



HYPER TEXT MARKUP LANGUAGE

LAXMANA RAO
LAXMANA RAO



HTML

HTML is the standard markup language for Web pages.

➤ What is HTML?

- HTML stands for Hyper Text Markup 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 elements tell the browser how to display the content
- HTML elements label pieces of content such as "this is a heading", "this is a paragraph", "this is a link", etc.

➤ Example Explained

- The `<!DOCTYPE html>` declaration defines that this document is an HTML5 document
- The `<html>` element is the root element of an HTML page
- The `<head>` element contains meta information about the HTML page
- The `<title>` element specifies a title for the HTML page (which is shown in the browser's title bar or in the page's tab)
- The `<body>` element defines the document's body, and is a container for all the visible contents, such as headings, paragraphs, images, hyperlinks, tables, lists, etc.
- The `<h1>` element defines a large heading
- The `<p>` element defines a paragraph

HTML Elements :-

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

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

Examples of some HTML elements:

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

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

➤ Example Explained

The `<html>` element is the root element and it defines the whole HTML document.

It has a start tag `<html>` and an end tag `</html>`.

Then, inside the `<html>` element there is a `<body>` element:

```
<body>

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

</body>
```

The `<body>` element defines the document's body.

It has a start tag `<body>` and an end tag `</body>`.

Then, inside the `<body>` element there are two other elements: `<h1>` and `<p>`:

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

HTML Attributes

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

✓ **The href Attribute**

The `<a>` tag defines a hyperlink. The `href` attribute specifies the URL of the page the link goes to:

Example

```
<a href="https://www.w3schools.com">Visit W3Schools</a>
```

✓ **The src Attribute**

The `` tag is used to embed an image in an HTML page. The `src` attribute specifies the path to the image to be displayed:

Example

```

```

✓ **The width and height Attributes**

The `` tag should also contain the `width` and `height` attributes, which specifies the width and height of the image (in pixels)

Example

```

```

✓ The alt Attribute

The required `alt` attribute for the `` tag specifies an alternate text for an image, if the image for some reason cannot be displayed. This can be due to slow connection, or an error in the `src` attribute, or if the user uses a screen reader.

Example

```

```

✓ 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>
```

Output= **This is a red paragraph.**

✓ 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>
```

➤ 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>Heading 1</h1>
```

```
<h2>Heading 2</h2>
```

```
<h3>Heading 3</h3>
```

```
<h4>Heading 4</h4>
```

```
<h5>Heading 5</h5>
```

```
<h6>Heading 6</h6>
```

[HTML paragraphs](#)

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

Example

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

➤ Solution - The HTML `<pre>` Element

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

Example

```
<pre>
  My Bonnie lies over the ocean.  <!--printing as a poem-->
  My Bonnie lies over the sea.
  My Bonnie lies over the ocean.
  Oh, bring back my Bonnie to me.
</pre>
```

The HTML Style Attribute

Setting the style of an HTML element, can be done with the `style` attribute.

The HTML `style` attribute has the following syntax:

```
<tagname style="property:value;">
```

The **property** is a CSS property. The **value** is a CSS value

✓ Background Color

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

Example

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

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

```
</body>
```

- `<body>`

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

```
</body>
```

✓ Text Color

The CSS `color` property defines the text color for an HTML element:

Example

```
<h1 style="color:blue;">This is a heading</h1>  
<p style="color:red;">This is a paragraph.</p>
```

✓ Fonts

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

Example

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

✓ Text Size

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

Example

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

✓ Text Alignment

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

Example

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

HTML formatting

<tagname>data</tagname>

- `` - Bold text <!-- **This text is bold** -- >
- `` - Important text <!-- **This text is important!** -- >
- `<i>` - Italic text <!-- *This text is italic.* -- >
- `` - Emphasized text <!-- *This text is emphasized.* -- >
- `<mark>` - Marked text <!-- This is some smaller text. -- >

- `<small>` - Smaller text `<!--Do not forget to buy milk today. -->`
- `` - Deleted text `<!--My favorite color is ,red -->`
- `<ins>` - Inserted text `<!-- My favorite color is red.-->`
- `<sub>` - Subscript text `<!--This is subscripted text. -->`
- `<sup>` - Superscript text `<!-- This is superscripted text.-->`

HTML Comments

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

HTML Colors

✓ Border color

Example

```
<h1 style="border:2px solid Tomato;">Hello World</h1>
<h1 style="border:2px solid DodgerBlue;">Hello World</h1>
<h1 style="border:2px solid Violet;">Hello World</h1>
```

HTML Links

The HTML `<a>` tag defines a hyperlink. It has the following syntax:

- `Link text`

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.

- `Visit W3Schools.com!`

Example

Use `target="_blank"` to open the linked document in a new browser window or tab:

- `Visit W3Schools!`

✓ HTML Links - Use an Image as a Link

To use an image as a link, just put the `` tag inside the `<a>` tag:

Example

- `

`

✓ 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):

Example

- `Send email`

✓ Button as a Link

- `<button onclick="document.location='default.asp'">HTML
Tutorial</button>`

[HTML links- create Bookmarks](#)

✓ 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>`

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

- `Jump to Chapter 4`

HTML Images

Images can improve the design and the appearance of a web page.

