

Mieux travailler le CSS

Les bases de SASS/Compass

FOIRE DU SAVOIR. 2014.08.22

Avant de commencer

- Offrir une solution de travail pour l'ensemble de la compagnie en regard de la production du CSS.
- Nous sommes ici pour amorcer une réflexion à ce sujet.
- Je ne suis pas seul à décider...
- ... mais j'aimerais bien qu'on me suive ;-)
- Ceci n'est pas une introduction au CSS.
- À la fin de cette présentation vous saurez utiliser préprocesseur, car le tout est vraiment très simple et tourne autour de quatre concepts.

Sommaire

- Qui suis-je ?
- Du bon usage de la paresse
- La lourdeur (et le bordel) CSS
- Les préprocesseurs à la rescousse
- Pourquoi SASS ?
- Les bases de SASS (Syntactically Awesome Style Sheets):
 - a) l'inclusion b) les variables c) les mixins d) les opérations mathématiques
- L'apport de Compass
- L'installation
- Conclusion

Qui suis-je ?

- Intégrateur web chez Innobec depuis octobre 2013
- Dix-sept ans en tant que travailleur autonome (graphisme, mise en pages, rapports techniques, intégration Web, assistance)
- Ai burlingué dans pas mal de technologies (homme à tout faire, donc bon à rien)
- Mes clients précédents : maison d'éditions, Hydro-Québec, Contractuelle, CARTV, etc.
- Écrivain (cinq romans publiés, le sixième, *Falaise*, paraîtra en octobre 2014, chez l'Hexagone.)
- www.guyverville.com

O

« De la paresse, tu useras intelligemment. »

NVVRP

DRY

Ne vous répétez pas (*Don't Repeat Yourself* en anglais, aussi désigné par l'acronyme DRY) est une philosophie en programmation informatique consistant à éviter la redondance de code au travers de l'ensemble d'une application afin de faciliter la maintenance, le test, le débogage et les évolutions de cette dernière.

http://fr.wikipedia.org/wiki/Ne_vous_répétez_pas

1

Le constat

Ce qu'on se répète en CSS...

La lourdeur (et le bordel) CSS

```
1512 }
1513 /* line 170, ../sass/composants/_google.scss */
1514 #infoCegeps #cadreBlanc {
1515     @import "iucpq_styles.css";
1516     @import "iucpq_blue_styles.css";
1517     @import "iucpq_vert02_styles.css";
1518     @import "iucpq_vert01_styles.css";
1519     @import "iucpq_red_styles.css";
1520     @import "iucpq_orange_styles.css";
1521 }
1522 }
1523 }
1524 }
1525 #infoC
1526 font
1527 line
1528 bord
1529 marg
1530 padd
1531 lett
1532 }
1533 #infoC
1534 marg
1535 padd
1536 }
1537 #infoC
1538 line
1539 list
1540 marg
1541 padd
1542 curs
1543 }
1544 #infoC
1545 bord
1546 }
1547 #infoC
1548 bord
1549 colo
1550 }
1551 #infoC
1552 colo
1553 }
1554 }
1555 .boite
1556 heig
1557 }
1558 }
```

Qu'est-ce qui cloche ?

- Rien...
- Long à faire
- Beaucoup d'écriture
- Source d'erreurs
- Avec de très gros projets, la gestion devient ardue.
- On se cherche beaucoup.

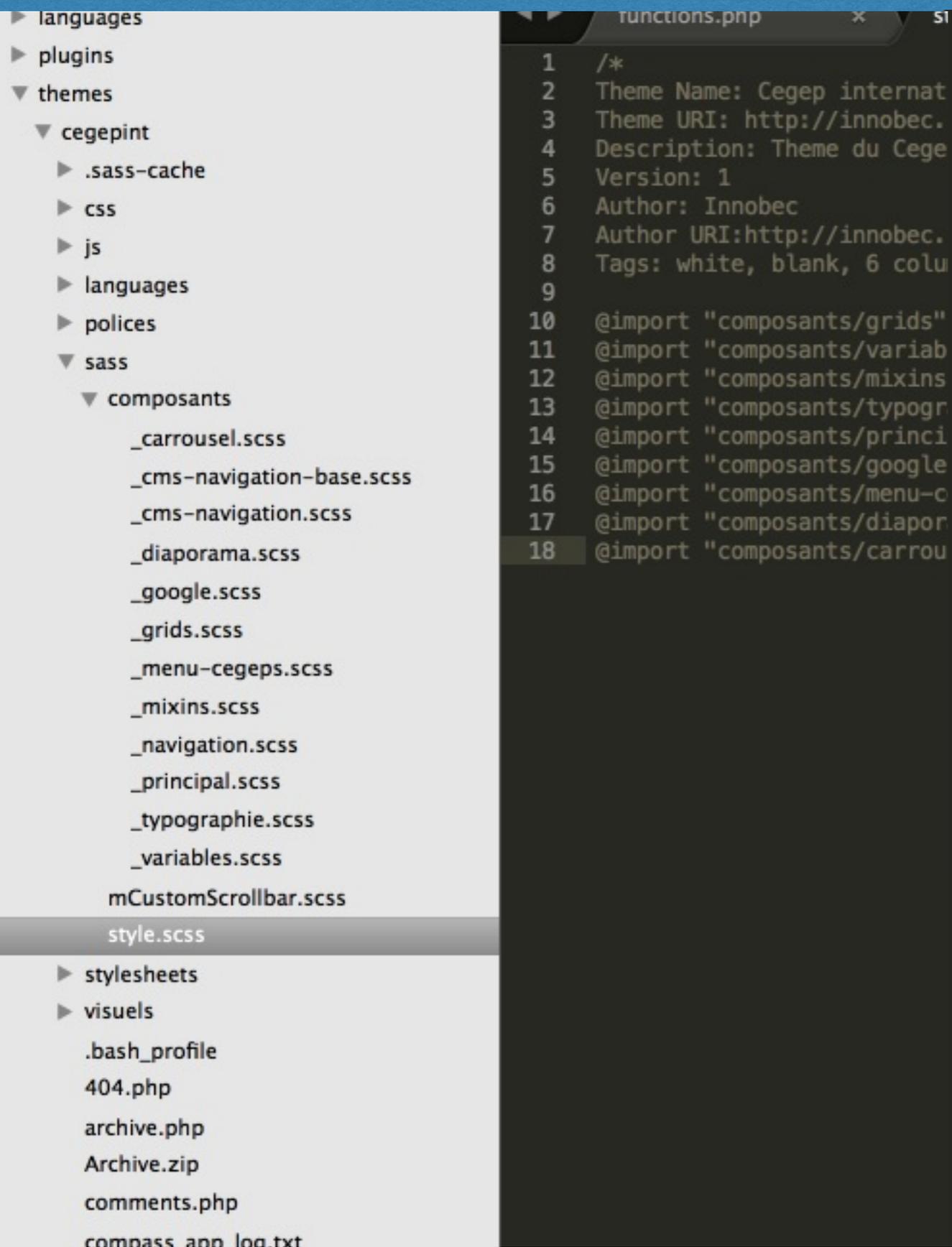
Le préprocesseur à la rescousse

- Organisation hiérarchique
- Emploi de variables
- Emploi de fonctions
- Simplicité du code
- Compréhension syntaxique

```
7 #infoCegeps {
8   @include span(4 of 12 last);
9   @include prefix(1 of 12);
10  #cadreBlanc {
11    position: relative;
12    padding: 10px;
13    height: 380px;
14    margin-right: 5px;
15    background-color: white;
16    @include box-shadow;
17    margin-bottom: 15px;
18  }
19  margin-top: 20px;
20  h2 {
21    font-size: 16px;
22    line-height: 1.3em;
23    border-bottom: 1px solid $beigeBordureFonce;
24    margin: 5px 0 15px 0;
25    padding: 0 0 10px;
26    letter-spacing: -.025em;
27  }
28  ul {
29    // overflow-y: scroll;
30    // ms-overflow-x: hidden;
31    margin: 0;
32    padding: 0;
33    li {
34      line-height: 1.2em;
35      list-style-type: none;
36      margin-bottom: 8px;
37      padding: 0;
38      cursor: pointer;
39      span {
40        border-bottom: 1px dotted $texte;
41        &:hover, &:focus {
42          border: none;
43          color: $lien;
44        }
45      }
46    }
47  }
48  a {
49    color: inherit;
50  }
51 }
52
53 .boiteCegeps {
```

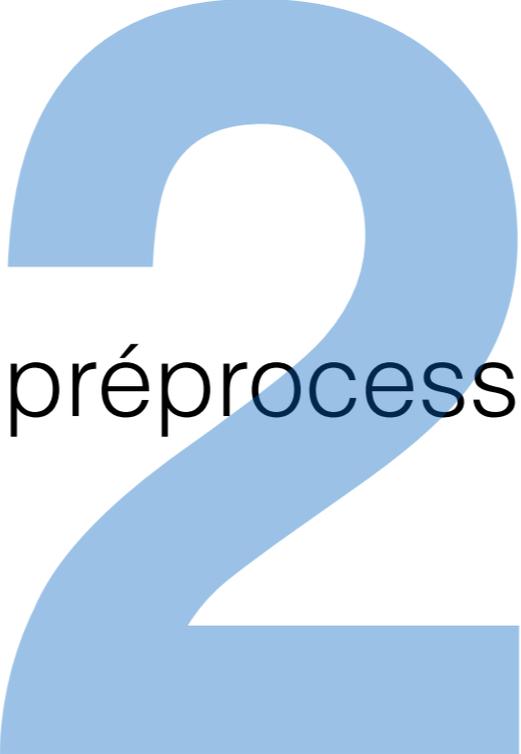
```
1528 border-bottom: 1px solid #c8c8c8;
1529 margin: 5px 0 15px 0;
1530 padding: 0 0 10px;
1531 letter-spacing: -.025em;
1532 }
1533 #infoCegeps ul {
1534   margin: 0;
1535   padding: 0;
1536 }
1537 #infoCegeps ul li {
1538   line-height: 1.2em;
1539   list-style-type: none;
1540   margin-bottom: 8px;
1541   padding: 0;
1542   cursor: pointer;
1543 }
1544 #infoCegeps ul li span {
1545   border-bottom: 1px dotted #3b3b3b;
1546 }
1547 #infoCegeps ul li span:hover, #infoCegeps ul li span:focus {
1548   border: none;
1549   color: #016fad;
1550 }
1551 #infoCegeps a {
1552   color: inherit;
1553 }
1554
1555 .boiteCegeps {
1556   height: 300px;
1557 }
1558
```

