# **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 <b>bold</b>, <i>italicized</i>, and <u>underlined</u>.

# **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>

<img src="cat.jpg" />

<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 <a href="http://google.com">goes to Google</a>.

# **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 <img/> 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.

**Exampl**

<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 <img/>, <a>, <strong> and <em>.

**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).<em> is used for something that you wish to emphasize and <strong> 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>

# **HTML5 Features**

Web languages need regular upgrades in order to stay current and solve new problems faced by web developers. HTML5 is the latest version of HTML. Below are some HTML5 features you will encounter as you learn with Codecademy.

## video

The video element allows you to easily stream video from a website.

<video width="450px" height="350px" controls>

<source src="video-url.mp4" type="video/mp4">

</video>

In the HTML above, width and height set the dimensions for the video element. The controls attribute creates playback buttons such as "Play" and "Pause". The source src tag provides the URL where the video is hosted and type specifies the video's type, in this case, "video/mp4".

## figure

Figure elements can be used to display visual content, such as photos, illustrations, diagrams or code snippets.

<figure class="gallery-item">

<img src="image-1.png">

</figure>

<figure class="gallery-item">

<img src="image-2.png">

</figure>

In the example code above, figure elements have the class "gallery-item", and each contains an img element.

## section

Section elements, like divs, can be used to organize webpage content into thematic groups.

<section class="contact-form">

<h2>Contact Us</h2>

<form>

...

</form>

</section>

Above, a section element is used to organize h2 and form elements for a website's "Contact Us" feature.

## nav

The nav element is used for the part of a website that links to other pages on the site. The links can be organized a number of ways. Below, the links are displayed within paragraph elements. An unordered list could also be used.

<nav>

<p><a href="login.html">Log In</a></p>

<p><a href="signup.html">Sign Up</a></p>

<p><a href="contact.html">Contact Us</a></p>

</nav>

## header

The header element can be used to group together introductory elements on a website, such as a company logo, navigation items, and sometimes, a search form.

<header>

<img src="company-logo.png">

<nav>

<p><a href="login.html">Log In</a></p>

<p><a href="signup.html">Sign Up</a></p>

<p><a href="contact.html">Contact Us</a></p>

</nav>

</header>

Above, the header element encloses the img and nav.

## footer

The footer element is typically found at the bottom or foot of a webpage. It can contain copyright information, links to social media and additional site navigation items.

<footer>

<p>&copy; Acme Granola Corporation 2016<p>

<div class="social">

<a href="#"><img src="instagram-icon.png"></a>

<a href="#"><img src="facebook-icon.png"></a>

<a href="#"><img src="twitter-icon.png"></a>

</div>

</footer>

Above, between <footer> and </footer>, copyright information is contained in the p element, and social media links are contained within the div with class "social".

# **CSS Glossary**

Programming reference for CSS covering Comments, Properties, and Selectors

# **Comments**

Comments in CSS are signified by a forward-slash and asterisk.

**Example**

/\* This is a single line comment \*/

**Example**

/\* This

is a multi-line

comment \*/

# **Properties**

## Definition

Properties are defined within selectors by defining a property and a value. They are separated with a colon and delineated with a semi-colon.

**Syntax**

selector {

property: value;

}

**Example**

h1 {

color: blue;

}

**Read more**

* <http://www.htmldog.com/reference/cssproperties/>

## Defining many properties

Each CSS rule can have as many properties as you like. Each of them applies to the elements that the selector applies to.

**Example**

h1 {

font-size: 24px;

font-weight: bold;

border: 1px solid black;

color: pink;

}

/\* This will make all <h1> headers big, bold, pink, and inside of a thin black rectangle! \*/

## Padding

The padding is the spacing between the content and the border (edge of the element.). We can adjust this value with CSS to move the border closer to or farther from the content. Here, the div with id 'box' will get 10px of padding all around it.

**Example**

#box {

padding: 10px;

}

## Margin

The margin is the space around the element. The larger the margin, the more space between our element and the elements around it. We can adjust the margin to move our HTML elements closer to or farther from each other. Here, the div with id 'box' will get 10px of margin above and below it, and 5px of margin to the left and right.

**Example**

#box {

margin: 10px 5px 10px 5px;

}

## font-family

The font-family property sets the font of an HTML element's text.

**Syntax**

p {

font-family: Arial, Helvetica, sans-serif;

}

# **Selectors**

What are selectors?

Selectors are used in CSS to select the parts of the HTML that are being styled. You can use several different methods for selecting an element.

**Syntax**

selector {

rules;

rules;

rules;

}

**Read more**

* <https://developer.mozilla.org/en-US/docs/Web/Guide/CSS/Getting_started/>

## Class name selectors

You can also select HTML elements by their Class name. Unlike ID selectors, Class selectors select all elements with a matching class.

**Example**

a.link {

font-size: 12px;

}

/\* HTML Selected: <a href="http://google.com" class="link">,

<a href="http://codecademy.com" class="link jumbo"> \*/

**Example**

.jumbo {

text-size: 1000px;

}

/\* HTML Selected: <a href="http://codecademy.com" class="link jumbo">,

<span class="jumbo"> \*/

## Element selectors

You are able to select HTML elements first by simply using the name of the element.

