

2 : LES ID ET LES CLASS, QUE CHOISIR ENTRE LES DEUX ?

I. Présentation des attributs HTML class et id et cas d'utilisation

Explications :

Les attributs HTML **class** et **id** sont des attributs dits globaux car on va pouvoir les ajouter dans la balise ouvrante de n'importe quel élément HTML.

Ces deux attributs vont être principalement utilisés dans un but de mise en forme : ils vont nous servir à appliquer des styles CSS aux éléments qui vont les contenir.

En effet, à la différence d'un attribut **href** par exemple, les attributs **class** et **id** ne vont pas servir à préciser le fonctionnement d'un élément HTML mais vont simplement être très utiles pour cibler un élément précisément.

Nous allons effectivement très facilement pouvoir nous resservir de ces deux attributs en CSS grâce aux sélecteurs associés **.class** et **#id**.

II. Premier exemple d'utilisation des attributs HTML class et id et des sélecteurs CSS associés

Voyons immédiatement de manière pratique comment vont fonctionner les attributs **class** et **id** et comment on va pouvoir les utiliser en CSS pour cibler et appliquer des styles particuliers à des éléments choisis.

Pour cela, nous allons créer une page HTML et allons placer des attributs **class** et **id** dans différents éléments.

Nous allons déjà devoir renseigner une valeur pour chaque attribut **class** et **id**. Les valeurs indiquées pour les attributs ne doivent contenir ni caractères spéciaux ni espaces et commencer par une lettre. En pratique, on essaiera d'attribuer des valeurs qui font sens à nos différents attributs.

Notez déjà que chaque **id** doit avoir une valeur propre ou unique dans une page. En revanche, on va pouvoir attribuer la même valeur à plusieurs attributs **class** différents.

Voyons maintenant cela avec un exemple en page suivante :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Cours HTML et CSS</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="cours.css">
  </head>

  <body>
    <h1 id="orange">Un titre de niveau 1</h1>
    <p class="bleu">Un premier paragraphe</p>
    <p class="bleu">Un autre paragraphe avec un <a href="#">lien</a></p>
    <ul>
      <li>Elément de liste 1</li>
      <li class="vert">Elément de liste 2</li>
    </ul>
    <p id="gros">Un troisième paragraphe</p>
  </body>
</html>
```

index.html

