##### HTML

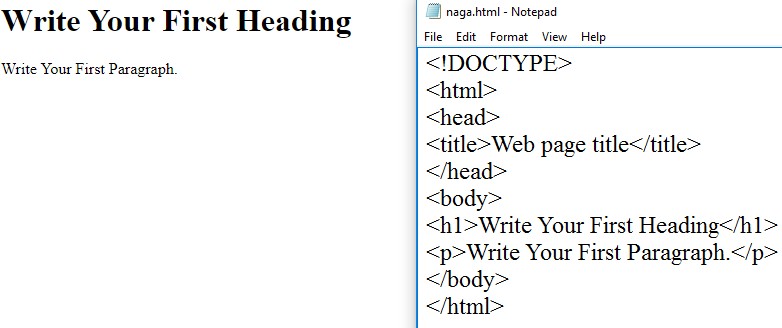
##### HTML Introduction

HTML or HTML5 provides basic and advanced concepts of HTML. Our HTML is developed for beginners and professionals. In our HTML, 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.

The major points of HTML are given below:

* HTML stands for HyperText 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.

Example with Output:



##### HTML Basics

* **<!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

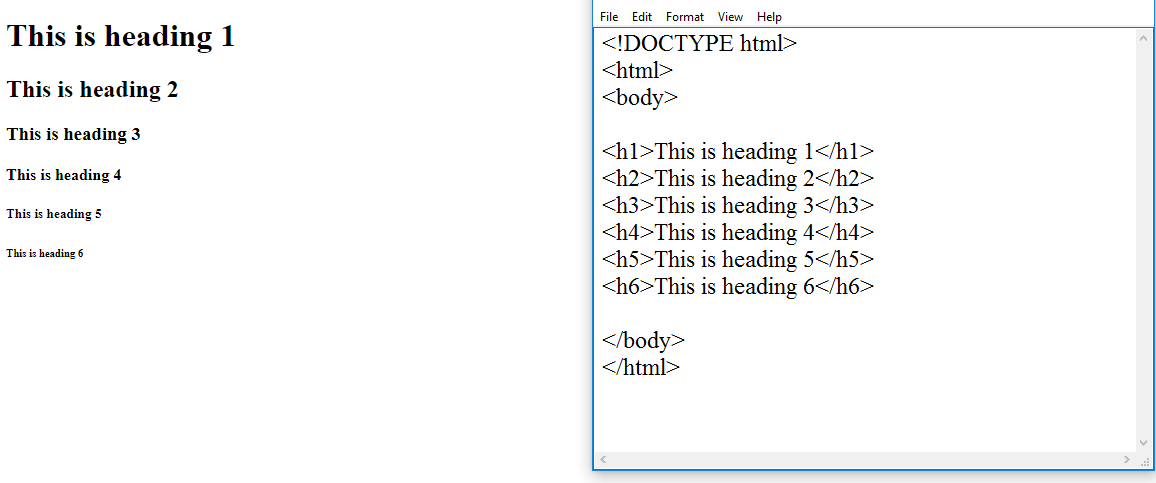
<!DOCTYPE>

* **<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.

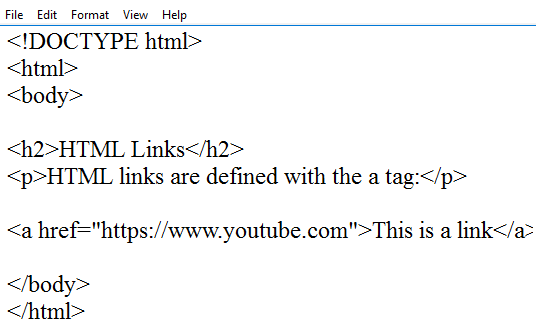
##### Features of HTML

* 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.

##### Example 1:



**Example 2:**



##### Output:



**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.

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.

##### Block-level and Inline HTML elements

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

1. Block-level element
2. Inline element

##### Block-level element:

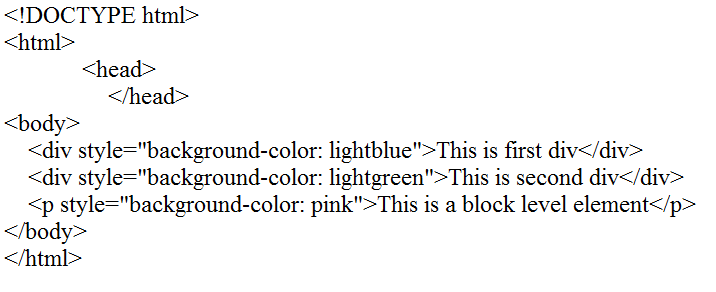
* + These are the elements, which structure main part of web page, by dividing a page into coherent blocks.
  + A block-level element always start with new line and takes the full width of web page, from left to right.
  + These elements can contain block-level as well as inline elements.

Following are the block-level elements in HTML.

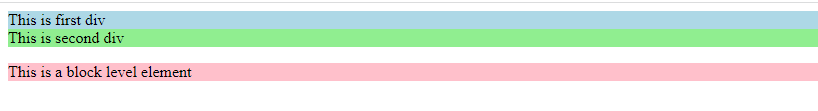
<address>, <article>, <aside>, <blockquote>, <canvas>, <dd>, <div>, <dl>, <dt>, <fieldset>,

<figcaption>, <figure>, <footer>, <form>, <h1>-<h6>, <header>, <hr>, <li>, <main>, <nav>,

<noscript>, <ol>, <output>, <p>, <pre>, <section>, <table>, <tfoot>, <ul> and <video>.



Output:



In the above example we have usedtag, 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>. 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) |

##### HTML5 Semantic

In any language, it is essential to understand the meaning of words during communication. And if this is a computer communication then it becomes more critical. So HTML5 provides more semantic elements which make easy understanding of the code.

Hence Semantics defines the meaning of words and phrases, i.e.

Semantic elements= elements with a meaning. Semantic elements have a simple and clear meaning for both, the browser and the developer.

##### For example:

In HTML4 we have seen <div>, <span> etc. are which are non-semantic elements. They don't tell anything about its content.

On the other hand, <form>, <table>, and <article> etc. are semantic elements because they clearly define their content.

HTML5 semantic elements are supported by all major browsers.

##### Why to use semantic elements?

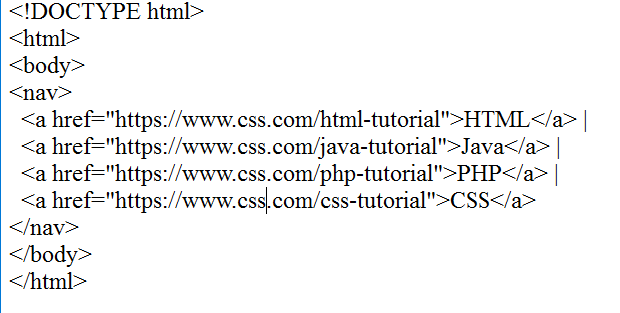
In HTML4, developers have to use their own id/class names to style elements: header, top, bottom, footer, menu, navigation, main, container, content, article, sidebar, topnav, etc.

This is so difficult for search engines to identify the correct web page content. Now in HTML5 elements (<header><footer><nav><section><article>), this will become easier. It now allows data to be shared and reused across applications, enterprises, and communities."

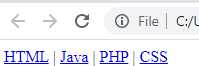
Semantic elements can increase the accessibility of your website, and also helps to create a better

|  |  |  |
| --- | --- | --- |
| **Index** | **Semantic**  **Tag** | **Description** |
| 1. | <article> | Defines an article |
| 2. | <aside> | Defines content aside from the page content |
| 3. | <details> | Defines additional details that the user can view or hide |
| 4. | <figcaption> | Defines a caption for a <figure> element |
| 5. | <figure> | Specifies self-contained content, like illustrations, diagrams, photos,  code listings, etc. |
| 6. | <footer> | Defines a footer for a document or section |
| 7. | <header> | Specifies a header for a document or section |
| 8. | <main> | Specifies the main content of a document |
| 9. | <mark> | Defines marked/highlighted text |
| 10. | <nav> | Defines navigation links |
| 11. | <section> | Defines a section in a document |
| 12. | <summary> | Defines a visible heading for a <details> element |
| 13. | <time> | Defines a date/time |