Le préprocesseur à la rescousse



- Organisation du code
- Réutilisation
- Base de travail commune
- Collaboration améliorée
- Grande compatibilité entre les fureteurs
- Le préprocesseur avertit des erreurs, ce qui permet de produire instantanément du code sûr
- Une certain niveau de programmation est possible, grâce aux opérations mathématiques

```
207 }
208 a {
209     color: inherit;
210 }
211 }
212
213 .boiteCegeps {
```



Les préprocesseurs

Les préprocesseurs populaires



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



<http://lesscss.org/>



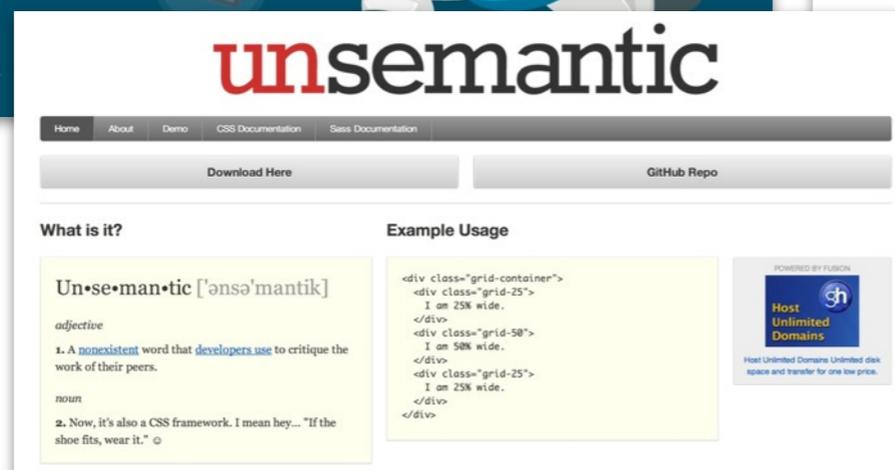
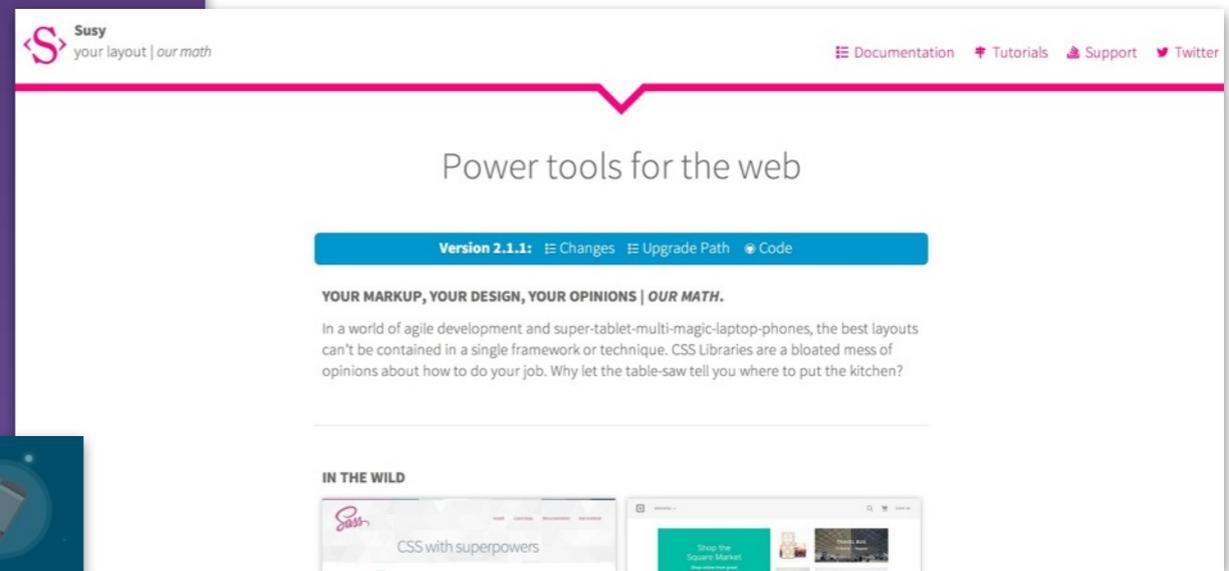
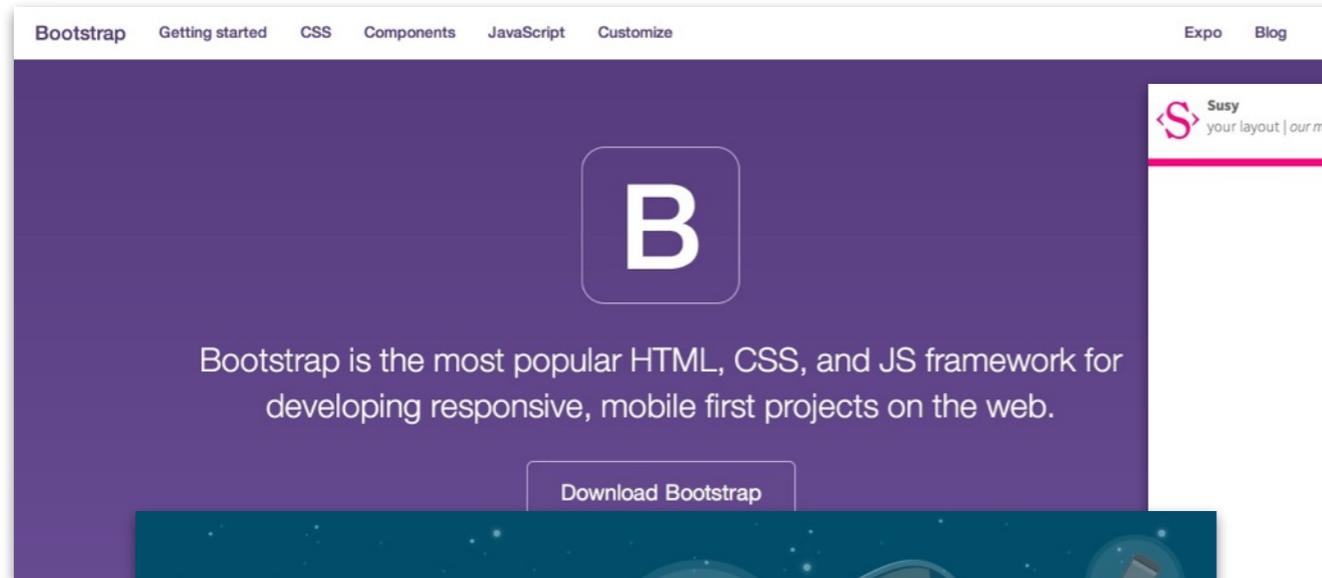
<http://learnboost.github.io/stylus/>

Pourquoi SASS ?

- Parce que je le connais...
- Peut-être le plus populaire.
- Vous n'avez pas à me croire sur parole :
<http://fr.slideshare.net/patricka1/css-preprocessors-sass-less-and-stylus>.
- 80 % de ces préprocesseurs sont identiques.
- Les 20 % restants font généralement peser la balance du côté de SASS.

Foundation, Bootstrap, Suzy, Gumby, etc.

Ils aiment tous SASS !



SASS en quatre étapes



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

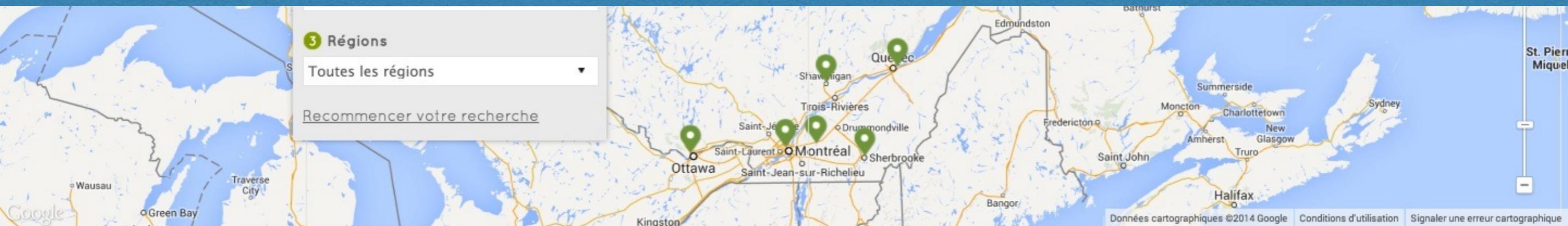
- Inclusion (mise en contexte)
- Variables
- Mixins (déjà tout cuits : Compass, Compass-Recipes, Bourbon)
- Opérations mathématiques

SASS ou SCSS ?

- Les fichiers SASS se terminent par .SCSS.
- C'est pour les distinguer de la première version de SASS (les fichiers se terminaient en... .SASS).
- On peut toujours utiliser l'ancienne syntaxe, mais vous serez tout seul(e) dans votre coin.

3 Démonstration

SASS par l'exemple



TECHNIQUES DE LABORATOIRE – SPÉCIALISATION EN BIOTECHNOLOGIES (210.AA)

Buts du programme

Ce programme vise à former des techniciens aptes à travailler dans les laboratoires des entreprises manufacturières, principalement celles des secteurs agroalimentaire et pharmaceutique, et dans les laboratoires spécialisés en environnement. De plus, ces personnes seront en mesure de travailler dans les laboratoires des entreprises spécialisées en biotechnologies ou dans les laboratoires des entreprises des secteurs de la chimie industrielle, des mines et de la métallurgie, de la pétrochimie, des matériaux et des pâtes et papiers.

Les techniciens de laboratoire seront capables de prélever des échantillons, d'effectuer des analyses de chimie organique et de biochimie en utilisant certaines méthodes d'analyse instrumentale, de compiler et de traiter les données, de rédiger des rapports et de transmettre les résultats, en respectant les règles de santé et de sécurité ainsi que les bonnes pratiques de laboratoire et de fabrication.

(6) cégeps au Québec offrent ce programme.

