# Chapter 2

# Hyper Text Markup Language (HTML)

# Introduction to HTML

## What is HTML?

HTML is an acronym which stands for Hyper Text Markup Language.

**Hyper Text: Hyper Text** simply means "Text within Text". A text has a link within it, is a hypertext. Every time when you click on a word which brings you to a new webpage, you have clicked on a hypertext.

**Markup language:** A markup language is a programming language that is used to make text more interactive and dynamic. It can turn a text into images, tables, links etc.

* HTML is used to create web pages.
* HTML is widely used language on the web.
* We can create static website by HTML only.

## What do you need to get started?

To get started HTML you need a computer installed with HTML editor and web browser.

* **HTML Editor:** HTML can be written and edited using many different editors like Dreamweaver and Visual Studio. However, in this course we use a plain text editor (like Notepad, Note pad ++) to edit HTML.
* **Web Browser:** such as Mozilla Firefox, Google Chrome, Opera etc.

# HTML Basics, Elements and Attributes

An HTML document is made of many HTML tags and each HTML tag contains different content.

# HTML Tags

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.

**Syntax:** <tag> content </tag>

* HTML tags are keywords surrounded by **angle brackets** like <html>
* HTML tags normally **come in pairs** like <b> and </b>
* The first tag in a pair is the **start tag,** the second tag is the **end tag**
* Start and end tags are also called **opening tags** and **closing tags**

## Unclosed HTML Tags

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

* **<br> Tag**: br stands for break line, it breaks the line of the code.
* **<hr> Tag**: hr stands for Horizontal Rule. This tag is used to put a line across the webpage.

**HTML Documents = Web Pages**

* HTML documents **describe web pages**
* HTML documents **contain HTML tags** and plain text
* HTML documents are also **called web pages**

The purpose of a web browser (like Internet Explorer or Firefox) is to read HTML documents and display them as web pages. The browser does not display the HTML tags, but uses the tags to interpret the content of the page:

|  |
| --- |
| <html><body><h1>My First Heading</h1> <p>My first paragraph.</p> </body></html> |

* The text between <html> and </html> describes the web page
* The text between <body> and </body> is the visible page content
* The text between <h1> and </h1> is displayed as a heading
* The text between <p> and </p> is displayed as a paragraph

|  |  |
| --- | --- |
| **Tag** | **Description** |
| [<head>](https://www.w3schools.com/tags/tag_head.asp) | Defines information about the document |
| [<title>](https://www.w3schools.com/tags/tag_title.asp) | Defines the title of a document |
| [<base>](https://www.w3schools.com/tags/tag_base.asp) | Defines a default address or a default target for all links on a page |
| [<link>](https://www.w3schools.com/tags/tag_link.asp) | Defines the relationship between a document and an external resource |
| [<meta>](https://www.w3schools.com/tags/tag_meta.asp) | Defines metadata about an HTML document |
| [<script>](https://www.w3schools.com/tags/tag_script.asp) | Defines a client-side script |
| [<style>](https://www.w3schools.com/tags/tag_style.asp) | Defines style information for a document |

The HTML <head> element has nothing to do with HTML headings. The <head> element is a container for metadata. HTML metadata is data about the HTML document. Metadata is not displayed. The <head> element is placed between the <html> tag and the <body> tag:

### Example

<!DOCTYPE html>  
<html><head> <title>My First HTML</title>  <meta charset="UTF-8"></head><body><h1>This is my first html page</h1></body></html>  
**Note:** Metadata typically define the document title, character set, styles, links, scripts, and other meta information.

**How to View HTML Source code?**

:

* Right-click in an HTML page and select "View Page Source" (in Chrome) or "View Source" (in IE), or similar in other browsers. This will open a window containing the HTML source code of the page.

## HTML Elements

An HTML element usually consists of a **start** tag and **end** tag, with the content inserted in between: <tagname>Content goes here...</tagname>

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

<p>My first paragraph.</p>

|  |  |  |
| --- | --- | --- |
| **Start tag** | **Element content** | **End tag** |
| <h1> | My First Heading | </h1> |
| <p> | My first paragraph. | </p> |
| <br> |  |  |

HTML elements with no content are called **empty elements**. Empty elements do not have an end tag, such as the <br> element (which indicates a line break).