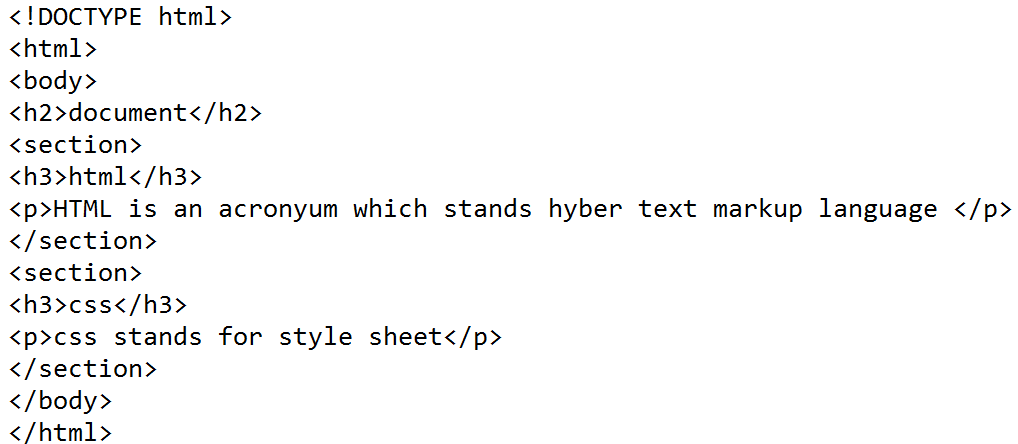
Semantic Elements in HTML5



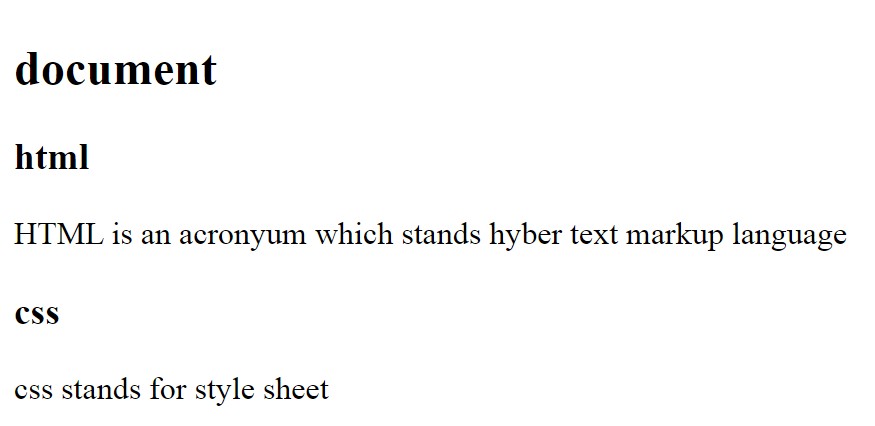
OUTPUT:



Example



##### Output:



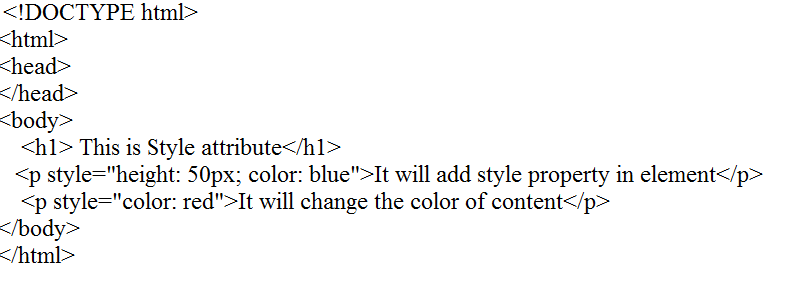
**HTML Attributes**

* + 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:



Output:



##### 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:

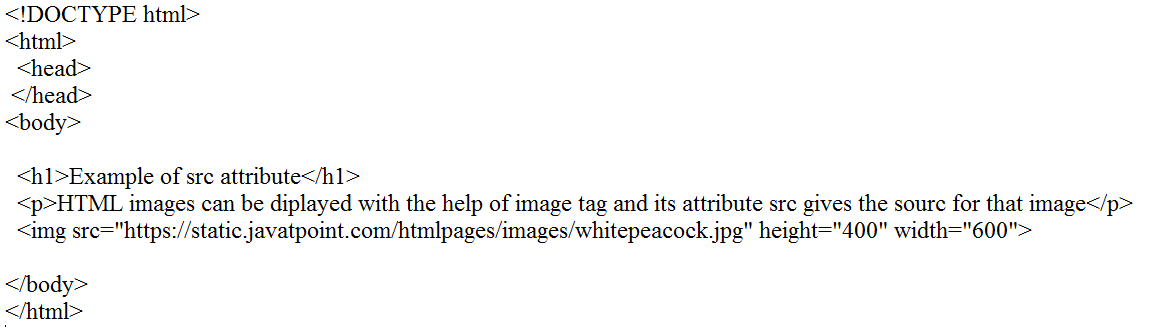
<p title="This is paragraph tag">Move the cursor over the heading and paragraph, and you will see a d

##### 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.

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

##### Example:



**Output:**



##### HTML Headings

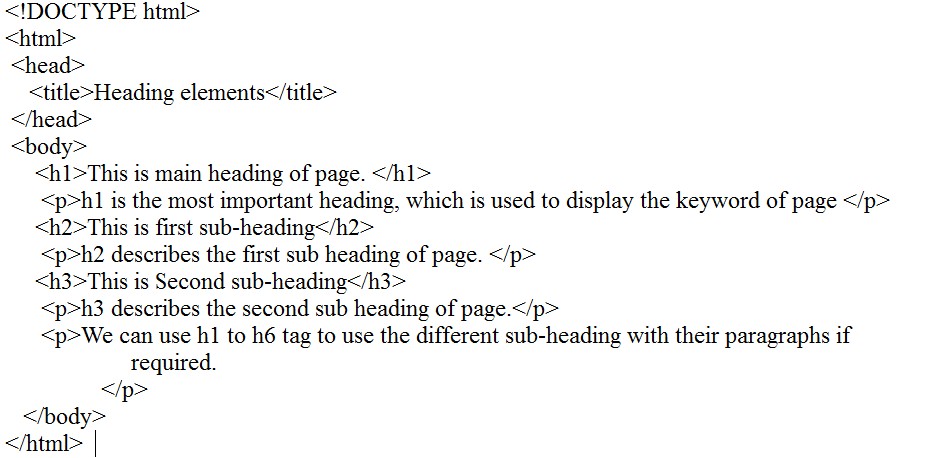
A HTML heading or HTML h tag can be defined as a title or a subtitle which you want to display on the webpage. When you place the text within the heading tags <h1> </h1>, it is displayed

on the browser in the bold format and size of the text depends on the number of heading.

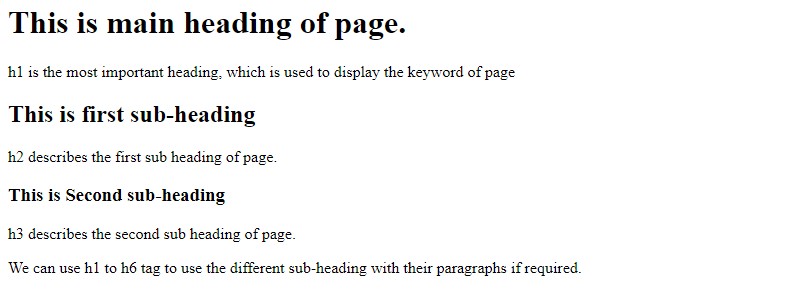
There are six different HTML headings which are defined with the <h1> to <h6> tags, from highest level h1 (main heading) to the least level h6 (least important heading).

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.

Example:



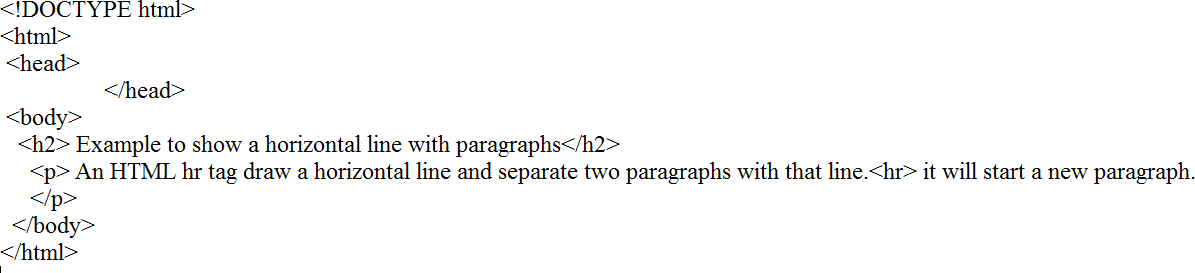
Output:



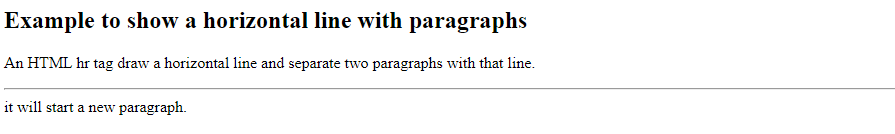
##### HTML Paragraph

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.

