HTML Introduction

HTML stands for HyperText Markup Language. HTML is the main markup language for describing the structure of Web pages.

Note: HyperText Markup Language (HTML), as described earlier is a markup language not a programming language.

Description

Hypertext is text displayed on a computer or other electronic device with references to other text that the user can immediately access, usually by a mouse click or key press.

Apart from text, hypertext may contain tables, images and other presentational elements. It is an easy-to-use and flexible format to share information over the Internet.

Markup Languages use sets of markup tags to characterize text elements within a document, which gives instructions to the Web browser on how the document should appear.

HTML Gives You Ability To:

- Publish documents online with text, headings, images, lists, tables, etc.
- Accessing online information or other web resources such as images, videos or other HTML document via hyperlinks.
- Create forms to collect user input like login information, feedback, comments or conducting transactions with remote servers, etc.
- Include videos, sound clips, flash movies, applications and other HTML document directly inside an HTML document.

HTML Tags and Elements

HTML is written in the form of HTML elements consisting of markup tags. These markup tags are the fundamental characteristic of HTML. Every markup tag is composed of a keyword, surrounded by angle brackets, like `<html>`, `<head>`, `<body>`, etc.

HTML tags normally come in pairs like `<html>` and `</html>`. The first tag in a pair is the opening tags the second tag is the closing tags. An opening tag and a closing tag are identical, except a slash (`/`) after the opening angle bracket of a closing tag, to tell the browser that the command has been completed. In between these tags you can add headings, paragraphs of text, tables, forms, images, videos etc. For example, a paragraph, which is represented by the `p` element, would be written as:

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

HTML Document Overview

All HTML document seem to be plain text files. They contain no images,sounds, videos etc. but just plain text. However, they may contain links to images, sounds and videos.

Note: The web browsers does not display the HTML tags, but uses the tags to interpret the content of the web pages.

HTML Get Started

An HTML file is simply a text file saved with an .html or .htm extension (i.e. as opposed to a .txt extension).

Getting Started

Here, you will learn how easy it is to create an HTML document (i.e. a web page). To begin coding HTML you need only two stuff: a simple-text editor and a web browser.

OK, let's get straight into it.

Creating Your First HTML Document

Let's walk through the following steps. At the end of the tutorial, you will have made an HTML file that displays "Hello world" message in your web browser.

Step 1: Creating the HTML file

Open up your computer's plain text editor and create a new file.

Step 2: Type some HTML code

Start with an empty window and type the following code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>A simple HTML document</title>
</head>
<body>
    <p>Hello World!<p>
</body>
</html>
```

Step 3: Saving the file

Now save the file on your desktop as "myfirstpage.html ".

Note: It is important that the extension `.html` is specified — some text editors, such as Notepad, will automatically save it as `.txt` otherwise.

To open the file in a browser. Navigate to your file then double click on it. It will open in your default Web browser. (If it does not, open your browser and drag the file to it.)

Explanation of code

You might think what that code was all about. OK, let's get straight into it.

- DOCTYPE declaration — A line defines the HTML5 document type.
- A declarative header section (enclosed by the HEAD element) — Provides information about the document, including its title, style information, and scripts.
- A document body (enclosed by the BODY element) — Contains the document's actual content that is rendered in the web browser and displayed to the user.

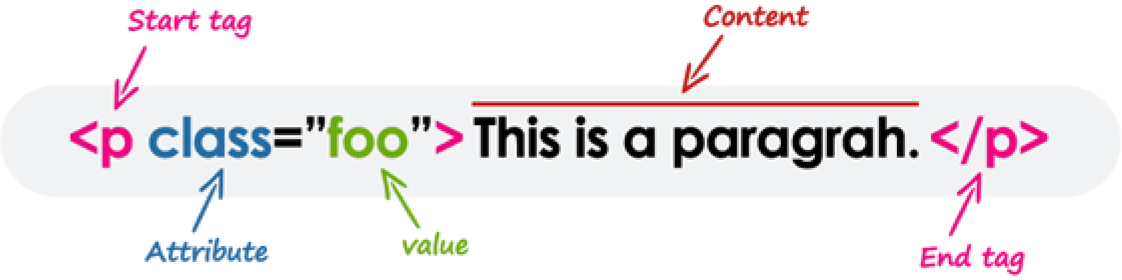
HTML Elements

An HTML element is an individual component of an HTML document.

HTML Element Syntax

HTML Elements represent semantics, or meaning. For example, The `title` element represents the title of the document. Most HTML elements are written with a *start tag* (or opening tag) and an *end tag* (or closing tag), with the content in between.

Elements can also contain attributes that define additional properties of an element. For example, a paragraph, which is represented by the `p` element, would be written as:



Note: Elements that don't require the end tag are know as *empty elements*, *self-closing elements* or *void elements*.

Empty HTML Elements

Empty elements (also called self-closing or void elements) are not container tags — that means, you can not write `<hr>`*some content*`</hr>` or `<br>`*some content*`</br>`.

A typical example of an empty element, is the `<br>` element, which represents a line break.

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

HTML Attributes

Attributes define additional properties of an element.

HTML Attributes

Certain parameters are frequently included within the opening tag to provide additional element properties (such as colorization, measurement, location, alignment, or other appearances) to the data between the markup tags. These parameters are called *Attributes*.

Note: Attributes are always specified in the start tag (or opening tag), and they can only contain the value of the attribute.

Attribute Values

Most attributes require a value. In HTML, the value can be left unquoted if it doesn't include spaces (`name=value`), or it can be quoted with single or double quotes (`name='value'` or `name="value"`). By the way, it is recommended to enclose Attribute values in quotes.

For example, the `abbr` element, which represents an abbreviation, expects a `title` attribute with its expansion. This would be written as:

```
<abbr title="Hyper Text Markup Language">HTML</abbr>
```

Attribute Example

In the example below href is attribute and the link provided is its value.

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

```
<!DOCTYPE html>
<head>
    <title>Example of HTML Attribute</title>
</head>
<body>
    <p><a href="https://www.microsoft.com" target="_blank">Visit Microsoft Website</a>.</p>
    <p><a href="https://www.google.com" target="_blank">Google</a> is the most popular search engine.</p>
</body>
</html>
```

Note: Attribute values are generally case-insensitive, except certain attribute values, like the id and class attributes. However, World Wide Web Consortium (W3C) recommends lowercase for attributes values in their specification.

HTML Headings

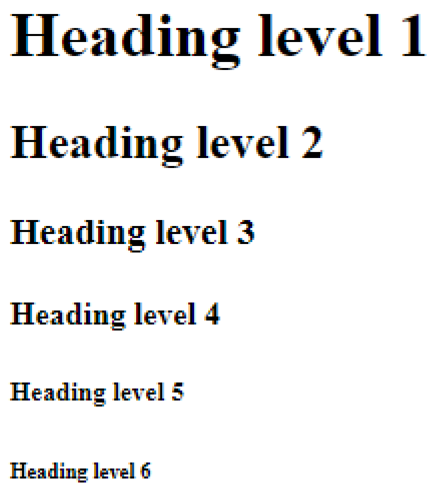
Headings help in defining the hierarchy and the structure of the web page.

HTML Headings

HTML uses six levels of heading tags `<h1>` to `<h6>`; the higher the heading level number, the greater it's importance — so `<h1>` defines the most important heading, whereas the `<h6>` defines the least important heading in the document.

