



[Learn
Catalog](#)



[Codecademy Articles](#) > web

HTML Glossary

Programming reference for HTML elements



Attributes

class

HTML elements can have one or more classes, separated by spaces. You can style elements using CSS by selecting them with their classes.

Example

```
<div class="big-box yellow-box">This is a big yellow box.</div>
```

id

An HTML element can have an id attribute to identify it. id elements should always be unique to that single element, and each element should never have more than one id.

Example

```
<div id="my-box">This is my box! Put your text in some other box.</div>
```

href

Links tell the browser where to go using an href attribute, which stores a URL.

Example

```
<a href="http://google.com">Google it!</a>
```

Basic Formatting

You can easily format text to be bold, italic, or underlined using simple formatting tags.

Example

This text is **bold**, *italicized*, and underlined.

Body

The body is the container for all of a page's content. Comes after the `<head>` tag, within the overall `<html>` tag.

Example

```
<html>
  <head>
    <title>An example of the body tag</title>
  </head>
  <body>
    This is inside the body!
  </body>
</html>
```

Read more

- <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/body>

Usage

Almost all content belongs inside the body tag. The main exceptions are script and style tags, as well as the page title tag. As you can see in this example, there is a heading, an image, and a link all inside the body tag. The head tag contains only external files and the page title.

Example

```
<html>
  <head>
    <title>My homepage</title>
    <link rel="stylesheet" type="text/css" href="homepage.css" />
    <script type="text/javascript" src="homepage.js"></script>
  </head>
  <body>
    <h1>Hello, this is a picture of my cat!</h1>
    
    <a href="mailto:cat@codecademy.com">Email my cat</a>
  </body>
</html>
```

Children

An element that is an immediate descendent of another element or nested within another element is called a child. These become useful when using CSS child selectors and psuedo-elements.

Example

```
<ul id="parent">
  <li id="child">I'm a child of parent!</li>
</ul>
```

Comments

HTML comments are sometimes used in code to explain parts of the markup. They are similar to comments in other languages. Users do not see comments in their browser.

Syntax

```
<!-- This is an HTML comment! -->
```

Div

A block level container (or 'division' of the web page) for content with no semantic meaning.

Syntax

```
<div>This is a div element.</div>
```

Head

Tag that surrounds important content that is invisible to the user, but is important to the browser. Elements within this tag contain metadata about the page and links to stylesheets, scripts, etc.

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

Headings

Heading elements like <h1>, <h2>, <h3>, ... allow you to use six levels of document headings, ranging from largest to smallest, breaking up the document into logical sections. For example, the word 'Headings' above is wrapped in a <h2> tag.

Syntax

```
<h1> This is a header! </h1>
```

Horizontal rules

This tag creates a black line one pixel thick that runs the all the way across its container. It can be styled to look differently with CSS.

Example

```
This text is divided
<hr>
...from this text!
```

Read more

- <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/hr>

HTML

What is HTML?

HTML stands for Hyper Text Markup Language. It is the language used to create all websites.

Read more

- http://www.w3.org/wiki/HTML/Training/What_is_HTML

<html> tag

All HTML files live within an over-arching html tag. This is the basic tag that defines an html document.

Syntax

```
<html>
  The rest of your web page goes in here!
</html>
```

Read more

- <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/html>

Hyperlinks

Hyperlinks (or just links) take the user to another webpage when they click on it. The most common attribute used with links is href, which tells the browser where the link goes.

Syntax

```
<a href="url this link goes to">Link text</a>
```

Example

The following text is `goes to Google`.

Images

The `img` tag embeds an image into your HTML. Always found with the `'src'` attribute, which tells the browser where to find the image. Note that the `` tag is self-closing.

Syntax

```
<img src='mylocalimage.jpg' />
```

Line breaks

This tag is used in a block of text to force a line break. This is to be used for things which are a single paragraph, but where this formatting is necessary such as poems or addresses. To separate paragraphs, separate each paragraph into a separate element instead. The resulting element on a web page will look like:

Example

```
<p> Some text <br/> that spans two lines </p>
```

Links

Link elements are used to connect your document to a related resource (very different from hyperlinks, which take you to another webpage when you click on them). Links appear only in the head section of a document so they do not alter the content, but only the presentation. Links are most commonly used to connect to a stylesheet, script, favicon, or alternate format of the page such as an RSS feed or PDF.