## Nested HTML Elements

HTML elements can be nested (elements can contain elements).

All HTML documents consist of nested HTML elements.

This example contains four HTML elements:

**Example**

<!DOCTYPE html> <html><body><h1>My First Heading</h1><p>My first paragraph.</p></body></html>

The <html> element defines the **whole document**. It has a **start** tag <html> and an **end** tag </html>. The element **content** is another HTML element (the <body> element). The <body> element defines the **document body**. It has a **start** tag <body> and an **end** tag </body>. The element **content** is two other HTML elements (<h1> and <p>). The <h1> element defines a **heading**. It has a **start** tag <h1> and an **end** tag </h1>. The element **content** is: My First Heading. The <p> element defines a **paragraph**. It has a **start** tag <p> and an **end** tag </p>.

## HTML Attributes

* All HTML elements can have **attributes**
* Attributes provide **additional information** about an element
* Attributes are always specified in **the start tag**
* Attributes usually come in name/value pairs like: **name="value"**

The HTML element attribute has the following **syntax**:

<tagname attributename="property:value">

### Example: style, href ,src are html attributes

* <a **href**="https://www.google.come">This is a link</a> - href(hypertext reference-source of file used by the tag) is used to specify a link address.
* <img **src**="img.jpg"> - src is used to specify filename of an image source.
* <img src="img.jpg" width="500" height="600"> - Images in HTML have a set of **size** attributes, which specifies the width and height of the image
* <p style="color:red;">I am a paragraph</p> - style attribute is used to specify the styling of an element. Such as color, font, size etc.

# HTML Basic Elements function

## HTML Headings

Headings are defined with the <h1> to <h6> tags. <h1> defines the most important heading. <h6> defines the least important heading.

### Example

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

**Note:** Browsers automatically add some white space (a margin) before and after a heading.

## 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:

### Example

<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>

## HTML Paragraphs

The HTML <p> element defines a **paragraph**:

**Example**

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

**Note:** Browsers automatically add some white space (a margin) before and after a paragraph.

## HTML Line Breaks

The HTML <br> element defines a **line break**.

Use <br> if you want a line break (a new line) without starting a new paragraph:

**Example**

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

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

## HTML Formatting Elements

Like Style attribute, HTML also defines special **elements** for defining text with a special **meaning**. HTML uses elements like <b> and <i> for formatting output, like **bold** or *italic* text.

Formatting elements were designed to display special types of text:

## HTML <b> and <strong> Elements

The HTML <b> element defines **bold** text, without any extra importance.

**Example**

<b>This text is bold</b>

The HTML <strong> element defines **strong** text, with added semantic "strong" importance.

***Example***

<strong>This text is strong</strong>

## HTML <i> and <em> Elements

The HTML <i> element defines italic text, without any extra importance.

Example

<i>This text is italic</i>

The HTML <em> element defines emphasized text, with added semantic importance.

Example

<em>This text is emphasized</em>

**Note:** Browsers display <strong> as <b>, and <em> as <i>. However, there is a difference in the meaning of these tags: <b> and <i> defines bold and italic text, but <strong> and <em> means that the text is "important".

## HTML <mark> Element

The HTML <mark> element defines marked or highlighted text:

**Example**

<h2>HTML <mark>Marked</mark> Formatting</h2>

## HTML <sub> Element

The HTML <sub> element defines subscripted text.

**Example**

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

## HTML <sup> Element

The HTML <sup> element defines superscripted text.

Example

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

# HTML Comments

Comment tags are used to insert comments in the HTML source code. 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 opening tag, but not in the closing tag.

**Note:** Comments are not displayed by the browser, but they can help document your HTML source code.

With comments you can place notifications and reminders in your HTML:

**Example**

<!-- This is a comment --> <p>This is a paragraph.</p>

# HTML Entities

Reserved characters in HTML must be replaced with character entities. Characters that are not present on your keyboard can also be replaced by entities. Some characters are reserved in HTML. If you use the less than (<) or greater than (>) signs in your text, the browser might mix them with tags. Character entities are used to display reserved characters in HTML.