```
#orange{
  color: orange;
}

#gros{
  font-size: 24px;
}

.bleu{
  color: blue;
}

.vert{
  color: green;
}
```

cours.css



Ici, on ajoute un attribut **id="orange"** dans la balise ouvrante de notre élément **h1** et un attribut **id="gros"** à notre dernier paragraphe. On ajoute également un même attribut **class="bleu"** à nos deux premiers paragraphes et un attribut **class="vert"** à un élément de notre liste.

Ensuite, on va lier des styles CSS à ces différents **id** et **class** en utilisant les sélecteurs CSS associés.

Pour cibler un **id** en particulier en CSS, on utilisera le symbole dièse **#** suivi de la valeur de l'**id** auquel on souhaite lier des styles.

Pour cibler une **class** en particulier en CSS, on utilisera le symbole point **.** suivi de la valeur de la **class** à laquelle on souhaite lier des styles.

III. Class vs id : Quelles différences et quel attribut utiliser ?

Il existe une différence notable entre les deux attributs **class** et **id** : chaque **id** doit avoir une valeur unique dans une même page tandis que plusieurs attributs **class** peuvent partager la même valeur.

Cela fait que l'attribut **id** est beaucoup plus spécifique que l'attribut **class** et que ces deux attributs vont avoir des rôles et buts différents notamment pour la mise en forme CSS.

Utilisation des classes en HTML et en CSS :

Ainsi, nous utiliserons généralement des attributs **class** pour définir des styles généraux et communs à plusieurs éléments dans une même page. Comme nous pouvons donner une même **class** à plusieurs éléments, ils hériteront tous des mêmes styles sauf en cas de conflit (c'est-à-dire dans le cas où le comportement d'une même propriété a déjà été défini en CSS) bien évidemment.

Toute la puissance des attributs **class** et du sélecteur CSS associé va résider dans le fait qu'on va tout à fait pouvoir définir des styles CSS généraux liés à des sélecteurs **.class** avant même de commencer à écrire notre code HTML. Nous n'aurons ensuite plus qu'à fournir les attributs **class** à nos éléments HTML lors de la création de la page.

L'idée va être la suivante : créer des styles CSS et les attacher à des sélecteurs **.class** à priori puis attribuer les attributs **class** relatifs à certains éléments HTML choisis afin que les styles CSS correspondants leur soient appliqués.

Cette façon de procéder peut sembler étrange et "à l'envers" pour les personnes non expertes. Cependant, je vous garantis que c'est une très bonne façon de faire qui peut faire gagner énormément de temps pour un gros projet. C'est par ailleurs toute l'idée derrière l'utilisation de la librairie Bootstrap par exemple.

Utilisation des id en HTML et en CSS :

En revanche, comme chaque **id** doit être unique dans une page, nous utiliserons ce sélecteur pour appliquer des styles très précis et pour être sûr de ne cibler qu'un élément HTML en CSS.

C'est la raison pour laquelle on se sert des attributs **id** pour créer des liens de type ancre par exemple. En effet, nous sommes sûrs de lever toute ambiguïté sur la sélection avec un **id** car encore une fois celui-ci doit être unique.

Les sélecteurs CSS **.class** et **#id** ne possèdent donc pas le même degré de précision et ainsi n'ont pas le même ordre de priorité dans les styles attribués aux éléments en cas de conflit. Je vous rappelle ici qu'en cas de conflit sur un style en CSS ce sont les styles du sélecteur le plus précis qui seront appliqués.

L'ordre de priorité d'application des styles en CSS est le suivant (du plus prioritaire ou moins prioritaire) :

1. Les styles liés à un sélecteur **#id** ;
2. Les styles liés à un sélecteur **.class**.
3. Les styles liés à un sélecteur élément ;

Plus d'exemples d'utilisation des attributs class et id en HTML et des sélecteurs CSS associés :

Avant tout, reprenez que les valeurs indiquées pour les attributs class et id ne doivent contenir ni caractères spéciaux ni espaces et commencer par une lettre. Idéalement, nous essaierons d'utiliser des noms qui font du sens pour nos attributs class et id.

On pourra par exemple utiliser des noms relatifs aux propriétés CSS définies avec les sélecteurs associés. Faites bien attention cependant à ne pas utiliser des noms protégés c'est-à-dire des noms déjà utilisés par le HTML et qui ont déjà une signification spéciale.

Attribuer un attribut class et un attribut id à un élément HTML :

On peut tout à fait fournir plusieurs attributs à un élément HTML et notamment un attribut class et un attribut id à un élément.

Reprenons l'exemple précédent en ajoutant un attribut class à notre dernier paragraphe pour illustrer cela :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Cours HTML et CSS</title>
    <meta charset="utf-8">
    <link rel="stylesheet" href="cours.css">
  </head>
  <body>
    <h1 id="orange">Un titre de niveau 1</h1>
    <p class="bleu">Un premier paragraphe</p>
    <p class="bleu">Un autre paragraphe avec un <a href="#">lien</a></p>
    <ul>
      <li>Elément de liste 1</li>
      <li class="vert">Elément de liste 2</li>
    </ul>
    <p class="bleu" id="gros">Un troisième paragraphe</p>
  </body>
</html>
```

```
#orange{
  color: orange;
}

#gros{
  font-size: 24px;
}

.bleu{
  color: blue;
}

.vert{
  color: green;
}
```

index.html } cours.css

Ici, le dernier paragraphe de notre page possède à la fois un attribut **class="bleu"** et un **id="gros"**. Les styles CSS liés à ces deux attributs donc être appliqués à l'élément.

Ici, nos deux attributs **class="bleu"** et **id="gros"** nous servent à appliquer des propriétés CSS différentes (**color** pour notre attribut class et **font-size** pour notre id). Il n'y a donc pas de risque de conflit.

En revanche, il y aurait eu conflit si on avait précisé des comportements différents pour la même propriété avec nos deux sélecteurs.