```
<!DOCTYPE html>

<html>

<body>

<h2>HTML Image</h2>





</body>

</html>
```

✓ The alt Attribute

The required `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:

Example

- ``

If a browser cannot find an image, it will display the value of the `alt` attribute:

- ``
There are two ways to set `width` and `height`
- ``
- ``

✓ Image Floating

Use the CSS `float` property to let the image float to the right or to the left of a text:

Float: right (or) float: left

Example

```
<p>
The image will float to the right of the text.</p>
```

```
<p>
The image will float to the left of the text.</p>
```

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>
```

```
<body>
```

```
<h2>Image Maps</h2>
```

```
<p>Click on the computer, the phone, or the cup of coffee to go to a new
page and read more about the topic:</p>
```

```

```

```
<map name="workmap">
```

```
  <area shape="rect" coords="34,44,270,350" alt="Computer"
href="computer.htm">
```

```
  <area shape="rect" coords="290,172,333,250" alt="Phone"
href="phone.htm">
```

```
  <area shape="circle" coords="337,300,44" alt="Cup of coffee"
href="coffee.htm">
```

```
</map>
```

```
</body>
```

</html>

HTML Background Images

✓ **Background Image**

To add a background image on an HTML element, use the HTML `style` attribute and the CSS `background-image` property:

Example

Add a background image on a HTML element:

```
<div style="background-image: url('img_girl.jpg');">
```

You can also specify the background image in the `<style>` element, in the `<head>` section:

Example

Specify the background image in the `<style>` element:

```
<style>
div
{
  background-image: url('img_girl.jpg');
}
</style>
```

Background Image on a Page

If you want the entire page to have a background image, you must specify the background image on the `<body>` element:

```
<style>
body {
  background-image: url('img_girl.jpg');
}
</style>
```

✓ **Background Repeat**

If the background image is smaller than the element, the image will repeat itself, horizontally and vertically, until it reaches the end of the element:

✓ **Example**

```
<style>
body {
```

```
background-image: url('example_img_girl.jpg');
}
```

To avoid the background image from repeating itself, set the `background-repeat` property to `no-repeat`.

✓ Example

```
<style>
body {
background-image: url('example_img_girl.jpg');
background-repeat: no-repeat;
}
```

✓ Background Cover

If you want the background image to cover the entire element, you can set the `background-size` property to `cover`.

Also, to make sure the entire element is always covered, set the `background-attachment` property to `fixed`:

This way, the background image will cover the entire element, with no stretching (the image will keep its original proportions):

Example

```
<style>
body {
background-image: url('img_girl.jpg');
background-repeat: no-repeat;
background-attachment: fixed;
background-size: cover;
}
```

HTML <picture> ELEMENT

The HTML `<picture>` element gives web developers more flexibility in specifying image resources.

The `<picture>` element contains one or more `<source>` elements, each referring to different images through the `srcset` attribute. This way the browser can choose the image that best fits the current view and/or device.

Each `<source>` element has a `media` attribute that defines when the image is the most suitable.

Example

Show different images for different screen sizes:

```
<picture>
  <source media="(min-width: 650px)" srcset="img_food.jpg">
  <source media="(min-width: 465px)" srcset="img_car.jpg">
  
</picture>
```

When to use the Picture Element

There are two main purposes for the `<picture>` element:

1. Bandwidth

If you have a small screen or device, it is not necessary to load a large image file. The browser will use the first `<source>` element with matching attribute values, and ignore any of the following elements.

2. Format Support

Some browsers or devices may not support all image formats. By using the `<picture>` element, you can add images of all formats, and the browser will use the first format it recognizes, and ignore any of the following elements.

Example

The browser will use the first image format it recognizes:

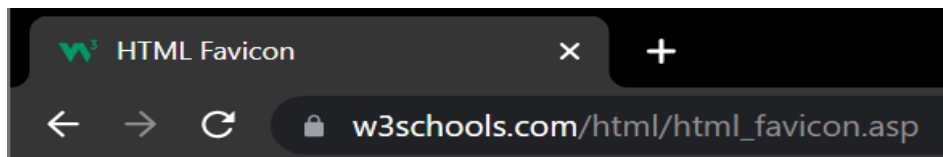
```
<picture>
  <source srcset="img_avatar.png">
  <source srcset="img_girl.jpg">
  
</picture>
```

HTML FAVICON

How To Add a Favicon in HTML

You can use any image you like as your favicon. You can also create your own favicon on sites like <https://favicon.cc>.

