



_ Le sass

SOMMAIRE

Présentation	3
Principe de base	4
Installation sous VsCode	5
Utilisation	6
Aperçu sans compilation	7
Aperçu après compilation	8
Mise en oeuvre	9
Ecriture imbriquée	9
Écriture non imbriquée	10

Utiliser des variables en scss	11
Utiliser des mixins en scss	12
Créer des composants en scss	16
Les feuilles partielles	16

Présentation

Sass est le langage d'extension CSS de niveau professionnel le plus mature, le plus stable et le plus puissant au monde.

Sass (Syntactically Awesome StyleSheets), est un préprocesseur CSS*. Il ajoute des fonctionnalités qui ne sont pas disponibles à l'aide de la syntaxe CSS de base.

Sass permet aux développeurs de simplifier et de gérer plus facilement les feuilles de style de leurs projets

. *En informatique, un préprocesseur est un programme qui traite ses données d'entrée pour produire une sortie utilisée comme entrée d'un autre programme.

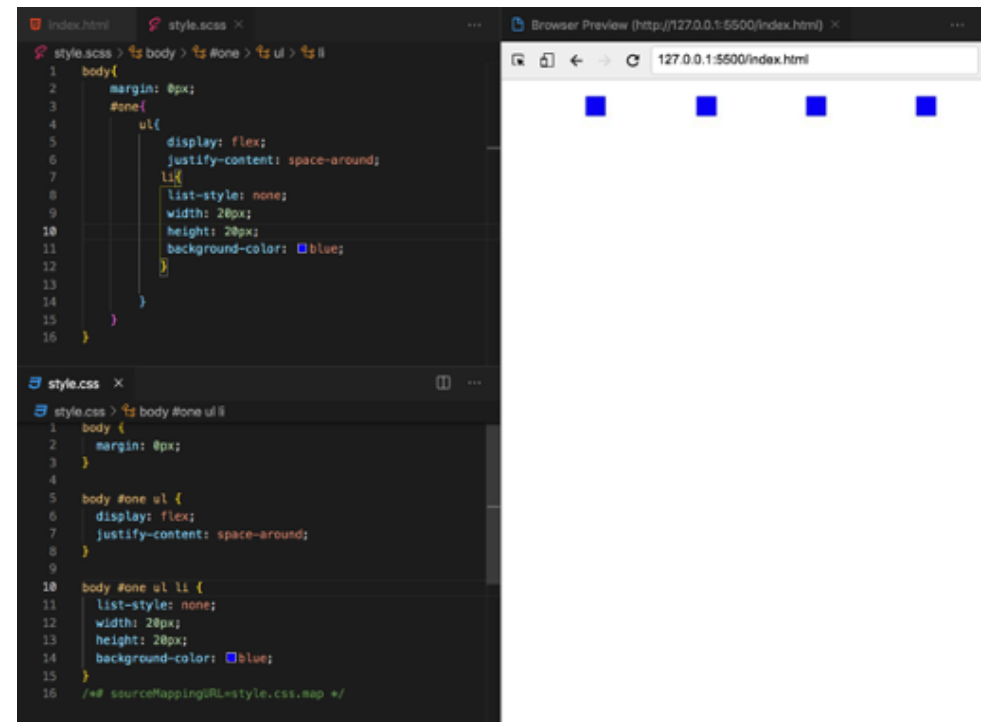


Principe de base

L'imbrication des règles est la principale caractéristique de SASS. Grâce à elle, l'écriture de votre code CSS est beaucoup plus rapide, simple et compréhensible.

L'imbrication vous aidera à avoir un code propre, logique et bien organisé, qui devrait être plus facile à entretenir au cours du temps.

Grâce à cette méthode vous pourrez également créer votre bibliothèque de composants scss.



Installation sous VsCode

Les navigateurs ne comprennent pas le code Sass. Pour utiliser Sass, nous allons devoir utiliser un compilateur dont le rôle va être de traduire les fichiers .sass ou .scss (fichiers Sass) en fichiers .css (fichiers CSS classiques) compréhensibles par le navigateur.

De nombreuses applications proposent une prise en charge et la compilation des fichiers Sass. C'est notamment le cas de Visual Studio Code.

Une extension nommée « Live Sass Compiler » va, comme son nom l'indique, compiler le code Sass en live en code CSS. Cherchez cette extension dans la barre de recherche et installez-la.

