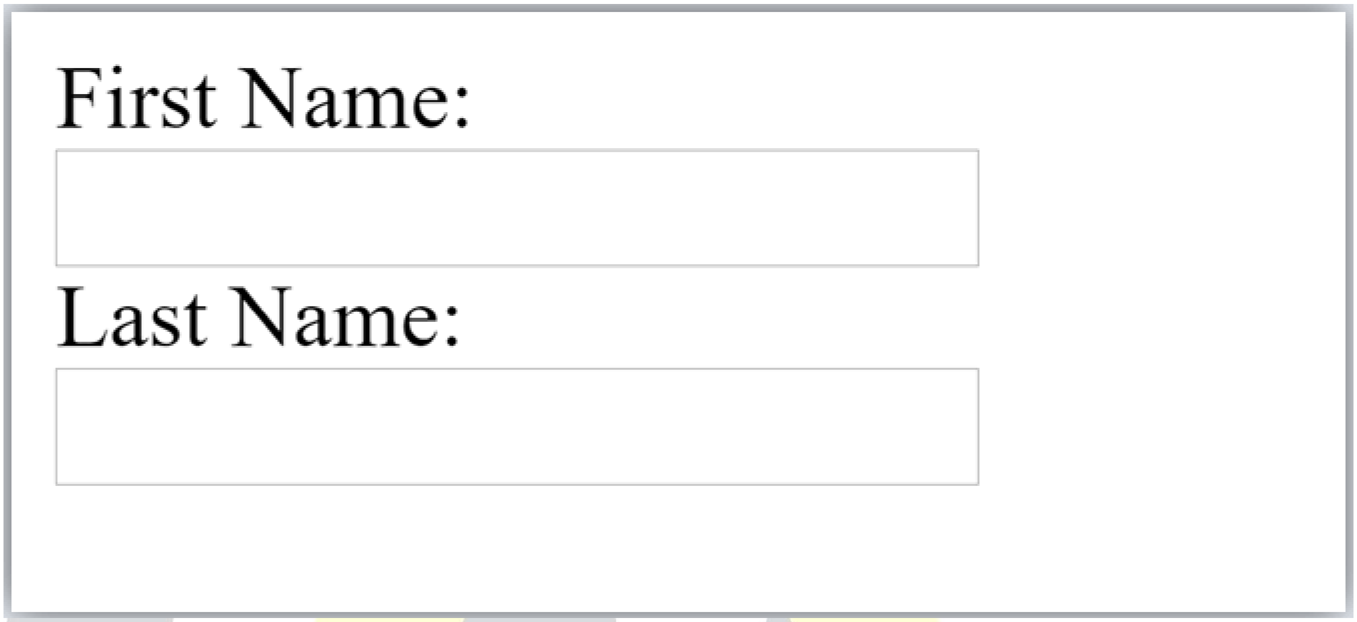
If you click on the label tag, it will focus on the text control. To do so, you need to have for attribute in label tag that must be same as id attribute of input tag.

NOTE: It is good to use <label> tag with form, although it is optional but if you will use it, then it will provide a focus when you tap or click on label tag. It is more worthy with touchscreens.

```
<form>
    <label for="firstname">First Name: </label> <br/>
              <input type="text" id="firstname" name="firstname"/> <br/>
   <label for="lastname">Last Name: </label>
              <input type="text" id="lastname" name="lastname"/> <br/>
```

```
 </form>
```

Output:



First Name:

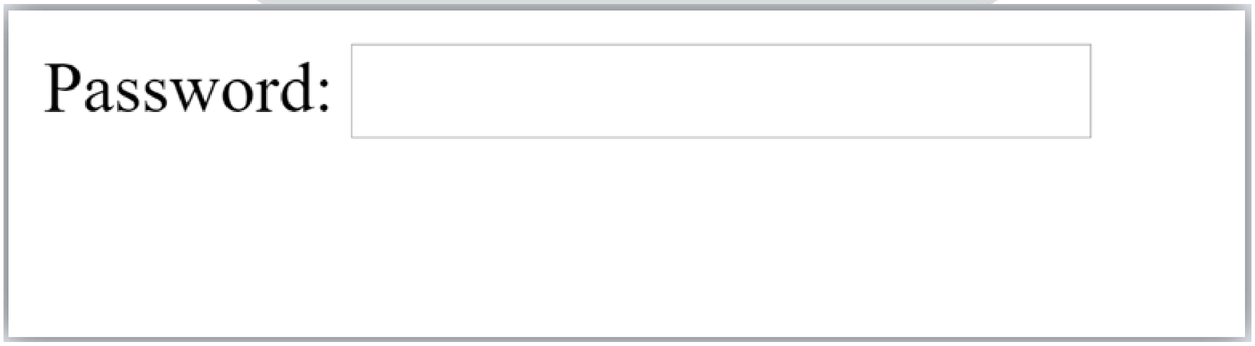
Last Name:

HTML Password Field Control

The password is not visible to the user in password field control.

```
<form>
    <label for="password">Password: </label>
              <input type="password" id="password" name="password"/> <br/>
</form>
```

Output:



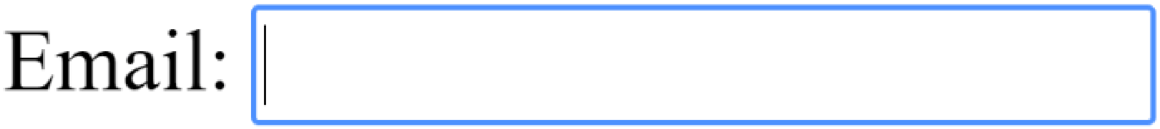
Password:

HTML 5 Email Field Control

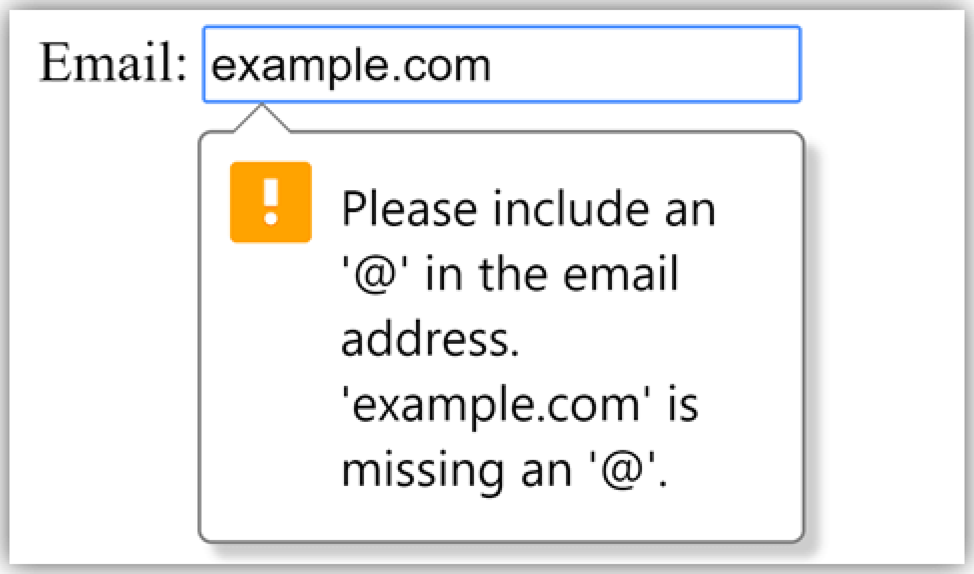
The email field in new in HTML 5. It validates the text for correct email address. You must use @ and . in this field.

```
<form>
    <label for="email">Email: </label>
              <input type="email" id="email" name="email"/> <br/>
</form>
```

It will display in browser like below:



Note: If we will not enter the correct email, it will display error like:



Email: example.com

Please include an '@' in the email address. 'example.com' is missing an '@'.

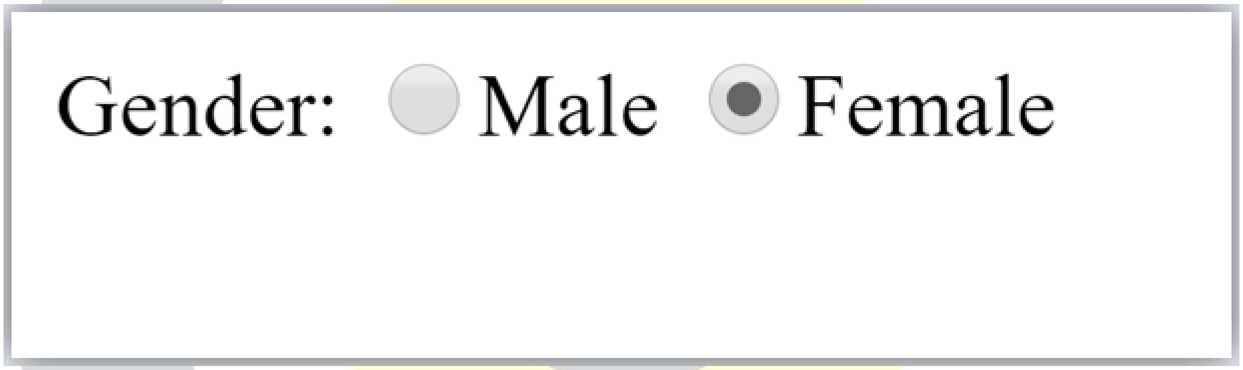
Radio Button Control

The radio button is used to select one option from multiple options. It is used for selection of gender, quiz questions etc.

If you use one name for all the radio buttons, only one radio button can be selected at a time.

Using radio buttons for multiple options, you can only choose a single option at a time.

```
<form>
   <label for="gender">Gender: </label>
          <input type="radio" id="gender" name="gender" value="male"/>Male
          <input type="radio" id="gender" name="gender" value="female"/>Female <br/>
</form>
```



Gender: ○ Male ● Female

Checkbox Control

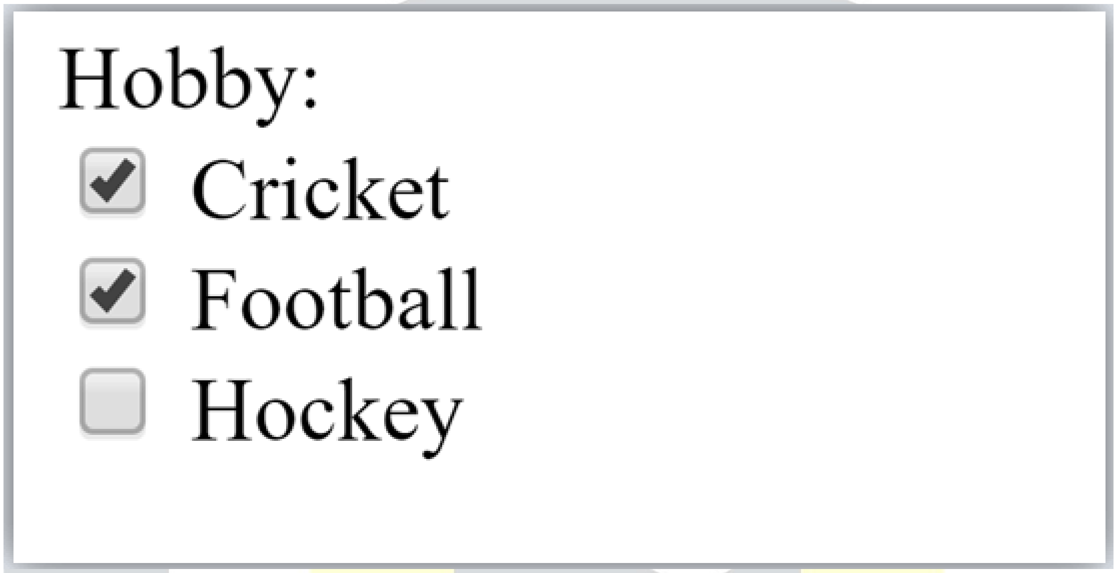
The checkbox control is used to check multiple options from given checkboxes.

```
<form>
Hobby:<br>
          <input type="checkbox" id="cricket" name="cricket" value="cricket"/>
            <label for="cricket">Cricket</label> <br>
          <input type="checkbox" id="football" name="football" value="football"/>
            <label for="football">Football</label> <br>
          <input type="checkbox" id="hockey" name="hockey" value="hockey"/>
```

```
        <label for="hockey">Hockey</label>
</form>
```

Note: These are similar to radio button except it can choose multiple options at a time and radio button can select one button at a time, and its display.

Output:



Hobby:

☑ Cricket

☑ Football

□ Hockey

Submit button control

HTML <input type="submit"> are used to add a submit button on web page. When user clicks on submit button, then form get submit to the server.

Syntax:

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

The type = submit , specifying that it is a submit button

The value attribute can be anything which we write on button on web page.

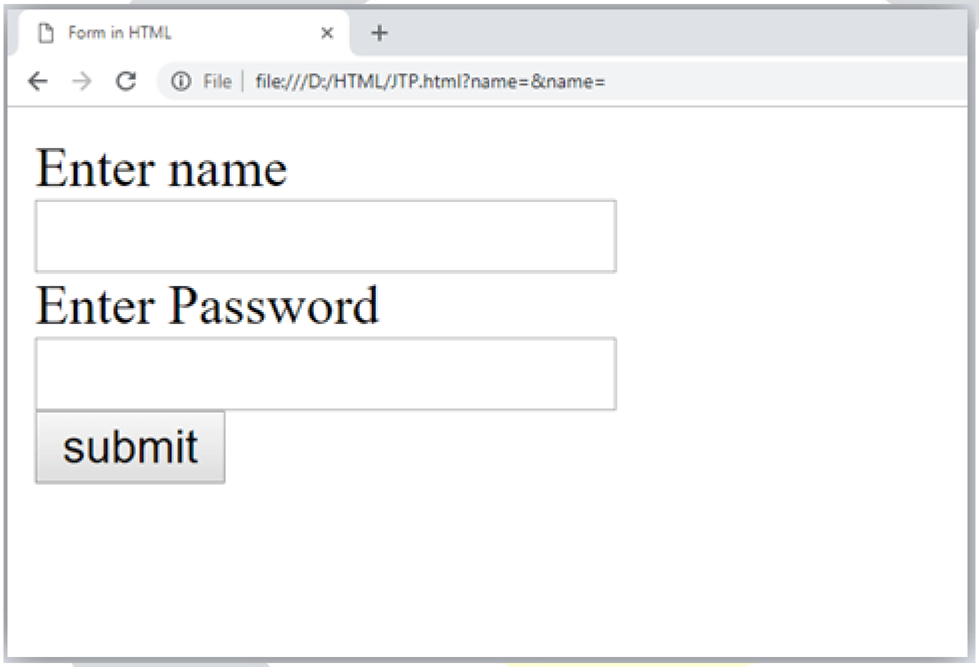
The name attribute can be omit here.

Example:

```
<form>
```

```
    <label for="name">Enter name</label><br>
    <input type="text" id="name" name="name"><br>
    <label for="pass">Enter Password</label><br>
    <input type="Password" id="pass" name="pass"><br>
    <input type="submit" value="submit">
</form>
```

Output:



HTML <fieldset> element:

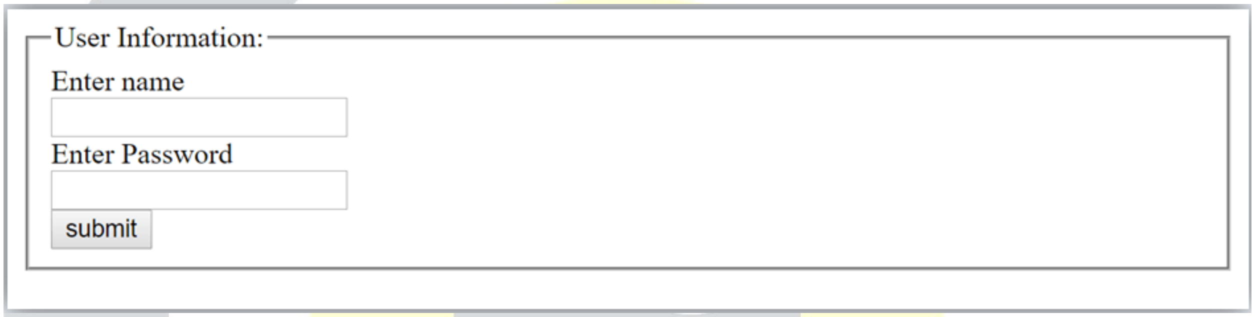
The <fieldset> element in HTML is used to group the related information of a form. This element is used with <legend> element which provide caption for the grouped elements.