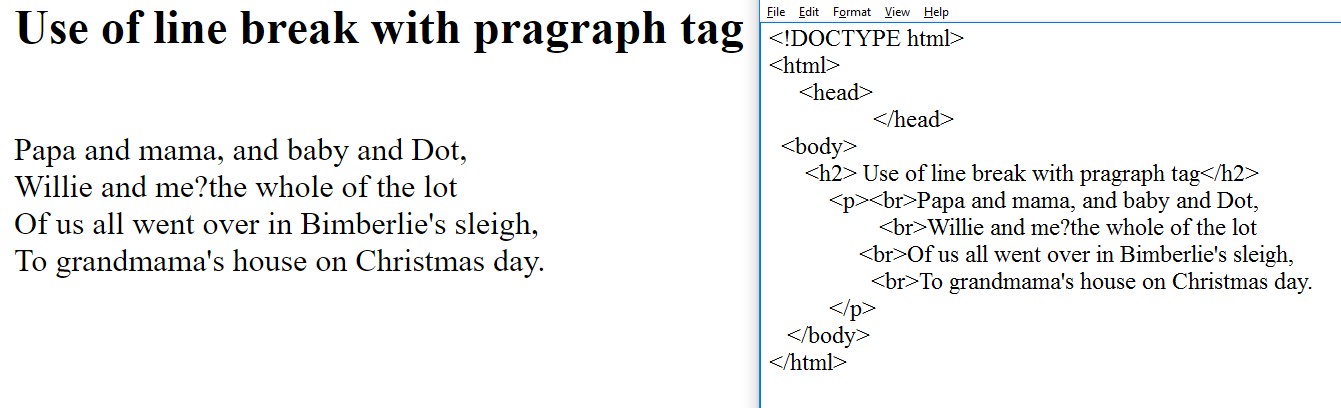
##### Example:



**Output:**



##### Example:



**HTML styles**

HTML Style is used *to change or add the style on existing HTML elements*. There is a default style for every HTML element e.g. background color is white, text color is black etc.

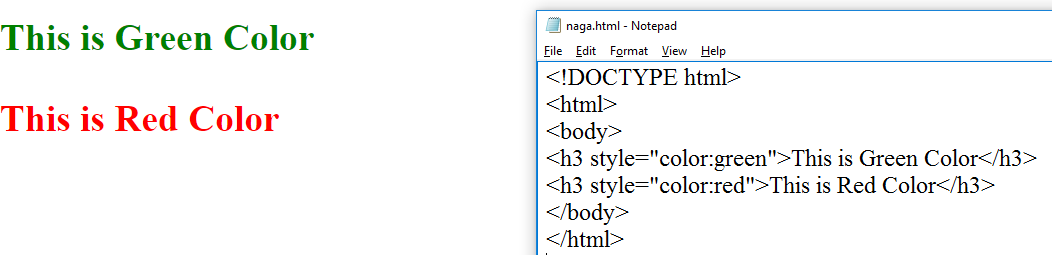
The style attribute can by used with any HTML tag. To apply style on HTML tag, you should have the basic knowledge of css properties e.g. color, background-color, text-align, font-family, font- size etc.

The syntax of style attribute is given below:

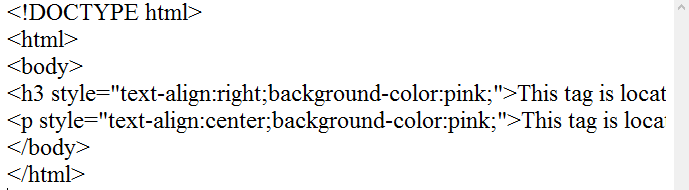
style= "property:value"

##### HTML Style color

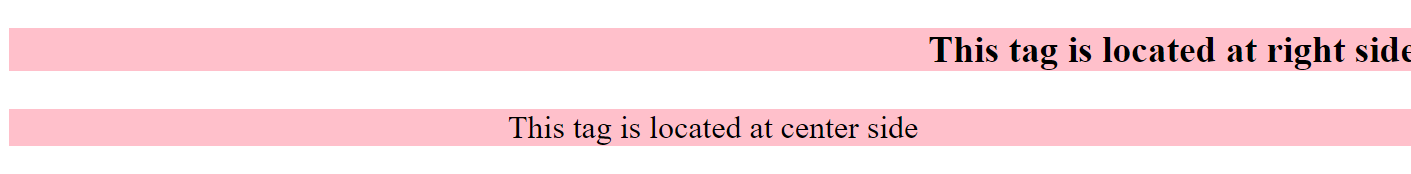
The color property is used to define the text color.



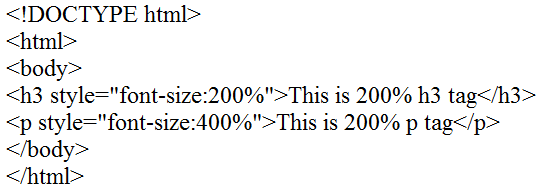
##### Example:



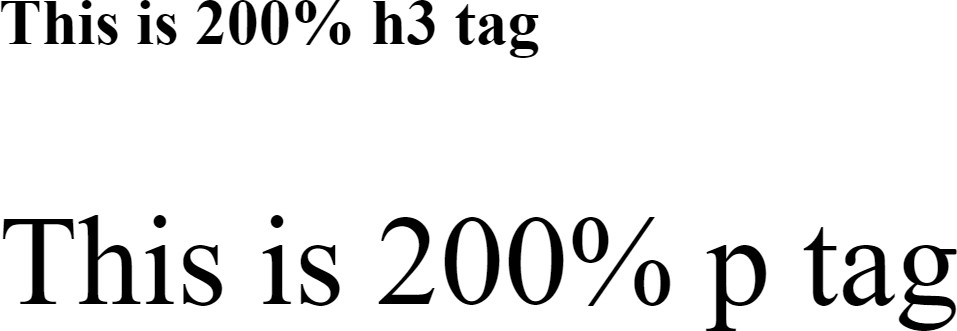
**Output:**



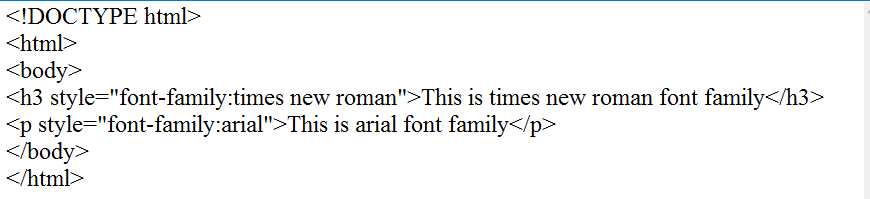
##### Example:



Output:



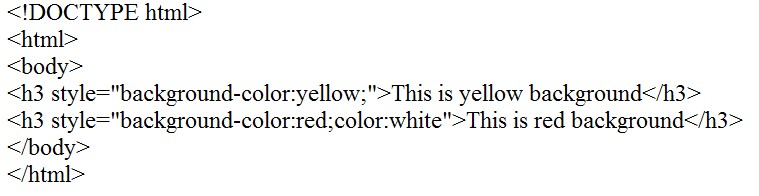
##### Example:



Output:



##### Example:



Output:



##### 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

|  |  |
| --- | --- |
| **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. |

How text appears in HTML and XHTML.

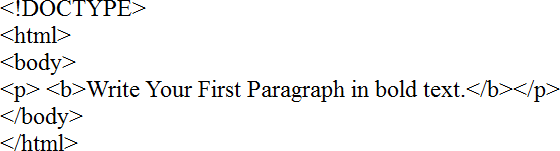
In HTML the formatting tags are divided into two categories:

* + 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.

##### 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.



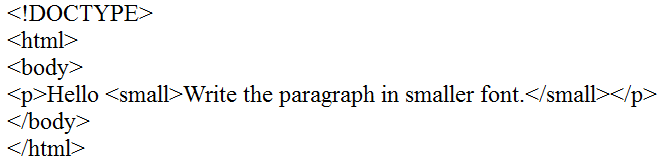
##### Output:



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.

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

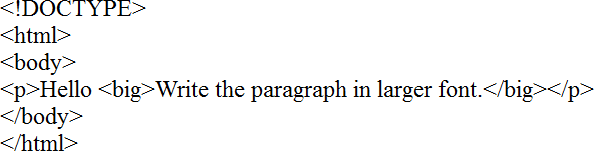
**Smallest text:**



Output:



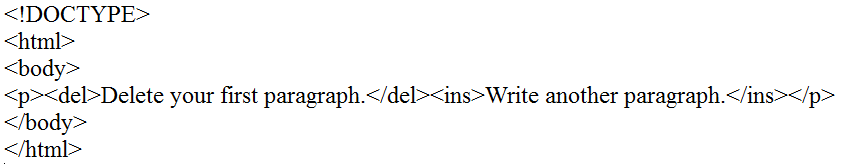
Example:



Output:



#### Indexed text:



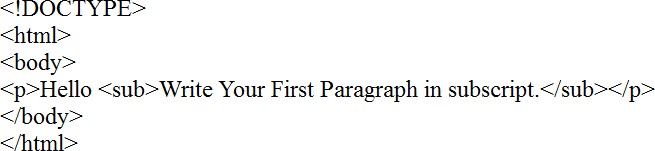
##### Output:



