

# Débuter avec Sass et Compass

<b>1. INSTALLATION SOUS WINDOWS .....</b>	<b>2</b>
<b>2. ENVIRONNEMENTS DE DEVELOPPEMENT .....</b>	<b>3</b>
<b>3. CREATION D'UNE FEUILLE DE STYLE SASS .....</b>	<b>3</b>
A. COMPILATION EN LIGNE DE COMMANDE AVEC L'INVITE DE COMMANDE RUBY .....	3
<i>Compilation automatique</i> .....	4
B. GUIDE « LANGAGE » SASS .....	5
<i>Variables</i> .....	5
<i>Nested</i> .....	6
<i>Partials</i> .....	7
<i>Mixins</i> .....	7
<i>Extend</i> .....	7
<i>Fonctions fournies par Sass</i> .....	8
<i>@functions, @if, @for, each</i> .....	8
<b>4. CREATION D'UN PROJET COMPASS .....</b>	<b>9</b>
A. TROUVER DES BIBLIOTHEQUES SASS .....	9
B. COMPILATION D'UN PROJET COMPASS .....	10

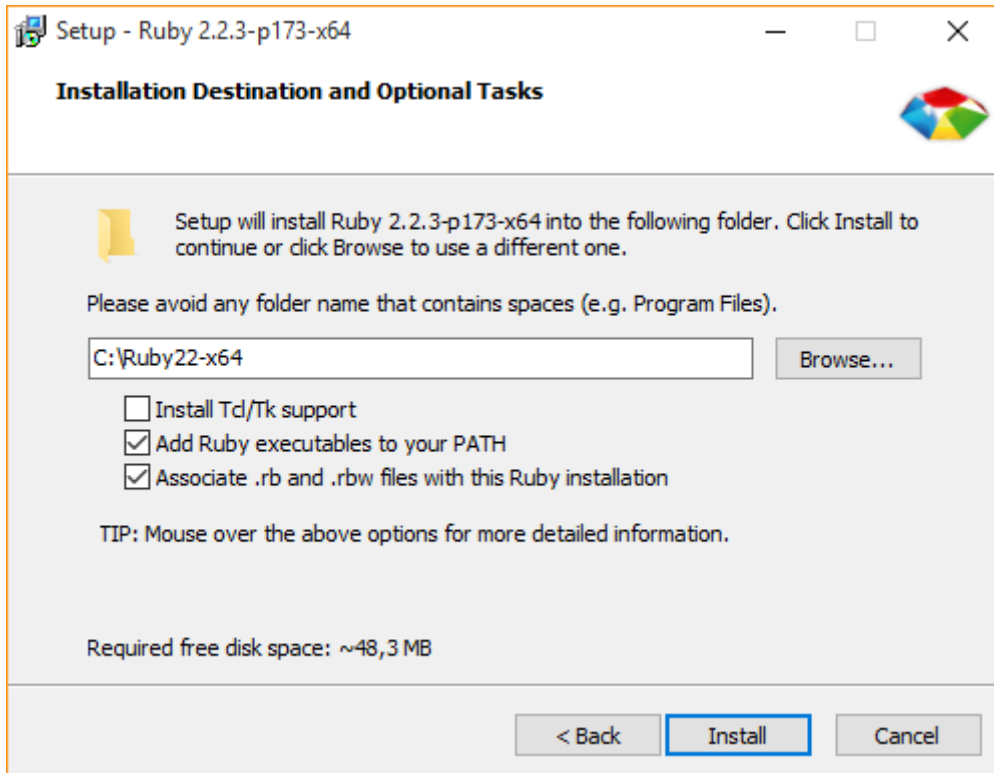
**Sass** est une extension de CSS3 (écrit en Ruby) permettant de définir des variables, mixins, de profiter de l'héritage... **Compass** est un Framework utilisant Sass. Le but de tout ça étant (normalement) d'écrire moins et gagner du temps.