[Télécharger Live Sass Compiler](#)



- Les navigateurs ne comprennent pas le code Sass, c'est la raison pour laquelle nous installons Live Sass. Dès que vous aurez créé un fichier style.scss, Live Sass en créera une copie au format css. Ce qui veut également dire que chaque modification effectuée sur le fichier scss d'origine sera prise en compte en temps réel sur le fichier de sortie .css.
- Vous devez donc linker comme d'habitude votre fichier style (`<link rel="stylesheet" href="/style.css">`)
- Lors du transfert des fichiers de votre site sur le serveur, uploadez uniquement le fichier CSS.

Utilisation

Au démarrage de VsCode activez le plug-in **Watch Sass**



- Créer un fichier index.html
- Linkez la feuille de style "style.css" même si le fichier n'est pas encore créer
- Créer un fichier "style.scss"
- Enregistrer le fichier "style.scss"



En regardant la fenêtre de terminal vous pouvez lire ce code qui vous indique le fichier style.css a été généré.

```
Change Detected...

style.scss
-----

Generated :

/Users/alex/Desktop/test sass/style.css

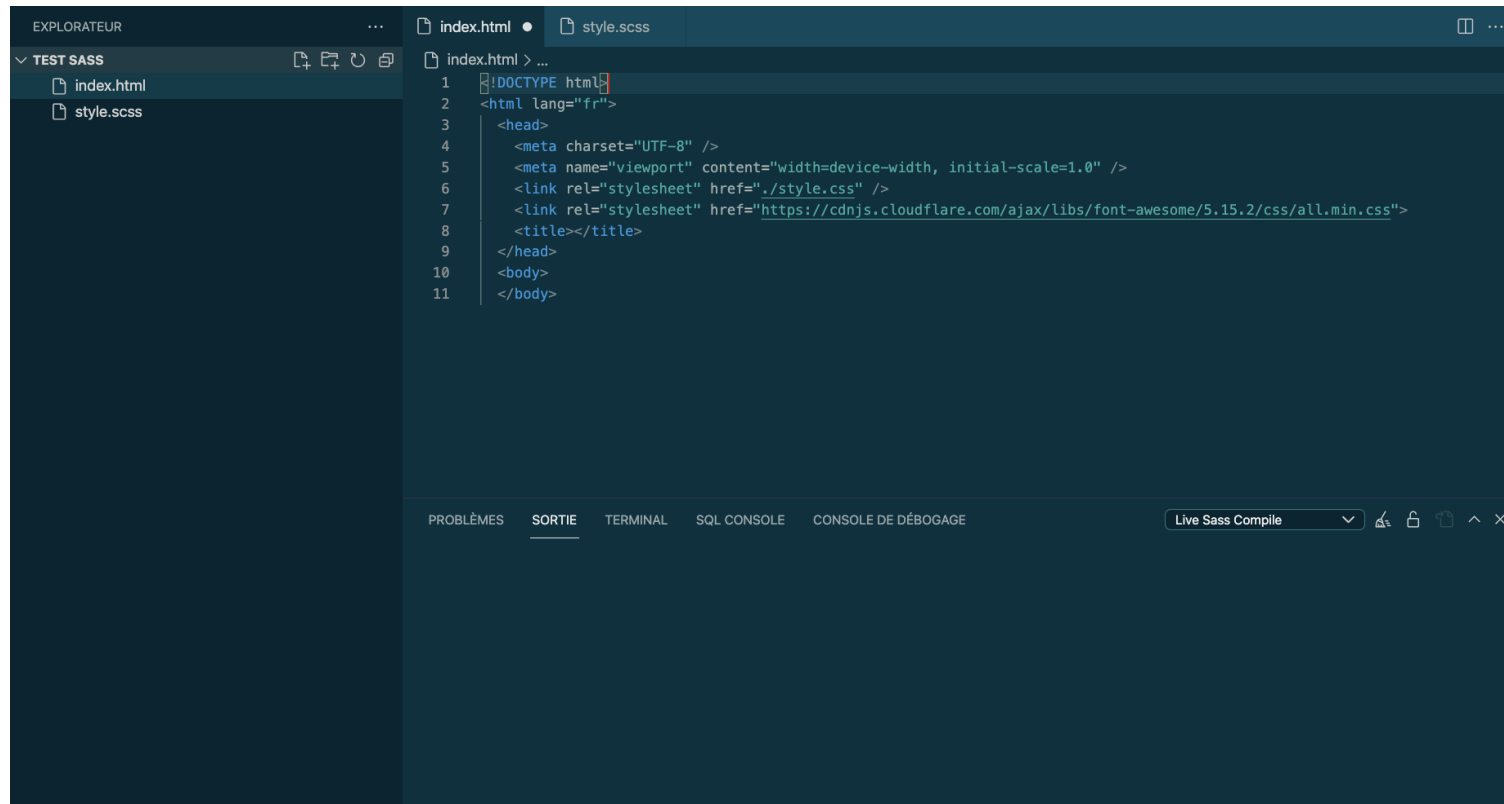
/Users/alex/Desktop/test sass/style.css.map

-----

Watching...

-----
```

