



LANGAGE CSS

Les CSS (Cascading Style Sheets, c'est à dire "feuilles de styles en cascade") permettent de définir la présentation des pages Web. Le code HTML servira à définir la structure du contenu tandis que les CSS permettront de créer les styles et la mise en page du site.

En clair, le HTML s'occupe du FOND, et le CSS de la FORME !

Mieux qu'un discours, un exemple : visitez le site <http://www.csszengarden.com/> et choisissez des styles différents...

Le code HTML de ces pages est strictement le même ! Seul le code CSS change et s'occupe de la mise en page !

Convaincu ? Passons maintenant à la pratique...

I. Avant toute chose...

- Création d'un fichier style.css
- Dans le code HTML, à l'intérieur de la balise HEAD, insérer le code suivant :

```
<link rel="stylesheet" href="style.css" />
```

II. Un exemple de code CSS

```
p
{
    color : blue ;
}
```

Ce code indique au navigateur que lorsqu'il interprète une balise HTML <p>, il doit appliquer la couleur bleue à la police.

En clair, le code CSS ressemble au schéma suivant :

```
sélecteur
{
    attribut : valeur ;
}
```



III. Grouper les sélecteurs

Si certaines propriétés s'appliquent à plusieurs sélecteurs, il est possible de les grouper. Ainsi, plutôt que d'écrire :

```
h1
{
    font-family:"sans serif";
    color: blue;
}
h2
{
    font-family:"sans serif";
    color: blue;
}
```

Il est possible, et plus rapide, de grouper les sélecteurs h1 et h2 et d'écrire :

```
h1,h2
{
    font-family:"sans serif";
    color: blue;
}
```

IV. Le sélecteur class

Avec le sélecteur de classe il est possible d'appliquer des styles différents pour des éléments HTML d'un même type. Par exemple, si nous souhaitons que certains paragraphes (<p>) soient affichés en rouge (disons, pour mettre en évidence des paragraphes importants), et que d'autres paragraphes aient un alignement centré, nous pouvons définir deux classes pour chacun de types de paragraphe :

```
p.introduction {text-align:center}
p.important {color:red}
```

Dans le document HTML, l'attribut class permet de choisir la classe qui sera utilisée :

```
<p class="introduction">Un paragraphe d'introduction</p>
<p class="important">Un paragraphe important</p>
```

Pour définir une classe qui s'applique à tous les éléments HTML, et pas uniquement à un seul élément HTML (comme <p> dans l'exemple ci-dessus), on utilise la syntaxe suivante : l'élément HTML n'est pas écrit ; on commence directement par un point, suivi du nom de la classe.

Exemple :

```
.important {color:red}
```

Ainsi, la classe important pourra être utilisée pour tous les éléments HTML :

```
<h1 class="important">Un titre important</h1>
<p class="important">Un paragraphe important</p>
```



V. Le sélecteur id

Le sélecteur id a presque la même fonction, à la différence importante qu'on ne peut l'utiliser **qu'une seule fois dans la page**, contrairement au sélecteur class. C'est pour cela qu'il est plutôt utilisé à la mise en page qu'à la mise en forme de caractères.

Exemple : construisons une section de page à deux colonnes. L'une des deux colonnes servira pour un menu de 100 pixels de large flottant à gauche, l'autre pour un contenu placé à 110 pixels du bord gauche (pour ne pas empiéter sur le menu bien sûr). Cette fois, ce n'est pas d'un point qu'il faudra faire précéder le nom du sélecteur id, mais d'un #.

```
#menu
{
    background-color:silver;
    width:100px;
    float:left;
}

#contenu
{
    margin-left:110px;
}
```

Dans le code html, il faudra donc cette fois indiquer un id (id="menu" et id="contenu") à la place de la syntaxe class.

```
<h1 id="haut">Exemple des sélecteurs "class" et "id"</h1>

<div id="menu">

<ul>
    <li> item 1 </li>
    <li> item 1 </li>
    <li> item 1 </li>
    <li> item 1 </li>
    <li> item 1 </li>
</ul>

</div>

<div id= "contenu" >

    <p>
        Blabla...
        <a href = "#haut#"> Haut de page </a>
    </p>
</div>
```



Il n'aura pas échappé à votre sagacité qu'un `id` non déclaré dans les styles est venu se glisser dans la balise de titre `h1` (`id="haut"`).

C'est le deuxième effet `id` ! En effet, cet attribut peut servir d'ancre. Ainsi, le lien interne de haut de page y est rattaché directement. Cela explique aussi pourquoi on ne peut utiliser qu'un seul nom de `id` par page.

VI. Conclusion

Je vous laisse trouver des liens intéressants vers des pages d'explications du code CSS, de ses attributs, de ses valeurs...etc.

Je termine en vous donnant l'adresse du validateur de votre code CSS :

<http://jigsaw.w3.org/css-validator/>