```
<!DOCTYPE html>
<head>
    <title>Example of HTML headings tag</title>
</head>
<body>
    <h1>Heading level 1</h1>
    <h2>Heading level 2</h2>
    <h3>Heading level 3</h3>
    <h4>Heading level 4</h4>
    <h5>Heading level 5</h5>
    <h6>Heading level 6</h6>
</body>
</html>
```

Output :



Heading level 1

Heading level 2

Heading level 3

Heading level 4

Heading level 5

Heading level 6

Importance of Headings

- HTML headings provide valuable information by highlighting important topics and the structure of the document.
- Don't use headings to make text BIG or bold. Use it only for highlighting the heading of your document and to show the document structure.
- As search engines use headings to index the structure and content of your web pages so use it very wisely in your webpage.
- Use `<h1>` headings as main headings, followed by `<h2>` headings, then the less important `<h3>` headings, and so on.

Tip: A document generally should have exactly one `<h1>` element to mark the most important heading, followed by the `<h2>`, `<h3>` and so on.

HTML Paragraphs

Paragraph element used to publish text on the web pages.

HTML Paragraphs

Paragraphs are defined with the `<p>` tag. Paragraph tag is very basic and typically the first tag you will need to publish your text on the web pages.

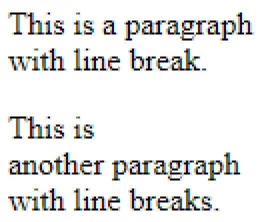
```
<!DOCTYPE html>
<head>
    <title>Example of HTML Paragraphs</title>
</head>
<body>
     <p>This is a paragraph.</p>
    <p>This is another paragraph.</p>
</body>
</html>
```

HTML Line Breaks

The `<br>` element is used to insert a line break without starting a new paragraph.

```
<!DOCTYPE html>
<head>
    <title>Example of Inserting Line Breaks in HTML</title>
</head>
<body>
     <p>This is a paragraph <br> with line break.</p>
    <p>This is <br>another paragraph <br> with line breaks.</p>
</body>
</html>
```

Output :



```
This is a paragraph
with line break.

This is
another paragraph
with line breaks.
```

Note: Don't use empty paragraph i.e. `<p></p>` to add extra space in your web pages. The browser may ignore the empty paragraph since it is logical tag.

HTML Comments

Comments are usually added with the purpose of making the source code easier to understand. It may help other developer (or you in the future when you edit the source code) to understand what you were trying to do with the HTML. Comments are significant to programmers but typically ignored by browsers.

An HTML comment begins with `<!--`, and ends with `-->`, See the example below:

```
<!DOCTYPE html>
<head>
    <title>Example of HTML comments</title>
</head>
<body>
    <!-- this is an HTML comment -->
    <!-- Comments are not displayed,
        by the browser -->
    <h1>This is a heading</h1>
    <p>This is a paragraph.</p>
</body>
</html>
```

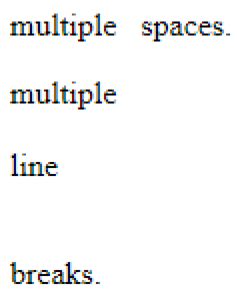
HTML Spaces

Normally the browser will display the multiple spaces created inside the HTML code by pressing the *space-bar key* on the keyboard as one space, while multiple line breaks created through pressing the enter key also displayed as single space. Insert `&nbsp;` for creating extra spaces, and `<br>` tag to create line breaks inside your HTML document.

Refer to the [HTML Character Entities]() for more.

```
<!DOCTYPE html>

<head>

    <title>Example of HTML Spaces</title>

</head>

<body>

     <p>multiple&nbsp;&nbsp;&nbsp;spaces.</p>

    <p>multiple<br><br>line<br><br><br>breaks.</p>

</body>

</html>
```

Output :



```
multiple   spaces.

multiple

line


breaks.
```

HTML Links

A link or hyperlink is a connection from one web resource to another.

HTML Links (Hyperlinks or Web links)

Links allow users to move seamlessly from one page to another, on any server anywhere in the world. A link has two ends called anchors and a direction. The link starts at the source anchor and points to the destination anchor, which may be any web resource (e.g. an image, an audio or video clip, an HTML document or an element within the document itself, etc.).

By default, links will appear as follows in most of the browsers:

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

HTML Link Syntax

Links are specified in HTML using the `<a>` tag.

A link or hyperlink could be a word, group of words, or image.

```
<a href="url">Text link</a>
```

The href attribute in source anchor specifies the address of the destination anchor.

```
<a href="https://www.microsoft.com/">Microsoft Website</a>
<a href="kites.jpg"><img src="kites-thumb.jpg" alt="kites"></a>
<a href="https://www.google.com/">Google Search</a>
```

The Target Attribute of Links

Target attribute tells the browser where to open linked document. There are four defined targets. Each target starts with an underscore(_): _blank, _parent, _self, _top.

```
<a href="https://www.microsoft.com/" target="_top">Microsoft
Website</a>
```

```
<a href="sky.jpg" target="_self"><img src="sky-thumb.jpg"
alt="Cloudy Sky"></a>

<a href="https://www.google.com/" target="_blank">Google</a>
```

Creating Bookmark Anchors

The id attribute is used to create a bookmark inside a web page in the following way.

```
<a href="#balloons"><img src="balloons-thumb.jpg"
alt="Balloons"></a>
```

```
<img src="balloons.jpg" alt="Hot Air Balloons" id="balloons">
```

Creating Download Links

Placing files available for download is done in exactly the same fashion as placing text links, just point the destination URL to the file you want to be available for download.

In this case it is a Zip file called test.zip.

```
<a href="downloads/test.zip">Download Zip file</a>
```

```
<a href="downloads/masters.pdf">Download PDF file</a>

<a href="downloads/sample.jpg">Download Image file</a>
```

Note: Clicking a link that points to a PDF file or Image file will not cause it to download to your hard drive directly. It will only open the file in your web browser. Further you can save it to your hard drive.

HTML Text Formatting

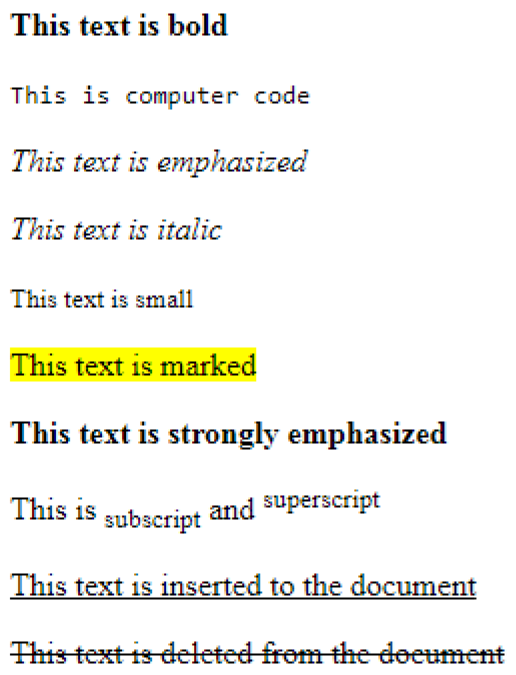
Using the text formatting tags you can make some text on your web pages to appear differently than normal text content.

HTML Formatting Tags

There are several tags exist in HTML to format text, like `<b>`, `<i>` etc. These formatting tags can make text bold, italic, sub/superscripted, and more.

