

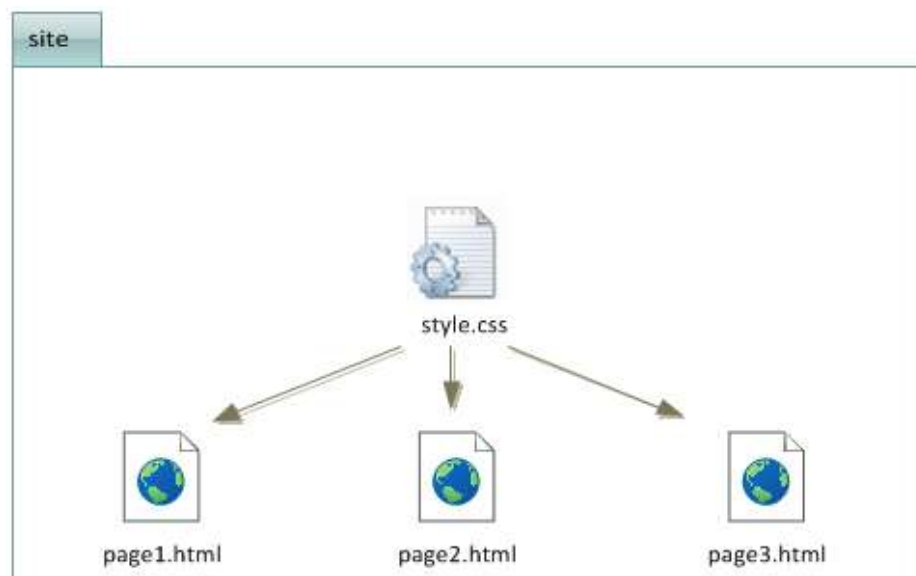
CHAPITRE 2 : les feuilles de styles CSS

Le CSS (*Cascading Style Sheets*), associé au HTML, permet de gérer de manière centralisée la mise en forme de votre site. Il à été recommandé par le W3C pour la première version en 1996. La version 3.0 ou CSS3 est celle en cours. Elles ne sont vraiment correctement supportées par les navigateurs que depuis 2000. C'est donc depuis cette date qu'elles sont devenues très présentes sur le Web. Elles peuvent être interne à la page HTML ou externe à celle-ci (dans ce cas elles seront regroupées dans un fichier ayant comme extension « **.CSS** »).

I/Notions de base :

a) Utilisation

Les CSS permettent la séparation de la forme (couleur, police, taille, positionnement...) des informations contenues dans la page HTML. **On peut ainsi avoir une même feuille de style pour l'ensemble des pages d'un site ce qui permet d'avoir une charte graphique homogène.** Par exemple, lorsque vous effectuerez une modification dans votre feuille de style, elle sera automatiquement répercutée sur l'ensemble de votre site sans avoir à reprendre toutes les pages. Ainsi, vous vous retrouvez avec un code source allégé et une plus grande lisibilité du HTML.



b) Où écrit-on le CSS ?

Syntaxe

Un fichier CSS est composé de plusieurs règles. Chaque règle permet de changer l'affichage de plusieurs balises HTML. Les règles ont la syntaxe suivante :

<pre>Sélecteur { propriété-1: valeur-1; propriété-2: valeur-2; propriété-n: valeur-n; }</pre>	<pre>h1 { text-align: center; color: blue; } a { color: #CCC; }</pre>
---	--

Insertion d'une feuille de style **interne**

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <style>
      p
      {
        color: blue;
      }
    </style>
    <title>Premiers tests du CSS</title>
  </head>
  <body>
    <h1>Mon super site</h1>

    <p>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>
```

Ou bien, sans partie CSS:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Premiers tests du CSS</title>
  </head>
  <body>
    <h1>Mon super site</h1>

    <p style="color: blue;">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>
```

Insertion d'une feuille de style **externe**

```
<!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>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>
```

Je vous recommande fortement de prendre l'habitude de travailler avec la deuxième méthode (**Insertion d'une feuille de style externe**).

Sinon, imaginer que vous placez le code CSS directement dans les fichiers HTML, il faudra copier ce code dans tous les fichiers HTML de votre site ! Et si demain vous changez d'avis, par exemple si vous voulez que vos paragraphes soient écrits en rouge et non en bleu, il faudra modifier chaque fichier HTML un à un !

c) **Interprétation du code**

Le code CSS est interprété directement par le navigateur et gère la mise en forme du HTML à l'aide des balises de celui-ci. En CSS, le plus difficile est de savoir cibler le texte dont on veut changer la forme. Pour cibler (sélectionner) les éléments de la page à modifier, on utilise ce qu'on appelle **des sélecteurs**. Schématiquement, une feuille de style CSS ressemble donc à ça :

Format de Code CSS :

```
balise1
{
  propriete: valeur;
  propriete: valeur;
  propriete: valeur;
}

balise2
{
  propriete: valeur;
  propriete: valeur;
  propriete: valeur;
  propriete: valeur;
}

balise3
{
  propriete: valeur;
}
```

Exemple

```
h1
{
  color: blue;
}

em
{
  color: blue;
}
```

ou bien

```
h1, em
{
  color: blue;
}
```

Il signifie que nos titres <h1> et nos textes importants doivent s'afficher en bleu.

