

I. Mettre en forme du texte html avec le CSS3

CSS (Cascading Style Sheets : feuilles de style en cascade) est un langage informatique qui sert à décrire la présentation des documents HTML et XML.

Le CSS comporte beaucoup de propriétés, qui sont un peu l'équivalent des balises en HTML.

C'est lui qui vous permet de choisir la couleur de votre texte, c'est lui qui vous permet de sélectionner la police utilisée sur votre site, c'est lui qui permet de définir la taille du texte, les bordures, le fond...

Et aussi, c'est lui qui permet de faire la mise en page de votre site.

Il permet entre autres que mon menu soit à gauche et occupe telle largeur, que l'en-tête de mon site soit en haut et qu'il soit toujours visible, etc. ...

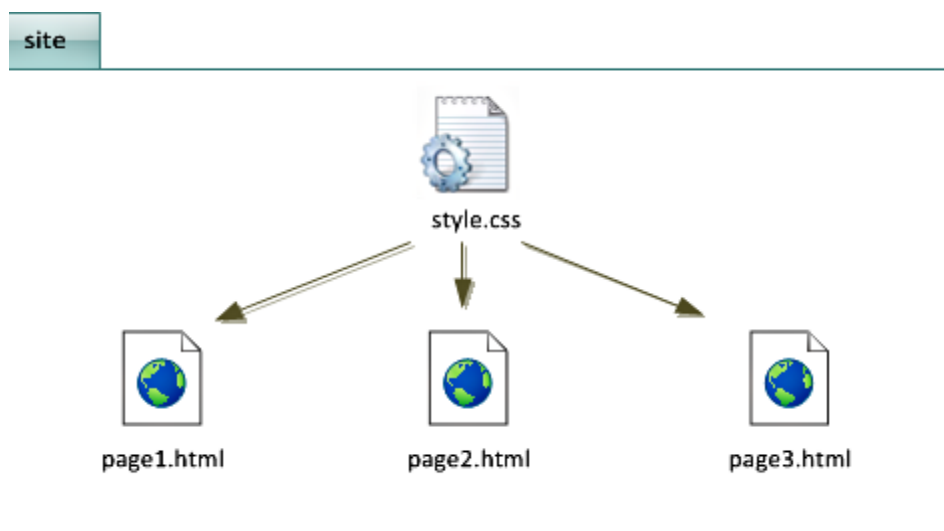
II. Emplacement du code CSS

Vous avez le choix, car on peut écrire du code en langage CSS à 3 endroits différents :

- Dans un fichier .css (**méthode la plus recommandée**) ;
- Dans l'en-tête <head> du fichier HTML ;
- Directement dans les balises du fichier HTML via un attribut style (méthode la moins recommandée) ;

Sous Notepad++, il suffit d'enregistrer le nouvel (ex : style) onglet en code css

Sous explorateur de fichiers, les 2 fichiers seront placés côte à côte. D'un côté le *.html, de l'autre le *.css



III. A quoi ressemble un fichier CSS?

Une feuille de style CSS ressemble donc à cela : (notez le point devant le nom de la balise)

```
.balise1
{
    propriete1: valeur1;
    propriete2: valeur2;
    propriete3: valeur3;
}
```

Par exemple si je veux que mes titres (nom de la balise: titre_principal) aient comme couleur le jaune je vais écrire dans le fichier CSS :

```
.titre_principal
{
    color: yellow;
}
```

Dans tous mes fichier HTML, quand je voudrait utiliser cette balise « titre_principal » je mettrais l'attribut « class »:

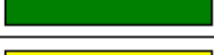
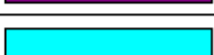
```
<h1 class="titre_principal"> Texte de mon titre </h1>
```

C'est un attribut que l'on peut mettre sur n'importe quelle balise, aussi bien titre que paragraphe, image, etc.

Par exemple :

```
<h1 class="titre_principal"> </h1>
<p class="liste_matières"></p>
<img class="Lycée" />
```

Les 16 couleurs
fondamentales :

white	
silver	
gray	
black	
red	
maroon	
lime	
green	
yellow	
olive	
blue	
navy	
fuchsia	
purple	
aqua	
teal	

IV. *Les balises universelles*

Il arrivera parfois que vous ayez besoin d'appliquer une class 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). On va voir la première de ces balises :

** ** : 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 **** et **** sont de la même famille. Cette balise s'utilise donc au milieu d'un paragraphe et c'est celle dont nous allons nous servir par exemple pour colorer « bienvenue ». :

```
<p>Bonjour et <span class= « toto » >bienvenue<span> sur mon site !</p>
```

Seize couleurs, c'est quand même un peu limite quand on sait que la plupart des écrans peuvent en afficher seize millions. D'un autre côté, remarquez, s'il avait fallu donner un nom à chacune des seize millions de couleurs...