## 1. Installation sous Windows

Pour Mac et Linux : <http://sass-lang.com/install>

1. Télécharger (version 2.2.3 x64 actuellement) et installer Ruby

<http://rubyinstaller.org/>



2. Puis ouvrir une invite de commande, naviguer jusqu'au dossier d'installation de Ruby (cd ...)

Et saisir pour installer Sass

```
gem install sass
```

Puis pour Compass

```
gem install compass
```

```
C:\Users\romag> cd C:\Ruby22-x64\bin
C:\Ruby22-x64\bin> gem install sass
Fetching: sass-3.4.20.gem (100%)
Successfully installed sass-3.4.20
Parsing documentation for sass-3.4.20
Installing ri documentation for sass-3.4.20
Done installing documentation for sass after 1
1 gem installed
C:\Ruby22-x64\bin> gem install compass
Fetching: chunky_png-1.3.5.gem (100%)
Successfully installed chunky_png-1.3.5
```

## 2. Environnements de développement

Compass.app (mac, linux, pc) <http://compass.kkbox.com/>

Il en existe d'autres open source, payants avec plus ou moins de fonctionnalités

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

## 3. Création d'une feuille de style Sass

Créer un fichier avec l'extension \*.scss avec son éditeur de texte préféré.

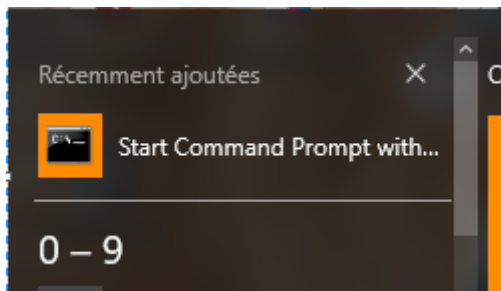
Ajouter quelques règles. Exemple deux variables et utilisation

```
$font-stack: Helvetica, sans-serif;
$primary-color: #f00;

body {
  font: 100% $font-stack;
  color: $primary-color;
}
```

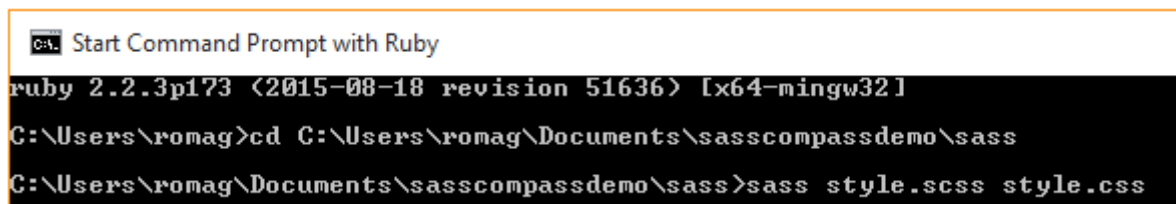
### a. Compilation en ligne de commande avec l'invite de commande Ruby

Une invite de commande Ruby a été ajoutée à la liste des applications







Naviguer jusqu'au dossier contenant une feuille de style sass (extension .scss) puis pour compiler

```
sass style.scss style.css
```



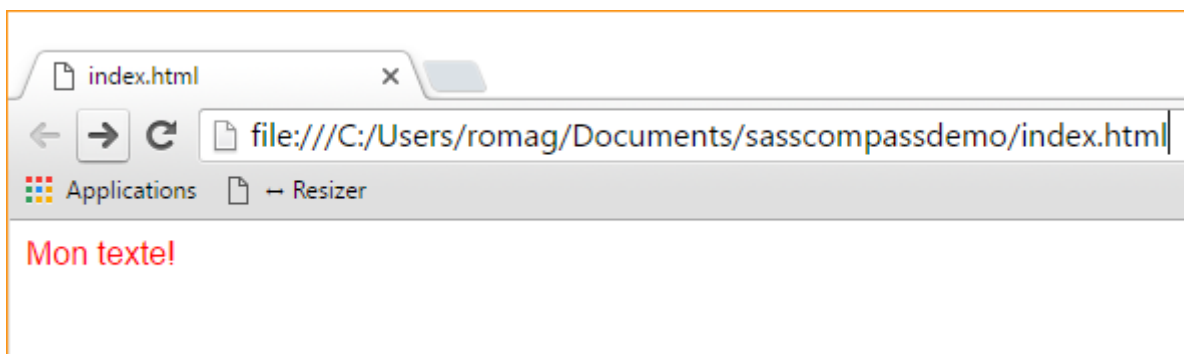
La feuille de style correspondante (nommée ici « style.css ») est générée

-  .sass-cache
-  style.css
-  style.css.map
-  style.scss

On peut désormais utiliser cette feuille de style dans une page html par exemple.

```
<!DOCTYPE html>
<html>
<head>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <title></title>
  <meta charset="utf-8" />
  <link href="sass/style.css" rel="stylesheet" />
</head>
<body>
  Mon texte!
</body>
</html>
```

Et tester dans le navigateur



### Compilation automatique

Compiler manuellement risque vite d'être redondant.

