

**NATHAN**  
**CRANK**

# SASS

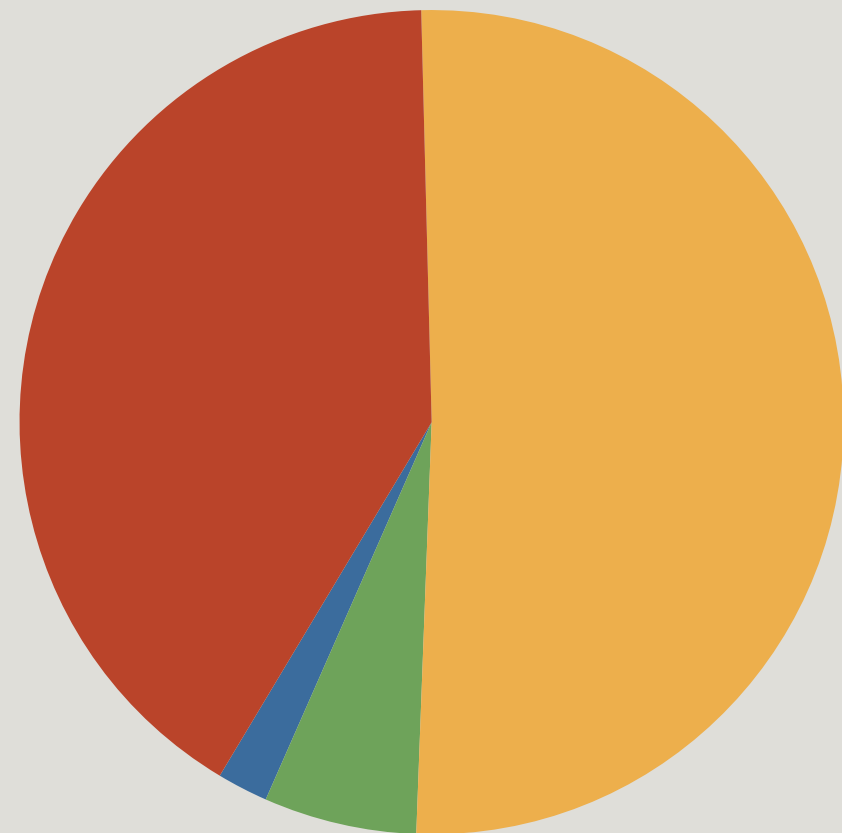
**SASS is a  
CSS preprocessor.**

# Developers are already preprocessing.

- ★ 54% of respondents used or have tried one
- ★ 46% of respondents had not tried one at all.

<http://css-tricks.com/poll-results-popularity-of-css-preprocessors/>

# Which preprocessors are developers using?



# **This is not a popularity contest.**

- ★ They all make CSS easier.
- ★ They all work pretty much everywhere.
- ★ They all have dedicated communities.
- ★ They all have bright futures.

**I like *SASS* best.**

- # Why?
- ★ Familiar syntax
  - ★ Lots of documentation
  - ★ **Compass**



- ★ **variables**
- ★ **nesting**
- ★ **@import**
- ★ **@extend**
- ★ **mixins**
- ★ **functions**
- ★ **control directives**
- ★ **compass**

★ **variables**

# \$varName: value;

```
$brandColor: #d00d1e;  
a {  
  color: $brandColor;  
}
```

```
a { color: #d00d1e; }
```

★ **Number** – 1, 4.2, 13, 25em, 8px

★ **String** – "sassy", 'sassy', sassy

★ **Color** – red, #d00d1e, rgba(235,156,78,.8)

★ **Boolean** – true, false

★ **Null** – null

★ **List** – 1em 4em 2em  
top bottom left  
Akzidenz-Grotesk, Franklin-Gothic, san-serif

