SASS

Sass est un pre-processeur css (code dynamique à compiler).Il represente une extension au CSS.
Ses plus grands avantages sont le dynamisme qui permet d'eviter la répétition du code.
Les fichier sass ont l'extension scss.

Comment ca marche ?

Puisque le navigateur ne comprend pas le sass, on a besoin d'un pre-processeur qui compile le sass et en genere le CSS. Ce processus est appelé le transpiling.

Installation de SASS

Aller à la page
sass-lang.com/install
On nous demande d'exécuter la commande suivante (il faut installer node avant)

```
npm install -g sass
```

dans le dossier css, créer un fichier styles.scss (vide pour l'instant) ➔via la console, se positionner dans le dossier de son projet ➔lancer ma commande

```
sass --watch styles.scss styles.css
```

➔ écrire son code sass dans styles.scss ➔ le transpileur surveille les fichiers source et destination et compile automatiquement des qu'il détecte le moindre changement.

Code SASS

Les variables

```scss
$arriere: skyblue;
$couleur: darkblue;
$font: 18px;

/* Use the variables */
body {
  background-color: $arriere;
  color: $couleur;
  font-size: $font;
}
```

Portée des variables

```scss
$tailleH1: 3rem;

h1 {
  $tailleH1: 2rem;
  color: $couleur;
  font-size: $tailleH1;
}

p {
  color: $couleur;
}
```

➔Voir ce que ca donne dans le css généré

Règles d'imbrication

L'imbrication se fait dans le meme ordre que le HTML

Code de champ modifié

```
 <section>
        <h1>Titre de la section</h1>
        <p>
            Lorem ipsum, dolor sit amet consectetur adipisicing elit. Ab libero
quae excepturi minima quasi laborum, sint odit fuga nulla ipsam? Excepturi
consectetur hic illum rerum quibusdam dolorem magnam. Reiciendis, ipsa!
        </p>
        <a href="https://google.com">Google</a>
    </section>
```

```
section {
    h1 {
        font-size: $tailleH1;
    }
    p {
        color: $couleur;
        padding : 1rem;
    }
    a{
        text-decoration: none;
    }
}
```

Inclusion de fichiers

Pour des raisons d'organisation et d'optimisation, le code sass peut etre écrit dans différents fichiers séparés l'un des autres.

Le code est écrit par morceaux dans divers fichiers tels que reset.scss, var.scss, couleurs.scss … etc

En css, on peut inclure un fichier externe via la directive @import mais l'inconvinient de cette directive est qu'à chaque fois qu'on appelle le fichier principal, on fait autant de requetes supplémenataires que de fichiers inclus dedans.

Pour remédier à ca, sass permet de diviser le fichier scss principal en sous parties, puis il les genere tous dans un seul.

Créer un fichier partie.scss et y mettre

```
h2 {
    padding: 1.5rem;
    background: greenyellow;
}
h3 {
    background: aqua;
}
h4 {
    color: red;
}
```

➔ intégrer celui-ci dans styles.scss

```
@import "partie";
```

➔regarder le css généré ➔ ajouter des elements dans le html correspondant aux selecteurs et voir leur application

@mixin et @include

Il s'agit de la création des parties de code réutilisable et les utiliser dans d'autres parties (dans le meme fichier ou dans d'autres qui sont inclus).

Dans le fichier partie :

```
@mixin description {
    font-size :1.5rem;
    color : gray;
    border-bottom : 1px solid gray;
}
```

Dans styles.scss :

```
p.description {
 @include description;
}
```