Example:

```
 <form>
     <fieldset>
      <legend>User Information:</legend>
```

```
    <label for="name">Enter name</label><br>
<input type="text" id="name" name="name"><br>
<label for="pass">Enter Password</label><br>
<input type="Password" id="pass" name="pass"><br>
<input type="submit" value="submit">
</fieldset>
</form>
```

Output:



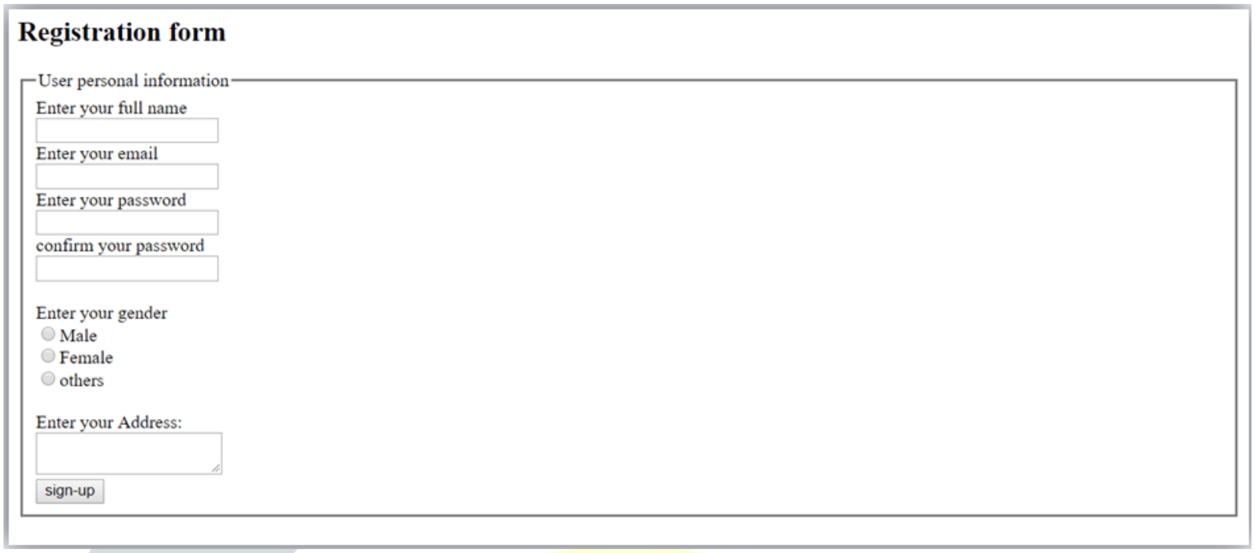
```
User Information:
Enter name
[                    ]
Enter Password
[                    ]
[submit]
```

HTML Form Example

Following is the example for a simple form of registration.

```
<!DOCTYPE html>
 <html>
 <head>
  <title>Form in HTML</title>
</head>
 <body>
     <h2>Registration form</h2>
    <form>
     <fieldset>
        <legend>User personal information</legend>
        <label>Enter your full name</label><br>
        <input type="text" name="name"><br>
         <label>Enter your email</label><br>
         <input type="email" name="email"><br>
         <label>Enter your password</label><br>
         <input type="password" name="pass"><br>
         <label>confirm your password</label><br>
         <input type="password" name="pass"><br>
         <br><label>Enter your gender</label><br>
         <input type="radio" id="gender" name="gender" value="male"/>Male  <br>
         <input type="radio" id="gender" name="gender" value="female"/>Female <br/>
         <input type="radio" id="gender" name="gender" value="others"/>others <br/>
          <br>Enter your Address:<br>
         <textarea></textarea><br>
         <input type="submit" value="sign-up">
     </fieldset>
  </form>
 </body>
</html>
```

Output:



Registration form

User personal information

Enter your full name

Enter your email

Enter your password

confirm your password

Enter your gender

○ Male

○ Female

○ others

Enter your Address:

HTML Form Example

Let's see a simple example of creating HTML form.

```
<form action="#">

<table>

<tr>

   <td class="tdLabel"><label for="register_name" class="label">Enter name:</label></td>

   <td><input type="text" name="name" value="" id="register_name"
style="width:160px"/></td>

</tr>

<tr>

   <td class="tdLabel"><label for="register_password" class="label">Enter
password:</label></td>

   <td><input type="password" name="password" id="register_password"
style="width:160px"/></td>

</tr>

<tr>

   <td class="tdLabel"><label for="register_email" class="label">Enter Email:</label></td>

   <td

><input type="email" name="email" value="" id="register_email" style="width:160px"/></td>
```

```
</tr>
<tr>
   <td class="tdLabel"><label for="register_gender" class="label">Enter Gender:</label></td>
   <td>
<input type="radio" name="gender" id="register_gendermale" value="male"/>
<label for="register_gendermale">male</label>
<input type="radio" name="gender" id="register_genderfemale" value="female"/>
<label for="register_genderfemale">female</label>
   </td>
</tr>
<tr>
   <td class="tdLabel"><label for="register_country" class="label">Select Country:</label></td>
   <td><select name="country" id="register_country" style="width:160px">
   <option value="india">india</option>
   <option value="pakistan">pakistan</option>
   <option value="africa">africa</option>
   <option value="china">china</option>
   <option value="other">other</option>
</select>
</td>
</tr>
<tr>
   <td colspan="2"><div align="right"><input type="submit" id="register_0" value="register"/>
</div></td>
</tr>
</table>
</form>
```

HTML Form Input

In HTML <input type=" "> is an important element of HTML form. The "type" attribute of input element can be various types, which defines information field. Such as <input type="text" name="name"> gives a text box.

Following is a list of all types of <input> element of HTML.

type=" "	Description
text	Defines a one-line text input field
password	Defines a one-line password input field
submit	Defines a submit button to submit the form to server
reset	Defines a reset button to reset all values in the form.
radio	Defines a radio button which allows select one option.
checkbox	Defines checkboxes which allow select multiple options form.
button	Defines a simple push button, which can be programmed to perform a task on an event.
file	Defines to select the file from device storage.
image	Defines a graphical submit button.

HTML5 added new types on <input> element. Following is the list of types of elements of HTML5

type=" "	Description

color	Defines an input field with a specific color.
date	Defines an input field for selection of date.
datetime-local	Defines an input field for entering a date without time zone.
email	Defines an input field for entering an email address.
month	Defines a control with month and year, without time zone.
number	Defines an input field to enter a number.
url	Defines a field for entering URL
week	Defines a field to enter the date with week-year, without time zone.
search	Defines a single line text field for entering a search string.
tel	Defines an input field for entering the telephone number.

Following is the description about types of <input> element with examples.

1. <input type="text">:

<input> element of type "text" are used to define a single-line input text field.

Example:

```
<form>
   <label>Enter first name</label><br>
   <input type="text" name="firstname"><br>
   <label>Enter last name</label><br>
   <input type="text" name="lastname"><br>
   <p><strong>Note:</strong>The default maximum character length is 20.</p>
```

```
</form>
```

Output:

Input "text" type:

The "text" field defines a single line input text field.

Enter first name

Enter last name

Note: The default maximum character length is 20.

2. <input type="password">:

The <input> element of type "password" allow a user to enter the password securely in a webpage. The entered text in password filed converted into "*" or ".", so that it cannot be read by another user.

Example:

```
<form>
   <label>Enter User name</label><br>
   <input type="text" name="firstname"><br>
   <label>Enter Password</label><br>
   <input type="Password" name="password"><br>
   <br><input type="submit" value="submit">
</form>
```

Output:

Input "password" type:

The "password" field defines a sinlge line input password field to enter the password securely.

Enter first name

Enter last name

submit

3. <input type="submit">:

The <input> element of type "submit" defines a submit button to submit the form to the server when the "click" event occurs.

Example:

```
<form action="https://www.javatpoint.com/html-tutorial">
   <label>Enter User name</label><br>
   <input type="text" name="firstname"><br>
   <label>Enter Password</label><br>
   <input type="Password" name="password"><br>
   <br><input type="submit" value="submit">
</form>
```

Output:

Input "submit" type:

Enter first name

Enter last name

submit

After clicking on submit button, this will submit the form to server and will redirect the page to action value.We will learn about "action" attribute in later chapters

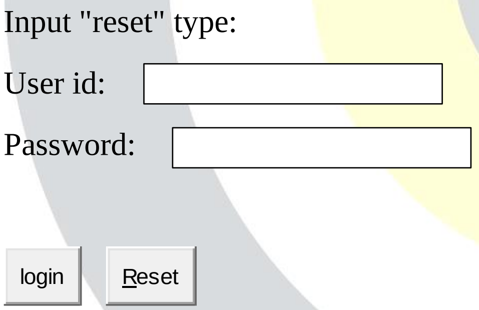
4. <input type="reset">:

The <input> type "reset" is also defined as a button but when the user performs a click event, it by default reset the all inputted values.

Example:

```
<form>
   <label>User id: </label>
    <input type="text" name="user-id" value="user"><br>
           <label>Password: </label>
    <input type="password" name="pass" value="pass"><br><br>
    <input type="submit" value="login">
     <input type="reset" value="Reset">
</form>
```

Output:



Input "reset" type:

User id:

Password:

Try to change the input values of user id and password, then when you click on reset, it will reset input fields with default values.

5. <input type="radio">:

The <input> type "radio" defines the radio buttons, which allow choosing an option between a set of related options. At a time only one radio button option can be selected at a time.

Example:

```
<form>
  <p>Kindly Select your favorite color</p>
  <input type="radio" name="color" value="red"> Red <br>
  <input type="radio" name="color" value="blue"> blue <br>
  <input type="radio" name="color" value="green">green <br>
  <input type="radio" name="color" value="pink">pink <br>
  <input type="submit" value="submit">
</form>
```

Output:

Input "radio" type

Kindly Select your favorite color

○ Red
○ blue
○ green
○ pink

submit

6. <input type="checkbox">:

The <input> type "checkbox" are displayed as square boxes which can be checked or unchecked to select the choices from the given options.

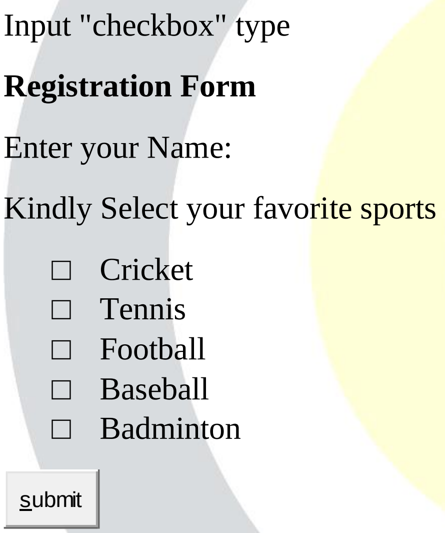
Note: The "radio" buttons are similar to checkboxes, but there is an important difference between both types: radio buttons allow the user to select only one option at a time, whereas checkbox allows a user to select zero to multiple options at a time.

Example:

```
<form>
    <label>Enter your Name:</label>
    <input type="text" name="name">
```

```
        <p>Kindly Select your favourite sports</p>
        <input type="checkbox" name="sport1" value="cricket">Cricket<br>
        <input type="checkbox" name="sport2" value="tennis">Tennis<br>
        <input type="checkbox" name="sport3" value="football">Football<br>
        <input type="checkbox" name="sport4" value="baseball">Baseball<br>
        <input type="checkbox" name="sport5" value="badminton">Badminton<br><br>
        <input type="submit" value="submit">
    </form>
```

Output:



Input "checkbox" type

Registration Form

Enter your Name:

Kindly Select your favorite sports

- □ Cricket
- □ Tennis
- □ Football
- □ Baseball
- □ Badminton

7. <input type="button">:

The <input> type "button" defines a simple push button, which can be programmed to control a functionally on any event such as, click event.

Note: It mainly works with JavaScript.

Example:

```
<form>
   <input type="button" value="submit " onclick="alert('you are learning HTML')">
</form>
```

Output:

Input "button" type.

Click the button to see the result:

Note: In the above example we have used the "alert" of JS, which you will learn in our JS tutorial. It is used to show a pop window.

8. <input type="file">:

The <input> element with type "file" is used to select one or more files from user device storage. Once you select the file, and after submission, this file can be uploaded to the server with the help of JS code and file API.

Example:

```
<form>
    <label>Select file to upload:</label>
    <input type="file" name="newfile">
    <input type="submit" value="submit">
</form>
```

Output:

Input "file" type.

We can choose any type of file until we do not specify it! The selected file will appear at next to "choose file" option

9. <input type="image">:

The <input> type "image" is used to represent a submit button in the form of image.

Example:

```
<!DOCTYPE html>
```

```
<html>
<body>
<h2>Input "image" type.</h2>
<p>We can create an image as submit button</p>
  <form>
    <label>User id:</label><br>
     <input type="text" name="name"><br><br>
     <input type="image" alt="Submit" src="login.png"  width="100px">
  </form>
 </body>
</html>
```

HTML5 newly added <input> types element

1. <input type="color">:

The <input> type "color" is used to define an input field which contains a colour. It allows a user to specify the colour by the visual colour interface on a browser.

Note: The "color" type only supports color value in hexadecimal format, and the default value is #000000 (black).

Example:

```
<form>
    Pick your Favorite color: <br><br>
    <input type="color" name="upclick" value="#a52a2a"> Upclick<br><br>
    <input type="color" name="downclick" value="#f5f5dc"> Downclick
</form>
```

Note:The default value of "color" type is #000000 (black). It only supports color value in hexadecimal format.

2. <input type="date">:

The <input> element of type "date" generates an input field, which allows a user to input the date in a given format. A user can enter the date by text field or by date picker interface.

Example:

```
<form>
  Select Start and End Date: <br><br>
    <input type="date" name="Startdate"> Start date:<br><br>
    <input type="date" name="Enddate"> End date:<br><br>
   <input type="submit">
</form>
```

3. <input type="datetime-local">:

The <input> element of type "datetime-local" creates input filed which allow a user to select the date as well as local time in the hour and minute without time zone information.

Example:

```
<form>
  <label>
    Select the meeting schedule: <br><br>
    Select date & time: <input type="datetime-local" name="meetingdate"> <br><br>
  </label>
    <input type="submit">
</form>
```

4. <input type="email">:

The <input> type "email" creates an input filed which allow a user to enter the e-mail address with pattern validation. The multiple attributes allow a user to enter more than one email address.

Example:

```
<form>
        <label><b>Enter your Email-address</b></label>
       <input type="email" name="email" required>
       <input type="submit">
        <p><strong>Note:</strong>User can also enter multiple email addresses separating by comma or whitespace as following: </p>
        <label><b>Enter multiple Email-addresses</b></label>
        <input type="email" name="email"  multiple>
       <input type="submit">
</form>
```

Note:User can also enter multiple email addresses separating by comma or whitespace as following:

5. <input type="month">:

The <input> type "month" creates an input field which allows a user to easily enter month and year in the format of "MM, YYYY" where MM defines month value, and YYYY defines the year value. New

Example:

```
<form>
   <label>Enter your Birth Month-year: </label>
   <input type="month" name="newMonth">
   <input type="submit">
</form>
```

6. <input type="number">:

The <input> element type number creates input filed which allows a user to enter the numeric value. You can also restrict to enter a minimum and maximum value using min and max attribute.

Example:

```
<form>
   <label>Enter your age: </label>
   <input type="number" name="num" min="50" max="80">
    <input type="submit">
</form>
```

Note:It will allow to enter number in range of 50-80. If you want to enter number other than range, it will show an error.

7. <input type="url">:

The <input> element of type "url" creates an input filed which enables user to enter the URL.

Example:

```
<form>
   <label>Enter your website URL: </label>
   <input type="url" name="website" placeholder="http://example.com"><br>
   <input type="submit" value="send data">
</form>
```

8. <input type="week">:

The <input> type week creates an input field which allows a user to select a week and year form the drop-down calendar without time zone.

Example:

```
<form>
```

```
    <label><b>Select your best week of year:</b></label><br><br>

    <input type="week" name="bestweek">

    <input type="submit" value="Send data">

 </form>
```

9. <input type="search">:

The <input> type "search" creates an input filed which allows a user to enter a search string. These are functionally symmetrical to the text input type, but may be styled differently.

Example:

```
<form>

    <label>Search here:</label>

    <input type="search" name="q">

    <input type="submit" value="search">

</form>
```

10. <input type="tel">:

The <input> element of type ?tel? creates an input filed to enter the telephone number. The "tel" type does not have default validation such as email, because telephone number pattern can vary worldwide.

Example:

```
<form>

      <label><b>Enter your Telephone Number(in format of xxx-xxx-xxxx):</b></label>

      <input type="tel" name="telephone" pattern="[0-9]{3}-[0-9]{3}-[0-9]{4}" required>

      <input type="submit"><br><br>

   </form>
```

Note: Here we are using two attributes that are "pattern" and"required" which will allow user to enter the number in given format and it is required to enter the number in input field.