A character entity looks like this: &*entity\_name*; OR &#*entity\_number*;

To display a less than sign (<) we must write: **&lt;** or **&#60;**

Example: Non-breaking Space **(&nbsp;)**

A common character entity used in HTML is the non-breaking space: **&nbsp;**

A non-breaking space is a space that will not break into a new line.

Two words separated by a non-breaking space will stick together (not break into a new line). This is handy when breaking the words might be disruptive.

## 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; |

**Note:** Entity names are case sensitive.

# HTML links, images, tables and lists

# HTML Links

Links are found in nearly all web pages. Links allow users to click their way from page to page. 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. It can be an image or any other HTML element.

## HTML Links - Syntax

In HTML, links are defined with the <a> tag: <a href="*url of required web page*">*link text*</a>

Example: <a href="https://www.dmu.edu.et/">Visit our website</a>

The href attribute specifies the destination address (https://www.dmu.edu.et) of the link.

The **link text** is the visible part (Visit our website). Clicking on the link text will send you to the specified address.

## Local Links

The example above used an absolute URL (a full web address). A local link (link to the same web site) is specified with a relative URL (without https://www....).

Example: <a href="vacancy.html">Vacancy</a>

## HTML Link Colors

By default, a link will appear like this (in all browsers):

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

You can change the default colors, by using CSS.

## HTML Links - The target Attribute

The target attribute specifies where to open the linked document. The target attribute can have one of the following values:

* \_blank - Opens the linked document in a new window or tab
* \_self - Opens the linked document in the same window/tab as it was clicked (this is default)
* \_parent - Opens the linked document in the parent frame
* \_top - Opens the linked document in the full body of the window
* framename - Opens the linked document in a named frame

**Tip:** If your webpage is locked in a frame, you can use target="\_top" to break out of the frame:

External pages can be referenced with a full URL or with a path relative to the current web page. This example uses a full URL to link to a web page:

*Example*: <a href="https://www.w3schools.com/html/default.asp">HTML tutorial</a>

This example links to a page located in the html folder on the current web site:

*Example*: <a href="/html/default.asp">HTML tutorial</a>

This example links to a page located in the same folder as the current page:

*Example*: <a href="default.asp">HTML tutorial</a>

# HTML Images

Images can improve the design and the appearance of a web page. In HTML, images are defined with the <img> tag. The <img> tag is empty, it contains attributes only, and does not have a closing tag. The src attribute specifies the URL (web address) of the image:

<img src="*url of image*">

**The alt Attribute**

The alt attribute provides an alternate text for an image, if the user for some reason cannot view it (because of slow connection, an error in the src attribute, or if the user uses a screen reader). The value of the alt attribute should describe the image.

**Image Size - Width and Height**

You can use the style attribute to specify the width and height of an image. The width and height attributes always define the width and height of the image in pixels.

**Example:** <img src="img\_dmulogo.jpg" alt="dmu logo" style="width:500px;height:600px;">

Alternatively, you can use the width and height attributes:

<img src=" img\_dmulogo.jpg" alt=" dmu logo " width="500" height="600">

# HTML Audio Tag

**HTML audio tag** is used to define sounds such as music and other audio clips. Audio tag supports some file formats such as : mp3.

HTML5 supports <video> and <audio> controls. The Flash, Silverlight and similar technologies are used to play the multimedia items.

## Example:

<audio controls>  
  <source src="horse.ogg" type="audio/ogg">  
  <source src="horse.mp3" type="audio/mpeg">  
Your browser does not support the audio element.  
</audio>

# HTML Video Tag

HTML 5 supports <video> tag also. The HTML video tag is used for streaming video files such as a movie clip, song clip on the web page. HTML5 Video tag supports some file formats such as mp4, webM and ogg

Example:

<video width="320" height="240" controls>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogg" type="video/ogg">  
Your browser does not support the video tag.  
</video>

# HTML Tables

An HTML table is defined with the <table> tag. Each table row is defined with the <tr> tag. A table header is defined with the <th> tag. By default, table headings are bold and centered. A table data/cell is defined with the <td> tag.