**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.

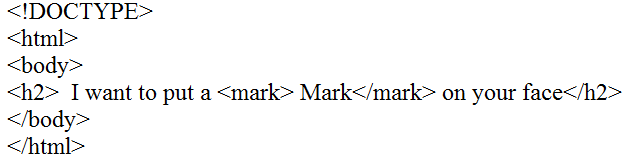
##### Example:



**Output:**



##### Html Marked formatting

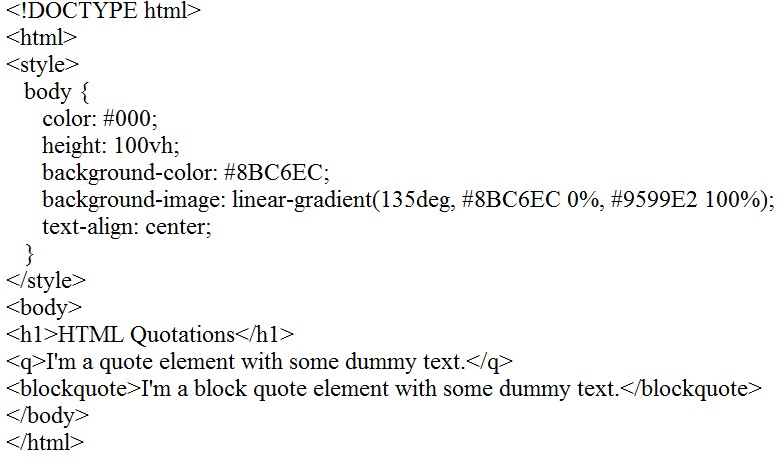


**Output:**

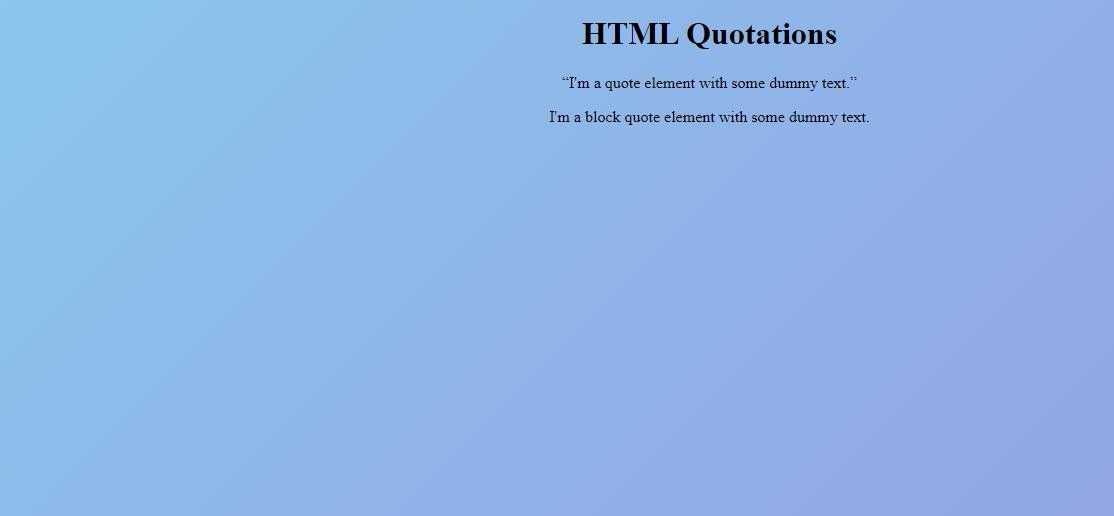


##### HTML Quotations

The HTML Quotations is used to create quotation in HTML. The HTML <q> tag is used to define a short quotation. It is a block-level element and HTML <blockquote> tag is used to define a section that is quoted from another source. It is also a block-level element.



##### Output:



**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.

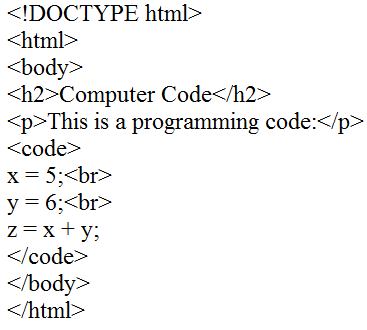
Following is a list of some tags which are used in HTML for this task.

* <code>
* <kbd>
* <samp>
* <var>
* <pre>

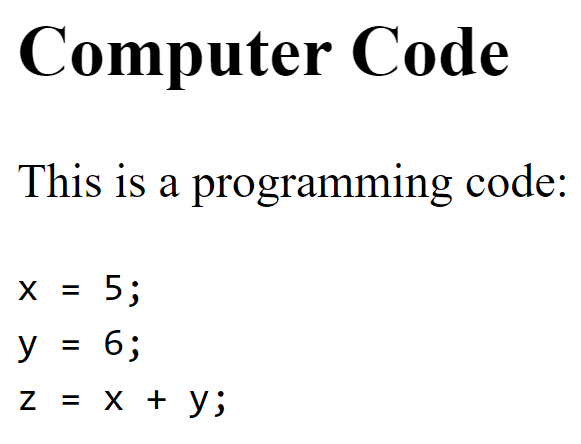
##### HTML <code> element

It is used to represent some programming code on your website. The content written between tag will be displayed in default monospace font.

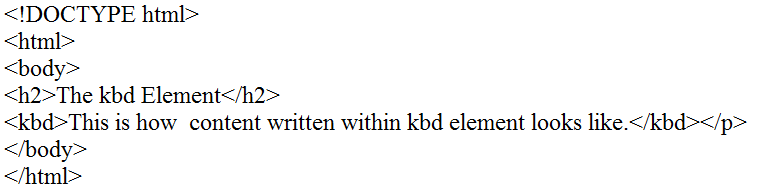
##### Example:



**Output:**



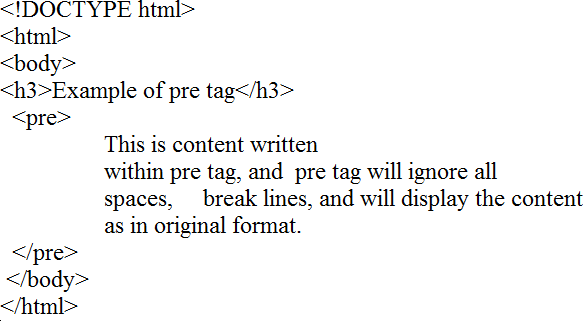
##### Example:



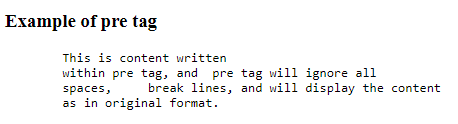
**Output:**



##### Example:



**Output:**



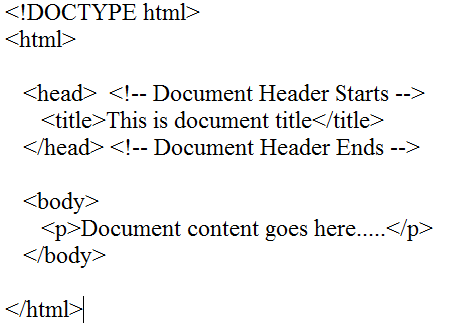
##### Comments & Colors

HTML Comment is a piece of code which is ignored by any web browser. It is a good practice to add comments into your HTML code, especially in complex documents, to indicate sections of a document, and any other notes to anyone looking at the code. Comments help you and others understand your code and increases code readability.

HTML comments are placed in between <!-- ... --> tags. So, any content placed with-in <!-- ... --

> tags will be treated as comment and will be completely ignored by the browser.

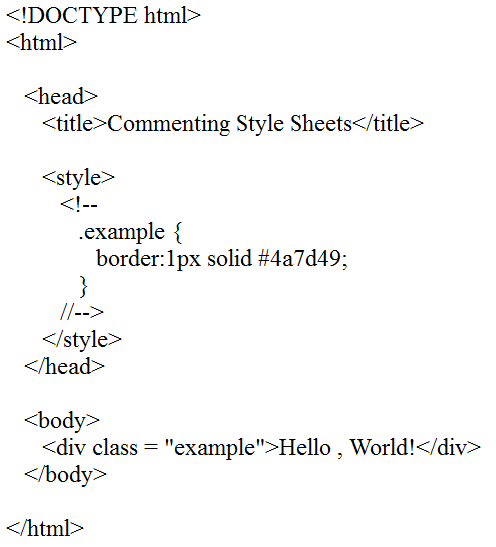
##### Example:



**Output:**



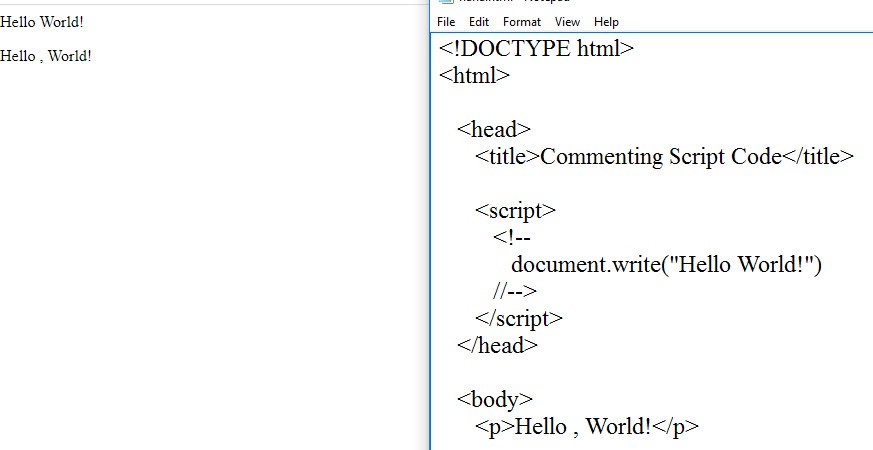
##### Example



**Output:**



##### Example:



**HTML CSS, Links and Images**

We can add image as a link and other HTML elements as a link. A link is a connection from one Web page to another web page.