HTML form Attribute

HTML <form> element attributes

In HTML there are various attributes available for <form> element which are given below:

HTML action attribute

The action attribute of <form> element defines the process to be performed on form when form is submitted, or it is a URI to process the form information.

The action attribute value defines the web page where information proceed. It can be .php, .jsp, .asp, etc. or any URL where you want to process your form.

Note: If action attribute value is blank then form will be processed to the same page.

Example:

```
<form action="action.html" method="post">
<label>User Name:</label><br>
<input type="text" name="name"><br><br>
<label>User Password</label><br>
<input type="password" name="pass"><br><br>
 <input type="submit">
   </form>
```

It will redirect to a new page "action.html" when you click on submit button

HTML method attribute

The method attribute defines the HTTP method which browser used to submit the form. The possible values of method attribute can be:

- **post:** We can use the post value of method attribute when we want to process the sensitive data as it does not display the submitted data in URL.

Example:
<form action="action.html" method="post">

- **get:** The get value of method attribute is default value while submitting the form. But this is not secure as it displays data in URL after submitting the form.

  Example:
  <form action="action.html" method="get">

When submitting the data, it will display the entered data in the form of:

file:///D:/HTML/action.html?name=JavaTPoint&pass=123

HTML target attribute

The target attribute defines where to open the response after submitting the form. The following are the keywords used with the target attribute.

- **_self:** If we use _self as an attribute value, then the response will display in current page only.
  Example:
  <form action="action.html" method="get" target="_self">

- **_blank:** If we use _blank as an attribute it will load the response in a new page.
  Example:
  <form action="action.html" method="get" target="_blank">

HTML autocomplete attribute

The HTML autocomplete attribute is a newly added attribute of HTML5 which enables an input field to complete automatically. It can have two values "on" and "off" which enables autocomplete either ON or OFF. The default value of autocomplete attribute is "on".

Example:

<form action="action.html" method="get" autocomplete="on">

Example:

<form action="action.html" method="get" autocomplete="off">

Note: it can be used with <form> element and <input> element both.

HTML enctype attribute

The HTML enctype attribute defines the encoding type of form-content while submitting the form to the server. The possible values of enctype can be:

- **application/x-www-form-urlencoded:** It is default encoding type if the enctype attribute is not included in the form. All characters are encoded before submitting the form.
  Example:
  <form action="action.html" method="post" enctype="application/x-www-form-urlencoded" >

- **multipart/form-data:** It does not encode any character. It is used when our form contains file-upload controls.
  Example:
  <form action="action.html" method="post" enctype="multipart/form-data">

- **text/plain (HTML5):** In this encoding type only space are encoded into + symbol and no any other special character encoded.
  Example:
  <form action="action.html" method="post" enctype="text/plain" >

HTML novalidate attribute HTML5

The novalidate attribute is newly added Boolean attribute of HTML5. If we apply this attribute in form then it does not perform any type of validation and submit the form.

Example:

<form action = "action.html" method = "get" novalidate>

Try to change the form detials with novalidate atttribute and without novalidate attribute and see the difference.

HTML <input> element attribute

HTML name attribute

The HTML name attribute defines the name of an input element. The name and value attribute are included in HTTP request when we submit the form.

Note: One should not omit the name attribute as when we submit the form the HTTP request includes both name-value pair and if name is not available it will not process that input field.

Example:

```
<form action = "action.html" method = "get">
        Enter name:<br><input type="name" name="uname"><br>
        Enter age:<br><input type="number" name="age"><br>
        Enter email:<br><input type="email"><br>
        <input type="submit" value="Submit">
      </form>
```

Note: If you will not use name attribute in any input field, then that input field will not be submitted, when submit the form.

Click on submit and see the URL where email is not included in HTTP request as we have not used name attribute in the email input field

HTML value attribute

The HTML value attribute defines the initial value or default value of an input field.

Example:

```
<form>
       <label>Enter your Name</label><br>
       <input type="text" name="uname" value="Enter Name"><br><br>
       <label>Enter your Email-address</label><br>
       <input type="text" name="uname" value="Enter email"><br><br>
         <label>Enter your password</label><br>
```

```
        <input type="password" name="pass" value=""><br><br>
        <input type="submit" value="login">
   </form>
```

Note: In password input filed the value attribute will always unclear

HTML required attribute HTML5

HTML required is a Boolean attribute which specifies that user must fill that filed before submitting the form.

Example:

```
<form>
        <label>Enter your Email-address</label><br>
        <input type="text" name="uname" required><br><br>
         <label>Enter your password</label><br>
        <input type="password" name="pass"><br><br>
        <input type="submit" value="login">
   </form>
```

If you will try to submit the form without completing email field then it will give an error pop up.

HTML autofocus attribute HTML5

The autofocus is a Boolean attribute which enables a field automatically focused when a webpage loads.

Example:

```
<form>
        <label>Enter your Email-address</label><br>
```

```
      <input type="text" name="uname" autofocus><br><br>
       <label>Enter your password</label><br>
      <input type="password" name="pass"><br><br>
      <input type="submit" value="login">
  </form>
```

HTML placeholder attribute HTML5

The placeholder attribute specifies a text within an input field which informs the user about the expected input of that filed.

The placeholder attribute can be used with text, password, email, and URL values.

When the user enters the value, the placeholder will be automatically removed.

Example:

```
<form>
      <label>Enter your name</label><br>
      <input type="text" name="uname" placeholder="Your name"><br><br>
         <label>Enter your Email address</label><br>
      <input type="email" name="email" placeholder="example@gmail.com"><br><br>
         <label>Enter your password</label><br>
      <input type="password" name="pass" placeholder="your password"><br><br>
      <input type="submit" value="login">
   </form>
```

HTML disabled attribute

The HTML disabled attribute when applied then it disable that input field. The disabled field does not allow the user to interact with that field.

The disabled input filed does not receive click events, and these input value will not be sent to the server when submitting the form.

Example:

```
<input type="text" name="uname" disabled><br><br>
```

HTML size attribute

The size attribute controls the size of the input field in typed characters.

Example:

```
<label>Account holder name</label><br>
      <input type="text" name="uname" size="40" required><br><br>
      <label>Account number</label><br>
      <input type="text" name="an" size="30" required><br><br>
      <label>CVV</label><br>
      <input type="text" name="cvv"  size="1" required><br><br>
```

HTML form attribute

HTML form attribute allows a user to specify an input filed outside the form but remains the part of the parent form.

Example:

```
User email: <br><input type="email" name="email"  form="fcontrol"  required><br>
       <input type="submit" form="fcontrol">
```

HTML with CSS

Let's suppose we have created our web page using a simple HTML code, and we want something which can present our page in a correct format, and visibly attractive. So to do this, we can style our web page with CSS (Cascading Stylesheet) properties.

CSS is used to apply the style in the web page which is made up of HTML elements. It describes the look of the webpage.

CSS provides various style properties such as background color, padding, margin, border-color, and many more, to style a webpage.

Each property in CSS has a name-value pair, and each property is separated by a semicolon (;).

Note: In this chapter, we have given a small overview of CSS. You will learn everything in depth about CSS in our CSS tutorial.

Example:

```
<body style="text-align: center;">

    <h2 style="color: red;">Welcome to javaTpoint</h2>

    <p style="color: blue; font-size: 25px; font-style: italic ;">This is a great website to learn technologies in very simple way. </p>

</body>
```

In the above example, we have used a style attribute to provide some styling format to our code.

Three ways to apply CSS

To use CSS with HTML document, there are three ways:

- **Inline CSS:** Define CSS properties using style attribute in the HTML elements.
- **Internal or Embedded CSS:** Define CSS using <style> tag in <head> section.
- **External CSS:** Define all CSS property in a separate .css file, and then include the file with HTML file using tag in section.

Inline CSS:

Inline CSS is used to apply CSS in a single element. It can apply style uniquely in each element.

To apply inline CSS, you need to use style attribute within HTML element. We can use as many properties as we want, but each property should be separated by a semicolon (;).

Example:

```
<h3 style="color: red;
        font-style: italic;
        text-align: center;
        font-size: 50px;
        padding-top: 25px;">Learning HTML using Inline CSS</h3>
```

Internal CSS:

An Internal stylesheets contains the CSS properties for a webpage in <head> section of HTML document. To use Internal CSS, we can use class and id attributes.

We can use internal CSS to apply a style for a single HTML page.

Example:

```
<!DOCTYPE html>
<html>
<head>
            <style>
    /*Internal CSS using element name*/
        body{background-color:lavender;
         text-align: center;}
         h2{font-style: italic;
          font-size: 30px;
          color: #f08080;}
        p{font-size: 20px;}
        /*Internal CSS using class name*/
            .blue{color: blue;}
            .red{color: red;}
            .green{color: green;}
      </style>
    </head>
  <body>
   <h2>Learning HTML with internal CSS</h2>
    <p class="blue">This is a blue color paragraph</p>
    <p class="red">This is a red color paragraph</p>
    <p class="green">This is a green color paragraph</p>
  </body>
</html>
```

Note: In the above example, we have used a class attribute which you will learn in the next chapter.

External CSS:

An external CSS contains a separate CSS file which only contains style code using the class name, id name, tag name, etc. We can use this CSS file in any HTML file by including it in HTML file using <link> tag.

If we have multiple HTML pages for an application and which use similar CSS, then we can use external CSS.

There are two files need to create to apply external CSS

First, create the HTML file

Create a CSS file and save it using the .css extension (This file only will only contain the styling code.)

Link the CSS file in your HTML file using tag in header section of HTML document.

Example:

```
<!DOCTYPE html>
<html>
<head>
    <link rel="stylesheet" type="text/css" href="style.css">
    </head>
  <body>
   <h2>Learning HTML with External CSS</h2>
    <p class="blue">This is a blue color paragraph</p>
    <p class="red">This is a red color paragraph</p>
    <p class="green">This is a green color paragraph</p>
  </body>
</html>
```

CSS file:

```
body{
background-color:lavender;
text-align: center;
}
h2{
font-style: italic;
size: 30px;
color: #f08080;
}
p{
font-size: 20px;
}
.blue{
color: blue;
}
.red{
color: red;
}
.green{
color: green;
}
```

Properties-name	Syntax	Description
background-color	background-color:red;	It defines the background color of that element.
color	color: lightgreen;	It defines the color of text of an element
padding	padding: 20px;	It defines the space between content and the border.
margin	margin: 30px; margin-left:	It creates space around an element.
font-family	font-family: cursive;	Font-family defines a font for a particular element.
Font-size	font-size: 50px;	Font-size defines a font size for a particular element.
text-align	text-align: left;	It is used to align the text in a selected position.

HTML Classes

Class Attribute in HTML

The HTML class attribute is used to specify a single or multiple class names for an HTML element. The class name can be used by CSS and JavaScript to do some tasks for HTML elements. You can use this class in CSS with a specific class, write a period (.) character, followed by the name of the class for selecting elements.

A class attribute can be defined within <style> tag or in separate file using the (.) character.

In an HTML document, we can use the same class attribute name with different elements.

Defining an HTML class

To create an HTML class, firstly define style for HTML class using <style> tag within <head> section as following example:

Example:

```
<head>
    <style>
        .headings{
            color: lightgreen;
            font-family: cursive;
            background-color: black; }
    </style>
</head>
```

We have define style for a class name "headings", and we can use this class name with any of HTML element in which we want to provide such styling. We just need to follow the following syntax to use it.

```
<tag class="ghf"> content </tag>
```

Example 1:

```
<!DOCTYPE html>
<html>
<head>
  <style>
    .headings{
      color: lightgreen;
      font-family: cursive;
      background-color: black; }
  </style>
</head>
<body>
<h1 class="headings">This is first heading</h1>
<h2 class="headings">This is Second heading</h2>
<h3 class="headings">This is third heading</h3>
<h4 class="headings">This is fourth heading</h4>
</body>
</html>
```

Another Example with different class name

Example:

Let's use a class name "Fruit" with CSS to style all elements.

```
<style>
.fruit {
  background-color: orange;
  color: white;
  padding: 10px;
}
```

```
</style>

<h2 class="fruit">Mango</h2>
<p>Mango is king of all fruits.</p>

<h2 class="fruit">Orange</h2>
<p>Oranges are full of Vitamin C.</p>

<h2 class="fruit">Apple</h2>
<p>An apple a day, keeps the Doctor away.</p>
```

Here you can see that we have used the class name "fruit" with (.) to use all its elements.

Note: You can use class attribute on any HTML element. The class name is case-sensitive.

Class Attribute in JavaScript

You can use JavaScript access elements with a specified class name by using the getElementsByClassName() method.

Example:

Let's hide all the elements with class name "fruit" when the user click on the button.

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

<h2>Class Attribute with JavaScript</h2>
<p>Click the button, to hide all elements with the class name "fruit", with JavaScript:</p>

<button onclick="myFunction()">Hide elements</button>
```

```
<h2 class="fruit">Mango</h2>
<p>Mango is king of all fruits.</p>

<h2 class="fruit">Orange</h2>
<p>Oranges are full of Vitamin C.</p>

<h2 class="fruit">Apple</h2>
<p>An apple a day, keeps the Doctor away.</p>

<script>
function myFunction() {
  var x = document.getElementsByClassName("fruit");
  for (var i = 0; i < x.length; i++) {
    x[i].style.display = "none";
  }
}
</script>

</body>
</html>
```

Note: You will learn more about JavaScript in our JavaScript tutorial.

Multiple Classes

You can use multiple class names (more than one) with HTML elements. These class names must be separated by a space.

Example:

Let's style elements with class name "fruit" and also with a class name "center".

```
<!DOCTYPE html>
<html>
<style>
.fruit {
   background-color: orange;
   color: white;
   padding: 10px;
}

.center {
   text-align: center;
}
</style>
<body>

<h2>Multiple Classes</h2>
<p>All three elements have the class name "fruit". In addition, Mango also have the class name "center", which center-aligns the text.</p>

<h2 class="fruit center">Mango</h2>
<h2 class="fruit">Orange</h2>
<h2 class="fruit">Apple</h2>

</body>
</html>
```

Test it Now

You can see that the first element <h2> belongs to both the "fruit" class and the "center" class.

Same class with Different Tag

You can use the same class name with different tags like <h2> and <p> etc. to share the same style.

Example:

```
<!DOCTYPE html>
<html>
<style>
.fruit {
  background-color: orange;
  color: white;
  padding: 10px;
}
</style>
<body>
<h2>Same Class with Different Tag</h2>
<h2 class="fruit">Mango</h2>
<p class="fruit">Mango is the king of all fruits.</p>
</body>
</html>
```

HTML Id Attribute

The id attribute is used to specify the unique ID for an element of the HTML document. It allocates the unique identifier which is used by the CSS and the JavaScript for performing certain tasks.

Note: In the Cascading Style sheet (CSS), we can easily select an element with the specific id by using the # symbol followed by id.

Note: JavaScript can access an element with the given ID by using the getElementById() method.

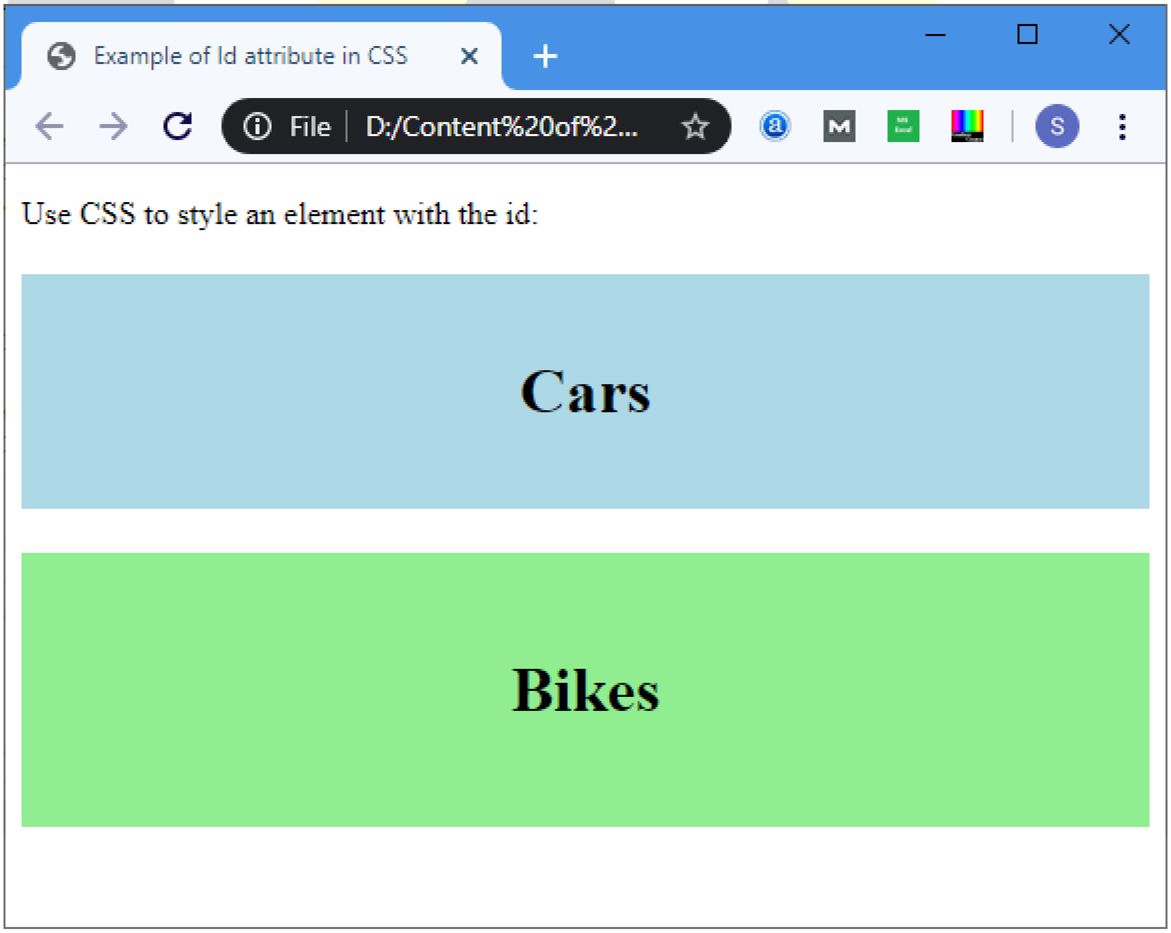
Syntax