Heureusement, il existe en CSS plusieurs façons de choisir une couleur parmi toutes celles qui existent.

V. Méthode 1 : notation hexadécimale :

Elle est couramment utilisée sur le Web. Un nom de couleur en hexadécimal, cela ressemble à : #FF5A28. Pour faire simple, c'est une combinaison de lettres et de chiffres qui indiquent une couleur. On doit toujours commencer par écrire un dièse (#), suivi de six lettres ou chiffres allant de 0 à 9 et de A à F.

Certains logiciels de dessin, comme Gimp ou Paint vous indiquent les couleurs en hexadécimal. Il vous est alors facile de copier-coller le code hexadécimal d'une couleur dans votre fichier CSS.

Cela donne par exemple :

```
. titre
{
  color : #FFFFFF;
```

VI. Méthode 2 : la méthode RGB

Que signifie RGB ? En anglais, Rouge-Vert-Bleu s'écrit Red-Green-Blue, ce qui s'abrège en « RGB ». Comme avec la notation hexadécimale, pour choisir une couleur, on doit définir une quantité de rouge, de vert et de bleu.

Cela donne par exemple :

```
. titre
{
  color : rgb(240,96,204);
}
```

VII. La taille

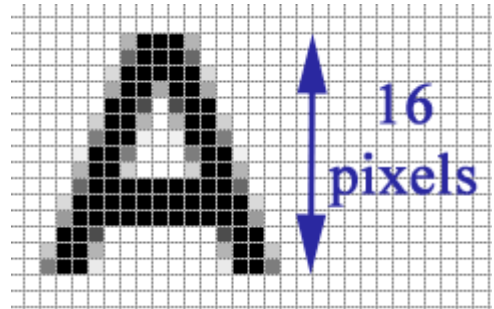
Pour modifier la taille du texte, on utilise la propriété CSS font-size. Mais comment indiquer la taille du texte ? C'est là que les choses se compliquent car plusieurs techniques existent :

- Indiquer une taille absolue : en pixels, en centimètres ou millimètres. Cette méthode est très précise mais il est conseillé de ne l'utiliser que si c'est absolument nécessaire, car on risque d'indiquer une taille trop petite pour certains lecteurs.
- Indiquer une taille relative : en pourcentage, « em » ou « ex », cette technique a l'avantage d'être plus souple. Elle s'adapte plus facilement aux préférences de taille des visiteurs.

Une taille absolue

Pour indiquer une taille absolue, on utilise généralement les pixels. Pour avoir un texte de 16 pixels de hauteur, vous devez donc écrire :

```
p
{
  font-size: 16px; /* Paragraphes de 16 pixels */
}
```



Remarque : au lieu de donner un nom de balise on a utilisé p ce qui signifie que tous les paragraphes auront cette taille de caractère

Une valeur relative

C'est la méthode recommandée car le texte s'adapte alors plus facilement aux préférences de tous les visiteurs.

Il y a plusieurs moyens d'indiquer une valeur relative. Vous pouvez par exemple écrire la taille avec des mots en anglais comme ceux-ci :

xx-small : minuscule ;
x-small : très petit ;
small : petit ;
medium : moyen ;
large : grand ;
x-large : très grand ;
xx-large : euh... gigantesque.

Ce qui donne :

```
p
{
  font-size: small;
}
```

Bon, cette technique a un défaut : il n'y a que sept tailles disponibles (car il n'y a que sept noms). Heureusement, il existe d'autres moyens. Ma technique préférée consiste à indiquer la taille en « em ».

- Si vous écrivez 1em, le texte a une taille normale.
- Si vous voulez grossir le texte, vous pouvez inscrire une valeur supérieure à 1, comme 1.3em.
- Si vous voulez réduire le texte, inscrivez une valeur inférieure à 1, comme 0.8em.

Faites attention : pour les nombres décimaux, il faut mettre un point et non une virgule. Vous devez donc écrire « 1.4em » et non pas « 1,4em » !

