

SASS

Authored by: Asfiya Khan

Presented by: Asfiya Khan

Document History

Version No.	Authored/ Modified by	Remarks/ Change History	Date <dd- mon-yy >
1	Asfiya Khan		13-09-2017

Course Structure

Target audience	SE,SSE
Level	1,2,3
Pre-requisites	CSS Knowledge
Training methods	Presentation and examples.
Evaluation	OTT

Agenda

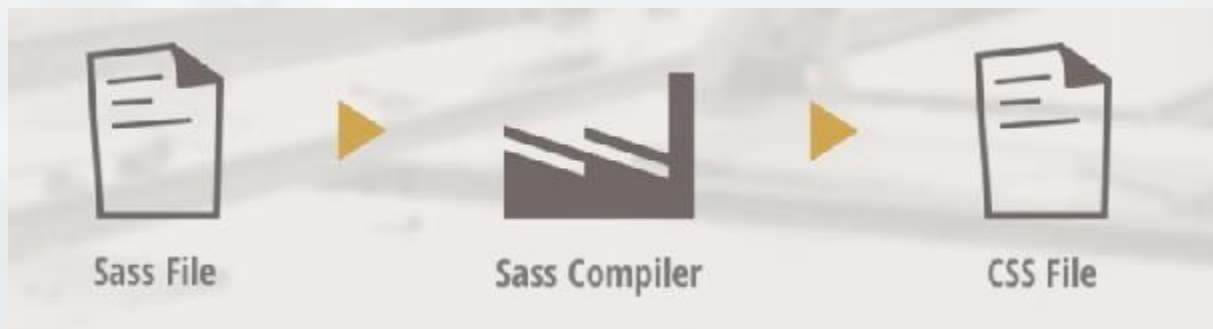
- Introduction to SASS
- Features of SASS
- How to use SASS
- Variables
- Nesting
- Comments
- Mixins
- Selector inheritance
- Include one sass into another
- Demo



An Introduction to



- Sass is simply syntactic sugar for CSS.
- Sass is a scripting language that is interpreted or compiled into CSS.
- Created by Hampton Catlin.
- It uses indentation to separate code blocks and newline characters to separate rules.
- Preprocessors like: CoffeScript and HAML



SASS and SCSS

An Introduction to



Sass has two syntaxes. The most commonly used syntax is known as “SCSS” (for “Sassy CSS”), and is a superset of CSS3’s syntax.

This means that every valid CSS3 stylesheet is valid SCSS as well. SCSS files use the extension .scss.

- Default Extension is .scss
- CSS is a valid SCSS
- Second syntax is (.sass) but we will focus on SCSS.

Introduction

- Lets begin with all the features of SASS.

Variables

- Variables begin with a dollar sign (\$) and assignment is done with a colon (:).
- SassScript supports four data types:
 1. Numbers (including units)
 2. Strings (with quotes or without)
 3. Colors (name, or names)
 4. Booleans

The Sass logo is written in a white, elegant script font and is centered within a magenta oval. The background of the slide features a faint, grayscale image of interlocking gears and a hand pointing towards them.

Variables

Sassmeister File View Options Login with Github	
Sass v3.4.21	CSS
<pre> 1 \$primary-color: #3bbfce; 2 \$margin: 16px; 3 4 .content-navigation { 5 border-color: \$primary-color; 6 color: darken(\$primary-color, 10%); 7 } 8 9 .border { 10 padding: \$margin / 2; 11 margin: \$margin / 2; 12 border-color: \$primary-color; 13 }</pre>	<pre> 1 .content-navigation { 2 border-color: #3bbfce; 3 color: #2b9eab; 4 } 5 6 .border { 7 padding: 8px; 8 margin: 8px; 9 border-color: #3bbfce; 10 } 11</pre>

Comments

application.scss

```
// These comments will  
// not be output to the  
// compiled CSS file  
  
/* This comment will */
```

application.css


```
/* This comment will */
```

Loops

- Sass allows for iterating over variables using **@for**, **@each** and **@while**, which can be used to apply different styles to elements with similar classes or ids.

Sassmeister File View Options Login with Github	
Sass v3.4.21	CSS Compiled in 0.074s
<pre> 1 \$squareCount: 3; 2 @for \$i from 1 through \$squareCount { 3 #square-#{ \$i } 4 { 5 background-color: red; 6 width: 50px * \$i; 7 height: 120px / \$i; 8 } 9 }</pre>	<pre> 1 #square-1 { 2 background-color: red; 3 width: 50px; 4 height: 120px; 5 } 6 7 #square-2 { 8 background-color: red; 9 width: 100px; 10 height: 60px; 11 } 12 13 #square-3 { 14 background-color: red; 15 width: 150px; 16 height: 40px; 17 } 18</pre>

Loops


File View Options Login with Github

Sass v3.4.21

```

1 $colors: red, orange, yellow, green, blue;
2 $i: 1;
3
4 @each $color in $colors {
5   li:nth-of-type(#{$i}) {
6     color: $color;
7   }
8
9   $i: $i + 1;
10 }
11
12
13 $flag: true;
14 @if $flag==true{
15   #mystyle-#{ $i+1 }{
16     background-color :#FFFFFF+1;
17   }
18 }
19 @else {
20   #mystyle-#{ $i+1 }{
21     color :#FFFFFF+1;
22   }
23 }
24