```
<tag id="value">
```

Example 1: The following example describes how to use the id attribute in CSS document:

```
<!DOCTYPE html>
<html>
<head>
<title>
Example of Id attribute in CSS
</title>
<style>
#Cars {
padding: 40px;
background-color: lightblue;
color: black;
text-align: center;
}
#Bikes
{
padding: 50px;
background-color: lightGreen;
text-align: center;
}
</style>
</head>
<body>
<p> Use CSS to style an element with the id: </p>
<h1 id="Cars"> Cars </h1>
<h1 id="Bikes"> Bikes </h1>
</body>
</html>
```

Output:



Use CSS to style an element with the id:

Cars

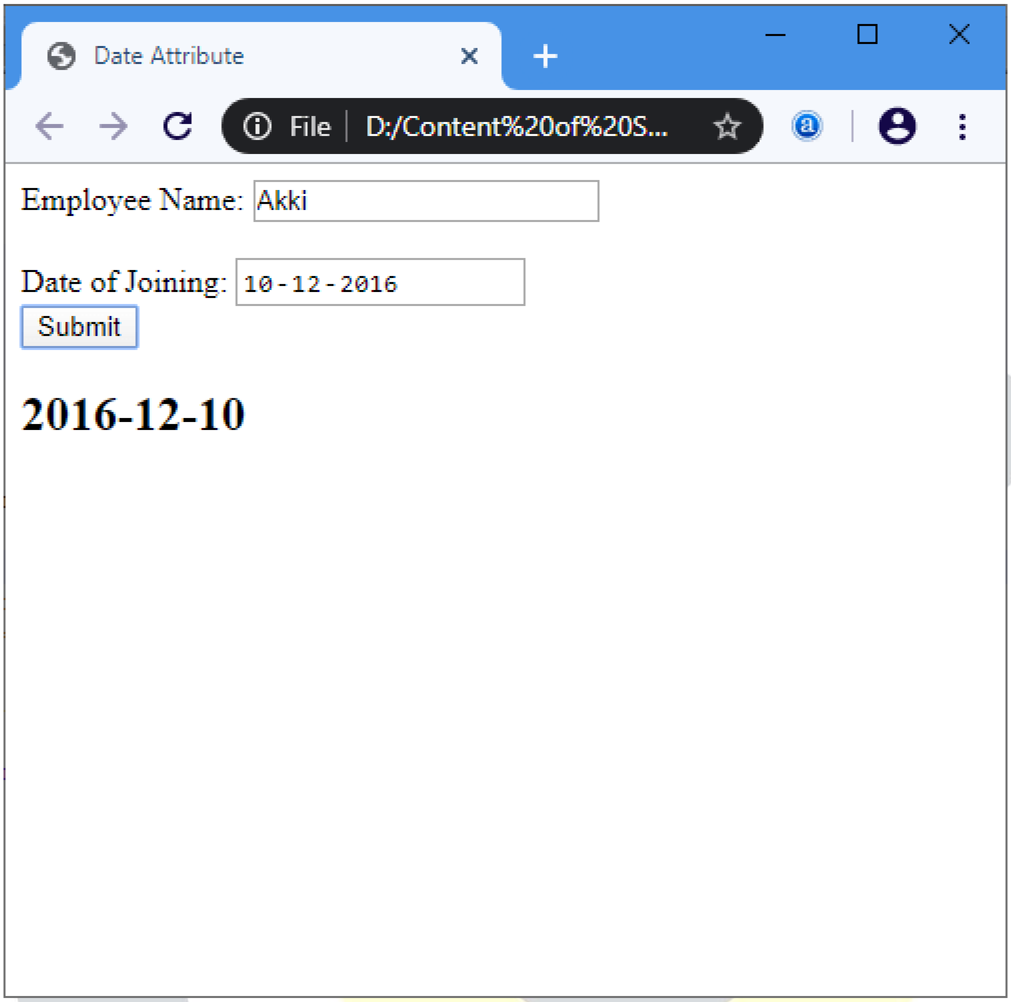
Bikes

Example 2: The following example describes how to use the ID attribute in JavaScript.

```
<!DOCTYPE html>
<html>
<head>
<title> Date Attribute </title>
<script>
function viewdate() {
var x = document.getElementById("dob").value;
document.getElementById("demo").innerHTML = x;
</script>
</head>
<body>
Employee Name: <input type="text" placeholder="Your Good name"/>
<br>
<br>
Date of Joining:
<input type="date" id="dob">
<br>
<button onclick="viewdate()"> Submit
</button>
<br>
<h2 id="demo"> </h2>
</body>
</html>
```

Test it Now

Output:



Employee Name: Akki

Date of Joining: 10-12-2016

2016-12-10

HTML iframes

HTML Iframe is used to display a nested webpage (a webpage within a webpage). The HTML <iframe> tag defines an inline frame, hence it is also called as an Inline frame.

An HTML iframe embeds another document within the current HTML document in the rectangular region.

The webpage content and iframe contents can interact with each other using JavaScript.

Iframe Syntax

An HTML iframe is defined with the <iframe> tag:

```
<iframe src="URL"></iframe>
```

Here, "src" attribute specifies the web address (URL) of the inline frame page.

Set Width and Height of iframe

You can set the width and height of iframe by using "width" and "height" attributes. By default, the attributes values are specified in pixels but you can also set them in percent. i.e. 50%, 60% etc.

Example: (Pixels)

```
<!DOCTYPE html>
<html>
<body>
<h2>HTML Iframes example</h2>
<p>Use the height and width attributes to specify the size of the iframe:</p>
<iframe src="https://www.javatpoint.com/" height="300" width="400"></iframe>
</body>
</html>
```

Test it Now

Example: (Percentage)

```
<!DOCTYPE html>
```

```
<html>
<body>
<h2>HTML Iframes</h2>
<p>You can use the height and width attributes to specify the size of the iframe:</p>
<iframe src="https://www.javatpoint.com/" height="50%" width="70%"></iframe>
</body>
</html>
```

You can also use CSS to set the height and width of the iframe.

Example:

```
<!DOCTYPE html>
<html>
<body>
<h2>HTML Iframes</h2>
<p>Use the CSS height and width properties to specify the size of the iframe:</p>
<iframe src="https://www.javatpoint.com/" style="height:300px;width:400px"></iframe>
</body>
</html>
```

Remove the border of iframe

By default, an iframe contains a border around it. You can remove the border by using <style> attribute and CSS border property.

Example:

```
<!DOCTYPE html>
<html>
<body>
<h2>Remove the Iframe Border</h2>
<p>This iframe example doesn't have any border</p>
```

```
<iframe src="https://www.javatpoint.com/" style="border:none;"></iframe>

</body>

</html>
```

You can also change the size, color, style of the iframe's border.

Example:

```
<!DOCTYPE html>

<html>

<body>

<h2>Custom Iframe Border</h2>

<iframe src="https://www.javatpoint.com/" style="border:2px solid tomato;"></iframe>

</body>

</html>
```

Iframe Target for a link

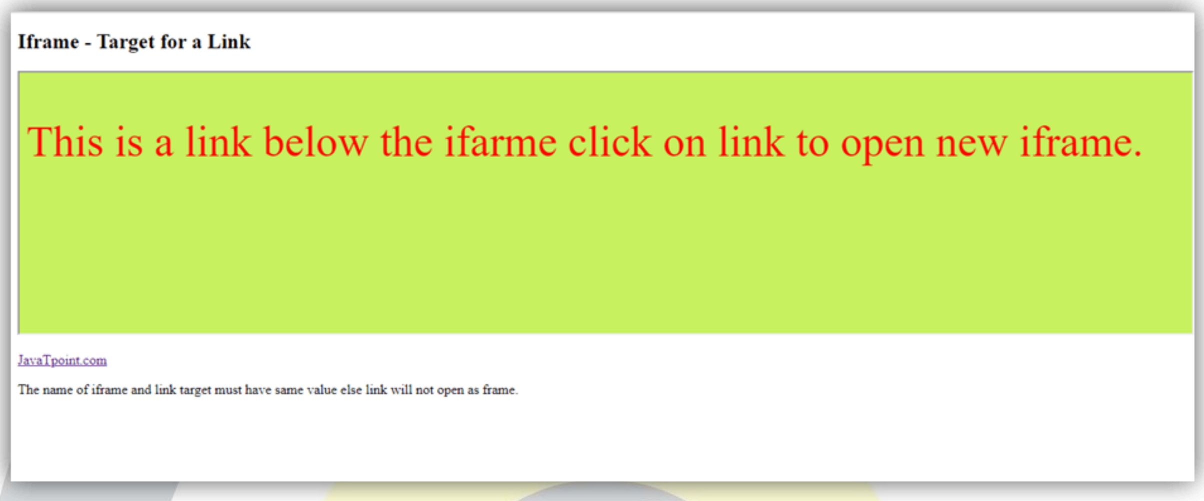
You can set a target frame for a link by using iframe. Your specified target attribute of the link must refer to the name attribute of the iframe.

Example:

```
<!DOCTYPE html>

<html>

<body>

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

<iframe height="300px" width="100%" src="new.html" name="iframe_a"></iframe>

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

<p>The name of iframe and link target must have same value else link will not open as a frame.
</p>


</body>
```

```
</html>
```

Output



Iframe - Target for a Link

This is a link below the ifarme click on link to open new iframe.

JavaTpoint.com

The name of iframe and link target must have same value else link will not open as frame.

new.hmtl output code:

```
<!DOCTYPE html>
<html>
<head>
        <style>
                p{ font-size: 50px;
              color: red;}
        </style>
</head>
<body style="background-color: #c7f15e;">
  <p>This is a link  below the ifarme click on link to open new iframe. </p>
</body>
</html>
```

Embed YouTube video using iframe

You can also add a YouTube video on your webpage using the <iframe> tag. The attached video will be played at your webpage and you can also set height, width, autoplay, and many more properties for the video.

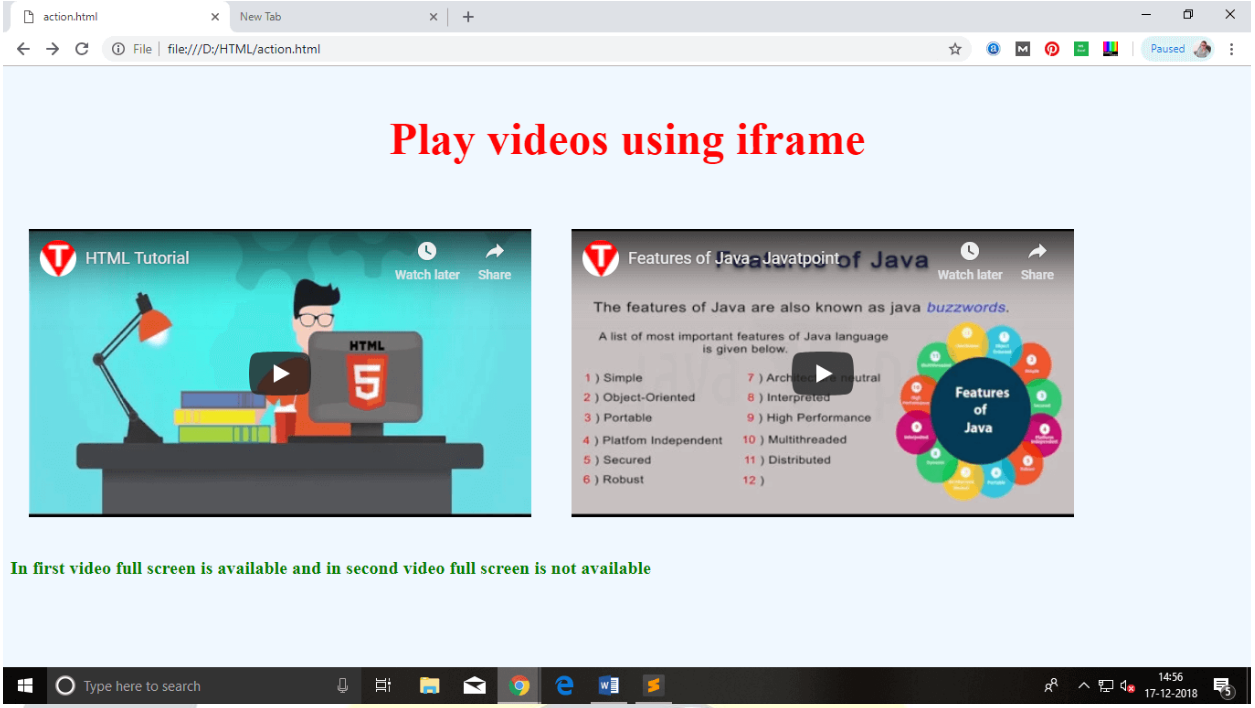
Following are some steps to add YouTube video on your webpage:

- Goto YouTube video which you want to embed.
- Click on SHARE ➦ under the video.
- Click on Embed < > option.
- Copy HTML code.
- Paste the code in your HTML file
- Change height, width, and other properties (as per requirement).

Example:

```
<iframe width="550" height="315" src="https://www.youtube.com/embed/JHq3pL4cdy4"
frameborder="0" allow="accelerometer; autoplay; encrypted-media; gyroscope; picture-in-
picture" allowfullscreen style="padding:20px;"></iframe>

        <iframe width="550" height="315" src="https://www.youtube.com/embed/O5hShUO6wxs"
frameborder="0" allow="accelerometer; autoplay; encrypted-media; gyroscope; picture-in-
picture" style="padding:20px;">></iframe>
```

Output:



Play videos using iframe

In first video full screen is available and in second video full screen is not available

Attributes of <iframe>

Attribute name	Value	Description
allowfullscreen		If true then that frame can be opened in full screen.
height	Pixels	It defines the height of the embedded iframe, and the default height is 150 px.
name	text	It gives the name to the iframe. The name attribute is important if you want to create a link in one frame.
frameborder	1 or 0	It defines whether iframe should have a border or not. (Not supported in HTML5).
Width	Pixels	It defines the width of embedded frame, and default width is 300 px.
src	URL	The src attribute is used to give the path name or file name which content to be loaded into iframe.

sandbox		
		This attribute is used to apply extra restrictions for the content of the frame
	allow-forms	It allows submission of the form if this keyword is not used then form submission is blocked.
	allow-popups	It will enable popups, and if not applied then no popup will open.
	allow-scripts	It will enable the script to run.
	allow-same-origin	If this keyword is used then the embedded resource will be treated as downloaded from the same source.
srcdoc		The srcdoc attribute is used to show the HTML content in the inline iframe. It overrides the src attribute (if a browser supports).
scrolling		
		It indicates that browser should provide a scroll bar for the iframe or not. (Not supported in HTML5)
	auto	Scrollbar only shows if the content of iframe is larger than its dimensions.
	yes	Always shows scroll bar for the iframe.
	no	Never shows scrollbar for the iframe.

HTML JavaScript

A Script is a small program which is used with HTML to make web pages more attractive, dynamic and interactive, such as an alert popup window on mouse click. Currently, the most popular scripting language is JavaScript used for websites.

Example:

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Date and Time example</h1>
<button type="button"
onclick="document.getElementById('demo').innerHTML = Date()">
Click me to display Date and Time.</button>
<p id="demo"></p>
</body>
</html>
```

HTML <script> Tag

The HTML <script> tag is used to specify a client-side script. It may be an internal or external JavaScript which contains scripting statements, hence we can place <script> tag within <body> or <head> section.

