

# HTML

The major points of HTML are given below:

- HTML stands for HyperTextMarkup 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 is the standard markup language for creating Web pages
- HTML describes the structure of a web page
- HTML consists of a series of elements
- HTML tells the browser how to display the content

## What is HTML

HTML 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:** HyperText 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. HyperText 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.

`<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

**<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 in 1991.

**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.

The World Wide Web Consortium (W3C) is **an international community where Member organizations, a full-time staff, and the public work together to develop Web standards**. Led by Web inventor and Director Tim Berners-Lee and CEO Jeffrey Jaffe, W3C's mission is to lead the Web to its full potential.

**HTML5 :** HTML5 is the newest version of HyperTextMarkup 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.

## 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).
- `<html>`
- `<head>`
- `<title>Web page</title>`
- `</head>`
- `<body>`
- `<h1>My first Web Page</h1>`
- `<p>Hello World</p>`
- `</body>`
- `</html>`

## 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>

## 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 Documents

All HTML documents must start with a document type declaration: <!DOCTYPE html>.

The HTML document itself begins with `<html>` and ends with `</html>`.

The visible part of the HTML document is between `<body>` and `</body>`.

## Example

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

<h1>My First Heading</h1>
<p>My first paragraph.</p>

</body>
</html>
```

## The `<!DOCTYPE>` Declaration

The `<!DOCTYPE>` declaration represents the document type, and helps browsers to display web pages correctly.

It must only appear once, at the top of the page (before any HTML tags).

The `<!DOCTYPE>` declaration is not case sensitive.

The `<!DOCTYPE>` declaration for HTML5 is:

```
<!DOCTYPE html>
```

## HTML Headings

HTML headings are defined with the `<h1>` to `<h6>` tags.

`<h1>` defines the most important heading. `<h6>` defines the least important heading:

## Example

```
<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
```

## HTML Paragraphs

HTML paragraphs are defined with the `<p>` tag:

## Example

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

# HTML Links

HTML links are defined with the `<a>` tag:

## Example

```
<a href="https://www.google.com">This is a link</a>
```

The link's destination is specified in the `href` attribute.

Attributes are used to provide additional information about HTML elements.

# HTML Images

HTML images are defined with the `<img>` tag.

The source file (`src`), alternative text (`alt`), `width`, and `height` are provided as attributes:

## Example

```

```

# HTML Elements

An HTML element is defined by a start tag, some content, and an end tag.

# HTML Elements

The HTML **element** is everything from the start tag to the end tag:

`<tagname>`Content goes here...`</tagname>`

Examples of some HTML elements:

`<h1>`My First Heading`</h1>`

`<p>`My first paragraph.`</p>`

Start tag	Element content	End tag
<code>&lt;h1&gt;</code>	My First Heading	<code>&lt;/h1&gt;</code>
<code>&lt;p&gt;</code>	My first paragraph.	<code>&lt;/p&gt;</code>
<code>&lt;br&gt;</code>	<i>none</i>	<i>none</i>

**Note:** Some HTML elements have no content (like the `<br>` element). These elements are called empty elements. Empty elements do not have an end tag!

## Nested HTML Elements

HTML elements can be nested (this means that elements can contain other elements).

All HTML documents consist of nested HTML elements.

The following example contains four HTML elements (`<html>`, `<body>`, `<h1>` and `<p>`):

### Example

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

```
<h1>My First Heading</h1>
<p>My first paragraph.</p>
```



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

## Empty HTML Elements

HTML elements with no content are called empty elements.

The `<br>` tag defines a line break, and is an empty element without a closing tag:

### Example

```
<p>This is a <br> paragraph with a line break.</p>
```

## The style Attribute

The `style` attribute is used to add styles to an element, such as color, font, size, and more.

### Example

```
<p style="color:red;">This is a red paragraph.</p>
```

## The title Attribute

The `title` attribute defines some extra information about an element.

The value of the title attribute will be displayed as a tooltip when you mouse over the element:

### Example

```
<p title="I'm a tooltip">This is a paragraph.</p>
```

## We Suggest: Always Use Lowercase Attributes

The HTML standard does not require lowercase attribute names.

The title attribute (and all other attributes) can be written with uppercase or lowercase like **title** or **TITLE**.

However, W3C **recommends** lowercase attributes in HTML, and **demands** lowercase attributes for stricter document types like XHTML.

At W3Schools we always use lowercase attribute names.

## Single or Double Quotes?

Double quotes around attribute values are the most common in HTML, but single quotes can also be used.

In some situations, when the attribute value itself contains double quotes, it is necessary to use single quotes:

```
<p title='John "ShotGun" Nelson'>
```

Or vice versa:

```
<p title="John 'ShotGun' Nelson">
```

## Bigger Headings

Each HTML heading has a default size. However, you can specify the size for any heading with the `style` attribute, using the CSS `font-size` property:

### Example

```
<h1 style="font-size:60px;">Heading 1</h1>
```

## HTML Paragraphs

The HTML `<p>` element defines a paragraph.

A paragraph always starts on a new line, and browsers automatically add some white space (a margin) before and after a paragraph.

### Example

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

# HTML Horizontal Rules

The `<hr>` tag defines a thematic break in an HTML page, and is most often displayed as a horizontal rule.

The `<hr>` element is used to separate content (or define a change) in an HTML page:

```
<!DOCTYPE html>

<html>

<body>

  <h1>This is heading 1</h1>
  <p>This is some text.</p>
  <hr>

  <h2>This is heading 2</h2>
  <p>This is some other text.</p>
  <hr>

  <h2>This is heading 2</h2>
  <p>This is some other text.</p>

</body>
</html>
```

The `<hr>` tag is an empty tag, which means that it has no end tag.

## The Poem Problem

This poem will display on a single line:

```
<!DOCTYPE html>
```

```
<html>
<body>

<p>In HTML, spaces and new lines are ignored:</p>

<p>

    My Bonnie lies over the ocean.

    My Bonnie lies over the sea.

    My Bonnie lies over the ocean.

    Oh, bring back my Bonnie to me.

</p>

</body>
</html>
```

## Solution - The HTML `<pre>` Element

The HTML `<pre>` element defines preformatted text.

The text inside a `<pre>` element is displayed in a fixed-width font (usually Courier), and it preserves both spaces and line breaks:

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

<p>The pre tag preserves both spaces and line breaks:</p>
```

```
<pre>
  My Bonnie lies over the ocean.

  My Bonnie lies over the sea.

  My Bonnie lies over the ocean.

  Oh, bring back my Bonnie to me.
</pre>

</body>
</html>
```

# HTML Styles

The HTML **style** attribute is used to add styles to an element, such as color, font, size, and more.

## Example

I am Red

I am Blue

I am Big

```
<!DOCTYPE html>
```

```
<html>
<body>

<p>I am normal</p>
<p style="color:red;">I am red</p>
<p style="color:blue;">I am blue</p>
<p style="font-size:50px;">I am big</p>

</body>
</html>
```

## Background Color

The CSS `background-color` property defines the background color for an HTML element.

### Example

Set the background color for a page to powderblue:

```
<!DOCTYPE html>
<html>
<body style="background-color:powderblue;">

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

## Example

Set background color for two different elements:

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

<h1 style="background-color:powderblue;">This is a heading</h1>
<p style="background-color:tomato;">This is a paragraph.</p>

</body>
</html>
```

## Fonts

The CSS **font-family** property defines the font to be used for an HTML element:

### Example

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

<h1 style="font-family:verdana;">This is a heading</h1>
<p style="font-family:courier;">This is a paragraph.</p>

</body>
</html>
```

# Text Size

The CSS `font-size` property defines the text size for an HTML element:

## Example

```
<!DOCTYPE html>

<html>

<body>


<h1 style="font-size:300%;">This is a heading</h1>
<p style="font-size:160%;">This is a paragraph.</p>


</body>
</html>
```

# Text Alignment

The CSS `text-align` property defines the horizontal text alignment for an HTML element:

```
<!DOCTYPE html>

<html>

<body>


<h1 style="text-align:center;">Centered Heading</h1>
<p style="text-align:center;">Centered paragraph.</p>


</body>
</html>
```



# HTML <small> Element

The HTML `<small>` element defines smaller text:

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

<p>This is some normal text.</p>
<p><small>This is some smaller text.</small> </p>

</body>
</html>
```

# HTML <mark> Element

The HTML `<mark>` element defines text that should be marked or highlighted:

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

<p>Do not forget to buy <mark>milk</mark> today.</p>

</body>

</html>

```
<!DOCTYPE html>
<html>
<body>
  <p>Original <mark>Marked</mark></p>
  <p><mark style="background: #eed7e4">Marked
Data</mark></p>
</body>
</html>
```

Another way to set the color of mark

```
<!DOCTYPE html>
<html>
<head>
<style>
mark {
  background-color: blue;
  color: white;
}
</style>
</head>
<body>
```

<p>A mark element is displayed like this:</p>

<mark>Highlighted text!!</mark>

<p>Change the default CSS settings to see the effect.</p>

</body>

</html>

## HTML <del> Element

The HTML `<del>` element defines text that has been deleted from a document. Browsers will usually strike a line through deleted text:

<!DOCTYPE html>

<html>

<body>

<p>My favoritecolor is <del>blue</del> red.</p>

</body>

</html>

# HTML **<ins>** Element

The HTML **<ins>** element defines a text that has been inserted into a document (This text is not there before). Browsers will usually underline inserted text:

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p>Myfavoritecolor is <del>blue</del><ins>red</ins>.</p>
```

```
</body>
```

```
</html>
```

# HTML **<sub>** Element

The HTML **<sub>** element defines subscript text. Subscript text appears half a character below the normal line, and is sometimes rendered in a smaller font. Subscript text can be used for chemical formulas, like H<sub>2</sub>O:

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p>This is <sub>subscripted</sub> text.</p>
```

```
</body>
```

```
</html>
```

## HTML **<sup>** Element

The HTML **<sup>** element defines superscript text. Superscript text appears half a character above the normal line, and is sometimes rendered in a smaller font.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p>This is <sup>superscripted</sup> text.</p>
```

```
</body>
```

```
</html>
```

## HTML **<q>** for Short Quotations

The HTML **<q>** tag defines a short quotation.

Browsers normally insert quotation marks around the quotation.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

<p>Browsers usually insert quotation marks around the q element.</p>

<p>WWF's goal is to: <q>Build a future where people live in harmony with nature.</q></p>

</body>

</html>

## HTML <blockquote> for Quotations

The <blockquote> tag in HTML is used to display the long quotations (a section that is quoted from another source or from another author). It changes the alignment to make it unique from others. It contains both opening and closing tags.

**Attribute:** It contains a single attribute *cite* which is used to specify the source of the quotation.

<!DOCTYPE html>

<html>

<body>

<p>Browsers usually indent blockquote elements.</p>

```
<blockquote
```

```
  cite="http://www.worldwildlife.org/who/index.html">
```

For nearly 60 years, WWF has been protecting the future of nature. The world's leading conservation organization, WWF works in 100 countries and is supported by more than one million members in the United States and close to five million globally.

```
</blockquote>
```

```
</body>
```

```
</html>
```

## HTML **<abbr>** for Abbreviations

The `<abbr>` tag (Abbreviation) in HTML is used to define the abbreviation or short form of an element. The `<abbr>` tag is used as shortened versions and used to represent a series of letters. The abbreviation is used to provide useful information to the browsers, translation systems, and search-engines.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p>The <abbr title="World Health  
Organization">WHO</abbr> was founded in 1948.</p>
```



<p>Marking up abbreviations can give useful information to browsers, translation systems and search-engines.</p>

</body>

</html>

## HTML <address> for Contact Information

The HTML <address> tag defines the contact information for the author/owner of a document or an article.

The contact information can be an email address, URL, physical address, phone number, social media handle, etc.

The text in the <address> element usually renders in *italic*, and browsers will always add a line break before and after the <address> element.

Usually give address in address tag

<!DOCTYPE html>

<html>

<body>

<p>The HTML address element defines contact information (author/owner) of a document or article.</p>

<address>

Written by John Doe.<br>

Visit us at:<br>

Example.com<br>

Box 564, Disneyland<br>

USA

</address>

</body>

</html>

# HTML Comments

HTML comments are not displayed in the browser, but they can help document your HTML source code.