Exempl

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

Lists

HTML supports two kinds of lists: ordered lists and unordered lists. Within lists each individual list item has its own tag.

Unordered Lists

Unordered lists are just lists whose items are denoted with bullet points.

Example

Shopping list

```
<ul>
  <li>Dish soap</li>
  <li>Kitty litter</li>
```

```
<li>Tomato sauce</li>
</ul>
```

Read more

- <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/ol>

Ordered Lists

Ordered lists' items are denoted with numbers.

Example

My numbered list

```
<ol>
  <li>First item!</li>
  <li>Second item!</li>
  <li>Last item!</li>
</ol>
```

Read more

- <https://developer.mozilla.org/en-US/docs/Web/HTML/Element/li>

Paragraphs

<p>

One of the most common tags in HTML - it denotes a paragraph of text. It often has other elements nested inside of it, such as , <a>, and .

Syntax

```
<p>This is paragraph text!</p>
```

Semantic formatting

These tags are similar to the previously mentioned formatting tags which have fallen out of favor. The difference is that these tags have semantic value (meaning). is used for something that you wish to emphasize and is used for something that is important. With both of these elements, you can convey the level of emphasis or importance with nesting. The more times that you nest the element within itself, the higher the magnitude of the text it contains.

Example

```
<p><strong><strong>Warning:</strong>Acid can cause severe burns</strong> </p>
```

Tables

An element for displaying information in rows and columns. Supports headers and footers for labeling columns. Divides information into rows (denoted by the `tr` tag) which contain cells (denoted by the `td` tag).

Example

```
<table>
  <thead>
    <tr>
      <th>Item</th>
      <th>Price</th>
    </tr>
  </thead>

  <tbody>
    <tr>
      <td>Banana</td>
      <td>$56.75</td>
    </tr>
    <tr>
      <td>Yogurt</td>
      <td>$12.99</td>
    </tr>
  </tbody>

  <tfoot>
    <tr>
      <td>Total</td>
      <td>$69.74</td>
    </tr>
  </tfoot>
</table>
```

Tags & Elements

Tags are basic labels that define and separate parts of your markup into elements. They are comprised of a keyword surrounded by angle brackets `<>`. Content goes between two tags and the closing one is prefixed with a slash (Note: there are some self-closing HTML tags, like image tags). Tags also have attributes, which are

Syntax

```
<tag attribute='value'>content</tag keyword>
```

Title

This tag tells the browser what to display as the page title at the top and tells search engines what the title of your site is. It goes inside `<head>` tags. Try and make your page titles descriptive, but not overly verbose.

Example

<title> HTML Glossary </title>



Teaching the world how to code.

Codecademy

- [About](#)
- [Stories](#)
- [We're Hiring](#)
- [For Business](#)
- [For Education](#)

Resources

- [Articles](#)
- [Forums](#)
- [Help](#)
- [Blog](#)

Learn To Code

- [Make a Website](#)
- [Make an Interactive Website](#)
- [Learn Sass](#)
- [Deploy a Website](#)
- [Learn JavaScript](#)
- [Learn Rails](#)
- [Learn AngularJS](#)
- [Learn ReactJS: Part I](#)
- [Learn ReactJS: Part II](#)
- [Ruby on Rails Authentication](#)
- [Learn the Command Line](#)
- [Learn Git](#)
- [Learn SQL](#)
- [SQL: Table Transformation](#)
- [SQL: Analyzing Business Metrics](#)
- [Learn Java](#)
- [Learn HTML & CSS](#)

- [HTML & CSS](#)
- [JavaScript](#)
- [jQuery](#)

- [PHP](#)
- [Python](#)
- [Ruby](#)
- [Learn APIs](#)

[Privacy Policy](#) [Terms](#)

Made in NYC © 2017 Codecademy

English

