Lorsque vous codez, pensez à adopter le DRY : ne vous répétez pas !

Lorsque vous créez des sélecteurs, faites attention à ne pas y entasser trop de règles. Séparer les sélecteurs par leur fonction vous aidera à garder votre code compréhensible et maintenable.

La spécificité permet de mesurer le poids d’un sélecteur au sein du HTML parmi les autres.

Il existe quatre niveaux de spécificité :

les feuilles de style locales (inline styles) ;

les id ;

les classes, pseudoclasses et attributs ;

les éléments et pseudoéléments.

Votre navigateur applique les éléments les plus spécifiques en premier.

Le CSS n’a pas de structure rigide, ce qui complique la tâche pour avoir une codebase propre et maintenable.

En créant et en appliquant votre propre structure, vous pouvez mettre de l’ordre dans un CSS brouillon...

BEM signifie bloc, élément, modificateur :

les blocs sont des bouts de code autonomes ;

les éléments sont les parties qui forment le bloc ;

les modificateurs changent l’apparence ou le comportement d’un bloc ou d’un élément.

Les blocs sont nommés en fonction de leur rôle :

les éléments indiquent le nom de leur bloc parent, suivi d’un dunder (\_\_) puis du rôle de l’élément : form\_\_label.

Les modificateurs utilisent le nom du bloc ou de l’élément qu’ils modifient, suivi de deux tirets (--) et de ce que le sélecteur modifie : button--green.

Vous n’avez pas besoin d’attribuer à chaque élément d’une page web un sélecteur de classe. Vous pouvez utiliser les sélecteurs pour que votre code HTML soit plus propre et concis.

Les préprocesseurs CSS vous permettent d’imbriquer votre code (on appelle ça le nesting) pour créer une hiérarchie plus facile à lire et regrouper des morceaux de code entre eux.

Il existe plein de préprocesseurs, mais le plus courant est Sass, qui signifie Syntactically Awesome Style Sheets (“Feuilles de style syntaxiquement fantastiques”).

Au-delà du nesting, les préprocesseurs vous permettent d’utiliser des fonctionnalités de programmation pour créer une codebase plus maintenable, tout en diminuant les énormes quantités de code à écrire.

La syntaxe .scss est très similaire à la syntaxe CSS.

On peut utiliser du code CSS préexistant dans Sass en l’intégrant dans un fichier .scss.

La syntaxe .sass est plus concise mais .scss  reste plus couramment utilisé.

Le nesting permet d’imbriquer les sélecteurs comme en HTML.

Dans Sass, les sélecteurs peuvent être imbriqués, ou nestés, dans d’autres sélecteurs.

Sass peut créer tous les combinateurs CSS.

Utilisez l’esperluette pour lier un sélecteur imbriqué à un sélecteur parent sans recourir à un combinateur.

Attention à ne pas créer des sélecteurs trop spécifiques.