### Example

<table style="width:100%">  
  <tr>    <th>Firstname</th>     <th>Lastname</th>     <th>Age</th>   </tr>  
  <tr>     <td>Selam</td>    <td>Taye</td>     <td>50</td>  </tr>  
  <tr>    <td>Abel</td>    <td>Mekonnen</td>     <td>44</td>  </tr>  
</table>

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

## HTML Table - Cells that Span Many Columns

To make a cell span more than one column, use the colspan attribute:

Example

<table style="width:100%">  <tr>    <th>Name</th>   <th colspan="2">Telephone</th>  </tr>  
  <tr>    <td>Abel Tilahun</td>    <td>55577854</td>    <td>55577855</td>  </tr> </table>

## HTML Table - Cells that Span Many Rows

To make a cell span more than one row, use the rowspan attribute:

### Example

<table style="width:100%">  <tr>    <th>Name:</th>    <td>Abel Tilahun</td>  </tr>  <tr>    <th rowspan="2">Telephone:</th>    <td>55577854</td>  </tr> <tr>    <td>55577855</td>  </tr></table>

## HTML Table - Adding a Caption

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

Example

<table style="width:100%">  <caption>Monthly savings</caption>  
  <tr>    <th>Month</th>    <th>Savings</th>  </tr>  <tr>    <td>January</td>    <td>2000 birr</td>  </tr>

<tr>    <td>February</td>    <td>3000 birr</td>  </tr></table>

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

# HTML 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

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

## Unordered HTML List - Choose List Item Marker

The CSS list-style-type property is used to define the style of the list item marker:

|  |  |
| --- | --- |
| **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

<ul style="list-style-type:disc">   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li></ul>

### 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>

## *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

<ol>   <li>Coffee</li>   <li>Tea</li>   <li>Milk</li> </ol>

**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>

# HTML Iframe and Forms

# HTML Iframes

An iframe is used to display a web page within a web page.

**Iframe Syntax**

An HTML iframe is defined with the <iframe> tag: <iframe src="URL"></iframe>

The src attribute specifies the URL (web address) of the inline frame page.

**Iframe - Set Height and Width**

Use the height and width attributes to specify the size of the iframe. The attribute values are specified in pixels by default, but they can also be in percent (like "80%").

Example

<iframe src="demo\_iframe.htm" height="200" width="300"></iframe>

Or you can use CSS to set the height and width of the iframe:

Example

<iframe src="demo\_iframe.htm" style="height:200px;width:300px;"></iframe>

**Iframe - Remove the Border**

By default, an iframe has a border around it. To remove the border, add the style attribute and use the CSS border property:

Example

<iframe src="demo\_iframe.htm" style="border:none;"></iframe>

With CSS, you can also change the size, style and color of the iframe's border:

Example

<iframe src="demo\_iframe.htm" style="border:2px solid red;"></iframe>

**Iframe - Target for a Link**

An iframe can be used as the target frame for a link.

The target attribute of the link must refer to the name attribute of the iframe:

Example

<iframe src="demo\_iframe.htm" name="iframe\_a"></iframe>  
<p><a href="https://www.w3schools.com" target="iframe\_a">W3Schools.com</a></p>

# HTML Forms

The HTML <form> element defines a form that is used to collect user input:

<form> *form elements* </form>

An HTML form contains **form elements**. Form elements are different types of input elements, like text fields, checkboxes, radio buttons, submit buttons, and more.

**Attributes of Form Tag**

## The Action Attribute

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

Normally, the form data is sent to a web page on the server when the user clicks on the submit button. In the example above, the form data is sent to a page on the server called "/action\_page.html". This page contains a server-side script that handles the form data:

<form **action="/action\_page.html**">

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

## The Target Attribute

The target attribute specifies if the submitted result will open in a new browser tab, a frame, or in the current window.

The default value is "\_self" which means the form will be submitted in the current window.