```html
<p><b>This text is bold</b></p>
<p><code>This is computer code</code></p>
<p><em>This text is emphasized</em></p>
<p><i>This text is italic</i></p>
<p><small>This text is small</small></p>
<p><mark>This text is marked</mark></p>
<p><strong>This text is strongly emphasized</strong></p>
<p>This is <sub>subscript</sub> and <sup>superscript</sup></p>
<p><ins>This text is inserted to the document</ins></p>
<p><del>This text is deleted from the document</del></p>
```

Output :



This text is bold

`This is computer code`

This text is emphasized

This text is italic

This text is small

This text is marked

This text is strongly emphasized

This is $_{\text{subscript}}$ and $^{\text{superscript}}$

This text is inserted to the document

~~This text is deleted from the document~~

HTML Text Formatting Tags

Tag	Description
<b>	Defines bold text.
<del>	Defines deleted text.
<em>	Defines emphasized text.
<i>	Defines italic text.
<ins>	Defines inserted text.
<mark>	Defines marked/highlighted text.
<small>	Defines small text.
<strong>	Defines strong text.
<sub>	Defines subscripted text.
<sup>	Defines superscripted text.

HTML Styles

Style rules describe how documents are presented on the web browsers.

Styling HTML Elements

HTML is quite limited when it comes to the presentation of document. It was originally designed as a simple way of presenting information. CSS (Cascading Style Sheets) was introduced in December 1996 by the W3C, to provide a better way to style HTML elements.

With CSS, it becomes easy to specify the colors used for the text and the backgrounds, the size and the style for the fonts, the amount of space between elements, adding border and outlines to the elements and a host of other styling.

Adding Styles to HTML Elements

Style information can be either attached as a separate document or embedded in the HTML document. These are the three methods of implementing styling information to the HTML document (from highest to lowest priority).

- **Inline styles** — Using the `style` attribute in the HTML start tag.
- **Embedded style** — Using the `<style>` element in the head section of the document.
- **External style sheet** — Using the `<link>` element, pointing to an external CSS files.

Inline Styles

Inline styles are used to apply the unique style rules to an element by putting the CSS rules directly into the start tag. It can be attached to an element using the `style` attribute.

The style attribute includes a series of CSS property and value pairs. Each `property: value` pair is separated by a semicolon (`;`), just as you would write into an embedded or external style sheet. But it needs to be all in one line i.e. no line break after the semicolon.

The following example will show you how to change the color and font-size of the text:

```
<h1 style="color:red; font-size:30px;">This is a heading</h1>

<p style="color:green; font-size:18px;">This is a paragraph.</p>

<div style="color:green; font-size:18px;">This is some text.</div>
```

Using the inline styles are generally considered as a bad practice. As style rules are embedded directly inside the html tag, it causes the presentation to become mixed with the content of the document; which negates the purpose of using CSS.

Embedded Style Sheets

Embedded or internal style sheets only affect the document they are embedded in.

Embedded style sheets are defined in the <head> section of an HTML document using the <style>tag. You can define any number of <style> elements inside the <head> section of an HTML document. See the example below:

```
<head>
    <style type="text/css">
        body {background-color: YellowGreen;}
        p {color: Black;}
    </style>
</head>
```

External Style Sheets

An external style sheet is ideal when the style is applied to many pages.

An external style sheet holds all the style rules in a separate document that you can link from any HTML file on your site. External style sheets are the most flexible because with an external style sheet, you can change the look of an entire website by changing just one file.

You can attach external style sheets in two ways — *linking* and *importing*:

Linking External Style Sheets

An external style sheet can be linked to an HTML document using the <link> tag. The `<link>` tag goes inside the <head> section:

```
<head>
    <link rel="stylesheet" type="text/css" href="css/style.css">
</head>
```

HTML Images

Images improve the appearance and illustrate many concepts of the web pages.

Inserting Images in HTML Documents

The web is not just about text, its multi-media and HTML's multimedia features allow you to include images, audio clips, video clips, and other multimedia element in the web pages.

The `<img>` tag is used to insert images in HTML documents. It is an empty element and contains attributes only. The syntax of `<img>` tag can be given with:

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

The following example inserts three images on the web page:

```
<img src="kites.jpg" alt="Flying Kites">
<img src="sky.jpg" alt="Cloudy Sky">
<img src="balloons.jpg" alt="Hot Air Balloons">
```

Note: Like `<br>` , the `<img>` element is also an empty element, and does not have a closing tag.

The Src Attribute of Images

Every image has a `src` attribute (*src* stands for *source*). The src attribute tells the browser where to find the image you want to display. The URL of the image provided as the value of src attribute points to the location where the image is stored.

For Example, An image named "sky.jpg", located in the "images" directory on "www.juniorcollege.com" has the URL:

`https://www.juniorcollege.com/images/sky.jpg`

```
<img url="https://www.juniorcollege.com/images/sky.jpg"
alt="Sky">
```

The Alt Attribute of Images

The `alt` attribute is the alternative description for an image, if the image cannot be displayed. The value of the alt attribute is an author-defined text.

```
<img url="https://www.juniorcollege.com/images/kites.jpg"
alt="Kites">
```

Note: The required alt attribute provides alternative information for an image if a user for some reason cannot able to view it because of slow connection, an error in the src attribute, or if the user uses a screen reader.

Setting Width and Height of an Image

The width and height attributes are used to specify the height and width of an image.

The attribute values are specified in pixels by default.

```
<img src="kites.jpg" alt="Flying Kites" width="300"
height="300">
```

```
<img src="sky.jpg" alt="Cloudy Sky" width="250" height="150">

<img src="balloons.jpg" alt="Hot Air Balloons" width="200"
height="200">
```

Note: It's a good practice to specify both the width and height attributes for an image. In the absence of these attributes the image loads with its original size.

HTML Tables

The HTML table allows to arrange data like, text, images, links, forms, form fields, other tables, etc. into rows and columns of cells.

Creating HTML Tables

Tables in HTML are defined with the `<table>` tag.

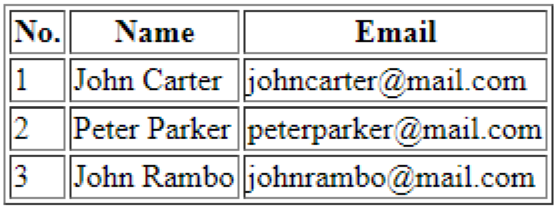
A table is divided into rows with the `<tr>` tag, which stands for table row, and each row is divided into data cells with the `<td>` tag, which stands for table data.

A `<td>` tag can contain text, links, images, lists, forms, other tables, etc.

