

Réseaux 1

Programmation Internet

Damien Sauveron

damien.sauveron@xlim.fr

<http://damien.sauveron.fr/>

CSS

Les feuilles de style

Feuilles de style : CSS

- Cascading Style Sheets
 - Servent à contrôler la mise en page des différents éléments qui composent un document HTML.
 - Permettent de créer un affichage particulier pour une ou plusieurs balises.
 - Cascading => une balise incluse à l'intérieur d'une autre, hérite soit par défaut du style de la balise parente, soit a la possibilité de modifier ce style
- Leur objectif principal : permettre une séparation entre le contenu et la forme d'un document web.

CSS

- Permet de décrire la présentation d'un document XML
 - séparation contenu / présentation
 - utilisé par les concepteurs et les lecteurs
- 3 versions: CSS1, CSS2.1, CSS3 (en préparation)

Notre exemple

- Supposons que nous désirions qu'un titre de taille 1 ait une couleur bleue.
- Il y a trois façons d'obtenir cet effet avec le CSS.

Solution 1 : Utilisation de l'attribut de style

On spécifie ici directement le style qui ne sera valable que pour l'élément h1 courant

```
<h1 style="color: blue">
```

Introduction.

```
</h1>
```

Résultat obtenu :

Introduction.

Un style est constitué de différentes propriétés (ici, la propriété « color ») auxquelles on affecte des valeurs (ici, le nom de couleur « blue »).

Propriétés

Les noms des propriétés et les valeurs ne sont pas sensibles à la casse.
On sépare les différentes propriétés d'un style par un point-virgule.

Exemple :

```
<p style="color: red; font-family: Arial; font-style: italic">  
    Un essai.<br />  
    Une tentative.<br />  
</p>
```

Affichage obtenu :



Un essai.
Une tentative.

Solution 2 : Feuille de style intégrée à la page HTML

On va définir un style par défaut pour tous les éléments h1 du document. On pourra toutefois utiliser un autre style en recourant à l'attribut style pour un élément h1 particulier (d'où le qualificatif de feuilles de style **en cascade**).

Pour insérer une feuille de style dans une page HTML, on ajoute à l'intérieur de l'élément head la structure suivante :

```
<style type="text/css">  
    <!-- définition du style -->  
</style>
```

Solution 2

```
<!DOCTYPE ...>
<html>
  <head>
    <title>Un titre significatif</title>
    <style type="text/css">
      h1 { color: blue }
    </style>
  </head>
  <body>
    <h1>Introduction</h1>
    <p>Bla bla bla.</p>
    <h1 style="color: red">Conclusion</h1>
    <p>Bla bla.</p>
  </body>
</html>
```

Affichage obtenu :

Introduction

Bla bla bla.

Conclusion

Bla bla.

Remarques

- « h1 { color : blue } » est appelée une règle. « h1 » est le sélecteur de la règle, « color : blue » la déclaration de la règle.
- Il est possible de regrouper des balises pour leur donner un même style. On les sépare alors par des virgules. Exemple :
h1, h2, h3 { font-style : italic }

Solution 3 : Les feuilles de style externes

Il s'agit de la solution à privilégier puisqu'elle permet :

- de ne pas mélanger du contenu HTML avec la définition d'un style,
- de pouvoir réutiliser simplement un style pour différentes pages.

Pour employer cette méthode, on ajoute une balise `<link>` dans l'élément head de la page HTML.

Exemple :

```
<link rel="stylesheet" type="text/css" href="monStyle.css">
```

L'attribut rel indique le type de lien (ici : une feuille de style).

L'attribut type sert à spécifier le type (MIME) de la feuille de style.

L'attribut href a pour valeur une URI référant la feuille de style.

Remarque : Le fichier monStyle.css ne doit pas contenir de code HTML.

Solution 3

```
<!DOCTYPE ...>
<html>
  <head>
    <title>Un titre significatif</title>
    <link rel="stylesheet" type="text/css" href="monStyle.css">
  </head>
  <body>
    <h1>Introduction</h1>
    <p>Bla bla bla.</p>
    <h1>Conclusion</h1>
    <p>Bla bla.</p>
  </body>
</html>
```

Résultat obtenu :

Introduction

Bla bla bla.

Conclusion

Bla bla.