Dans l'invite de commande Ruby

```
sass --watch style.scss :style.css
```

Feuille SASS à suivre

Feuille CSS mise à jour

```
C:\Users\romag\Documents\sasscompassdemo\sass>sass --watch style.scss:style.css
>>> Sass is watching for changes. Press Ctrl-C to stop.
>>> Change detected to: style.scss
      write style.css
      write style.css.map
```

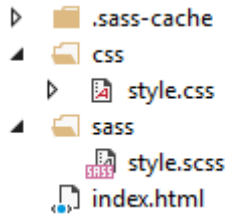
**CTRL + C pour arrêter de la compilation automatique.**

Ainsi si on change par exemple la couleur de la font (\$primary-color) automatiquement la feuille de style CSS sera mis à jour. On pourra donc immédiatement observer les changements dans le navigateur.

Note on peut également demander à observer un dossier. Exemple un dossier sass et mettre à jour un dossier css. Naviguer jusqu'au dossier parent

```
sass --watch sass :css
```

```
C:\Users\romag\Documents\sasscompassdemo>sass --watch sass:css
>>> Sass is watching for changes. Press Ctrl-C to stop.
      write css/style.css
      write css/style.css.map
```



## b. Guide « langage » Sass

### Guide et Référence

#### Variables

Cela peut être une couleur (hexadécimale avec « #... », un nom comme « red »), une taille (px, em, %, rem) ou même une combinaison de valeurs (exemple 1px solid #777, une liste de fonts , etc.).

```
@import url(https://fonts.googleapis.com/css?family=Roboto:400,300,500,700,900);
@import url(https://fonts.googleapis.com/css?family=Open+Sans:400,300,600);
```

```
$color-highlight:orange;
$color-main:#777;
```

```
$font-size:15px;
```

```
$font-highlight: Roboto, Helvetica, sans-serif;
$font-size-main: 'Open Sans', Helvetica, sans-serif;
```

```
html{
}
```

```
body {
  font-size: $font-size;
  color: $color-main;
  font-family:$font-highlight;
}
```

```
h1,h2,h3,h4,h5,h6{
  font-family:$font-highlight;
  color:$color-highlight;
}
```

Variables globales  
déclarées en haut de la  
feuille de styles.

Une variable peut  
également être locale  
seulement à un style.

On peut faire également des opérations mathématiques.

```
$font-size-main:15px;
```

```
.smaller{
  font-size:$font-size-main - 5px;
}
```

## Injection de chaîne

```
$images-folder: "../images/";

body {
  background-image: url("#{ $images-folder }image_1.jpg");
}
```

Ce qui donnera  
« ../images/image\_1.jpg »

## Autre exemple

```
$base-name: p;

#{$base-name}
{
  background-color: red;
}
```

Les paragraphes (balise « p »)  
seront avec fond rouge.  
On peut aussi appliquer à une classe  
ou un identifiant

## Nested

Le logo d'un site pour lequel on cache le « span »

## HTML

```
<a href="#" class="logo">
  <span>Nom de l'entreprise</span>
</a>
```

## SCSS

```
a.logo {
  display: block;
  background: url(..images/logo.png) no-repeat 0 0;
  background-size: contain;
  width: 166px;
  height: 60px;

  span {
    display: none;
  }
}
```

Au lieu de saisir « .logo span » on inclut le code  
dans le « parent »

## CSS

```
a.logo {
  display: block;
  background: url(..images/logo.png) no-repeat 0 0;
  background-size: contain;
  width: 166px;
  height: 60px;
}
a.logo span {
  display: none;
}
```

Autre exemple avec les liens (balise « a »)

### SCSS

```
a{
  text-decoration: none;
  &:hover{
    text-decoration: underline;
  }
}
```

On met « & » pour  
référencer le parent

### CSS

```
a {
  text-decoration: none;
}
a:hover {
  text-decoration: underline;
}
```

### Partials

On a plusieurs feuilles SCSS. On importe celles-ci (code en haut de la feuille SCSS)

```
@import "autresfichier.scss";
```