It is mainly used to manipulate images, form validation and change content dynamically. JavaScript uses document.getElementById() method to select an HTML element.

Example:

```
<!DOCTYPE html>
<html>
<body>
<h2>Use JavaScript to Change Text</h2>
```

```
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML = "Hello JavaTpoint";
</script>
</body>
</html>
```

HTML events with JavaScript

An event is something which user does, or browser does such as mouse click or page loading are examples of events, and JavaScript comes in the role if we want something to happen on these events.

HTML provides event handler attributes which work with JavaScript code and can perform some action on an event.

Syntax:

```
<element event = "JS code">
```

Example:

```
<input type="button" value="Click" onclick="alert('Hi, how are you')">
```

HTML can have following events such as:

- **Form events:** reset, submit, etc.
- **Select events:** text field, text area, etc.
- **Focus event:** focus, blur, etc.
- **Mouse events:** select, mouseup, mousemove, mousedown, click, dblclick, etc.

Following are the list for Window event attributes:

Event Event Name	Handler Name	Occurs when
onBlur	blur	When form input loses focus

onClick	click	When the user clicks on a form element or a link
onSubmit	submit	When user submits a form to the server.
onLoad	load	When page loads in a browser.
onFocus	focus	When user focuses on an input field.
onSelect	select	When user selects the form input filed.

Note: You will learn more about JavaScript Events in our JavaScript tutorial.

Let's see what JavaScript can do:

1) JavaScript can change HTML content.

Example:

```
<!DOCTYPE html>
<html>
<body>
<p>JavaScript can change the content of an HTML element:</p>
<button type="button" onclick="myFunction()">Click Me!</button>
<p id="demo"></p>
<script>
function myFunction() {
   document.getElementById("demo").innerHTML = "Hello JavaTpoint!";
}
</script>
</body>
</html>
```

2) JavaScript can change HTML style

Example:

```
<!DOCTYPE html>
<html>
<body>
<p id="demo">JavaScript can change the style of an HTML element.</p>
<script>
function myFunction() {
    document.getElementById("demo").style.fontSize = "25px";
    document.getElementById("demo").style.color = "brown";
    document.getElementById("demo").style.backgroundColor = "lightgreen";
}
</script>
<button type="button" onclick="myFunction()">Click Me!</button>
</body>
</html>
```

3) JavaScript can change HTML attributes.

Example:

```
<!DOCTYPE html>
<html>
<body>
<script>
function light(sw) {
    var pic;
    if (sw == 0) {
        pic = "pic_lightoff.png"
    } else {
        pic = "pic_lighton.png"
    }
    document.getElementById('myImage').src = pic;
}
</script>
<img id="myImage" src="pic_lightoff.png" width="100" height="180">
<p>
<button type="button" onclick="light(1)">Light On</button>
<button type="button" onclick="light(0)">Light Off</button>
</p>
</body>
</html>
```

Use External Script

Suppose, you have various HTML files which should have same script, then we can put our JavaScript code in separate file and can call in HTML file. Save JavaScript external files using .js extension.

Note: Do not add <script> tag in the external file, and provide the complete path where you have put the JS file.

Syntax:

```
<script type="text/javascript" src="URL "></script>
```

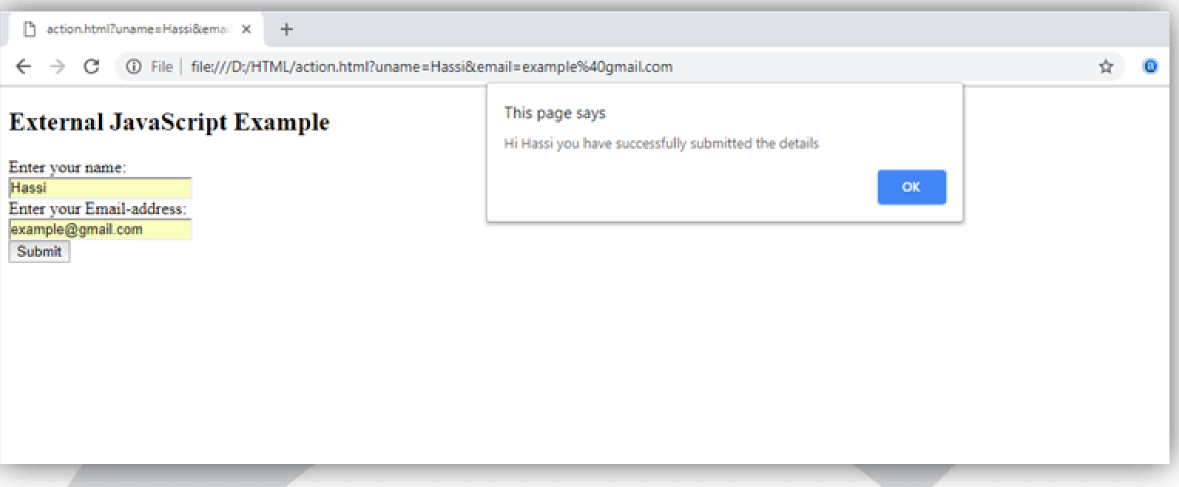
Example:

```
<!DOCTYPE html>
<html>
  <head>
    <script type="text/javascript" src="external.js"></script>
    </head>
     <body>
      <h2>External JavaScript Example</h2>
       <form onsubmit="fun()">
         <label>Enter your name:</label><br>
      <input type="text" name="uname" id="frm1"><br>
       <label>Enter your Email-address:</label><br>
      <input type="email" name="email"><br>
      <input type="submit">
  </form>
 </body>
 </html>
```

JavaScript code:

```
function fun() {
       var x = document.getElementById("frm1").value;
        alert("Hi"+" "+x+ "you have successfully submitted the details");
    }
```

Output:



file:///D:/HTML/action.html?uname=Hassi&email=example%40gmail.com

External JavaScript Example

Enter your name: Hassi

Enter your Email-address: example@gmail.com

This page says

Hi Hassi you have successfully submitted the details

HTML <noscript> Tag

HTML <noscript> tag is used to write disabled script in the browser. The text written within <noscript></noscript> tag is not displayed on the browser.

Example:

```
<!DOCTYPE html>
<html>
<body>
<p id="demo"></p>
<script>
document.getElementById("demo").innerHTML = "Hello JavaScript!";
</script>
<noscript>This text is not visible in the browser.</noscript>
</body>
</html>
```

HTML Comments

Comments are some text or code written in your code to give an explanation about the code, and not visible to the user. Comments which are used for HTML file are known as HTML comments. Anything written between these tags will be ignored by the browser, so comments will not be visible on the webpage.

Comments of any code make code easy to understand and increase readability of code.

Comments are also part of the code, which gives an explanation of the code.

How to add comment in HTML

You can add comments in your HTML file using <! -- ... --> tag. So, if you will write anything between theses comment tag that will be treated as comment and browser will not read it.

Syntax

```
<! -- Write commented text here -->
```

Note: The commented code will not be visible to a webpage, and hence you can use comment tag for documentation purpose, and debugging purpose:

Such as:

```
<!--   <p>There is some text</p>
       <p>There is second text</p> -->
```

Example:

```
<!DOCTYPE html>
<html>
<!-- This is Header section -->
<head>
   <!-- Internal CSS -->
   <style>
      body{
         text-align: center;
```

```
            background-color: #f0f8ff;
            font-size: 30px;
            color: red;
        }
    </style>
</head>
<!-- This is body section, write code here which you want to display on web-page -->
<body>
    <!-- heading tag -->
 <h2>First WebPage</h2>
 <!-- Paragraph tag -->
 <p>Write your Content here!!!</p>
</body>
</html>
```

Multiline Comment

In HTML code, we can also comments multiple lines at a time. In multiline comment we can use any description about code or multiple line code to debug, etc.

Syntax

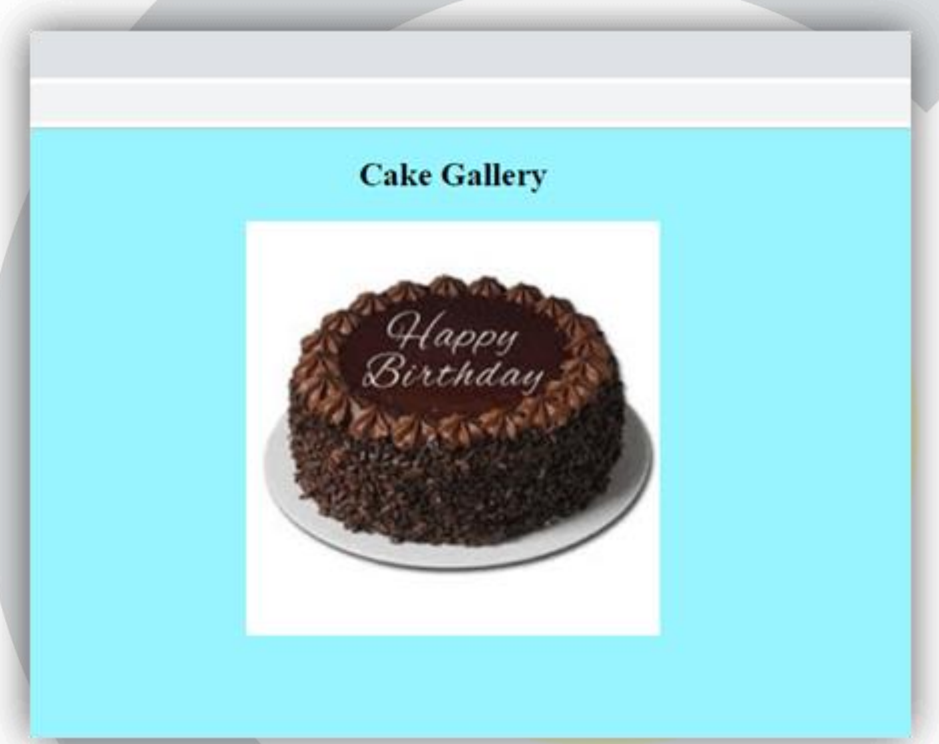
```
<!---
Your code is commented.
Write description of code.
It will not display at webpage.
-->
```

Example:

```
 <h2>Cake Gallery</h2>
<!-- This is image for a yummy cake
```

```
you can see it on your web-page

of your favorite browser -->

<img src="https://static.javatpoint.com/htmlpages/images/cake.png" alt="cake image"
height="300px"

 width="300px">
```

Output:



Cake Gallery

HTML File Paths

An HTML file path is used to describe the location of a file in a website folder. File paths are like an address of file for a web browser. We can link any external resource to add in our HTML file with the help of file paths such as images, file, CSS file, JS file, video, etc.

The src or href attribute requires an attribute to link any external source to HTML file.

Following are the different types to specify file paths:

1. <img src="picture.jpg"> It specifies that picture.jpg is located in the same folder as the current page.
2. <img src="images/picture.jpg"> It specifies that picture.jpg is located in the images folder in the current folder.
3. <img src="/images/picture.jpg"> It specifies that picture.jpg is located in the images folder at the root of the current web.
4. <img src="../picture.jpg"> It specifies that picture.jpg is located in the folder one level up from the current folder.

File paths are used on webpages to link external files like:

1. Web pages
2. Images
3. Style sheets
4. JavaScript

There are two types of File Paths:

1. Absolute File Paths
2. Relative File Paths

Absolute File Paths

Absolute file path specifies full URL address.

Example:

<!DOCTYPE html>

<html>

```
<body>
<h2>Using a Full URL File Path</h2>
<img src="https://www.javatpoint.com/images/nature-1.jpg" alt="image" style="width:300px">
</body>
</html>
```

Relative File Paths

The relative file path specifies to a file which is related to the location of current page.

Example:

Let's take an example to see how the file path points to a file in the images folder located at the root of the current web.

```
<!DOCTYPE html>
<html>
<body>
<h2>Using a Relative File Path</h2>
<img src="/images/nature-2.jpg" alt="Mountain" style="width:300px">
</body>
</html>
```

Example:

This is how a file path points to a file in the images folder located in the current folder.

```
<!DOCTYPE html>
<html>
<body>
<h2>Using a Relative File Path</h2>
<img src="images/nature-3.jpg" alt="Mountain" style="width:300px">
```

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

Example:

When the images folder located in the folder one level above the current folder.

```
<!DOCTYPE html>
<html>
<body>
<h2>Using a Relative File Path</h2>
<img src="../images/nature4.jpg" alt="Mountain" style="width:300px">
</body>
</html>
```

Important Points for File path:

Always remember to use proper URL, file name, image name, else it will not display on the webpage.

Try to use relative file paths, so that your code will be independent of URL.

HTML Head

The HTML <head> element is used as a container for metadata (data about data). It is used between <html> tag and <body> tag.

The head of an HTML document is a part whose content is not displayed in the browser on page loading. It just contains metadata about the HTML document which specifies data about the HTML document.

An HTML head can contain lots of metadata information or can have very less or no information, it depends on our requirement. But head part has a crucial role an HTML document while creating a website.

Metadata defines the document title, character set, styles, links, scripts, and other meta information.

Following is a list of tags used in metadata:

1) <title>
2) <style>
3) <meta>
4) <link>
5) <script>
6) <base>

HTML <title> Element

The HTML <title> element is used to define the title of the document. It is used in all HTML/XHTML documents. The <title> element must be placed between <head> element, and one document can only have one title element.

What does <title> element do?

1. It defines a title in the browser tab.
2. It provides a title for the page when it is added to favorites.
3. It displays a title for the page in search engine results.

Note: The title element must be specific about the document and its recommended length is 65 to 70 characters including spaces.

Example:

```
<!DOCTYPE html>

<html>

<head>

  <title>This Page Title</title>

</head>

<body>

<p>The body's content is displayed in the browser window.</p>

<p>The content of the title element is displayed in the browser tab, in favorites and in search engine results.</p>

</body>

</html>
```

HTML <style> Element

The HTML <style> element is used to style the HTML page. The <style> element can have CSS properties for that HTML page only. If we want to apply CSS for multiple pages then we should use separate CSS file.

Example:

```
<!DOCTYPE html>

<html>

<head>

  <title>This is Page Title</title>

  <style>

    body {background-color: pink;}

    h1 {color: red;}

    p {color: blue;}

  </style>

</head>
```

```
<body>
<h1>This is a Heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```

HTML <link> Element

The HTML <link> element is used to link an external style sheet to your webpage. The <link> element contains main two attributes which are "rel" and "href". The rel attribute indicates that it is a stylesheet, and href gives the path to that external file.

Example:

```
<!DOCTYPE html>
 <html>
 <head>
    <title>This is title</title>
    <link rel="stylesheet" href="style.css">
   </head>
   <body>
    <h2>Web-page with external CSS</h2>
    <p>This is looking a cool page</p>
   </body>
 </html>
```

HTML <meta> Element

The HTML <meta> element is used to specify the character set, page description, keywords, authors and other metadata on the webpage.

Metadata is mainly used by browsers, search engines, and other web services to rank your webpage better.

Let's see how to use metadata:

To define a character set:

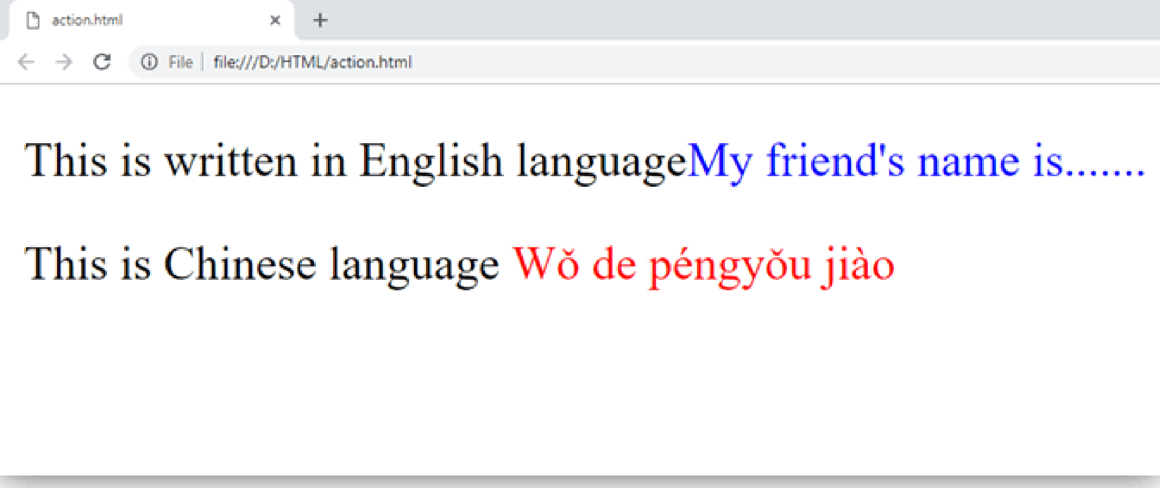
```
<meta charset="UTF-8">
```

The charset attribute specifies the character encoding. In this example we have set it to "UTF-8" which means it can handle to display any language.

Example:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
</head>
<body>
<p>This is written in English language<span style="color: blue"> My friend's name is.......</span></p>
<p>This is Chinese language <span style="color: red">Wǒ de péngyǒu jiào</span></p>
</body>
</html>
```

Output:



This is written in English languageMy friend's name is.......