## HTML Comment Tag

You can add comments to your HTML source by using the following syntax:

```
<!-- Write your comments here -->
```

Notice that there is an exclamation point (!) in the start tag, but not in the end tag

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<!-- This is a comment -->
```

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

```
<!-- Comments are not displayed in the browser -->
```

```
</body>
```

```
</html>
```

# Hide Content

Comments can be used to hide content.

Which can be helpful if you hide content temporarily:

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

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

```
<!--<p>This is another paragraph </p> -->
```

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

```
</body>
```

```
</html>
```

Comments are also great for debugging HTML, because you can comment out HTML lines of code, one at a time, to search for errors.

# HTML Links

Links are found in nearly all web pages. Links allow users to click their way from page to page.

# HTML Links - Hyperlinks

HTML links are hyperlinks.

You can click on a link and jump to another document.

When you move the mouse over a link, the mouse arrow will turn into a little hand.

**Note:** A link does not have to be text. A link can be an image or any other HTML element!

The most important attribute of the `<a>` element is the `href` attribute, which indicates the link's destination.

The *link text* is the part that will be visible to the reader.

Clicking on the link text, will send the reader to the specified URL address.

By default, links will appear as follows in all browsers:

- An unvisited link is underlined and blue
- A visited link is underlined and purple

## HTML Links - The target Attribute

By default, the linked page will be displayed in the current browser window. To change this, you must specify another target for the link.

The `target` attribute specifies where to open the linked document.

The `target` attribute can have one of the following values:

- `_self` - Default. Opens the document in the same window/tab as it was clicked
- `_blank` - Opens the document in a new window or tab

## Absolute URLs vs. Relative URLs

Both examples above are using an **absolute URL** (a full web address) in the `href` attribute.

A local link (a link to a page within the same website) is specified with a **relative URL** (without the "https://www" part):

```
<!DOCTYPE html>

<html>

<body>


<h2>Absolute URLs</h2>

<p><a href="https://www.google.com/">Google</a></p>


<h2>Relative URLs</h2>

<p><a href="html_images.asp">HTML Images</a></p>

<p><a href="/css/default.asp">CSS Tutorial</a></p>


</body>

</html>
```

## Link to an Email Address

Use **mailto:** inside the **href** attribute to create a link that opens the user's email program (to let them send a new email):

```
<!DOCTYPE html>

<html>

<body>


<h2>Link to an Email Address</h2>

<p>Contact us on <a href="mailto: admin@gmail.com
">admin@gmail.com</a></p>


</body>

</html>
```

Now if we want to add subject with mailto then code will be:

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Link to an Email Address</h2>
```

```
<p>Contact us on <a href="mailto:  
admin@gmail.com?subject=Hello ,how are you  
>admin@gmail.com</a></p>
```

```
</body>
```

```
</html>
```

## Mailto Parameters

The mailto attribute accepts several parameters, as described below:

- mailto: This parameter specifies the email address of the recipient.
- cc: This parameter is used to add another email address that will receive the mail's carbon copy. It is optional.
- bcc: This parameter specifies another email that will receive the blind carbon copy of the mail. It is optional.
- subject: This parameter is used to fill the subject of the mail. It is optional.

- body: This parameter is used to fill the content of the mail. It is optional.

```
<p>Contact us on <a  
href="mailto:admin@gmail.com?cc=admin1@gmail.com&bc  
c=admin2@gmail.com&subject=hello&body=Welcome to the  
organization">admin@gmail.com</a></p>
```

## Link Titles

The **title** attribute specifies extra information about an element. The information is most often shown as a tooltip text when the mouse moves over the element.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Link Titles</h2>
```

```
<p>The title attribute specifies extra information about an  
element. The information is most often shown as a tooltip  
text when the mouse moves over the element.</p>
```

```
<a href="https://www.google.com" title="Go to google  
search engine">Google search engine</a>
```

```
</body>
```



</html>

# HTML Links - Create Bookmarks

HTML links can be used to create bookmarks, so that readers can jump to specific parts of a web page.

## Create a Bookmark in HTML

Bookmarks can be useful if a web page is very long.

To create a bookmark - first create the bookmark, then add a link to it.

When the link is clicked, the page will scroll down or up to the location with the bookmark.

## Example

First, use the `id` attribute to create a bookmark:

```
<h2 id="C4">Chapter 4</h2>
```

Then, add a link to the bookmark ("Jump to Chapter 4"), from within the same page:

### Example

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<p><a href="#C4">Jump to Chapter 4</a></p>
```

<p><a href="#C10">Jump to Chapter 10</a></p>

<h2>Chapter 1</h2>

<p>This chapter explains bablabla</p>

<h2>Chapter 2</h2>

<p>This chapter explains bablabla</p>

<h2>Chapter 3</h2>

<p>This chapter explains bablabla</p>

<h2 id="C4">Chapter 4</h2>

<p>This chapter explains bablabla</p>

<h2>Chapter 5</h2>

<p>This chapter explains bablabla</p>

<h2>Chapter 6</h2>

<p>This chapter explains bablabla</p>

<h2>Chapter 7</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 8</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 9</h2>

<p>This chapter explains bablabla</p>

## <h2 id="C10">Chapter 10</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 11</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 12</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 13</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 14</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 15</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 16</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 17</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 18</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 19</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 20</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 21</h2>

<p>This chapter explains bablabla</p>

## <h2>Chapter 22</h2>

<p>This chapter explains bablabla</p>

<h2>Chapter 23</h2>

<p>This chapter explains bablabla</p>

</body>

</html>

You can also add a link to a bookmark on another page:

<a href="html\_demo.html#C4">Jump to Chapter 4</a>

# 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.

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

    <h2>HTML Image Example</h2>
    

  </body>
</html>
```

## 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.

### **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.

In any website right click any image then click on copy image address and then paste in url to get image url (suppose we want to copy image address from blockchain Wikipedia site)

### **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.

### **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.

## *height*

It is use to specify height of the image. It is not recommended now. We should apply CSS in place of height attribute.

## Use of height and width attribute with img tag

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

    <h2>HTML Image Example</h2>
    

  </body>
</html>
```

As we define image height and width both the image has been squished  
So, to prevent that from happening, we'll define either width or height

The **<img> align attribute** is used to *set the alignment of an image*. It is an inline element. It is used to specify the alignment of the image according to surrounding elements.

It is not supported by HTML 5. HTML 5 uses CSS property instead of this attribute.

### **Syntax:**

```
<img align="left|right">
```

```
<!DOCTYPE html>
<html>
<body>
  <h1>My Subjects</h1>
  <p>This is the text</p>
  
  <ol reversed="False">
    <li>Math</li>
    <li>English</li>
    <li>Hindi</li>
```

```
        <li>Science</li>
        <li>Chemistry</li>
    </ol>
</body>
</html>
```

## Marquee HTML

The <marquee> tag is a container tag of HTML is implemented for creating scrollable text or images within a web page from either left to right or vice versa, or top to bottom or vice versa. But this tag has been deprecated in the new version of HTML, i.e., HTML 5.

Example

```
<!DOCTYPE html>
<html>
    <body>
        <h2>Marquee Example</h2>
        <marquee>This is a moving text</marquee>

    </body>
</html>
```

## HTML Scroll Marquee

It is a by default property. It is used to scroll the text from right to left, and restarts at the right side of the marquee when it is reached to the end of left side.

```
<!DOCTYPE html>
<html>
    <body>
        <h2>Marquee Example</h2>
        <marquee width="100%" behavior="scroll" bgcolor="yellow">
            This is a moving text
        </marquee>

    </body>
</html>
```

## HTML Slide Marquee

In slide marquee, all the contents to be scrolled will slide the entire length of marquee but stops at the end to display the content permanently.

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



```
<body>
  <h2>Marquee Example</h2>
  <marquee width="100%" behavior="slide" bgcolor="yellow">
    This is a moving text
  </marquee>

</body>
</html>
```

## HTML Alternate Marquee

It scrolls the text from right to left and goes back left to right.

```
<!DOCTYPE html>
<html>
  <body>
    <h2>Marquee Example</h2>
    <marquee width="50%" behavior="alternate" bgcolor="yellow">
      This is a moving text
    </marquee>

  </body>
</html>
```

## Direction in HTML marquee

This is used to change the direction of scrolling text. Let's take an example of marquee scrolling to the right.

```
<!DOCTYPE html>
<html>
  <body>
    <h2>Marquee Example</h2>
    <marquee width="50%" direction="right" bgcolor="yellow" >
      This is a moving text
    </marquee>

  </body>
</html>
```

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

```
<body>
  <h2>Marquee Example</h2>
  <marquee direction="up" height="200" bgcolor="yellow">
    This is a moving text
  </marquee>

</body>
</html>
```

# HTML Lists

HTML lists allow web developers to group a set of related items in lists.

## Unordered HTML List

An unordered list starts with the `<ul>` tag. Each list item starts with the `<li>` tag.

The list items will be marked with bullets (small black circles) by default:

### Example

```
<!DOCTYPE html>

<html>

<body>


<h2>An unordered HTML list</h2>


<ul>
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>


</body>
```

```
</html>
```

## Ordered HTML List

An ordered list starts with the `<ol>` tag. Each list item starts with the `<li>` tag.

The list items will be marked with numbers by default:

### Example

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>An ordered HTML list</h2>
```

```
<ol>
```

```
<li>Coffee</li>
```

```
<li>Tea</li>
```

```
<li>Milk</li>
```

```
</ol>
```

```
</body>
```

```
</html>
```

# HTML Description Lists

HTML also supports description lists.

A description list is a list of terms, with a description of each term.

The `<dl>` tag defines the description list, the `<dt>` tag defines the term (name), and the `<dd>` tag describes each term (Dictionary is the example of description lists):

## Example

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>A Description List</h2>
```

```
<dl>
```

```
  <dt>Coffee</dt>
```

```
  <dd>- black hot drink</dd>
```

```
  <dt>Milk</dt>
```

```
  <dd>- white cold drink</dd>
```

```
</dl>
```

```
</body>
```

```
</html>
```

# Unordered HTML List - Choose List Item Marker

The CSS `list-style-type` property is used to define the style of the list item marker. It can have one of the following values:

Value	Description
disc	Sets the list item marker to a bullet (default)
circle	Sets the list item marker to a circle
square	Sets the list item marker to a square
none	The list items will not be marked

## Example - Disc

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Unordered List with Disc Bullets</h2>
```

```
<ul style="list-style-type:disc;">
```

```
<li>Coffee</li>
<li>Tea</li>
<li>Milk</li>
</ul>
```

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

## Example - Circle

```
<ul style="list-style-type:circle;">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

## Example - Square

```
<ul style="list-style-type:square;">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

## Example - None

```
<ul style="list-style-type:none;">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ul>
```

# Nested HTML Lists

Lists can be nested (list inside list):

## Example

```
<ul>
  <li>Coffee</li>
  <li>Tea
    <ul>
      <li>Black tea</li>
      <li>Green tea</li>
    </ul>
  </li>
  <li>Milk</li>
</ul>
```

## Ordered HTML List - The Type Attribute

The `type` attribute of the `<ol>` tag, defines the type of the list item marker:

Type	Description
type="1"	The list items will be numbered with numbers (default)
type="A"	The list items will be numbered with uppercase letters
type="a"	The list items will be numbered with lowercase letters
type="I"	The list items will be numbered with uppercase roman numbers

type="i"    The list items will be numbered with lowercase roman numbers

## Numbers:

```
<ol type="1">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

## Uppercase Letters:

```
<ol type="A">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

## Lowercase Letters:

```
<ol type="a">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

## Uppercase Roman Numbers:

```
<ol type="I">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```



## Lowercase Roman Numbers:

```
<ol type="i">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

## Control List Counting

By default, an ordered list will start counting from 1. If you want to start counting from a specified number, you can use the `start` attribute:

### Example

```
<ol start="50">
  <li>Coffee</li>
  <li>Tea</li>
  <li>Milk</li>
</ol>
```

The **HTML <ol> reversed Attribute** is a Boolean Attribute and used to *ordered the list in Descending Order(9, 8, 7, 6 .....)* instead of ascending order(1, 2, 3 ....).

```
<!DOCTYPE html>
<html>
<body>
  <h1>My Subjects</h1>
  <ol reversed>
    <li>Math</li>
    <li>English</li>
    <li>Hindi</li>
    <li>Science</li>
    <li>Chemistry</li>
  </ol>
</body>
</html>
```

# Nested HTML Lists

Lists can be nested (list inside list):

## Example

```
<ol>
  <li>Coffee</li>
  <li>Tea
    <ol>
      <li>Black tea</li>
      <li>Green tea</li>
    </ol>
  </li>
  <li>Milk</li>
</ol>
```

# HTML Tables

**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.

## Define an HTML Table

A table in HTML consists of table cells inside rows and columns

## Example

A simple HTML table:

```
<!DOCTYPE>

<html>

<body>

<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>

</body>

</html>
```

## HTML Table with Border

There are two ways to specify border for HTML tables.