[Collège de Shawinigan](#)

[Cégep de Lévis-Lauzon](#)

[Collège Ahuntsic](#)

[Cégep de l'Outaouais](#)

[Cégep de Saint-Hyacinthe](#)

[Cégep de Sherbrooke](#)



Comment faire sa demande

SASS par l'exemple



TECHNIQUES DE LABORATOIRE - SPÉCIALISATION EN BIOTECHNOLOGIES (210.AA)

Buts du programme

Ce programme vise à former des techniciens aptes à travailler dans les laboratoires des

(6) cégeps au Québec offrent ce programme.

[Collège de Shawinigan](#)
[Cégep de Lévis-Lauzon](#)
[Collège Ahuntsic](#)
[Cégep de l'Outaouais](#)
[Cégep de Saint-Hyacinthe](#)
[Cégep de Sherbrooke](#)

Comment faire sa demande

```
<div id="infoCegeps">
  <div id="cadreBlanc">
    <h2> (6) cégeps au Québec offrent ce programme.</h2>
    <div id="listeCegeps">
      <div class="boiteCegeps">
        <ul>
          <li class="cegepOffre"><span>Collège de Shawinigan</span></li>
          <li class="cegepOffre"><span>Cégep de Lévis-Lauzon</span></li>
          <li class="cegepOffre"><span>Collège Ahuntsic</span></li>
          <li class="cegepOffre"><span>Cégep de l'Outaouais</span></li>
          <li class="cegepOffre"><span>Cégep de Saint-Hyacinthe</span></li>
          <li class="cegepOffre"><span>Cégep de Sherbrooke</span></li>
        </ul>
      </div>
    </div>
  </div>
</div>
```

SASS par l'exemple

```
#infoCegeps {  
  width: 323px;  
  float: right;  
  margin-right: 0;  
  padding-left: 85px;  
  margin-top: 20px;  
}
```

```
#infoCegeps #cadreBlanc {  
  position: relative;  
  padding: 10px;  
  height: 380px;  
  margin-right: 5px;  
  background-color: white;  
  -moz-box-shadow: 0 0 8px rgba(0, 0, 0, 0.35);  
  -webkit-box-shadow: 0 0 8px rgba(0, 0, 0, 0.35);  
  box-shadow: 0 0 8px rgba(0, 0, 0, 0.35);  
  margin-bottom: 15px;  
}
```

```
#infoCegeps h2 {  
  font-size: 16px;  
  line-height: 1.3em;  
  border-bottom: 1px solid #c8c8c8;  
  margin: 5px 0 15px 0;  
  padding: 0 0 10px 0;  
  letter-spacing: -.025em;  
}
```

```
#infoCegeps ul {  
  margin: 0;  
  padding: 0;  
}
```

```
#infoCegeps ul li {  
  line-height: 1.2em;  
  list-style-type: none; etc.
```



TECHNIQUES DE LABORATOIRE - SPÉCIALISATION EN BIOTECHNOLOGIES (210.AA)

Le programme

Ce programme vise à former des techniciens aptes à travailler dans les laboratoires des entreprises manufacturières, principalement ceux des secteurs agroalimentaire et pharmaceutique, et dans les laboratoires spécialisés en environnement. De plus, ces personnes seront en mesure de travailler dans les laboratoires des entreprises spécialisées en biotechnologie ou dans les laboratoires des entreprises des secteurs de la chimie industrielle, des mines et de la métallurgie, de la pétrochimie, des matériaux et des pâtes et papiers.

Les techniciens de laboratoire seront capables de prélever des échantillons, d'effectuer des analyses de chimie organique et de biochimie en utilisant certaines méthodes d'analyse instrumentale, de compiler et de traiter les données, de rédiger des rapports et de transmettre les résultats, en respectant les règles de santé et de sécurité ainsi que les bonnes pratiques de laboratoire et de fabrication.

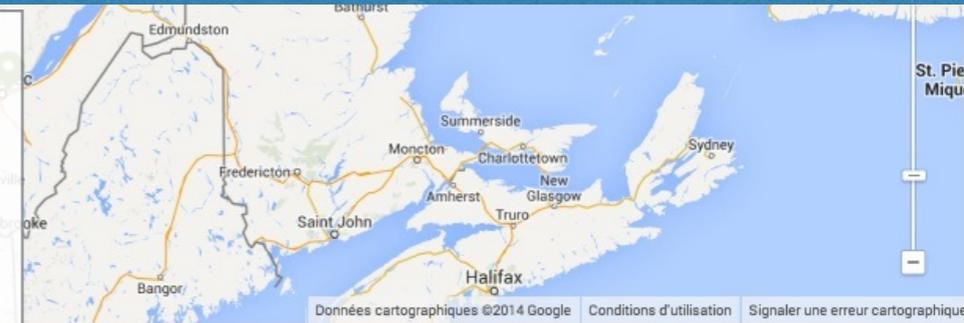
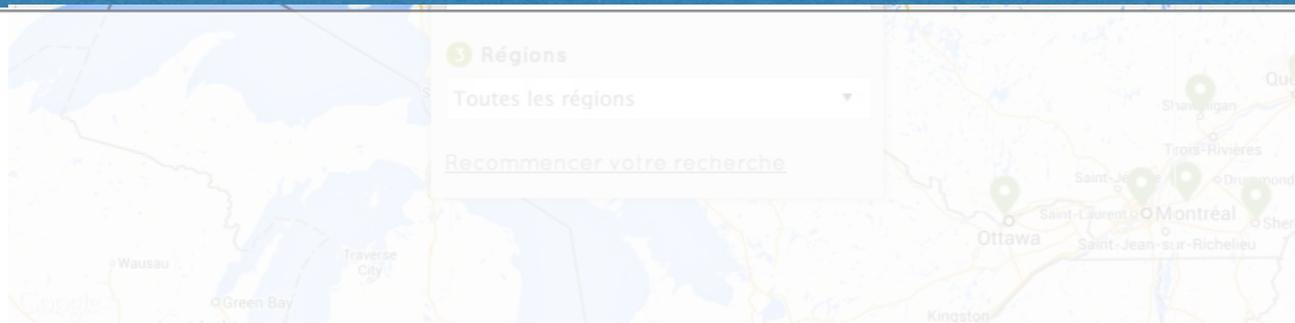
(6) cégeps au Québec offrent ce programme.

- [Collège de Shawinigan](#)
- [Cégep de Lévis-Lauzon](#)
- [Collège Ahuntsic](#)
- [Cégep de l'Outaouais](#)
- [Cégep de Saint-Hyacinthe](#)
- [Cégep de Sherbrooke](#)

 Comment faire sa demande

SASS par l'exemple

```
#infoCegeps {  
  width: 323px;  
  float: right;  
  margin-right: 0;  
  padding-left: 85px;  
  margin-top: 20px;  
}
```



TECHNIQUES DE LABORATOIRE - SPÉCIALISATION EN BIOTECHNOLOGIES (210.AA)

Buts du programme

Ce programme vise à former des techniciens aptes à travailler dans les laboratoires des entreprises manufacturières, principalement celles des secteurs agroalimentaire et pharmaceutique, et dans les laboratoires spécialisés en environnement. De plus, ces personnes seront en mesure de travailler dans les laboratoires des entreprises

des entreprises des secteurs de la pétrochimie, des matériaux et

ver des échantillons, d'effectuer
utilisant certaines méthodes
nnées, de rédiger des rapports et
e santé et de sécurité ainsi que les

(6) cégeps au Québec offrent ce programme.

- [Collège de Shawinigan](#)
- [Cégep de Lévis-Lauzon](#)
- [Collège Ahuntsic](#)
- [Cégep de l'Outaouais](#)
- [Cégep de Saint-Hyacinthe](#)
- [Cégep de Sherbrooke](#)

 [Comment faire sa demande](#)

```
<div id="infoCegeps">  
  <div id="cadreBlanc">  
    <h2>(6) cégeps au Québec offrent ce programme.</h2>  
    <div id="listeCegeps">  
      <div class="boiteCegeps">  
        <ul>  
          <li class="cegepOffre"><span>Collège de Shawinigan</span></li>  
          <li class="cegepOffre"><span>Cégep de Lévis-Lauzon</span></li>  
          <li class="cegepOffre"><span>Collège Ahuntsic</span></li>  
          <li class="cegepOffre"><span>Cégep de l'Outaouais</span></li>  
          <li class="cegepOffre"><span>Cégep de Saint-Hyacinthe</span></li>  
          <li class="cegepOffre"><span>Cégep de Sherbrooke</span></li>  
        </ul>  
      </div>  
    </div>  
  </div>  
</div>
```

SASS par l'exemple (les commandes)

```
#infoCegeps {  
  @include span(4 of 12 last);  
  @include prefix(1 of 12);  
  margin-top: 20px;  
}
```



Les commandes **@include span()** et **@include prefix()** sont propres à Susy, un cadre de développement comme Bootstrap, Foundation, Unsemantic, Gumby, etc.).

Les commandes appellent des *mixins*.

Dans cet exemple, la commande appelle *span* et transmet les paramètres de colonnes. Par la suite, une seconde commande appelle *prefix* (une largeur de colonne sur la gauche).

Nous y reviendrons.

LABORATOIRE -
SPÉCIALISATION EN
BIOTECHNOLOGIES

Buts du programme

Ce programme vise à former des techniciens aptes à travailler dans les laboratoires des entreprises manufacturières, principalement celles des secteurs agroalimentaire et pharmaceutique, et dans les laboratoires spécialisés en environnement. De plus, ces personnes seront en mesure de travailler dans les laboratoires des entreprises des secteurs de la pétrochimie, des matériaux et

ver des échantillons, d'effectuer
ilisant certaines méthodes
nnées, de rédiger des rapports et
e santé et de sécurité ainsi que les

(6) cégeps au Québec offrent ce programme.