```
<table border="1">
    <thead>
        <tr>
            <th>No.</th>
            <th>Name</th>
            <th>Email</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <td>1</td>
            <td>John Carter</td>
            <td>johncarter@mail.com</td>
        </tr>
        <tr>
            <td>2</td>
            <td>Peter Parker</td>
```

```
           <td>peterparker@mail.com</td>
       </tr>
       <tr>
           <td>3</td>
           <td>John Rambo</td>
           <td>johnrambo@mail.com</td>
       </tr>
   </tbody>
</table>
```

Output :



No.	Name	Email
1	John Carter	johncarter@mail.com
2	Peter Parker	peterparker@mail.com
3	John Rambo	johnrambo@mail.com

Table Cellpadding and Cellspacing

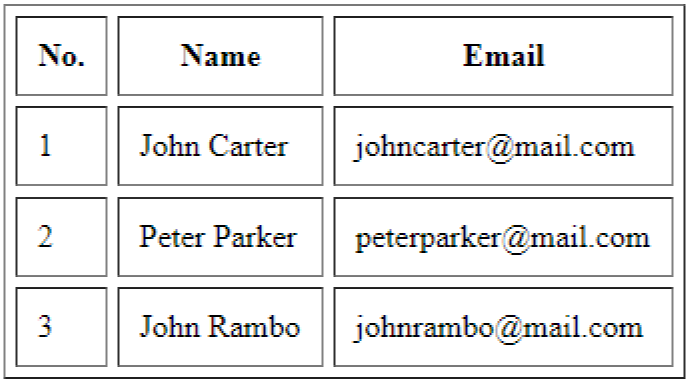
The `cellpadding` and `cellspacing` attributes are used to adjust white space inside a table.

- cellpadding adjust the white space between table cell border and its content.
- cellspacing adjust the white space between table cells.

```
<table border="1" cellpadding="10" cellspacing="5">
    <thead>
        <tr>
            <th>No.</th>
            <th>Name</th>
            <th>Email</th>
        </tr>
    </thead>
    <tbody>
        <tr>
            <td>1</td>
            <td>John Carter</td>
            <td>johncarter@mail.com</td>
        </tr>
        <tr>
            <td>2</td>
            <td>Peter Parker</td>
            <td>peterparker@mail.com</td>
        </tr>
        <tr>
            <td>3</td>
            <td>John Rambo</td>
```

```
                <td>johnrambo@mail.com</td>
            </tr>
        </tbody>
    </table>
```

Output :



No.	Name	Email
1	John Carter	johncarter@mail.com
2	Peter Parker	peterparker@mail.com
3	John Rambo	johnrambo@mail.com

Spanning Multiple Rows and Cells

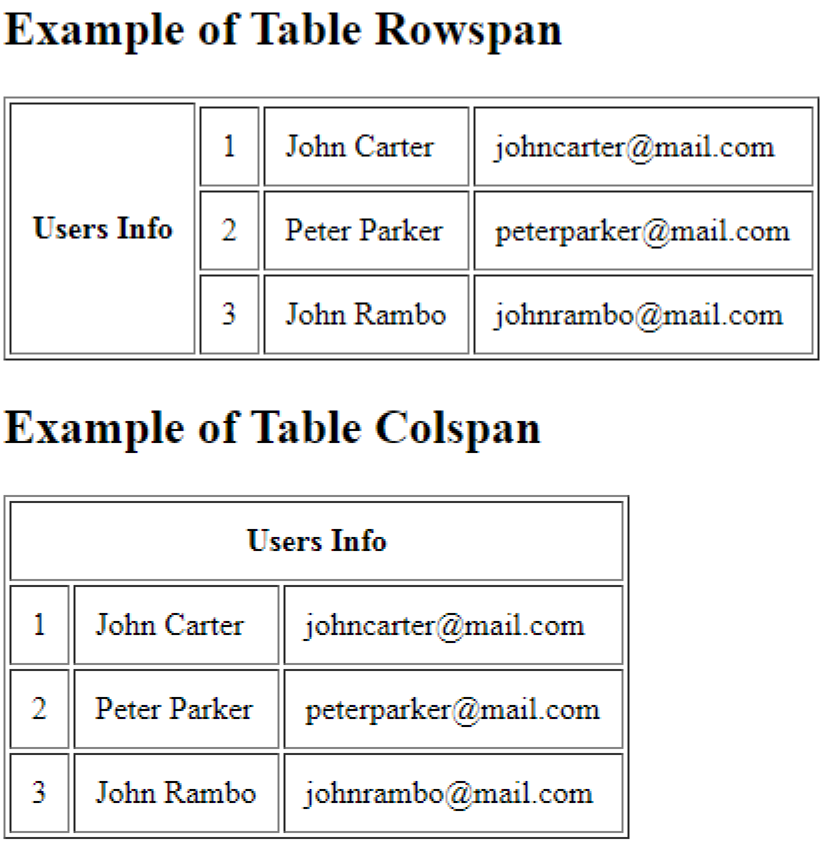
Spanning allow you to extend columns and rows across multiple other columns and rows.

Normally, when we creating a table cell, it cannot pass over into the space below or above another table cell. But, you can use the `colspan` attribute to span multiple columns and `rowspan` attribute to span multiple rows in a table. Here's is an example:

```
<table border="1" cellpadding="10">
    <tr>
        <th rowspan="4">Users Info</th>
    </tr>
    <tr>
        <td>1</td>
        <td>John Carter</td>
        <td>johncarter@mail.com</td>
    </tr>
    <tr>
        <td>2</td>
        <td>Peter Parker</td>
        <td>peterparker@mail.com</td>
    </tr>
    <tr>
        <td>3</td>
        <td>John Rambo</td>
        <td>johnrambo@mail.com</td>
    </tr>
</table>
<h2>Example of Table Colspan</h2>
<table border="1" cellpadding="10">
    <tr>
        <th colspan="3">Users Info</th>
    </tr>
    <tr>
        <td>1</td>
        <td>John Carter</td>
```

```
            <td>johncarter@mail.com</td>
        </tr>
        <tr>
            <td>2</td>
            <td>Peter Parker</td>
            <td>peterparker@mail.com</td>
        </tr>
        <tr>
            <td>3</td>
            <td>John Rambo</td>
            <td>johnrambo@mail.com</td>
        </tr>
    </table>
```

Output :



Example of Table Rowspan

Users Info	1	John Carter	johncarter@mail.com
	2	Peter Parker	peterparker@mail.com
	3	John Rambo	johnrambo@mail.com

Example of Table Colspan

Users Info		
1	John Carter	johncarter@mail.com
2	Peter Parker	peterparker@mail.com
3	John Rambo	johnrambo@mail.com

HTML Lists

HTML lists are used to group related pieces of information together.

Understanding HTML Lists

HTML lists are used to present list of information in well formed and semantic way. There are three different types of list in HTML and each one has a specific purpose and meaning:

- **Unordered list** — Used to group a set of related items, in no particular order.
- **Ordered list** — Used to group a set of related items, in a specific order.
- **Description list** — Used to display a list of terms and their descriptions.

Note: Inside a list item you can put text, line breaks, images, links, etc. You can also place an entire list inside a list item to create nested list.

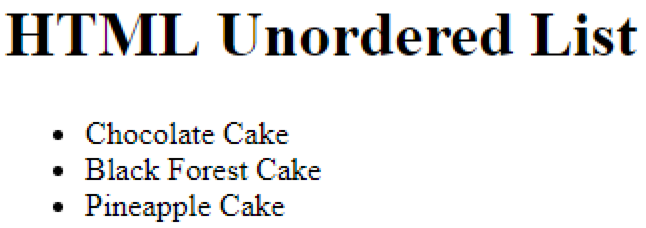
HTML Unordered Lists

An unordered list created using the `<ul>` tag, and each list item starts with the `<li>` tag.

The list items in unordered lists are marked with bullets (small black circles), by default.