**Example**

body {

background-color: #333;

}

**Example**

h1 {

color: blue;

}

**Example**

a {

text-underline: none;

}

## ID selectors

ID selectors are used to select only a single item on a page. Like the term ("identification") indicates, ID selectors will ONLY select the first element with a matching ID.

**Example**

#thatThingINeededToStyle {

color: blue;

font-size: 24px;

}

/\* HTML Selected: <span id="thatThingINeededToStyle"> \*/

**Example**

a#codecademy {

color: purple;

}

/\* HTML Selected: <a href="http://codecademy.com" id="codecademy"> \*/

## Attribute selectors

HTML elements are also able to be selected by their attributes.

**Example**

a[href="http://codecademy.com"] {

color: purple;

}

/\* HTML Selected: <a href="http://codecademy.com"> \*/

**Example**

input[type="text"] {

width: 100px;

}

/\* HTML Selected: <input type="text"> \*/

**Example**

input[required] {

border: 1px red solid;

}

/\* HTML Selected: <input type="text" required> \*/

**Read more**

* <http://dev.opera.com/articles/view/27-css-basics/#attribute>

## Child selectors

You can also use multiple selectors to get the exact elements you want, by using parental nesting. By using the "greater-than" symbol (>), you can select only the direct children of an element, going down only one level.

**Example**

ul > li {

display: inline-block

}

/\* Selects only the first-level list items in all unordered lists in the HTML \*/

**Example**

ul a {

text-underline: none;

}

/\* Selects all anchors which have an unordered list their ancestry \*/

**Example**

ul + span {

display: inline;

}

/\* Selects only spans that directly follow an unordered list \*/

**Example**

a ~ h1 {

color: blue;

}

/\* Selects all h1 elements that are in the general vicinity of an anchor \*/

**Read more**

* <https://developer.mozilla.org/en-US/docs/Web/CSS/Descendant_selectors>

## Universal selector

The universal selector (\*) may be used to select all the elements in a particular range. Be aware that the universal selector is the most performance taxing selector, and should be used sparingly.

**Example**

\* {

background-color: blue;

}

/\* Selects ALL HTML elements in the page \*/

**Example**

body \* {

color: red;

}

/\* Selects ALL children of the body \*/

**Example**

div > \* {

color: red;

}

/\* Selects ALL first-level children of all divs on the page \*/

**Read more**

* <https://developer.mozilla.org/en-US/docs/Web/CSS/Universal_selectors>
* <http://www.stevesouders.com/blog/2009/06/18/simplifying-css-selectors/>
* <http://dev.opera.com/articles/view/27-css-basics/#universal>

## Pseudo class selectors

Pseudo Selectors can be used to narrow down a selection with certain rules.

**Example**

li:first-child {

color: red;

}

/\*

This selects only <li> elements that have no elements before them

<ul>

<li>Selected; will be red</li>

<li>Not selected</li>

<li>Not selected</li>

</ul>

\*/

li:last-child {

color: red;

}

/\* This does the opposite; only the last <li> will be red. \*/

**Example**

a:hover {

text-decoration: underline;

}

/\* Will underline all links when the user puts their mouse over them \*/

a:active {

font-weight: bold;

}

/\* Will make all links bold while the user is clicking on them. \*/

**Read more**

* <https://developer.mozilla.org/en-US/docs/Web/CSS/Pseudo-classes>
* <http://dev.opera.com/articles/view/27-css-basics/#pseudoclasses>

# **CSS3 Features**

Overview of CSS3 features used in the Make a Website course.

# **CSS3 Features**

CSS3 is the latest version of CSS and contains a number of exciting, new features that make it easier for web developers to create great styling for websites.

## rem values

You will encounter rem values as you learn about the CSS font-size property, and other CSS properties that specify element size. In the case of the font-size property, a rem value displays a font-size relative to the font-size of the root element, called simply html.

For example, consider the following CSS:

html {

font-size: 20px;

}

p {

font-size: 0.75rem;

}

In the code, font-size is set to 20px for the html selector. The p selector has a font-size of 0.75rem, which displays a size 2/3 that of html, or 15px.

## Flexbox

The CSS3 flexbox feature makes it much easier for web developers to arrange HTML elements vertically or horizontally. Website layouts designed with flexbox can respond to users with various screen widths, including mobile devices. To access the feature in CSS, the display property must be set to flex, as seen below:

.main {

display: flex;

}

HTML elements that are children of an element with the "main" class are now flex items and can be arranged using flexbox properties. Here two such such properties:

* flex-wrap: arranges flex items into a single line or wraps them across multiple lines of a webpage layout.
* justify-content: can orient flex items in a number of different positions on a webpage, including the center.

Interested in learning more about flexbox? [Here is a great supplemental resource](https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Flexible_Box_Layout/Using_CSS_flexible_boxes).

## background-image

The CSS3 background-image property is used to set a background image for an HTML element.

For example, consider the following CSS:

body {

background-image: url("http://image-gallery.io/mountain-scene.png");

}

## background-size

The CSS background-size property controls the size of an HTML element's background image. Used along with the background-image property, background-size values control how a background image is proportioned and scaled. One such value is cover as seen in the following CSS:

body {

background-image: url("http://image-gallery.io/mountain-scene.png");

background-size: cover;

}

Here, the image covers the entire HTML body element.