

List of CSS Declarations (Properties & Values)

This list is obviously not every single CSS declaration possible, but it does contain the ones you will most likely be using. I have also left out some of the possible values the properties will accept in the name of simplifying.

For Sizing:

Properties	Accepted Values
width: defines width of an element	pixel value or percentage
height: defines height of an element	pixel value, percentage, or auto (makes div expand as much as needed based on content)
font-size: defines size of text	pixel value, small, medium, large, x-large

For Placement:

margin: applies a margin amount pushing off the element	pixel value, percentage amount of containing element, auto
padding: applies padding to the inside of an element	pixel value, percentage amount of containing element
float: moves an element to the left or right (never up or down), breaks normal stack order, other content (such as text) can wrap around floated item	left, right
clear: stops elements after floated elements from continuing to stack next to them- reverts the stack order back to normal	left, right, both
text-align: specifies the horizontal alignment of text inside an element	left, right, center, justify
display: used to turn in-line elements into block elements (like img) or make something display horizontal when the default is vertical (like a list)	inline, block, inline-block

- A definition of “auto” on both the left and right margin of an element will center it on the page; this is most often done on the wrapper/container div
- To define equal margin or padding on all sides of an element you merely need to write one number:
 - margin: 20px;
 - padding: 10px;

- To define different amounts on each side, define the numbers in a clockwise fashion from top, right, bottom, left
 - margin: 0px 15px 15px 0px; (make sure not to use commas)
- Images are inline elements. We often turn them into blocks by using display: inline-block so we can do things like center them. Images also come with annoying borders, so I usually add border: 0; to my image css

For Aesthetics:

font-family: uses a “fallback” system where if the computer does not have a listed font installed, the next one in the list will be used	font names separated by commas: “Times New Roman”, Georgia, Arial; Multiple word fonts, like TNR above, need to be in quotation marks for some browsers End list with serif or sans-serif
color: defines the color of text inside an element	color name or hexadecimal value for more precise color choices
background-color: fills the element with a color	color name or hexadecimal value for more precise color choices
line-height: controls page leading	percentage value of current space
list-style-type: defines the look of bullets in a list	circle, square, none
font-weight: defines look of font	normal, bold, bolder, lighter
letter-spacing: adds kerning (space between letters)	pixel value
border-style: defines the style of border around an element	dotted, dashed, solid, double, groove
border-color: defines the color of the border	color name or hexadecimal value for more precise color choices
border-width: defines the thickness of the border	thin, medium, thick, or pixel value
border-radius: rounds the edges of a div	pixel amounts written with same rules as border/padding/margin (4 different amounts or all 4 at once)

- Border examples above will place a border on all sides. Declarations can be written for individual side properties too
 - border-top-style
 - border-right-width
- You can also define an all sides border with one declaration:

- border: thin solid black; (must be done in this order: weight, type, color)
- Though thin, medium, and thick work for weight values, on a fixed (not fluid) site, it's best to use an exact pixel amount—border: 2px solid black;

For Links:

The 3 elements you will want to define are

- a:link- defines a hyperlink
- a:visited- defines what a link should look like after it's clicked
- a:hover- defines what a link should look like upon a mouse hover

You may also define an “active state”

- a:active- defines what a link should look like when it's clicked and that page is active

In your CSS, **you need to define them in the order I have above** or they may not work properly.

color: change the color of the link	color name or hexadecimal value for more precise color choices
text-decoration: defines formatting of the link	underline (this is default), none
font-size: defines size of link font	pixel value, small, medium, large, x-large
background-color: if links have been displayed as blocks, will fill link in with chosen color. Very useful for hover effects.	color name or hexadecimal value for more precise color choices
font-style: defines style of link font	normal, italic, oblique

Remember, sometimes you will want similar elements to do the same thing. There is no need to write out a CSS declaration for each one. Use the comma:

```
a:link, a:visited {
    color: blue;
    text-decoration: none;
}
```

Now both the link itself and its visited state will have the same values.