[Collège de Shawinigan](#)
[Cégep de Lévis-Lauzon](#)
[Collège Ahuntsic](#)
[Cégep de l'Outaouais](#)
[Cégep de Saint-Hyacinthe](#)
[Cégep de Sherbrooke](#)

 Comment faire sa demande

```
<div id="infoCegeps">  
  <div id="cadreBlanc">  
    <h2>(6) cégeps au Québec offrent ce programme.</h2>  
    <div id="listeCegeps">  
      <div class="boiteCegeps">  
        <ul>  
          <li class="cegepOffre"><span>Collège de Shawinigan</span></li>  
          <li class="cegepOffre"><span>Cégep de Lévis-Lauzon</span></li>  
          <li class="cegepOffre"><span>Collège Ahuntsic</span></li>  
          <li class="cegepOffre"><span>Cégep de l'Outaouais</span></li>  
          <li class="cegepOffre"><span>Cégep de Saint-Hyacinthe</span></li>  
          <li class="cegepOffre"><span>Cégep de Sherbrooke</span></li>  
        </ul>  
      </div>  
    </div>  
  </div>  
</div>
```

SASS par l'exemple (l'inclusion)

```
#infoCegeps {  
  @include span(4 of 12 last);  
  @include prefix(1 of 12);  
  margin-top: 20px;  
  #cadreBlanc {  
    position: relative;  
    padding: 10px;  
    height: 380px;  
    margin-right: 5px;  
    background-color: white;  
    @include box-shadow;  
    margin-bottom: 15px;  
  }  
}
```

```
#infoCegeps #cadreBlanc {  
  position: relative;  
  padding: 10px;  
  height: 380px;  
  margin-right: 5px;  
  background-color: white;  
  -moz-box-shadow: 0 0 8px rgba(0, 0, 0, 0.35);  
  -webkit-box-shadow: 0 0 8px rgba(0, 0, 0, 0.35);  
  box-shadow: 0 0 8px rgba(0, 0, 0, 0.35);  
  margin-bottom: 15px;  
}
```

La commande **@include box-shadow** appartient à Compass, une librairie de mixins. Sass et Compass sont intimement liés. Il existe d'autres librairies, telle Bourbon.

```
<div id="infoCegeps">  
  <div id="cadreBlanc">  
    <h2> (6) cégeps au Québec offrent ce programme.</h2>  
    <div id="listeCegeps">  
      <div class="boiteCegeps">  
        <ul>  
          <li class="cegepOffre"><span>Collège de Shawinigan</span></li>  
          <li class="cegepOffre"><span>Cégep de Lévis-Lauzon</span></li>  
          <li class="cegepOffre"><span>Collège Ahuntsic</span></li>  
          <li class="cegepOffre"><span>Cégep de l'Outaouais</span></li>  
          <li class="cegepOffre"><span>Cégep de Saint-Hyacinthe</span></li>  
          <li class="cegepOffre"><span>Cégep de Sherbrooke</span></li>  
        </ul>  
      </div>  
    </div>  
  </div>  
</div>
```



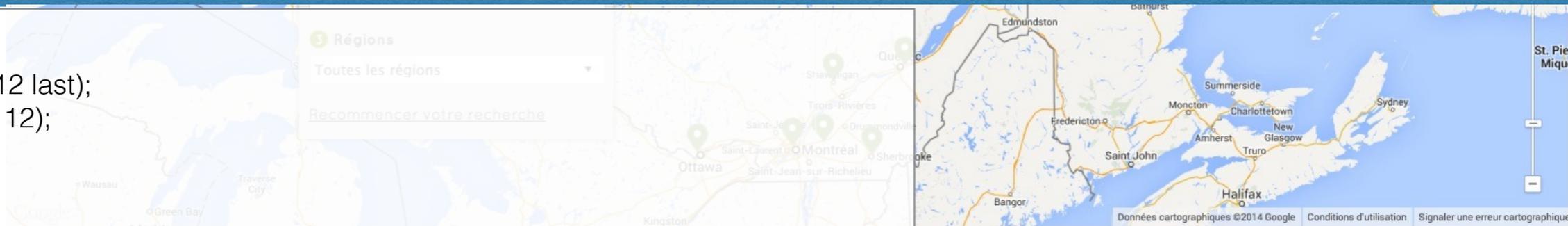
(6) cégeps au Québec offrent ce programme.

[Collège de Shawinigan](#)
[Cégep de Lévis-Lauzon](#)
[Collège Ahuntsic](#)
[Cégep de l'Outaouais](#)
[Cégep de Saint-Hyacinthe](#)
[Cégep de Sherbrooke](#)

 Comment faire sa demande

SASS par l'exemple

```
#infoCegeps {
  @include span(4 of 12 last);
  @include prefix(1 of 12);
  margin-top: 20px;
  #cadreBlanc {
    position: relative;
    padding: 10px;
    height: 380px;
    margin-right: 5px;
    background-color: white;
    @include box-shadow;
    margin-bottom: 15px;
  }
  h2 {
    font-size: 16px;
    line-height: 1.3em;
    border-bottom: 1px solid $beigeBordureFonce;
    margin: 5px 0 15px 0;
    padding: 0 0 10px 0;
    letter-spacing: -.025em;
  }
  ul {
    margin: 0;
    padding: 0;
    li {
      line-height: 1.2em;
      list-style-type: none;
      margin-bottom: 8px;
      padding: 0;
      cursor: pointer;
      span {
        border-bottom: 1px dotted $texte;
        &:hover, &:focus {
```



TECHNIQUES DE
LABORATOIRE -
SPÉCIALISATION EN
BIOTECHNOLOGIES
(210.AA)

Buts du programme

Ce programme vise à former des techniciens aptes à travailler dans les laboratoires des entreprises manufacturières, principalement dans les secteurs de l'agroalimentaire et des produits pharmaceutiques. Les personnes seront en mesure de travailler dans les laboratoires des entreprises spécialisées en biotechnologies ou dans les laboratoires des entreprises des secteurs de la chimie, de la métallurgie, de la mécanique, de l'électronique, de l'aérospatial, de l'automobile, de la santé, de la recherche et de développement.

(6) cégeps au Québec offrent ce programme.

- [Collège de Shawinigan](#)
- [Cégep de Lévis-Lauzon](#)
- [Collège Ahuntsic](#)
- [Cégep de l'Outaouais](#)
- [Cégep de Saint-Hyacinthe](#)
- [Cégep de Sherbrooke](#)

```
<div id="infoCegeps">
  <div id="cadreBlanc">
    <h2> (6) cégeps au Québec offrent ce programme.</h2>
    <div id="listeCegeps">
      <div class="boiteCegeps">
        <ul>
          <li class="cegepOffre"><span>Collège de Shawinigan</span></li>
          <li class="cegepOffre"><span>Cégep de Lévis-Lauzon</span></li>
          <li class="cegepOffre"><span>Collège Ahuntsic</span></li>
          <li class="cegepOffre"><span>Cégep de l'Outaouais</span></li>
          <li class="cegepOffre"><span>Cégep de Saint-Hyacinthe</span></li>
          <li class="cegepOffre"><span>Cégep de Sherbrooke</span></li>
        </ul>
      </div>
    </div>
  </div>
</div>
```

Variable

Variables

CSS

```
a{...
```

```
  color: #900000
```

```
}
```

```
.avertissement{...
```

```
  color: #900000;
```

```
  border-left: 1px solid #900000}
```

SASS

```
$rouge: #900000;
```

```
a{color: $rouge}
```

```
.avertissement{
```

```
  color: $rouge
```

```
  border-left: 1px solid $rouge}...
```

La variable sera aussi déplacée dans un autre fichier afin de mieux la repérer.

Les graphistes aimeront les variables.

```
1 // Ombre
2 $default-box-shadow-h-offset: 0;
3 $default-box-shadow-v-offset: 0;
4 $default-box-shadow-blur: 8px;
5 $default-box-shadow-color: rgba(0, 0, 0, 0.35);
6
7 // Couleurs
8 $beige: #f4f4f2;
9 $beigeBordure: #e2e2e2;
10 $beigeBordureFonce: #c8c8c8;
11
12 $bleu: #0096eb;
13 $bleuBordure: #007bc0;
14 $lien: #016fad;
15
16 $gris: #eaeaea;
17 $grisSecond: #dbdbdb;
18 $grisPale: #f4f4f2;
19 $grisMedium: #575757;
20 $grisFonce: #a7a7a7;
21 $grisCarrousel: #eeeeee;
22 $grisFrontal: #454545;
23
24
25 $vertFonce: #849a00;
26 $vertPale: #a0b713;
27
28
29 $survol: #f3f3f3;
30 $actif: #d1d1d1;
31
32
33 $texte: #3b3b3b;
34 $texteEfface: #707070;
35 $creme: #f5f5f1;
36
37 $rougeMenu: #ef6149;
38 $vertMenu: #9ab206;
39 $mauveMenu: #6054cf;
40 $bleuMenu: #007bc0;
```

- Fini les ombres portées trop grosses ou trop foncées !
- Fini les arcs-en-ciel de nuances !
- {presque} Adieu les incohérences !
- Adieu la pipette !
- Bonjour la flexibilité pour tout le monde.
- Un meilleur respect du cahier des charges.

Esperluétons (&)

Esperluette, perluette, perluète, « et commercial », les quatre se disent, comme l'affirma, autrefois, Vaugelas.



*Fermeté ! Apprendre à la
France à bien parler
et à bien écrire, telle est ma
devise.*

Dans CSS

```
a{  
  
    color: #900000;  
  
}  
  
a:hover{  
  
    opacity: .5  
  
}
```

Dans SASS

```
a{  
  
    color: $rouge;  
  
&:hover{  
  
    opacity: .5  
  
}
```

Tant qu'à esperluéter (&)

Dans CSS

```
a{
    color: #900000;
}

a:hover{
    opacity: .5
}

#container a:hover{
    color: #c8c8c8}
```

Dans SASS

```
a{
    color: $rouge;

    &:hover{
        @include opacity (.5);

        #container &{
            color: $bleu}
    }
}
```

Les deux perluettes ne possèdent pas la même identité. Dans le premier cas **& = a**, tandis que dans l'autre, **& = a:hover**.