```
<body>
     <h1>HTML Unordered List</h1>
     <ul>
        <li>Chocolate Cake</li>
        <li>Black Forest Cake</li>
        <li>Pineapple Cake</li>
    </ul>
</body>
```

Output :



HTML Unordered List

- Chocolate Cake
- Black Forest Cake
- Pineapple Cake

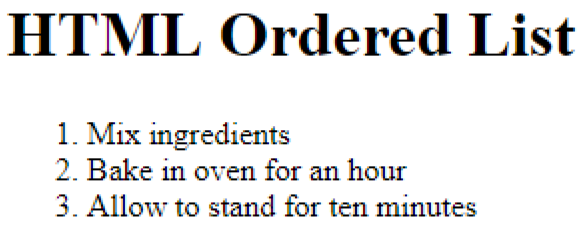
HTML Ordered Lists

An ordered list, created using the `<ol>` tag, and each list item starts with the `<li>` tag. Ordered list contain information where order should be emphasized.

The list items in ordered lists are marked with numbers.

```
<body>
     <h1>HTML Ordered List</h1>
     <ol>
        <li>Mix ingredients</li>
        <li>Bake in oven for an hour</li>
        <li>Allow to stand for ten minutes</li>
    </ol>
</body>
```

Output :



HTML Ordered List

1. Mix ingredients
2. Bake in oven for an hour
3. Allow to stand for ten minutes

HTML Forms

An HTML form is a section of document that contains interactive controls that enable a user to submit information to a web server.

What is HTML Form

HTML Forms are required to collect different kinds of user inputs, such as contact details like name, email address, phone numbers, or details like credit card information, etc.

Forms contain special elements called controls like inputbox, checkboxes, radio-buttons, submit buttons, etc. Users generally complete a form by modifying its controls e.g. entering text, selecting items, etc. and submitting this form to a web server for processing.

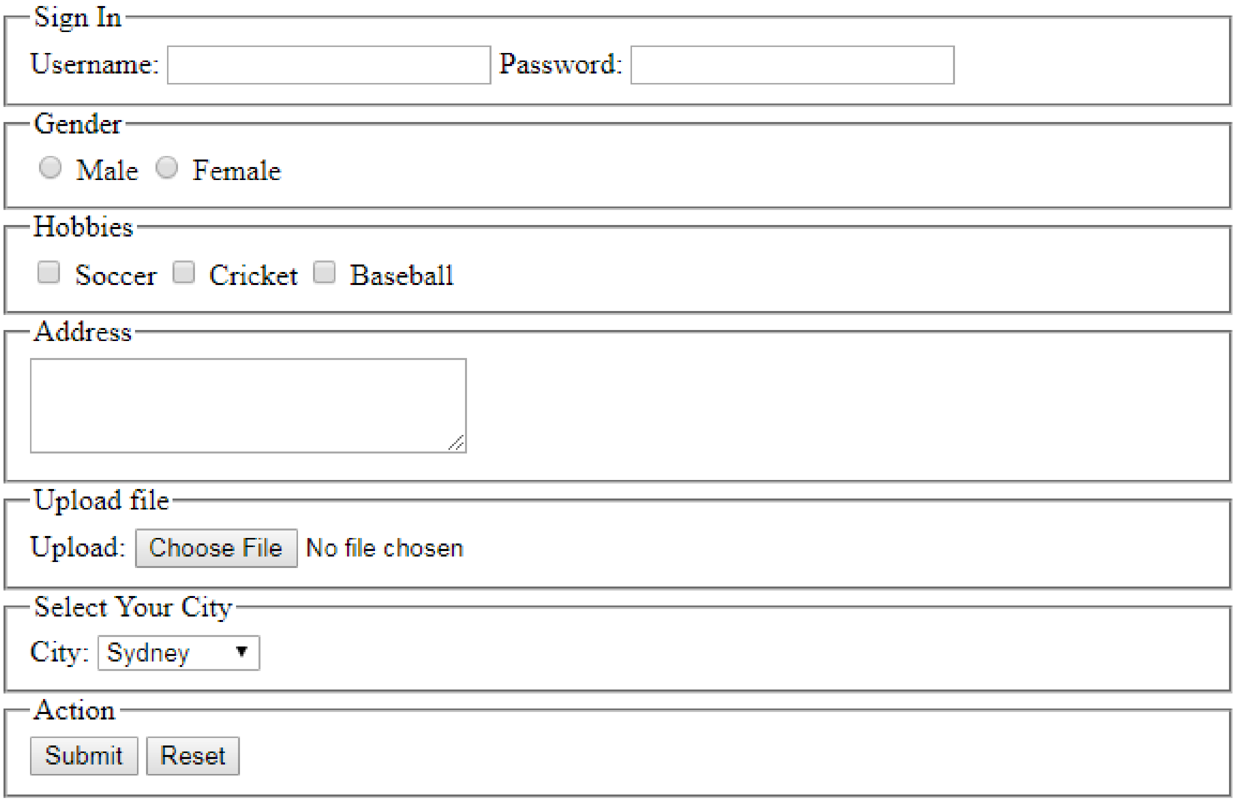
The <form> tag is used to create an HTML form. Here's a simple example of a login form:

```
<!DOCTYPE html>
<head>
    <title>Example of HTML Form Controls</title>
</head>
<body>
     <!-- Browser may generate "Not secure" warning due to
presence of password field in insecure form. We've created
this form just for demo purpose. -->
```

