**Appliquer un style : class et id**

Ce que je vous ai montré jusqu'ici a quand même un défaut : cela implique par exemple que TOUS les paragraphes possèdent la même présentation (ici, ils seront donc tous écrits en bleu).  
Comment faire pour que certains paragraphes seulement soient écrits d'une manière différente ? On pourrait placer le code CSS dans un attribut style  sur la balise que l'on vise (c'est la technique que je vous ai présentée un peu plus tôt) mais, comme je vous l'ai dit, ce n'est pas recommandé (il vaut mieux utiliser un fichier CSS externe).

Pour résoudre le problème, on peut utiliser ces attributs spéciaux *qui fonctionnent sur toutes les balises* :

* l'attribut class  ;
* l'attribut id  .

Que les choses soient claires dès le début : les attributs class  et id  sont quasiment identiques. Il y a seulement une petite différence que je vous dévoilerai plus bas.  
Pour le moment, et pour faire simple, on ne va s'intéresser qu'à l'attribut class  .

Comme je viens de vous le dire, c'est un attribut que l'on peut mettre sur n'importe quelle balise, aussi bien titre que paragraphe, image, etc.

<h1 class=""> </h1>

<p class=""> </p>

<img class="" />

Oui, mais que met-on comme valeur à l'attribut class  ?

En fait, vous devez écrire un nom qui sert à identifier la balise. Ce que vous voulez, du moment que le nom commence par une lettre.

Par exemple, je vais associer la classe introduction  à mon premier paragraphe (ligne 12) :

<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8" />

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

<title>Premiers tests du CSS</title>

</head>

<body>

<h1>Mon super site</h1>

<p class="introduction">Bonjour et bienvenue sur mon site !</p>

<p>Pour le moment, mon site est un peu <em>vide</em>. Patientez encore un peu !</p>

</body>

</html>

Maintenant que c'est fait, votre paragraphe est identifié. Il a un nom : introduction  . Vous allez pouvoir réutiliser ce nom dans le fichier CSS pour dire : « *Je veux que seules les balises qui ont comme nom 'introduction' soient affichées en bleu* ».

Pour faire cela en CSS, indiquez le nom de votre classe en commençant par un point, comme ci-dessous :

.introduction

{

*color*: blue;

}

Testez le résultat : seul votre paragraphe appelé introduction  va s'afficher en bleu (figure suivante) !

Seul le premier paragraphe s'affiche en bleu

Et l'attribut id  , alors ?

Lui, il fonctionne exactement de la même manière que class  , à un détail près : il ne peut être utilisé *qu'une fois* dans le code.

Quel intérêt ? Il y en a assez peu pour tout vous dire ; cela vous sera utile si vous faites du JavaScript plus tard, pour reconnaître certaines balises. D'ailleurs, nous avons déjà vu l'attribut id  dans le chapitre sur les liens (pour réaliser des ancres). En pratique, nous ne mettrons des id  que sur des éléments qui sont uniques dans la page, comme par exemple le logo :

<img src="images/logo.png" alt="Logo du site" id="logo">

Si vous utilisez des id  , lorsque vous définirez leurs propriétés dans le fichier CSS, il faudra faire précéder le nom de l' id  par un dièse (#) :

#logo

{

/\* Indiquez les propriétés CSS ici \*/

}

Je ne vous propose pas de le tester, cela fonctionne exactement comme class  .

Si vous vous emmêlez les pinceaux entre class  et id  , retenez que deux balises peuvent avoir le même nom avec l'attribut class  . Un nom d' id  doit en revanche être unique dans la page HTML.

**Les balises universelles**

Il arrivera parfois que vous ayez besoin d'appliquer une class  (ou un id  ) à certains mots qui, à l'origine, ne sont pas entourés par des balises.

En effet, le problème de class  , c'est qu'il s'agit d'un attribut. Vous ne pouvez donc en mettre que sur une balise. Si, par exemple, je veux modifier uniquement « bienvenue » dans le paragraphe suivant :

<p>Bonjour et bienvenue sur mon site !</p>

Cela serait facile à faire s'il y avait une balise autour de « bienvenue » , mais malheureusement, il n'y en a pas. Par chance, on a inventé… la balise-qui-ne-sert-à-rien.

En fait, on a inventé deux balises dites ***universelles***, qui n'ont aucune signification particulière (elles n'indiquent pas que le mot est important, par exemple). Il y a une différence minime (mais significative !) entre ces deux balises :

* <span> </span>  : c'est une balise de type **inline**, c'est-à-dire une balise que l'on place au sein d'un paragraphe de texte pour sélectionner certains mots uniquement. Les balises <strong>  et <em>  sont de la même famille. Cette balise s'utilise donc au milieu d'un paragraphe et c'est celle dont nous allons nous servir pour colorer « bienvenue » ;
* <div> </div>  : c'est une balise de type **block**, qui entoure un bloc de texte. Les balises <p>  ,  <h1>  , etc., sont de la même famille. Ces balises ont quelque chose en commun : elles créent un nouveau « bloc » dans la page et provoquent donc obligatoirement un retour à la ligne. <div>  est une balise fréquemment utilisée dans la construction d'un design, comme nous le verrons plus tard.

Pour le moment donc, nous allons utiliser plutôt la balise <span>  . On la met autour de « bienvenue », on lui ajoute une classe (du nom qu'on veut), on crée le CSS et c'est gagné !

