Sass @mixin and @include

- description : Sass @mixin and @include
- author : 오션
- email : shlim@repia.com
- lastupdate : 2021-04-08

Ref

Sass @mixin and @include

Sass Mixins

@mixin 지시문을 사용하면 웹 사이트 전체에서 재사용할 CSS 코드를 만들 수 있습니다.

mixin을 사용(포함)할 수 있도록 @include 지시문을 생성합니다.

Defining a Mixin

mixin은 @mixin 지시어로 정의됩니다.

Sass @mixin Syntax

```
@mixin name {
  property: value;
  property: value;
  ...
}
```

하기의 예제는 “important-text”라는 mixin을 생성합니다:

SCSS Syntax

```
@mixin important-text {
  color: red;
  font-size: 25px;
```

```
  font-weight: bold;
  border: 1px solid blue;
}
```

Tip:
Sass의 하이픈 및 밑줄에 대한 팁:하이픈과 밑줄은 동일한 것으로 간주됩니다.
이는 @mixin important-text { } 및 @mixin important_text { }가 동일한 mixin으로 간주된다는 것을 의미합니다!

Using a Mixin

@include 지시문은 mixin을 포함시키기 위해 사용됩니다.

Sass @include mixin Syntax:

```
selector {
  @include mixin-name;
}
```

위에서 생성된 important-text minin을 포함시키기 위해서, 아래와 같은 구문을 포함시깁니다.

SCSS Syntax:

```
.danger {
  @include important-text;
  background-color: green;
}
```

구문 정리 (파일 명: mystyle.scss)

```
@mixin important-text {
  color:red;
  font-size: 25px;
  font-weight: bold;
  border: 1px solid blue;
}

.danger {
  @include importan-text;
  background-color: green;
```

```
}
```

Sass 트랜스파일러는 상기의 예제를 일반 CSS로 변환합니다.

CSS output

```
.danger {
  color: red;
  font-size: 25px;
  font-weight: bold;
  border: 1px solid blue;
  background-color: green;
}
```

mixin은 다른 mixin들도 포함합니다.

SCSS Syntax

```
@mixin special-text {
  @include important-text;
  @include link;
  @include special-border;
}
```

Passing Variables to a Mixin

mixin은 인수(arguments)를 받습니다. 이렇게 하면 변수를 mixin에 전달할 수 있습니다.

인수로 mixin을 정의하는 방법은 다음과 같습니다.

SCSS Syntax

```
/* Define mixin with two argements */
@mixin bordered($color, $width) {
  border: $width solid $color;
}

.myArticle {
  @include bordered(blue, 1px);  // Call mixin with two values
}

.myNotes {
  @include bordered(red, 2px);  // Call mixin with two values
```

```
}
```

인수는 변수로 설정되고, border 속성의 값 (색상 및 너비)으로 사용됩니다.

컴파일 후 CSS는 다음과 같습니다.

CSS Output

```
.myArticle {
  border: 1px solid blue;
}

.myNotes {
  border: 2px solid red;
}
```

Default Values for a Mixin

mixin 변수에 대한 기본 값을 정의할 수도 있습니다.

SCSS Syntax:

```
@mixin bordered($color: blue, $width: 1px) {
  border: $width solid $color;
}
```

그런 다음, mixin을 포함할 때, 변경되는 값만 지정하면 됩니다.

SCSS Syntax

```
.myTips {
  @include bordered($color: orange);
}
```

Using a Minin For Vendor Prefixes

mixin의 또 다른 좋은 용도는 vendor prefixes(공급업체 접두사)입니다.

다음은 변환의 예입니다.

SCSS Sybtax:

```
@mixin transform($property) {
  -webkit-transform: $property;
  -ms-transform: $property;
  transform: $property;
}

.myBox {
  @include transform(rotate(20deg));
}
```

컴파일 후, CSS는 하기와 같습니다.

```
.myBox {
  -webkit-transform: rotate(20deg);
  -ms-transform: rotate(20deg);
  transform: rotate(20deg);
}
```

오션,, Sass, @mixin, and, @include

From:
http://125.132.25.164/dokuwiki/ - **문제를 잘 정의하는 것은 문제를 절반 해결한 것이다. - 2023.12**

Permanent link:
http://125.132.25.164/dokuwiki/doku.php?id=wiki:sass:sass_mixin&rev=1626911947

Last update: **2022/03/10 19:52**