A favicon image is displayed to the left of the page title in the browser tab, like this:



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".

Next, add a `<link>` element to your "index.html" file, after the `<title>` element, like this:

```
<!DOCTYPE html>
<html>
<head>
  <title>My Page Title</title>
  <link rel="icon" type="image/x-icon" href="/images/favicon.ico">
</head>
<body>

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

</body>
</html>
```

Tag	Description
<link>	Defines the relationship between a document and an external resource

HTML Tables

✓ Table borders

HTML tables can have borders of different styles and shapes.

✓ How To Add a Border

When you add a border to a table, you also add borders around each table cell:

To add a border, use the CSS `border` property on `table`, `th`, and `td` elements:

Example

```
table, th, td {  
  border: 1px solid black;  
}
```

✓ Collapsed Table Borders

To avoid having double borders like in the example above, set the CSS `border-collapse` property to `collapse`.

This will make the borders collapse into a single border:

Example

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

✓ 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:

--	--	--

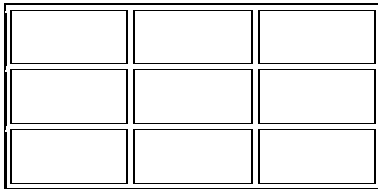


Example

```
table, th, td {  
  border: 1px solid white;  
  border-collapse: collapse;  
}  
th, td {  
  background-color: #96D4D4;  
}
```

✓ Round Table Borders

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



Example

```
table, th, td {  
  border: 1px solid black;  
  border-radius: 10px;  
}
```

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

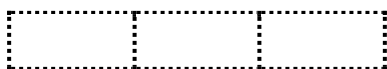


Example

```
th, td {  
  border: 1px solid black;  
  border-radius: 10px;  
}
```









✓ Dotted Table Borders

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





The following values are allowed:

- dotted 
- dashed 
- solid 
- double 
- groove 
- ridge 
- inset 
- outset 
- none
- hidden

Example

```
th, td {  
  border-style: dotted;  
}
```

Border Color

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



Example

```
th, td {  
  border-color: #96D4D4;  
}
```

✓ Creating tables:-

HTML tables allow web developers to arrange data into rows and columns.

```
<!DOCTYPE html>
```

```
<html>
```

```
<style>
```

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

```
<!--this is important for creating table -->
```

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

```
}
```

```
</style>
```

```
<body>
```

Example

Company	Contact	Country
Alfreds Futterkiste	Maria Anders	Germany
Centro comercial Moctezuma	Francisco Chang	Mexico
Ernst Handel	Roland Mendel	Austria
Island Trading	Helen Bennett	UK
Laughing Bacchus Winecellars	Yoshi Tannamuri	Canada
Magazzini Alimentari Riuniti	Giovanni Rovelli	Italy

✓ Define an HTML Table

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

Example

A simple HTML table:

```
<table>
  <tr>
    <th>Company</th>
    <th>Contact</th>
    <th>Country</th>
  </tr>
  <tr>
    <td>Alfreds Futterkiste</td>
    <td>Maria Anders</td>
    <td>Germany</td>
  </tr>
  <tr>
```

```
    <td>Centro comercial Moctezuma</td>
    <td>Francisco Chang</td>
    <td>Mexico</td>
  </tr>
</table>
```

✓ 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>
```

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.

Note: There are times where a row can have less or more cells than another.

Table Headers

Sometimes you want your cells to be headers, in those cases use the `<th>` tag instead of the `<td>` tag:

Example

Let the first row be table headers:

```
<table>
  <tr>
    <th>Person 1</th>
    <th>Person 2</th>
    <th>Person 3</th>
  </tr>
  <tr>
    <td>Emil</td>
    <td>Tobias</td>
    <td>Linus</td>
  </tr>
  <tr>
    <td>16</td>
    <td>14</td>
    <td>10</td>
  </tr>
</table>
```

By default, the text in `<th>` elements are bold and centered, but you can change that with CSS.

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:

Example

Set the width of the table to 100%:

```
<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>
```

Note: Using a percentage as the size unit for a width means how wide will this element be compared to its parent element, which in this case is the `<body>` element.

✓ HTML Table Column Width

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

Example

Set the width of the first column to 70%:

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

HTML Table Row Height

field	field	field
	data	

HTML Table Row Height

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

Example

Set the height of the second row to 200 pixels:

```
<table style="width:100%">
  <tr>
    <th>Firstname</th>
    <th>Lastname</th>
    <th>Age</th>
  </tr>
  <tr style="height:200px">
    <td>Jill</td>
    <td>Smith</td>
    <td>50</td>
  </tr>
```

```
<tr>
  <td>Eve</td>
  <td>Jackson</td>
  <td>94</td>
</tr>
</table>
```

HTML LISTS

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

Example

An unordered HTML list:

- Item
- Item
- Item
- Item

An ordered HTML list:

- 1.First item
- 2.second item
- 3.third item
- 4.fourth item

✓ Unordered HTML List

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

✓ Ordered HTML List

An ordered list starts with the `` tag. Each list item starts with the `` tag.

The list items will be marked with numbers by default:

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

✓ 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:

Example

```
<dl>
  <dt>Coffee</dt>
  <dd>- black hot drink</dd>
  <dt>Milk</dt>
  <dd>- white cold drink</dd>
</dl>
```

HTML List Tags

Tag	Description
<code></code>	Defines an unordered list
<code></code>	Defines an ordered list
<code></code>	Defines a list item
<code><dl></code>	Defines a description list
<code><dt></code>	Defines a term in a description list
<code><dd></code>	Describes the term in a description list

HTML Unordered lists

An unordered list starts with the `` tag. Each list item starts with the `` tag.

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

```
<!DOCTYPE html>
<html>
<body>
<h2>An unordered HTML list</h2>
<ul>
```



```
<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):

```

<!DOCTYPE html>
<html>
<body>

<h2>A Nested List</h2>
<p>Lists can be nested (list inside list):</p>

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

</body>
</html>

```

A Nested List

Lists can be nested (list inside list):

- Coffee
- Tea
 - Black tea
 - Green tea
- Milk

✓ Horizontal List with CSS

HTML lists can be styled in many different ways with CSS.

One popular way is to style a list horizontally, to create a navigation menu:

```

<!DOCTYPE html>
<html>
<head>
<style>
ul {
  list-style-type: none;
  margin: 0;
  padding: 0;
  overflow: hidden;
  background-color: #333333;
}

li {
  float: left;
}

li a {
  display: block;
  color: white;
  text-align: center;
  padding: 16px;
  text-decoration: none;
}

li a:hover {
  background-color: #111111;
}
</style>
</head>
<body>

<h2>Navigation Menu</h2>
<p>In this example, we use CSS to style the list horizontally, to create a navigation menu:</p>

<ul>
  <li><a href="#home">Home</a></li>
  <li><a href="#news">News</a></li>
  <li><a href="#contact">Contact</a></li>
  <li><a href="#about">About</a></li>
</ul>

</body>
</html>

```

Navigation Menu

In this example, we use CSS to style the list horizontally, to create a navigation menu

Home News Contact About

Tag	Description
<code></code>	Defines an unordered list
<code></code>	Defines an ordered list
<code></code>	Defines a list item
<code><dl></code>	Defines a description list
<code><dt></code>	Defines a term in a description list
<code><dd></code>	Describes the term in a description list

HTML Ordered lists

An ordered list starts with the `` tag. Each list item starts with the `` tag.

The list items will be marked with numbers by default:

<pre><!DOCTYPE html> <html> <body> <h2>An ordered HTML list</h2> Coffee Tea Milk </body> </html></pre>	<h3>An ordered HTML list</h3> <ol style="list-style-type: none">1. Coffee2. Tea3. Milk
---	--

Ordered HTML List - The Type Attribute

The `type` attribute of the `` 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:

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

<h2>Ordered List with Numbers</h2>

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

</body>
</html>
```

Ordered List with Numbers

1. Coffee
2. Tea
3. Milk

✓ Uppercase Letters:

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

<h2>Ordered List with Letters</h2>

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

</body>
</html>
```

Ordered List with Letters

- A. Coffee
- B. Tea
- C. Milk

✓ Lowercase Letters:

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

<h2>Ordered List with Lowercase
Letters</h2>

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

</body>
</html>
```

Ordered List with Lowercase Letters

- a. Coffee
- b. Tea
- c. Milk

✓ Uppercase Roman Numbers:

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

<h2>Ordered List with Roman
Numbers</h2>

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

</body>
</html>
```

Ordered List with Roman Numbers

- I. Coffee
- II. Tea
- III. Milk

✓ Lowercase Roman Numbers:

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

<h2>Ordered List with Lowercase Roman
Numbers</h2>

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

</body>
</html>
```

Ordered List with Lowercase Roman Numbers

- i. Coffee
- ii. Tea
- iii. Milk

✓ 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:

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

<h2>The start attribute</h2>
<p>By default, an ordered list will
start counting from 1. Use the start
attribute to start counting from a
specified number:</p>

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

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

</body>
</html>
```

