

# Les préprocesseurs CSS avec SASS

Créé par Jean-Baptiste Lavis ( [@jlavis](#) )

# C'est quoi un préprocesseur CSS?

Un outil qui génère dynamiquement des fichiers CSS

**Objectif:** apporter plus de flexibilité à l'écriture des fichiers CSS  
Ceux-ci ont tendance à devenir complexe et dur à maintenir

Les plus utilisés aujourd'hui:

- **SASS**
- LESS
- Stylus

# Le SASS

“ Le but n’est pas de complexifier une pratique et un processus simple de création de sites web (HTML/CSS), mais d’offrir aux développeurs web des outils de développement plus avancés. C’est pourquoi la solution n’a pas été de substituer le CSS, mais plutôt d’automatiser et de dynamiser sa rédaction.

-- <https://nixa.ca/fr/blog/pourquoi-adopter-sass/>

”

Pour expérimenter rapidement:

- Pour expérimenter en ligne: <https://www.sassmeister.com/>
- Une extension VS Code avec un rechargement automatique (live reload): Live Sass Compiler

# Syntaxes du SASS

Deux syntaxes distinctes:

- SASS: Transcription nécessaire du CSS au SASS

```
.info  
  padding: 10px  
  width: 300px
```

- **SCSS**: Depuis SASS 3.0, devenu le format par défaut.  
Beaucoup plus proche de l'écriture du CSS

```
.info {  
  padding: 10px;  
  width: 300px;  
}
```

# Format du SCSS

Les fichiers s'écrivent avec l'extension `.scss`

Architecture simple avec du SCSS:

```
css/  
index.html  
scss/  
  main.scss
```

# Lignes de commande

```
# Generate style.css in css folder from main.scss in scss folder  
sass scss/main.scss css/style.css
```

```
# After each save on .scss file, relaunch the generation  
sass --watch main.scss css/style.css
```

```
# Generate all files in scss folder by css files with same name in css folder  
# SCSS files must not start with an underscore (_)  
sass --watch scss:css
```

```
# Generate a minified css file  
sass scss/main.scss css/style.min.css --style compressed
```

# Fonctionnalités de SASS

## Les partials

Fichiers dont le nom commence par un underscore `_`

Exemple: `_nav.scss`

Permet de découper en plusieurs fichiers son CSS: plus facile à maintenir.

Pour les importer, utilisation de la direction `@import`

# Les partials: exemple

```
/* _nav.scss */  
.nav { color: red;}
```

```
/* main.scss */  
@import 'nav';  
h1 { font-size: 2rem; }
```

```
/* Generated CSS */  
.nav { color: red;}  
h1 { font-size: 2rem; }
```



# Les maths

Permet de profiter des formules arithmétiques et opérations mathématiques comme  $+$ ,  $-$ ,  $*$ ,  $/$ :

```
.picture {  
  float: left;  
  width: 600px / 960px * 100%;  
}
```

# Les variables

Emplacement pour stocker une information que l'on souhaite rapidement réutiliser plusieurs fois

Donner un nom à la variable, et précéder du symbole `$`

```
/* SCSS code */
$primary-color: #333;
.info {
  color: $primary-color;
}
h3 {
  color: $primary-color;
  margin: 1rem 0;
}
```

```
/* CSS generation */
.info {
  color: #333;
}

h3 {
  color: #333;
  margin: 1rem 0;
}
```

# L'Imbrication

Les règles de style peuvent être imbriquées pour améliorer la modularité de ses éléments.

```
/* SCSS code */
.navbar {
  margin: 10px;

  ul {
    list-style-type: none;
  }

  li {
    margin: 1rem;
  }
}
```

```
/* CSS generation */
.navbar {
  margin: 10px;
}

.navbar ul {
  list-style-type: none;
}

.navbar li {
  margin: 1rem;
}
```

## Le sélecteur parent &

Le symbole `&` est un raccourci qui récupérera le nom de son sélecteur parent.

```
/* SCSS code */  
.navbar {  
  &.is-mobile {  
    display: flex;  
  }  
  
  &:hover {  
    background-color: red;  
  }  
}
```

```
/* CSS generation */  
.navbar.is-mobile {  
  display: flex;  
}  
  
.navbar:hover {  
  background-color: red;  
}
```

# Utilisation de BEM en SCSS

```
/* --- SCSS code --- */
```

```
/* Block */
```

```
.box {  
  margin: 1rem;  
}
```

```
/* Element */
```

```
&__title {  
  font-size:1.5rem;  
}
```

```
/* Modifier */
```

```
&--featured {  
  color: blue;  
}  
}
```

```
/* --- CSS generation --- */
```

```
/* Block */
```

```
.box {  
  margin: 1rem;  
}
```

```
/* Element */
```

```
.box__title {  
  font-size: 1.5rem;  
}
```

```
/* Modifier */
```

```
.box--featured {  
  color: blue;  
}
```

# Les mixins

## Répéter de multiples propriétés

```
/* SCSS */  
@mixin circleBorder {  
    border-radius: 50%;  
    border: 3px solid gray;  
}  
.box {  
    @include circleBorder;  
}
```

```
/* CSS */  
.box {  
    border-radius: 50%;  
    border: 3px solid gray;  
}
```

# Rendre intelligent le CSS grâce aux fonctions

```
/* SCSS */
@mixin colorize($color) {
  color: $color;
  border: 1px solid $color;
}
.box {
  @include colorize(#FF0000);
}
```

```
/* CSS */
.box {
  color: #FF0000;
  border: 1px solid #FF0000;
}
```

# Architectures SASS

## Pour un simple site avec très peu de pages

```
_base.scss  
_layout.scss  
_components.scss  
main.scss
```

- Base: Contient les resets, variables, mixins et autres utilitaires
- Layout: Contient le CSS propre à la disposition (ex: header, footer)
- Composants: Tous les éléments réutilisables comme les boutons, blocks BEM etc...
- Main: Uniquement les imports des fichiers précédents



# Architecture simple mais avec plus d'éléments

```
_base.scss  
layout/  
  _grid.scss  
  _header.scss  
  _sidebar.scss  
  _forms.scss  
components/  
  _buttons.scss  
  _carousel.scss  
main.scss
```

# Pattern 7-1 pour gros projet

```
sass/  
|- abstracts/ (or utilities/)  
|   |- _variables.scss    // Sass Variables  
|   |- _functions.scss    // Sass Functions  
|   |- _mixins.scss       // Sass Mixins  
|  
|- base/  
|   |- _reset.scss        // Reset/normalize  
|   |- _typography.scss   // Typography rules  
|  
|- components/ (or modules/)  
|   |- _buttons.scss      // Buttons  
|   |- _carousel.scss     // Carousel  
|   |- _slider.scss       // Slider  
|  
|- layout/  
|   |- _navigation.scss   // Navigation  
|   |- _grid.scss         // Grid system
```

```
|   | - _header.scss           // Header
|   | - _footer.scss          // Footer
|   | - _sidebar.scss         // Sidebar
|   | - _forms.scss           // Forms
|
| - pages/
|   | - _home.scss            // Home specific styles
|   | - _about.scss           // About specific styles
|   | - _contact.scss         // Contact specific styles
|
| - themes/
|   | - _theme.scss           // Default theme
|   | - _admin.scss           // Admin theme
|
| - vendors/
|   | - _bootstrap.scss       // Bootstrap
|   | - _jquery-ui.scss       // jQuery UI
|
| - main.scss                 // Main Sass file
```