Résultat :



IV. Un point sur l'ordre de priorité d'application de styles CSS

Imaginons maintenant qu'on passe un attribut **class** et un attribut **id** à un même élément et qu'on définisse une même propriété CSS de manière différente pour ces **id** et **class** :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Cours HTML et CSS</title>
    <meta charset="utf-8">
    <link rel="stylesheet" href="cours.css">
  </head>
  <body>
    <h1 class="bleu" id="orange">Un titre de niveau 1</h1>
    <p class="bleu">Un premier paragraphe</p>
    <p class="bleu">Un autre paragraphe avec un <a href="#">lien</a></p>
    <ul>
      <li>Elément de liste 1</li>
      <li class="vert">Elément de liste 2</li>
    </ul>
    <p class="bleu" id="gros">Un troisième paragraphe</p>
  </body>
</html>
```

```
#orange{
  color: orange;
}

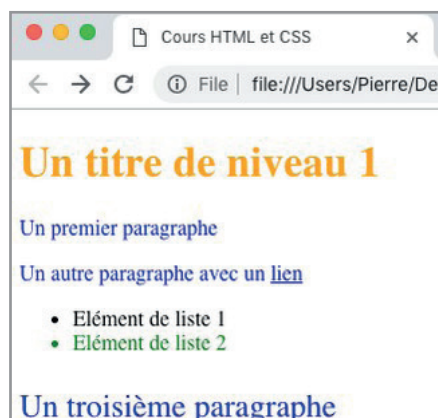
#gros{
  font-size: 24px;
}

.bleu{
  color: blue;
}

.vert{
  color: green;
}
```

index.html cours.css

Résultat :



Ici on passe un attribut **class="bleu"** et **id="orange"** à notre titre **h1**. Or, on définit le comportement de la même propriété (la propriété **color**) de manière différente dans les sélecteurs **.bleu** et **#orange**.

Il y a donc conflit sur les styles.

Comme vous pouvez le voir, notre titre s'affiche en orange ce qui signifie que ce sont les styles liés à l'**id** qui vont être pris en compte plutôt que ceux liés à la **class**.

Vous pouvez ici retenir la règle suivante dans l'application des styles CSS : ce seront toujours les styles liés au sélecteur le plus précis qui seront appliqués en cas de conflit.

Par « **précis** », on entend le sélecteur qui permet d'identifier le plus précisément l'élément auxquels vont être appliqués les styles.

Ici, comme chaque **id** doit posséder une valeur unique dans une page, le sélecteur CSS lié à notre **id** est très précis et beaucoup plus précis que le sélecteur lié à l'attribut **class** puisqu'il permet d'identifier un élément de manière unique alors qu'un attribut **class** peut être partagé par plusieurs éléments et ne permet donc pas d'identifier un élément en particulier.

Cette notion de précision peut vous sembler un peu floue pour le moment car c'est le genre de notion qu'il est difficile de comprendre sans connaître l'ensemble du langage. Pas d'inquiétude, tout cela va se préciser au fil des leçons et à chaque nouvelle notion que nous allons aborder.

V. Attribuer plusieurs attributs class à un élément HTML

L'un des grands intérêts de l'attribut HTML **class** est qu'un même attribut (et donc les styles CSS liés) va pouvoir être partagé par différents éléments. Cela facilite grandement la gestion des styles de nos fichiers HTML et nous permet de gagner beaucoup de temps.

Réciproquement, un même élément HTML va tout à fait pouvoir recevoir différents attributs. Pour cela, il va suffire d'indiquer les différentes valeurs séparées par un espace.

Ainsi, une très bonne pratique en CSS et pour la création d'un site va être de ne pas surcharger un sélecteur **.class** avec de nombreux styles CSS mais au contraire d'utiliser de multiples sélecteurs **.class** qui se contenteront de définir chacun un comportement ou plusieurs propriétés d'un même « type ».

Fonctionner comme cela permet d'avoir un code beaucoup plus clair et d'avancer beaucoup plus vite dans la création d'un site. Regardez plutôt l'exemple ci-dessous pour bien comprendre cette utilisation des attributs **class** :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Cours HTML et CSS</title>
    <meta charset= "utf-8">
    <link rel="stylesheet" href="cours.css">
  </head>

  <body>
    <h1 class="gros souligne orange">Un titre de niveau 1</h1>
    <p class="bleu">Un premier paragraphe</p>
    <p>Un autre paragraphe avec un <a href="#">lien</a></p>
    <ul>
      <li>Elément de liste 1</li>
      <li class="souligne">Elément de liste 2</li>
    </ul>
    <h2 class="gros italique bleu">Un titre de niveau 2</h2>
  </body>
</html>
```

