

SASS / SCSS

ePorftolio, Simon Knab, 30.05.2017

Agenda

The background of the slide features abstract, overlapping geometric shapes in various shades of green, ranging from light lime to dark forest green. These shapes are primarily located on the right side and bottom of the slide, creating a modern, dynamic feel. The main content area on the left is a plain white background.

Definition

- ▶ SASS (Syntactically Awesome Stylesheets)
- ▶ Stylesheet language
- ▶ Preprocessor for easier creation of Cascading Style Sheets (CSS)
- ▶ Possible syntax:

SASS	SCSS
Nesting through indentation	Nesting through curly brackets
<pre>\$meineFarbe: #3BBFCE .navigation border-color: \$meineFarbe color: darken(\$meineFarbe, 9%)</pre>	<pre>\$meineFarbe: #3BBFCE; .navigation { border-color: \$meineFarbe; color: darken(\$meineFarbe, 9%); }</pre>

Partials / Import

- ▶ Split the stylesheet in different parts for a better overview
- ▶ Partials names should start with a underscore
e.g. "_Whatever.scss"
- ▶ Import with `@import "Whatever"`
(underscore and file ending are not needed)
- ▶ Final result is one merged CSS-File for the browser.

Variables

- Use \$ to make something a variable

```
$font-stack: Helvetica, sans-serif;  
$primary-color: #333;  
  
body {  
  font: 100% $font-stack;  
  color: $primary-color;  
}
```

```
body {  
  font: 100% Helvetica, sans-serif;  
  color: #333;  
}
```

Nesting

► Nesting through curly brackets

```
nav {  
  ul {  
    margin: 0;  
    padding: 0;  
    list-style: none;  
  }  
  
  li { display: inline-block; }  
  
  a {  
    display: block;  
    padding: 6px 12px;  
    text-decoration: none;  
  }  
}
```

```
nav ul {  
  margin: 0;  
  padding: 0;  
  list-style: none;  
}  
  
nav li {  
  display: inline-block;  
}  
  
nav a {  
  display: block;  
  padding: 6px 12px;  
  text-decoration: none;  
}
```

Extend/Inheritance

- Share informations through inheritance

```
.message {  
  border: 1px solid #ccc;  
  padding: 10px;  
  color: #333;  
}  
  
.success {  
  @extend .message;  
  border-color: green;  
}  
  
.error {  
  @extend .message;  
  border-color: red;  
}
```

```
.message, .success, .error, .warning {  
  border: 1px solid #cccccc;  
  padding: 10px;  
  color: #333;  
}  
  
.success {  
  border-color: green;  
}  
  
.error {  
  border-color: red;  
}
```

Mixins

- ▶ Work similar to inheritance but you can use parameters

```
@mixin border-radius($radius) {  
  -webkit-border-radius: $radius;  
  -moz-border-radius: $radius;  
  -ms-border-radius: $radius;  
  border-radius: $radius;  
}  
  
.box { @include border-radius(10px); }
```

```
.box {  
  -webkit-border-radius: 10px;  
  -moz-border-radius: 10px;  
  -ms-border-radius: 10px;  
  border-radius: 10px;  
}
```


Operators

- You can use the math operators +, -, *, / and % to calculate some values

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

article[role="main"] {
  float: left;
  width: 600px / 960px * 100%;
}

aside[role="complementary"] {
  float: right;
  width: 300px / 960px * 100%;
}
```

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

article[role="main"] {
  float: left;
  width: 62.5%;
}

aside[role="complementary"] {
  float: right;
  width: 31.25%;
}
```

Controlstructures

- ▶ @if
- ▶ @for
- ▶ @each
- ▶ @while

Demo

- ▶ At any time you want you can reproduce the steps i'm gonna show you in the demo by following the commits under <https://github.com/W-i-Z-o/sassDemo> starting with the first one.

Quellen

- ▶ <http://sass-lang.com>
- ▶ <http://thesassway.com>

Are there any questions?

Thank you for your Attention!