

千锋教育 HTML5 讲师：王杨

SASS 和 SCSS

SASS 是 CSS3 的一个扩展，增加了规则嵌套、变量、混合、选择器继承等等。

SCSS 是 SASS 的新语法，是 Sassy CSS 的简写，是 CSS3 语法的超集，也就是说所有有效的 CSS3 样式也同样适合于 SASS。

创建和编译

HBuilder

1. 打开 HBuilder，新建 scss 文件

1.scss

```
@charset "UTF-8";
$blue: blue;
$c: #fff;

html,body {
    background: $blue;
    color: $c;
}
```

2. 鼠标在该文件上右键，编译，生成 css 文件

```
<link rel="stylesheet" href="1.css" type="text/css" />
```

语法

1. 基本用法

1 变量

SASS允许使用变量，所有变量以\$开头。

```
$blue : #1875e7;  
div {  
  color : $blue;  
}
```

如果变量需要镶嵌在字符串之中，就必须需要写在#{ }之中。

```
$side : left;  
.rounded {  
  border-top-#{ $side }-radius: 50px;  
}
```

2 计算功能

SASS允许在代码中使用算式：

```
$var:20;  
body {  
  margin: (14px/2);  
  top: 50px + 100px;  
  right: $var * 10%;  
}
```

3 嵌套

SASS允许选择器嵌套。比如，下面的CSS代码：

```
div h1 {  
  color : red;  
}
```

可以写成：

```
div {
```

```
h1 {  
  color:red;  
}
```

子节点

```
.div1{  
  >ul{  
    border: 10px solid green;  
  }  
}
```

属性也可以嵌套：

```
p {  
  border-color: red;  
}
```

可以写成：

```
p {  
  border: {  
    color: red;  
    width: 10px;  
    style: dashed;  
  }  
}
```

注意，border后面必须加上冒号。

连接：

```
.ab{  
  color:red  
}
```

可以写成：

```
.a{
  &b{
    color:red;
  }
}
```

4 注释

SASS共有两种注释风格。

标准的CSS注释 `/* comment */`，会保留到编译后的文件。

单行注释 `// comment`，只保留在SASS源文件中，编译后被省略。

2. 代码的重用

1 继承

SASS允许一个选择器，继承另一个选择器。比如，现有class1：

```
.class1 {
  border: 1px solid #ddd;
}
```

class2要继承class1，就要使用@extend命令：

```
.class2 {
  @extend .class1;
  font-size:120%;
}
```

2 Mixin

Mixin有点像C语言的宏（macro），是可以重用的代码块。

使用@mixin命令，定义一个代码块。

```
@mixin left {
  float: left;
  margin-left: 10px;
}
```

```
}
```

使用@include命令，调用这个mixin。

```
div {  
  @include left;  
}
```

mixin的强大之处，在于可以指定参数和缺省值。

```
@mixin left($value: 10px) {  
  float: left;  
  margin-right: $value;  
}
```

使用的时候，根据需要加入参数：

```
div {  
  @include left(20px);  
}
```

3. 高级用法

1 条件语句

@if可以用来判断：

```
p {  
  @if 2==2 {  
  
  }@elseif 2>3 {  
  
  }@else {  
  
  }  
}
```

2 循环语句

SASS支持for循环：

```
@for $i from 1 to 10 {  
  .border-#{ $i } {  
    border: #{ $i }px solid blue;  
  }  
}
```

也支持while循环：

```
$i: 6;  
@while $i > 0 {  
  .item-#{ $i } { width: 2em * $i; }  
  $i: $i - 2;  
}
```

each命令，作用与for类似：

```
@each $member in a, b, c, d {  
  .#{ $member } {  
    background-image: url("/image/#{ $member }.jpg");  
  }  
}
```

3 自定义函数

SASS允许用户编写自己的函数。

```
@function double($n) {  
  @return $n * 2;  
}  
  
#sidebar {  
  width: double(5px);  
}
```

(完)