

SCSS 基础语法与使用（导入文件）

（1）@import 用法：导入外部样式

- css 有一个特性，即@import 规则，它允许在一个 css 文件中导入其他 css。然而只有执行到@import 时，浏览器才会去下载其他 css 文件，这样导致页面加载起来特别慢。
- SCSS 也有一个@import 规则，但不同的是，SCSS 的@import 规则在生成 css 文件时就会把相关文件导入进来，这就意味着所有相关的样式被归纳到了同一个 css 文件中，而无需发起额外的下载请求。

（2）@import 用法：只导入部分外部样式

- 当通过@import 把 SCSS 样式分散到多个文件时，你通常只想生成少数几个 css 文件。那些专门为@import 命令而编写的 SCSS 文件，并不需要生成对应的独立 css 文件，这样的 SCSS 文件称为局部文件。
- SCSS 用一个特殊的约定来命名这些文件。即：SCSS 局部文件的文件名以下划线开头。这样，SCSS 就不会在编译时单独编译这个文件输出 css，而只把这个文件用作导入。
- 当你@import 一个局部文件时，还可以不写文件的全名，即省略文件名开头的下划线。举例来说，你想导入 themes/_icons.SCSS 这个局部文件里的变量，你只需在样式表中写@import "themes/ icons";。

（3）默认变量值：一般情况下，你反复声明一个变量，只有最后一处声明有效且它会覆盖前边的值。

示例：

SCSS 样式文件：

```
$link-color:blue;

$link-color:red;

a{

    color:$link-color;

}
```

在上边的例子中，超链接的 color 会被设置为 red。这可能并不是你想要的结果，假如你写了一个可被他人通过@import 导入的 SCSS 库文件，你可能希望导入者可以定制修改 SCSS 库文件中的某些值。使用 SCSS 的 !default 标签可以实现这个目的。它很像 css 属性中!important 标签的对立面，不同的是!default 用于变量，含义是：**如果这个变量被声明赋值了，那就用它声明的值，否则就用这个默认值。**

示例：

SCSS 样式文件：

```
$box-width:400px !default;

.box{

    width: $fancybox-width;

}
```

在上例中 如果用户在导入你的 SCSS 局部文件之前声明了一个\$box-width 变量 , 那么你的局部文件中对\$box-width 赋值 400px 的操作就无效。如果用户没有做这样的声明 , 则\$box-width 将默认为 400px。

(4) 嵌套导入 :

- 跟原生的 css 不同 , SCSS 允许@import 命令写在 css 规则内。这种导入方式下 , 生成对应的 css 文件时 , 局部文件会被直接插入到 css 规则内导入它的地方。

示例 : 有一个名为_blue-project.SCSS 的局部文件 , 如下 :

```
aside{  
  
    background:blue;  
  
    color:white;  
  
}
```

然后把它导入到一个 css 规则内 , 如下所示 :

```
.blue-project{ @import "blue-project" }
```

//生成的结果跟你直接在 blue-project 选择器内写_blue-project.SCSS 文件的 内容完全一样。

```
.blue-project{  
  
    aside{  
  
        background:blue;  
  
        color:#fff;  
  
    }  
  
}
```