1. By border attribute of table in HTML
  2. 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. border attribute is not supported by HTML 5 (depreciated)

```
<!DOCTYPE html>
<html>
  <body>
    <h2>Table Example</h2>
    <table border="">

      <tr>
        <th>Subject</th>
        <th>Marks</th>
        <th>Passed</th>
      </tr>

      <tr>
        <td>Math</td>
        <td>90</td>
        <td>Yes</td>
      </tr>
      <tr>
        <td>Hindi</td>
        <td>80</td>
        <td>Yes</td>
      </tr>
      <tr>
        <td>English</td>
        <td>30</td>
        <td>No</td>
      </tr>

    </table>

  </body>
</html>
```

The outside border is of table and inside border of <td>(table data)

```
<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>
```

## In Table we can have image hyperlink etc

```
<!DOCTYPE html>
<html>
  <body>
    <h2>Table Example</h2>
    <table border="1">
      <tr>
        <th>Col 1</th>
        <th>Col 2</th>
        <th>Col 3</th>
      </tr>
      <tr>
        <td>Data 1</td>
        <td></td>
        <td><a href="https://www.google.com" target="_blank">google.com</a></td>
      </tr>
    </table>

  </body>
</html>
```

## 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>**

# Table Cells

Each table cell is defined by a `<td>` and a `</td>` tag.

`td` stands for table data.

Everything between `<td>` and `</td>` are the content of the table cell.

## Example

```
<table>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
</table>
```

**Note:** table data elements are the data containers of the table. They can contain all sorts of HTML elements; text, images, lists, other tables, etc.

# Table Rows

Each table row starts with a `<tr>` and end with a `</tr>` tag.

`tr` stands for table row.

## Example

```
<table>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
  <tr>
    <td>16</td>
    <td>14</td>
    <td>10</td>
  </tr>
</table>
```

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

You can have as many rows as you like in a table, just make sure that the number of cells are the same in each row.

## Style Table Borders

If you set a background color of each cell, and give the border a white color (the same as the document background), you get the impression of an invisible border:

```
<!DOCTYPE html>  
  
<html>  
  
<head>  
  
<style>  
table, th, td {  
    border: 1px solid white;  
    border-collapse: collapse;  
}  
th, td {  
    background-color: #96D4D4;  
}  
</style>  
</head>  
  
<body>  
  
<h2>Table With Invisible Borders</h2>
```

```
<p>Style the table with white borders and a background color of the cells to make the
```



impression of invisible borders.</p>

```
<table style="width:100%">
```

```
<tr>
```

```
<th>Firstname</th>
```

```
<th>Lastname</th>
```

```
<th>Age</th>
```

```
</tr>
```

```
<tr>
```

```
<td>Jill</td>
```

```
<td>Smith</td>
```

```
<td>50</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Eve</td>
```

```
<td>Jackson</td>
```

```
<td>94</td>
```

```
</tr>
```

```
<tr>
```

```
<td>John</td>
```

```
<td>Doe</td>
```

```
<td>80</td>
```

```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

# Round Table Borders

With the `border-radius` property, the borders get rounded corners:

```
<!DOCTYPE html>
<html>
<head>
<style>
table, th, td {
  border: 1px solid black;
  border-radius: 10px;
}
</style>
</head>
<body>
```

```
<h2>Table With Rounded Borders</h2>
```

```
<p>Use the CSS border-radius property to add rounded corners to the borders.</p>
```

```
<table style="width:100%">
<tr>
  <th>Firstname</th>
  <th>Lastname</th>
  <th>Age</th>
</tr>
<tr>
  <td>Jill</td>
  <td>Smith</td>
  <td>50</td>
</tr>
<tr>
  <td>Eve</td>
  <td>Jackson</td>
```

```
<td>94</td>
</tr>
<tr>
  <td>John</td>
  <td>Doe</td>
  <td>80</td>
</tr>
</table>

</body>
</html>
```

Skip the border around the table by leaving out **table** from the css selector:

```
<!DOCTYPE html>
<html>
<head>
<style>
th, td {
  border: 1px solid black;
  border-radius: 10px;
}
</style>
</head>
<body>
```

```
<h2>Table With Rounded Borders</h2>
```

```
<p>Use the CSS border-radius property to add rounded corners to the table cells.</p>
```

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr>
```

```
<td>Jill</td>
<td>Smith</td>
<td>50</td>
</tr>
<tr>
<td>Eve</td>
<td>Jackson</td>
<td>94</td>
</tr>
<tr>
<td>John</td>
<td>Doe</td>
<td>80</td>
</tr>
</table>

</body>
</html>
```

## Border Color

With the `border-color` property, you can set the color of the border.

```
<!DOCTYPE html>

<html>

<head>

<style>

table,th, td {

    border-style:solid;

    border-color: #96D4D4;
```

```
border-collapse: collapse;
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>Table With Border Color</h2>
```

```
<p>Use the CSS border-color property to set the color of the
borders.</p>
```

```
<table style="width:100%">
```

```
<tr>
```

```
<th>Firstname</th>
```

```
<th>Lastname</th>
```

```
<th>Age</th>
```

```
</tr>
```

```
<tr>
```

```
<td>Jill</td>
```

```
<td>Smith</td>
```

```
<td>50</td>
```

```
</tr>
```

```
<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td>94</td>
</tr>
<tr>
  <td>John</td>
  <td>Doe</td>
  <td>80</td>
</tr>
</table>

</body>
</html>
```

# HTML Table Sizes

HTML tables can have different sizes for each column, row or the entire table.

Use the **style** attribute with the **width** or **height** properties to specify the size of a table, row or column.

## HTML Table Width

To set the width of a table, add the **style** attribute to the **<table>** element:

```
<!DOCTYPE html>

<html>

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

<body>

<h2>100% wide HTML Table</h2>

<table style="width:100%">
    <tr>
        <th>Firstname</th>
        <th>Lastname</th>
        <th>Age</th>
    </tr>
    <tr>
        <td>Jill</td>
        <td>Smith</td>
```

```
<td>50</td>
</tr>
<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td>94</td>
</tr>
<tr>
  <td>John</td>
  <td>Doe</td>
  <td>80</td>
</tr>
</table>

</body>
</html>
```

## HTML Table Column Width

To set the size of a specific column, add the `style` attribute on a `<th>` or `<td>` element:

```
<!DOCTYPE html>
```



```
<html>

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

<body>

<h2>Set the first column to 70% of the table width</h2>

<table style="width:100%">
    <tr>
        <th style="width:70%">Firstname</th>
        <th>Lastname</th>
        <th>Age</th>
    </tr>
    <tr>
        <td>Jill</td>
        <td>Smith</td>
        <td>50</td>
    </tr>
    <tr>
        <td>Eve</td>
        <td>Jackson</td>
```

```
<td>94</td>

</tr>

<tr>

  <td>John</td>

  <td>Doe</td>

  <td>80</td>

</tr>

</table>

</body>

</html>
```

## HTML Table Row Height

To set the height of a specific row, add the **style** attribute on a table row element:

```
<!DOCTYPE html>

<html>

<style>

table, th, td {

  border: 1px solid black;

  border-collapse: collapse;

}

</style>
```

```
<body>
```

```
<h2>Set the height of the second row to 200 pixels</h2>
```

```
<table style="width:100%">
```

```
  <tr>
```

```
    <th>Firstname</th>
```

```
    <th>Lastname</th>
```

```
    <th>Age</th>
```

```
  </tr>
```

```
  <tr style="height:100px">
```

```
    <td>Jill</td>
```

```
    <td>Smith</td>
```

```
    <td>50</td>
```

```
  </tr>
```

```
  <tr>
```

```
    <td>Eve</td>
```

```
    <td>Jackson</td>
```

```
    <td>94</td>
```

```
  </tr>
```

```
  <tr>
```

```
    <td>John</td>
```

```
    <td>Doe</td>
```

```
    <td>80</td>
```

```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

## Vertical Table Headers

To use the first column as table headers, define the first cell in each row as a **th** element:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
table, th, td {
```

```
    border: 1px solid black;
```

```
    border-collapse: collapse;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>Vertical Table Headers</h2>
```

<p>The first column becomes table headers if you set the first table cell in each table row to a TH element:</p>

```
<table style="width:100%">
```

```
<tr>
```

```
<th>Firstname</th>
```

```
<td>Jill</td>
```

```
<td>Eve</td>
```

```
</tr>
```

```
<tr>
```

```
<th>Lastname</th>
```

```
<td>Smith</td>
```

```
<td>Jackson</td>
```

```
</tr>
```

```
<tr>
```

```
<th>Age</th>
```

```
<td>50</td>
```

```
<td>94</td>
```

```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

# Align Table Headers

By default, table headers are bold and centered:

Firstname	Lastname	Age
Jill	Smith	50
Eve	Jackson	94

To left-align the table headers, use the CSS `text-align` property:

```
<!DOCTYPE html>

<html>

<head>

<style>

table, th, td {

    border: 1px solid black;

    border-collapse: collapse;

}

th {

    text-align: left;

}

</style>

</head>

<body>


<h2>Left-align Headers</h2>
```

<p>To left-align the table headers, use the CSS text-align property.</p>

```
<table style="width:100%">
```

```
<tr>
```

```
<th>Firstname</th>
```

```
<th>Lastname</th>
```

```
<th>Age</th>
```

```
</tr>
```

```
<tr>
```

```
<td>Jill</td>
```

```
<td>Smith</td>
```

```
<td>50</td>
```

```
</tr>
```

```
<tr>
```

```
<td>Eve</td>
```

```
<td>Jackson</td>
```

```
<td>94</td>
```

```
</tr>
```

```
</table>
```

```
</body>
```

```
</html>
```

# Header for Multiple Columns

You can have a header that spans over two or more columns.

Name		Age
Jill	Smith	50
Eve	Jackson	94

To do this, use the `colspan` attribute on the `<th>` element:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
table, th, td {
```

```
    border: 1px solid black;
```

```
    border-collapse: collapse;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>A header that spans two columns</h2>
```

```
<p>Use the colspan attribute to have a header span over multiple  
columns.</p>
```

```
<table style="width:100%">
```



```

<tr>
  <th colspan="2">Name</th>
  <th>Age</th>
</tr>
<tr>
  <td>Jill</td>
  <td>Smith</td>
  <td>50</td>
</tr>
<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td>94</td>
</tr>
</table>
</body>
</html>

```

## Table Caption

You can add a caption that serves as a heading for the entire table.

Monthly savings

Month	Savings
January	\$100

February	\$50
----------	------

To add a caption to a table, use the `<caption>` tag:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
table, th, td {
```

```
    border: 1px solid black;
```

```
    border-collapse: collapse;
```

```
}
```

```
th, td {
```

```
    padding: 5px;
```

```
    text-align: left;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>Table Caption</h2>
```

```
<p>To add a caption to a table, use the caption tag.</p>
```

```
<table style="width:100%">
```

```
    <caption>Monthly savings</caption>
```

```
    <tr>
```

```
<th>Month</th>
<th>Savings</th>
</tr>
<tr>
  <td>January</td>
  <td>$100</td>
</tr>
<tr>
  <td>February</td>
  <td>$50</td>
</tr>
</table>

</body>
</html>
```

**Note:** The `<caption>` tag should be inserted immediately after the `<table>` tag.

# HTML Table Padding & Spacing

HTML tables can adjust the padding inside the cells, and also the space between the cells.

## HTML Table - Cell Padding

Cell padding is the space between the cell edges and the cell content.

By default the padding is set to 0.

To add padding on table cells, use the CSS `padding` property:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
table, th, td {
```

```
    border: 1px solid black;
```

```
    border-collapse: collapse;
```

```
}
```

```
th, td {
```

```
    padding: 15px;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>Cellpadding</h2>
```

```
<p>Cell padding specifies the space between the cell content and its borders.</p>
```

```
<table style="width:100%">
```

```
  <tr>
```

```
    <th>Firstname</th>
```

```
    <th>Lastname</th>
```

```
    <th>Age</th>
</tr>

<tr>

    <td>Jill</td>

    <td>Smith</td>

    <td>50</td>

</tr>

<tr>

    <td>Eve</td>

    <td>Jackson</td>

    <td>94</td>

</tr>

<tr>

    <td>John</td>

    <td>Doe</td>

    <td>80</td>

</tr>
</table>
```

<p><strong>Tip:</strong> Try to change the padding to 5px.</p>

```
</body>

</html>
```

To add padding only above the content, use the **padding-top** property.

