Return

**Overview**

A **return statement** causes execution to leave the current function and resume at the point in the code immediately after where the function was called. Return statements in many languages allow a function to specify a return value to be passed back to the code that called the function.[[1]](https://press.rebus.community/programmingfundamentals/chapter/return-statement/#footnote-1523-1)

<https://sass-lang.com/documentation/at-rules/function/#:~:text=args...)-,%40return,must%20end%20with%20a%20%40return%20>.

# Return

@return. The @return at-rule indicates the value to use as the result of calling a function. It's only allowed within a @function body, and each @function must end with a @return .

The @return at-rule indicates the value to use as the result of calling a function. It’s only allowed within a @function body, and each @function must end with a @return.

When a @return is encountered, it immediately ends the function and returns its result. Returning early can be useful for handling edge-cases or cases where a more efficient algorithm is available without wrapping the entire function in an [@else block](https://sass-lang.com/documentation/at-rules/control/if#else).

Some times you genuinely need a value computed by the function so the vast majority of the cases you need to return some value and so you need to have a return statement in the function to return that value. The only time you do NOT need “return” is when you are returning nothing at the bottom of a function.

# When should a function return a value?

Generally, a return value is used where the function is an intermediate step in a calculation of some kind. You want to get to a final result, which involves some values that need to be calculated by a function.

# When would you use a return value instead of print the value out?

In Python, we may use the print statements to display the final output of a code on the console, whereas the return statement returns a final value of a function execution which may be used further in the code.

# Why print does not need to be returned

If your only purpose in running a function is to make an output visible for human consumption, then the program itself doesn’t really need it, and it will be doing nothing with it. That is why it doesn’t need to be explicitly returned.

# Arguments

Arguments allow functions’ behavior to be customized each time they’re called. The arguments are specified in the @function rule after the function’s name, as a list of variable names surrounded by parentheses. The function must be called with the same number of arguments in the form of [SassScript expressions](https://sass-lang.com/documentation/syntax/structure" \l "expressions). The values of these expression are available within the function’s body as the corresponding variables.

Keyword Arguments

When a function is called, arguments can be passed by name in addition to passing them by their position in the argument list. This is especially useful for functions with multiple optional arguments, or with [boolean](https://sass-lang.com/documentation/values/booleans) arguments whose meanings aren’t obvious without a name to go with them. Keyword arguments use the same syntax as [variable declarations](https://sass-lang.com/documentation/variables) and [optional arguments](https://sass-lang.com/documentation/at-rules/function/#optional-arguments).