<p>Bonjour et <span class="salutations">bienvenue</span> sur mon site !</p>

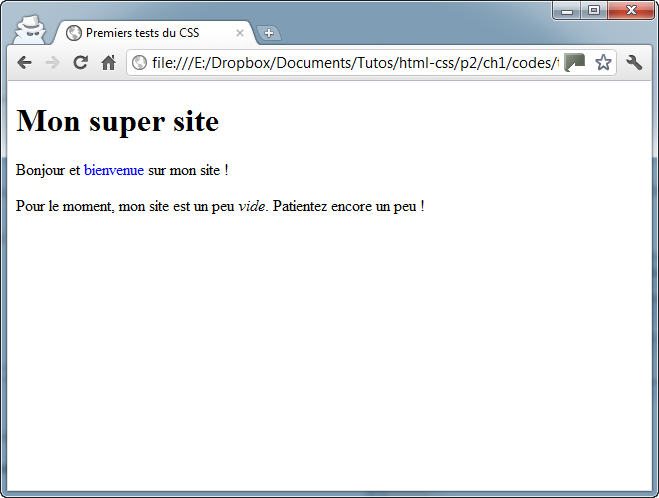
.salutations

{

*color*: blue;

}

Vous pouvez voir le résultat à la figure suivante.

Le mot « bienvenue » est écrit en bleu

**Appliquer un style : les sélecteurs avancés**

En CSS, le plus difficile est de savoir cibler le texte dont on veut changer la forme. Pour cibler (on dit « sélectionner ») les éléments de la page à modifier, on utilise ce qu'on appelle des ***sélecteurs***. Vous en avez déjà utilisé quelques-uns un peu plus tôt dans ce chapitre, résumons-les pour commencer.

**Les sélecteurs que vous connaissez déjà**

Ces sélecteurs, que nous avons vus précédemment, sont de loin les plus couramment utilisés. Il faut les connaître par cœur. Commençons par la base de la base :

p

{

}

… signifie « Je veux toucher tous les paragraphes ». Après, c'est à vous de dire ce que vous faites à ces paragraphes (vous les écrivez en bleu, par exemple).

Nous avons aussi vu :

h1, em

{

}

… qui signifie « Tous les titres et tous les textes importants ». Nous avons sélectionné deux balises d'un coup.

Et enfin, nous avons vu comment sélectionner des balises précises à qui nous avons donné un nom, grâce aux attributs class  et id  :

.class

{

}

#id

{

}

Vous savez quoi ? Il existe des dizaines d'autres façons de cibler des balises en CSS ! Nous n'allons pas toutes les voir car il y en a beaucoup et certaines sont complexes, mais voici déjà de quoi vous permettre d'être plus efficace en CSS !

**Les sélecteurs avancés**

**\* : sélecteur universel**

\*

{

}

Sélectionne toutes les balises sans exception. On l'appelle le *sélecteur universel*.

**A B : une balise contenue dans une autre**

h3 em

{

}

Sélectionne toutes les balises <em>  situées à l'intérieur d'une balise <h3>  . Notez qu'il n'y a pas de virgule entre les deux noms de balises.

Exemple de code HTML correspondant :

<h3>Titre avec <em>texte important</em></h3>

**A + B : une balise qui en suit une autre**

h3 + p

{

}

Sélectionne la première balise <p>  située après un titre <h3>  .

Exemple :

<h3>Titre</h3>

<p>Paragraphe</p>

**A[attribut] : une balise qui possède un attribut**

a[title]

{

}

Sélectionne tous les liens <a>  qui possèdent un attribut title  .

Exemple :

<a href="http://site.com" title="Infobulle">

**A[attribut="Valeur"] : une balise, un attribut et une valeur exacte**

a[title="Cliquez ici"]

{

}

Idem, mais l'attribut doit en plus avoir exactement pour valeur « Cliquez ici ».

Exemple :

<a href="http://site.com" title="Cliquez ici">

**A[attribut\*="Valeur"] : une balise, un attribut et une valeur**

a[title\*="ici"]

{

}

Idem, l'attribut doit cette fois contenir dans sa valeur le mot « ici » (peu importe sa position).

Exemple :

<a href="http://site.com" title="Quelque part par ici">

**D'autres sélecteurs existent !**

Je ne vous ai présenté ici qu'une partie des sélecteurs CSS, mais sachez qu'il en existe beaucoup d'autres. Si vous voulez une liste complète, vous pouvez vous renseigner directement à la source : sur le [site du W3C](https://www.w3.org/Style/css3-selectors-updates/WD-css3-selectors-20010126.fr.html#selectors) ! C'est très complet.

Sachez que nous découvrirons certains de ces autres sélecteurs dans la suite de ce cours !

**En résumé**

* CSS est un autre langage qui vient compléter le HTML. Son rôle est de mettre en forme votre page web.
* Il faut être vigilant sur la compatibilité des navigateurs avec certaines fonctionnalités récentes de CSS3. Quand un navigateur ne connaît pas une instruction de mise en forme, il l'ignore simplement.
* On peut écrire le code CSS à plusieurs endroits différents, le plus conseillé étant de créer un fichier séparé portant l'extension .css  (exemple : style.css  ).
* En CSS, on sélectionne quelles portions de la page HTML on veut modifier, et on change leur présentation avec des propriétés CSS :

balise1

{

propriete1: valeur1;

propriete2: valeur2;

}

* Il existe de nombreuses façons de sélectionner la portion de la page que l'on veut mettre en forme. Par exemple, on peut viser :
  + toutes les balises d'un même type, en écrivant simplement leur nom ( h1  , par exemple) ;
  + certaines balises spécifiques, auxquelles on a donné des noms à l'aide des attributs class  ou id  ( .nomclasse  ou #nomid  ) ;
  + uniquement les balises qui se trouvent à l'intérieur d'autres balises ( h3 em  ).
  + etc.