And the others sides with the `padding-bottom`, `padding-left`, and `padding-right` properties:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
table, th, td {
```

```
    border: 1px solid black;
```

```
    border-collapse: collapse;
```

```
}
```

```
th, td {
```

```
    padding-top: 10px;
```

```
    padding-bottom: 20px;
```

```
    padding-left: 30px;
```

```
    padding-right: 40px;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>Cellpadding - top - bottom - left - right </h2>
```

```
<p>We can specify different padding for all fours sides of the cell  
content.</p>
```

```
<table style="width:100%">
```

```
<tr>
  <th>Firstname</th>
  <th>Lastname</th>
  <th>Age</th>
</tr>
<tr>
  <td>Jill</td>
  <td>Smith</td>
  <td>50</td>
</tr>
<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td>94</td>
</tr>
<tr>
  <td>John</td>
  <td>Doe</td>
  <td>80</td>
</tr>
</table>

</body>
</html>
```

# HTML Table - Cell Spacing

Cell spacing is the space between each cell.

By default the space is set to 2 pixels.

To change the space between table cells, use the CSS `border-spacing` property on the `table` element:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
table, th, td {
```

```
    border: 1px solid black;
```

```
}
```

```
table {
```

```
    border-spacing: 30px;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>Cellspacing</h2>
```

```
<p>Change the space between the cells with the border-spacing  
property.</p>
```

```
<table style="width:100%">
```

```
    <tr>
```



```
<th>Firstname</th>
<th>Lastname</th>
<th>Age</th>
</tr>
<tr>
  <td>Jill</td>
  <td>Smith</td>
  <td>50</td>
</tr>
<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td>94</td>
</tr>
<tr>
  <td>John</td>
  <td>Doe</td>
  <td>80</td>
</tr>
</table>

</body>
</html>
```

# HTML Table - Rowspan

To make a cell span over multiple rows, use the `rowspan` attribute:

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
table, th, td {
```

```
    border: 1px solid black;
```

```
    border-collapse: collapse;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>Cell that spans two rows</h2>
```

```
<p>To make a cell span more than one row, use the rowspan attribute.</p>
```

```
<table style="width:100%">
```

```
  <tr>
```

```
    <th>Name</th>
```

```
    <td>Jill</td>
```

```
  </tr>
```

```
  <tr>
```

```
    <th rowspan="2">Phone</th>
```

<td>555-1234</td>

</tr>

<tr>

<td>555-8745</td>

</tr>

</table>

</body>

</html>

# HTML Table Styling

## HTML Table - Zebra Stripes

If you add a background color on every other table row, you will get a nice zebra stripes effect.

To style every other table row element, use the `:nth-child(even)` selector like this:

```
<!DOCTYPE html>

<html>

<head>

<style>

table {

    border-collapse: collapse;

    width: 100%;

}

th, td {

    text-align: left;

    padding: 8px;

}

tr:nth-child(even) {

    background-color: #D6EEEE;

}

</style>
```

```
</head>
```

```
<body>
```

```
<h2>Zebra Striped Table</h2>
```

```
<p>For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:</p>
```

```
<table>
```

```
  <tr>
```

```
    <th>First Name</th>
```

```
    <th>Last Name</th>
```

```
    <th>Points</th>
```

```
  </tr>
```

```
  <tr>
```

```
    <td>Peter</td>
```

```
    <td>Griffin</td>
```

```
    <td>$100</td>
```

```
  </tr>
```

```
  <tr>
```

```
    <td>Lois</td>
```

```
    <td>Griffin</td>
```

```
    <td>$150</td>
```

```
  </tr>
```

```
  <tr>
```

```
    <td>Joe</td>
```

```
<td>Swanson</td>

<td>$300</td>

</tr>

<tr>

<td>Cleveland</td>

<td>Brown</td>

<td>$250</td>

</tr>

</table>

</body>

</html>
```

**Note:** If you use `(odd)` instead of `(even)`, the styling will occur on row 1,3,5 etc. instead of 2,4,6 etc.

## HTML Table - Vertical Zebra Stripes

To make vertical zebra stripes, style every other *column*, instead of every other *row*.

Set the `:nth-child(even)` for table data elements like this:

### Example

```
td:nth-child(even), th:nth-child(even) {
  background-color: #D6EEEE;
}
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

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

th:nth-child(even),td:nth-child(even) {
    background-color: #D6EEEE;
}
</style>
</head>
<body>
```

```
<h2>Striped Table</h2>
```

<p>For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:</p>

```
<table style="width:100%">
  <tr>
    <th>MON</th>
    <th>TUE</th>
    <th>WED</th>
    <th>THU</th>
    <th>FRI</th>
    <th>SAT</th>
```

<th>SUN</th>

</tr>

<tr>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

</tr>

<tr>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

</tr>

<tr>

<td> </td>

<td> </td>

<td> </td>

<td> </td>



```
<td> </td>

<td> </td>

<td> </td>

</tr>

<tr>

  <td> </td>

  <td> </td>

  <td> </td>

  <td> </td>

  <td> </td>

  <td> </td>

  <td> </td>

  <td> </td>

</tr>

</table>

</body>

</html>
```

**Note:** Put the `:nth-child()` selector on both `th` and `td` elements if you want to have the styling on both headers and regular table cells.

## Combine Vertical and Horizontal Zebra Stripes

You can combine the styling from the two examples above and you will have stripes on every other row and every other column.

## Example

```
tr:nth-child(even) {
  background-color: #D6EEEE;
}

th:nth-child(even),td:nth-child(even) {
  background-color: #D6EEEE;
}

<!DOCTYPE html>

<html>

<head>

<style>

table, th, td {

  border: 1px solid black;

  border-collapse: collapse;

}

tr:nth-child(even) {

  background-color: rgba(150, 212, 212, 0.4);

}

th:nth-child(even),td:nth-child(even) {

  background-color: rgba(150, 212, 212, 0.4);

}

</style>

</head>

<body>
```

## <h2>Striped Table</h2>

<p>For zebra-striped tables, use the nth-child() selector and add a background-color to all even (or odd) table rows:</p>

```
<table style="width:100%">
```

```
<tr>
```

```
<th>MON</th>
```

```
<th>TUE</th>
```

```
<th>WED</th>
```

```
<th>THU</th>
```

```
<th>FRI</th>
```

```
<th>SAT</th>
```

```
<th>SUN</th>
```

```
</tr>
```

```
<tr>
```

```
<td> </td>
```

```
<td> </td>
```

```
<td> </td>
```

```
<td> </td>
```

```
<td> </td>
```

```
<td> </td>
```

```
<td> </td>
```

```
</tr>
```

```
<tr>
```

```
<td> </td>
```

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

</tr>

<tr>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

</tr>

<tr>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

<td> </td>

</tr>

```
</table>
```

```
</body>
```

```
</html>
```

## Horizontal Dividers

First Name	Last Name	Savings
Peter	Griffin	\$100
Lois	Griffin	\$150
Joe	Swanson	\$300

If you specify borders only at the bottom of each table row, you will have a table with horizontal dividers.

Add the `border-bottom` property to all `tr` elements to get horizontal dividers:

### Example

```
tr {  
  border-bottom: 1px solid #ddd;  
}
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>

<style>

table {

    border-collapse: collapse;

    width: 100%;

}

tr {

    border-bottom: 1px solid #ddd;

}

</style>

</head>

<body>
```

```
<h2>Bordered Table Dividers</h2>
```

```
<p>Add the border-bottom property to the tr elements for horizontal
dividers:</p>
```

```
<table>

<tr>

    <th>Firstname</th>

    <th>Lastname</th>

    <th>Savings</th>

</tr>

<tr>
```

```
<td>Peter</td>
<td>Griffin</td>
<td>$100</td>
</tr>
<tr>
<td>Lois</td>
<td>Griffin</td>
<td>$150</td>
</tr>
<tr>
<td>Joe</td>
<td>Swanson</td>
<td>$300</td>
</tr>
<tr>
<td>Cleveland</td>
<td>Brown</td>
<td>$250</td>
</tr>
</table>

</body>
</html>
```

# Hoverable Table

Use the `:hover` selector on `tr` to highlight table rows on mouse over:

## Example

```
tr:hover {background-color: #D6EEEE;}
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
table {
```

```
    border-collapse: collapse;
```

```
    width: 100%;
```

```
}
```

```
th, td {
```

```
    padding: 8px;
```

```
    text-align: left;
```

```
    border-bottom: 1px solid #DDD;
```

```
}
```

```
tr:hover {background-color: #D6EEEE;}
```

```
</style>
```

```
</head>
```

```
<body>
```



<h2>Hoverable Table</h2>

<p>Move the mouse over the table rows to see the effect.</p>

<table>

<tr>

<th>First Name</th>

<th>Last Name</th>

<th>Points</th>

</tr>

<tr>

<td>Peter</td>

<td>Griffin</td>

<td>\$100</td>

</tr>

<tr>

<td>Lois</td>

<td>Griffin</td>

<td>\$150</td>

</tr>

<tr>

<td>Joe</td>

<td>Swanson</td>

<td>\$300</td>

</tr>

<tr>

```
<td>Cleveland</td>

<td>Brown</td>

<td>$250</td>

</tr>

</table>

</body>

</html>
```

# HTML Table Colgroup

The `<colgroup>` element is used to style specific columns of a table.

## HTML Table Colgroup

If you want to style the two first columns of a table, use the `<colgroup>` and `<col>` elements.

The `<colgroup>` element should be used as a container for the column specifications.

Each group are specified with a `<col>` element.

The `span` attribute specifies how many columns that gets the style.

The `style` attribute specifies the style to give the columns.

## Example

```
<table>
  <colgroup>
    <col span="2" style="background-color: #D6EEEE">
  </colgroup>
  <tr>
```

```
<th>MON</th>
<th>TUE</th>
<th>WED</th>
<th>THU</th>
```

...

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<style>
```

```
table, th, td {
```

```
    border: 1px solid black;
```

```
    border-collapse: collapse;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h2>Colgroup</h2>
```

<p>Add the a colgroup with a col element that spans over two columns to define a style for the two columns:</p>

```
<table style="width: 100%;">
```

```
<colgroup>
```

```
    <col span="2" style="background-color: #D6EEEE">
```

```
</colgroup>
```

```
<tr>
```

```
<th>MON</th>
```

<th>TUE</th>

<th>WED</th>

<th>THU</th>

<th>FRI</th>

<th>SAT</th>

<th>SUN</th>

</tr>

<tr>

<td>1</td>

<td>2</td>

<td>3</td>

<td>4</td>

<td>5</td>

<td>6</td>

<td>7</td>

</tr>

<tr>

<td>8</td>

<td>9</td>

<td>10</td>

<td>11</td>

<td>12</td>

<td>13</td>

<td>14</td>

</tr>

```
<tr>
<td>15</td>
<td>16</td>
<td>17</td>
<td>18</td>
<td>19</td>
<td>20</td>
<td>21</td>
</tr>
<tr>
<td>22</td>
<td>23</td>
<td>24</td>
<td>25</td>
<td>26</td>
<td>27</td>
<td>28</td>
</tr>
</table>

</body>

</html>
```

**Note:** The `<colgroup>` tag must be a child of a `<table>` element and should be placed before any other table elements, like `<thead>`, `<tr>`, `<td>` etc., but after the `<caption>` element, if present.

# Multiple Col Elements

If you want to style more columns with different styles, use more `<col>` elements inside the `<colgroup>`:

## Example

```
<table>
  <colgroup>
    <col span="2" style="background-color: #D6EEEE">
    <col span="3" style="background-color: pink">
  </colgroup>
  <tr>
    <th>MON</th>
    <th>TUE</th>
    <th>WED</th>
    <th>THU</th>
  ...
```

# Empty Colgroups

If you want to style columns in the middle of a table, insert a "empty" `<col>` element (with no styles) for the columns before:

## Example

```
<table>
  <colgroup>
    <col span="3">
    <col span="2" style="background-color: pink">
  </colgroup>
  <tr>
    <th>MON</th>
    <th>TUE</th>
    <th>WED</th>
    <th>THU</th>
  ...
```

## HTML <thead> tag

HTML <thead> elements is used to define header of an HTML table. The <thead> tag is used along with <tbody> and <tfoot> tags which defines table header, table body, and table footer in an HTML table.