Remarque : Pour faire un commentaire, Tapez `/*`, suivi de votre commentaire, puis `*/` pour terminer votre commentaire. Vos commentaires peuvent être écrits sur une ou plusieurs lignes.

d) Appliquer un style : class et id

Ce que nous venons de voir suppose que **TOUS** les paragraphes soient écrits par exemple en bleu.

Ici nous verrons comment faire pour que seulement certains paragraphes soient écrits d'une manière différente. On pourrait placer notre code CSS dans un attribut style sur la balise que l'on vise, 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**

Les attributs **class** et **id** sont quasiment identiques à un détail près : l'attribut **id** ne peut être utilisé *qu'une fois* dans le code. Donc cela vous sera utile si vous faites du *Javascript* plus tard pour reconnaître certaines balises. En pratique, nous ne mettrons des **id** que sur des éléments qui sont uniques dans la page, comme par exemple le logo.

```
<!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>
```

Code CSS :

```
.introduction
{
  color: blue;
}
```



```

```

Code CSS :

```
#logo
{
    /* Indiquez les propriétés CSS ici */
}
```

II/La taille

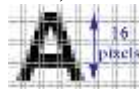
Pour modifier la taille du texte, on utilise la propriété CSS **font-size**. Plusieurs techniques sont proposées :

- 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 parfois 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.

a) Une taille absolue

Pour indiquer une taille absolue, on utilise généralement les pixels. Pour avoir un texte de 16 pixels de hauteur, vous devez écrire le **Code CSS : font-size: 16px;**

Les lettres auront une taille de 16 pixels, comme le montre l'image suivante :



Voici un exemple d'utilisation placé dans un fichier .css:

Code : CSS

```
P
{
    font-size: 14px; /* Paragraphes de 14 pixels */
}
h1
{
    font-size: 40px; /* Titres de 40 pixels */
}
```

Si vous le souhaitez, des tailles en centimètres ou millimètres sont aussi disponibles. Remplacez "px" par "cm" ou "mm". Ces unités sont cependant moins bien adaptées aux écrans.

b) 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** : gigantesque.

Exemples:

Code : CSS

```
P
{
  font-size: small;
}
h1
{
  font-size: large;
}
```

Cette technique a un défaut : il n'y a que 7 tailles disponibles (car il n'y a que 7 noms). Heureusement il existe d'autres moyens. La taille en "**em**" est une unité spécifique au CSS.

- 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, il faut mettre un point à la place de la virgule pour les nombres décimaux. Vous devez donc écrire "1.4em" et non pas "1,4em" !

Exemple :

Code : CSS

```
P
{
  font-size: 0.8em;
}
h1
{
  font-size: 1.3em;
}
```

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%...).

III/Type de police

a) 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 :

Code : CSS

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

Seulement, pour éviter qu'il n'y ait de problème si l'internaute n'a pas la même police que vous, on précise en général *plusieurs* noms de police, séparés par des virgules :

Code : CSS

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

Exemple :

```
P  
{  
  font-family: Impact, "Arial Black", Arial, Verdana, sans-serif;  
}
```

Le navigateur essaiera d'abord de mettre la police1. S'il ne l'a pas, il essaiera de mettre la police2. S'il ne l'a pas, il essaiera la police3 et ainsi de suite.

En général, on indique en tout dernier **serif**, ce qui correspond à une police standard (qui ne se met que si aucune autre police n'a été trouvée).