```

CSS


```

1 li:nth-of-type(1) {
2   color: red;
3 }
4
5 li:nth-of-type(2) {
6   color: orange;
7 }
8
9 li:nth-of-type(3) {
10  color: yellow;
11 }
12
13 li:nth-of-type(4) {
14  color: green;
15 }
16
17 li:nth-of-type(5) {
18  color: blue;
19 }
20
21 #mystyle-7 {
22   background-color: #FFFFFF1;
23 }
24

```

Problems

application.css




```
.content {  
  border: 1px solid #ccc;  
  padding: 20px;  
}  
▶ .content h2 {  
  font-size: 3em;  
  margin: 20px 0;  
}  
▶ .content p {  
  font-size: 1.5em;  
  margin: 15px 0;  
}
```

Nesting is the Solution!!!

application.scss


```
.content {  
  border: 1px solid #ccc;  
  padding: 20px;  
  h2 {  
    font-size: 3em;  
    margin: 20px 0;  
  }  
  p {  
    font-size: 1.5em;  
    margin: 15px 0;  
  }  
}
```

application.css



```
.content {  
  border: 1px solid #ccc;  
  padding: 20px;  
}  
.content h2 {  
  font-size: 3em;  
  margin: 20px 0;  
}  
.content p {  
  font-size: 1.5em;  
  margin: 15px 0;  
}
```

Nesting


File **View** **Options** **Login with Github**

Sass v3.4.21	CSS
<pre> 1 table.hl { 2 margin: 2em 0; 3 td.ln { 4 text-align: right; 5 } 6 } 7 8 li { 9 font: { 10 family: serif; 11 weight: bold; 12 size: 1.3em; 13 } 14 }</pre>	<pre> 1 table.hl { 2 margin: 2em 0; 3 } 4 table.hl td.ln { 5 text-align: right; 6 } 7 8 li { 9 font-family: serif; 10 font-weight: bold; 11 font-size: 1.3em; 12 } 13</pre>

imports

application.scss

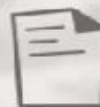
```
// Imports styles found in 'buttons.scss'  
// when the compiler processes application.scss  
  
@import "buttons";
```



application.scss



buttons.scss



application.css

How to Resolve?

application.scss

```
// Imports styles found in 'buttons.scss'  
// when the compiler processes application.scss  
  
@import "buttons";
```



application.scss



buttons.scss



application.css



buttons.css

*buttons.css is created
even if we're importing*

Partials

application.scss

```
// Will import _buttons.sass, buttons.sass,  
// _buttons.scss, or buttons.scss
```

```
@import "buttons";
```



application.scss



_buttons.scss



application.css

imports

// _reset.scss	// base.scss	//Complied CSS
<pre>html, body, ul, ol { margin: 0; padding: 0; }</pre>	<pre>@import 'reset'; body { font: 100% Helvetica, sans-serif; background-color: #efefef; }</pre>	<pre>html, body, ul, ol { margin: 0; padding: 0; } body { font: 100% Helvetica, sans-serif; background-color: #efefef; }</pre>

Mixins



The screenshot shows the Sassmeister web editor interface. The top bar is pink with the 'Sassmeister' logo and navigation links: 'File', 'View', 'Options', and 'Login with Github'. Below the bar, the editor is split into two panes. The left pane, titled 'Sass v3.4.21', contains the following Sass code:

```
1 ▾ @mixin border-radius($radius) {  
2   -webkit-border-radius: $radius;  
3   -moz-border-radius: $radius;  
4   -ms-border-radius: $radius;  
5   border-radius: $radius;  
6 }  
7  
8 .box { @include border-radius(10px); }
```

The right pane, titled 'CSS', shows the compiled CSS output:

```
1 ▾ .box {  
2   -webkit-border-radius: 10px;  
3   -moz-border-radius: 10px;  
4   -ms-border-radius: 10px;  
5   border-radius: 10px;  
6 }  
7
```

The top right of the right pane indicates 'Compiled in 0.007s'.

Extend/Inheritance


File View Options Login with Github

Sass v3.4.21

```

1  .message {
2    border: 1px solid #ccc;
3    padding: 10px;
4    color: #333;
5  }
6
7  .success {
8    @extend .message;
9    border-color: green;
10 }
11
12 .error {
13   @extend .message;
14   border-color: red;
15 }
16
17 .warning {
18   @extend .message;
19   border-color: yellow;
20 }

```

CSS

```

1  .message, .success, .error, .warning {
2    border: 1px solid #ccc;
3    padding: 10px;
4    color: #333;
5  }
6
7  .success {
8    border-color: green;
9  }
10
11 .error {
12   border-color: red;
13 }
14
15 .warning {
16   border-color: yellow;
17 }
18

```

Any questions!



DAY 2

Installation of SASS on your machine!!!!

Project Structure of SASS

```
bootstrap4-sass/  
|-- sass/  
|   |-- base/  
|   |   |-- _import.scss  
|   |   |-- _typography.scss  
|   |-- helpers/  
|   |   |-- _import.scss  
|   |   |-- _typography.scss  
|   |-- style.scss  
|   ...  
|-- js/  
|   |-- custom.js  
|   ...  
|-- img/  
|   |-- card-img.jpeg  
|   |-- slide-1.jpeg  
|   |-- slide-2.jpeg  
|   |-- slide-3.jpeg  
|   ...  
|-- css/  
|   |-- style.css  
|-- END
```


Any questions!





Thank You!