The <thead> tag must be child of <table> element, and it must be used before any <tbody>, <tr>, or <tfoot> elements.

The <thead> tag should contain at least one row <tr> element inside it.

## HTML <tbody> tag

HTML <tbody> tag is used to group the table rows (<tr>) together, which indicates that this is body part of a table (<table>).

The <tbody> tag must be a child of <table> element.

The <tbody> is used along with <thead> and <tfoot> which shows the different part of the table that are table head, table body, and table footer, however, it does not affect the layout of the table.

## HTML <tfoot> tag

HTML <tfoot> tag is used to define the set of rows which represents footer of an HTML table. The <tfoot> tag must contain one or more <tr> element.

The <tfoot> tag is used as a child element of HTML table (<table>) along with <thead> and <tbody> elements, where <thead> defines table header and <tbody> defines the table body.

The table look similar with <thead> <tbody> and <tfoot> and with these tags

## So What is difference between simple <table> and <table> with <thead> <tbody> and <tfoot>

- Difference is just that we can use them (table content) separately in a proper way
- We can assign different behaviours to the section in the table
- So, this is just a good practice, but not a required practice

```
<!DOCTYPE html>
<html>
  <body>
    <h2>Table Example</h2>
    <table border="1">
      <thead style="color:red">
        <tr>
          <th>Column 1</th>
          <th>Column 2</th>
          <th>Column 3</th>
        </tr>
      </thead>
      <tbody style="color:green">
        <tr>
          <td>Data 1</td>
          <td>Data 2</td>
          <td>Data 3</td>
        </tr>
        <tr>
          <td>Data 4</td>
          <td>Data 5</td>
          <td>Data 6</td>
        </tr>
      </tbody>
    </table>
  </body>
</html>
```



```

        </tr>
        <tr>
            <td>Data 7</td>
            <td>Data 8</td>
            <td>Data 9</td>
        </tr>
    </tbody>
    <tfoot style="color:blue">
        <tr>
            <th>Column 1</th>
            <th>Column 2</th>
            <th>Column 3</th>
        </tr>
    </tfoot>
</table>
</body>
</html>

```

### Another example of <thead> <tbody> <tfoot>

```

<!DOCTYPE html>
<html>
    <head>
        <style>
            th {
                text-align: left;
            }
            table, th, td {
                border: 1px solid black;
                border-collapse: collapse;
            }

        </style>
    </head>
    <body>
        <table border="1">
            <caption>A complex table</caption>
            <thead>
                <tr>
                    <th colspan="3">
                        Invoice #12345789
                    </th>
                    <th>14 January 2025</th>
                </tr>
                <tr>
                    <td colspan="2">

```

```

        <strong>Pay to:</strong><br>
        Acmc Billing Co.<br>
        123 Main St. <br>
        CityVille,NA 12345

</td>
<td colspan="2">
    <strong>Customer:</strong><br>
    John Smith<br>
    321 Willow Way <br>
    Southcast Northwestershire, MA 54321

</td>
</tr>
</thead>
<tbody>
<tr>
    <th>Name/Description</th>
    <th>Qty.</th>
    <th>@</th>
    <th>Cost</th>

</tr>
<tr>
    <td>Paperclips</td>
    <td>1000</td>
    <td>0.01</td>
    <td>10.00</td>
</tr>
<tr>
    <td>Staples (box)</td>
    <td>100</td>
    <td>1.00</td>
    <td>100.00</td>
</tr>
</tbody>
<tfoot>
<tr>
    <th colspan="3">Subtotal</th>
    <td>110.00</td>
</tr>
<tr>
    <th colspan="2">Tax</th>
    <td>8%</td>
    <td>8.80</td>
</tr>
<tr>

```

```
<th colspan="3">Grand Total</th>
<td>$118.80</td>

</tr>
</tfoot>

</table>

</body>
</html>
```

Every HTML element has a default display value, depending on what type of element it is.

There are two display values: block and inline.

## Block-level Elements

A block-level element always starts on a new line, and the browsers automatically add some space (a margin) before and after the element.

A block-level element always takes up the full width available (stretches out to the left and right as far as it can).

Two commonly used block elements are: `<p>` and `<div>`.

The `<p>` element defines a paragraph in an HTML document.

The `<div>` element defines a division or a section in an HTML document.

The `<p>` element is a block-level element.

The `<div>` element is a block-level element.

### Example

```
<!DOCTYPE html>

<html>

<body>
```

```
<p style="border: 1px solid black">Hello World</p>
```

```
<div style="border: 1px solid black">Hello World</div>
```

<p>The P and the DIV elements are both block elements, and they will always start on a new line and take up the full width available (stretches out to the left and right as far as it can).</p>

```
</body>
```

```
</html>
```

## Inline Elements

An inline element does not start on a new line.

An inline element only takes up as much width as necessary.

**Note:** An inline element cannot contain a block-level element!

## HTML <span> Tag

### Example

A <span> element which is used to color a part of a text:

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1>The span element</h1>
```

```
<p>My mother has <span style="color:blue;font-weight:bold">blue</span>  
eyes and my father has <span style="color:darkolivegreen;font-  
weight:bold">dark green</span> eyes.</p>
```

</body>

</html>

## Definition and Usage

The `<span>` tag is an inline container used to mark up a part of a text, or a part of a document. `<b>`, `<u>`, `<i>` are inline element

The `<span>` tag is easily styled by CSS or manipulated with JavaScript using the class or id attribute.

The `<span>` tag is much like the `<div>` element, but `<div>` is a block-level element and `<span>` is an inline element.

# Div Tag

The div tag is known as Division tag. The div tag is used in HTML to make divisions of content in the web page like (text, images, header, footer, navigation bar, etc). Div tag has both open (<div>) and closing (</div>) tag and it is mandatory to close the tag. The Div is the most usable tag in web development because it helps us to separate out data in the web page and we can create a particular section for particular data or function in the web pages.

- Div tag is Block level tag
- Div never create a design
- For designing we use css with div

## Example

```
<!DOCTYPE html>

<html>

<body>


<div style="background-color:black;color:white;padding:20px;">

    <h2>London</h2>

    <p>London is the capital city of England. It is the most populous city in the United Kingdom, with a metropolitan area of over 13 million inhabitants.</p>

    <p>Standing on the River Thames, London has been a major settlement for two millennia, its history going back to its founding by the Romans, who named it Londinium.</p>

</div>


</body>

</html>
```

## How to float div side by side using CSS?

Three or more different div can be put side-by-side using CSS. Use CSS property to set the height and width of div and use display property to place div in side-by-side format.

- **float:left;** This property is used for those elements(div) that will float on left side.
- **float:right;** This property is used for those elements(div) that will float on right side.

## Table using Div tags

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Table using Div</title>
</head>
<body>

  <div style="height:200px;width:1305px;border:1px solid black">
    <div style="float:left;height:200px;width:198px;border:1px solid
black;background-color:purple ;"></div>
    <!--first make height of first 4 div to 100px and then subtract whole
width minus width of the square i.e. 1305-200 and divide the result by
4 (because we need four similar square )the result will be 276.25 but each
div take border left right(2px)
so we have to minus 2px from each width also from first div not width of
the first div will be 198
To display each block side by side we use css property float:left in every
div-->
    <!-- <div style="float:left;height:100px;width:274.25px;border:1px solid
black;background-color: red;"></div>
    <div style="float:left;height:100px;width:274.25px;border:1px solid
black;background-color: green;"></div>
    <div style="float:left;height:100px;width:274.25px;border:1px solid
black;background-color: yellow;"></div>
```





```

    <div style="float:left;height:98px;width:259px;border:1px solid
black;background-color: green;"></div>
    <div style="float:left;height:98px;width:259px;border:1px solid
black;background-color: rgb(24, 129, 199);"></div>
    <div style="float:left;height:98px;width:259px;border:1px solid
black;background-color: black;"></div>
    <div style="float:left;height:98px;width:259px;border:1px solid
black;background-color: olive;"></div>

</div>
<!-- Now add another row (div) -->
<div style="height:98px;width:1305px;border:1px solid black;background-color:
cyan;">
    <!-- Now we have to add 3 div 1105 divide by 3 i.e. 435-2 =433 -->
    <div style="float:left;height:98px;width:433px;border:1px solid
black;background-color: cyan;"></div>
    <div style="float:left;height:98px;width:433px;border:1px solid
black;background-color: cyan;"></div>
    <div style="float:left;height:98px;width:433px;border:1px solid
black;background-color: cyan;"></div>
</div>
</body>
</html>

```

## Center a div using margin auto

So in **margin: 0 auto**, the top/bottom margin is 0, and the left/right margin is auto, Where auto means that the left and right margins are automatically set by the browser based on the container, to make the element centered.

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Div center</title>
    <style>
        .outer{
            height:700px;
            width:700px;
            border:1px solid red;
            margin:0px auto; /* to make div center */
        }
    </style>

```

```
</head>
<body>
  <div class="outer">
  </div>
</body>
</html>
```

# HTML Forms

An HTML form is used to collect user input. The user input is most often sent to a server for processing.

## The `<form>` Element

The HTML `<form>` element is used to create an HTML form for user input:

```
<form>
.
form elements
.
</form>
```

The `<form>` element is a container for different types of input elements, such as: text fields, checkboxes, radio buttons, submit buttons, etc.

## The `<input>` Element

The HTML `<input>` element is the most used form element. It is an inline element.

An `<input>` element can be displayed in many ways, depending on the `type` attribute.

Here are some examples:

Type	Description
<code>&lt;input type="text"&gt;</code>	Displays a single-line text input field . (by default its size is of 30 characters)
<code>&lt;input type="radio"&gt;</code>	Displays a radio button (for selecting one of many choices)
<code>&lt;input type="checkbox"&gt;</code>	Displays a checkbox (for selecting zero or more of many choices)
<code>&lt;input type="submit"&gt;</code>	Displays a submit button (for submitting the form)
<code>&lt;input type="button"&gt;</code>	Displays a clickable button

## Text Fields

The `<input type="text">` defines a single-line input field for text input.

### Example

A form with input fields for text:

```
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname">
</form>
```

```
<!DOCTYPE html>

<html>

<body>


<h2>Text input fields</h2>


<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe">
</form>


<p>Note that the form itself is not visible.</p>


<p>Also note that the default width of text input fields is 30
characters.</p>


</body>
</html>
```

## The <label> Element

Notice the use of the `<label>` element in the example above.

The `<label>` tag defines a label for many form elements.

The `for` attribute of the `<label>` tag should be equal to the `id` attribute of the `<input>` element to bind them together.

The `<label>` element also help users who have difficulty clicking on very small regions (such as radio buttons or checkboxes) - because when the user clicks the text within the `<label>` element, it toggles the radio button/checkbox.

## The Submit Button

The `<input type="submit">` defines a button for submitting the form data to a form-handler.

The form-handler is typically a file on the server with a script for processing input data.

The form-handler is specified in the form's `action` attribute.

### Example

A form with a submit button:

```
<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe"><br><br>
  <input type="submit" value="Submit">
</form>
```

## The Name Attribute for `<input>`

Notice that each input field must have a `name` attribute to be submitted.

If the `name` attribute is omitted, the value of the input field will not be sent at all.

### Example

This example will not submit the value of the "First name" input field:

```
<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" value="John"><br><br>
  <input type="submit" value="Submit">
</form>
```

# Radio Buttons

The `<input type="radio">` defines a radio button.

Radio buttons let a user select ONE of a limited number of choices.

## Example

A form with radio buttons:

`<p>Choose your favorite Web language:</p>`

```
<form>
  <input type="radio" id="html" name="fav_language" value="HTML">
  <label for="html">HTML</label><br>
  <input type="radio" id="css" name="fav_language" value="CSS">
  <label for="css">CSS</label><br>
  <input type="radio" id="javascript" name="fav_language" value="JavaScript">
  <label for="javascript">JavaScript</label>
</form>
```

`<!DOCTYPE html>`

`<html>`

`<body>`

`<h2>Radio Buttons</h2>`

`<p>Choose your favorite Web language:</p>`

```
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" ><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" ><br>
```

```
<input type="radio" id="html" name="fav_language" value="HTML">
<label for="html">HTML</label><br>
```

```

<input type="radio" id="css" name="fav_language" value="CSS">
<label for="css">CSS</label><br>
<input type="radio" id="javascript" name="fav_language" value="JavaScript">
<label for="javascript">JavaScript</label><br/>
<input type="submit" value="Submit">
</form>

</body>
</html>

```

## Checkboxes

The `<input type="checkbox">` defines a **checkbox**.

Checkboxes let a user select ZERO or MORE options of a limited number of choices.

### Example

A form with checkboxes:

```

<form>
  <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">
  <label for="vehicle1"> I have a bike</label><br>
  <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">
  <label for="vehicle2"> I have a car</label><br>
  <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">
  <label for="vehicle3"> I have a boat</label>
</form>

```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Checkboxes</h2>
```

```
<p>The <strong>input type="checkbox"</strong> defines a checkbox:</p>
```

```
<form >

  <input type="checkbox" id="vehicle1" name="vehicle1" value="Bike">
  <label for="vehicle1"> I have a bike</label><br>
  <input type="checkbox" id="vehicle2" name="vehicle2" value="Car">
  <label for="vehicle2"> I have a car</label><br>
  <input type="checkbox" id="vehicle3" name="vehicle3" value="Boat">
  <label for="vehicle3"> I have a boat</label><br><br>
  <input type="submit" value="Submit">

</form>

</body>
</html>
```

## HTML <input type="color">

The **HTML <input type="color">** is used to *define a color picker*. the value should be a seven character hexadecimal notation. It has a Default value which is **#000000**(black).