The start attribute

By default, an ordered list will start counting from 1. Use the start attribute to start counting from a specified number:

50. Coffee
51. Tea
52. Milk

L. Coffee
LI. Tea
LII. Milk

Nested HTML Lists

Lists can be nested (list inside list):

```

<!DOCTYPE html>
<html>
<body>

<h2>A Nested List</h2>
<p>Lists can be nested (list inside list):</p>

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

</body>
</html>

```

A Nested List

Lists can be nested (list inside list):

1. Coffee
2. Tea
 1. Black tea
 2. Green tea
3. Milk

Tag	Description
<code></code>	Defines an unordered list
<code></code>	Defines an ordered list
<code></code>	Defines a list item
<code><dl></code>	Defines a description list
<code><dt></code>	Defines a term in a description list
<code><dd></code>	Describes the term in a description list

✓ HTML 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:

```

<dl>
  <dt>Coffee</dt>
  <dd>- black hot drink</dd>
  <dt>Milk</dt>
  <dd>- white cold drink</dd>
</dl>

```

HTML Block and Inline elements

A block level element has a top and a bottom margin, whereas an inline element does not.

```
<!DOCTYPE html>
<html>
<body>
|
<div style="border: 1px solid
black">Hello World</div>

<p>The DIV element is a block element,
and 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>
```

Hello World

The DIV element is a block element, and 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).

✓ Inline Elements

An inline element does not start on a new line.

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

This is a element inside a paragraph.

Example

```
<span>Hello World</span>
```

This is an inline span Hello World element inside a paragraph.

✓ The Element

The `` element is an inline container used to mark up a part of a text, or a part of a document.

The `` element has no required attributes, but `style`, `class` and `id` are common.

When used together with CSS, the `` element can be used to style parts of the 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>
```

The span element

My mother has **blue** eyes and my father has **dark green** eyes.

✓ HTML Footer Element

The `<footer>` element defines a footer for a document or section.

A `<footer>` element typically contains:

- authorship information
- copyright information
- contact information
- sitemap
- back to top links
- related documents

You can have several `<footer>` elements in one document.

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

<footer>
  <p>Author: Hege Refsnes</p>
  <p><a
href="mailto:hege@example.com">hege@exa
mple.com</a></p>
</footer>

</body>
</html>
```

Author: Hege Refsnes
hege@example.com

HTML `<nav>` Element

The `<nav>` element 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 block of navigation links.

Browsers, such as screen readers for disabled users, can use this element to determine whether to omit the initial rendering of this content.

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

<nav>
  <a href="/html/">HTML</a> |
  <a href="/css/">CSS</a> |
  <a href="/js/">JavaScript</a> |
  <a href="/jquery/">jQuery</a>
</nav>

</body>
</html>
```

[HTML](#) | [CSS](#) | [JavaScript](#) | [jQuery](#)

✓ HTML `<aside>` Element

The `<aside>` element defines some content aside from the content it is placed in (like a sidebar).

The `<aside>` content should be indirectly related to the surrounding content.

```

<html>
<head>
<style>
aside {
  width: 30%;
  padding-left: 15px;
  margin-left: 15px;
  float: right;
  font-style: italic;
  background-color: lightgray;
}
</style>
</head>
<body>

<p>My family and I visited The Epcot center this
summer. The weather was nice, and Epcot was
amazing! I had a great summer together with my
family!</p>

<aside>
<p>The Epcot center is a theme park at Walt Disney
World Resort featuring exciting attractions,
international pavilions, award-winning fireworks
and seasonal special events.</p>
</aside>

<p>My family and I visited The Epcot center this
summer. The weather was nice, and Epcot was
amazing! I had a great summer together with my
family!</p>
<p>My family and I visited The Epcot center this
summer. The weather was nice, and Epcot was
amazing! I had a great summer together with my
family!</p>

</body>
</html>

```

My family and I visited The Epcot center this summer. The weather was nice, and Epcot was amazing! I had a great summer together with my family!

My family and I visited The Epcot center this summer. The weather was nice, and Epcot was amazing! I had a great summer together with my family!

My family and I visited The Epcot center this summer. The weather was nice, and Epcot was amazing! I had a great summer together with my family!

The Epcot center is a theme park at Walt Disney World Resort featuring exciting attractions, international pavilions, award-winning fireworks and seasonal special events.

✓ HTML <figure> and <figcaption> Elements

The <figure> tag specifies self-contained content, like illustrations, diagrams, photos, code listings, etc.

The <figcaption> tag defines a caption for a <figure> element.

The <figcaption> element can be placed as the first or as the last child of a <figure> element.

The element defines the actual image/illustration.