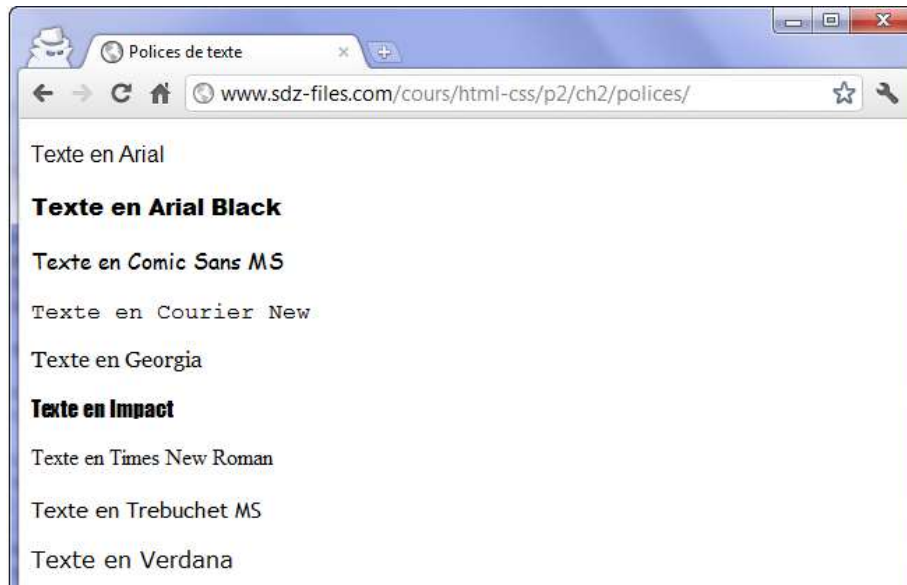
Il existe aussi une autre police par défaut appelée sans-serif. Il y a une petite différence entre ces deux polices par défaut : serif possède des pattes de liaison en bas des lettres, tandis que sans-serif n'en possède pas.

Voici une liste de polices qui fonctionnent bien sur la plupart des navigateurs :

- Arial
- Arial Black
- Comic Sans MS
- Courier New

- Georgia
- Impact
- Times New Roman
- Trebuchet MS
- Verdana

Voici à quoi ressemblent ces polices :



Si le nom de la police comporte des espaces, je conseille de l'entourer de guillemets, comme je l'ai fait pour "Arial Black". Avec CSS 3, il est possible de définir votre propre police. Voir ^{1,2}

b) Italique, gras, souligné...

Il existe en CSS une série de propriétés de mises en forme classiques du texte. Nous allons découvrir ici la mise en gras, italique, souligné... et au passage nous verrons qu'il est même possible d'aller jusqu'à faire clignoter le texte.

Mettre en italique

La balise ****, est fait pour insister sur des mots. Ca veut dire que les mots qu'il entoure sont assez importants. Pour représenter cette importance, la plupart des navigateurs choisissent d'afficher le texte en **italique** (mais ce n'est pas une obligation).

Le CSS lui, permet de dire réellement : "Je veux que ce texte soit en italique". Rien ne vous empêche par exemple de décider que tous vos titres seront en italique.

¹ <https://fr.flossmanuals.net/css3/changer-la-police/>

² <https://openclassrooms.com/fr/courses/1603881-apprenez-a-creez-votre-site-web-avec-html5-et-css3/1605329-formatez-du-texte>

Concrètement, pour mettre en italique en CSS on utilise **font-style**, qui peut prendre 3 valeurs :

- italic : le texte sera mis en italique.
- oblique : le texte sera aussi mis en italique (en penchant les lettres).
- normal : le texte sera normal (par défaut). Cela vous permet d'annuler une mise en italique. Par exemple, si vous voulez que les textes entre **** ne soient plus en italique, vous devrez écrire :

Code : CSS

```
1 Em
2 {
3   font-style: normal;
4 }
```

Sur l'exemple suivant, je me sers par exemple de **font-style** pour mettre en italique tous mes titres **<h2>** :

Code : CSS

```
1 h2
2 {
3   font-style: italic;
4 }
```

c) Mettre en gras

Pareil, n'oubliez pas que ce n'est pas **** qui permet de mettre en gras (son rôle est d'indiquer que le texte est important, *donc* le navigateur l'affiche en gras). La mise en gras en CSS permet de mettre en gras par exemple les titres, certains paragraphes entiers, etc. C'est à vous de voir.

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 :

Code : CSS

```
1 h1
2 {
3   font-weight: bold;
4 }
```

d) Soulignement et autres décorations

Cette propriété CSS porte bien son nom : **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 marche pas sur tous les navigateurs.
- none : normal (par défaut).

Ce CSS va vous permettre de tester les effets de text-decoration :

Code : CSS

```
h1
{
  text-decoration: blink;
}
.souligne
{
  text-decoration: underline;
}
.barre
{
  text-decoration: line-through;
}
.ligne_dessus
{
  text-decoration: overline;
}
```

e) L'alignement

Le langage CSS nous permet de faire tous les alignements que l'on connaît : à gauche, centré, à droite et justifié.

C'est tout simple. 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 tout le temps justifiés.

Regardez les différents alignements sur cet exemple :

Code : CSS

```
h1
{
    text-align: center;
}

p
{
    text-align: justify;
}

.signature
{
    text-align: right;
}
```

Suite à ces bases, d'autres exemples seront abordés en TP en classe.