This is Chinese language Wǒ de péngyǒu jiào

To define a description of your webpage:

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

If you give a meta description then it will be useful for the relevant search to perform by search engines.

To define keywords for search engines:

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

The keyword value is also used to provide keywords for a search engine, but it may ignore by browser due to spammers.

To define author of the webpage:

```
<meta name="author" content="Akon">
```

The author value specifies the name of the person who wrote the page content, and it is useful to automatically extract author information by some content management systems.

To refresh document every 30 seconds:

```
<meta http-equiv="refresh" content="30">
```

Meta refresh is used to provide instructions to the browser to automatically refresh the page after the given time interval. As in above example it will automatically refresh after 30 sec

```
<meta http-equiv="refresh" content="10; url=https://www.javatpoint.com/html-head>
```

If you add an URL with content value, then it will redirect to that page after the time limit will over.

Example:

```
<!DOCTYPE html>
<html>
 <head>
   <meta http-equiv="refresh" content="5; url=https://www.javatpoint.com/html-head">
 </head>
```

```
  <body>
   <h2>Meta element Example</h2>
  <p style="color: green;">Kindly wait for 5 seconds and after 5 seconds it will automatically redirect to URL specified in meta tag</p>
 </body>
</html>
```

Following is an example to show how to use all Meta elements within HTML head

Example:

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<meta name="description" content="Free Web tutorials">
<meta name="keywords" content="HTML,CSS,XML,JavaScript">
<meta name="author" content="Akon">
</head>
<body>
<p>All the meta information are set.</p>
</body>
</html>
```

Use <meta> tag to set the Viewport

This method is introduced in HTML5 to take control over the viewport by using <meta> tag.

Viewport is the user's visible area of a webpage. It changes from device to device and appears smaller on mobile phones than computer screens.

Syntax for <meta> viewport element:

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

Here, the <meta> viewport element specifies how to control the page's dimensions and scaling.

The width=device-width is used to set the width of the page to follow the screen-width of the device (which will vary depending on the device).

The initial-scale=1.0 is used to set the initial zoom level when the page is first loaded by the browser.

Example of a web page without the viewport <meta> tag:

```
<!DOCTYPE html>
<html>
<body>
<p><b>To understand this example, you should open this page on a phone or a tablet.</b></p>
<img src="image.jpg" alt="image" width="460" height="345">
<p>
Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut
laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation
ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel
eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu
feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum
zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis
eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum.
Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat
facer possim assum.
</p>
</body>
</html>
```

Example of a web page with the viewport <meta> tag:

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0"/>
<style>
img {
   max-width: 100%;
   height: auto;
}
</style>
</head>
<body>
<p><b>To understand this example, you should open this page on a phone or a tablet.</b></p>
<img src="image.jpg" alt="image" width="460" height="345">
<p>
Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut
laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation
ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel
eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu
feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum
zzril delenit augue duis dolore te feugait nulla facilisi. Nam liber tempor cum soluta nobis
eleifend option congue nihil imperdiet doming id quod mazim placerat facer possim assum.
Nam liber tempor cum soluta nobis eleifend option congue nihil imperdiet doming id quod mazim placerat
facer possim assum.
</p>
```

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

Note: To see the difference clearly, open this page on smartphone or tablet.

HTML <base> Element

The HTML <base> element is used to specify the base URL and base target for all relative URLs in a page.

Example:

```
<!DOCTYPE html>
<html>
<head>
<title>Page Title</title>
<base href="https://static.javatpoint.com/htmlpages/images/" target="_blank">
</head>
<body>
<img src="html5.png">
<p>We have specified a base URL, the browser will look for the image "html5.png"
at "https://static.javatpoint.com/htmlpages/images/html5.png"</p>
<p><a href=" https://www.javatpoint.com">JavatPoint</a></p>
<p>The link above will open in a new window because base target is set to "_blank".</p>
</body>
</html>
```

HTML <script> element

HTML <script> element is used to apply client side JavaScript for the same page or to add an external JavaScript file to current page.

Example:

```
<!DOCTYPE html>
<html>
<head>
   <script>
      function fun() {
      document.getElementById("p").style.color="green";
      }
   </script>
</head>
<body>
<h2>Script within Head element</h2>
<p id="p">This will change the color</p>
<button type="button" onclick="fun()">Click me</button>
</body>
</html>
```

If we want to use some external JavaScript file then it can be applied by:

```
<script src=".js file_path">
```

Excluding <html>, <head> and <body> elements

HTML 5 facilitates us to omit the <html>, the <body>, and the <head> tag.

Example:

```
<!DOCTYPE html>
<title>Page Title</title>
```

```
<h1>This is a heading</h1>

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

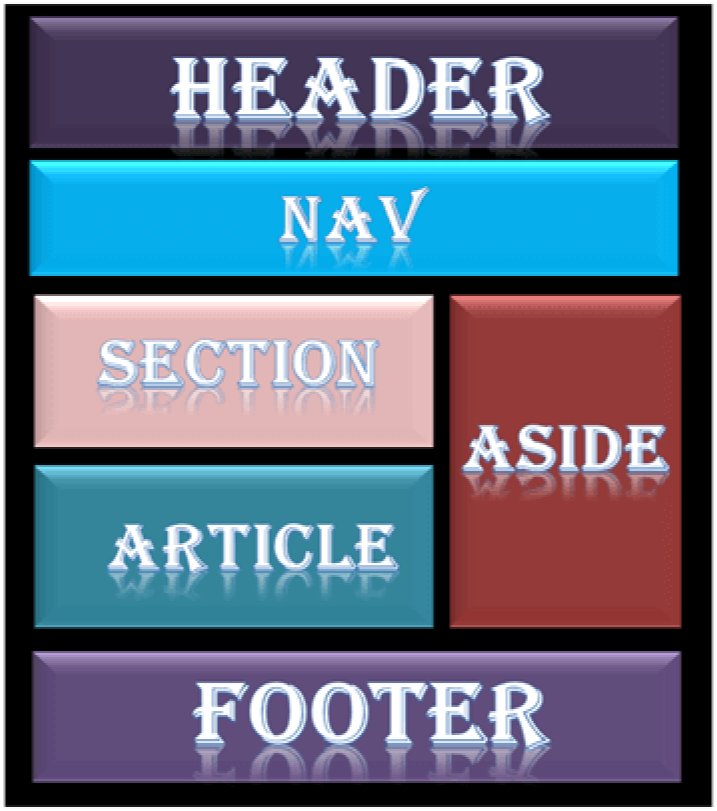
Note: It is not recommended to omit the <html> and <body> tags. Omitting these tags can crash DOM or XML software and produce errors in older browsers (IE9).

However, you can omit the <head> tag.

HTML Layouts

HTML layouts provide a way to arrange web pages in well-mannered, well-structured, and in responsive form or we can say that HTML layout specifies a way in which the web pages can be arranged. Web-page layout works with arrangement of visual elements of an HTML document.

Web page layout is the most important part to keep in mind while creating a website so that our website can appear professional with the great look. You can also use CSS and JAVASCRIPT based frameworks for creating layouts for responsive and dynamic website designing.



Every website has a specific layout to display content in a specific manner.

Following are different HTML5 elements which are used to define the different parts of a webpage.

1) <header>: It is used to define a header for a document or a section.
2) <nav>: It is used to define a container for navigation links
3) <section>: It is used to define a section in a document
4) <article>: It is used to define an independent self-contained article
5) <aside>: It is used to define content aside from the content (like a sidebar)
6) <footer>: It is used to define a footer for a document or a section

7) <details>: It is used to define additional details
8) <summary>: It is used to define a heading for the <details> element

NOTE: HTML layouts create an individual space for every part of the web page. So that every element can arrange in a significant order.

Description of various Layout elements

HTML <header>

The <header> element is used to create header section of web pages. The header contains the introductory content, heading element, logo or icon for the webpage, and authorship information.

Example:

```
<header style="background-color: #303030;  height: 80px; width: 100%">
    <h1 style="font-size: 30px; color: white;text-align: center; padding-top: 15px;">Welcome to MyFirstWebpage</h1>
  </header>
```

HTML <nav>

The <nav> elements is a container for the main block of navigation links. It can contain links for the same page or for other pages.

Example:

```
<nav style="background-color:#bcdeef;">
      <h1 style="text-align: center;">Navgation Links</h1>
      <ul>
         <li><a href="#">link1</a></li>
         <li><a href="#">link2</a></li>
         <li><a href="#">link3</a></li>
         <li><a href="#">link4</a></li>
      </ul>

    </nav>
```

HTML <section>

HTML <section> elements represent a separate section of a web page which contains related element grouped together. It can contain: text, images, tables, videos, etc.

Example:

```
<section style="background-color:#ff7f50; width: 100%; border: 1px solid black;">

    <h2>Introduction to HTML</h2>

    <p>HTML is a markup language which is used for creating attractive web pages with the help of styling, and which looks in a nice format on a web browser..</p>

  </section>
```

HTML <article>

The HTML tag is used to contain a self-contained article such as big story, huge article, etc.

Example:

```
<article style="width: 100%; border:2px solid black; background-color: #fff0f5;">

    <h2>History of Computer</h2>

    <p>Write your content here for the history of computer</p>

</article>
```

HTML <aside>

HTML <aside> define aside content related to primary content. The <aside> content must be related to the primary content. It can function as side bar for the main content of web page.

Example:

```
<aside style="background-color:#e6e6fa">

    <h2>Sidebar information</h2>
    <p>This conatins information which will represent like a side bar for a webpage</p>
  </aside>
```

HTML <footer>

HTML <footer> element defines the footer for that document or web page. It mostly contains information about author, copyright, other links, etc.

Example:

```
<footer style="background-color: #f0f8ff; width: 100%; text-align: center;">
    <h3>Footer Example</h3>
    <p>© Copyright 2018-2020. </p>
</footer>
```

HTML <details>

HTML <details> element is used to add extra details about the web page and use can hide or show the details as per requirement.

Example:

```
<details style="background-color: #f5deb3">
    <summary>This is visible section: click to show other details</summary>
    <p>This section only shows if user want to see it. </p>
 </details>
```

HTML <summary>

HTML <summary> element is used with the <details> element in a web page. It is used as summary, captions about the content of <details> element.

Example:

```
<details>
```

```
  <summary>HTML is acronym for?</summary>
  <p style="color: blue; font-size: 20px;">Hypertext Markup Language</p>
</details>
```

HTML Layout Techniques

Creating layouts are the most important things while designing a website, as it will ensure that your website looks in a well-arranged way and the content appears easy to understand. There are various techniques, and frameworks available for creating layouts, but here we will learn about simple techniques. You can use the following methods to create multicolumn layouts:

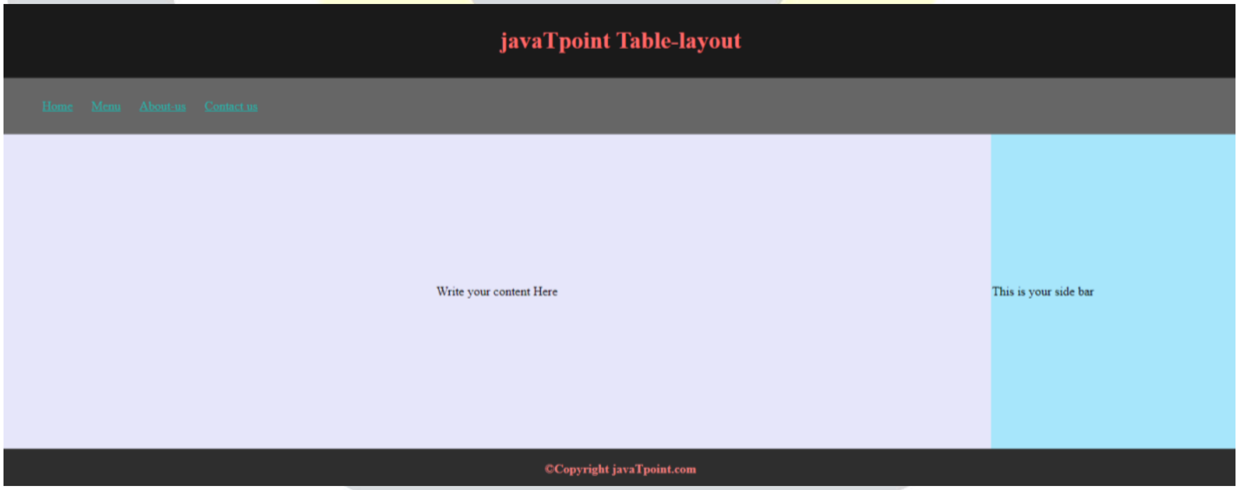
1) HTML tables (Try not to use)
2) CSS float property
3) CSS framework
4) CSS flexbox
5) Layout using div

HTML Tables (Not Recommended)

HTML table-based layout is one of the easiest ways for creating a layout, as table use only rows and column-based format, but HTML tables are not recommended for your page layout.

The element is designed to display tabular data. It is not good for a layout. Although first creating a layout is easy, but if you want to change or redesign your website, then it will be a complicated task.

Following is an example for the creation of a simple web page layout using HTML table.



javaTpoint Table-layout

Write your content Here

This is your side bar

©Copyright javaTpoint.com

Example:

```
<!DOCTYPE html>

<html>

 <head>

    <style>
```

```
        li{
          display: inline-block;
          padding: 10px;}
        a{
          color:#20b2aa;
        }
      </style>
  </head>
<body>
    <!-- Header Section -->
      <table width="100%" style="border-collapse:collapse;">
         <tr>
          <td colspan="2" style="background-color:#1a1a1a; text-align: center;">
            <h3 style="font-size: 30px; color: #ff6a6a;">javaTpoint Table-layout</h3>
          </td>
        </tr>
   <!-- Nav Section -->
        <tr>
                    <td colspan="2" style="background-color:#666666;">
            <ul>
                                        <li><a href="#">Home</a></li>
            <li><a href="#">Menu</a></li>
            <li><a href="#">About-us</a></li>
            <li><a href="#">Contact us</a></li>
          </ul>
                                  </td>
        </tr>
   <!-- Main Section -->
```

```
          <tr>
             <td style="background-color:#e6e6fa; width:80%; height: 400px; text-align: center;">
              <p>Write your content Here</p>
             </td>
             <td style="background-color:#a7e6fb; height: 400px;">
               <p>This is your side bar</p>
             </td>
          </tr>
      <!-- Footer Section -->
           <tr>
             <td colspan="2" style="background-color:#2e2e2e; text-align: center;">
               <p style="color:#f08080">©<strong>Copyright javaTpoint.com</strong></p>
            </td>
          </tr>
     </table>
   </body>
</html>
```

Note: This example is just for show you how to create layout using table but it's not recommended to use table layout.

CSS Frameworks

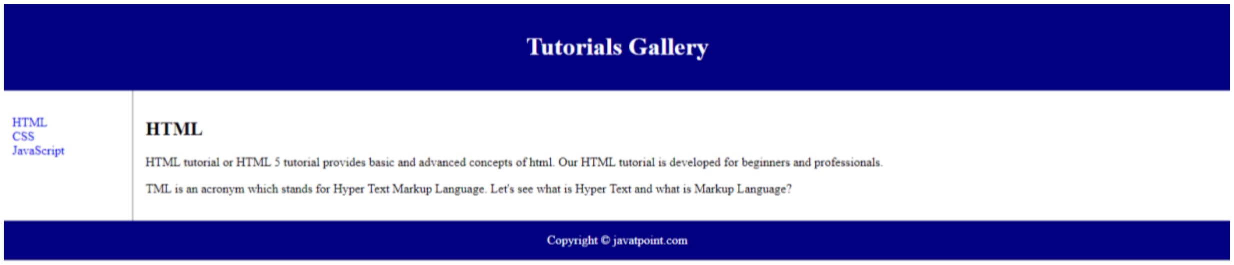
CSS provides many frameworks like W3.CSS, Bootstrap, and many more, to create your layout fast. Using CSS frameworks you can easily create a responsive and attractive web layout. You just need to add a link for these frameworks, and you can use all properties available in the framework.

CSS Float

You can create an entire web layout using CSS float property.

Advantage: It is very easy to learn and use. You just learn how the float and clear properties work.

Disadvantage: Floating elements are tied to the document flow, which may harm the flexibility.



Tutorials Gallery

HTML
CSS
JavaScript

HTML

HTML tutorial or HTML 5 tutorial provides basic and advanced concepts of html. Our HTML tutorial is developed for beginners and professionals.

TML is an acronym which stands for Hyper Text Markup Language. Let's see what is Hyper Text and what is Markup Language?

Copyright © javatpoint.com

Example:

```
<!DOCTYPE html>
<html>
<head>
<style>
div.container {
   width: 100%;
   border: 1px solid gray;
}

header, footer {
   padding: 1em;
   color: white;
   background-color: #000080;
   clear: left;
   text-align: center;
}

nav {
   float: left;
   max-width: 160px;
   margin: 0;
   padding: 1em;
}

nav ul {
   list-style-type: none;
   padding: 0;
}

nav ul a {
   text-decoration: none;
}

article {
   margin-left: 170px;
   border-left: 1px solid gray;
   padding: 1em;
   overflow: hidden;
}
</style>
</head>
<body>
```

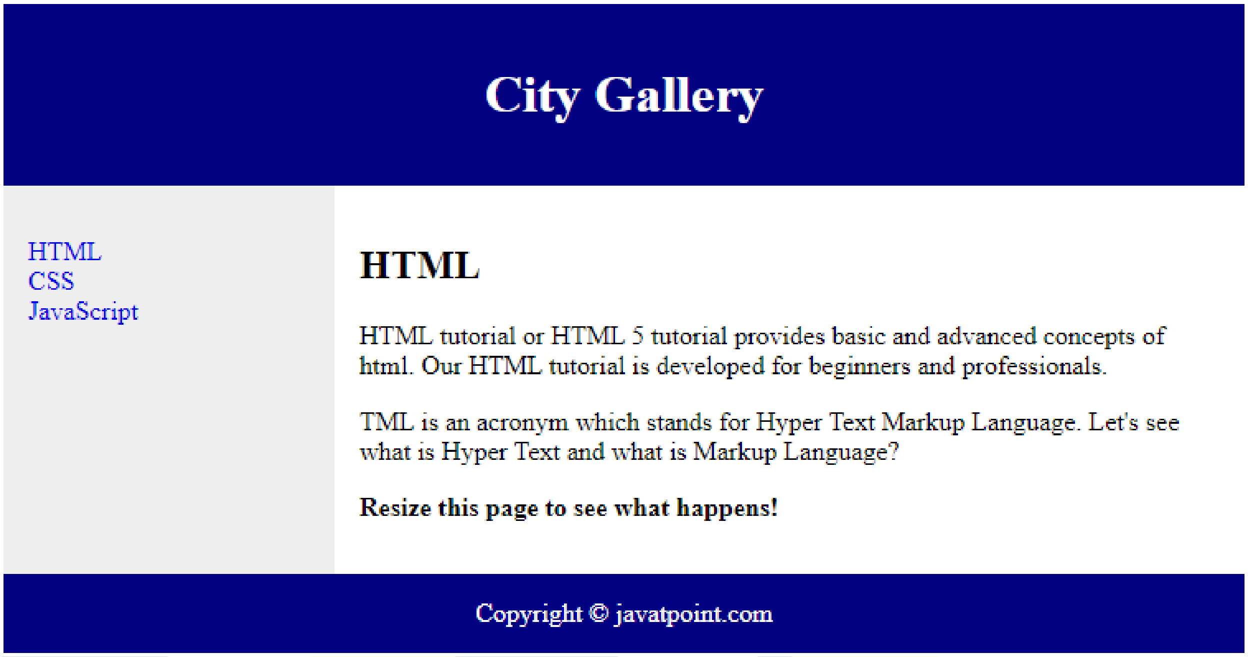
```
<div class="container">


<header>

<h1>Tutorials Gallery</h1>

</header>


<nav>

<ul>

<li><a href="#">HTML</a></li>

<li><a href="#">CSS</a></li>

<li><a href="#">JavaScript</a></li>

</ul>

</nav>

<article>

<h1>HTML</h1>

<p>HTML tutorial or HTML 5 tutorial provides basic and advanced concepts of html. Our
HTML tutorial is

developed for beginners and professionals.</p>

<p>TML is an acronym which stands for Hyper Text Markup Language. Let's see what is Hyper
Text and what is Markup Language?</p>

</article>

<footer>Copyright © javatpoint.com</footer>

</div>

</body>

</html>
```

CSS Flexbox

Flexbox is a new layout mode in CSS3.

Advantage: It ensures that the page layout must accommodate different screen sizes and different display devices.

Disadvantages: It does not work in IE10 and earlier.



City Gallery

HTML
CSS
JavaScript

HTML

HTML tutorial or HTML 5 tutorial provides basic and advanced concepts of html. Our HTML tutorial is developed for beginners and professionals.

TML is an acronym which stands for Hyper Text Markup Language. Let's see what is Hyper Text and what is Markup Language?

Resize this page to see what happens!

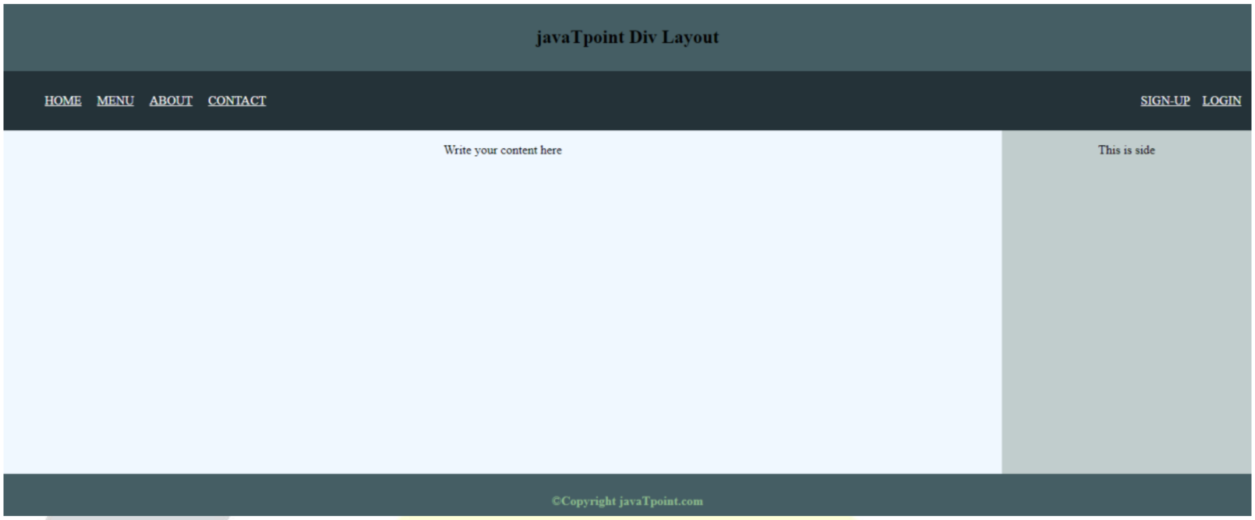
Copyright © javatpoint.com

Example:

```
<!DOCTYPE html>

<html>

<head>

<style>

.flex-container {

   display: -webkit-flex;

   display: flex;

   -webkit-flex-flow: row wrap;

   flex-flow: row wrap;

   text-align: center;

}


.flex-container > * {
    padding: 15px;
    -webkit-flex: 1 100%;
    flex: 1 100%;
}

.article {
    text-align: left;
}

header {background: #000080;color:white;}
footer {background: #000080;color:white;}
.nav {background:#eee;}

.nav ul {
    list-style-type: none;
    padding: 0;
}
.nav ul a {
    text-decoration: none;
}

@media all and (min-width: 768px) {
    .nav {text-align:left;-webkit-flex: 1 auto;flex:1 auto;-webkit-order:1;order:1;}
    .article {-webkit-flex:5 0px;flex:5 0px;-webkit-order:2;order:2;}
    footer {-webkit-order:3;order:3;}
}
</style>
</head>
```

```
<body>

<div class="flex-container">
<header>
<h1>City Gallery</h1>
</header>

<nav class="nav">
<ul>
<li><a href="#">HTML</a></li>
<li><a href="#">CSS</a></li>
<li><a href="#">JavaScript</a></li>
</ul>
</nav>

<article class="article">
<h1>HTML</h1>
<p>HTML tutorial or HTML 5 tutorial provides basic and advanced concepts of html. Our
HTML tutorial is
developed for beginners and professionals.</p>
<p>TML is an acronym which stands for Hyper Text Markup Language. Let's see what is Hyper
Text and what is Markup Language?</p>
<p><strong>Resize this page to see what happens!</strong></p>
</article>

<footer>Copyright © javatpoint.com</footer>
</div>
  </body>
</html>
```

Layout using div



javaTpoint Div Layout

Write your content here

This is side

©Copyright javaTpoint.com

```
<!DOCTYPE html>
<html>
<head>
  <title>Webpage using div</title>
  <style>
    body{
      margin:0px;
    }
    .header{
        padding: 10px;
        background-color:#455e64;
        text-align: center;
       }
     .header h2{
      color: black; }
       /*===============[Nav CSS]==========*/
    .nav{
      background-color:#243238;
      padding: 5px;
      }

    .nav li{
    list-style: none;
    display: inline-block;
    padding: 8px;
     }
 .nav a{
  color: #fff;
 }

.nav ul li a:hover{
  text-decoration: none;
  color: #7fffd4;
 }
    .lside{
       float: left;
     width: 80%;
     min-height: 440px;
     background-color: #f0f8ff;
     text-align: center;
    }
 .rside
 {
  text-align: center;
```

```
      float: right;
      width: 20%;
       min-height: 440px;
      background-color:  #c1cdcd;
      }
      .footer{
      height: 44px;
        background-color:#455e64;
      text-align: center;
      padding-top: 10px;}

      .footer p{
        color:  #8fbc8f;
      }

  </style>
</head>
<body>
<div>
  <div class="header">
   <h2>javaTpoint Div Layout</h2>
  </div>
      <!-- Nav -->
    <div class="nav">
      <ul>
        <li><a href="#">HOME</a></li>
        <li><a href="#">MENU</a></li>
        <li><a href="#">ABOUT</a></li>
```

```
                <li><a href="#">CONTACT</a></li>
                <li style="float: right;"><a href="#">LOGIN</a></li>
                <li style="float: right;"><a href="#">SIGN-UP</a></li>
            </ul>
        </div>

      <!-- main -->
        <div style="height:440px">
            <div class="lside">
                <p>Write your content here</p>
            </div>
        <!-- side -->
            <div class="rside">
                <p>This is side</p>
            </div>
        </div>
      <!-- footer -->
       <div class="footer">
        <p>©<strong>Copyright javaTpoint.com</strong></p>
    </div>
 </div>
</body>
</html>
```

HTML Responsive

Responsive Web design

Responsive web design is used to make your web page look appropriate, good, and well placedon all devices (desktop, tablet, smartphone etc.)

Responsive web design uses HTML and CSS to resize, hide, shrink, enlarge, or move the content. It makes the content look good on any screen.

Responsive Images

Images which can be scaled nicely to fit any browser size are known as responsive images.

How to make Image Responsive?

1) By using the width property

Set the CSS width property to 100% to make the image responsive and scale up and down.

Example

```
<!DOCTYPE html>

<html>

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

<body>

<h2>Responsive Image</h2>

<p>When we set the CSS width property to 100%, it makes the image responsive.

Resize the browser window to see the effect.</p>

<img src="img_girl.jpg" style="width:100%;">( change image)

</body>

</html>
```

Note: A problem with the above method (width: 100%) is that the image can be scaled up to be larger than its original size. So, it is better to use the max-width property instead.

2) By using the max-width Property

This method is best and most used because it facilitates that the image will scale down if it has to, but never scale up to be larger than its original size.

Example

```
<!DOCTYPE html>
<html>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<body>
<h2>Responsive Image</h2>
<p>"max-width:100%" makes the image responsive and also ensures that the image
doesn't get bigger than its original size.</p>
<p>Resize the browser window to see the effect.</p>
<img src="img_girl.jpg" style="max-width:100%;height:auto;"> (Change the image)
</body>
</html>
```

3) Change images according to the browser width

By using the HTML <picture> element, you can set two or more imagesaccording to the browser width. It will change the picture when you change the browser-size. i.e. desktop and phone.

Example

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

```
<h2>Change Images Depending on Browser Width</h2>

<p>Resize the browser width and the image will change at 600px and 1500px.</p>

<picture>

<source srcset="img_smallflower.jpg" media="(max-width: 600px)">(Change image)

<source srcset="img_flowers.jpg" media="(max-width: 1500px)">(Change image)

<source srcset="flowers.jpg">

<img src="img_flowers.jpg" alt="Flowers" style="width:auto;">

</picture>

</body>

</html>
```

Responsive Text-size

We can make the text size responsive by using the "uv" unit. It means viewport-width. By using this, we can make the text size to follow the browserwindow screen.

Example

```
<!DOCTYPE html>

<html>

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

<body>

<h1 style="font-size:10vw;">Here size is 10vw.</h1>

<p style="font-size:6vw;">Here size is 6vw.</p>

<p style="font-size:4vw;">Here size is 4vw.</p>

<p>Resize the browser window to see how the text size changes.</p>

</body>

</html>
```

Note: viewport specifies the browser window size. 1vw = 1% of viewport width. Means, if the viewport is 100cm wide, 1vw is 1.0cm.

Media Query

We can also use media query to make responsive websites. Read the details of media query from here: Media Query

HTML Computer code

When we are programming, sometimes it is mandatory to show the Output result, error message, or coding part to user on a webpage. Hence to solve this issue HTML uses different tags for the user inputs, codes, programs, etc. With the help of these tags, you will be able to write codes to display on your webpage.

Following is a list of some tags which are used in HTML for this task.

1) <code>
2) <kbd>
3) <samp>
4) <var>
5) <pre>

HTML <code> element

It is used to represent some programming code on your website. The content written between tag will be displayed in default monospace font.

Example:

```
<!DOCTYPE html>

<html>

<body>

<h2>Computer Code</h2>

<p>This is a programming code:</p>

<code>

x = 5;<br>

y = 6;<br>

z = x + y;

</code>

</body>

</html>
```

HTML <kbd> Element

It is used to represent user input, keyboard input, voice command etc. Text written within <kbd>.....</kbd> tags is typically displayed in the browser's default monospace font.

Example:

```
<!DOCTYPE html>
<html>
<body>
<h2>The kbd Element</h2>
<kbd>This is how  content written within kbd element looks like.</kbd></p>
</body>
</html>
```

HTML <samp> Element

The HTML <samp> element is used to represent a program's output. Text written within samp element is typically displayed in the browser's default monospace font.

Example:

```
<!DOCTYPE html>
<html>
<body>
<h2>The samp Element</h2>
<samp>This is how the content within samp element looks like. </samp>
</body>
</html>
```

HTML <var> element

The HTML <var> element is used to define a variable. The variable could be a variable in a mathematical expression or a variable in programming context.

Example:

```
<!DOCTYPE html>

<html>

<body>

<h2>The var Element</h2>

<p>This is a famous formula: <var>E</var> = <var>mc</var><sup>2</sup>.</p>

</body>

</html>
```

HTML <pre> element

The <pre> element defines preformatted text, which displays the content within it in a fixed-width font. It keeps the content into its original format and ignores all formatting.

Example:

```
<!DOCTYPE html>

<html>

<body>

<h3>Example of pre tag</h3>

  <pre>

    This is content written

    within pre tag, and  pre tag will ignore all

    spaces,     break lines, and will display the content

    as in original format.