```
    <form>

        <fieldset>

         <legend>Sign In</legend>

            <label for="user-name">Username:</label>

         <input type="text" name="username" id="user-name">

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

         <input type="password" name="user-password"
id="user-pwd">

        </fieldset>

        <fieldset>

         <legend>Gender</legend>

            <input type="radio" name="sex" id="male">

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

            <input type="radio" name="sex" id="female">

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

        </fieldset>

        <fieldset>

         <legend>Hobbies</legend>

            <input type="checkbox" name="sports"
id="soccer">

            <label for="soccer">Soccer</label>

            <input type="checkbox" name="sports"
id="cricket">

            <label for="cricket">Cricket</label>

            <input type="checkbox" name="sports"
id="cricket">

            <label for="baseball">Baseball</label>

        </fieldset>

        <fieldset>

         <legend>Address</legend>
```

```
            <textarea rows="3" cols="30"></textarea>
        </fieldset>
        <fieldset>
         <legend>Upload file</legend>
            <label for="file-select">Upload:</label>
         <input type="file" name="upload" id="file-select">
        </fieldset>
        <fieldset>
         <legend>Select Your City</legend>
            <label for="city">City:</label>
            <select name="city" id="city">
                <option value="sydney">Sydney</option>
                <option
value="melbourne">Melbourne</option>
                <option value="cromwell">Cromwell</option>
            </select>
        </fieldset>
        <fieldset>
         <legend>Action</legend>
            <input type="submit" value="Submit">
         <input type="reset" value="Reset">
        </fieldset>
    </form>
</body>
</html>
```

Output :



Sign In

Username: Password:

Gender

○ Male ○ Female

Hobbies

□ Soccer □ Cricket □ Baseball

Address

Upload file

Upload: Choose File No file chosen

Select Your City

City: Sydney ▼

Action

Submit Reset

The following section describes different types of controls that you can use in your form.

Input Element

This is the most commonly used element within HTML forms.

It allows you to specify various types of user input fields, depending on the type attribute. An input element can be of type *text field*, *checkbox*, *password field*, *radio button*, *submit button*, *reset button*, etc. and several new input types introduced in HTML5.

The most used input types are described below.

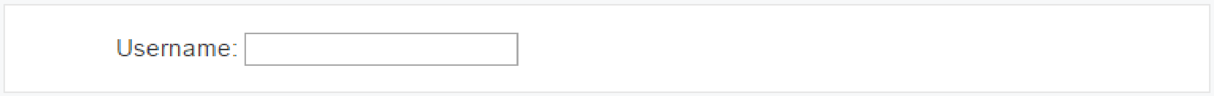
Text Fields

Text fields are one line areas that allow the user to input text.

Single-line text input controls are created using an `<input>` element, whose `type` attribute has a value of `text`. Here's an example of a single-line text input used to take username:

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

— The output of the above example will look something like this:


Username:

Note: The <label> tag to define labels for <input> elements. If you want your user to enter several lines you should use a <textarea> instead.

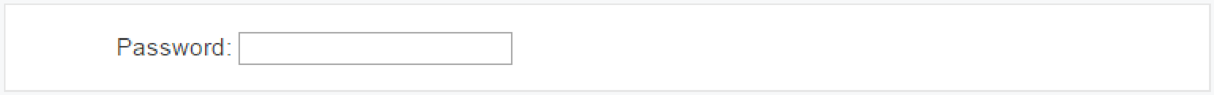
Password Field

Password fields are similar to text fields. The only difference is; characters in a password field are masked i.e. shown as asterisks or dots. This is to prevent others from reading the password on the screen. This is also a single-line text input controls created using an `<input>` element whose `type` attribute has a value of `password`.

Here's an example of a single-line password input used to take user password:

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

— The output of the above example will look something like this:

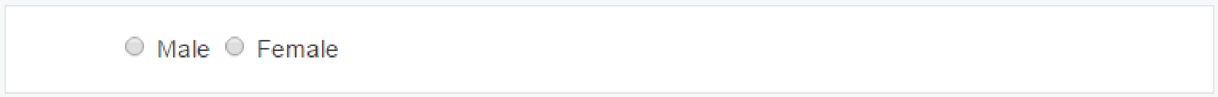

Password:

Radio Buttons

Radio buttons are used to let the user select exactly one option from a pre-defined set of options. It is created using an `<input>` element whose `type` attribute has a value of `radio`.

```
<form>
    <input type="radio" name="sex" id="male">
    <label for="male">Male</label>
    <input type="radio" name="sex" id="female">
    <label for="female">Female</label>
</form>
```

— The output of the above example will look something like this:



○ Male ○ Female

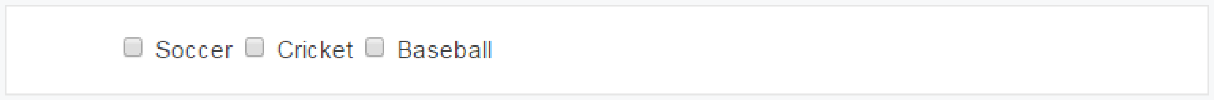
Note: If you want to allow your user to select more than one option at the same time you should use the check boxes instead.

Checkboxes

Checkboxes allows the user to select one or more option from a pre-defined set of options. It is created using an `<input>` element whose `type` attribute has a value of `checkbox`.

```
<form>
    <input type="checkbox" name="sports" id="soccer">
    <label for="soccer">Soccer</label>
    <input type="checkbox" name="sports" id="cricket">
    <label for="cricket">Cricket</label>
    <input type="checkbox" name="sports" id="baseball">
    <label for="baseball">Baseball</label>
</form>
```

— The output of the above example will look something like this:



□ Soccer □ Cricket □ Baseball

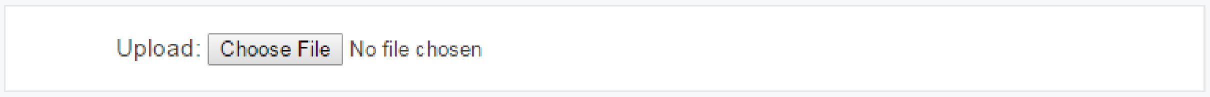
File Select box

The file fields allow a user to browse for a local file and send it as an attachment to the form data. It normally rendered as a text box with a button that enables the user to browse for a file. However, the user can also type the path and name of the file in the text box.

This is also created using an `<input>` element, whose `type` attribute value is set to `file`.

```
<form>
    <label for="file-select">Upload:</label>
    <input type="file" name="upload" id="file-select">
</form>
```

— The output of the above example will look something like this:

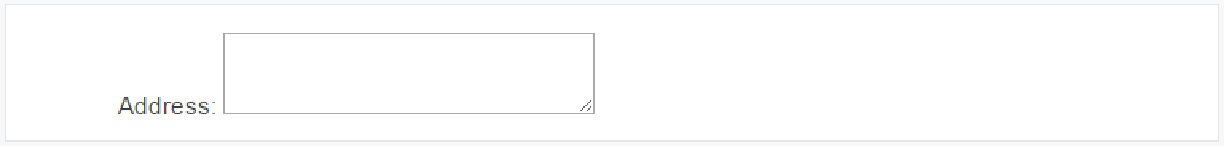


Textarea

Textarea is a multiple-line text input control that allows a user to enter more than one line of text. Multi-line text input controls are created using an `<textarea>` element.

```
<form>
    <label for="address">Address:</label>
    <textarea rows="3" cols="30" name="address"
id="address"></textarea>
</form>
```

— The output of the above example will look something like this:



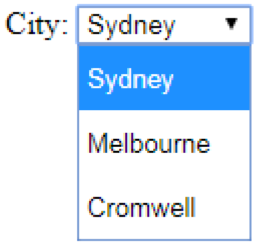
Select Boxes

A select box is a dropdown list of options that allows user to select one or more option from a pull-down list of options. Select box is created using the `<select>` element and <option> element. The option elements within the `<select>` element define each list item.

```
<form>
    <label for="city">City:</label>
    <select name="city" id="city">
        <option value="sydney">Sydney</option>
        <option value="melbourne">Melbourne</option>
        <option value="cromwell">Cromwell</option>
    </select>
```

```
</form>
```

— The output of the above example will look something like this:

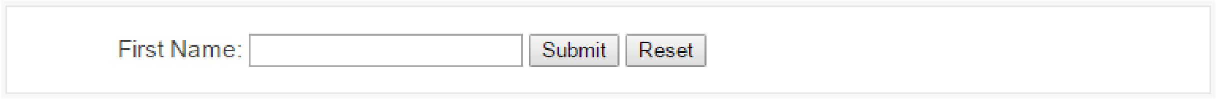


Submit and Reset Buttons

A submit button is used to send the form data to a web server. When submit button is clicked the form data is sent to the file specified in the form's action attribute to process the submitted data. A reset button resets all the forms control to default values.

