

Ce qu'il faut et ne pas faire en CSS - Partie 1 : La sélection CSS

par Alen Grakalic ([CSS Globe](#))

Date de publication : 09/06/2008

CSS Globe démarre une série de petits articles appelés "Ce qu'il faut et ne pas faire en CSS". Ces séries ont pour but de pointer les mauvaises habitudes lorsqu'on utilise le CSS et en général les standards du Web. On tentera de répondre aux questions les plus fréquentes relatives au CSS. Cette première partie portera sur la sélection CSS.

Introduction.....	3
Classes & Id.....	3
Les classes multiples.....	3
Les sélecteurs de types.....	3
Les sélecteurs descendants.....	4
Les combinaisons.....	4
Support multi-navigateur et multi-plateforme.....	4
Résumé.....	4
Remerciements.....	4

(1) (2)

Introduction

La sélection d'un élément est le processus de base pour appliquer un style en CSS. Si vous ne ciblez pas un certain élément vous ne pourrez pas le styliser, n'est-ce pas ? Cet article peut vous aider à corriger des erreurs que vous auriez pu commettre mais aussi à répondre aux questions relatives aux meilleures pratiques pour l'utilisation des sélecteurs en CSS.

Comme vous le savez, il y a plusieurs **types de sélecteurs CSS**. Nous nous focaliserons sur leur utilisation pratique.

Classes & Id

```
<p class="signature">...</p>
<ul id="nav">...</ul>
```

N'utilisez pas des noms de classes trop détaillés comme "red", "blueDot" ou "roundedTop". Je n'utiliserai même pas de noms comme "left", "right" ou "large". Pourquoi ? Eh bien, imaginez que vous ayez conçu 2 colonnes sur votre site, une colonne nommée "left" (gauche) et l'autre nommée "right" (droite) et qu'à un moment vous deviez intervertir la position des colonnes. Votre client pourrait ne pas aimer votre thème bleu et vouloir le changer, eh bien vous êtes coincé avec un bouton de classe "blueSmall" (class="blueSmall")...

Le nom des classes devraient donc décrire le sens et le but du contenu mais pas son apparence. Ainsi au lieu d'utiliser un **div** avec **id="left"** ajouter un **div** avec **class="main"** voire même **class="primary"**. Essayez le même nom de classe sur différents éléments, dans diverses situations. Les noms n'ont pas besoin d'être liés à un seul type d'élément. Vous pouvez utiliser **class="first"** sur le premier élément du menu ainsi que le premier paragraphe de votre contenu. Lorsqu'il faut choisir entre un nom de classe ou un id, déterminez si l'élément à styliser est vraiment unique. Dans ce cas, utilisez : **id**. S'il n'est pas unique utilisez les classes.

Cela vous évite d'utiliser plein d'id uniquement pour vos styles.

Les classes multiples

```
<button class="image submit">
```

Les noms de classes multiples sur un seul élément peuvent être une bonne façon d'optimiser votre css et d'éviter des noms de classes complexes pour le balisage.

J'utilise fréquemment une définition de classe, **class="image"**, qui possède un style général pour la technique de remplacement d'images. Puis j'ajoute des caractéristiques aux autres classes où je décris l'image à utiliser, je définis la largeur *etc.*

Les sélecteurs de types

```
a color:#ff0;
```

J'utilise toujours les sélecteurs de types au début de mes CSS, pour définir ou réinitialiser le style par défaut de certains éléments. Personnellement je ne crois pas à la puissance du sélecteur universel (à moins d'être utilisé en bas de hiérarchie). Si vous prévoyez d'utiliser des resets, utilisez ceux appropriés, c'est-à-dire **le reset d'Eric Meyer**.

Les sélecteurs descendants

Utilisez-les ! Quand vous les maîtriserez vous trouverez que votre travail est beaucoup plus aisé contrairement à l'utilisation d'une centaine de noms de classes dans chaque partie de votre document. J'utilise fréquemment quelque chose comme **#main p a** pour rajouter un lien souligné que j'avais enlevé dans la réinitialisation.

Les combinaisons

```
#main h2.title span
```

Ce type de sélecteur vous permet de cibler avec précision un élément.

Le principe que j'utilise est le suivant : J'essaye de déterminer le conteneur le plus petit dans la hiérarchie qui possède des éléments non répétitifs. J'ajoute un nom de classe à ce conteneur et j'y indique les éléments cibles avec des sélecteurs de même type que ceux décrits plus haut. Si vous avez une balise **h2** que vous utilisez pour les titres, et que celle-ci possède seulement un span, il n'y a aucune utilité à ajouter un nom de classe à ce span. Vous pouvez la cibler avec un sélecteur de combinaison, comme vu précédemment.

Support multi-navigateur et multi-plateforme

Certains sélecteurs ne sont pas pris en charge par certains navigateurs, vous ne devez donc pas compter sur eux. C'est-à-dire, que vous ne pouvez compter que sur le pseudo-élément `":first-child "` pour avoir les coins du haut arrondis, sachant qu'ils ne sont pas supportés par IE6. En vérité de nombreux utilisateurs utilisent toujours IE6 et il faut vivre avec ça.

Résumé

À faire

- Lorsque vous choisissez d'utiliser des noms de classes, utilisez des noms qui décrivent le sens et le but du contenu, et pas son apparence.
- Utilisez quelques noms de classes et utilisez-les sur divers éléments.
- Utilisez des sélecteurs de type pour (ré)initialiser le style par défaut. Si vous le souhaitez, utilisez des techniques de réinitialisation appropriées (mentionnées ci-dessus).
- Utilisez des noms de classes multiples sur un seul élément pour optimiser votre CSS.
- Utilisez des combinaisons de sélecteurs et ciblez précisément votre élément.

À ne pas faire

- Utiliser pas des noms de classes comme "red", "blueDot" *etc.*
- Utiliser pas des noms de classes compliqués ou des id comme "sideIntroSecondaryContentTop". Utilisez plutôt des sélecteurs descendants.
- Utiliser pas de sélecteur universel pour la réinitialisation du document.
- Utiliser pas des sélecteurs qui sont dépendant des navigateurs.
- Utiliser pas des noms de classes qui ne sont pas indispensables.

Remerciements

Tous mes remerciements à **RideKick** pour sa relecture.

1 :
Cet article a été publié à l'origine en anglais à l'adresse suivante : **CSS do's and dont's Part 1: CSS Selecting**

2 :
Article traduit par l'équipe de traduction de la rédaction de developpez.com