SASS CHEATSHEET BY @CODE4MK

» SCSS

Variable

```scss
$color: red;
$bg-color: $color;
```

Nesting / child

```scss
.users{
    color: red;
    background: blue;
    .profile{
        color: white;
    }
}
/*.users .profile*/
```

Import

```scss
@import  "mixins/_button";
```

Mixin & include

```scss
@mixin bg-style($font-color, $bg-color:
yellow) {
    color: $font-color;
    background: $bg-color;
    /*content purpose */
    @content
}
.profile{
  @include bg-style(red, blue);
}
/*content purpose */
.profile{
  @include bg-style(red, blue){
      font-size: 25px;
  }
}
```

Inheritance / @extend

```scss
.parent{
    color: red;
    font-size: 25px;
}
.child{
  . border: 2px solid red;
   @extend .parent;
}
```

Ebmbrace %

```scss
%parent{
    color: red;
    font-size: 25px;
}
.child{
    border: 2px solid red;
    @extend %parent;
}
```

Function

```scss
$total-width: 100/12;

@function col-size($n){
   return $total-width*$n;
}
.col-md-4{
   width: col-size(4);
}
```

Condition

```scss
@if condition {
   ...........................
}
@else if condition {
   ...........................
}
@else {
   ...........................
}
```

For

```scss
@for $i from 1 through 3{
  .color-#{$i}{
    color: red;
  }
}
```

Each

```scss
$topic:  puma, sea-slug, egret, salamander;
@each $animal in $topic {
  .#{$animal}-icon {
    color: red;
  }
}
```

Each /map

```scss
$icons: (
  checkmark: a,
  plus: b,
  minus: c
);
@each $name, $value in $icons {
  .icon--#{$name} {
    content: $value;
  }
}
```

List

```scss
$list: red, blue;

.body{
  color: nth($list);
}
```

Interpolation

```
#{$var}
```

.sass extension

Basic change

- No need semicolon ;
- Not need curly bracket {}
- Mixin means =
- Include means +

Mixin / include

```sass
=bg-style($font-color, $bg-color: yellow)
  color: $font-color
  background: $bg-color

  /*content purpose
  @content

.profile
  +bg-style(red, blue)

/*content purpose
.profile
  +bg-style(red, blue)
    font-size: 25px
```

Each

```sass
$topic:  puma, sea-slug, egret, salamander;
@each $animal in $topic {
  .#{$animal}-icon {
    color: red;
  }
}
```

@code4mk // Mostafa Kamal

SASS Functions Lists

RGB Functions

rgb($red, $green, $blue)
rgba($red, $green, $blue, $alpha)
red($color_value)
green($color_value)
blue($color_value)
mix($color1_name, $color2_name, $weight)

HSL Functions

hsl($hue, $saturation, $lightness)
hsla($hue, $saturation, $lightness, $alpha)
hue($color)
saturation($color)
lightless($color)
adjust-hue($color, $degrees)
lighten($color,$amount)
darken($color,$amount)
saturate($color,$amount)
desaturate($color,$amount)
complement($color)
grayscale($color)
inver($color)

String Functions

unquote($string)
quote($string)
str-length($string)
str-insert($string, $insert, $index)
str-index($string, $substring)
str-index($string, $substring)
str-slice($string, $start-at, [$end-at])
to-upper-case($string)
to-lower-case($string)

Number Functions

percentage($number)
round($number)
ceil($number)
floor($number)
abs($number)
Min($numbers…)
Max($numbers…)
Random($limit)

List Functions

length($list)
nth($list, $n)
set-nth($list, $n, $value)
join($list1, $list2, [$separator, $bracketed])
append($list1, $val, [$separator])
zip($lists...)
index($list, $value)
list-separator($list)
is-bracketed($list)

Opacity Functions

alpha($color) / opacity($color)
rgba($color, $alpha)
opacify($color, $amount) / fade-in($color, $amount)
transparentize($color, $amount) / fade-out($color, $amount)

Map Functions

map-get($map, $key)
map-merge($map1, $map2)
map-remove($map, $keys...)
map-keys($map)
map-values($map)
map-has-key($map, $key)
keywords($args)

Selector Functions

selector-nest($selectors...)
selector-append($selectors...)
selector-extend($selector, $extendee, $extender)
selector-replace($selector, $original,$replacement)
selector-unify($selector1, $selector2)
is-superselector($super, $sub)
selector-parse($selector)

Code4mk organization

https://code4mk.org
Let's code with fun
@code4mk

//Mostafa Kamal // code4mk //