★ 3em – 1em	2em
★ #d00d1e + #d00d1e	#ff1a3c
★ 4 * 20px	80px
★ 1200 / 12	100
★ \$width / 4	\$width divided by 4
★ 8px + 12px auto	20px auto
★ "dung" + " beetle"	dung beetle
★ sans- + serif	sans-serif
★ 4em + (5 * 2em)	14em

```
// Spec
$brand: #ba442a; // red
$brandAlt: #272727; // dark gray
$brandBack: #e0ded9; // beige

$headlines: CooperBlack;
$bodyCopy: FranklinGothic, sans-serif;

$padding: .5em;
$paddingLarge: $padding * 2;

// Style
header {
  padding: $paddingLarge;
  background-color: $brandBack - #111;
  font-family: $headlines;
  color: $brand;
}

article {
  padding: $padding;
  background-color: $brandBack;
  font-family: $bodyCopy;
  color: $brandAlt;
}
```

```
header {
  padding: 1em;
  background-color: #cfcfdc8;
  font-family: CooperBlack;
  color: #ba442a;
}

article {
  padding: 0.5em;
  background-color: #e0ded9;
  font-family: FranklinGothic, sans-serif;
  color: #272727;
}
```

```
$mediaQuery: "(min-width: 57em)";
$borderSide: top;
$selectorName: article;
$pseudoName: after;
$eaten: 4;
$fontSize: 1em;
$lineHeight: 1.4em;

@media #{ $mediaQuery } {
  #{ $selectorName } {
    border-#{ $borderSide }: 1px solid red;
  }
  #{ $selectorName }:#{ $pseudoName } {
    content: "Still #{12 - $eaten} donuts";
    font: #{ $fontSize }/#{ $lineHeight };
  }
}
```

```
@media (min-width: 57em) {
  article {
    border-top: 1px solid red;
  }
  article:after {
    content: "Still 8 donuts";
    font: 1em/1.4em;
  }
}
```

# Demo



# Demo

- ★ **variables**
- ★ **nesting**
- ★ **@import**
- ★ **@extend**
- ★ **mixins**
- ★ **functions**
- ★ **control directives**
- ★ **compass**

★ **nesting**

```
article {  
  border: {  
    bottom: 2px solid red;  
    right: 2px solid blue;  
    left: 2px solid blue;  
  }  
  width: 25em;  
  @media (min-width: 77em) {  
    width: 30em;  
  }  
  a {  
    color: blue;  
  }  
}
```

```
article {  
  border-bottom: 2px solid red;  
  border-right: 2px solid blue;  
  border-left: 2px solid blue;  
  width: 25em;  
}  
  
@media (min-width: 77em) {  
  article {  
    width: 30em;  
  }  
}  
  
article a {  
  color: blue;  
}
```

```
article {
  border: {
    bottom: 2px solid red;
    right: 2px solid blue;
    left: 2px solid blue;
  }
  width: 25em;
  @media (min-width: 77em) {
    width: 30em;
  }
  a {
    color: blue;
    &:hover {
      color: red;
    }
  }
}
```

```
article {
  border-bottom: 2px solid red;
  border-right: 2px solid blue;
  border-left: 2px solid blue;
  width: 25em;
}

@media (min-width: 77em) {
  article {
    width: 30em;
  }
}

article a {
  color: blue;
}

article a:hover {
  color: red;
}
```

```
article {
  border: {
    bottom: 2px solid red;
    right: 2px solid blue;
    left: 2px solid blue;
  }
  width: 25em;
  @media (min-width: 77em) {
    width: 30em;
  }
  .oldie & {
    width: 30em;
  }
  a {
    color: blue;
    &:hover {
      color: red;
    }
  }
}
```

```
article {
  border-bottom: 2px solid red;
  border-right: 2px solid blue;
  border-left: 2px solid blue;
  width: 25em;
}

@media (min-width: 77em) {
  article {
    width: 30em;
  }
}

.oldie article {
  width: 30em;
}

article a {
  color: blue;
}

article a:hover {
  color: red;
}
```

```

article {
  @media (min-width: 57em) {
    width: 30em;
    padding: 1em 2em;
    @media (min-height: 27em) {
      padding: 1.5em 2em;
    }
    @media (orientation: portrait) {
      background-color: #d00d1e;
    }
    @media (orientation: landscape) {
      background-color: #d33d1e;
    }
  }
  @media (min-width: 77em) {
    width: 40em;
    background-image: url("bg.png");
    @media (min-resolution: 2ppx),
      (-webkit-min-device-pixel-
        ration: 2) {
      background-image: url("bg@2x.png");
    }
  }
}

```

```

@media (min-width: 57em) {
  article {
    width: 30em;
    padding: 1em 2em;
  }
}
@media (min-width: 57em) and (min-height: 27em) {
  article { padding: 1.5em 2em; }
}
@media (min-width: 57em) and (orientation: portrait) {
  article { background-color: #d00d1e; }
}
@media (min-width: 57em) and (orientation: landscape) {
  article { background-color: #d33d1e; }
}
@media (min-width: 77em) {
  article {
    width: 40em;
    background-image: url("bg.png");
  }
}
@media (min-width: 77em) and (min-resolution: 2ppx),
(min-width: 77em) and (-webkit-min-device-pixel-
  ration: 2) {
  article {
    background-image: url("bg@2x.png");
  }
}