To make the form result open in a new browser tab, use the value "\_blank":

Example

<form action="/action\_page.html" **target="\_blank"**>

Other legal values are "\_parent", "\_top", or a name representing the name of an iframe.

## The Method Attribute

The method attribute specifies the HTTP method (**GET** or **POST**) to be used when submitting the form data:

Example: <form action="/action\_page.html" **method="get"**> or

<form action="/action\_page.html" **method="post"**>

**When to Use GET?**

The default method when submitting form data is GET. However, when GET is used, the submitted form data will be **visible in the page address field**:

/action\_page.html?firstname=Selam&lastname=Yibeltal

**Notes on GET:**

* Appends form-data into the URL in name/value pairs
* The length of a URL is limited (about 3000 characters)
* Never use GET to send sensitive data! (will be visible in the URL)
* Useful for form submissions where a user wants to bookmark the result
* GET is better for non-secure data, like query strings in Google

**When to Use POST?**

Always use POST if the form data contains sensitive or personal information. The POST method does not display the submitted form data in the page address field.

**Notes on POST:**

POST has no size limitations, and can be used to send large amounts of data.

Form submissions with POST cannot be bookmarked

## The Name Attribute

Each input field must have a name attribute to be submitted. If the name attribute is omitted, the data of that input field will not be sent at all. This example will only submit the "Last name" input field:

Example

<form action="/action\_page.html">  
  First name:<br>  <input type="text" name=”firstname” value="Selam"><br>  
  Last name:<br>  <input type="text" name="lastname" value="Yibeltal"><br><br>  
  <input type="submit" value="Submit">  
</form>

# HTML Form Elements

## The <input> Element

The most important form element is the <input> element. The <input> element can be displayed in several ways, depending on the type attribute.

## HTML Input types

The <input> element is the most important form element. The <input> element can be displayed in several ways, depending on the **type** attribute

## Input Type Text

<input type="text"> defines a **one-line text input field**:

## Input Type Password

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

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

## Input Type Submit

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

The form-handler is typically a server page with a script for processing input data. The form-handler is specified in the form's action attribute:

### Example

<form action="/action\_page.html">  
  First name:<br>   <input type="text" name="firstname" value="Abel"><br>  
  Last name:<br>   <input type="text" name="lastname" value="Degu"><br><br>  
  <input type="submit" value="Submit">  
</form>

If you omit the submit button's value attribute, the button will get a default text.

## Input Type Reset

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

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

## Input Type Radio

<input type="radio"> defines a **radio button**.

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

## Input Type Checkbox

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

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

## Input Type Button

<input type="button"> defines a **button**:

Example

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

## 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. You can also use the min and max attributes to add restrictions to dates:

### Example

<form>  
  Enter a date before 1980-01-01:  <input type="date" name="bday" max="1979-12-31"><br>  
  Enter a date after 2000-01-01:  <input type="date" name="bday" min="2000-01-02"><br>  
</form>

## 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 adds ".com" to the keyboard to match email input.

## Input Type File

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

## Input Type Number

The <input type="number" min="1" max="5">> defines a **numeric** input field. You can also set restrictions on what numbers are accepted.

## Input Restrictions

Here is a list of some common input restrictions (some are new in HTML5):

|  |  |
| --- | --- |
| **Attribute** | **Description** |
| disabled | Specifies that an input field should be disabled |
| max | Specifies the maximum value for an input field |
| maxlength | Specifies the maximum number of character for an input field |
| min | Specifies the minimum value for an input field |
| pattern | Specifies a regular expression to check the input value against |
| readonly | Specifies that an input field is read only (cannot be changed) |
| required | Specifies that an input field is required (must be filled out) |
| size | Specifies the width (in characters) of an input field |
| step | Specifies the legal number intervals for an input field |
| value | Specifies the default value for an input field |

## Input Type Range

The <input type="range"> defines a control for entering a number whose exact value is not important (like a slider control). Default range is 0 to 100. However, you can set restrictions on what numbers are accepted with the min, max, and step attributes:

Example

<form><input type="range" name="points" min="0" max="10"></form>

## Input Type Search

The <input type="search"> is used for search fields (a search field behaves like a regular text field).