```
<form action="action.php" method="post" id="users">
    <label for="first-name">First Name:</label>
    <input type="text" name="first-name" id="first-name">
    <input type="submit" value="Submit">
    <input type="reset" value="Reset">
</form>
```

— The output of the above example will look something like this:



Most frequently used form attributes are:

Attribute	Description
action	URL of the program that processes the information submitted via form.
method	The HTTP method that the browser uses to submit the form. Possible values are get and post.
target	A name or keyword indicating the target page where the result of the script will be displayed. The reserved keywords are _blank, _self, _parent and _top.

Appendix A

HTML5 Tags/Elements

The following section contains a brief overview of HTML5 Tags.

HTML5 Tags

This section contains a complete list of standard tags belonging to the latest HTML5 and XHTML 1.1 specifications. All the tags are grouped into categories.

Structural Tags

Tag	Description
<a>	Defines a hyperlink.
<article>	Defines an article.
<aside>	Defines some content loosely related to the page content.
<body>	Defines the document's body.
<br>	Produces a single line break.
<details>	Represents a widget from which the user can obtain additional information or controls on-demand.
<div>	Specifies a division or a section in a document.
<h1> to <h6>	Defines HTML headings.

<head>	Defines the head portion of the document that contains information about the document.
<header>	Represents the header of a document or a section.
<hgroup>	Defines a group of headings.
<hr>	Produce a horizontal line.
<html>	Defines the root of an HTML document.
<footer>	Represents the footer of a document or a section.
<nav>	Defines a section of navigation links.
<p>	Defines a paragraph.
<section>	Defines a section of a document, such as header, footer etc.
<span>	Defines an inline styleless section in a document.
<summary>	Defines a summary for the <details> element.

Metadata Tags

Tag	**Description**
<base>	Defines the base URL for all linked objects on a page.
<link>	Defines the relationship between the current document and an external resource.
<meta>	Provides structured metadata about the document content.
<style>	Inserts style information (commonly CSS) into the head of a document.

<title>	Defines a title for the document.

Form Tags

Tag	Description
<button>	Creates a clickable button.
<datalist>	Represents a set of pre-defined options for an <input> element.
<fieldset>	Specifies a set of related form fields.
<form>	Defines an HTML form for user input.
<input>	Defines an input control.
<keygen>	Represents a control for generating a public-private key pair.
<label>	Defines a label for an <input> control.
<legend>	Defines a caption for a <fieldset> element.
<meter>	Represents a scalar measurement within a known range.
<optgroup>	Defines a group of related options in a selection list.
<option>	Defines an option in a selection list.
<select>	Defines a selection list within a form.
<textarea>	Defines a multi-line text input control (text area).

Formatting Tags

Tag	Description
<abbr>	Defines an abbreviated form of a longer word or phrase.
<acronym>	Defines an acronym.
<address>	Specifies the author's contact information.
<b>	Displays text in a bold style.
<bdi>	Represents text that is isolated from its surrounding for the purposes of bidirectional text formatting.
<bdo>	Overrides the current text direction.
<big>	displays text in a large size.
<blockquote>	Defines a long quotation.
<cite>	Indicates a citation or reference to another source.
<code>	Specifies text as computer code.
<del>	Specifies a block of deleted text.
<dfn>	Specifies a definition.
<em>	Specifies emphasized text.
<i>	Displays text in an italic style.
<ins>	Defines a block of text that has been inserted into a document.
<kbd>	Specifies text as keyboard input.
<mark>	Represents text highlighted for reference purposes.

<output>	Represents the result of a calculation.
<pre>	Defines a block of preformatted text.
<progress>	Represents the completion progress of a task.
<q>	Defines a short inline quotation.
<rp>	Provides fall-back parenthesis for browsers that that don't support ruby annotations.
<rt>	Defines the pronunciation of character presented in a ruby annotations.
<ruby>	Represents a ruby annotation.
<samp>	Specifies text as sample output from a computer program.
<small>	Displays text in a smaller size.
<strong>	Indicate strongly emphasized text.
<sub>	Defines subscripted text.
<sup>	Defines superscripted text.
<tt>	Displays text in a teletype style.
<var>	Defines a variable.
<wbr>	Represents a line break opportunity.

List Tags

Tag	Description

<dd>	Specifies a definition for a term in a definition list.
<dl>	Defines a definition list.
<dt>	Defines a term (an item) in a definition list.
<li>	Defines a list item.
<ol>	Defines an ordered list.
<menu>	Represents a list of commands.
<ul>	Defines an unordered list.

Table Tags

Tag	Description
<caption>	Defines the title of a table.
<col>	Defines attribute values for one or more columns in a table.
<colgroup>	Specifies attributes for multiple columns in a table.
<table>	Defines a data table.
<tbody>	Groups a set of rows defining the main body of the table data.
<td>	Defines a cell in a table.
<tfoot>	Groups a set of rows summarizing the columns of the table.
<thead>	Groups a set of rows that describes the column labels of a table.
<th>	Defines a header cell in a table.

<tr>	Defines a row of cells in a table.

Scripting Tags

Tag	Description
<noscript>	Defines alternative content to display when the browser doesn't support scripting.
<script>	Places script in the document for client-side processing.

Embedded Content Tags

Tag	Description
<area>	Defines a specific area within an image map.
<audio>	Embeds a sound, or an audio stream in an HTML document.
<canvas>	Defines a region in the document, which can be used to draw graphics on the fly via scripting (usually JavaScript).
<embed>	Embeds external application, typically multimedia content like audio or video into an HTML document.
<figcaption>	Defines a caption or legend for a figure.
<figure>	Represents a figure illustrated as part of the document.
<frame>	Defines a single frame within a frameset.
<frameset	Defines a collection of frames or other frameset.

>	
<iframe>	Displays a URL in an inline frame.
<img>	Displays an inline image.
<map>	Defines a client-side image-map.
<noframes>	Defines an alternate content that displays in browsers that do not support frames.
<object>	Defines an embedded object.
<param>	Defines a parameter for an object or applet element.
<source>	Defines alternative media resources for the media elements like <audio> or <video>.
<time>	Represents a time and/or date.
<video>	Embeds video content in an HTML document.

