HTML:

There are six heading elements — <h1>, <h2>, <h3>, <h4>, <h5>, and <h6>. Each element represents a different level of content in the document; <h1> represents the main heading, <h2> represents subheadings, <h3> represents sub-subheadings, and so on

Unordered lists are used to mark up lists of items for which the order of the items doesn't matter:

<ul>

<li>milk</li>

<li>eggs</li>

<li>bread</li>

<li>hummus</li>

</ul>

Ordered List:

<ol>

<li>Drive to the end of the road</li>

<li>Turn right</li>

<li>Go straight across the first two roundabouts</li>

<li>Turn left at the third roundabout</li>

<li>The school is on your right, 300 meters up the road</li>

</ol>

Markups:

<em>this will be italic</em>

<p>I am <em>glad</em> you weren't <em>late</em>.</p>

<Strong></Strong> and <b></b> does the same job. Bold.

* [<i>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/i) is used to convey a meaning traditionally conveyed by italic: Foreign words, taxonomic designation, technical terms, a thought...
* [<b>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/b) is used to convey a meaning traditionally conveyed by bold: Key words, product names, lead sentence...
* [<u>](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/u) is used to convey a meaning traditionally conveyed by underline: Proper name, misspelling...

<p>I'm creating a link to

<a href="https://www.mozilla.org/en-US/">the Mozilla homepage</a>.

</p>

<a href="https://www.mozilla.org/en-US/">

<img src="mozilla-image.png" alt="mozilla logo that links to the mozilla homepage">

</a>

You can turn any block into a link! Just add <a></a>

<a href = “subdirectory/name.html”> </a>OR <a href = “../parentdirname.html”></a>

Hash:

A hash - # within a hyperlink specifies an html element id to which the window should be scrolled.

href="#some-id" would scroll to an element on the **current page** such as <div id="some-id">.

href="//site.com/#some-id" would go to site.com and scroll to the id on that page.

Scroll to Top:

href="#" doesn't specify an id name, but does have a corresponding location - the top of the page. Clicking an anchor with href="#" will move the scroll position to the top.

Document Fragments:

<h2 id="Mailing\_address">Mailing address</h2>

<p>Want to write us a letter? Use our <a href="contacts.html#Mailing\_address">mailing address</a>.</p>

Span tag inline style:

<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>

HTML class: global variable:

*<element* class="*classname*">

<br> 🡪 new line in HTML

var a = eval("x \* y") + "<br>"; //the <br> tag has no end tag

HTML DOM : Document Object Model: everything is a node;

DOM Events: mouse movements, clicks, etc;

The HTML5 <canvas> tag is used to draw graphics, on the fly, with JavaScript.

<input type="text">

In HTML, the characters <, >,",' and & are special characters.

| **Literal character** | **Character reference equivalent** |
| --- | --- |
| < | &lt; |
| > | &gt; |
| " | &quot; |
| ' | &apos; |
| & | &amp; |

special markers <!-- and --> are comments

CSS

Float: left/right

Clear: left/ right/both

JS 🡪 JSON.parse() is safer than eval(), it throws an exception if the text contain anything dangerous;

var obj = JSON.parse('{ "name":"John", "age":30, "city":"New York"}');

document.getElementById("demo").innerHTML = obj.name + ", " + obj.age;

CSS selector:

Select by id 🡪 #idname{}

Select by element 🡪 .p { }

Select by class 🡪 .classname{}

<style>

p.center {

text-align: center;

color: red;

}

p.large {

font-size: 200%;

}

</style>

</head>

<body>

<h1 class="center">This heading will not be affected</h1>

<p class="center">This paragraph will be red and center-aligned.</p>

<p class="center large">This paragraph will be red, center-aligned, and in a large font-size.</p>

</body>

</html>

With CSS, a color is most often specified by:

* a valid color name - like "red"
* a HEX value - like "#ff0000"
* an RGB value - like "rgb(255,0,0)"

Border Style

The border-style property specifies what kind of border to display.

The following values are allowed:

* dotted - Defines a dotted border
* dashed - Defines a dashed border
* solid - Defines a solid border
* double - Defines a double border
* groove - Defines a 3D grooved border. The effect depends on the border-color value
* ridge - Defines a 3D ridged border. The effect depends on the border-color value
* inset - Defines a 3D inset border. The effect depends on the border-color value
* outset - Defines a 3D outset border. The effect depends on the border-color value
* none - Defines no border
* hidden - Defines a hidden border

The border-style property can have from one to four values (for the top border, right border, bottom border, and the left border).

If the margin property has two values:

* **margin: 25px 50px;**
  + top and bottom margins are 25px
  + right and left margins are 50px

**margin: 25px 50px 75px 100px;**

* top margin is 25px
* right margin is 50px
* bottom margin is 75px
* left margin is 100px

You can set the margin property to auto to horizontally center the element within its container.

## CSS Padding

The CSS padding properties are used to generate space around an element's content, inside of any defined borders.

* padding-top
* padding-right
* padding-bottom
* padding-left

**padding: 25px 50px 75px 100px;**

* top padding is 25px
* right padding is 50px
* bottom padding is 75px
* left padding is 100px

The total width of an element should be calculated like this:

* Total element width = width + left padding + right padding + left border + right border + left margin + right margin

Use the <i> element only when there is not a more appropriate semantic element, such as:

* [<em>](https://www.w3schools.com/tags/tag_em.asp) (emphasized text)
* [<strong>](https://www.w3schools.com/tags/tag_strong.asp) (important text)
* [<mark>](https://www.w3schools.com/tags/tag_mark.asp) (marked/highlighted text)
* [<cite>](https://www.w3schools.com/tags/tag_cite.asp) (the title of a work)
* [<dfn>](https://www.w3schools.com/tags/tag_dfn.asp) (a definition term)

jQuery Syntax

The jQuery syntax is tailor-made for **selecting** HTML elements and performing some **action** on the element(s).

Basic syntax is: **$(*selector*).*action*()**

* A $ sign to define/access jQuery
* A (*selector*) to "query (or find)" HTML elements
* A jQuery *action*() to be performed on the element(s)

Examples:

$(this).hide() - hides the current element.

$("p").hide() - hides all <p> elements.

$(".test").hide() - hides all elements with class="test".

$("#test").hide() - hides the element with id="test".

$(document).ready(function(){  
  
   *// jQuery methods go here...this is to prevent jQuery running before document is loaded*  
  
});

When our website has a lot of pages and I want to reuse my jQuery function, put my function in a separate file.

<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js">  
</script>  
<script src="my\_jquery\_functions.js"></script>  
</head>



Above is jQuery Dimensions