

## **1) Introduction à CSS**

- A) Qu'est-ce que qu'une feuille de Style CSS ?
- B) Comment insérer une feuille CSS ?
- C) Classes et style, une feuille de base

## **2) Le fichier de style**

- A) Les unités
- B) Les polices de caractères
- C) Les marges
- D) Les évènements
- E) Le positionnement absolu

## Introduction à CSS

### 1.A) Qu'est ce qu'une feuille de style

CSS : Cascading Steel Sheets (feuilles de style en cascade), est une technologie permettant la mise en page d'une page HTML ou XML.

Une feuille de style CSS se résumé à un fichier au format texte, chargé de décrire la façon dont les éléments d'une page Web seront affichés. On peut personnaliser les polices de caractères, leur taille, la couleur d'arrière plan, etc.

L'avantage de CSS, par rapport à une description logée dans une page HTML, est de centraliser toutes ces informations, et de pouvoir les réutiliser pour décrire tout un lot de pages Web. Ainsi, l'internaute n'aura à charger la feuille CSS qu'une seule fois, et l'appliquer aux pages Web d'un même site. De cette manière on allège le poids des pages HTML, on réduit la bande passante utilisée, et la maintenance de la charte graphique du site Web s'en trouve grandement facilitée.

En effet, pour changer l'aspect d'un site tout entier, il suffit de modifier la ou les quelques feuilles CSS. Sans CSS, il faudrait éditer chacune des pages Web. Imaginez donc une plateforme dotée de quelques milliers de pages ...

### 1.B) Comment insérer une feuille CSS ?

Une page HTML se composant de deux parties : un en-tête (head) et un corps (body), on préférera faire référence à la feuille CSS dans l'en-tête.

Une seule ligne suffit à charger une feuille de style CSS :

```
<link rel="stylesheet" type="text/css" href="/blog/themes/old/style.css"
media="screen" />
```

L'insertion d'une feuille CSS s'effectue grâce à la balise **link**.

Le **type** nous dit qu'il sagit d'une feuille CSS au format texte (c'est le type MIME).

L'attribut **href** donne l'adresse de la feuille de style CSS. On peut utiliser une adresse absolue ou relative.

Enfin, **media** nous indique que cette feuille CSS est adaptée aux écrans d'ordinateurs. On peut aussi spécifier des feuilles pour les télévisions, les téléphones portables, etc.

## 1.C) Classes et style, une feuille de base

Voici à quoi peut ressembler une page CSS de base (un fichier d'extension « css ») :

```
body {  
    color: blue;  
    background-color: white;  
}  
p {  
    color: red;  
}
```

Dans cette feuille de style CSS, on distingue deux « classes » : **body** et **p**. Ces deux classes vont donc décrire de quelle façon interpréter les balises **<body>** et **<p>**. C'est ainsi que l'on va pouvoir référencer et décrire toutes les balises présentes dans un lot de pages HTML. Notez qu'il est aussi possible de décrire un ensemble de balises, ce en les séparant par des virgules : **p, h1, h2, em, strong**.

Les accolades servent à annoncer le début et la fin de la description d'une classe.

Entre les accolades, on distingue plusieurs styles :

- **color** : ce style sert à spécifier la couleur d'une zone de texte.  
Le style « **color: blue;** » décrit une zone de texte de couleur bleue. La couleur peut être une couleur prédéfinie (blue, red, etc) ou bien un code hexadécimal (#FFF, etc).
- **background-color** : ce style spécifie une couleur d'arrière plan.  
Le style « **background-color: white;** » décrit un arrière plan rouge dans la balise décrite.

Vous remarquerez que « body » est le corps de la page HTML, tandis que « p » n'est qu'un élément contenu dans la page HTML, et plus précisément dans le corps (« body »). CSS permet la superposition de styles : on parle de superposition de calques. Notez que c'est le dernier style décrit qui prend la priorité sur le précédent. Ainsi, dans cet exemple, la couleur du texte dans les balises « <p> » sera rouge, le reste étant bleu.

Enfin, lorsque vous voulez appliquez plusieurs styles aux mêmes balises, il est possible de nommer ces styles, en faisant précéder un nom à la balise :

```
p.DD {  
    color: purple;  
}
```

Cette classe correspondrait au code HTML suivant :

```
<p class="DD">
```

on peut aussi personnaliser des ancres :

```
#ancre1 {  
    color: white;  
}
```

Cette classe correspondrait au code HTML suivant :

```
<p id="ancre1">
```

## Le fichier de style

### 2.A) Les unités

Il existe plusieurs unités de mesure :

- px : le pixel,
- em une unité extensible selon la configuration de l'écran,
- cm : le centimètre,
- mm : le millimètre,
- % : une portion en pourcentage d'une dimension de la page.
- Etc.

px, cm et mm sont des unités dites fixes. Quel que soit la résolution de la page, leur taille ne changera pas.

em et % sont des unités adaptables. Elles varient dynamiquement suivant la résolution de la page.

Pour ces raisons de lisibilité et d'accessibilité, on préférera les unités adaptables de par leur compatibilité avec les petites résolutions, mais aussi avec les plus grandes.