Exemple :

```
p
{
  font-size: 0.8em;
}
```

D'autres unités sont disponibles. Vous pouvez essayer le « ex » (qui fonctionne sur le même principe que le em mais qui est plus petit de base) et le pourcentage (80%, 130%...).

VIII. Modifier la police utilisée

La propriété CSS qui permet d'indiquer la police à utiliser est `font-family`. Vous devez écrire le nom de la police comme ceci :

```
balise
{
  font-family: police;
}
```

Mais il se pose un problème : pour qu'une police s'affiche correctement, il faut que tous les internautes l'aient. Si un internaute n'a pas la même police que vous, son navigateur prendra une police par défaut (une police standard) qui n'aura peut-être rien à voir avec ce à quoi vous vous attendiez.

Exemple de polices qui fonctionnent sur la plupart des navigateurs :

- Arial
- Arial Black
- Comic Sans MS
- Courier New
- Georgia
- Impact
- Times New Roman
- Trebuchet MS
- Verdana

Astuce : pour éviter les problèmes si l'internaute n'a pas la même police que vous, on précise en général plusieurs noms de police, séparés par des virgules :

```
balise
{
  font-family: police1, police2, police3, police4;
}
```

IX. Italique, gras, souligné...

En CSS, pour mettre en italique, on utilise `font-style` qui peut prendre trois valeurs :

- `italic` : le texte sera mis en italique.
- `oblique` : le texte sera passé en oblique (les lettres sont penchées, le résultat est légèrement différent de l'italique proprement dit).
- `normal` : le texte sera normal (par défaut). Cela vous permet d'annuler une mise en italique.

Exemple :

```
h2
{
  font-style: italic;
}
```

IX. 1. Mettre en gras :

la propriété CSS pour mettre en gras est `font-weight` et prend les valeurs suivantes :

- bold : le texte sera en gras ;
- normal : le texte sera écrit normalement (par défaut).

Voici par exemple comment écrire les titres en gras :

```
h1
{
  font-weight: bold;
}
```

IX. 2. Soulignement et autres décorations :

La propriété CSS associée est : text-decoration. Elle permet, entre autres, de souligner le texte, mais pas seulement. Voici les différentes valeurs qu'elle peut prendre :

- underline : souligné.
- line-through : barré.
- overline : ligne au-dessus.
- blink : clignotant. Ne fonctionne pas sur tous les navigateurs
- none : normal (par défaut).

Exemple :

```
.souligne
{
  text-decoration: underline;
}
```

X. Aligner un texte ou une image

Le langage CSS nous permet de faire tous les alignements connus : à gauche, centré, à droite et justifié.

On utilise la propriété text-align et on indique l'alignement désiré :

- left : le texte sera aligné à gauche (c'est le réglage par défaut).
- center : le texte sera centré.
- right : le texte sera aligné à droite.
- justify : le texte sera « justifié ». Justifier le texte permet de faire en sorte qu'il prenne toute la largeur possible sans laisser d'espace blanc à la fin des lignes. Les textes des journaux, par exemple, sont toujours justifiés.

Exemple :

```
p
{
  text-align: center;
}
```

XI. Les flottants

Le CSS nous permet de faire flotter un élément autour du texte. On dit aussi qu'on fait un « habillage ».

Exemple : (on crée une balise flottante)

```
.imageflottante
{
  float: left; /* on peut bien sur faire l'inverse avec right */
}
```

```
}  
pour stopper le texte flottant :  
.dessous  
{  
  clear: both;  
}
```

- left : le texte se poursuit en-dessous après un float: left;
- right : le texte se poursuit en-dessous après un float: right;
- both : le texte se poursuit en-dessous, que ce soit après un float: left; ou après un float: right;.

et dans le fichier :

```
<p></p>
```

```
<p>Ce texte est écrit à côté de l'image flottante.</p>
```

```
<p class="dessous">Ce texte est écrit sous l'image flottante.</p>
```

XII. Couleur de fond

Pour indiquer une couleur de fond, on utilise la propriété CSS `background-color`. Elle s'utilise de la même manière que la propriété `color`, c'est-à-dire que vous pouvez taper le nom d'une couleur, l'écrire en notation hexadécimale ou encore utiliser la méthode RGB.

Pour indiquer la couleur de fond de la page web, il faut travailler sur la balise `<body>`. Eh oui, `<body>` correspond à l'ensemble de la page web, c'est donc en modifiant sa couleur de fond que l'on changera la couleur d'arrière-plan de la page.