We can add page links to a web page. HTML links are hyperlinks. The <a> tag defines a hyperlink and used to link from one page to another. href attribute is used with the <a> tag, which indicates the link's destination.

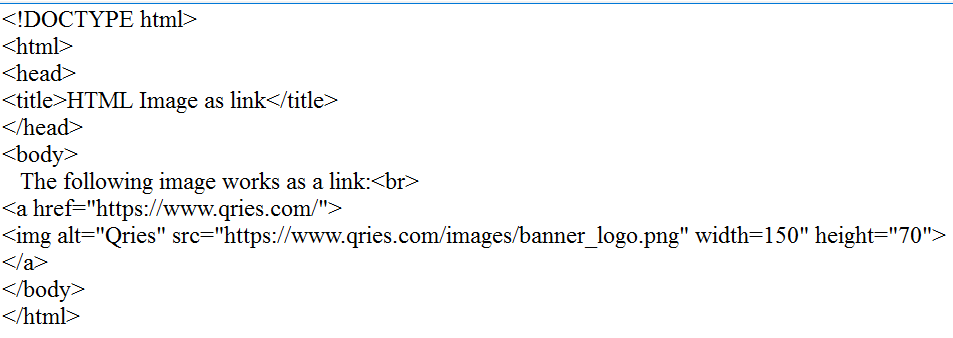
To make page links in an HTML page, use the <a> and </a> tags, with href attribute used to define the links. We should use the <a>…</a> tags inside <body>…</body> tags.

##### Syntax

Following is the syntax to add image as a link on the web page.

<a href="link address"><imgsrc="image destination"></a>

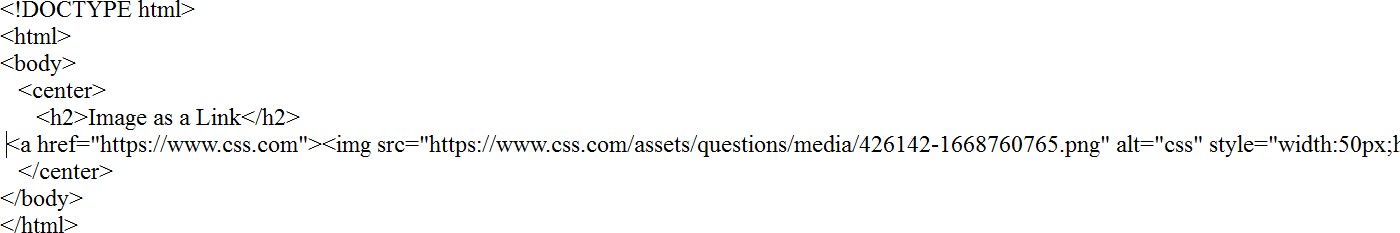
##### Code:



Output:



##### Images link



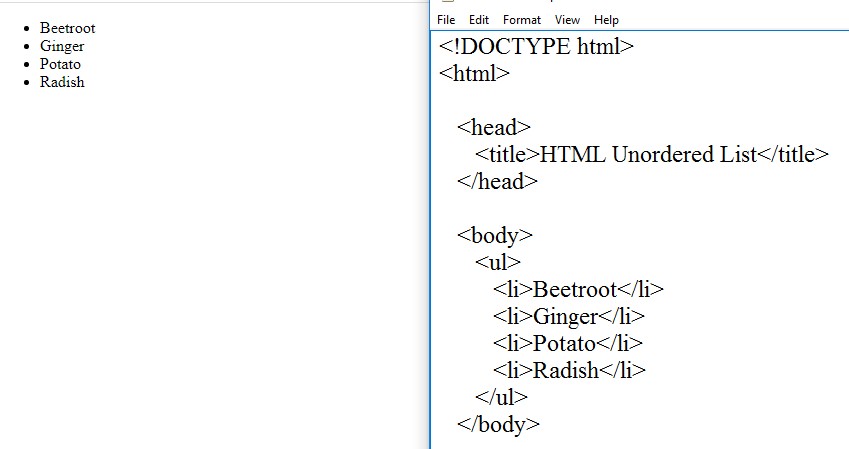
**HTML Lists**

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

* <ul> − An unordered list. This will list items using plain bullets.
* <ol> − An ordered list. This will use different schemes of numbers to list your items.
* <dl> − A definition list. This arranges your items in the same way as they are arranged in a dictionary.

##### HTML Unordered Lists

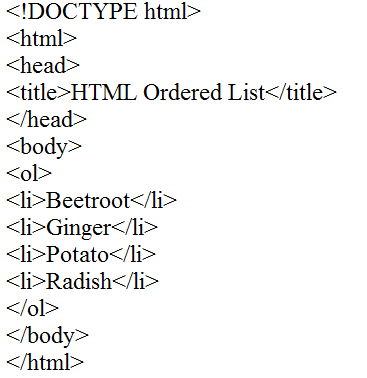
An unordered list is a collection of related items that have no special order or sequence. This list is created by using HTML <ul> tag. Each item in the list is marked with a bullet.



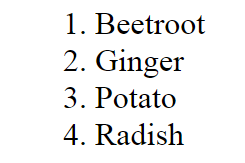
##### HTML Ordered Lists

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

##### Example



**Output:**



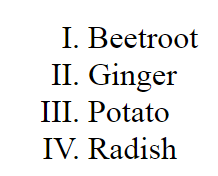
##### The type Attribute

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

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

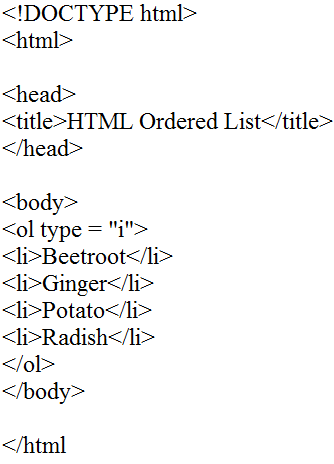
##### Example

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

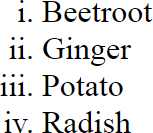


##### Example

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

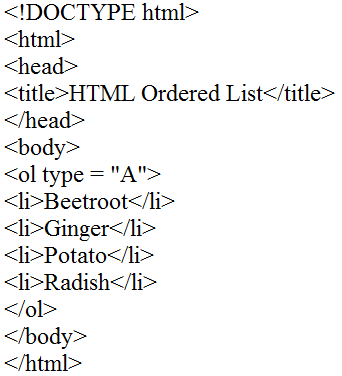


This will produce the following result −

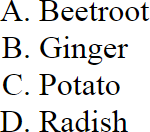


##### Example

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

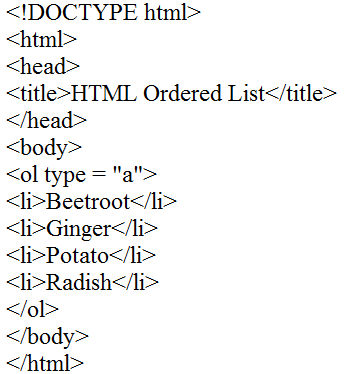


This will produce the following result −



##### Example

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



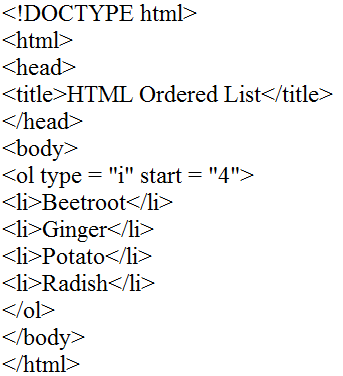
This will produce the following result –

##### The start Attribute

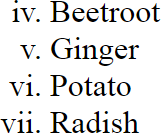
You can use start attribute for <ol> tag to specify the starting point of numbering you need. Following are the possible options −

##### Example

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



This will produce the following result −



##### HTML Definition Lists

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

Definition List makes use of following three tags.