Aperçu sans compilation



The screenshot shows a code editor with a dark theme. On the left, the 'EXPLORATEUR' (Explorer) pane shows a project named 'TEST SASS' containing two files: 'index.html' and 'style.scss'. The 'index.html' file is selected and its content is displayed in the main editor. The HTML code includes a DOCTYPE declaration, a language attribute, a head section with meta tags for charset and viewport, and two link tags for stylesheets. The first link tag points to './style.css', and the second points to a CDN for Font Awesome. The 'style.scss' file is listed in the explorer but its content is not visible. At the bottom of the editor, the 'SORTIE' (Output) pane is active, showing a 'Live Sass Compile' button and some icons, indicating that the SCSS file has not been compiled into the CSS file.

```
1 <!DOCTYPE html>
2 <html lang="fr">
3   <head>
4     <meta charset="UTF-8" />
5     <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6     <link rel="stylesheet" href="./style.css" />
7     <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.2/css/all.min.css">
8   </head>
9   <body>
10
11 </body>
```

figure sans avoir compilé le style.scss

Aperçu après compilation

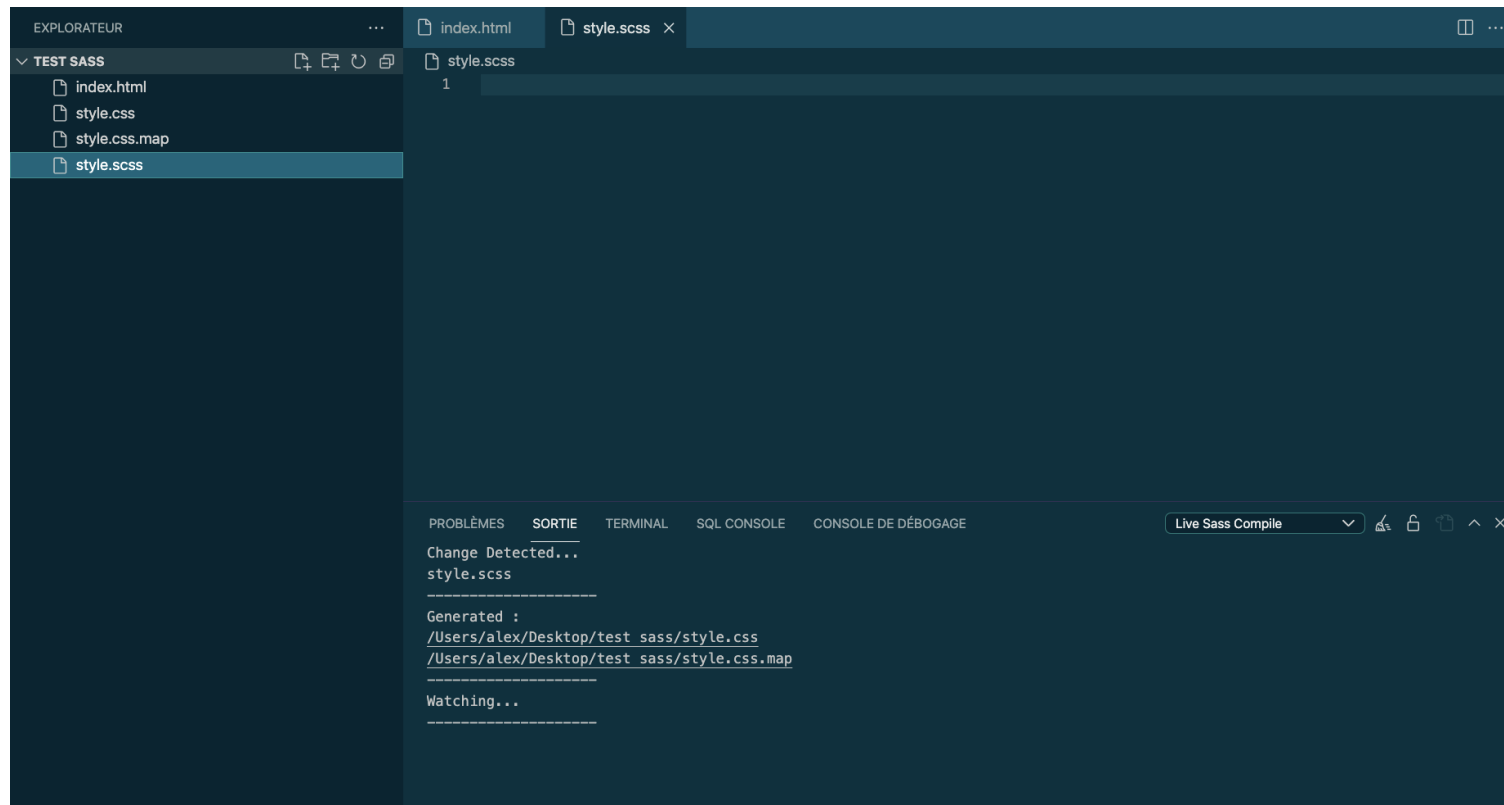


Figure après avoir compilé le style.scss

Mise en oeuvre

Ecriture imbriquée

Le principe du Sass repose sur l'imbrication des secteurs.

L'imbrication de sélecteurs est une fonctionnalité des préprocesseurs CSS qui permet d'emboîter des sélecteurs à l'intérieur d'autres sélecteurs pour créer des raccourcis d'écriture.

```
// Ecriture scss  
  
body{  
  
  section{  
  
    display: flex;  
  
    justify-content: space-around;  
  
    .rouge{  
  
      width: 200px;  
  
      height: 200px;  
  
      background-color: red;  
  
    }  
  
  }  
  
}
```

Figure de code imbriqué

Écriture non imbriquée

```
/* Ecriture css */

body section {

  display: flex;

  justify-content: space-around; }

body section .rouge {

  width: 200px;

  height: 200px;

  background-color: red; }

/*# sourceMappingURL=style.css.map */
```

Figure de code imbriqué

Utiliser des variables en scss

Une variable Sass est un conteneur pour une valeur. L'idée est de lier un nom à une valeur puis d'utiliser ensuite ce nom à la place de la valeur dans le code.

Par exemple : votre site internet utilise 2 couleurs; le noir et le rouge

Au lieu d'écrire pour chaque élément les codes hexadécimaux #f00020 pour le rouge et #000000 pour le noir on va créer un fichier contenant les variables.

```
@import 'variables';
// Ecriture scss
body{
  section{
    display: flex;