On a le fichier monStyle.css suivant :

```
/* Commentaire */
h1 { color: blue }
```

Remarques

- Il est possible d'avoir plusieurs balises `<link>` pour un même document HTML. Ceci permet de combiner des définitions de style.
- Les commentaires CSS ont la même forme que les commentaires du C.

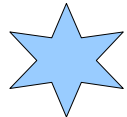
Classes

Nous avons vu précédemment que l'on peut associer un style (constitué d'une ou plusieurs propriétés) à un élément.

Exemple :

```
p { font-size : 12pt; color : white }
```

Ce style s'appliquera alors par défaut à tous les paragraphes du document.



Pour distinguer différents types de paragraphes et leur assigner différents styles, on dispose de la notion de classe. On commence par définir dans la feuille de style différentes classes pour une balise. On positionne ensuite l'attribut class dans une balise du document HTML pour lui associer le style de la classe indiquée. Les balises n'ayant pas d'attribut class auront le style par défaut déterminé par le navigateur.

Illustration classe 1/2

```
<!DOCTYPE ...>
<html>
  <head>
    <title>Un titre significatif</title>
    <style type="text/css">
      p.important { font-size: 1cm; font-weight: bold;
                    color: red }
      p.futile { font-size: 5mm; color: blue }
    </style>
  </head>
  <body>
    <p class="important">Lisez Login:</p>
    <p class="futile">Bla bla.</p>
    <p>Normal</p>
  </body>
</html>
```

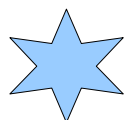
Résultat obtenu :

Lisez Login:

Bla bla.

Normal

Illustration classe 2/2



Si l'on souhaite affecter un style à tous les éléments d'une classe, on dispose de la notation :

```
.maClasse { color : green }
```

Résultat obtenu :

```
<!DOCTYPE ...>
<html>
  <head>
    <title>Un titre expressif</title>
    <style type="text/css">
      .important { font-size: 0.6cm; font-weight: bold;
                  color: red }
    </style>
  </head>
  <body>
    <h1 class="important">Un message important :</h1>
    <p class="important"><em>Lisez Linux Magazine</em></p>
  </body>
</html>
```

Un message important :
Lisez Linux Magazine

Identificateur



Pour limiter un effet de style à une seule balise de la page HTML, on a recours à un identifiant.

```
<!DOCTYPE ...>
```

```
<html>
```

```
  <head>
```

```
    <title>Un titre significatif</title>
```

```
    <style type="text/css">
```

```
      #beau { text-align: right; color: green }
```

```
    </style>
```

```
  </head>
```

```
  <body>
```

```
    <p id="beau">Lisez Linux Magazine</p>
```

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

```
    <!--
```

```
      <p id="beau">Bla bla</p>  → Interdit puisque un
```

identifiant doit être unique.

```
      <h2 id="beau">1</p>      → Interdit pour la même
```

raison.

```
    -->
```

Affichage obtenu :

```
  </body>
```

```
</html>
```

Lisez Linux Magazine

Normal

Remarque : L'interdiction qui est mentionnée en commentaire est souvent ignorée par les navigateurs. Elle ne l'est pas par le validateur du W3C.

L'élément Div

La balise `<div>` sert à faire un regroupement dans un document HTML.

```
<!DOCTYPE ...>
<html>
  <head>
    <title>Th&egrave;se</title>
    <style type="text/css">
      div.resume { text-align : justify;
                  color: red }
    </style>
  </head>
  <body>
    <div class="resume">
      <h1>R&eacute;sum&eacute;</h1>
      <p>
        Ces travaux ont pour objectif essentiel de faire
        ceci, cela, et encore cela, choses fondamentales
        pour remplir ce magnifique rapport que vous lisez
        en moment.
      </p>
      <p>
        Bla bla bla.
      </p>
      <em>Mots-cl&eacute;s</em> : Bla.
    </div>
  </body>
</html>
```

Affichage obtenu :

Résumé

Ces travaux ont pour objectif essentiel de faire ceci, cela, et encore cela, choses fondamentales pour remplir ce magnifique rapport que vous lisez en moment.

Bla bla bla.

Mots-clés : Bla.