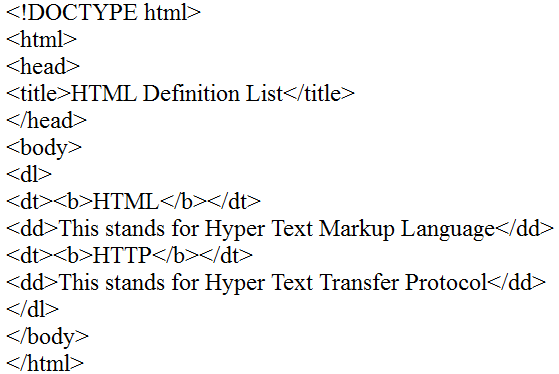
<dl> − Defines the start of the list

<dt> − A term

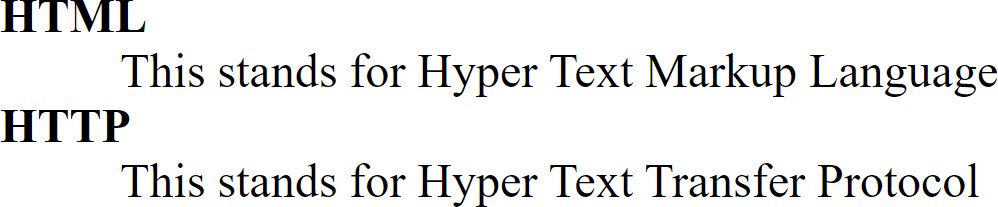
<dd> − Term definition

</dl> − Defines the end of the list

Example



This will produce the following result –



##### HTML Layout

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.

##### HTML Layouts

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.

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

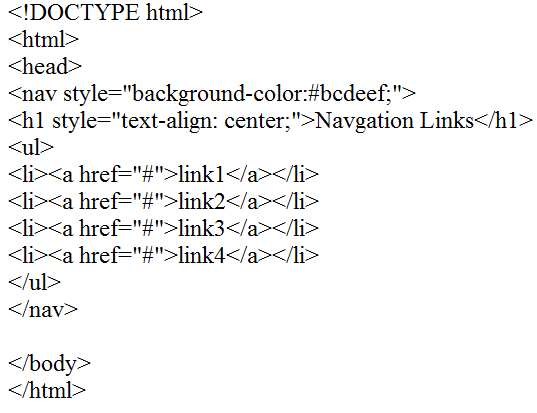
##### HTML <header>

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

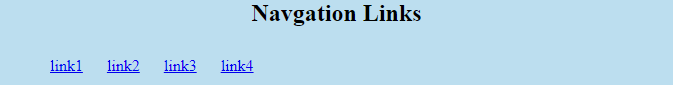
##### HTML <nav>

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:



Output:



##### 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.

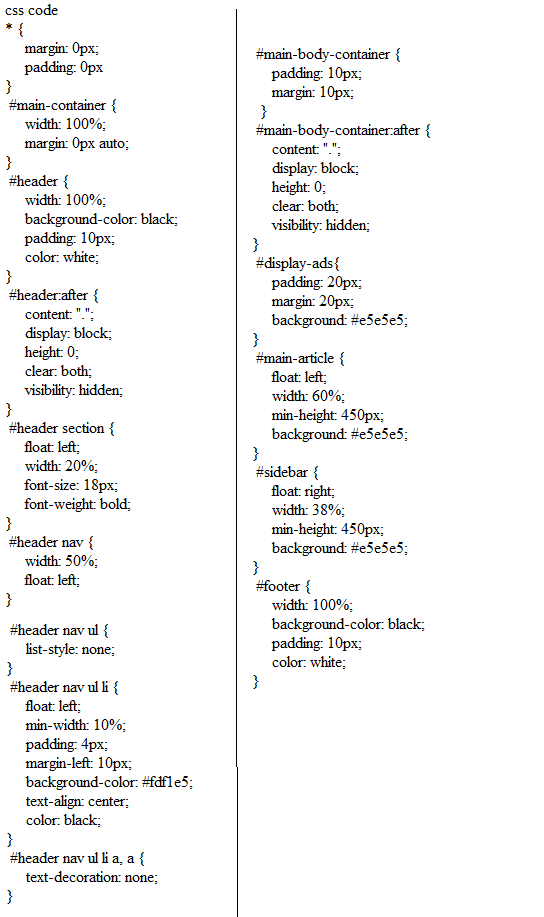
##### Example:



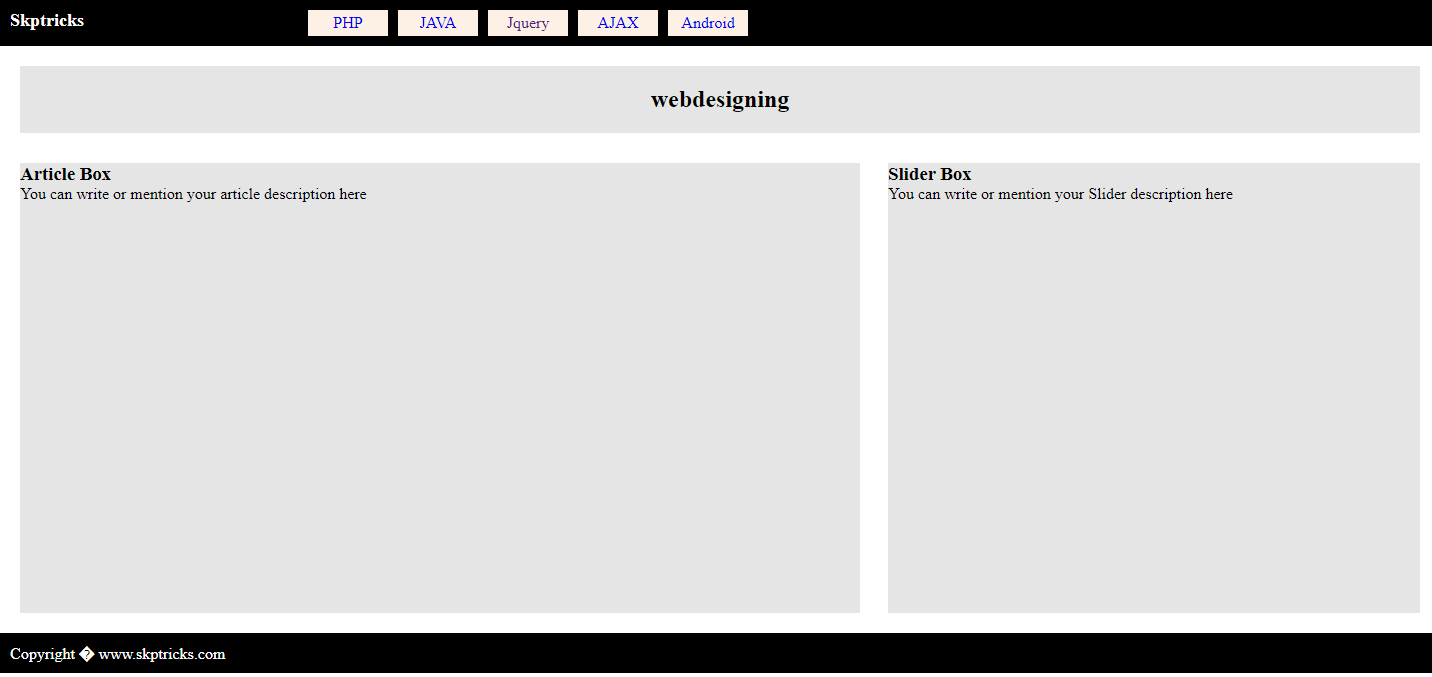
CSS Code



# HTML



Output:



##### HTML Responsive

The trends in responsive design are going crazy! Every web application built today is responsive to diversified devices to be accessed by the end-users. As you are already here, you must have known the importance of Responsive HTML of a website or application.

We will be working in CSS, not in HTML. The HTML code remains the same, the CSS styles will present the HTML code in different layouts for mobiles, tablets, desktops. This is one of the major reasons to separate content from presentation.

Nowadays, we have got many CSS frameworks to make our website responsive at ease. These frameworks use pure CSS to achieve the required level responsiveness. So, exposure with CSS for responsive design makes it easy to understand any CSS frameworks such as Bootstrap.

Let’s discuss how to make HTML content responsive to CSS styles.

##### What is Responsive?

A web page design that adapts to different screen sizes is called ‘responsive’. A desktop layout cannot fit smaller screen sizes like mobile or tablets. In the early days, the desktop site as shown in the mobiles, the user has to zoom in, scroll in X and Y direction to read the content on the webpage, which led to poor user experience. This is where a responsive design gives in. The

layouts are made to adapt to different screen sizes, making the website attractive and convenient for the user in different devices. Here well will focus on three areas in responsive design.

* Layouts
* Images
* Text
* Media Queries

Media Query is provided by CSS to achieve the concept of responsiveness. This is a way to conditionally apply CSS rules.

The syntax goes like this,

@media (query) {

// css rules

}

@media rule indicates the media query using which different styles can be applied based on

media types, screen sizes, orientation.