```

# Demo



# Demo

- ★ **variables**
- ★ **nesting**
- ★ **@import**
- ★ **@extend**
- ★ **mixins**
- ★ **functions**
- ★ **control directives**
- ★ **compass**

★ @import

```
@import "normalize",
```

```
    "spec";
```

```
...
```

```
blah {
```

```
    blah: blah;
```

```
}
```

```
...
```



\_normalize.scss



\_spec.scss



# Demo

# Demo

- ★ **variables**
- ★ **nesting**
- ★ **@import**
- ★ **@extend**
- ★ **mixins**
- ★ **functions**
- ★ **control directives**
- ★ **compass**

★ @extend



```
.padding {  
  padding: 1em;  
}  
article {  
  @extend .padding;  
  color: #d00d1e;  
  ...  
}
```

```
.padding, article {  
  padding: 1em;  
}  
article {  
  color: #d00d1e;  
  ...  
}
```

```
%padding {  
  padding: 1em;  
}  
article {  
  @extend %padding;  
  color: #d00d1e;  
  ...  
}  
.sidebar {  
  @extend %padding;  
  ...  
}
```

```
article, .sidebar {  
  padding: 1em;  
}  
article {  
  color: #d00d1e;  
  ...  
}  
.sidebar {  
  ...  
}
```

```
%button {  
  padding: 1em;  
  border: 1px #d00d1e + #111 solid;  
  ...  
}  
.btn-default {  
  @extend %button;  
  background-color: #d00d1e;  
  ...  
}
```

```
.btn-default {  
  padding: 1em;  
  border: 1px #e11e2f solid;  
  ...  
}  
.btn-default {  
  background-color: #d00d1e;  
  ...  
}
```

```
.error {  
  color: red;  
}  
.fail {  
  @extend .error;  
  ...  
}  
.superFail {  
  @extend .fail  
  ...  
}
```

```
.error, .fail, .superFail {  
  color: red;  
}  
.fail, .superFail {  
  ...  
}  
.superFail {  
  ...  
}
```

```
.foo + .bar { ... }
```

```
#foo.bar { ... }
```

```
.doodle {  
  @extend .foo + .bar;  
  @extend #foo.bar;  
}
```

Sass Error: Invalid CSS after ".foo + .bar  
{ ... ": expected "{", was "}"

# Demo

# Demo

- ★ **variables**
- ★ **nesting**
- ★ **@import**
- ★ **@extend**
- ★ **mixins**
- ★ **functions**
- ★ **control directives**
- ★ **compass**





```
header {  
  -webkit-transition: all .5s ease-in-out;  
  -moz-transition: all .5s ease-in-out;  
  -o-transition: all .5s ease-in-out;  
  transition: all .5s ease-in-out;  
}
```

```
@mixin animate($time: .5s, $ease: ease-in-out, $prop: all) {  
  -webkit-transition: $prop $time $ease;  
  -moz-transition: $prop $time $ease;  
  -o-transition: $prop $time $ease;  
  transition: $prop $time $ease;  
}  
  
header { @include animate; }
```

```
@mixin animate($time: .5s, $ease: ease-in-out, $prop: all) {  
  -webkit-transition: $prop $time $ease;  
  -moz-transition: $prop $time $ease;  
  -o-transition: $prop $time $ease;  
  transition: $prop $time $ease;  
}  
  
header { @include animate; }  
article { @include animate(); }  
aside { @include animate(1s); }  
footer { @include animate(2s, linear, "color"); }  
$ourTime: 3s;  
$ourEase: linear;  
$ourProperty: color;  
.modal { @include animate($ourTime, $ourEase, $ourProperty); }  
button { @include animate($prop: background; }
```

```
@mixin animate($args) {  
  -webkit-transition: $args;  
  -moz-transition: $args;  
  -o-transition: $args;  
  transition: $args;  
}  
  
header { @include animate(all .5s ease-in-out); }
```

```
@mixin box-sizing {  
  -webkit-box-sizing: border-box;  
  -moz-box-sizing: border-box;  
  box-sizing: border-box;  
}  
  
* {  
  @include box-sizing;  
}
```

```
* {  
  -webkit-box-sizing: border-box;  
  -moz-box-sizing: border-box;  
  box-sizing: border-box;  
}
```

```
@mixin linkColors($color: blue) {  
  color: $color;  
  &:hover {  
    color: $color + #222;  
  }  
  &:active {  
    color: $color - #222;  
  }  
}  
  
a { @include linkColors(#d00d1e); }
```

```
a { color: #d00d1e; }  
a:hover { color: #f22f40; }  
a:active { color: #ae0000; }
```

```
@mixin tablet {
  @media (min-width: 57em) {
    @content
  }
}

@mixin desktop {
  @media (min-width: 67em) {
    @content
  }
}

article {
  width: 100%;
  @include tablet { width: 70%; }
  @include desktop { width: 60%; }
}
```

```
article { width: 100%; }
@media (min-width: 57em) {
  article { width: 70%; }
}
```

```
@media (min-width: 67em) {
  article { width: 60%; }
}
```



```
@mixin ie7 {  
  .lt-ie8 & { @content }  
}  
@mixin ie8 {  
  .lt-ie9 & { @content }  
}  
article {  
  width: 20em;  
  @include ie8 { width: 22em };  
  @include ie7 { width: 21em };  
}
```

```
article {  
  width: 20em;  
}  
.lt-ie9 article {  
  width: 22em;  
}  
.lt-ie8 article {  
  width: 21em;  
}
```