Tant qu'à esperluéter (&)

Dans SASS

a{

```
a:hover{  
  filter: progid:DXImageTransform.Microsoft.Alpha(Opacity=5);  
  opacity: .5  
}
```

color: \$rouge;

&:hover{

@include opacity (.5);

#container &{

color: \$bleu}

}}

Les cousins

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <title>Un exemple SASS</title>
    <link rel="stylesheet" href="">
  </head>
  <body>
    <div class="boiteCegeps">
      <ul>
        <li class="cegep0ffre"><span>Collège de Shawinigan</span></li>
        <li class="cegep0ffre"><span>Cégep de Lévis-Lauzon</span></li>
        <li class="cegep0ffre"><span>Collège Ahuntsic</span></li>
        <li class="cegep0ffre"><span>Cégep de l'Outaouais</span></li>
        <li class="cegep0ffre"><span>Cégep de Saint-Hyacinthe</span></li>
        <li class="cegep0ffre"><span>Cégep de Sherbrooke</span></li>
      </ul>
    </div>
    <div class="autreListe">
      <ul>
        <li class="cegepProgramme"><span>Collège de Shawinigan</span></li>
        <li class="cegepProgramme"><span>Cégep de Lévis-Lauzon</span></li>
        <li class="cegepProgramme"><span>Collège Ahuntsic</span></li>
        <li class="cegepProgramme"><span>Cégep de l'Outaouais</span></li>
        <li class="cegepProgramme"><span>Cégep de Saint-Hyacinthe</span></li>
        <li class="cegepProgramme"><span>Cégep de Sherbrooke</span></li>
      </ul>
    </div>
  </body>
</html>
```

```
.boiteCegeps {
  ul {
    margin: 0;
    padding: 0;
    li {
      line-height: 1.2em;
      list-style-type: none;
      margin-bottom: 8px;
      padding: 0;
      cursor: pointer;
    }
  }
}
```

```
.boiteCegeps, .autreListe {
  ul {
    margin: 0;
    padding: 0;
    li {
      line-height: 1.2em;
      list-style-type: none;
      margin-bottom: 8px;
      padding: 0;
      cursor: pointer;
    }
  }
}
```

```
.boiteCegeps ul {margin: 0; padding: 0;}
.boiteCegeps ul li {
  line-height: 1.2em;
  list-style-type: none;
  margin-bottom: 8px;
  padding: 0;
  cursor: pointer;}}}
```

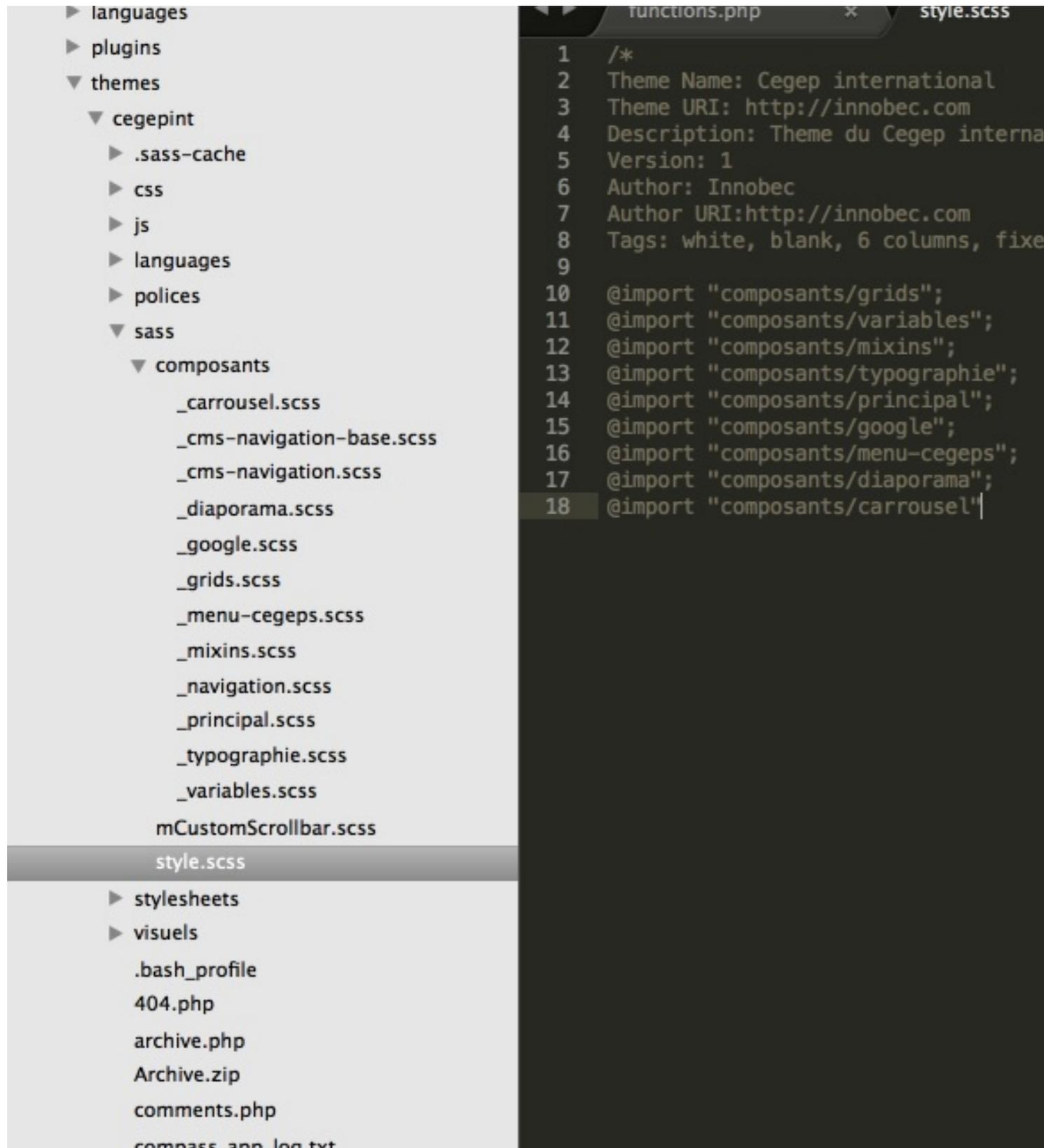
```
.boiteCegeps ul, .autreListe ul {margin: 0; padding: 0;}
.boiteCegeps ul li, .autreListe ul li {
  line-height: 1.2em;
  list-style-type: none;
  margin-bottom: 8px;
  padding: 0;
  cursor: pointer;}}}
```

```
.autreListe {
  @extend .boiteCegeps;
  autre code....
}
```

Faudrait quand même pas exagérer...

- Tout comme durant l'écriture normale du CSS, l'inclusion a ses limites, car les déclarations finales peuvent devenir trop complexes.
- L'utilisation de la perluette à l'intérieur de trop de niveaux pourrait confondre le préprocesseur.
- N'inclure que 4 niveaux.
- Au-delà, la modération a bien meilleur goût. Le code pourrait sans doute être plus simple.
- L'usage efficace des identités et des classes peut régler cette multiplication gourmande.
- **Vous demeurez responsable de la qualité de votre code.**

Les *partials*



- Des fichiers SCSS dont le nom débute par « `_` »...
- ... qu'on peut inclure dans d'autres fichiers avec la commande **@import**.
- ...et qui seront, lors du traitement, rassemblés en un fichier (ou le nombre de fichiers voulus).
- **L'ordre d'introduction est important.**

Mix[in]ons!

Vous savez ce qu'est une fonction ; vous savez ce qu'est un *mixin**

```
@mixin texteRegulier(){
```

```
    font-family: 'quicksandregular', Arial, Helvetica, sans-serif;  
    font-weight: normal;
```

```
}
```

```
.boiteCegep{
```

```
    @include texteRegulier()
```

```
}
```

```
.boiteCegep{
```

```
    font-family: 'quicksandregular', Arial, Helvetica, sans-serif;  
    font-weight: normal;
```

```
}
```

* Un mixin est une supervariable. Sass possède également une directive @function.

Mixons et varions

L'emploi de variables est permis. Abusez-en.

```
@mixin texteRegulier($grosseur:12px, $couleur:#000){  
    font-family: 'quicksandregular', Arial, Helvetica, sans-serif;  
    font-weight: normal;  
    font-size: $grosseur;  
    color: $couleur;  
}
```

```
.boiteCegep{  
    @include texteRegulier()  
}
```

```
.boiteCegep{  
    font-family: 'quicksandregular', Arial, Helvetica, sans-serif;  
    font-weight: normal;  
    font-size: 12px;  
    color: #000  
}
```

Mixons et varions

L'emploi de variables est permis. Abusez-en.

```
@mixin texteRegulier($grosseur:12px, $couleur:#000){  
  font-family: 'quicksandregular', Arial, Helvetica, sans-serif;  
  font-weight: normal;  
  font-size: $grosseur;  
  color: $couleur;  
}
```

```
.boiteCegep{  
  @include texteRegulier(24px,$rouge)  
}
```

```
.boiteCegep{  
  font-family: 'quicksandregular', Arial, Helvetica, sans-serif;  
  font-weight: normal;  
  font-size: 24px;  
  color: #900000  
}
```

Les librairies Compass, Bourbon, etc.

Ces librairies sont essentiellement de vastes collections de *mixins*, parfois gonflées en *framework*.

The image displays three overlapping web pages. The top-left page is the Compass documentation site, featuring a dark theme with the Compass logo and navigation links. The top-right page is the Bourbon website, which has an orange header and a white main area with the text 'A simple and lightweight mixin library for Sass.' and a code snippet. The bottom page is 'Compass Recipes', which has a dark background and features a 'Recipes' notebook icon, the text 'compass Recipes using Sass', and a list of recipe categories under an 'Overview' section.

Compass Documentation

Home Reference Help Blog Get Involved Install Search the docs

compass

Compass is an open-source *CSS Authoring Framework*.

Why designers love Compass.

1. Experience cleaner markup without presentational classes.
2. It's chock full of the web's best [reusable patterns](#).
3. It makes creating [sprites](#) a breeze.
4. Compass mixins make [CSS3](#) easy.
5. Create beautiful [typographic rhythms](#).
6. Download and create extensions with ease.

Compass uses Sass.

Sass is an external CSS pre-processor that makes it easier to manage and maintain stylesheets.

Host Unlimited Domains

Bourbon

A simple and lightweight mixin library for Sass.