Media queries can be used to specify certain media types such as screen, speech, print and certain media features such as min-width, max-width, min-height, max-height, orientation and more.

The CSS rules are defined by specifying a media type and media features only if those specified criteria are met with the CSS which was applied.

@media screen and (max-width: 768px) {

.content { width: 100%;

}

}

The above sample illustrates the element with class content, which will be applied with the style

width 100% only when the type is a screen and the browser width is less than or equal to 768px. To combine more than one media feature the not keyword is used.

@media only screen and (min-width: 480px) and (max-width: 768px) {

.content { width: 100%;

}

}

This example applies styles to the browser width 480px to 768px.

Separate stylesheets can be used for specific media query like,

<link rel="stylesheet" media="screen and (min-width: 960px)" href="largedevice.css">

<link rel="stylesheet" media="screen and (max-width: 480px)" href="smalldevice.css">

##### Viewport

The viewport is the actual area in the rendering surface where the web page is displayed. The width or height constraint specified in the media query refers to the viewport width or height usually on the browser.

To make an HTML page to be responsive, the viewport Meta tag has to be included.

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

This sets the page width to device-width and initial zoom to 1. If the Meta tag is not included the mobile or tablet will try to fit the desktop layout but, it might not fit properly.

##### Breakpoints

Breakpoints in CSS are the places where different styles are applied, specified in the media query. For example, up to 480px styles for mobile devices get applied at the breakpoint. A standard breakpoint cannot be provided for each device. But if you want to know the commonly used breakpoints you can navigate to this link.

##### Responsive Layout

The most important part of the responsive design is making the page layout align the best according to the screen size. For desktops, contents can be shown in three columns; for tablets two columns and for mobile devices laying out content in one column is preferred for better user experience.

There are different layouts that refer to design patterns such as fluid, column drop, layout shifter, tiny tweaks, and off-canvas. To understand these layouts in details you can navigate to this link.

1. Fixed layout – Fixed width layout remains the same in all devices. The elements are fixed with absolute units such as pixels, inches, centimeters …
2. Fluid layout – Fluid layout stretches or shrinks with device width. The element’s properties are specified with relative units such as %, em, vw, vh …

Let’s take a look at the example of how media queries conditionally apply CSS styles. Here, the flexbox is used to create a responsive layout at ease.

Mobile-First Strategy

In a mobile-first strategy, a web page layout is created for small devices at first since there won’t be much work just stacking every element vertically. Then progress to medium devices and then to large-screen devices. This makes development easier and faster.

For large-screen devices,

@media only screen and (min-width: 769px) {

.section1, .section2, .section3 { width: 33.3%;

height: 350px;

}

}

large - responsive design

##### Responsive Image

To make an image responsive, set width to 100% and height to auto. This will automatically resize the image based on the browser width.

<imgsrc="micky.jpg" class="image">

.image { width: 100%; height: auto;

}

Image - responsive design

Responsive Text

Responsive text can be achieved by using viewport units such as vw (viewport width), vh (viewport height), vmin, vmax. Viewport units indicate the browser’s viewport, 1vw = 1% of viewport width.

<div class="text">

Angular is a TypeScript-based open-source web application framework…

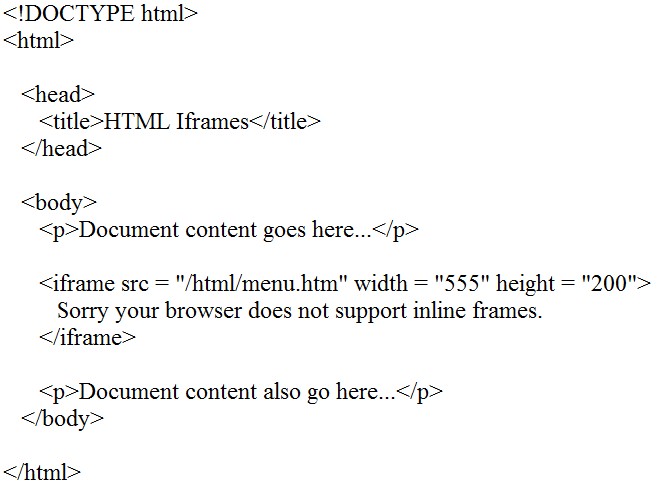
</div>

.text { font-size: 4vw; }

##### HTML iframes

You can define an inline frame with HTML tag <iframe>. The <iframe> tag is not somehow related to <frameset> tag, instead, it can appear anywhere in your document. The <iframe> tag defines a rectangular region within the document in which the browser can display a separate document, including scrollbars and borders. An inline frame is used to embed another document within the current HTML document.

The src attribute is used to specify the URL of the document that occupies the inline frame.



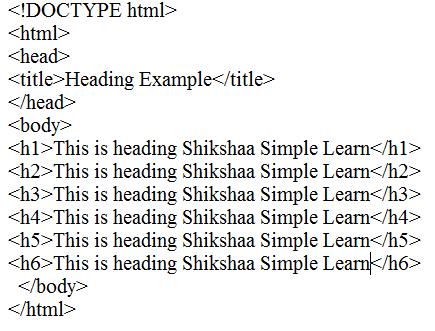
# HTML

|  |  |
| --- | --- |
| Sr.No | Attribute & Description |
| 1 | Src  This attribute is used to give the file name that should be loaded in the frame. Its value can be any URL. For example, src = "/html/top\_frame.htm" will load an HTML file available in html directory. |
| 2 | Name  This attribute allows you to give a name to a frame. It is used to indicate which frame a document should be loaded into. This is especially important when you want to create links in one frame that load pages into an another frame, in which case the second frame needs a name to identify itself as the target of the link. |
| 3 | Frameborder  This attribute specifies whether or not the borders of that frame are shown; it overrides the value given in the frameborder attribute on the <frameset> tag if one is given, and this can take values either 1 (yes) or 0 (no). |
| 4 | Marginwidth  This attribute allows you to specify the width of the space between the left and right of the frame's borders and the frame's content. The value is given in pixels. For example marginwidth = "10". |
| 5 | Marginheight  This attribute allows you to specify the height of the space between the top and bottom of the frame's borders and its contents. The value is given in pixels. For example marginheight = "10". |
| 6 | Height  This attribute specifies the height of <iframe>. |
| 7 | Scrolling  This attribute controls the appearance of the scrollbars that appear on the frame. This takes values either "yes", "no" or "auto". For example scrolling = "no" means it should not have scroll bars. |
| 8 | Longdesc  This attribute allows you to provide a link to another page containing a long description of the contents of the frame. For example longdesc = "framedescription.htm" |
| 9 | width  This attribute specifies the width of <iframe>. |

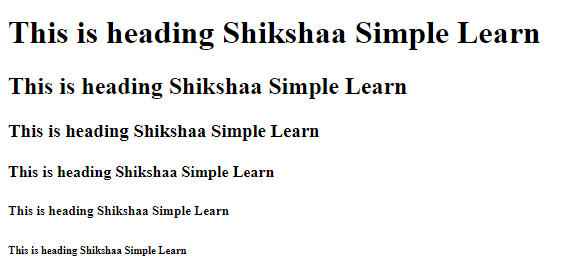
### HTML Heading Tag:

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

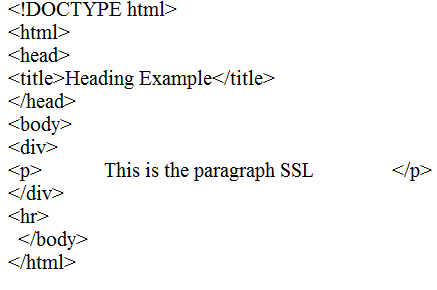
##### Example:



**Output:**



##### Example:



**Output:**



### Html heads

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.