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

<h1>Show a Color Picker</h1>

<form action="">
  <label for="favcolor">Select your favorite color:</label>
  <input type="color" id="favcolor" name="favcolor" value="#009900">

</form>

</body>
</html>
```



# HTML Form Attributes

## The Action Attribute

The **action** attribute defines the action to be performed when the form is submitted.

Usually, the form data is sent to a file on the server when the user clicks on the submit button.

In the example below, the form data is sent to a file called "action\_page.php". This file contains a server-side script that handles the form data:

### Example

On submit, send form data to "action\_page.php":

```
<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe"><br><br>
  <input type="submit" value="Submit">
</form>
```

**ip:** If the **action** attribute is omitted, the action is set to the current page.

## The Target Attribute

The **target** attribute specifies where to display the response that is received after submitting the form.

The **target** attribute can have one of the following values:

Value	Description
<code>_blank</code>	The response is displayed in a new window or tab
<code>_self</code>	The response is displayed in the current window

The default value is `_self` which means that the response will open in the current window.

## Example

Here, the submitted result will open in a new browser tab:

```
<form action="/action_page.php" target="_blank">
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>The form target attribute</h2>
```

```
<p>When submitting this form, the result will be opened in a new  
browser tab:</p>
```

```
<form action="/action_page.php" target="_blank">
```

```
  <label for="fname">First name:</label><br>
```

```
  <input type="text" id="fname" name="fname" value="John"><br>
```

```
  <label for="lname">Last name:</label><br>
```

```
  <input type="text" id="lname" name="lname" value="Doe"><br><br>
```

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

```
</form>
```

```
</body>
```

```
</html>
```

## The Method Attribute

The `method` attribute specifies the HTTP method to be used when submitting the form data.

The form-data can be sent as URL variables (with `method="get"`) or as HTTP post transaction (with `method="post"`).

The default HTTP method when submitting form data is GET.

### Example

This example uses the GET method when submitting the form data:

```
<form action="/action_page.php" method="get">
```

### Example

This example uses the POST method when submitting the form data:

```
<form action="/action_page.php" method="post">
```

## HTML Form Elements

### The <select> Element

The `<select>` element defines a drop-down list:

### Example

```
<label for="cars">Choose a car:</label>
<select id="cars" name="cars">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="fiat">Fiat</option>
  <option value="audi">Audi</option>
</select>
```

```
<!DOCTYPE html>

<html>

<body>


<h2>The select Element</h2>


<p>The select element defines a drop-down list:</p>


<form action="/action_page.php">
  <label for="cars">Choose a car:</label>
  <select id="cars" name="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select>
  <input type="submit">
</form>


</body>
</html>
```

The `<option>` elements defines an option that can be selected.

By default, the first item in the drop-down list is selected.

To define a pre-selected option, add the `selected` attribute to the option:

## Example

```
<option value="fiat" selected>Fiat</option>
```

## Visible Values:

Use the `size` attribute to specify the number of visible values:

### Example

```
<label for="cars">Choose a car:</label>
<select id="cars" name="cars" size="3">
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="fiat">Fiat</option>
  <option value="audi">Audi</option>
</select>

<!DOCTYPE html>

<html>

<body>

<h2>Visible Option Values</h2>

<p>Use the size attribute to specify the number of visible values.</p>

<form action="/action_page.php">
  <label for="cars">Choose a car:</label>
  <select id="cars" name="cars" size="3">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select><br><br>
  <input type="submit">
</form>

</body>

</html>
```

## Allow Multiple Selections:

Use the `multiple` attribute to allow the user to select more than one value:

### Example

```
<label for="cars">Choose a car:</label>
<select id="cars" name="cars" size="4" multiple>
  <option value="volvo">Volvo</option>
  <option value="saab">Saab</option>
  <option value="fiat">Fiat</option>
  <option value="audi">Audi</option>
</select>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Allow Multiple Selections</h2>
```

```
<p>Use the multiple attribute to allow the user to select more than one
value.</p>
```

```
<form action="/action_page.php">
```

```
  <label for="cars">Choose a car:</label>
```

```
  <select id="cars" name="cars" size="4" multiple>
```

```
    <option value="volvo">Volvo</option>
```

```
    <option value="saab">Saab</option>
```

```
    <option value="fiat">Fiat</option>
```

```
    <option value="audi">Audi</option>
```

```
  </select><br><br>
```

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

```
</form>
```

```
<p>Hold down the Ctrl (windows) / Command (Mac) button to select multiple options.</p>
```

```
</body>
```

```
</html>
```

## The <textarea> Element

The <textarea> element defines a multi-line input field (a text area):

### Example

```
<textarea name="message" rows="10" cols="30">
```

The cat was playing in the garden.

```
</textarea>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Textarea</h2>
```

```
<p>The textarea element defines a multi-line input field.</p>
```

```
<form action="/action_page.php">
```

```
  <textarea name="message" rows="10" cols="30">The cat was playing in the garden.</textarea>
```

```
  <br><br>
```

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

```
</form>
```

```
</body>
```

```
</html>
```

The **rows** attribute specifies the visible number of lines in a text area.

The **cols** attribute specifies the visible width of a text area.

You can also define the size of the text area by using CSS:

## Example

```
<textarea name="message" style="width:200px; height:600px;">
The cat was playing in the garden.
</textarea>
```

# The <button> Element

The **<button>** element defines a clickable button:

## Example

```
<button type="button" onclick="alert('Hello World!')">Click
Me!</button>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>The button Element</h2>
```

```
<button type="button" onclick="alert('Hello World!')">Click
Me!</button>
```

```
</body>
```

```
</html>
```



# The <fieldset> and <legend> Elements

The <fieldset> element is used to group related data in a form.

The <legend> element defines a caption for the <fieldset> element.

## Example

```
<form action="/action_page.php">
  <fieldset>
    <legend>Personalia:</legend>
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname" value="John"><br>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname" value="Doe"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
</form>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Grouping Form Data with Fieldset</h2>
```

<p>The fieldset element is used to group related data in a form, and the legend element defines a caption for the fieldset element.</p>

```
<form action="/action_page.php">
  <fieldset>
    <legend>Personalia:</legend>
    <label for="fname">First name:</label><br>
    <input type="text" id="fname" name="fname" value="John"><br>
    <label for="lname">Last name:</label><br>
    <input type="text" id="lname" name="lname" value="Doe"><br><br>
    <input type="submit" value="Submit">
  </fieldset>
```

```
</form>
```

```
</body>
```

```
</html>
```

# Input Type Password

`<input type="password">` defines a **password field**:

## Example

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

The characters in a password field are masked (shown as asterisks or circles).

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Password field</h2>
```

<p>The <strong>input type="password"</strong> defines a password field:</p>

```
<form action="/action_page.php">
  <label for="username">Username:</label><br>
  <input type="text" id="username" name="username"><br>
  <label for="pwd">Password:</label><br>
  <input type="password" id="pwd" name="pwd"><br><br>
```

```
<input type="submit" value="Submit">
</form>
```

<p>The characters in a password field are masked (shown as asterisks or circles).</p>

```
</body>
```

```
</html>
```

## Input Type Reset

<input type="reset"> defines a **reset button** that will reset all form values to their default values:

### Example

```
<form action="/action_page.php">
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe"><br><br>
  <input type="submit" value="Submit">
  <input type="reset">
</form>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h2>Reset Button</h2>
```

<p>The <strong>input type="reset"</strong> defines a reset button that resets all form values to their default values:</p>

```
<form action="/action_page.php">
```

```
<label for="fname">First name:</label><br>
<input type="text" id="fname" name="fname" value="John"><br>
<label for="lname">Last name:</label><br>
<input type="text" id="lname" name="lname" value="Doe"><br><br>
<input type="submit" value="Submit">
<input type="reset">
</form>

<p>If you change the input values and then click the "Reset" button,
the form-data will be reset to the default values.</p>

</body>
</html>
```

## Input Type Date

The `<input type="date">` is used for input fields that should contain a date.

Depending on browser support, a date picker can show up in the input field.

### Example

```
<form>
  <label for="birthday">Birthday:</label>
  <input type="date" id="birthday" name="birthday">
</form>

<!DOCTYPE html>

<html>

<body>

<h2>Date Field</h2>
```

<p>The `<strong>input type="date"</strong>` is used for input fields that should contain a date.</p>

```
<form action="/action_page.php">
  <label for="birthday">Birthday:</label>
  <input type="date" id="birthday" name="birthday">
  <input type="submit" value="Submit">
</form>
```

```
<p><strong>Note:</strong> type="date" is not supported in Internet
Explorer 11 or prior Safari 14.1.</p>
```

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

## Input Type Email

The `<input type="email">` is used for input fields that should contain an e-mail address.

Depending on browser support, the e-mail address can be automatically validated when submitted.

Some smartphones recognize the email type, and add ".com" to the keyboard to match email input.

### Example

```
<form>
  <label for="email">Enter your email:</label>
  <input type="email" id="email" name="email">
</form>
```

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

```
<h2>Email Field</h2>
```

<p>The <strong>input type="email"</strong> is used for input fields that should contain an e-mail address:</p>

```
<form action="/action_page.php">

  <label for="email1">Enter your email:</label>

  <input type="email" id="email1" name="email">

  <input type="submit">

</form>
```

```
</body>
```

```
</html>
```

## Input Type File

The <input type="file"> defines a file-select field and a "Browse" button for file uploads.

### Example

```
<form>
  <label for="myfile">Select a file:</label>
  <input type="file" id="myfile" name="myfile">
</form>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1>File upload</h1>
```

```
<p>Show a file-select field which allows a file to be chosen for  
upload:</p>
```

```
<form action="/action_page.php">
```

```
  <label for="myfile">Select a file:</label>
```

```
  <input type="file" id="myfile" name="myfile"><br><br>
```

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

```
</form>
```

```
</body>
```

```
</html>
```

## The readonly Attribute

The input `readonly` attribute specifies that an input field is read-only.

A read-only input field cannot be modified (however, a user can tab to it, highlight it, and copy the text from it).

The value of a read-only input field will be sent when submitting the form!

### Example

A read-only input field:

```
<form>  
  <label for="fname">First name:</label><br>  
  <input type="text" id="fname" name="fname" value="John" readonly><br>  
  <label for="lname">Last name:</label><br>  
  <input type="text" id="lname" name="lname" value="Doe">  
</form>
```

# The disabled Attribute

The input `disabled` attribute specifies that an input field should be disabled.

A disabled input field is unusable and un-clickable.

The value of a disabled input field will not be sent when submitting the form!

## Example

A disabled input field:

```
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" value="John" disabled><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname" value="Doe">
</form>
```

# The size Attribute

The input `size` attribute specifies the visible width, in characters, of an input field.

The default value for `size` is 30.