### 2.B) Les polices de caractères

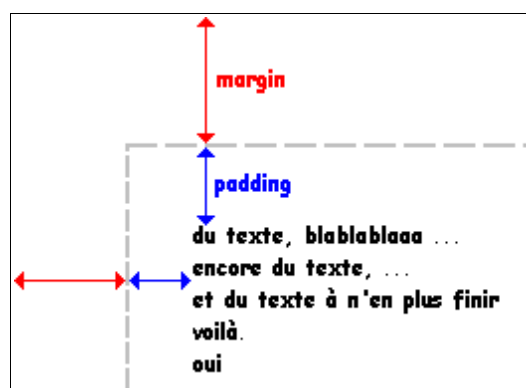
Voici les principaux styles applicables sur les caractères :

- font-family : la police d'impression : Bitsream Vera Sans, Arial, serif, etc.
- font-size : la taille des caractères, avec les unités vues précédemment (en 2.A).
- font-style : le style des caractères : normal (« normal »), en italique (« italic »), ou en oblique (« oblique »).
- font-weight : le poids des caractères : normal (« normal »), en gras (« bold »), plus gras (« bolder », on augmente de 1 le poids), moins gras (« lighter », on diminue de 1 le poids), ou une valeur : 100, 200, etc.
- letter-spacing : l'espace horizontal entre les caractères, avec les unités vues précédemment.
- line-height : la hauteur des lignes, avec les unités vues précédemment.

### 2.C) Les marges

Avec CSS, on va manipuler deux sortes de marges : « margin » et « padding ». On les utilise avec les unités de mesure vues précédemment.

Voici à quoi elles peuvent ressembler :



« margin » est la marge dite extérieure, tandis que « padding » est la marge intérieure. Ici, la bordure discontinue simule le cadre de la zone dans laquelle se trouve le texte.

Notez que les marges peuvent être générales : elles ont la même valeur pour les écart de gauche, de droite, en haut et en bas. Elles peuvent aussi être découpées en 4 sous-marges : gauche, droite, haut et bas.

Deux exemples avec « margin » :

```
p{
    margin: 10px;
}
```

Ici, le margin est général et de 10 pixels.

```
p{
    margin-left: 10px;
    margin-right: 10px;
    margin-top: 10px;
    margin-bottom: 10px;
}
```

Ici, on spécifie manuellement chaque sous-marge de margin, de 10 pixels. Notez que les deux exemples donneront exactement le même résultat.

On pourrait donner les même exemple avec padding.

**Attention** : veillez à toujours donner au moins une spécification des marges dans la classe « body ». La majorité des navigateurs Web donnent une valeur par défaut à ces marges si elles ne sont pas spécifiées. Sans cette manipulation, votre site Web risquerait fortement de provoquer des affichages différents suivants les navigateurs.

## 2.D) Les événements

Nous avons vu comment décrire les éléments de page Web de façon statique : une fois la page chargée le style ne varie pas tant que l'on n'a pas changé de feuille CSS ou de page Web. C'est là que les événements entrent en jeu.

On distingue plusieurs événements :

- hover : lorsque l'on passe le curseur sur la zone en question.
- visited : lorsque le lien hypertexte a déjà été cliqué au moins une fois.
- link : si la zone est un lien hypertexte.

Un événement est précédé de la classe à laquelle il s'applique.

De cette manière, on peut personnaliser les liens, et tout autre élément d'une page Web. Voici un exemple de personnalisation de la classe « type1 » et des liens hypertextes :

```
type1:hover {
    color:#000;
}
a:link {
    color:#484;
}
a:visited {
    color:#484;
}
a:hover {
    color:#000;
}
```

## 2.E) Le positionnement absolu

Une manière couramment utilisée sur le Web pour découper une page consiste à disposer des calques via des positions absolues.

On utilisera l'attribut `position: absolute`.

De cette manière, il sera possible de spécifier les coordonnées du coin supérieur gauche d'un calque, l'origine de la page étant le coin supérieur gauche de la zone d'affichage du navigateur Web.

```
p.type1 {
    position: absolute;
    left: 10px;
    top: 9px;
}
p.type2 {
    position: absolute;
    right: 100px;
}
```

Dans cet exemple, la classe `p.type1` sera positionnée à 10 pixels du bord gauche du navigateur, et à 9 pixels du haut.

La classe `p.type2` sera positionnée à 100 pixels du bord droit du navigateur, la coordonnée verticale n'étant pas spécifiée.

Notez qu'avec de tels attributs, si deux calques se trouvent l'un sur l'autre, c'est le dernier utilisé dans le code HTML qui apparaîtra sur l'autre. On peut ainsi superposer différents calques sans aucune surprise.

### *Spécifications CSS versions 1&2 du W3C*

Spécifications du W3C sur CSS1 :

- version EN : <http://www.w3.org/TR/REC-CSS1>
- traduction FR : <http://www.yoyodesign.org/doc/w3c/css1/index.html>

Spécifications du W3C sur CSS2 :

- version EN : <http://www.w3.org/TR/REC-CSS2/>
- traduction FR : <http://www.yoyodesign.org/doc/w3c/css2/cover.html>