L'élément Div (suite)

Les blocs constitués par les éléments div peuvent être positionnés au pixel près.

```
<div id="bloc1" style="position: absolute; left: 10; top: 10;  
color : green; font-size: 0.7cm ">
```

```
<p>
```

Contenu du bloc 1

```
</p>
```

```
</div>
```

```
<div id="bloc2" style="position: absolute; left: 8; top: 9;  
color : red; font-size: 0.5cm ">
```

```
<p>
```

*** Le bloc 2 ***

```
</p>
```

```
</div>
```

Résultat obtenu :



*** Le bloc 2 ***
Contenu du bloc 1

L'élément span

L'élément span est similaire à l'élément div. On utilise habituellement div pour des blocs de texte entiers et span pour des petites portions de texte (mots ou lettres).

L'attribut media

Il s'agit d'un attribut de la balise link. Avec cet attribut, on demande au navigateur de charger une feuille de style différente en fonction du medium de sortie. On peut ainsi écrire :

```
<link rel="stylesheet" type="text/css" href="normal.css" media="screen">  
<link rel="stylesheet" type="text/css" href="impression.css" media="print">
```

Cette utilisation des feuilles de style est à privilégier sur des solutions où l'on met à la disposition de l'internaute une page "normale" pour l'affichage et une page "spéciale" pour l'impression.

D'autres types de media ont une valeur standard : projection, handheld, etc.

Quelques propriétés et valeurs

Propriété	Valeurs possibles	Description
font-size	small, medium, large. Taille définie en pixels (px), en cm (cm), en mm (mm), en pourcentage (%)	Taille de la police
font-family	serif, sans-serif, cursive, ... Nom de la police (Arial, Verdana, etc.)	Type de police
font-weight	normal, bold, bolder, lighter	Poids de la police
font-style	normal, italic, oblique	Style de la police

Propriété	Valeurs possibles	Description
color	Nom ou valeur hexadécimale de la couleur.	Couleur de la police
background-image	URL du fichier. Exemple : url(marble.jpg)	Image d'arrière plan d'un élément
background-repeat	repeat, no-repeat, repeat-x, repeat-y	Répétition de l'image d'arrière-plan
background-color	Nom ou valeur hexadécimale de la couleur.	Couleur d'arrière-plan d'un élément

Quelques propriétés et valeurs

Propriété	Valeurs possibles	Description
text-align	left, right, center, justify	Alignement du texte
text-indent	Valeur en pixels ou en pourcentage	Retrait de la première ligne
text-decoration	none, underline, overline	Décoration du texte
text-transform	none, capitalize, uppercase, lowercase	Casse du texte

Unités et Couleurs

- Il existe plusieurs unités pour les valeurs:
 - `em` (taille de la fonte), `pt` (point), `px` (pixel), `%` (pourcentage)
- De même, il existe plusieurs manières de décrire une couleur:
 - `red`
 - `rgb(255, 0, 0)`
 - `rgb(100%, 0%, 0%)`
 - `#ff0000`
 - `#f00`