```
ol {
  > li {
    background-color: #F4F5F7;
    @include background-image( linear-gradient(to left, #FFF, #F4F5F7) );
    float: left;
    font-family: $lucida-grande;
  }
}
```

CSS Output

```
-webkit-transition: all 0.8s;
-moz-transition: all 0.8s;
transition: all 0.8s;
```

compass Recipes using Sass

Lots of CSS Recipes, made with [Compass](#) using [Sass](#). Ready to use.

Star Fork Follow @Compass_Recipes

Overview

- Backgrounds, patterns & gradients effects
- Color, Variables Names, advanced color functions like 'brightness()' & color scheme functions)

Les opérations mathématiques

Les opérations mathématiques permettent une programmation avancée. Elles ont permis la conception de plusieurs *framework* CSS.

SASS

```
.container { width: 100%; }  
  
article[role="main"] {  
  float: left;  
  width: 600px / 960px * 100%;  
}  
  
aside[role="complimentary"] {  
  float: right;  
  width: 300px / 960px * 100%;  
}
```

CSS

```
.container {  
  width: 100%;  
}  
  
article[role="main"] {  
  float: left;  
  width: 62.5%;  
}  
  
aside[role="complimentary"] {  
  float: right;  
  width: 31.25%;  
}
```

Un cas *hot*. Les *sprites*

1 Bas-Saint-Laurent

Détail [+]

Population: 201 091 habitants

Superficie: 22 186 km²

2 Saguenay-Lac-Saint-Jean

Détail [+]

Population: 278 069 habitants

Superficie: 95 762 km²

3 Capitale-Nationale

Détail [+]

Population: 725 095 habitants

Superficie: 18 644 km²

4 Mauricie

Détail [+]

Population: 266 542 habitants

Superficie: 35 448 km²

5 Estrie

Détail [+]

Population: 318 350 habitants

Superficie: 10 197 km²

6 Montréal

Détail [+]

Population: 1 959 987 habitants



Un cas *hot*. Les sprites

1 Bas-Saint-Laurent

Détail [+]

Population: 201 091 habitants

Superficie: 22 186 km²

2 Saguenay-Lac-Saint-Jean

Détail [+]

Population: 278 069 habitants

Superficie: 95 762 km²

3 Capitale-Nationale

Détail [+]

Population: 725 095 habitants

Superficie: 18 644 km²

4 Mauricie

Détail [+]

Population: 266 542 habitants

Superficie: 35 448 km²

5 Estrie

Détail [+]

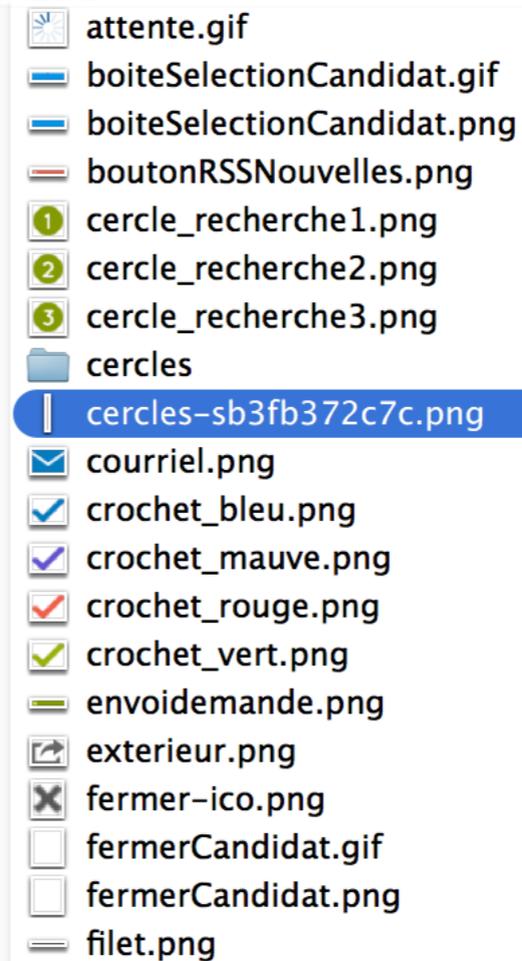
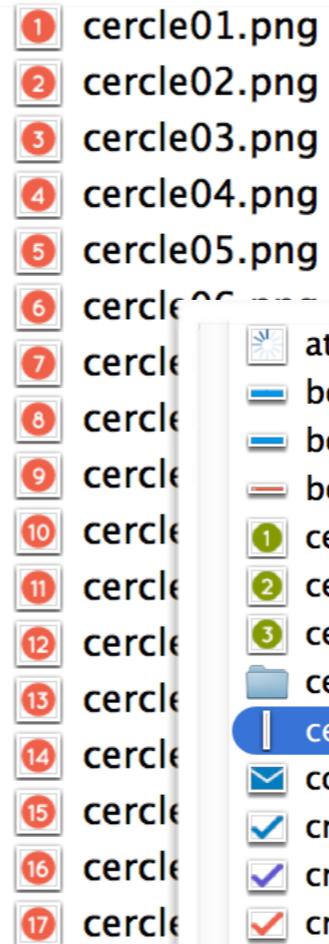
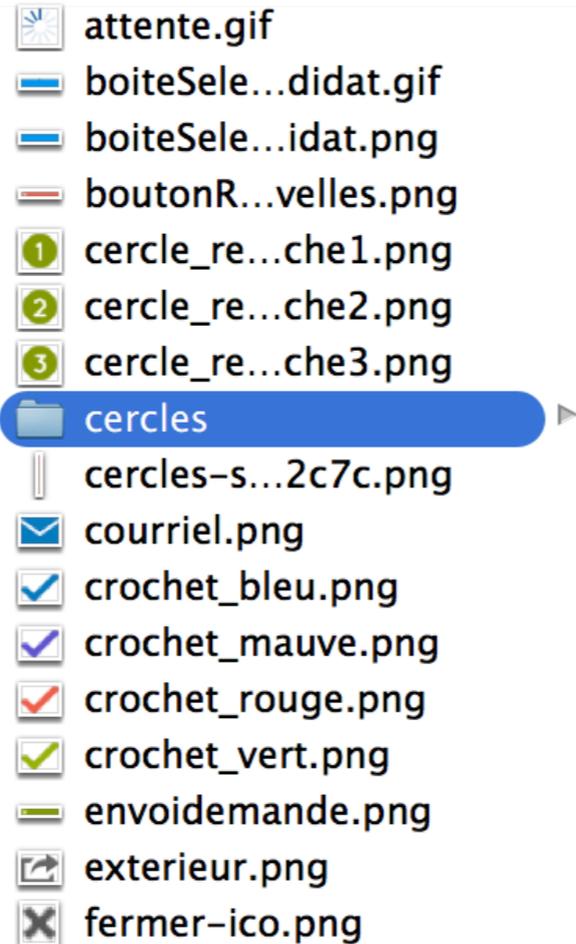
Population: 318 350 habitants

Superficie: 10 197 km²

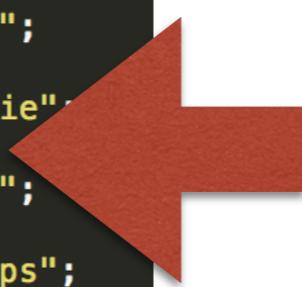
6 Montréal

Détail [+]

Population: 1 959 987 habitants



```
16 */
17 @import "recipes/shadow/drop";
18 @import "composants/grids";
19 @import "composants/variables";
20 @import "composants/mixins";
21 @import "composants/typographie";
22 @import "cercles/*.png";
23 @import "composants/principal";
24 @import "composants/google";
25 @import "composants/menu-cegeps";
26 @import "composants/diaporama";
27 @import "composants/carrousel";
28 @import "composants/manuel";
29
```



```
h2 {
  color: $rougeMenu;
  background-repeat: no-repeat;
  background-image: image-url("cercles-sb3fb372c7c.png");
  padding-left: 35px;
  height: 27px;
  padding-top: 2px;
  margin-bottom: 0;
}
```

Un cas *hot*. Les sprites

1 Bas-Saint-Laurent

Détail [+]

Population: 201 091 habitants
Superficie: 22 186 km²

2 Saguenay-Lac-Saint-Jean

Détail [+]

Population: 278 069 habitants
Superficie: 95 762 km²

3 Capitale-Nationale

Détail [+]

Population: 725 095 habitants
Superficie: 18 644 km²

4 Mauricie

Détail [+]

Population: 266 542 habitants
Superficie: 35 448 km²

5 Estrie

Détail [+]

Population: 318 350 habitants
Superficie: 10 197 km²

6 Montréal

Détail [+]

Population: 1 959 987 habitants

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17

```
&.region02 h2 {  
  background-position: 0 -28px;  
}  
&.region03 h2 {  
  background-position: 0 -56px;  
}  
&.region04 h2 {  
  background-position: 0 -84px;  
}  
&.region05 h2 {  
  background-position: 0 -112px;  
}  
&.region06 h2 {  
  background-position: 0 -140px;  
}  
&.region07 h2 {  
  background-position: 0 -168px;  
}  
&.region08 h2 {  
  background-position: 0 -196px;  
}  
&.region09 h2 {  
  background-position: 0 -224px;  
}  
&.region10 h2 {  
  background-position: 0 -252px;  
}  
&.region11 h2 {  
  background-position: 0 -280px;  
}  
&.region12 h2 {  
  background-position: 0 -308px;  
}  
&.region13 h2 {  
  background-position: 0 -336px;  
}  
&.region14 h2 {  
  background-position: 0 -364px;  
}  
&.region15 h2 {  
  background-position: 0 -392px;  
}  
&.region16 h2 {  
  background-position: 0 -420px;  
}  
&.region17 h2 {  
  background-position: 0 -448px;  
}
```

```
@for $i from 1 through 8 {  
  &.region0#{$i+1} h2  
  { background-position: 0 -28px * $i; }  
}  
@for $i from 9 through 16 {  
  &.region#{$i+1} h2  
  { background-position: 0 -28px * $i; }  
}
```

```
h2 {  
  color: $rougeMenu;  
  background-repeat: no-repeat;  
  background-image: image-url("cercles-sb3fb372c7c.png");  
  padding-left: 35px;  
  height: 27px;  
  padding-top: 2px;  
  margin-bottom: 0;  
}  
@for $i from 1 through 8 {  
  &.region0#{$i+1} h2  
  { background-position: 0 -28px * $i; }  
}  
@for $i from 9 through 16 {  
  &.region#{$i+1} h2  
  { background-position: 0 -28px * $i; }  
}
```

1 Bas-Saint-Laurent

Détail [+]

Population: 201 091 habitants
Superficie: 22 186 km²

2 Saguenay-Lac-Saint-Jean

Détail [+]

Population: 278 069 habitants
Superficie: 95 762 km²

3 Capitale-Nationale

Détail [+]

Population: 725 095 habitants
Superficie: 18 644 km²

4 Mauricie

Détail [+]

Population: 266 542 habitants
Superficie: 35 448 km²

5 Estrie

Détail [+]

Population: 318 350 habitants
Superficie: 10 197 km²

6 Montréal

Détail [+]

Population: 1 959 987 habitants

- 1
- 2
- 3
- 4
- 5
- 6
- 7
- 8
- 9
- 10
- 11
- 12
- 13
- 14
- 15
- 16
- 17