Exemple (fichier CSS) :

```
body
{
    background-color: black; /* Le fond de la page sera noir */
    color: white; /* Le texte de la page sera blanc */
}
```

XIII. Appliquer une image de fond

La propriété permettant d'indiquer une image de fond est `background-image`. Comme valeur, on doit renseigner `url("nom_de_l_image.png")`. Par exemple :

```
body
{
    background-image: url("neige.png");
}
```

XIV. Options disponibles pour l'image de fond :

background-attachment : fixer le fond

La propriété CSS `background-attachment` permet de « fixer » le fond. L'effet obtenu est intéressant car on voit alors le texte « glisser » par-dessus le fond. Deux valeurs sont disponibles :

- `fixed` : l'image de fond reste fixe ;
- `scroll` : l'image de fond défile avec le texte (par défaut).

Par exemple :

```
body
{
    background-image: url("neige.png");
    background-attachment: fixed; /* Le fond restera fixe */
}
```

XV. `background-repeat` : répétition du fond

Par défaut, l'image de fond est répétée en mosaïque. Vous pouvez changer cela avec la propriété `background-repeat` :

- `no-repeat` : le fond ne sera pas répété. L'image sera donc unique sur la page.
- `repeat-x` : le fond sera répété uniquement sur la première ligne, horizontalement.
- `repeat-y` : le fond sera répété uniquement sur la première colonne, verticalement.
- `repeat` : le fond sera répété en mosaïque (par défaut).

Exemple d'utilisation :

```
body
{
    background-image: url("soleil.png");
    background-repeat: no-repeat;
}
```

XVI. `background-position` : position du fond

On peut indiquer où doit se trouver l'image de fond avec `background-position`. Cette propriété n'est intéressante que si elle est combinée avec `background-repeat: no-repeat`; (un fond qui ne se répète pas).

Vous devez donner à `background-position` deux valeurs en pixels pour indiquer la position du fond par rapport au coin supérieur gauche de la page (ou du paragraphe, si vous appliquez le fond à un paragraphe). Ainsi, si vous tapez : `background-position: 30px 50px`;

... votre fond sera placé à 30 pixels de la gauche et à 50 pixels du haut. Il est aussi possible d'utiliser ces valeurs en anglais :

- top : en haut ;
- bottom : en bas ;
- left : à gauche ;
- center : centré ;
- right : à droite.

Il est possible de combiner ces mots. Par exemple, pour aligner une image en haut à droite, vous taperez :

background-position: top right;

Par exemple : si je veux afficher un soleil en image de fond, en un unique exemplaire (no-repeat), toujours visible (fixed) et positionné en haut à droite (top right), je vais écrire ceci :

```
body
{
  background-image: url("soleil.png");
  background-attachment: fixed; /* Le fond restera fixe */
  background-repeat: no-repeat; /* Le fond ne sera pas répété */
  background-position: top right; /* Le fond sera placé en haut à droite */
}
```

XVII. Plusieurs images en fond

il est possible de donner plusieurs images de fond à un élément. Pour cela, il suffit de séparer les déclarations par une virgule, comme ceci :

```
body
{
    background: url("soleil.png") fixed no-repeat top right, url("neige.png") fixed;
}
```

La première image de cette liste sera placée par-dessus les autres. Attention donc, l'ordre de déclaration des images a son importance : si vous inversez le soleil et la neige dans le code CSS précédent, vous ne verrez plus le soleil !

XVIII. Réactualisation de la page web :

Ceci est utile pour lire une entrée/sortie qui évolue dans le temps.

Il suffit d'inclure la commande entre <head> et </head>

Exemple : `<meta http-equiv= "refresh" content="5"/>`

Le navigateur rafraîchira la page toutes les 5 secondes.

XIX. La propriété opacity

La propriété opacity, très simple, permet d'indiquer le niveau d'opacité (c'est l'inverse de la transparence).

Avec une valeur de 1, l'élément sera totalement opaque : c'est le comportement par défaut.

Avec une valeur de 0, l'élément sera totalement transparent.

Il faut donc choisir une valeur comprise entre 0 et 1. Ainsi, avec une valeur de 0.6, votre élément sera opaque à 60%... et on verra donc à travers !

Par exemple : `opacity: 0.6;`

Cette propriété fonctionne pour les images, le texte,