index.html

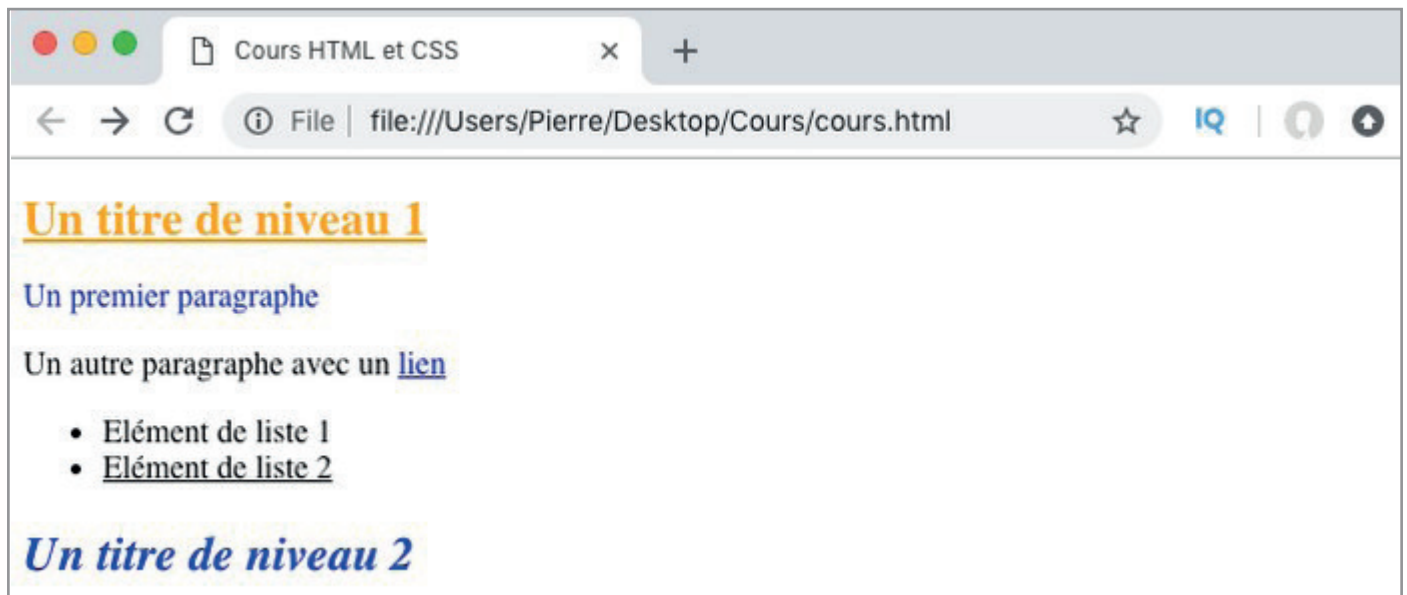
```

/*On commence par définir des styles CSS qu'on lie a des sélecteurs de
*class. On n'aura plus qu'à utiliser les attributs class correspondants
*en HTML ensuite pour appliquer les styles liés à nos éléments*/
.gros{
    font-size: 24px;
}
/*Texte en italique*/
.italique{
    font-style: italic;
}
/*Crée un trait de soulignement sous le texte*/
.souligne{
    text-decoration: underline;
}
.orange{
    color: orange;
}
.bleu{
    color: blue;
}
.vert{
    color: green;
}

```

cours.css

Résultat :



Notez que faire ceci avec des attributs `id` n'aurait aucun sens puisque les styles des sélecteurs liés n'ont pas vocation à être partagés entre différents éléments (c'est-à-dire à être « réutilisés ») mais sont liés à un élément en particulier et il est donc logique ici de placer tous les styles voulus pour l'élément dans un seul `id`.