### Mixins

Placées dans un fichier à part

```
@mixin bgImage($image-url) {
  background-image: url($image-url);
  background-repeat: no-repeat;
  background-size: cover;
}
```

... que l'on importe. Puis on utilise les mixins créés avec @include.

```
@import "mixins.scss";

body {
  @include bgImage('../images/image_1.jpg');
}
```

### Extend

```
.btn{
  display: inline-block;
  padding: 8px 12px;
  border-radius: 3px;
  color: #fff;
  cursor:pointer;
}
```

Une classe de  
base

```
.btn-error{  
  @extend .btn;  
  background-color: red;  
}
```

On change juste la couleur  
de fond pour les classes  
qui l'étendent

```
.btn-warning{  
  @extend .btn;  
  background-color: orange;  
}
```

### Fonctions fournies par Sass

Sass dispose de nombreuses fonctions de couleur, opacité, de mise en forme de texte, etc. cliquer sur chaque fonction pour le détail de son utilisation.

<http://sass-lang.com/documentation/Sass/Script/Functions.html>

Exemple : rend transparent

```
p{  
  background-color: transparentize($color-main,.5);  
}
```

@functions, @if, @for, each

[http://sass-lang.com/documentation/file.SASS\\_REFERENCE.html#if](http://sass-lang.com/documentation/file.SASS_REFERENCE.html#if)



## 4. Création d'un projet Compass

Ouvrir l'invite de commande Ruby. Naviguer jusqu'au dossier qui contiendra le projet Compass (cd ...)

Puis saisir la commande

```
compass create compassdemo
```

Nom du projet

```

C:\Users\romag>cd "C:\Users\romag\Documents\compass projects"
C:\Users\romag\Documents\compass projects>compass create compassdemo
directory compassdemo/
directory compassdemo/sass/
directory compassdemo/stylesheets/
  create compassdemo/config.rb
  create compassdemo/sass/screen.scss
  create compassdemo/sass/print.scss
  create compassdemo/sass/ie.scss
  write compassdemo/stylesheets/ie.css
  write compassdemo/stylesheets/print.css
  write compassdemo/stylesheets/screen.css
*****
Congratulations! Your compass project has been created.
You may now add and edit sass stylesheets in the sass subdirectory of your project.
Sass files beginning with an underscore are called partials and won't be
compiled to CSS, but they can be imported into other sass stylesheets.
You can configure your project by editing the config.rb configuration file.
You must compile your sass stylesheets into CSS when they change.
This can be done in one of the following ways:
  1. To compile on demand:
     compass compile [path/to/project]
  2. To monitor your project for changes and automatically recompile:
     compass watch [path/to/project]
More Resources:
* Website: http://compass-style.org/
* Sass: http://sass-lang.com
* Community: http://groups.google.com/group/compass-users/
To import your new stylesheets add the following lines of HTML (or equivalent) to your webpage:
<head>
<link href="/stylesheets/screen.css" media="screen, projection" rel="stylesheet" type="text/css" />
<link href="/stylesheets/print.css" media="print" rel="stylesheet" type="text/css" />
<!--[if IE]
  <link href="/stylesheets/ie.css" media="screen, projection" rel="stylesheet" type="text/css" />
<![endif]-->
</head>
C:\Users\romag\Documents\compass projects>

```

Note : on peut avoir la liste de commandes disponibles avec « compass help »

### a. Trouver des bibliothèques Sass

<https://rubygems.org/>

Importer une bibliothèque

```
@import "compass";
@import "compass/reset";
```

<http://compass-style.org/reference/compass/>

+

<http://compass-style.org/frameworks/>

## Exemple

```
@import "compass/reset";  
@import "compass/typography/links";
```

Import de la bibliothèque

```
$color1:blue;  
$color2:red;
```

```
a{  
  @include link-colors($color1, $color2);  
}
```

Utilisation de mixin de la bibliothèque

### b. Compilation d'un projet Compass

Se déplacer sur le répertoire du projet (cd ...). Dans l'exemple c'est le dossier « compassdemo »

#### Manuelle

```
compass compile
```

#### Automatique

```
compass watch
```