

SASS



Mark Zhang

What is SASS?

- SASS is an extension of CSS.
- It provides useful functionality to CSS such as nesting, variables, and functions.
- Allows you to reuse your CSS and keeps your CSS modular.
- It's very easy to learn and use.

Installation

- You first have to install Ruby.
- Specific instructions on the SASS website.
 - ▣ <http://sass-lang.com/>

Using SASS

□ `sass -watch style.sass:style.css`

Two Syntaxes

- SASS – uses indentation to specify blocks
- SCSS – uses braces and semicolons

- SASS is more terse and less like normal CSS.

Questions?



Essential SASS Features

- Nesting
- Variables
- Built-in Functions
- Mixins
- Selector Inheritance
- Importing Partials

Nesting

CSS:

```
table tr.foo {
  padding: 0;
}
table .h1 {
  margin: 2em 0;
}
li {
  font-family: serif;
  font-weight: bold;
  font-size: 15px;
}
```

SASS:

```
table
  tr.foo
    padding: 0
  h1
    margin: 2em 0
  li
    font
      family: serif
      weight: bold
      size: 15px
```


The & symbol

CSS:

```
a {  
  color: #ce4dd6;  
}  
a:hover {  
  color: #ffb3ff;  
}  
a:visited {  
  color: #c458cb;  
}
```

SASS:

```
a  
  color: #ce4dd6  
&:hover  
  color: #ffb3ff;  
&:visited  
  color: #c458cb;
```

Variables

CSS:

```
.wrapper {  
  border-color: #3bbfce;  
  padding: 8px  
}  
.nav {  
  color: #3bbfce;  
  margin-bottom: 16px;  
}
```

SASS:

```
$accent: #3bbfce  
$spacing: 16px  
.wrapper  
  border-color: $accent  
  padding: $spacing/2  
.nav  
  color: $accent  
  margin-bottom: $spacing
```

Built-in Functions

- standard math operations (+, -, *, /, and %)
- color functions like “lighten” and “darken”

SASS:

```
$navbar-color:  
  #ce4dd6
```

```
background-color:  
  lighten($navbar-color,  
  10%)
```

CSS:

```
background-color:  
  #d976e0;
```

Mixins

- Are reusable style blocks
- They can take variables!
- `@mixin` and `@include`

Mixins

CSS:

```
.nav {
```

```
  box-shadow: 1px 1px 4px 2px #ccc
```

```
  -webkit-box-shadow: 1px 1px 4px 2px #ccc
```

```
  -moz-shadow: 1px 1px 4px 2px #ccc
```

```
  -o-shadow: 1px 1px 4px 2px #ccc }
```

```
.wrapper {
```

```
  box-shadow: -1px -1px 2px 0 #ccc
```

```
  -webkit-box-shadow: -1px -1px 2px 0 #ccc
```

```
  -moz-box-shadow: -1px -1px 2px 0 #ccc
```

```
  -o-box-shadow: -1px -1px 2px 0 #ccc }
```

Mixins

SASS:

```
@mixin box-shadow($blur, $spread, $h, $v, $color)
```

```
-moz-box-shadow: $h $v $blur $spread $color
```

```
-webkit-box-shadow: $h $v $blur $spread $color
```

```
-o-box-shadow: $h $v $blur $spread $color
```

```
box-shadow: $h $v $blur $spread $color
```

.nav

```
@include box-shadow(1px, 1px, 4px, 2px, #ccc)
```

.wrapper

```
@include box-shadow(-1px, -1px, 2px, 0, #ccc)
```

Mixins

- Can even take variable numbers of arguments or default arguments.

Selector Inheritance

- Uses “@extend”
- Allows one selector to inherit all styles of another selector (included styles within nested selectors).

Selector Inheritance

SASS

`.error`

`border: 1px #f00`

`background: #fdd`

`.intrusion`

`font-size: 1.3em`

`font-weight: bold`

`.badError`

`@extend .error`

`border-width: 3px`

CSS

`.error, .badError`

`{ border: 1px #f00;`
`background: #fdd; }`

`.error.intrusion, .badError`

`.intrusion { font-size:`
`1.3em; font-weight:`
`bold; }`

`.badError {`

`border-width: 3px; }`

Partials and Importing

- You can separate your SASS into files.
- Files that are “incomplete” and meant to be loaded into other SASS files are called partials.
- Naming convention: `_colors.scss`
- `@import "colors"` would include all the styles in `_colors.scss` during compilation

Essential SASS Features

- Nesting
- Variables
- Built-in Functions
- Mixins
- Selector Inheritance
- Importing Partials

Summary

- SASS is an extension of CSS.
- It provides features that allow you to easily reuse and tweak your styling.
- It reduces repetition in your styling.

SASS



Appian

