

Sass

Syntactically Awesome Stylesheet

구문적 놀라운 스타일 시트(...)

Sass는?

Syntactically Awesome Stylesheet

CSS전처리기 (Pre-processor)

Sass와 같은 'CSS확장 언어'의 파일을 웹 브라우저에서 해석할 수 있는
css파일로 만들어주는 처리기

Sass 기본구조

Sass basic structure

```
div.container {  
  padding: 15px;  
  margin: 0;  
  
  > p#main-title {  
    font-size: 16px;  
    font-weight: bold;  
  }  
  
  > .fixed {  
    position: fixed;  
    bottom: 10px;  
    right: 10px;  
  }  
}
```

Sass의 출력 스타일 - expanded

Sass output style

Sass

```
div.container {  
  padding: 15px;  
  margin: 0;  
  
  > p#main-title {  
    font-size: 16px;  
    font-weight: bold;  
  }  
  
  > .fixed {  
    position: fixed;  
    bottom: 10px;  
    right: 10px;  
  }  
}
```

CSS

```
div.container {  
  padding: 15px;  
  margin: 0;  
}  
  
div.container p#main-title {  
  font-size: 16px;  
  font-weight: bold;  
}  
  
div.container .fixed {  
  position: fixed;  
  bottom: 10px;  
  right: 10px;  
}
```

Sass의 출력 스타일 - nested

Sass output style

Sass

```
div.container {  
  padding: 15px;  
  margin: 0;  
  
  > p#main-title {  
    font-size: 16px;  
    font-weight: bold;  
  }  
  
  > .fixed {  
    position: fixed;  
    bottom: 10px;  
    right: 10px;  
  }  
}
```

CSS

```
div.container {  
  padding: 15px;  
  margin: 0; }  
div.container p#main-title {  
  font-size: 16px;  
  font-weight: bold; }  
div.container .fixed {  
  position: fixed;  
  bottom: 10px;  
  right: 10px; }
```

Sass의 출력 스타일 - compact

Sass output style

Sass

```
div.container {  
  padding: 15px;  
  margin: 0;  
  
  > p#main-title {  
    font-size: 16px;  
    font-weight: bold;  
  }  
  
  > .fixed {  
    position: fixed;  
    bottom: 10px;  
    right: 10px;  
  }  
}
```

CSS

```
div.container { padding: 15px; margin: 0; }  
div.container p#main-title { font-size: 16px; font-weight: bold; }  
div.container .fixed { position: fixed; bottom: 10px; right: 10px; }
```

Sass의 출력 스타일 - compressed

Sass output style

Sass

```
div.container {  
  padding: 15px;  
  margin: 0;  
  
  > p#main-title {  
    font-size: 16px;  
    font-weight: bold;  
  }  
  
  > .fixed {  
    position: fixed;  
    bottom: 10px;  
    right: 10px;  
  }  
}
```

CSS

```
div.container{padding:15px;margin:0}div.container p#main-title{font-size:16px;font-weight:bold}div.container .fixed{position:fixed;bottom:10px;right:10px}
```

sass-autocompile

atom package

atom 패키지 설치
sass-autocompile

sass-autocompile

atom packages - Settings

Compile on Save

Compile with 'compressed'

Compile with 'compact'

Compile with 'nested'

Compile with 'expanded'

저장시 자동 컴파일

Compressed 방식 파일 생성

Compact 방식 파일 생성

Nested 방식 파일 생성

Expanded 방식 파일 생성

sass-autocompile

atom packages - Settings

상대 경로 지정
'../css/\$1.css'

Sass 문법

Sass syntax

Sass 사용법에 대해 간단히 알아봅니다

주석

Comments

//주석내용

또는

/* 주석내용 */ (CSS기본)
을 사용합니다

주석

Comments

```
// 한 줄 주석은 Sass에서만 지원하며,  
// CSS컴파일 시 나타나지 않습니다.  
div.container {  
  padding: 15px;  
  margin: 0;  
  
  /* 기존에 CSS에서 사용하는 주석 형태는 컴파일 시에도 나타납니다 */  
  p#main-title {  
    font-size: 16px;  
    font-weight: bold;  
  }  
  /*  
    여러 줄 주석은  
    기존 CSS주석 형태로만 사용 가능합니다  
  */  
  .fixed {  
    position: fixed;  
    bottom: 10px;  
    right: 10px;  
  }  
}
```

주석

Comments

```
@charset "UTF-8";
div.container {
    padding: 15px;
    margin: 0;
    /* 기존에 CSS에서 사용하는 주석 형태는 컴파일 시에도 나타나게 됩니다 */
    /*
        여러 줄 주석은
        기존 CSS주석 형태로만 사용 가능합니다
    */
}

div.container p#main-title {
    font-size: 16px;
    font-weight: bold;
}

div.container .fixed {
    position: fixed;
    bottom: 10px;
    right: 10px;
}
```

중첩

Nested

CSS에서의 .container > #page를 뜻합니다

CSS에서의 .container > #page p.section-title을 뜻합니다

```
.container {  
  width: 1100px;  
  margin: 0 auto;  
  
  > #page {  
    width: 800px;  
    border: 1px solid #eee;  
    padding: 10px;  
  
    p.section-title {  
      font-size: 20px;  
      text-align: center;  
    }  
  
    p.section-content {  
      font-size: 12px;  
      color: #999;  
    }  
  }  
}
```