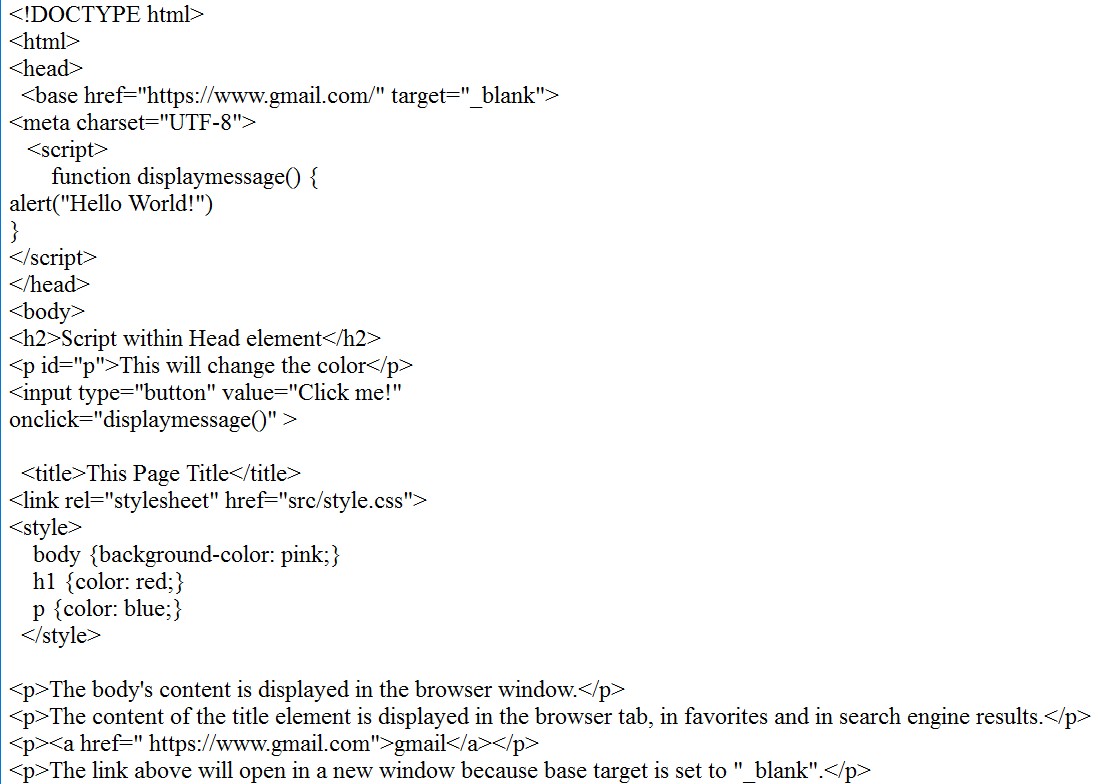
Metadata defines the document title, character set, styles, links, scripts, and other meta information.

Following is a list of tags used in metadata:

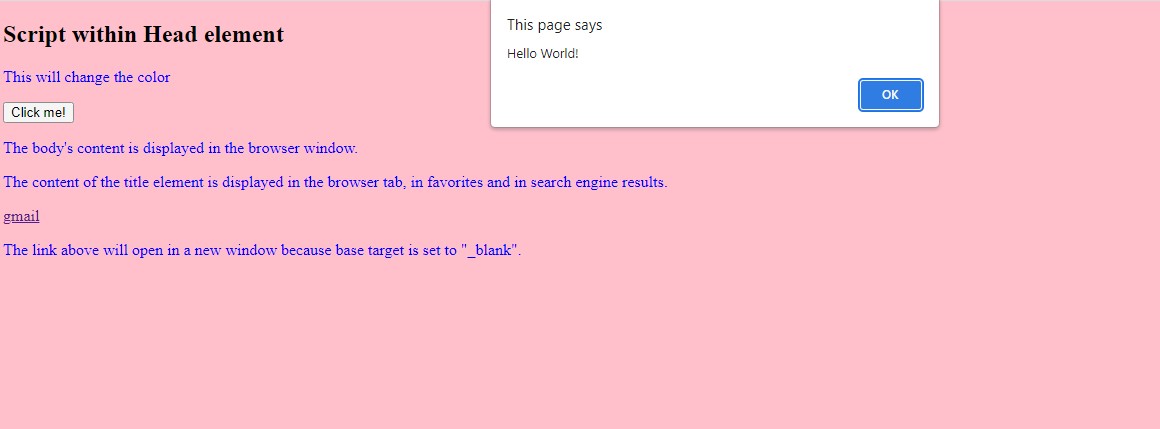
* <title>
* <style>
* <meta>
* <link>
* <script>
* <base>

**Metadata Tag:**

#### Example:



Output:



## FORMATTING Tag:

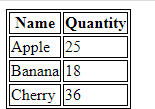
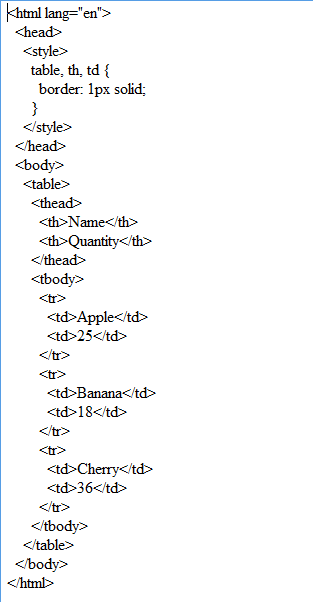
**Form Tag:**

**Example:**

# HTML



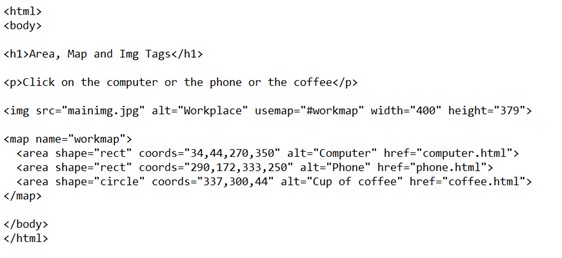
## Table Tag:



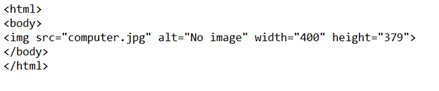
**Embedded Tag:**

#### Example:

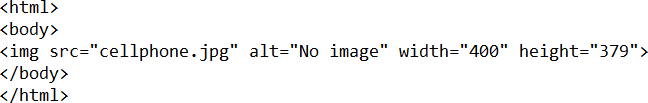
Main.html



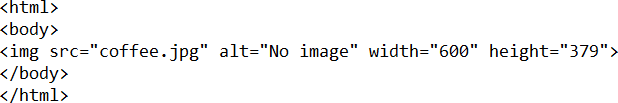
Computer.html



Phone.html



Coffee.html



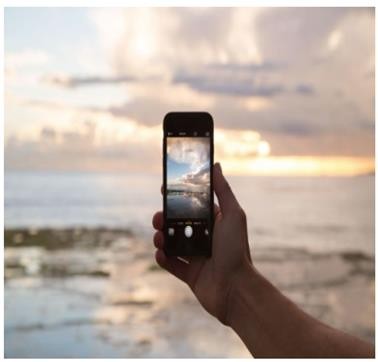
Output:



To click computer img area, to display



Or to click phn img area, to display



Or to click coffee img area, to display



#### <audio> and <source>:



**Output:**



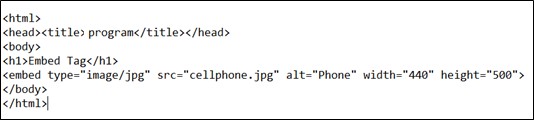
#### <figure> and <figcaption>:



**Output:**



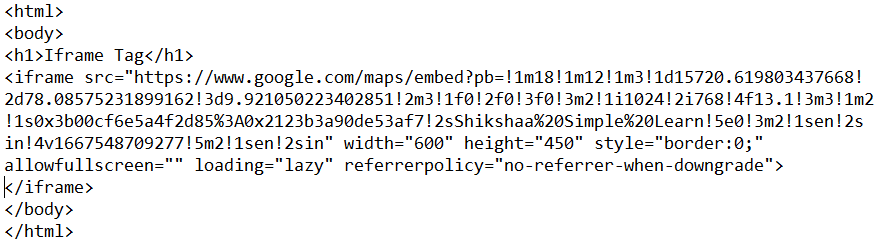
## <embed> Tag:



**Output:**



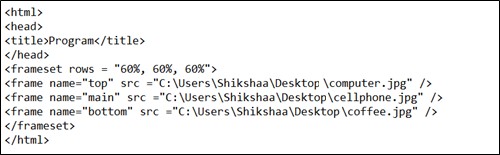
**<iframe> Tag:**



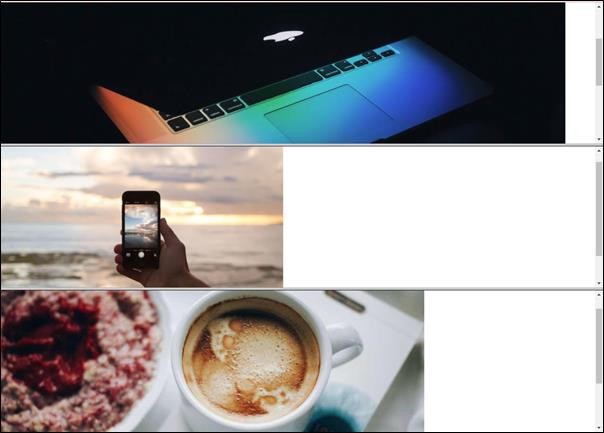
**Output**



#### <frame>, <frameset> , <noframe>:



**Output:**



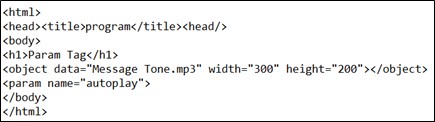
#### <object> Tag:



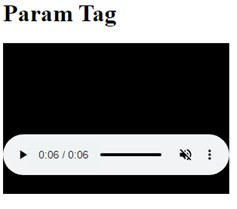
**Output:**



#### <param> Tag:



**Output:**



\*\*\*\*\*\*\*\*\*\* THANK YOU… \*\*\*\*\*\*\*\*\*\*