```
h2 {
  color: $rougeMenu;
  background-repeat: no-repeat;
  background-image: image-url("cercles-sb3fb372c7c.png");
  padding-left: 35px;
  height: 27px;
  padding-top: 2px;
  margin-bottom: 0;
}
@for $i from 1 through 8 {
  &.region0#{$i+1} h2
  { background-position: 0 -28px * $i; }
}
@for $i from 9 through 16 {
  &.region#{$i+1} h2
  { background-position: 0 -2
```

```
h2 {
  color: $rougeMenu;
  padding-left: 35px;
  height: 27px;
  padding-top: 2px;
  margin-bottom: 0;
  background-repeat: no-repeat;
  background-image: image-url("cercles-sb3fb372c7c.png");
}
&.region0 {
  @include faireSprite(1, 8, "h2", -28px);
}
&.region {
  @include faireSprite(9, 16, "h2", -28px);
}
```

```
@mixin faireSprite($debut,$fin,$enfant,$increment){
  @for $i from $debut through $fin {
    &#{$i+1} #{$enfant}
    { background-position: 0 $increment * $i; }
  }
}
```

Un cas *hot*. Les sprites

CAS EXTRÊME

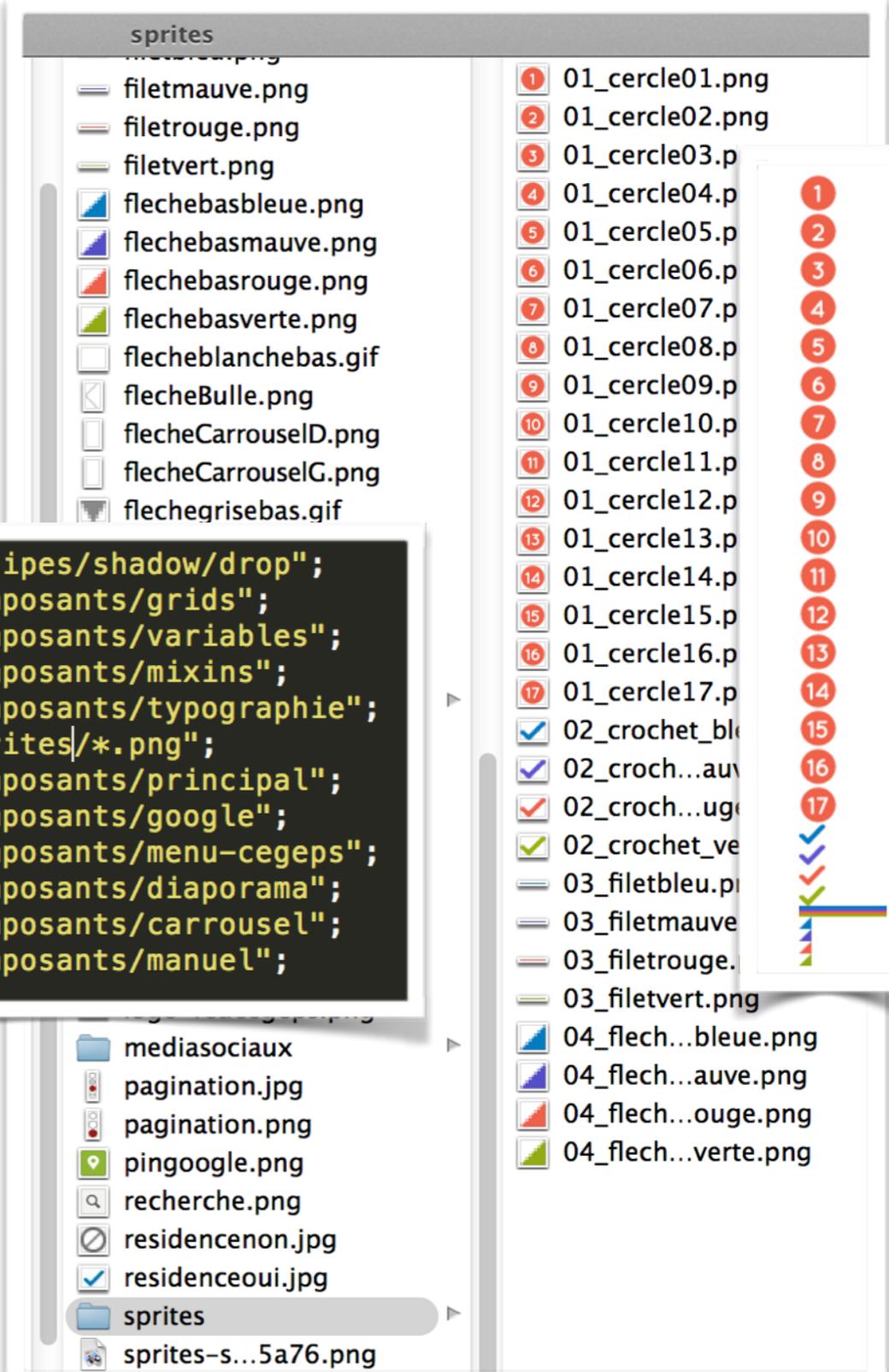
```
@mixin faireSprite($debut,$fin,$enfant,$increment){  
  @for $i from $debut through $fin {  
    &#{$i+1} #{$enfant}  
    { background-position: 0 $increment * $i; }  
  }  
}
```

```
@mixin faireSprite($decalage,$debut,$fin,$enfant,$increment){  
  @for $i from $debut through $fin {  
    &#{$i+1} #{$enfant}  
    { background-position: 0 ($increment * $i - $decalage)}  
  }  
}
```

```
&.region0 {  
  @include faireSprite(0px, 1, 8, "h2", -28px);  
}  
&.region {  
  @include faireSprite(0px, 9, 16, "h2", -28px);  
}
```

```
li{  
  @include faireSprite(-448px, 1, 4, "span", -20px);  
}
```

```
@import "recipes/shadow/drop";  
@import "composants/grids";  
@import "composants/variables";  
@import "composants/mixins";  
@import "composants/typographie";  
@import "sprites/*.png";  
@import "composants/principal";  
@import "composants/google";  
@import "composants/menu-cegeps";  
@import "composants/diaporama";  
@import "composants/carrousel";  
@import "composants/manuel";
```



SASS et plus encore

Plusieurs aspects de SASS ne sont pas couverts dans cette présentation.

L'ensemble des opérations possibles avec SASS est documenté ici:

[http://sass-lang.com/documentation/
file.SASS_REFERENCE.html](http://sass-lang.com/documentation/file.SASS_REFERENCE.html)

ex. : @at-root, @media, @extend%Only Selectors,
@function @if, @else, @while, @warn, @debug

Comment on installe ça ?

- Ça dépend de votre OS. Sur un Mac, c'est facile. Sur un PC, ça dépend... (mais c'est quand même facile)
- SASS roule sous Ruby (Ruby est intégré au Mac, on doit l'installer sous Windows).
- Roule maintenant sous C/C++ (LibSass).
- Après, on passe en mode commande
gem install sass -- pre.
- Installons tout de suite Compass:
gem install compass -- pre.

Pour lancer

- On reste dans le terminal
- On navigue à l'endroit de notre projet
- **sass watch**
- après ça, c'est magique

Si on a installé Compass, c'est la même chose, mais on peut faire davantage, comme créer de nouveaux projets.

- On navigue à l'endroit de notre projet
- **compass watch**
- après ça, c'est tout aussi magique

Pour lancer

compass watch

```
in the load path matchin
cegepint — ruby — 80x24
Last login: Fri Aug  8 18:30:08 on console
You have mail.
genesia:~ gverville$ cd /Users/gverville/Documents/web/cegep_international/wp-co
ntent/themes/cegepint
genesia:cegepint gverville$ compass watch
>>> Compass is watching for changes. Press Ctrl-C to stop.
  create css/style.css
Moving: /Users/gverville/Documents/web/cegep_international/wp-content/themes/ceg
epint/css/style.css
  info sass/style.scss was modified
  error sass/style.scss (Line 28: No files were found in the load path matchin
g "visuels/cercles/*.png". Your current load paths are: /Users/gverville/Documen
ts/web/cegep_international/wp-content/themes/cegepint/visuels)
  create css/style.css
  info sass/style.scss was modified
  create visuels/cercles-sb3fb372c7c.png
  overwrite css/style.css
Moving: /Users/gverville/Documents/web/cegep_international/wp-content/themes/ceg
epint/css/style.css
  info sass/composants/_principal.scss was modified
  unchanged visuels/cercles-sb3fb372c7c.png
  create css/style.css
Moving: /Users/gverville/Documents/web/cegep_international/wp-content/themes/ceg
epint/css/style.css
```



Gestion des erreurs

```
in the load path matching
cegepint — ruby — 80x24
Last login: Fri Aug  8 18:30:08 on console
You have mail.
genesia:~ gverville$ cd /Users/gverville/Documents/web/cegep_international/wp-content/themes/cegepint
genesia:cegepint gverville$ compass watch
>>> Compass is watching for changes. Press Ctrl-C to Stop.
  create css/style.css
Moving: /Users/gverville/Documents/web/cegep_international/wp-content/themes/cegepint/css/style.css
  info sass/style.scss was modified
  error sass/style.scss (Line 28: No files found in the load path matching "visuels/cercles/*.png". Your current load paths are: /Users/gverville/Documents/web/cegep_international/wp-content/themes/cegepint/visuels)
  create css/style.css
  info sass/style.scss was modified
  create visuels/cercles-sb3fb372c7c.png
  overwrite css/style.css
Moving: /Users/gverville/Documents/web/cegep_international/wp-content/themes/cegepint/css/style.css
  info sass/composants/_principal.scss was modified
  unchanged visuels/cercles-sb3fb372c7c.png
  create css/style.css
Moving: /Users/gverville/Documents/web/cegep_international/wp-content/themes/cegepint/css/style.css
```



Si vous n'aimez pas le terminal...

- Des applications existent autant pour Windows, Linux que pour Mac (livereload, compass.app, scout, etc.)



- Des greffons existent également pour plusieurs environnements tels Visual Studio, PHPStorm, Dreamweaver, SublimeText, etc.
- Google est votre ami(e).
- Quelques bonnes adresses.

Convertissez-les !

- Inutile de recommencer son code CSS. On peut le convertir en SASS
- dans le terminal :
sass-convert [options] [INPUT] [OUTPUT]
sass-convert [— from css -- to scss] -- fichier.css - - fichier.scss
sass-convert fichier.css fichier.scss
- un équivalent récursif existe :
sass convert -- r -- from css -- to scss

pour de l'aide : **sass-convert -h**



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

COMPASS : Le fichier `config.rb`

```
require 'susy'
require 'compass/import-once/activate'
# Require any additional compass plugins here.

