#Sass Functions Cheat Sheet

##RGB Functions

`rgb($red, $green, $blue)` Creates a color from red, green, and blue values.

`rgba($red, $green, $blue, $alpha)` Creates a color from red, green, blue, and alpha values.

`red($color)` Gets the red component of a color.

`green($color)` Gets the green component of a color.

`blue($color)` Gets the blue component of a color.

`mix($color1, $color2, [$weight])` Mixes two colors together.

==========

##HSL Functions

`hsl($hue, $saturation, $lightness)` Creates a color from hue,

saturation, and lightness values.

`hsla($hue, $saturation, $lightness, $alpha)` Creates a color from hue, saturation, lightness, and alpha values.

`hue($color)` Gets the hue component of a color.

`saturation($color)` Gets the saturation component of a color.

`lightness($color)` Gets the lightness component of a color.

`adjust-hue($color, $degrees)` Changes the hue of a color.

`lighten($color, $amount)` Makes a color lighter.

`darken($color, $amount)` Makes a color darker.

`saturate($color, $amount)` Makes a color more saturated.

`desaturate($color, $amount)` Makes a color less saturated.

`grayscale($color)` Converts a color to grayscale.

`complement($color)` Returns the complement of a color.

`invert($color)` Returns the inverse of a color.

==========

####Opacity Functions

`alpha($color) / opacity($color)` Gets the alpha component (opacity) of a color.

`rgba($color, $alpha)` Changes the alpha component for a color.

`opacify($color, $amount) / fade-in($color, $amount)` Makes a

color more opaque.

`transparentize($color, $amount) / fade-out($color, $amount)`
Makes a color more transparent.

==========

####Other Color Functions

Visit Sass Functions.

==========

####List Functions

Visit Sass Functions.

==========

####Map Functions

Visit Sass Functions.

==========

####Selector Functions

`selector-nest($selectors...)` Nests selector beneath one another like they would be nested in the stylesheet.

`selector-replace($selector, $original, $replacement)` Replaces `$original` with `$replacement` within `$selector`.

More at Sass Functions.

==========

String Functions

`unquote($string)` Removes quotes from a string.

`quote($string)` Adds quotes to a string.

`str-length($string)` Returns the number of characters in a string.

More at Sass Functions.

==========

Number Functions

`percentage($number)` Converts a unitless number to a percentage.

`round($number)` Rounds a number to the nearest whole number.

`ceil($number)` Rounds a number up to the next whole number.

`floor($number)` Rounds a number down to the previous whole number.

`abs($number)` Returns the absolute value of a number.

`min($numbers...)` Finds the minimum of several numbers.

`max($numbers...)` Finds the maximum of several numbers.

`random([$limit])` Returns a random number.

==========

Introspection Functions

`feature-exists($feature)` Returns whether a feature exists in the current Sass runtime.

`variable-exists($name)` Returns whether a variable with the given name exists in the current scope.

`global-variable-exists($name)` Returns whether a variable with the given name exists in the global scope.

`function-exists($name)` Returns whether a function with the given name exists.

`mixin-exists($name)` Returns whether a mixin with the given name exists.

`inspect($value)` Returns the string representation of a value as it would be represented in Sass.

`type-of($value)` Returns the type of a value.

`unit($number)` Returns the unit(s) associated with a number.

`unitless($number)` Returns whether a number has units.

`comparable($number1, $number2)` Returns whether two numbers can be added, subtracted, or compared.

`call($name, $args…)` Dynamically calls a Sass function.

==========

###Miscellaneous Functions

`if($condition, $if-true, $if-false)` Returns one of two values, depending on whether or not `$condition` is true.

`unique-id()` Returns a unique CSS identifier.