Less and Sass

Less and Sass are shorthand ways to write CSS and they make life easier for developers.

You can create variables in Less and Sass and then call them in your code.

For example you could create a color variable and then use it for the copy on the site.

// VARIABLE:

@myBlack: #222;

// SASS

body {

color: @myBlack;

}

Variables really help if you need to change your styles.

Let’s look at the example we just created, @myBlack.

If you need to go back and make that black a little lighter, you change the definition of the variable.

// VARIABLE:

@myBlack: #333;

Doing this will change the definition of @myBlack to #333 every time it is called.

In regular CSS you would have to sift through your stylesheet, find each instance where #222 is used and switch it to #333. Which is a tedious and painful process.

You can also nest your styles similar to how you write HTML.

// NESTING

.heroSection {

color@myBlack;

p {

font-size: 20px;

font-weight: bold;

}

}

This allows you control parent and child elements more easily.

It also makes code cleaner and easier to read.

There are some differences in syntax for Less and Sass but overall they have the same principles.

For instance you define variable in Less with @ and in Sass you use $.

Different frameworks have either Less and Sass baked in to them.

Bootstrap uses Less, but you can download a template of Bootstrap using Sass via Compass.

Foundation uses Sass.

Here are links to the Less and Sass sites:

http://lesscss.org/

http://sass-lang.com/