  </pre>

 </body>

</html>
```

HTML Entities

HTML character entities are used as a replacement of reserved characters in HTML. You can also replace characters that are not present on your keyboard by entities.

These characters are replaced because some characters are reserved in HTML. HTML entities provide a wide range of characters which can allow you to add icons, geometric shapes, mathematical operators, etc.

For example: if you use less than (<) or greater than (>) symbols in your text, the browser can mix them with tags that's why character entities are used in HTML to display reserved characters.

How to use an entity:

You can use an entity in your HTML document by name or by a numerical character reference. Each entity starts with symbol ampersand (&) and ends with a semicolon (;).

Syntax:

&entity_name;

OR

&#entity_number;

Most used HTML Character Entities

Result	Description	Entity Name	Entity Number
	non-breaking space	&nbsp;	160
<	less than	&lt;	60
>	greater than	&gt;	62
&	ampersand	&amp;	38

"	double quotation mark	&quot;	34
'	single quotation mark (apostrophe)	&apos;	39
¢	cent	&cent;	162
£	pound	&pound;	163
¥	yen	&yen;	165
€	Euro	&euro;	8364
©	copyright	&copy;	169
®	registered trademark	&reg;	174

Note:Entity names are case sensitive.

Advantage of entity name: An entity name is easy to remember.

Disadvantage of entity name: Browsers may not support all entity names, but the support for numbers is good.

Example:

```
<!DOCTYPE html>
<html>
<head>
<title></title>
</head>
 <body>
    <h3>HTML entity example</h3>
    <p> "This is the content written within entity"</p>

    <p> <p> Paragraph tag </p>

 </body>

</html>
```

Diacritical Marks in HTML

There are some special types of letters used in HTML whichhave some glyph added to the top or below the letters. These glyphs are called diacritical mark.

Some diacritical marks, like grave (̀) and acute (́) are called accents.Diacritical marks can be used both above and below a letter, inside a letter, and between two letters.

Following is a list of some diacritical marks:

Mark	Character	Construct	Result
`	a	a&#768;	à
´	a	a&#769;	á
ˆ	a	a&#770;	â
˜	a	a&#771;	ã
`	O	O&#768;	Ò
´	O	O&#769;	Ó
ˆ	O	O&#770;	Ô
˜	O	O&#771;	Õ

HTML Symbols

There are many mathematical, technical and currency symbols which are not present on a normal keyboard. We have to use HTML entity names to add such symbols to an HTML page.

If there no entity name exists, you can use an entity number, a decimal, or hexadecimal reference.

Example:

```
<!DOCTYPE html>
    <html>
     <body>
       <h3>The Currency Symbols</h3>
             <p>This is Indian Rupee symbol <b>₹<b></p>
      <p>This is Euro symbol <b>€</b></p>
      <p> This is Dollar symbol <b>#36;</b></p>
     </body>
   </html>
```

Mathematical Symbols Supported by HTML

Char	Number	Entity	Description
∀	&#8704;	&forall;	FOR ALL
∂	&#8706;	&part;	PARTIAL DIFFERENTIAL
∃	&#8707;	&exist;	THERE EXISTS
∅	&#8709;	&empty;	EMPTY SETS
∇	&#8711;	&nabla;	NABLA
∈	&#8712;	&isin;	ELEMENT OF
∉	&#8713;	&notin;	NOT AN ELEMENT OF

∋	&#8715;	&ni;	CONTAINS AS MEMBER
∏	&#8719;	&prod;	N-ARY PRODUCT
∑	&#8721;	&sum;	N-ARY SUMMATION

Greek Symbols Supported by HTML

Char	Number	Entity	Description
Α	&#913;	&Alpha;	GREEK CAPITAL LETTER ALPHA
Β	&#914;	&Beta;	GREEK CAPITAL LETTER BETA
Γ	&#915;	&Gamma;	GREEK CAPITAL LETTER GAMMA
Δ	&#916;	&Delta;	GREEK CAPITAL LETTER DELTA
Ε	&#917;	&Epsilon;	GREEK CAPITAL LETTER EPSILON
Ζ	&#918;	&Zeta;	GREEK CAPITAL LETTER ZETA

Some Important Symbols Supported by HTML

Char	Number	Entity	Description
©	&#169;	&copy;	COPYRIGHT SIGN
®	&#174;	&reg;	REGISTERED SIGN
€	&#8364;	&euro;	EURO SIGN
™	&#8482;	&trade;	TRADEMARK
←	&#8592;	&larr;	LEFTWARDS ARROW
↑	&#8593;	&uarr;	UPWARDS ARROW
→	&#8594;	&rarr;	RIGHTWARDS ARROW
↓	&#8595;	&darr;	DOWNWARDS ARROW

♠	&#9824;	&spades;	BLACK SPADE SUIT
♣	&#9827;	&clubs;	BLACK CLUB SUIT
♥	&#9829;	&hearts;	BLACK HEART SUIT
♦	&#9830;	&diams;	BLACK DIAMOND SUIT

HTML Charset

HTML Charset is also called HTML Character Sets or HTML Encoding. It is used to display an HTML page properly and correctly because for displaying anything correctly, a web browser must know which character set (character encoding) to use.

HTML Character Encoding

There are various types of Character Encoding which are given below:

ASCII Character Set

ASCII stands for American Standard Code for Information Interchange. In HTML, the first ever character encoding standard is the ASCII standard. ASCII provides 128 different alphanumeric characters that could be used on the internet: numbers (0-9), English letters (A-Z), and some special characters like! $ + - ( ) @ <> .

The main problem with ASCII encoding was it contains a limited range of characters. It contains mainly 128 characters.

ANSI Character Set

ANSI stands for American National Standard Institute. It is character set standard which is an extended version of standard ASCII character set. It supports 256 character set. ANSI also called as Windows-1252, and it was the default character set for Windows up to Windows 95.

ISO-8859-1 Character Set

ISO-8859-1 was the default character encoding in HTML 2.0. It was also an extension of ASCII standard with International characters. It also used full bytes (8-bits) to show characters.

UTF-8 Character Set

UTF-8 is a variable width character encoding which covers almost all of the characters and symbols in the world. ANSI (Windows-1252) was the original Windows character set, which supported 256 different character codes.

ISO-8859-1 was the default character set for HTML 4. This character set also supported 256 different character codes.

Why UTF 8 is also supported in HTML4?

Because ANSI and ISO-8859-1 were so limited, HTML 4 also supported UTF-8.The default character encoding for HTML5 is UTF-8.

UTF-8 syntax for HTML4:

```
<meta http-equiv="Content-Type" content="text/html;charset=ISO-8859-1">
```

UTF-8 syntax for HTML5:

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

HTML URL Encode

What is URL?

URL stands for Uniform Resource Locator. It is actually a web address. A URL can contain words i.e. (javatpoint.com) or an Internet Protocol (IP) address i.e.195.201.68.81. But most of the user use URL in the form of words because it is easy to remember than numbers.

Syntax of a URL:

scheme://prefix.domain:port/path/filename

Here,

scheme is used to define the type of Internet service (most common is http or https).

prefix is used to define a domain prefix (default for http is www).

domain is used to define the Internet domain name (like javaTpoint.com).

port is used to define the port number at the host (default for http is 80).

path is used to define a path at the server (If omitted: the root directory of the site).

filename is used to define the name of a document or resource.

Following is a list of some common types of schemes used in URL:

http(HyperText Transfer Protocol):Common web pages. Not encrypted.

https(Secure HyperText Transfer Protocol):Secure web pages. Encrypted.

ftp(File Transfer Protocol): Downloading or uploading files.

file: A file on your computer.

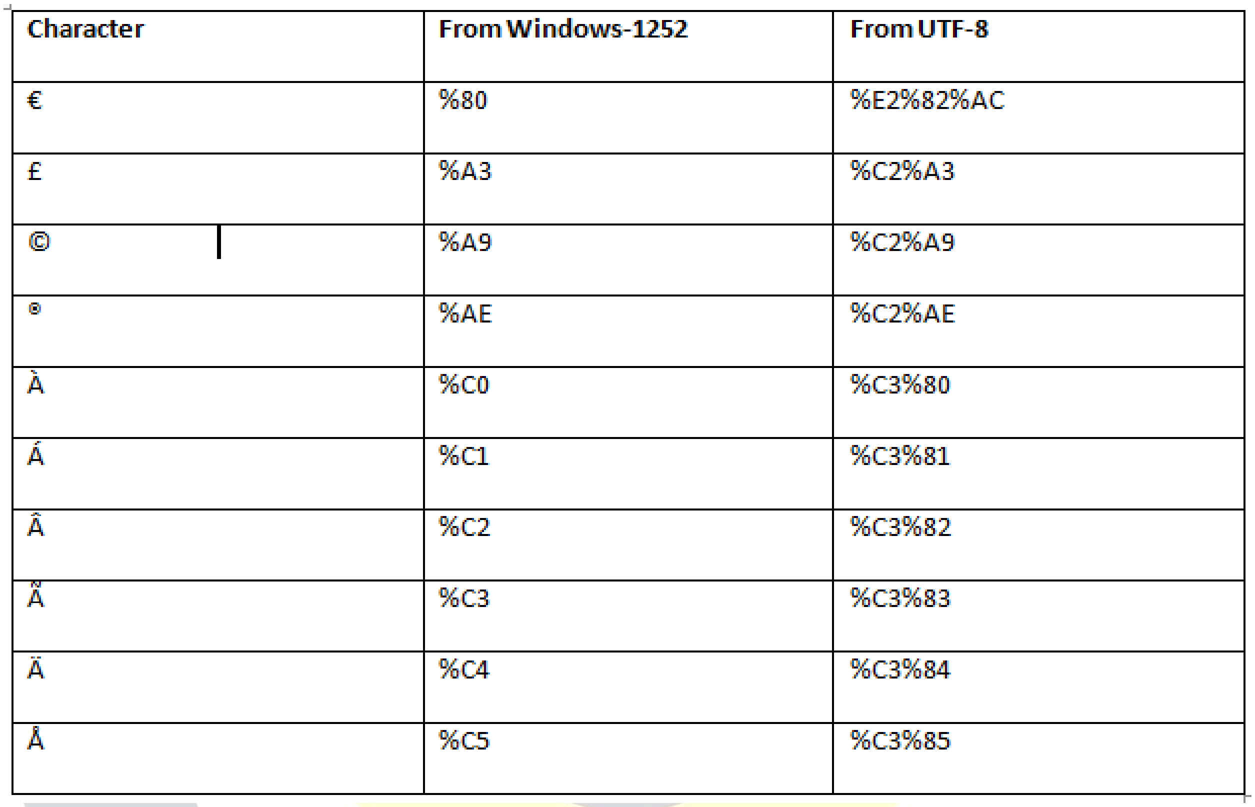
URL Encoding

URL encoding is used to convert non-ASCII characters into a format that can be used over the Internet because a URL is sent over the Internet by using the ASCII character-set only. If a URL contains characters outside the ASCII set, the URL has to be converted.

In URL encoding, the non-ASCII characters are replaced with a "%" followed by hexadecimal digits.

URLs cannot contain spaces. URL encoding normally replaces a space with a plus (+) sign, or %20.

Following is a list of some character sets which are encoded by browser after submitting the text.



Character	From Windows-1252	From UTF-8
€	%80	%E2%82%AC
£	%A3	%C2%A3
©	%A9	%C2%A9
®	%AE	%C2%AE
À	%C0	%C3%80
Á	%C1	%C3%81
Â	%C2	%C3%82
Ã	%C3	%C3%83
Ä	%C4	%C3%84
Å	%C5	%C3%85

Global Attributes

HTML global attributes are those attributes which are common for all HTML elements. The global attributes are supported by both standard and non-standard element.

The global attributes can be used with all elements, although it may not have any effect on some elements.

Following is the complete list of global attributes with their description:

Attributes	value	Description
accesskey	character	It is used to generate keyboard shortcuts for the current element.
class	classname	It is used to provide the class name for the current element. It is mainly used with the stylesheet.
Contenteditable	true false	It determines whether the content within an element is editable or not.
contextmenu	menu_id	It defines the id for the <menu> element which is used as a context menu (a menu appear on right click) for an element.
data-*	somevalue	It is used to store element-specific private data which can be accessed by Javascript.
dir	rtl ltr auto	It specifies the direction of the content inside the current element.

draggable 5	true false auto	It specifies whether the content within an element is movable or not using Drag and Drop API.
dropzone 5	copy move link	It specifies the action is taken on the dragged element when it is dropped, ¬¬ such as whether it is copied, moved or linked.
hidden 5		It is used to hide the element from view.
id	id	It specifies a unique id for the element. It can be used with CSS and JavaScript.
lang	language_code	It specifies the primary language for the content of an element.
style	style	It is used to apply inline CSS to the current element.
spellcheck 5	true false	It specifies whether the content should be checked for spelling errors or not.
tabindex	number	It determines the tabbing order of an element.
title	text	It is used to provide the title, name, or some extra information about the element.
translate 5	yes no	It specifies whether the content of the element should be translated when the page is localized or not.

HTML Event Attributes

When a browser reacts on user action, then it is called as an event. For example, when you click on the submit button, then if the browser displays an information box.

In HTML5 there are lots of event attributes available which can be activated using a programming language such as JavaScript.

Following is a table of event attributes, using these attributes you can perform several events.

Windows Event Attributes

Windows events are related for the window object, and it can only be applied with <body> tag.

Attribute	Description
onafterprint	Executed the script after the document is printed.
onbeforeprint	Executed the script before the document is printed.
onbeforeunload	Executed the script before a document being unloaded.
onerror	Executed the script when an error occurs.
onhashchange	Executed the script when the anchor part in URL of the webpage is changed.
onload	Executed the script when the webpage is entirely loaded.
onmessage	Executed the script when a message event occurs.
onoffline	Executed the script when the network connection is disconnected, and browser started working offline.
ononline	Executed the script when the browser started working online

onpagehide	Executed the script when the current webpage is hidden such as if the user has moved away from the current webpage.
onpageshow	Executed the script when the current webpage is focused.
onpopstate	Executed the script when the window's active history is changed.
onresize	Executed the script when the window is resized.
onstorage	Executed the script when web storage is updated.
onunload	Executed the script when the current webpage is unloaded, or window is closed.

Form Event Attributes

Form event occurs when the user performs some action within the form such as submitting the form, selecting input field, etc.

The form events can be used with any element, but these are mainly used with HTML form elements.

Following is the list of all Form Event attributes:

Attribute	Description
onblur	Executed the script when form element loses the focus.
onchange	Executed the script when the value of the element is changed.
onfocus	Trigger an event when the element gets focused.
oninput	Executed the script when the user enters input to the element.
oninvalid	Executed the script when the element does not satisfy its predefined constraints.

onreset	Triggers the event when user reset the form element values.
onsearch	Triggers the event when a search field receives some input.
onselect	Triggers the event when the user has selected some text.
onsubmit	Triggers the event when a form is submitted.

Keyboard Event Attributes

Keyboard event occurs when a user interacts with the keyboard. Following is a list of the Keyboard event.

Attribute	Description
onkeydown	Triggers the event when the user presses down a key on the keyboard.
onkeypress	Trigger the event when the user presses the key which displays some character.
onkeyup	Trigger the event when the user releases the currently pressed key.

Mouse Event Attributes

Attribute	Description
onclick	Trigger the event when the mouse clicks on the element.
ondblclick	Trigger the event when mouse double-click occurs on the element.
onmousedown	Trigger the event when the mouse button is pressed on the element.
onmousemove	Trigger the event when the mouse pointer moves over the element.

onmouseout	Trigger the event when the mouse moves outside the element.
onmouseover	Trigger the event when the mouse moves onto the element.
onmouseup	Trigger the event when the mouse button is released.
onmousewheel	Deprecated. Use the onwheel attribute.
onwheel	Trigger the event when the mouse wheel rolls up or down on the element

Clipboard Event Attributes

Attribute	Description
oncopy	Trigger the event when the user copies the content to the system clipboard.
oncut	Trigger the event when the content of an element is cut and copy to the clipboard.
onpaste	Trigger the event when the user pastes some content in an element.

Media Event Attributes

Attribute	Description
onabort	Executed the script when media playback is aborted.
oncanplay	Executed the script when the media file is ready to play.
oncanplaythrough	Executed the script when the media file is ready to play without buffering or stopping.

oncuechange	Executed the script text cue of <track> element is changed.
ondurationchange	Executed the script when the media file duration is changed.
onemptied	Executed the script if media occurs some fatal error, and the file becomes unavailable.
onended	Executed the script when the media file occurs its end point.
onerror	Executed the script when some error occurred while fetching the media data.
onloadeddata	Executed the script when media data is loaded.
onloadedmetadata	Executed the script when metadata of media file is loaded.
onloadstart	Executed the script when loading of media file starts.
onpause	Executed the script when media playback is paused.
onplay	Executed the script when media file ready to play after being paused.
onplaying	Executed the script when media file is started playing.
onprogress	Executed the script when the browser is in the process of getting the media data.
onratechange	Executed the script when playback speed changed.
onseeked	Executed the script when seek operation is ended and seeking attribute is set to false.
onseeking	Executed the script when seek operation is active and seeking attribute is set to true.

onstalled	Executed the script when browser unexpectedly stopped fetching the data media.
onsuspend	Executed the script if fetching of media data is intentionally stopped.
ontimeupdate	Executed the script when playback position is changed, such as if a user fasts forward the track.
onvolumechange	Executed the script when media volume is changed (muted or unmuted).
onwaiting	Executed the script if playback pause to wait for loading more data.