```

<!DOCTYPE html>
<html>
<body>

<h2>Places to Visit</h2>

<p>Puglia's most famous sight is the unique conical
houses (Trulli) found in the area around Alberobello,
a declared UNESCO World Heritage Site.</p>

<figure>
  
  <figcaption>Fig.1 - Trulli, Puglia, Italy.
</figcaption>
</figure>

</body>
</html>

```

Places to Visit

Puglia's most famous sight is the unique conical houses (Trulli) found in the area around Alberobello, a declared UNESCO World Heritage Site.



Fig.1 - Trulli, Puglia, Italy.

HTML Symbols

Symbols that are not present on your keyboard can also be added by using entities.

HTML Symbol Entities

Many mathematical, technical, and currency symbols, are not present on a normal keyboard.

To add such symbols to an HTML page, you can use the entity name or the entity number (a decimal or a hexadecimal reference) for the symbol.

```

<!DOCTYPE html>
<html>
<body>

<p>I will display &euro;</p>
<p>I will display &#8364;</p>
<p>I will display &#x20AC;</p>

</body>
</html>

```

I will display €

I will display €

I will display €

Some Mathematical Symbols Supported by HTML

Char	Number	Entity	Description
\forall	∀	∀	FOR ALL
∂	∂	∂	PARTIAL DIFFERENTIAL
\exists	∃	∃	THERE EXISTS
\emptyset	∅	∅	EMPTY SETS
∇	∇	∇	NABLA
\in	∈	∈	ELEMENT OF
\notin	∉	∉	NOT AN ELEMENT OF
\ni	∋	∋	CONTAINS AS MEMBER
\prod	∏	∏	N-ARY PRODUCT
\sum	∑	∑	N-ARY SUMMATION

Some Greek Letters Supported by HTML

Char	Number	Entity	Description
A	Α	Α	GREEK CAPITAL LETTER ALPHA
B	Β	Β	GREEK CAPITAL LETTER BETA
Γ	Γ	Γ	GREEK CAPITAL LETTER GAMMA
Δ	Δ	Δ	GREEK CAPITAL LETTER DELTA
E	Ε	Ε	GREEK CAPITAL LETTER EPSILON
Z	Ζ	Ζ	GREEK CAPITAL LETTER ZETA

Some Other Entities Supported by HTML

Char	Number	Entity	Description
©	©	©	COPYRIGHT SIGN
®	®	®	REGISTERED SIGN
€	€	€	EURO SIGN
™	™	™	TRADEMARK
←	←	←	LEFTWARDS ARROW
↑	↑	↑	UPWARDS ARROW
→	→	→	RIGHTWARDS ARROW
↓	↓	↓	DOWNWARDS ARROW
♠	♠	♠	BLACK SPADE SUIT
♣	♣	♣	BLACK CLUB SUIT
♥	♥	♥	BLACK HEART SUIT
♦	♦	♦	BLACK DIAMOND SUIT

HTML Form Attribute

This chapter describes the different attributes for the HTML `<form>` element.

✓ 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:

```
<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>
```

HTML Forms

First name:

Last name:

If you click the "Submit" button, the form-data will be sent to a page called "/action_page.php".

Submitted Form Data

Your input was received as:

The server has processed your input and returned this answer.

Note: This tutorial will not teach you how servers are processing input. Processing input is explained in our [PHP tutorial](#).

✓ 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.

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

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

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

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

✓ The Autocomplete Attribute

The `autocomplete` attribute specifies whether a form should have autocomplete on or off.

When autocomplete is on, the browser automatically complete values based on values that the user has entered before.

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

<h1>The form autocomplete attribute</h1>

<p>Fill in and submit the form, then reload the
page, start to fill in the form again - and see
how autocomplete works.</p>

<p>Then, try to set autocomplete to "off".</p>

<form action="/action_page.php"
autocomplete="on">
  <label for="fname">First name:</label>
  <input type="text" id="fname" name="fname">
<br><br>
  <label for="email">Email:</label>
  <input type="text" id="email" name="email">
<br><br>
  <input type="submit">
</form>

</body>
</html>
```

The form autocomplete attribute

Fill in and submit the form, then reload the page, start to fill in the form again - and see how autocomplete works.

Then, try to set autocomplete to "off".

First name:

Email:

Submit

✓ The Novalidate Attribute

The `novalidate` attribute is a boolean attribute.

When present, it specifies that the form-data (input) should not be validated when submitted.