## Example

Set a width for an input field:

```
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" size="50"><br>
  <label for="pin">PIN:</label><br>
  <input type="text" id="pin" name="pin" size="4">
</form>
```



# The maxlength Attribute

The input `maxlength` attribute specifies the maximum number of characters allowed in an input field.

**Note:** When a `maxlength` is set, the input field will not accept more than the specified number of characters. However, this attribute does not provide any feedback. So, if you want to alert the user, you must write JavaScript code.

## Example

Set a maximum length for an input field:

```
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname" size="50"><br>
  <label for="pin">PIN:</label><br>
  <input type="text" id="pin" name="pin" maxlength="4" size="4">
</form>
```

# HTML

## <input> required Attribute

The required attribute is a boolean attribute.

When present, it specifies that an input field must be filled out before submitting the form

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1>The input required attribute</h1>
```

```
<form action="/action_page.php">
```

```
<label for="username">Username:</label>
<input type="text" id="username" name="username" required>
<input type="submit">
</form>

</body>
</html>
```

# HTML

## <input> placeholder Attribute

The `placeholder` attribute specifies a short hint that describes the expected value of an input field (e.g. a sample value or a short description of the expected format).

The short hint is displayed in the input field before the user enters a value.

By default label is inline element , inline element does not have height and width to apply, to apply width to the label we have to make it inline-block

### CSS Layout - **display: inline-block**

**“display: inline-block” Property:** This property is used to display an element as an inline-level block container. The element itself is formatted as an inline element, but it can apply height and width values. It is placed as an inline element (on the same line as adjacent content). It looks like an inline element but it behaves as a block element and doesn't force to line break. It allows having a block-level appearance while still being laid out inline with other elements.

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width,
initial-scale=1.0">
    <title>Form </title>
    <style>
        .labelClass{
            display:inline-block;
            width:150px;
            font-weight: bold;

        }
        .row{
            margin:4px;

        }
        h2{
            text-align: center;
        }
        #form{
            width:440px;
            height:330px;
            margin-top:10px;
            padding:4px;
            border-radius:10px;
            background-color: black;
            color:white;
        }
        #submit{
            width:7cm;
            border-radius:10px;
            margin-top:7px;
            color:white;
            padding:4px;
            font-weight: bold;
            background-color: dodgerblue;
        }
        #innerblock{
            text-align: center;
        }
    </style>

```

```

</style>
</head>
<body>
    <div id="form">
        <h2>Registration Form</h2>
        <form>
            <div class="row">
                <label class="labelClass"
for="name">Name</label>
                <input type="text" id="name" name="name"/>
            </div>
            <div class="row">
                <label class="labelClass"
for="fname">Father's Name</label>
                <input type="text" id="fname"
name="fname"/>
            </div>
            <div class="row">
                <label class="labelClass"
for="mname">Mother's Name</label>
                <input type="text" id="mname"
name="mname"/>
            </div>
            <div class="row">
                <label class="labelClass"
for="email">Email</label>
                <input type="email" id="email"
name="email"/>
            </div>
            <div class="row">
                <label class="labelClass">Gender</label>
                <input type="radio" id="Female"
name="gender" value="female">
                <label for="Female">Female</label>
                <input type="radio" id="Male"
name="gender" value="male">
                <label for="Male">Male</label>
                <input type="radio" id="Other"
name="gender" value="other">
                <label for="Other">Other</label>
            </div>
            <div class="row">

```

```

        <label class="labelClass" for="dob">Date
of Birth</label>
        <input type="date" id="dob" name="dob"/>
    </div>
    <div class="row">
        <label class="labelClass" for="bg">Blood
Group</label>
        <select id="bg" name="bg">
            <option value="0"
selected>Select</option>
            <option value="A+">A+</option>
            <option value="B+">B+</option>
            <option value="0+">0+</option>
        </select>

    </div>
    <div class="row">
        <label class="labelClass">Course</label>
        <input type="checkbox" id="Java"
name="Java" value="Java">
        <label for="Java">Java</label>
        <input type="checkbox" id="C" name="C" value="C">
        <label for="C">C</label>
        <input type="checkbox" id="Python" name="Python"
value="Python">
        <label for="Python">Python</label>
        <input type="checkbox" id="Php" name="Php" value="Php">
        <label for="Php">Php</label>
    </div>
    <div class="row">
        <label class="labelClass"
for="photo">Photo</label>
        <input type="file" id="photo"
name="photo"/>
    </div>
    <div id="innerblock">
        <input type="submit" name="submit"
id="submit" />
    </div>
</form>
</div>
</body>

```

</html>

## **HTML Semantic Elements**

Semantic HTML, also known as semantic markup, refers to the use of HTML tags that convey the meaning—or semantics—of the content contained within them.

By adding semantic HTML tags to your pages, you provide additional information that helps define the roles and relative importance of the different parts of your page.

(As opposed to non-semantic HTML, which uses tags that don't directly convey meaning.)

### **What is HTML Semantics?**

The core characteristic of a semantic element is that it clearly communicates its meaning to both the developer and the browser. These elements clearly define its content.

### **Why Do You Need to Use Semantic Tags in HTML?**

There are several advantages of using semantic tags in HTML:

- The semantic HTML tags help the search engines and other user devices to determine the importance and context of web pages.
- The pages made with semantic elements are much easier to read.
- It has greater accessibility. It offers a better user experience.

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 website structure.

## **HTML Semantic Tag Structure**

The following HTML tags can be used to break your page into identified parts:

- `<header>`: It defines a header for a web page.
- `<nav>`: It defines a container for navigation links.
- `<section>`: This defines a section in a web page.
- `<article>`: This element contains the main part, containing information about the web page.
- `<aside>`: The `<aside>` content is often placed as a sidebar in a document.
- `<footer>`: It defines a footer for a document or a section.

**We can use div in case of semantic tag but semantic tag are more useful to seo**

## HTML `<header>` Tag

The `<header>` element represents a container for introductory content or a set of navigational links.

A `<header>` element typically contains:

- one or more heading elements (`<h1>` - `<h6>`)
- logo or icon

## HTML `<nav>` Tag

The `<nav>` tag defines a set of navigation links.

Notice that NOT all links of a document should be inside a `<nav>` element. The `<nav>` element is intended only for major blocks of navigation links.

## HTML `<section>` Tag

**Section tag defines the section of documents such as chapters, headers, footers or any other sections. The section tag divides the content into section and subsections. The section tag is used when requirements of two headers or footers or any other section of documents needed. Section tag**

grouped the generic block of related contents. The main advantage of the section tag is, it is a semantic element, which describes its meaning to both browser and developer.

## HTML <article> Tag

The <article> tag is independent of the other content of the page (even though it can be related).

In other words, The article element represents a component of a page that consists of self-contained composition in a document, page, or site

## HTML <details> Tag

The <details> tag is used for the content/information which is initially hidden but could be displayed if the user wishes to see it. The content of the details tag is visible when open the set attributes. The summary tag is used with the **details** tag for specifying visible heading.

## HTML 5 <summary> Tag

The <summary> tag in HTML is used to define a summary for the <details> element.

- The <summary> element is used along with the <details> element and provides a summary visible to the user.
- When the summary is clicked by the user, the content placed inside the <details> element becomes visible which was previously hidden.
- The <summary> tag was added in HTML 5.
- The <summary> tag requires both starting and ending tag.

**Note:** The <summary> element should be the first child element of the <details> element.

### Syntax:

<summary> Content </summary>



## HTML5 figure Tag

**HTML <figure> tag** is used to mark up a photo in the document on a web page.

As we know image tag is already available in HTML to display the pictures on web pages. But HTML 5 <figure> tag is used to handle the group of diagrams, photos, code listing etc. with some embedded content. You can also add a caption for the photo with the help of <figcaption> tag.

HTML figure is a new tag introduced in HTML5.

### HTML figcaption tag

The <figcaption> element is used to provide a caption to an image.

It is an optional tag and can appear before or after the content within the <figure> tag.

Only one <figcaption> element can be nested within a <figure> tag although the <figure> element itself may contain multiple other elements like <img> or <code>.

The <figcaption> element is used with <figure> element and it can be placed as the first or last child of the <figure> element.

## HTML Footer Tag

**HTML <footer> tag** is used to define a footer for a document or a section. It is generally used in the last of the section (bottom of the page).

The footer tag is included in HTML5.

HTML <footer> tag contains information about its containing elements for like social media links etc

## HTML Aside Tag

The HTML `<aside>` tag is used to represent a portion of a document that is indirectly related to the main content. It is most commonly used as a sidebar in the document.

## HTML Main Tag

**HTML `<main>` tag** is used to represent the main content of the `<body>` tag.

The `<main>` tag is written within `<body>` tag. It is used to accurately describe the primary content of a page.

The content of the main tag is directly related to the central topic of the document.

The HTML `<main>` tag is a semantic tag used to define the main content of the document. The main content contains the primary information or functionality that the page is meant to provide to the user.

### **Points to remember:**

Author should not include more than one `<main>` tag within a document.

The `<main>` element should not be used as a child of an `<article>`, `<aside>`, `<header>`, `<footer>`, or `<nav>` element.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Html Semantic Tag</title>
  <style>

    header{
      text-align:center;
    }
    section{
      margin-top:50px;
    }
    img{
      width:200px;
    }
    details{
      display:block;
    }
    figure{
      text-align:center;
    }
    footer{
      text-align:center;
    }
    aside {

      border:1px solid black;
      padding:5px;
      float:right;
    }
  </style>
</head>
<body>
  <header>
    <h1>Courier Website</h1>
    <p>Best Courier service in the city</p>
    <nav>
```

[Home](#) |  
[Log In](#) |  
[Contact Us](#) |  
[About Us](#)

</nav>

</header>

<main>

<p>Introduction to HTML</p>

</main>

<section>

<article>

<br/>

<a href="#">Save Animals</a><br/>

<p>Koalas will be extinct in the Australian state of New South Wales (NSW) by 2050 unless there is urgent action, an inquiry has found.

The once-thriving marsupial has been ravaged by habitat loss, disease and climatic events in recent years.

About 5,000 koalas are thought to have died in devastating recent bushfires, the report to state parliament said.

It urged lawmakers to ensure that remaining populations did not perish in rapidly diminishing habitats.</p>

</article>

</section>

<section>

<details>

<summary>

Cities in which our service is active

</summary>

<ul>

<li>New Delhi</li>

<li>Mumbai</li>

<li>Lucknow</li>

```
        <li>Bhopal</li>
    </ul>
</details>
</section>
<section>
    <figure>
        
        <figcaption>Short Story: A Backup</figcaption>
    </figure>
</section>
<section>
    <footer>
        <p>Instagram | Facebook | Whatsapp No</p>
        <!-- To write copyright symbol -->
        <p>&copy; 2023</p>
    </footer>
</section>
<section>
    <aside>
        <ul>

            <li> <a href="#">Home</a> </li>
            <li> <a href="#">Log In</a></li>
            <li> <a href="#">Contact Us</a></li>
            <li><a href="#">About Us</a> </li>

