Maintainable CSS
Syntactically Awesome Style Sheets

Why ?

Writing a lot of CSS can be overwhelming and hard to maintain.
Thanks to this CSS pre–processor, it's now possible to write
DRY CSS code and allows you to declare variables and «some»
other features that can be re-used all throughout the style.

Meaningful features...

SASS comes with a set of additional features to write
your generate your CSS such as Variables,
Nesting, Mixins, Property inheritance and Operators

Beautiful syntax...

SASS is a space sensitive way to write CSS and uses indentation instead of { } to delimit code blocks.

—

Everything that would be within { and } after a statement must be on a new line and indented one level deeper than the statement

—

Tabs and spaces are not the same even if they look the same!

styles/stylesheet.css

```css
#main {
  color: blue;
  font-size: 0.3em;
}
```

styles/stylesheet.sass

```sass
#main
    color: blue
    font-size: 0.3em
```

Easy in Rails...

Rails compile .sass files for you, all you need is to change .scss file estention to .sass.

Or in basic environments

sass --watch sass_folder:stylesheets_folder

Sass provides higher level style syntax and takes your
preprocessed SASS file and save it as a normal CSS
file that you can use in your web site. (we will demo that)

styles/stylesheet.sass

```sass
$cool: #FF4848

#main
    color: $cool


p
    color: $cool
```

styles/stylesheet.css

```css
#main {
    color: #FF4848;
}


p {
    color: #FF4848;
}
```

styles/stylesheet.sass

```sass
$myFontsize: 13px
$myFontsize2: 15px
$myWidth: 500px
$myMargin: 0px auto

#container
        width: $myWidth
        margin: $myMargin

        p

                font-family: Arial
                font-size: $myFontsize


        h2

                font-family: Helvetica
                font-size: $myFontsize2
```

styles/stylesheet.css

```css
//Creates the following CSS...

#container p {
    font-family: Arial;
    font-size: 13px;
}


#container h1 {
    font-family: Tahoma;
    font-size: 15px;
}


#container h2 {
    font-family: Helvetica;
    font-size: 14px;
}
```

styles/stylesheet.sass

```sass
@mixin border-radius($amount: 5px)
    -moz-border-radius: $amount
    -webkit-border-radius: $amount
    -ms-border-radius: $amount
    border-radius: $amount

h1
    @include border-radius(2px)

.h2
    @include border-radius
```

styles/stylesheet.css

```css
//Creates the following CSS...

h1 {
  -moz-border-radius: 2px;
  -webkit-border-radius: 2px;
  -ms-border-radius: 2px;
  border-radius: 2px;
}

h2 {
  -moz-border-radius: 5px;
  -webkit-border-radius: 5px;
  -ms-border-radius: 5px;
  border-radius: 5px;
}
```

styles/stylesheet.sass

```sass
.msg
    border: 1px solid #ccc
    padding: 10px
    color: #333


.success
    @extend .msg border-color: green


.error
    @extend .msg border-color: red


.warning
    @extend .msg border-color: yellow
```

styles/stylesheet.css

```css
.message, .success, .error, .warning {
  border: 1px solid #cccccc;
  padding: 10px;
}


.success {
  border-color: green;
}


.error {
  border-color: red;
}


.warning {
  border-color: yellow;
}
```

styles/stylesheet.sass

```sass
.container
    width: 100%

article
    width: 600px / 960px * 100%

aside
    width: 300px / 960px * 100%
```

styles/stylesheet.css

```css
.container {
  width: 100%;
}

article {
  width: 62.5%;
}

aside {
  width: 31.25%;
}
```

Q&A

.sass? .less? .scss? .css? .map?

CSS is fun again!