

CSS : Les sélec- teurs

Contenu

Sélecteurs de base.....	2
Le nom d'une balise HTML :	2
Le nom d'une classe (commence par un point) :	2
L'ID de l'élément :	3
Le sélecteur universel	4
Sélection multiple.....	5
Autres sélecteurs	6
Les pseudo-classes	7
Les pseudo-classes pour les liens	7
Autres pseudo-classes courantes	7
Les pseudo-éléments.....	8
Exercices	9
A retenir.....	10

Sélecteurs de base

Considérons la déclaration de style suivante :

```
body {  
    background-color:# FFFFCC;  
    margin-left: 100px;  
}  
  
h1 {  
    font-size: 48pt;  
    color :#FF0000;  
    font-style: italic;  
    border-bottom: solid thin black;  
}  
  
p {  
    font-size: 12pt;  
    line-height: 14pt;  
    font-family: Arial;  
    letter-spacing: 0.2mm;  
    word-spacing: 0.8mm;  
    color: blue;  
}
```

On déclare des sélecteurs : ici *body*, *h1* ou *p*. Par "sélecteur" on entend ce qui se trouve devant les accolades. Le sélecteur désigne quelle balise sera affectée par les propriétés qui suivent. Les sélecteurs peuvent être une (ou plusieurs) balises HTML.

Le sélecteur peut être le nom d'une balise HTML, le nom d'une classe ou encore l'identifiant d'un élément HTML défini par l'attribut `id`. Enfin, il existe un sélecteur universel.

Le nom d'une balise HTML :

`body h1 p`

C'est le cas de l'exemple ci-dessus. `h1 {...}` : concerne toutes les balises `<h1>`.

Le nom d'une classe (commence par un point) :

`.fondrouge .cadrevert`

On peut aussi définir des sélecteurs qui nous sont propres (et que l'on appellera par l'attribut `class`). Ces sélecteurs ne concernent pas forcément une balise ou un ensemble de balise, mais uniquement les éléments qui seront marqués comme devant appliquer le style défini. Le marquage se fait par l'attribut HTML **class**.

Par exemple :

```
<html>
<head>
  <style>
    .one {
      color: red;
    }
  </style>
</head>
<body>
  <h3 class="one">Titre rouge</h3>
  <h3>Titre normal</h3>
  <p class="rouge">Texte écrit en rouge</p>
</body>
</html>
```

Dans cet exemple, on définit un format nommé *one*, qui va s'appliquer à toutes les balises.

Toutes les balises dans lesquelles on retrouve l'attribut `class="one"` appliqueront le style défini.

L'ID de l'élément :

#rouge

Dans ce cas, le style s'applique à la balise portant un identifiant **unique** correspondant au sélecteur (sans le dièse). En HTML l'attribut `id` permet de spécifier le nom de l'identifiant.

```
<html>
<head>
  <style>
    #one {
      background-color: silver;
    }
  </style>
</head>
<body>
  <h3 id="one">Titre rouge</h3>
  <h3>Titre normal </h3>
  <p class="one">Texte écrit en rouge</p>
</body>
</html>
```

Attention : dans une page web, un attribut `id` doit porter un nom unique (valeur). Il ne peut donc y avoir plusieurs balises portant le même nom d'id.

Par contre, on remarque dans l'exemple qu'un id peut porter le même nom qu'une classe : `#one` et `.one`. Toutefois, dans la pratique, ceci est à éviter afin d'éliminer tout risque de confusion.

Le sélecteur universel

L'astérisque (*) est le sélecteur universel en CSS.

S'il est utilisé seul, comme dans l'exemple ci-dessous, il correspond à un élément de n'importe quel type et s'applique donc à tous les éléments de votre page web.

```
* {  
  color: red;  
}
```

Mais le sélecteur universel est surtout employé pour sélectionner tous les éléments d'un même type :

```
*.p {  
  color: red;  
}
```

Cet exemple sélectionne toutes les balises `<p>` pour leur appliquer la couleur rouge.

Si le sélecteur universel concerne un élément unique, il devient alors facultatif :

- `*.warning` et `.warning` sont équivalents.
- `*#myid` et `#myid` sont équivalents.

Sélection multiple

Il est possible de combiner plusieurs sélecteurs, ce qui permet d'appliquer aux différents éléments sélectionnés les mêmes instructions CSS. Les instructions CSS s'appliqueront alors à tous les éléments sélectionnés.