# Demo

# Demo

- ★ **variables**
- ★ **nesting**
- ★ **@import**
- ★ **@extend**
- ★ **mixins**
- ★ **functions**
- ★ **control directives**
- ★ **compass**

★ **functions**

```
article {  
  color: rgba(#d00d1e, .8);  
  font-size: #{floor(3.14)}em;  
}
```

```
article {  
  color: rgba(208, 13, 30, 0.8);  
  font-size: 3em;  
}
```

```
@function xPi($var) {  
  @return 3.1415926 * $var;  
}  
  
article {  
  font-size: #{xPi(2)}em;  
}
```

```
article {  
  font-size: 6.28319em;  
}
```

★ **rgba(\$color, \$opacity)**

★ **invert(\$color)**

★ **round(\$value)**

★ **ceil(\$value)**

★ **floor(\$value)**

★ **abs(\$value)**

<http://sass-lang.com/docs/yardoc/Sass/Script/Functions.html#list-functions>



- ★ **variables**
- ★ **nesting**
- ★ **@import**
- ★ **@extend**
- ★ **mixins**
- ★ **functions**
- ★ **control directives**
- ★ **compass**

★ **control directives**

```

@mixin sens-variable-sizing($divisor: 4, $min: 57, $max: 157) {
  @for $i from $min through $max {
    @include sens-min("#{i}em") {
      font-size: #{exponent($i,1,3)/$divisor}em;
    }
  }
}
article {
  @include sens-variable-sizing;
}

```

```

@media (min-width: 57em) {
  article {
    font-size: 0.96213em;
  }
}
@media (min-width: 58em) {
  article {
    font-size: 0.96772em;
  }
}
@media (min-width: 59em) {
  article {
    font-size: 0.97325em;
  }
}
...
@media (min-width: 155em) {
  article {
    font-size: 1.34292em;
  }
}
@media (min-width: 156em) {
  article {
    font-size: 1.3458em;
  }
}
@media (min-width: 157em) {
  article {
    font-size: 1.34867em;
  }
}
}

```

```

@mixin sens-variable-sizing($divisor: 4, $min: 57, $max: 157, $inc: 2)
{
  $i: $min;
  @while $i <= $max {
    @include sens-min("#{i}em) {
      font-size: #{exponent($i,1,3)/$divisor}em;
    }
    $i: $i + $inc;
  }
}
article {
  @include sens-variable-sizing;
}

```

```

@media (min-width: 57em) {
  article {
    font-size: 0.96213em;
  }
}
@media (min-width: 59em) {
  article {
    font-size: 0.97325em;
  }
}
@media (min-width: 61em) {
  article {
    font-size: 0.98412em;
  }
}
...
@media (min-width: 153em) {
  article {
    font-size: 1.33712em;
  }
}
@media (min-width: 155em) {
  article {
    font-size: 1.34292em;
  }
}
@media (min-width: 157em) {
  article {
    font-size: 1.34867em;
  }
}
}