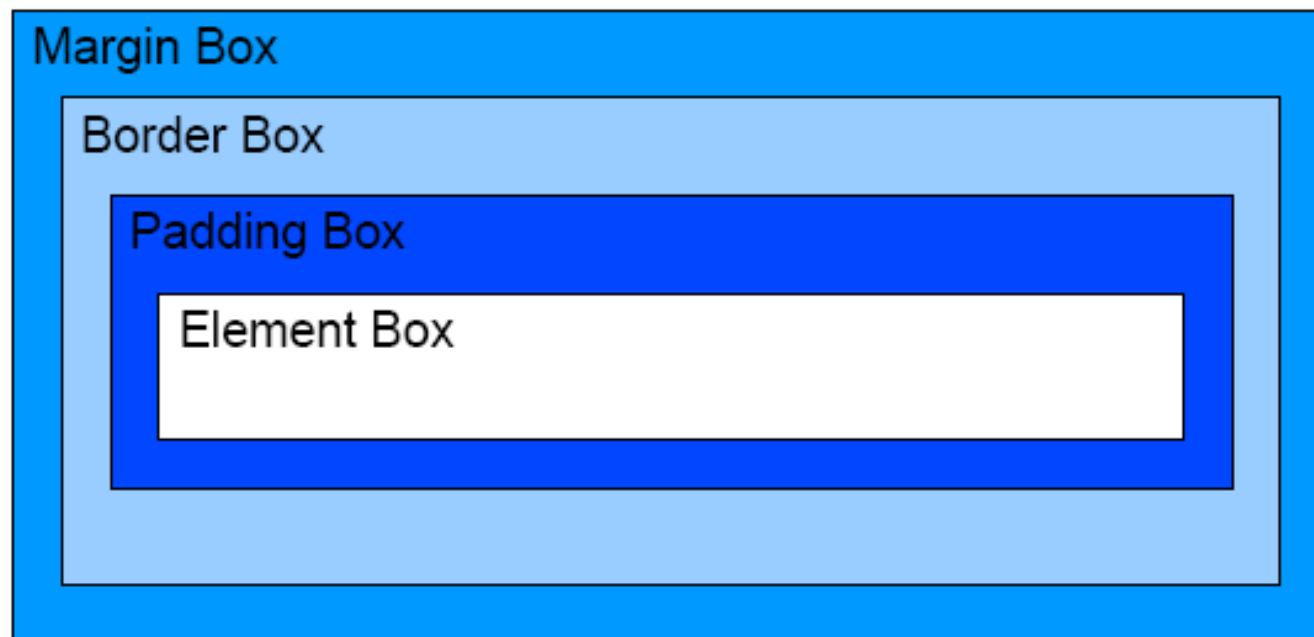
CSS: Texte et Polices

- **Fontes**: font-family, font-size, font-weight, font-style, font-variant
- **Mise en page**: line-height, letter-spacing, word-spacing, text-align, text-decoration, text-indent, text-transform, vertical-align, white-space

```
body {  
    font-family: arial;  
    font-size: 0.8em;  
}  
  
h1 { font-size: 2em; }  
  
h2 { font-size: 1.5em; }  
  
a { text-decoration: none; }  
  
strong {  
    font-style: italic;  
    text-transform: uppercase;  
}  
  
p {  
    letter-spacing: 0.5em;  
    word-spacing: 2em;  
    line-height: 1.5;  
    text-align: center;  
}
```

CSS: Marge et Boîte englobante

- Un élément bloc possède:
 - une marge (`margin-*`)
 - une boîte englobante (`border-*`)
 - La boîte est configurable graphiquement (couleur, motifs des lignes...)
 - un alignement dans la boîte englobante (`padding-*`)



Les liens hypertextes

On dispose de facilités pour attribuer un style aux différents liens hypertextes d'une page HTML.

Exemple :

a:link { color: red }	/* Style pour des liens non visités */
a:visited { color: blue }	/* Style pour des liens déjà visités */
a:hover {color: green }	/* Style lorsque le pointeur de la souris survole le lien */

```
<!DOCTYPE ...>
<html>
  <head>
    <title>Titre</title>
    <style type="text/css">
      a:link { color: red}
      a:visited { color : blue }
      a:hover {color : green }
    </style>
  </head>
  <body>
    <a href="asuivre.html">
      A suivre...
    </a>
  </body>
</html>
```

« link », « visited », etc. sont appelées des pseudo-classes.

CSS: classe et ID (1)

- Pour l'instant, nous avons vu comment appliquer des sélecteurs sur les éléments HTML
- Nous pouvons définir nos propres sélecteurs:
 - class (#) : permet de définir un sélecteur associé à plusieurs éléments
 - id (.) : permet de définir un sélecteur à un élément

```
<div id="top">  
<h1>Chocolate curry</h1>  
<p class="intro">The Recipe...</p>  
<p class="intro">Mmm mm mmmmm</p>  
</div>
```

```
#top {  
  background-color: #ccc;  
  padding: 1em  
}  
  
.intro {  
  color: red;  
  font-weight: bold;  
}
```

CSS: classe et ID (2)

- Il est même possible d'optimiser:
 - si le document HTML / XHTML est bien formé, le CSS peut prendre compte de l'emboîtement de structures

```
<div id="top">  
  
  <h1>Chocolate curry</h1>  
  
  <p>The Recipe...</p>  
  
  <p>Mmm mm mmmmm</p>  
  
</div>
```

```
#top {  
  background-color: #ccc;  
  padding: 1em  
}  
  
#top h1 {  
  color: #ff0;  
}  
  
#top p {  
  color: red;  
  font-weight: bold;  
}
```

- <http://www.csszengarden.com/>