



SASS

Podstawy

Problemy w CSS

- Wolny rozwój architektury języka
- Powtórzenia w kodzie
- Utrzymanie jakości kodu

Czym jest SASS

SASS (Syntactically Awesome Style Sheets)

to język, za pomocą którego możemy generować kod CSS łatwiej i szybciej.

Wprowadza wiele udogodnień i stara się uzupełniać braki w architekturze CSS.

Co wprowadza SASS

- Zmienne
- Mechanizmy zagnieżdżania i dziedziczenia
- Mixins
- Operacje na kolorach oraz operacje matematyczne
- ...
- dodatkowe frameworki np. COMPASS

Instalacja i dokumentacja SASS

Witryna projektu:

<http://sass-lang.com/install>

SASS jest napisany w języku RUBY i do działania wymaga środowiska RUBY oraz doinstalowania odpowiedniego pakietu

- Linux: `sudo su -c "gem install sass"`
- Windows: `gem install sass`

Jak działa SASS



Składnia SASS

```
$font-stack      : Helvetica, sans-serif  
$primary-color   : #333
```

```
body  
  font   : 100% $font-stack  
  color  : $primary-color
```

plik.sass

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

plik.css

Składnia SCSS

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

plik.scss

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

plik.css

Preprocessing

```
sass źródło.sass wynik.css
```

Istnieje możliwość ustawienia obserwacji (watch) zmian w plikach źródłowych i ich automatyczny preprocessing po dokonaniu zmian.

Automatyczny preprocessing jest wspierany np. w środowiskach JetBrains

Zmienne

```
$font-stack      : Helvetica, sans-serif;  
$font-size       : 16px;  
$font-color      : #333;
```

```
div {  
  font   : $font-size $font-stack;  
  color  : $font-color;  
}
```

```
span {  
  font   : $font-size Arial, sans-serif;  
  color  : $font-color;  
}
```

plik.scss

```
div {  
  font: 16px Helvetica, sans-serif;  
  color: #333;  
}
```

```
span {  
  font: 16px Arial, sans-serif;  
  color: #333;  
}
```

plik.css

Zagnieżdżanie

```
.naglowek-strony {  
  nav {  
    ...;  
    ...;  
  }  
  
  .logo {  
    ...;  
  }  
}
```

plik.scss

```
.naglowek-strony nav {  
  ...;  
}  
  
.naglowek-strony .logo {  
  ...;  
}
```

plik.css

Import

Umieszczenie pliku o nazwie poprzedzonej znakiem podkreślenia np: `"_plik.scss"` poinformuje preprocesor o tym, aby nie generował z niego pliku CSS

Możliwe jest natomiast zawarcie deklaracji z takiego pliku w innym pliku poprzez polecenie `@import`

```
@import '_jakisPlik.scss'
```

```
// dalszy kod
```

```
.naglowek-strony {...}
```

Mixins

Generowanie fragmentów kodu

```
@mixin nazwa-mixina($radius) {  
    -webkit-border-radius: $radius;  
    -moz-border-radius: $radius;  
    -ms-border-radius: $radius;  
    border-radius: $radius;  
}  
  
.klasa1 { @include nazwa-mixina(10px); }  
.klasa2 { @include nazwa-mixina(20px); }
```

Rozszerzanie

```
.komunikat {  
    padding: 20px 10px;  
    color: #aaa;  
}  
  
.blad {  
    @extend .komunikat;  
    color: #f00;  
}
```

Operator

Wyliczenia bazujące na operatorach i jednostkach **+**, **-**, *****, **/**, **%**

```
$zmienna = 200px;  
  
.klasa1 {  
    width: 600px - $zmienna;  
}
```

... i dużo więcej ...

```
@each $animal in puma, sea-slug, egret, salamander {  
  
  .#{ $animal }-icon {  
    background-image: url('/images/#{ $animal }.png');  
  }  
  
}
```




SASS

Podstawy ... koniec