```

```
@mixin borders($sides) {  
  @each $side in $sides {  
    border-#{ $side}: 2px solid red;  
  }  
}  
article {  
  @include borders(top left right);  
}
```

```
article {  
  border-top: 2px solid red;  
  border-left: 2px solid red;  
  border-right: 2px solid red;  
}
```

```

@mixin sens-background-image($img: "default",
                             $state: "",
                             $inc: 1,
                             $ext: "png",
                             $max: 2,
                             $min: 1,
                             $delim: " ",
                             $delim2: "@") {
  @if $state != "" {
    $state: $delim + $state;
  }
  $i: $min;
  @while $i <= $max {
    @if $i == 1 {
      background-image: url($img + $state + "." + $ext);
    } @else {
      @include sens-min-density($i) {
        background-image: url($img + $state + $delim2 + $i + "x" + "." +
$ext);
      };
    }
    $i: $i + $inc;
  }
}

```

```

article {
  background-image: url("picture.png");
}
@media (-webkit-min-device-pixel-ratio: 2), (min-resolution: 2dppx) {
  article {
    background-image: url("picture@2x.png");
  }
}

```

```

@mixin queryFor($q: null) {
  @if $q == mobile {
    @media (max-width: 32em) {
      @content
    }
  } @else if $q == tablet {
    @media (min-width: 32em) {
      @content
    }
  } @else if $q == notebook {
    @media (min-width: 57em) {
      @content
    }
  } @else if $q == desktop {
    @media (min-width: 77em) {
      @content
    }
  } @else if $q == oldie {
    .oldie & {
      @content
    }
  } @else if $q == null {
    @warn "You're doing it wrong!";
  }
  @else {
    @media (#{ $q }) {
      @content
    }
  }
}

article {
  width: 100%;
  @include queryFor(tablet) {
    width: 70%;
  }
  @include queryFor(notebook) {
    width: 60%;
  }
  @include queryFor(oldie) {
    width: 80%;
  }
  @include queryFor("min-width: 99em") {
    width: 50%;
  }
  @include queryFor {
    width: auto;
  }
}

```

```

article {
  width: 100%;
}
@media (min-width: 32em) {
  article {
    width: 70%;
  }
}
@media (min-width: 57em) {
  article {
    width: 60%;
  }
}

.oldie article {
  width: 80%;
}

@media (min-width: 99em) {
  article {
    width: 50%;
  }
}

```

```

@function aproxequal($val1, $val2, $prec: .00002) {
  @if abs($val1 - $val2) <= $prec { @return true; }
  @else { @return false; }
}

@function notaproxequal($val1, $val2) {
  @if aproxequal($val1, $val2) == false { @return true; }
  @else { @return false; }
}

@function nthroot($num,$n: 2) {
  $neg: false;
  @if $n < 0 {
    $neg: true;
  }
  $num: abs($num);
  $value: $num;
  $testValue: exponent($value, $n);
  @while notaproxequal($num,$testValue) {
    $value: (1 / $n) * ( ( ($n - 1) * $value) + ($num /
exponent($value,$n - 1)) );
    @if $testValue == Infinity { @return null };
  }
  @if $neg == true {
    @return 1/$value;
  } @else {
    @return $value;
  }
}

```

```

@function sqrt($num) {
  @return nthroot($num);
}

@function cubert($num) {
  @return nthroot($num, 3);
}

@function exponent($num,$expN,$expD: 1) {
  $value: $num;
  $neg: false;
  $expDCalc: false;
  @if $expN < 0 {
    $expN: $expN * -1;
    $neg: true;
  }
  @if $expD != 1 {
    $expDCalc: true;
    @if $expD < 0 {
      $expD: $expD * -1;
      @if $neg == true { $neg: false; }
      @else { $neg: true; }
    }
  }
  @for $i from 1 to $expN {
    $value: $value * $num;
  }
  @if $expDCalc == true { $value: nthroot($value,$expD); }
  @if $neg == true {
    @return 1/$value;
  } @else {
    @return $value;
  }
}

@function square($num) {
  @return exponent($num, 2);
}

@function cube($num) {
  @return exponent($num, 3);
}