    justify-content: space-around;
    .rouge{
      width: 200px;
      height: 200px;
      background-color: $cprim;
    }
  }
}
```

Fichier style.scss

```
$cprim :#f00020;
$csec: #000000;
```

Fichier _variables.scss



Un fichier “_variables.scss” est créé. Notez l'underscore (_)précédent le nom du fichier. Il évite que ce fichier soit converti en css. En effet, le contenu du code _variables est intégré dans le fichier style.scss grâce à la commande `@import 'variables' ;`

Utiliser des mixins en scss

Une mixin est un morceau de code paramétrable et réutilisable n'importe où dans votre fichier SASS. Comme pour une fonction, une mixin peut prendre des paramètres.

Par exemple : votre site internet comprend 2 styles de paragraphe différent, créer des mixins vous évite de réécrire l'ensemble des règles css.

```
<!DOCTYPE html>

<html lang="fr">

  <head>

    <meta charset="UTF-8" />

    <meta name="viewport" content="width=device-width, initial-scale=1.0" />

    <link rel="stylesheet" href="./style.css" />

    <link rel="preconnect" href="https://fonts.googleapis.com">

<link rel="preconnect" href="https://fonts.gstatic.com" crossorigin>

<link href="https://fonts.googleapis.com/css2?family=Libre+Baskerville&family=Roboto:wght@100&display=swap" rel="stylesheet">

    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/5.15.2/css/all.min.css">

    <title></title>

  </head>

  <body>

    <section id="une">
```

```

    <article><p class="rouge">Lorem ipsum dolor sit amet, consectetur adipisicing elit. Tenetur provident repellat minus,
eligendi inventore reprehenderit illum dignissimos vero est voluptate. Voluptate saepe iusto architecto temporibus. Possimus quae sunt
fugit. Nisi!</p></article>

    <article><p class="bleu">Lorem ipsum dolor sit amet, consectetur adipisicing elit. Tenetur provident repellat minus, eligendi
inventore reprehenderit illum dignissimos vero est voluptate. Voluptate saepe iusto architecto temporibus. Possimus quae sunt fugit.
Nisi!</p></article>

</section>

<section id="deux">

    <article><p class="rouge">Lorem ipsum dolor sit amet, consectetur adipisicing elit. Tenetur provident repellat minus, eligendi
inventore reprehenderit illum dignissimos vero est voluptate. Voluptate saepe iusto architecto temporibus. Possimus quae sunt fugit.
Nisi!</p></article>

    <article><p class="bleu">Lorem ipsum dolor sit amet, consectetur adipisicing elit. Tenetur provident repellat minus, eligendi
inventore reprehenderit illum dignissimos vero est voluptate. Voluptate saepe iusto architecto temporibus. Possimus quae sunt fugit.
Nisi!</p></article>

</section>

</body>