Example

A form with a novalidate attribute:

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


List of All <form> Attributes

Attribute	Description
<u>accept-charset</u>	Specifies the character encodings used for form submission
<u>action</u>	Specifies where to send the form-data when a form is submitted
<u>autocomplete</u>	Specifies whether a form should have autocomplete on or off
<u>enctype</u>	Specifies how the form-data should be encoded when submitting it to the server (only for method="post")
<u>method</u>	Specifies the HTTP method to use when sending form-data
<u>name</u>	Specifies the name of the form
<u>novalidate</u>	Specifies that the form should not be validated when submitted
<u>rel</u>	Specifies the relationship between a linked resource and the current document
<u>target</u>	Specifies where to display the response that is received after submitting the form

List of All <form> Attributes

Attribute	Description
<u>accept-charset</u>	Specifies the character encodings used for form submission
<u>action</u>	Specifies where to send the form-data when a form is submitted
<u>autocomplete</u>	Specifies whether a form should have autocomplete on or off
<u>enctype</u>	Specifies how the form-data should be encoded when submitting it to the server (only for method="post")
<u>method</u>	Specifies the HTTP method to use when sending form-data
<u>name</u>	Specifies the name of the form
<u>novalidate</u>	Specifies that the form should not be validated when submitted
<u>rel</u>	Specifies the relationship between a linked resource and the current document
<u>target</u>	Specifies where to display the response that is received after submitting the form

HTML Form elements

✓ The HTML `<form>` Elements

The HTML `<form>` element can contain one or more of the following form elements:

- `<input>`
- `<label>`
- `<select>`
- `<textarea>`
- `<button>`
- `<fieldset>`
- `<legend>`
- `<datalist>`
- `<output>`
- `<option>`
- `<optgroup>`

✓ The `<input>` Element

One of the most used form element is the `<input>` element.

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

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

<h2>The input Element</h2>

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

</body>
</html>
```

The input Element

First name:

✓ The <label> Element

The <label> element defines a label for several form elements.

The <label> element is useful for screen-reader users, because the screen-reader will read out loud the label when the user focus on the input element.

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

The <select> Element

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

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

<h2>The select Element</h2>

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

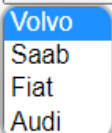
<form action="/action_page.php">
  <label for="cars">Choose a car:</label>
  <select id="cars" name="cars">
    <option value="volvo">Volvo</option>
    <option value="saab">Saab</option>
    <option value="fiat">Fiat</option>
    <option value="audi">Audi</option>
  </select>
  <input type="submit">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

</body>
</html>
```

The select Element

The select element defines a drop-down list:

Choose a car:



✓ 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>
```

Visible Option Values

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

Choose a car:

✓ 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>
```

✓ 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>
```

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:


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

<h2>The button Element</h2>

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

</body>
</html>
```

The button Element



✓ **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.

```
<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>
```

Grouping Form Data with Fieldset

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

Personalia:

First name: <input type="text" value="John"/>
Last name: <input type="text" value="Doe"/>
<input type="submit" value="Submit"/>

✓ The <datalist> Element

The <datalist> element specifies a list of pre-defined options for an <input> element.

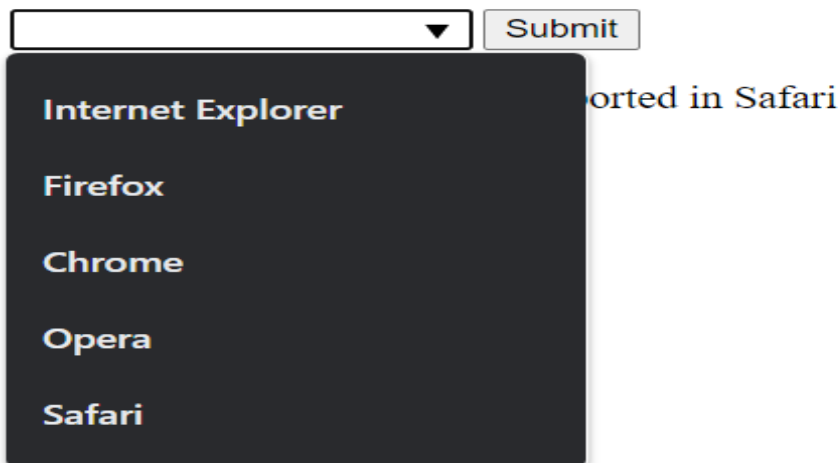
Users will see a drop-down list of the pre-defined options as they input data.

The `list` attribute of the <input> element, must refer to the `id` attribute of the <datalist> element.

```
<form action="/action_page.php">
  <input list="browsers">
  <datalist id="browsers">
    <option value="Internet Explorer">
    <option value="Firefox">
    <option value="Chrome">
    <option value="Opera">
    <option value="Safari">
  </datalist>
</form>
```

The datalist Element

The `datalist` element specifies a list of pre-defined options for an input element.