부모 참조 선택자 (&)

Referencing parent selectors

Sass

&은 자신의
부모선택자를
참조합니다

```
#page {  
  width: 800px;  
  border: 1px solid #eee;  
  padding: 10px;  
  
  a {  
    text-decoration: none;  
  
    &:hover {  
      color: red;  
    }  
  
    .link-container & {  
      font-size: 30px;  
    }  
  }  
}
```

& 앞에 선택자가 있을 경우,
해당 선택자 다음에 현재 위치에서의 부모선택자를 참조합니다

CSS

```
#page {  
  width: 800px;  
  border: 1px solid #eee;  
  padding: 10px;  
}  
  
#page a {  
  text-decoration: none;  
}  
  
#page a:hover {  
  color: red;  
}  
  
.link-container #page a {  
  font-size: 30px;  
}
```


중첩 속성

Nested properties

Sass

```
.container {  
  margin: {  
    left: auto;  
    right: auto;  
  }  
}
```

속성 다음에 -(dash)
로 이어지는 속성은
아래로 내려
작성할 수 있습니다

CSS

```
.container {  
  margin-left: auto;  
  margin-right: auto;  
}
```

선택자 상속 (@extend)

Selector inheritance

Sass

```
.btn {  
  background-color: #cdcdcd;  
  font-weight: bold;  
  color: white;  
  padding: 5px 20px;  
}  
.btn-ok {  
  @extend .btn;  
  background-color: #d9edf7;  
}  
.btn-cancel {  
  @extend .btn;  
  background-color: #bbb;  
}
```

CSS

```
.btn, .btn-ok, .btn-cancel {  
  background-color: #cdcdcd;  
  font-weight: bold;  
  color: white;  
  padding: 5px 20px;  
}  
.btn-ok {  
  background-color: #d9edf7;  
}  
.btn-cancel {  
  background-color: #bbb;  
}
```

상속받은 선택자가 .btn의 설정을 함께 공유합니다

상속을 사용하면 같은 속성에서 일부만 바뀐 요소를 쉽게 컨트롤 할 수 있습니다.

대체 선택자 (%)

Placeholder selector

Sass

```
%btn {
  background-color: #cdcdcd;
  font-weight: bold;
  color: white;
  padding: 5px 20px;
}

.btn-ok {
  @extend %btn;
  background-color: #d9edf7;
}

.btn-cancel {
  @extend %btn;
  background-color: #bbb;
}
```

대체선택자로 선언된 구문은 CSS로 해석되지 않습니다

%btn에 지정한 속성은 CSS에 나타나지 않습니다

CSS

```
.btn-ok, .btn-cancel {
  background-color: #cdcdcd;
  font-weight: bold;
  color: white;
  padding: 5px 20px;
}

.btn-ok {
  background-color: #d9edf7;
}

.btn-cancel {
  background-color: #bbb;
}
```

변수 (\$)

Variables

```
$padding: 10px;  
$bg-color: #ececec;  
$title-font-weight: bold;
```

변수명은 \$로 시작합니다

```
div.container {  
  background-color: $bg-color;  
  padding: $padding;  
  margin: 0;
```

자주 쓰이는 color값을 저장합니다

```
p#main-title {  
  font-size: 16px;  
  font-weight: $title-font-weight;  
}
```

중복으로 쓰이는 값에 유용합니다

```
p#sub-title {  
  font-size: 14px;  
  font-weight: $title-font-weight;  
}
```

```
.fixed {  
  position: fixed;  
  padding: $padding;  
  bottom: 10px;  
  right: 10px;  
}
```

변수의 데이터 유형

Variables

| 유형 | 설명 |
|-----------------|---|
| 숫자 (Number) | 정수, 소수, 단위숫자 ex) 1.0, 2.3em, 50px |
| 문자 (String) | 작은따옴표 혹은 큰따옴표로 감싸진 문자 ex) 'images/025.png', "Sass사용법" |
| 색상 (Colors) | 색상 정보 ex) #ff2342, rgb(255,233,210) |
| 참/거짓값 (Boolean) | 참 또는 거짓 값 true, false |
| 리스트 (List) | 공백이나 콤마로 구분해서 나열 2px solid gray, Helvetica, Sans-serif |
| 맵 (Map) | 파이썬의 dict와 유사 ex: {color: white, font-size: 12px} |
| 비어있음 (Null) | 값 초기화시 사용 |

Sass파일 호출

Import

현재 파일 위치를 기준으로 불러옵니다

```
@import 'variables';

%btn {
  background-color: $brand-primary;
  font-weight: bold;
  color: white;
  padding: 5px 20px;
}
```

CSS파일을 불러올 때는 확장자를 입력하여야 하지만, Sass파일을 불러올 때는 확장자를 입력하지 않아도 됩니다.

**_로 시작하는 파일은 CSS파일로 컴파일 되지 않습니다. import시에만 사용하는 파일은 _로 네이밍하세요