        </ul>
    </aside>
</section>
</body>
</html>
```

## HTML <meta> tag

HTML <meta> tag is used to represent the metadata about the HTML document. It specifies page description, keywords, copyright, language, author of the documents, etc.

The metadata does not display on the webpage, but it is used by search engines, browsers and other web services which scan the site or webpage to know about the webpage.

With the help of meta tag, you can experiment and preview that how your webpage will render on the browser.

The <meta> tag is placed within the <head> tag, and it can be used more than one times in a document.

**Following are some specific syntaxes of meta tag which shows the different uses of meta Tag.**

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

It defines the character encoding. The value of charset is "utf-8" which means it will support to display any language.

The HTML5 specification encourages web developers to use the UTF-8 character set!

### What Is UTF-8?

UTF-8 stands for “Unicode Transformation Format - 8 bits.”

UTF-8 is a variable-length character encoding standard used for electronic communication. Defined by the Unicode Standard, the name is derived from Unicode (or Universal Coded Character Set) Transformation Format – 8-bit.

2. `<meta name="keywords" content="HTML, CSS, JavaScript, Bootstrap">`

It specifies the list of keyword which is used by search engines.

3. `<meta name="description" content="Computer Centre">`

It defines the website description which is useful to provide relevant search performed by search engines.

## What is The Viewport?

The viewport is the user's visible area of a web page.

The viewport varies with the device, and will be smaller on a mobile phone than on a computer screen.

Before tablets and mobile phones, web pages were designed only for computer screens, and it was common for web pages to have a static design and a fixed size.

Then, when we started surfing the internet using tablets and mobile phones, fixed size web pages were too large to fit the viewport. To fix this, browsers on those devices scaled down the entire web page to fit the screen.

This was not perfect!! But a quick fix.

## Setting The Viewport

HTML5 introduced a method to let web designers take control over the viewport, through the `<meta>` tag.

You should include the following `<meta>` viewport element in all your web pages:

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

This gives the browser instructions on how to control the page's dimensions and scaling.

The `width=device-width` part sets 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` part sets the initial zoom level when the page is first loaded by the browser.



## <iframe>: The Inline Frame element

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.

### Iframe Syntax

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

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

### Set Width and Height of iframe

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

For example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <h1>iframe example</h1>
  <h2>Comparing two news channel</h2>
  <iframe src="https://zeenews.india.com/hindi" width="500"
height="500"></iframe>
  <iframe src="https://ndtv.in/topic/-
%E0%A4%AA%E0%A4%B0%E0%A4%BF%E0%A4%B5%E0%A4%B0%E0%A5%8D%E0%A4%A4%E0%A4%A8"
width="500" height="500"></iframe>

</body>
</html>
```

```

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <h1>iframe example</h1>
  <h2>K D Singh Babu Stadium</h2>
  <iframe
src="https://www.google.com/maps/embed?pb=!1m18!1m12!1m3!1d3559.505888510166!2d80.93511557432294!3d26.855663862541988!2m3!1f0!2f0!3f0!3m2!1i1024!2i768!4f13.1!3m3!1m2!1s0x399bfda16d3bbd4b%3A0x84ee3e6b3a4c42ed!2sKD%20Singh%20Babu%20Stadium!5e0!3m2!1sen!2sin!4v1695637099052!5m2!1sen!2sin" width="600" height="450" style="border:0;" allowfullscreen="" loading="lazy" referrerpolicy="no-referrer-when-downgrade"></iframe>
</body>
</html>

```

## How to Embed PDF file using HTML ?

Sometimes, we may want to insert a PDF file into an HTML document or code, to make the content more interactive. Because the formats are so different, which is not easy to accomplish the task.

To embed the PDF HTML code into a webpage to make it interactive. The following embedding methods can be used to add PDF to an HTML file:

- Using Object Tag
- Using an iframe
- Using embed tag

**Method 1: Using Object Tag:** HTML's object tag is the first way to embed PDF files. In the below example, the pdf file will be displayed on a web page, which is an object.

```

<!DOCTYPE html>
<html>
<head>
    <title>PDF in HTML</title>
</head>
<body>
    <center>

        <h3>Embedding the PDF file Using Object Tag</h3>

        <object data=
"gitnotes.pdf" type="application/pdf"
            width="800"
            height="500">
        </object>
    </center>
</body>
</html>

```

## Embed PDF in HTML Using the <embed> Tag

To embed external resources in a webpage, HTML provides an embed tag. Using the embed tag, we can include external resources such as PDFs, media players, and webpages. The tag has the src attribute, which allows us to specify the path to the file that will be embedded. The type attribute allows us to specify the type of the embedded file. The type attribute for PDF should be application/pdf. The embed tag is self-contained.

```

<!DOCTYPE html>
<html>
<head>
    <title>PDF in HTML</title>
</head>
<body>
    <center>
        <h1 style="color: green">Embed Tag</h1>
        <h3>Embedding the PDF file Using Object Tag</h3>
        <embed src="gitnotes.pdf" type="application/pdf" height="500"
width="800" >
    </center>
</body>
</html>

```

## Embed Youtube Video

**Go to youtube video click on share button and then click embed button and then copy the code**

**For example**

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Embed Youtube Video</title>
</head>
<body>
  <h2>Embed Youtube Video</h2>
  <iframe width="560" height="315"
src="https://www.youtube.com/embed/JqywUG3zujk?si=DRaqdTAmMRI0yPB
p" title="YouTube video player" style="border:2px solid blue"
allow="accelerometer; autoplay; clipboard-write; encrypted-media; gyroscope;
picture-in-picture; web-share" allowfullscreen ></iframe>
</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:

We can use an entity in the HTML document by name or by a numerical character reference. Each entity starts with symbol ampersand (&) and ends with a semicolon (;).

### Syntax:

1. &entity\_name;
2. OR
3. &#entity\_number;

For example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>
  <!-- To print <div></div> -->
  &lt;div&gt; &nbsp; &lt;&#47;div&gt;
</body>
</html>
```

## Some Useful 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;
/	slash		&#47;

**Note:** Entity names are case sensitive.

### What is &nbsp; In HTML?

In HTML, &nbsp; is a special entity that is used for representing a non-breaking space. It is a type of space that prevents the browser from automatically wrapping text to the next line if it exceeds the available width.

[illegible]

# HTML Symbols

Symbols that are not present on your keyboard can also be added by using entities.

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
Σ	&#8721;	&sum;	N-ARY SUMMATION

## What are Emojis?

Emojis look like images, or icons, but they are not.

They are letters (characters) from the UTF-8 (Unicode) character set.

UTF-8 covers almost all of the characters and symbols in the world.

## The HTML charset Attribute

To display an HTML page correctly, a web browser must know the character set used in the page.

This is specified in the `<meta>` tag:

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

If not specified, UTF-8 is the default character set in HTML.





Emoji	Value
⚠️	&#128507;
🙏	&#128508;
🙌	&#128509;
🇯🇵	&#128510;
👉	&#128511;
😊	&#128512;
😄	&#128513;
😂	&#128514;
😁	&#128515;
😊	&#128516;



&#128517;

Go to site <https://imagemap.org/>

# HTML Image Maps

With HTML image maps, you can create clickable areas on an image.

## Image Maps

The HTML `<map>` tag defines an image map. An image map is an image with clickable areas. The areas are defined with one or more `<area>` tags.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>

  </style>
</head>
<body>
  
  <map name="image_map">
    <area alt="google" target="_blank" title="google"
href="https://www.google.com/" coords="10,15,92,119" shape="rect">
    <area alt="yahoo" target="_blank" title="yahoo"
href="https://www.yahoo.com/" coords="97,57,113,84" shape="rect">
```

```
<area alt="jagran" target="_blank" title="jagran"
href="https://www.jagran.com/" coords="114,100,13" shape="circle">
</map>
</body>
</html>
```

## HTML Datalist Tag

The HTML <datalist> tag is used to provide an auto complete feature on form element. It provides a list of predefined options to the users to select data.

The datalist tag is introduced in HTML5.

The <datalist> tag should be used with an <input> element that contains a "list" attribute. The value of "list" attribute is linked with the datalist id.

For example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Document</title>
</head>
<body>

  <form>
    <label>
      Enter your favorite cricket player: Press any character<br />
      <input type="text" id="favCktPlayer" list="CktPlayers">
      <datalist id="CktPlayers">
        <option value="Sachin Tendulkar">
        <option value="Brian Lara">
        <option value="Jacques Kallis">
        <option value="Ricky Ponting">
        <option value="Rahul Dravid">
        <option value="Shane Warne">
        <option value="Rohit Sharma">
```

```
<option value="Donald Bradman">
<option value="Saurav Ganguly ">
<option value="AB diVilliers">
<option value="Mahendra Singh Dhoni">
<option value="Adam Gilchrist">
</datalist>
</label>
</form>

</body>
</html>
```

## Canonical Tag

Canonical Tag is also known as canonical URL or URL canonicalization. It is incorporated in the HTML code of a webpage to show the original source of content. So, it is an HTML element that is used to prevent duplicate content issues. It specifies the canonical URL "the preferred version" of a web page and tells search engines that the other similar URLs are not different or duplicates; they are one and the same. Thus, it prevents the issues that arise when the same content appears on multiple URLs

### How does Canonical Tag look like/ Parts of a canonical tag:

Canonical Tags have simple and consistent syntax and are placed within the <head> section of a web page. It appears as rel="canonical".

**For example:** <link rel="canonical" href="car.html"/>

The meaning of each part of the code:

1. link rel= "canonical": The link contained in this tag is the original (canonical) version of this page.
2. href="car.html": You can visit this URL to find the canonical version.

## Why canonical tag is important in terms of SEO:

Search engines like Google don't like duplicate content. It creates confusion for them, here's how:

1. Google does not know which version of a page to index out of multiple pages.
2. Which page to rank for queries.
3. Whether consolidate link equity on one page or split it among multiple versions.
4. Google may waste time crawling different URL with similar content instead of discovering other new pages on your website.

So, you must use a canonical URL if you have two or more pages of similar content on your website or if the content of your site is also used on another site. In this way, you can point Google to the original content and make sure the main page gets all of the credit and SEO benefits.

## What Is a Meta Robots Tag?

A meta robots tag is HTML code that tells search engine robots how to crawl, index, and display a page's content.

It goes in the <head> section of the page and can look like this:

```
<meta name="robots" content="noindex">
```

The meta robot tag in the example above tells all robots not to index the page.

## What Are Robots Meta Tags Used For?

Robots meta tags help control how Google crawls and indexes a page's content. Including whether to:

- Include a page in search results
- Follow the links on a page

Google supports the following “content” values:

### *Noindex*

The meta robots “noindex” value tells crawlers not to include the page in the index or display in the SERPs.

### *Nofollow*

Tells crawlers not to crawl the links on the page.

```
<meta name="robots" content="noindex,nofollow">
```

```
<meta name="robots" content="index,allow">
```

## HTML Favicon

A favicon is a small image displayed next to the page title in the browser tab.

To add a favicon to your website, either save your favicon image to the root directory of your webserver, or create a folder in the root directory called images, and save your favicon image in this folder. A common name for a favicon image is "favicon.ico".

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Demo Page</title>
  <link rel="icon" href="image/favicon.png" type="image/png" />
</head>
<body>

</body>
</html>
```

For all device like laptop,tab,mobile etc we used favicon generator

<https://realfavicongenerator.net/>

copy the code and paste in the head tag of the web page

and download the favicon package and paste it in image folder and extract the file and rename the folder to favicon and add favicon name to each href

so the code look like this:

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Demo Page</title>
  <link rel="apple-touch-icon" sizes="180x180" href="favicon/apple-touch-
icon.png">
  <link rel="icon" type="image/png" sizes="32x32" href="favicon/favicon-
32x32.png">
  <link rel="icon" type="image/png" sizes="16x16" href="favicon/favicon-
16x16.png">
  <link rel="manifest" href="favicon/site.webmanifest">
  <link rel="mask-icon" href="favicon/safari-pinned-tab.svg" color="#5bbad5">
  <meta name="msapplication-TileColor" content="#da532c">
  <meta name="theme-color" content="#ffffff">
</head>
<body>

</body>
</html>
```

## HTML <noscript> Tag

Use of the <noscript> tag:

```
<script>
document.write("Hello World!")
</script>
<noscript>Your browser does not support JavaScript!</noscript>
```

The <noscript> tag defines an alternate content to be displayed to users that have disabled scripts in their browser or have a browser that doesn't support script.



The `<noscript>` element can be used in both `<head>` and `<body>`. When used inside `<head>`, the `<noscript>` element could only contain `<link>`, `<style>`, and `<meta>` elements.

# HTML dir Attribute

The `dir` attribute specifies the text direction of the element's content.

The `dir` attribute can have the following values:

- `ltr` - means left-to-right text direction
- `rtl` - means right to left text direction
- 

```
<!DOCTYPE html>
<html dir="rtl" lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-
scale=1.0">
  <title>Demo Page</title>
  <link rel="apple-touch-icon" sizes="180x180"
href="favicon/apple-touch-icon.png">
  <link rel="icon" type="image/png" sizes="32x32"
href="favicon/favicon-32x32.png">
  <link rel="icon" type="image/png" sizes="16x16"
href="favicon/favicon-16x16.png">
  <link rel="manifest" href="favicon/site.webmanifest">
  <link rel="mask-icon" href="favicon/safari-pinned-tab.svg"
color="#5bbad5">
  <meta name="msapplication-TileColor" content="#da532c">
  <meta name="theme-color" content="#ffffff">
</head>
<body>
<p>This is the sample page</p>
</body>
</html>
```

# HTML lang Attribute

The lang attribute specifies the language of the element's content.

Common examples are "en" for English, "es" for Spanish, "fr" for French, and so on.

## W3C World Wide Web Consortium

It is the organisation that develop and maintained HTML and also provide tool to check whether html code is ok or not.

To check the site go to following url

[validator.w3.org](https://validator.w3.org)