A screenshot of a web form. At the top, there is a dropdown menu with a downward arrow and a "Submit" button to its right. Below the dropdown menu, a dark grey menu is open, listing five browser names: "Internet Explorer", "Firefox", "Chrome", "Opera", and "Safari". To the right of the menu, the text "orted in Safari" is partially visible.

✓ The `<output>` Element

The `<output>` element represents the result of a calculation (like one performed by a script).

Example

Perform a calculation and show the result in an `<output>` element:

```
<form action="/action_page.php"
  oninput="x.value=parseInt(a.value)+parseInt(b.value)">
  0
  <input type="range" id="a" name="a" value="50">
  100 +
  <input type="number" id="b" name="b" value="50">
  =
  <output name="x" for="a b"></output>
  <br><br>
  <input type="submit">
</form>
```

The output Element

The output element represents the result of a calculation.

0  100 +
50 = 140

Submit

Note: The output element is not supported in Edge prior version 13.

HTML Video

The HTML `<video>` element is used to show a video on a web page.

To show a video in HTML, use the `<video>` element:

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>
```

How it Works

The `controls` attribute adds video controls, like play, pause, and volume.

It is a good idea to always include `width` and `height` attributes. If height and width are not set, the page might flicker while the video loads.

The `<source>` element allows you to specify alternative video files which the browser may choose from. The browser will use the first recognized format.

The text between the `<video>` and `</video>` tags will only be displayed in browsers that do not support the `<video>` element.

✓ HTML `<video>` Autoplay

To start a video automatically, use the `autoplay` attribute:

Example

```
<video width="320" height="240" autoplay>  
  <source src="movie.mp4" type="video/mp4">
```



```
<source src="movie.ogg" type="video/ogg">
Your browser does not support the video tag.
</video>
```

Add **muted** after **autoplay** to let your video start playing automatically (but muted):

Example

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

HTML AUDIO

To play an audio file in HTML, use the `<audio>` element:

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 Audio - How It Works

The `controls` attribute adds audio controls, like play, pause, and volume.

The `<source>` element allows you to specify alternative audio files which the browser may choose from. The browser will use the first recognized format.

The text between the `<audio>` and `</audio>` tags will only be displayed in browsers that do not support the `<audio>` element.

✓ HTML `<audio>` Autoplay

To start an audio file automatically, use the `autoplay` attribute:

Example

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

Note: Chromium browsers do not allow autoplay in most cases. However, muted autoplay is always allowed.

Add `muted` after `autoplay` to let your audio file start playing automatically (but muted):

Example

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

[HTML YOUTUBE VIDEOS](#)

✓ YouTube Video Id

YouTube will display an id (like `tgbNymZ7vqY`), when you save (or play) a video.

You can use this id, and refer to your video in the HTML code.

✓ Playing a YouTube Video in HTML

To play your video on a web page, do the following:

- Upload the video to YouTube
- Take a note of the video id
- Define an `<iframe>` element in your web page
- Let the `src` attribute point to the video URL
- Use the `width` and `height` attributes to specify the dimension of the player
- Add any other parameters to the URL (see below)

Example

```
<iframe width="420" height="315"
src="https://www.youtube.com/embed/tgbNymZ7vqY">
</iframe>
```

✓ YouTube Autoplay + Mute

You can let your video start playing automatically when a user visits the page, by adding `autoplay=1` to the YouTube URL. However, automatically starting a video is annoying for your visitors!

Note: Chromium browsers do not allow autoplay in most cases. However, muted autoplay is always allowed.

Add `mute=1` after `autoplay=1` to let your video start playing automatically (but muted).

✓ YouTube - Autoplay + Muted

```
<iframe width="420" height="315"
src="https://www.youtube.com/embed/tgbNymZ7vqY?autoplay=1&mute=1" >
</iframe>
```

✓ YouTube Playlist

A comma separated list of videos to play (in addition to the original URL).

✓ YouTube Loop

Add `loop=1` to let your video loop forever.

Value 0 (default): The video will play only once.

Value 1: The video will loop (forever).

```
<iframe width="420" height="315"
src="https://www.youtube.com/embed/tgbNymZ7vqY?playlist=tgbNymZ7vqY&loop=1" >
</iframe>
```

✓ YouTube Controls

Add `controls=0` to not display controls in the video player.

Value 0: Player controls does not display.

Value 1 (default): Player controls display.

```
<iframe width="420" height="315"
src="https://www.youtube.com/embed/tgbNymZ7vqY?controls=0" >
</iframe>
```