```

Fichier index.html

```
$cprim :#f00020;
$csec: #000000;
$cter: #003c8b;
```

fichier _variables.scss

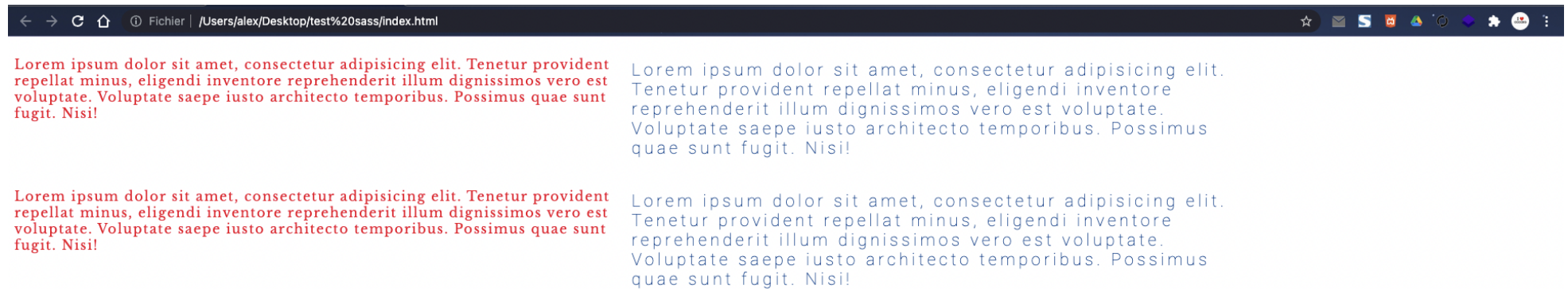
```
@mixin textrouge{
    font-family: 'Libre Baskerville',
    serif;
    font-size: 1em;
    color: $cprim;
    letter-spacing: 0.1em;
}

@mixin textbleu{
    font-family: 'Roboto', sans-serif;
    font-size: 1.3em;
    color: $cter;
    letter-spacing: 0.2em;
}
```

fichier _mixins.scss

```
@import 'variables';
@import 'mixins';
// Ecriture scss
body{
    section{
        display: flex;
        article{
            width: 40%;
            display: flex;
            justify-content: space-around;
            .rouge{
                @include textrouge;
            }
            .bleu{
                @include textbleu;
            }
        }
    }
}
```

fichier _style.scss



Visuel page index.html dans le navigateur

Créer des composants en scss

Les feuilles partielles

Durant le processus de développement, il est souvent intéressant de diviser son code en plusieurs fichiers, pour mieux s'organiser.



Grâce à la directive `@import`, plutôt que de laisser le navigateur charger une à une les feuilles de styles importées, il les importe dès à la compilation. Ainsi, on a plusieurs fichiers SCSS, pour une meilleure organisation, mais on n'a qu'un fichier CSS, pour plus de performance.

```
sass
├─ partials
│   ├── _nav.scss
│   ├── _variables.scss
│   ├── _mixins.scss
│   ├── _contact.scss
│   ├── _header_footer.scss
│   └── _reset.scss
└─ style.scss
```



Avec Sass, la directive `@import` permet d'importer durant la compilation le contenu d'un fichier SCSS dans un autre fichier SCSS.

Il est inutile de préciser l'extension du fichier importé à Sass.

Une feuille partielle ou partial, dont le nom commence par un underscore, est un fichier qui a uniquement vocation à être importé dans d'autres feuilles de styles. Aucun fichier CSS n'est généré pour lui à la compilation.