# Set this to the root of your project when deployed:
http_path = "/"
css_dir = "css"
sass_dir = "sass"
fonts_dir = "polices"
images_dir = "visuels"
javascripts_dir = "js"

# You can select your preferred output style here (can be overridden)
# output_style = :expanded or :nested or :compact or :compressed

# To enable relative paths to assets via compass helper functions.
relative_assets = true

# To disable debugging comments that display the original location
# line_comments = false

# If you prefer the indented syntax, you might want to regenerate
# project again passing --syntax sass, or you can uncomment this:
# preferred_syntax = :sass
# and then run:
# sass-convert -R --from scss --to sass sass scss && rm -rf sass &
require 'fileutils'
on_stylesheet_saved do |file|
  if File.exists?(file) && File.basename(file) == "style.css"
    puts "Moving: #{file}"
    FileUtils.mv(file, File.dirname(file) + "/../" + File.basename(file))
  end
end
```

- L'utilisation de Compass permet de structurer un projet avec le fichier `config.rb` qui sera lu au départ de la compilation.
- Dans cet exemple, SASS prendra en compte les *mixins* de Susy, intégrera une librairie particulière de Compass, indiquera où se trouvent les différents répertoires utilisés et, au final, déplacera le fichier CSS résultant (contexte Wordpress).
- La commande **compass create [chemin/du/projet/]** permet de bâtir automatiquement ce fichier.

COMPASS : image-url

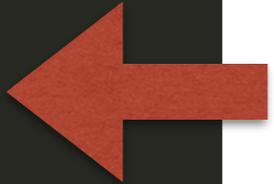
```
require 'susy'
require 'compass/import-once/activate'
# Require any additional compass plugins here

# Set this to the root of your project
http_path = "/"
css_dir = "css"
sass_dir = "sass"
fonts_dir = "polices"
images_dir = "visuels"
javascripts_dir = "js"

# You can select your preferred output style here
# output_style = :expanded or :nested or :compact
# To enable relative paths to assets
relative_assets = true

# To disable debugging comments that display the original SASS
# line_comments = false

# If you prefer the indented syntax
# project again passing --syntax sass
# preferred_syntax = :sass
# and then run:
# sass-convert -R --from scss --to sass
require 'fileutils'
on_stylesheet_saved do |file|
  if File.exists?(file) && File.basename(file, ".css").end_with?(".sass")
    puts "Moving: #{file}"
    FileUtils.mv(file, File.dirname(file) + File.basename(file, ".sass").replace(".sass", ".css"))
  end
end
```



- Compass possède de nombreux *mixins* pratiques dont : **image-url**

SASS

```
.container {
  background-image : image-url('image.jpg');
  background-repeat : no-repeat;
}
```

CSS

```
.container {
  background-image : url('visuels/image.jpg');
  background-repeat : no-repeat;
}
```

COMPASS : @font-face

```
require 'susy'
require 'compass/import-once/activate'
# Require any additional compass plugins here.

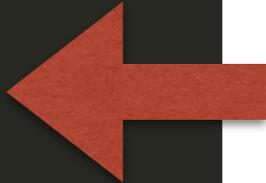
# Set this to the root of your project
http_path = "/"
css_dir = "css"
sass_dir = "sass"
fonts_dir = "polices"
images_dir = "visuels"
javascripts_dir = "js"

# You can select your preferred output style here
# output_style = :expanded or :nested or :compressed

# To enable relative paths to assets, compile here
relative_assets = true

# To disable debugging comments that display the original SASS
# line_comments = false

# If you prefer the indented syntax, you can use the following
# project again passing --syntax sass
# preferred_syntax = :sass
# and then run:
# sass-convert -R --from scss --to sass
require 'fileutils'
on_stylesheet_saved do |file|
  if File.exists?(file) && File.basename(file, ".css").end_with?('scss')
    puts "Moving: #{file}"
    FileUtils.mv(file, File.dirname(file) + File.basename(file, ".scss") + ".css")
  end
end
```



- Pour les polices, choix du chemin avec **@font-face**.

SASS

```
@mixin texteLeger($grosueur : 14px){
  @include font-face("PacificoRegular", font-files("pacifico/Pacifico-
webfont.woff", "pacifico/Pacifico-webfont.ttf", "pacifico/Pacifico-
webfont.svg"), "pacifico/Pacifico-webfont.eot");
  font-size: $grosueur
}
```

CSS

```
@font-face {
  font-family: "PacificoRegular";
  src: url('polices/pacifico/Pacifico-webfont.eot');
  src: url('polices/pacifico/Pacifico-webfont.eot?#iefix') format('eot'),
  url('polices/pacifico/Pacifico-webfont.woff') format('woff'),
  url('polices/pacifico/Pacifico-webfont.ttf') format('truetype'),
  url('polices/pacifico/Pacifico-webfont.svg') format('svg');
}
```

<http://blog.grayghostvisuals.com/compass/font-face-helper/>

COMPASS : Le format de sortie

```
# You can select your preferred output style here (can be overridden via the command line):
output_style = :expanded or :nested or :compact or :compressed

# To disable debugging comments that display the original location of your selectors. Uncomment:
line_comments = false

# To enable relative paths to assets via compass helper functions. Uncomment:
relative_assets = true
```

Le fichier config.rb permet également de déterminer le format CSS final.

output_style = :expanded / line_comments = true

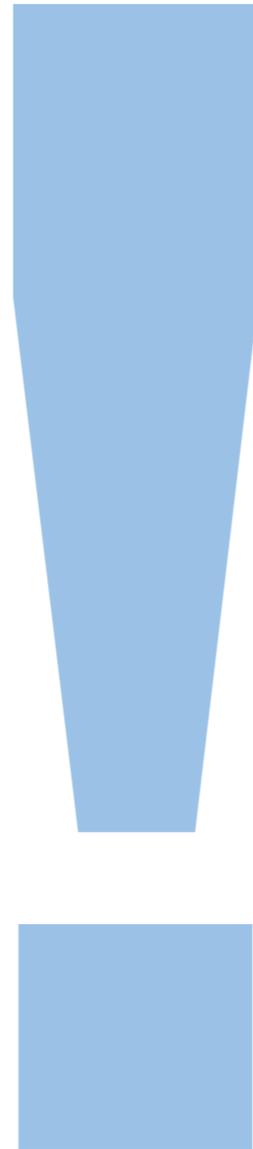
```
/* line 9, ../sass/composants/_grids.scss */
#container {
  width: 1003px;
  margin-left: auto;
  margin-right: auto;
  padding-top: 50px;
  margin-top: 0;
  z-index: 100000;
  overflow: hidden;
  *zoom: 1;
}
/* line 12, ../Library/Ruby/Gems
#container:after {
  content: " ";
  display: block;
  clear: both;
}
/* line 15, ../sass/composants/_grids.scss */
```

output_style = :compressed / line_comments = false

```
#container{width:1003px;margin-left:auto;margin-right:auto;
index:100000;overflow:hidden;*zoom:1}#container:after
";display:block;clear:both}#container.frontale{margin
nouvelles,#container.temoignages,#container.evenement
demieGauche{width:493px;float:left;margin-right:17px}
demieDroite{width:493px;float:right;margin-right:0}#c
right:17px}#colD{width:323px;float:right;margin-right
family:'quicksandlight';src:url("../polices/quicksand
quicksand-light-webfont.eot?#iefix") format("embedde
light-webfont.woff") format("woff"),url("../polices/c
format("truetype"),url("../polices/quicksand-light-we
format("svg");font-weight:normal;font-style:normal}@f
family:'quicksanditalic';src:url("../polices/quicksar
/quicksand-italic-webfont.eot?#iefix") format("embedd
italic-webfont.woff") format("woff"),url("../polices/
format("truetype"),url("../polices/quicksand-italic-w
format("svg");font-weight:normal;font-style:normal}@f
family:'quicksandbold_italic';src:url("../polices/qui
eot");src:url("../polices/quicksand-bolditalic-webfor
,url("../polices/quicksand-bolditalic-webfont.woff")
bolditalic-webfont.ttf") format("truetype"),url("../p
svg#quicksandbold_italic") format("svg");font-weight:
```

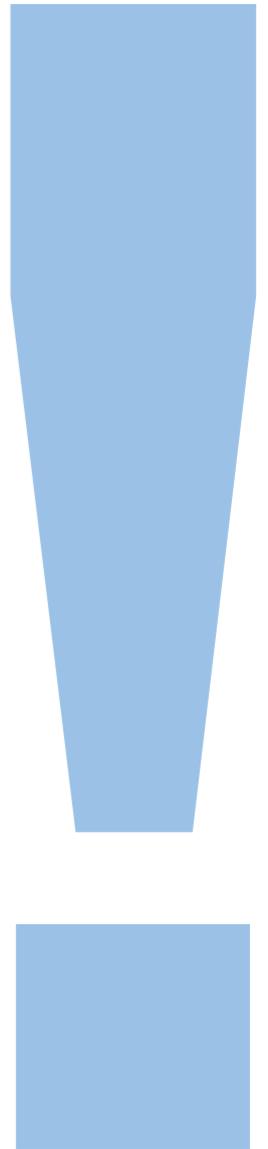


Conclusion



Conclusion

- On est tous devenus « DRY »
- On optimise son code
- On s'organise
- On se fait comprendre
- On lègue à la pérennité un code intelligible
- On respecte plus rapidement le cahier des charges
- Le département de l'Assurance qualité a moins de boulot
- On aime ça d'même



Bon code