Appendix B

HTML Character Entities

The following sections present the complete lists of character entity references.

Essential Entities

The following table lists the the essential entities in HTML.

Character	Entity Name	Entity Number	Description
&	&amp;	&#38;	Ampersand
"	&quot;	&#34;	Double quote mark
<	&lt;	&#60;	Less than symbol
>	&gt;	&#62;	Greater than symbol
'	&apos;	&#39;	Apostrophe (***XHTML only***)

Currency Symbols

The following table lists the entities for currency symbols.

Character	Entity Name	Entity Number	Description
¢	&cent;	&#162;	Cent
£	&pound;	&#163;	Pound

¤	&curren;	&#164;	General currency
¥	&yen;	&#165;	Yen
€	&euro;	&#8364;	Euro

Copyright, Trademark, and Registered Symbol

The following table lists the entities for copyright, trademark, and registered symbol.

Character	Entity Name	Entity Number	Description
©	&copy;	&#169;	Copyright
®	&reg;	&#174;	Registered
™	&trade;	&#8482;	Trademark

General Punctuation

The following table lists the entities for general punctuation.

Character	Entity Name	Entity Number	Description
	&ensp;	&#8194;	En space
	&emsp;	&#8195;	Em space
	&thinsp;	&#8201;	Thin space
	&nbsp;	&#160;	Nonbreaking space

–	&ndash;	&#8211;	En dash
—	&mdash;	&#8212;	Em dash
‘	&lsquo;	&#8216;	Left/Opening single-quote
’	&rsquo;	&#8217;	Right/Closing single-quote and apostrophe
‚	&sbquo;	&#8218;	Single low-9 quotation mark
“	&ldquo;	&#8220;	Left/Opening double-quote
”	&rdquo;	&#8221;	Right/Closing double-quote
„	&bdquo;	&#8222;	Double low-9 quotation mark
‹	&lsaquo;	&#8249;	Left-pointing single angle quotation mark
›	&rsaquo;	&#8250;	Left-pointing single angle quotation mark
«	&laquo;	&#171;	Left-pointing double angle quotation mark
»	&raquo;	&#187;	Right-pointing double angle quotation mark
†	&dagger;	&#8224;	Dagger
‡	&Dagger;	&#8224;	Double dagger
•	&bull;	&#8226;	Bullet
…	&hellep;	&#8230;	Ellipses
‰	&permil;	&#8240;	Per mille symbol (per thousand)
′	&prime;	&#8242;	Prime, minutes, feet
″	&Prime;	&#8243;	Double prime, seconds, inches
‾	&oline;	&#8254;	Overline
⁄	&frasl;	&#8260;	Fraction slash

Arrows

The following table lists the entities for arrows.

Character	Entity Name	Entity Number	Description
←	&larr;	&#8592;	Left arrow
↑	&uarr;	&#8593;	Up arrow
→	&rarr;	&#8594;	Right arrow
↓	&darr;	&#8595;	Down arrow
↔	&harr;	&#8596;	Left-right arrow
↵	&crarr;	&#8629;	Down arrow with corner leftward
⇐	&lArr;	&#8656;	Leftward double arrow
⇑	&uArr;	&#8657;	Upward double arrow
⇒	&rArr;	&#8658;	Rightward double arrow
⇓	&dArr;	&#8659;	Downward double arrow
⇔	&hArr;	&#8660;	Left-right double arrow

Mathematical Symbols

The following table lists the entities for mathematical symbols.

Character	Entity Name	Entity Number	Description
∀	&forall;	&#8704;	for all

∂	&part;	&#8706;	Partial differential
∃	&exist;	&#8707;	There exists
∅	&empty;	&#8709;	Empty set, null set, diameter
∇	&nabla;	&#8711;	Nabla, backward difference
∈	&isin;	&#8712;	Element of
∉	&notin;	&#8713;	Not an element of
∋	&ni;	&#8715;	Contains as a member
∏	&prod;	&#8719;	N-ary product, product sign
∑	&sum;	&#8721;	N-ary summation
−	&minus;	&#8722;	Minus sign
∗	&lowast;	&#8727;	Asterisk operator
√	&radic;	&#8730;	Square root, radical sign
∝	&prop;	&#8733;	Proportional to
∞	&infin;	&#8734;	Infinity
∠	&ang;	&#8736;	Angle
∧	&and;	&#8743;	Logical and, wedge
∨	&or;	&#8744;	Logical or, vee
∩	&cap;	&#8745;	Intersection, cap
∪	&cup;	&#8746;	Union, cup
∫	&int;	&#8747;	Integral
∴	&there4;	&#8756;	Therefore

∼	&sim;	&#8764;	Tilde operator, varies with, similar to
≅	&cong;	&#8773;	Approximately equal to
≈	&asymp;	&#8776;	Almost equal to, asymptotic to
≠	&ne;	&#8800;	Not equal to
≡	&equiv;	&#8801;	Equivalent to
≤	&le;	&#8804;	Less than or equal to
≥	&ge;	&#8805;	Greater than or equal to
⊂	&sub;	&#8834;	Subset of
⊃	&sup;	&#8835;	Superset of
⊄	&nsub;	&#8836;	Not a subset of
⊆	&sube;	&#8838;	Subset of or equal to
⊇	&supe;	&#8839;	Superset of or equal to
⊕	&oplus;	&#8853;	Circled plus, direct sum
⊗	&otimes;	&#8855;	Circled times, vector product
⊥	&perp;	&#8869;	Up tack, orthogonal to, perpendicular
⋅	&sdot;	&#8901;	Dot operator

Greek Letters

The following table lists the entities for greek letters.

Character	**Entity Name**	**Entity Number**	**Description**

Α	&Alpha;	&#913;	Greek capital alpha
Β	&Beta;	&#914;	Greek capital beta
Γ	&Gamma;	&#915;	Greek capital gamma
Δ	&Delta;	&#916;	Greek capital delta
Ε	&Epsilon;	&#917;	Greek capital epsilon
Ζ	&Zeta;	&#918;	Greek capital zeta
Η	&Eta;	&#919;	Greek capital eta
Θ	&Theta;	&#920;	Greek capital theta
Ι	&Iota;	&#921;	Greek capital iota
Κ	&Kappa;	&#922;	Greek capital kappa
Λ	&Lambda;	&#923;	Greek capital lambda
Μ	&Mu;	&#924;	Greek capital mu
Ν	&Nu;	&#925;	Greek capital nu
Ξ	&Xi;	&#926;	Greek capital xi
Ο	&Omicron;	&#927;	Greek capital omicron
Π	&Pi;	&#928;	Greek capital pi
Ρ	&Rho;	&#929;	Greek capital rho
Σ	&Sigma;	&#931;	Greek capital sigma
Τ	&Tau;	&#932;	Greek capital tau
Υ	&Upsilon;	&#933;	Greek capital upsilon
Φ	&Phi;	&#934;	Greek capital phi

Χ	&Chi;	&#935;	Greek capital chi
Ψ	&Psi;	&#936;	Greek capital psi
Ω	&Omega;	&#937;	Greek capital omega
α	&alpha;	&#945;	Greek small alpha
β	&beta;	&#946;	Greek small beta
γ	&gamma;	&#947;	Greek small gamma
δ	&delta;	&#948;	Greek small delta
ε	&epsilon;	&#949;	Greek small epsilon
ζ	&zeta;	&#950;	Greek small zeta
η	&eta;	&#951;	Greek small eta
θ	&theta;	&#952;	Greek small theta
ι	&iota;	&#953;	Greek small iota
κ	&kappa;	&#954;	Greek small kappa
λ	&lambda;	&#955;	Greek small lambda
μ	&mu;	&#956;	Greek small mu
ν	&nu;	&#957;	Greek small nu
ξ	&xi;	&#958;	Greek small xi
ο	&omicron;	&#959;	Greek small omicron
π	&pi;	&#960;	Greek small pi
ρ	&rho;	&#961;	Greek small rho
ς	&sigmaf;	&#962;	Greek small letter final sigma

σ	&sigma;	&#963;	Greek small sigma
τ	&tau;	&#964;	Greek small tau
υ	&upsilon;	&#965;	Greek small upsilon
φ	&phi;	&#966;	Greek small phi
χ	&chi;	&#967;	Greek small chi
ψ	&psi;	&#968;	Greek small psi
ω	&omega;	&#969;	Greek small omega
ϑ	&thetasym;	&#977;	Greek small theta symbol
ϒ	&upsih;	&#978;	Greek upsilon with hook
ϖ	&piv;	&#982;	Greek pi symbol

Miscellaneous

The following table lists the other useful entities supported by HTML.

Character	Entity Name	Entity Number	Description
⌈	&lceil;	&#8968;	Left ceiling
⌉	&rceil;	&#8969;	Right ceiling
⌊	&lfloor;	&#8970;	Left floor
⌋	&rfloor;	&#8971;	Right floor
⟨	&lang;	&#10216;	Left-pointing angle bracket
⟩	&rang;	&#10217;	Right-pointing angle bracket
◊	&loz;	&#9674;	Lozenge
ℑ	&image;	&#8465;	Blackletter capital I, imaginary part
℘	&weierp;	&#8472;	Script capital P, power set
ℜ	&real;	&#8476;	Blackletter capital R, real part
ℵ	&alefsym;	&#8501;	Alef symbol, or first transfinite cardinal
♠	&spades;	&#9824;	Black spade suit
♣	&clubs;	&#9827;	Black club suit
♥	&hearts;	&#9829;	Black heart suit
♦	&diams;	&#9830;	Black diamond suit

https://www.tutorialrepublic.com/html-tutorial/