Sélecteur	Exemple
Plusieurs balises HTML	<pre>p, h1, div { color : red; font-size : 12%; border-bottom: solid thin black; }</pre>
Plusieurs classes	<pre>.classe1, .classe2, .classe3 { color : red; font-size : 12%; border-bottom: solid thin black; }</pre>
Plusieurs id	<pre>#one, #two, #three { color : red; font-size : 12%; border-bottom: solid thin black; }</pre>

Il est aussi possible de combiner des sélecteurs de différents types : c'est-à-dire à la fois des balises HTML, des classes et des ID :

```
h1, classe1, #one {  
  color: red;  
}
```

Autres sélecteurs

A titre d'information, sachez que les sélecteurs suivants existent également :

- des sélecteurs d'enfants (signe « > ») qui permettent de cibler des éléments fils
- des combineurs d'adjacence (signe « + ») qui permettent de cibler des éléments frères (= voisins)
- des sélecteurs d'attributs (signe « [] ») qui permettent de cibler une balise HTML en fonction de la présence d'un attribut ou de la valeur donnée à un attribut

Les pseudo-classes

Une **pseudo-classe** est un mot-clé qui peut être ajouté à un sélecteur afin d'indiquer l'**état** (= statut) spécifique dans lequel l'élément doit se trouver pour être ciblé par la déclaration.

Une pseudo-classe commence par le signe « : » et suit immédiatement le sélecteur :

Exemple

```
a: hover {  
    color: red;  
}
```

Dans cet exemple, le texte des liens deviendra rouge au survol de la souris.

Une pseudo-classe ne s'applique pas forcément à toutes les balises HTML, certaines sont spécifiques à un seul élément. Par exemple les pseudo-classes les plus utilisées concernent les liens (comme dans l'exemple ci-dessus) ou le curseur de la souris (changement de la forme du curseur).

△ Certaines pseudo-classes se terminent par des parenthèses, par exemple :`nth-child()`.

△ Certaines pseudo-classes ne sont pas interprétées par tous les navigateurs web. Il est donc nécessaire de s'assurer de leur compatibilité. Les pseudo-classes évoquées dans ce document sont toutes compatibles avec les navigateurs courants.

Les pseudo-classes pour les liens

Nom de la pseudo-classe	Rôle
:active	lien actif = en train d'être cliqué
:focus	lien qui devient actif (focus)
:hover	lien au survol de la souris
:link	lien non encore visité (état par défaut d'un lien, affiché en bleu)
:visited	lien déjà visité (s'affiche par défaut en violet)

Autres pseudo-classes courantes

Nom de la pseudo-classe	Rôle
:not(balise html)	sélectionne toutes les balises HTML à l'exclusion de celle(s) passée(s) en argument
:first-child	Sélectionne le premier fils de l'élément HTML
:last-child	Sélectionne le dernier fils de l'élément HTML
:nth-child(numéro)	Accès à l'élément fils du numéro passé en argument

➔ [Documentation des pseudo-classes](#)

Les pseudo-éléments

Un **pseudo-élément** est un mot-clé ajouté à un sélecteur qui permet de mettre en forme certaines parties de l'élément ciblé.

Un pseudo-élément est précédé du signe « :: » (deux points-virgules, Δ ne pas confondre avec le « : » d'une pseudo-classe).

Exemple

```
a::before {  
  content: "*";  
}
```

Ici, le signe « * » sera ajouté avant chaque lien.

Les pseudos-éléments les plus rencontrés sont *::before* et *::after*, permettant d'ajouter du contenu avant ou après un élément HTML.

Δ Certaines pseudo-classes ne sont pas interprétées par tous les navigateurs web. Il est donc nécessaire de s'assurer de leur compatibilité.

→ [Documentation des pseudo-éléments](#)

Exercices

Créer un fichier HTML et un fichier CSS et testez les différents exemples de ce cours.

A retenir

En CSS, les sélecteurs permettent de cibler les éléments sur lesquels on souhaite travailler.

Il existe 3 types majeurs de sélecteurs : les balises, HTML, les classes et identifiants (« id »).

Ceux-ci peuvent être mélangés (sélection multiple).

Des instructions CSS plus poussées (combinateurs, pseudo-classes ou pseudo-éléments) complètent l'utilisation des sélecteurs pour des ciblage plus précis.