```



**the what?**

the  
**Newton–Raphson**  
method

```

@function aproxequal($val1, $val2, $prec: .00002) {
  @if abs($val1 - $val2) <= $prec { @return true; }
  @else { @return false; }
}

@function notaproxequal($val1, $val2) {
  @if aproxequal($val1, $val2) == false { @return true; }
  @else { @return false; }
}

@function nthroot($num,$n: 2) {
  $neg: false;
  @if $n < 0 {
    $neg: true;
  }
  $num: abs($num);
  $value: $num;
  $testValue: exponent($value, $n);
  @while notaproxequal($num,$testValue) {
    $value: (1 / $n) * ( ( ($n - 1) * $value) + ($num /
exponent($value,$n - 1)) );
    @if $testValue == Infinity { @return null };
  }
  @if $neg == true {
    @return 1/$value;
  } @else {
    @return $value;
  }
}

```

```

@function sqrt($num) {
  @return nthroot($num);
}

@function cubert($num) {
  @return nthroot($num, 3);
}

@function exponent($num,$expN,$expD: 1) {
  $value: $num;
  $neg: false;
  $expDCalc: false;
  @if $expN < 0 {
    $expN: $expN * -1;
    $neg: true;
  }
  @if $expD != 1 {
    $expDCalc: true;
    @if $expD < 0 {
      $expD: $expD * -1;
      @if $neg == true { $neg: false; }
      @else { $neg: true; }
    }
  }
  @for $i from 1 to $expN {
    $value: $value * $num;
  }
  @if $expDCalc == true { $value: nthroot($value,$expD); }
  @if $neg == true {
    @return 1/$value;
  } @else {
    @return $value;
  }
}

@function square($num) {
  @return exponent($num, 2);
}

@function cube($num) {
  @return exponent($num, 3);
}

```

- ★ **variables**
- ★ **nesting**
- ★ **@import**
- ★ **@extend**
- ★ **mixins**
- ★ **functions**
- ★ **control directives**
- ★ **compass**



```
article {  
  -webkit-box-shadow: rgba(208, 13, 30, 0.5) 0 3px 3px inset;  
  -moz-box-shadow: rgba(208, 13, 30, 0.5) 0 3px 3px inset;  
  box-shadow: rgba(208, 13, 30, 0.5) 0 3px 3px inset;  
}
```

```
article {  
  @include box-shadow(rgba(#d00d1e, 0.5) 0 3px 3px inset);  
}
```

```
@include box-shadow(rgba(#d00d1e, 0.5) 0 3px 3px inset);  
@include border-radius(4px);  
@include background-image(linear-gradient(#d00d1e + #111, #d00d1e));  
@include transition(transform 1s linear);  
@include background(image-url("foo.png"),  
                    linear-gradient(top left, #333, #0c0),  
                    radial-gradient(#c00, #fff 100px)  
);
```

<http://compass-style.org>



# Demo

# Demo

# Where does it run?

**Mac | PC | Linux**

**GUI | Command Line(!)**

**Basically everywhere ...**

# GUI tools

- ★ LiveReload (Mac, Windows, Linux)
- ★ Compass.app (Mac, Windows, Linux)
- ★ Scout (Mac, Windows)
- ★ Codekit (Mac)

# **\*nix command line**

★ `sudo gem install sass`

★ `sudo gem install compass`

# Windows Command Line

★ Google it.

★ Just Kidding:

<http://www.impressivewebs.com/sass-on-windows/>



## Ruby

It's included by default!



## Wordpress

<http://thesassway.com/projects/sass-for-wordpress>



## Sublime Text

More options & plugins than any human should ever have access to



## Visual Studio

<http://www.mindscapehq.com/products/web-workbench>



## Eclipse



## NetBeans



**SASS fits in any workflow**

# Check out my SASS

★ **Sensible**  
[sensible.nathancrank.com](http://sensible.nathancrank.com)

★ **Zealot**  
[zealot.nathancrank.com](http://zealot.nathancrank.com)

★ **Github**  
[github.nathancrank.com](http://github.nathancrank.com)



# Awesome Resources

## ★ Useful SASS Mixins

<http://sachagreif.com/useful-sass-mixins/>

## ★ The SASS way

<http://thesassway.com>

## ★ SASS\_REFERENCE

[http://sass-lang.com/docs/yardoc/  
file.SASS\\_REFERENCE.html#boolean\\_operations](http://sass-lang.com/docs/yardoc/file.SASS_REFERENCE.html#boolean_operations)

**Download today's  
slides and demos:  
[sass.nathancrank.com](http://sass.nathancrank.com)**

**NATHAN**  
**CRANK**

**www.**

**twitter.**

**linkedin.**

**me@nathancrank.com**

**github.**

**sass.**

**sensible.**

**zealot.**