8 Understanding How to Use SCSS Functions

<https://sass-lang.com/documentation/at-rules/function/#arguments>

function Name

invert is the name of the function, but you can name it anything you want, as long as it is descriptive of what you are trying to do.

3 official Functions in Body

You will notice that in the body of the function block, you are using 3 different official SCSS functions.

* change-color
* hue
* mix

The Expression

change-color and hue are being used in an expression. We are naming this color manipulation code to be $inverse. Take note that the variable of $inverse is not the same as our function name which is invert.

In this $inverse formula, we will be messing with 2 of our official SCSS functions

* change-color
* hue

These official functions are given to you to use; just plug them into your code, and they will work. You can find them at this link:

<https://www.w3schools.com/sass/sass_functions_color.php>

The @return keyword

Use the @return keyword to return what you have in the function definition to the place in the code where it is called, or in other words, **needed**.

We will be using our 3rd official SCSS function in the return. Now if you look at the argument list for mix, it is very interesting. It is not only taking the variables of $color and $amount that was specified in the invert’s argument list, but on top of that it is also returning the mix of colors that was preformed above it and named $inverse. So, not only were the variables returned, but also the expression used to mix one of the two colors. That expression will create one of the colors that is required by the official mix function, the second color will be represented by the $color variable.

When we use the invert function, we must use only the variables that are stated in the parenthesis after the name, and they must be in the same order.

We created a variable to hold our color. It is named $primary-color. Notice the syntax for this variable assignment is the same as a familiar CSS assignment in the body of a rule. It just is using a variable in place of the name that will be associated with the value in the name:value pair.

When we throw the $primary-color into the space where color was at, in the definition, it knows that this is our color. The second one was $amount, and so the answer to that question of amount is 80% (it can be anything we want) and it was put in the second place value. As long as the same placement, and it answers those variable designation of $color, and $amount correctly, the background color will react to our function.



<https://www.w3schools.com/sass/sass_functions_color.php>