## Input Type Time

The <input type="time"> allows the user to select a time (no time zone). Depending on browser support, a time picker can show up in the input field.

## The <select> Element

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

Example

<select name="cars">  <option value="volvo">Volvo</option>  
  <option value="saab">Saab</option>

<option value="fiat">Fiat</option>  
  <option value="audi">Audi</option>

</select>

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

<select 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>

**Allow Multiple Selections:**

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

Example

<select 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>

## The <textarea> Element

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

Example

<textarea name="message" rows="10" cols="30" style="width:200px; height:600px"> The cat was playing in the garden. </textarea>

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

## Grouping Form Data with <fieldset>

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.html">  <fieldset>  
    <legend>Personal information:</legend>  
    First name:<br>    <input type="text" name="firstname" value="Abel"><br>  
    Last name:<br>    <input type="text" name="lastname" value="Degu"><br><br>  
    <input type="submit" value="Submit">  
  </fieldset></form>

## The form Attribute

The form attribute specifies one or more forms an <input> element belongs to.

* The multiple attribute specifies that the user is allowed to enter more than one value in the <input> element. The multiple attribute works with the following input types: email, and file. Example: Select images: <input type="file" name="img" multiple>
* The pattern attribute specifies a regular expression that the <input> element's value is checked against. The pattern attribute works with the following input types: text, search, url, tel, email, and password. Example: <input type="text" name="country\_code" pattern="[A-Za-z]{3}" title="Three letter country code">
* The placeholder attribute specifies a hint that describes the expected value of an input field (a sample value or a short description of the format). The hint is displayed in the input field before the user enters a value. The placeholder attribute works with the following input types: text, search, url, tel, email, and password.
* The required attribute specifies that an input field must be filled out before submitting the form. The required attribute works with the following input types: text, search, url, tel, email, password, date pickers, number, checkbox, radio, and file.

## Migration from HTML4 to HTML5

This section demonstrates how to convert an HTML4 page into an HTML5 page, without destroying anything of the original content or structure.

|  |  |
| --- | --- |
| **Typical HTML4** | **Typical HTML5** |
| <div id="header"> | <header> |
| <div id="menu"> | <nav> |
| <div id="content"> | <section> |
| <div class="article"> | <article> |
| <div id="footer"> | <footer> |

# HTML Layouts

## HTML Layout Elements

Websites often display content in multiple columns (like a magazine or newspaper).

HTML5 offers new semantic elements that define the different parts of a web page:

|  |  |
| --- | --- |
| HTML5 Semantic Elements | * <header> - Defines a header for a document or a section * <nav> - Defines a container for navigation links * <section> - Defines a section in a document * <article> - Defines an independent self-contained article * <aside> - Defines content aside from the content (like a sidebar) * <footer> - Defines a footer for a document or a section * <details> - Defines additional details * <summary> - Defines a heading for the <details> element |

## HTML Layout Techniques

There are five different ways to create multicolumn layouts. Each way has its pros and cons:

* HTML tables (not recommended)
* CSS float property
* CSS flexbox
* CSS framework
* CSS grid

### HTML Tables

The <table> element was not designed to be a layout tool! The purpose of the <table> element is to display tabular data. So, do not use tables for your page layout! They will bring a mess into your code. And imagine how hard it will be to redesign your site after a couple of months.

**Tip:** Do NOT use tables for your page layout!

### CSS Frameworks

If you want to create your layout fast, you can use a framework, like W3.CSS or Bootstrap.

### CSS Floats

It is common to do entire web layouts using the CSS float property. Float is easy to learn - you just need to remember how the float and clear properties work.

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

### CSS Flexbox

Flexbox is a new layout mode in CSS3. Use of flexbox ensures that elements behave predictably when the page layout must accommodate different screen sizes and different display devices.

**Disadvantages:** Does not work in IE10 and earlier.

### CSS Grid View

The CSS Grid Layout Module offers a grid-based layout system, with rows and columns, making it easier to design web pages without having to use floats and positioning.

**Disadvantages:** Does not work in IE nor in Edge 15 and earlier.