

XML, et HTML, comme langages de structuration

Les feuilles de style hiérarchisées de présentation
dites « CSS »

Les CSS qu'est-ce ?

- CSS est un langage de style qui sert à définir la présentation de documents HTML, SVG, et plus généralement XML.
- CSS couvre les usages des polices, des couleurs,
- CSS règle les marges, les lignes, la hauteur, la largeur,
- CSS choisit les images d'arrière-plan, les positionnements évolués et bien d'autres choses.
- CSS met en facteur commun des formes de présentations autrement définies directement en HTML.
- À l'usage des CSS, HTML n'est censé définir que des catégories de contenus, les feuilles CSS associées se chargeant de définir leurs modalités de présentation.
- L'adoption des CSS a nécessité des adaptations des navigateurs, qui se sont (tant bien que mal) aujourd'hui généralisées.

Syntaxe de base

Autrefois en HTML pour définir un fond Bleu, on écrivait :

```
<html>
  <body bgcolor="#0000FF">
    <!-- -->
  </body>
</html>
```

Maintenant avec une feuille CSS on écrit d'une part :

```
<html>
  <link rel="stylesheet" type="text/css" href="./style.css" />
  <body>
    <!-- -->
  </body>
</html>
```

Et d'autre part, dans une ressource (ici style.css) :

```
.body {background-color: #0000FF;}
```

Modalité de spécification interne de style CSS

```
<html>
  <head>
    <title>Exemple de CSS</title>
    <style type="text/css">
      body {background-color: #00FF00;}
    </style>
  </head>
  <body>
    <p>Cette page est verte</p>
  </body>
</html>
```

Cette façon est moins puissante, puisqu'elle n'apporte pas la possibilité de mutualiser les définitions.

La précédente permet de séparer la définition des contenus de la celle de la forme, qui dès lors peut être modifiée à tout moment, pour tous les documents concernés

Principes syntaxiques des CSS

le modèle fondamental de CSS :

```
selector {property: value;}
```

↑
À quelle(s)
balise(s) HTML
la propriété
s'applique
(p.ex. "body")

↑
La propriété, par
exemple, la couleur
d'arrière-plan
("background-color")

↙ La valeur de la
propriété de couleur
d'arrière-plan, par
exemple, le rouge
("#FF0000")

Invocation depuis un fichier HTML :

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

Invocation depuis un fichier XML :

```
<?xml-stylesheet type="text/css" href="biblio.css"?>
```

Propriétés : couleurs et décors de fond

Les couleurs se définissent par les propriétés :

- **color** :

```
h1 {color: #ff0000;} /*donne un titre en rouge */
```
- **background-color** :

```
h2 {  
background-color: #000000;  
color: #ff0000;}  
/* donne un titre en rouge sur fond noir */
```
- **background-image** :

```
body {  
background-image: url("flower.jpeg") ;  
background-repeat: repeat;  
}  
/* decore le fond avec un (magnifique) papier peint à fleurs*/
```
- **background** : factorise plusieurs propriétés de fond

```
body { background: url("flower.jpeg") no-repeat fixed  
right bottom;} /* place au fond une fleur fixe en bas à droite*/
```

Que fait donc la police ?

Les polices définissent en CSS avec les propriétés :

- **font-family** sert à indiquer la liste prioritaire des polices à utiliser pour l'affichage d'un élément donné ou d'une page Web
`h1 {font-family: arial, verdana, sans-serif;}`
- **font-style** définit les inclinaisons de style (normal, italique, oblique)
`h2 {font-style: italic}`
- **font-variant** choisit une variante entre normal ou small-caps (petites capitales) d'une police
`h3 {font-variant: small-caps}`
- **font-weight** définit la graisse de la police
`h4 {font-weight: bold}`
- **font-size** définit la taille (absolue ou proportionnelle) de la police
`p {font-size: 18pt} title{font-size: 18%}`
- **font** met en facteur plusieurs propriétés de polices
`h5 {font: italic bold 30px arial, sans-serif}`

L'aspect des textes

Pour formater l'aspect d'un texte CSS traite les propriétés :

- **text-indent** décale la première ligne d'un paragraphe

```
p {text-indent: 10px;}
```
- **text-align** définit l'alignement gauche, droite, centré, justifié

```
th {text-align: right;}  
td {text-align: center;}  
p {text-align: justify;}
```
- **text-decoration** pour surligner, barrer, ou souligner un texte.

```
h1 {text-decoration: overline; }  
h2 {text-decoration: line-through;}  
h3 {text-decoration: underline;}
```
- **letter-spacing** règle l'espace entre caractères

```
p {letter-spacing: 3px;}
```
- **text-transform** contrôle la capitalisation du texte
(capitalize, uppercase, lowercase, none)

```
h1 {text-transform: uppercase;}
```


identification et regroupement d'éléments en classes

Affecter un style à une balise l'impose à toutes ses occurrences .

Pour pouvoir autoriser plusieurs présentations d'un type de balise, les CSS utilisent une notion de classes attribuables à une balise.

Une classe se spécifie isolément ou en association explicite avec une balise.

Soit par exemple dans la feuille de style :

```
p.blue {color: #0000FF;}  
.green {color: #00FF00;}
```

Et dans le document :

```
<h2 class='green'> un titre en vert </h2>  
<p class= 'blue'> un paragraphe en bleu </p>
```

Les CSS ont aussi une notion d'identifiant de présentation associé à l'identifiant d'un élément d'un document.

```
#toto {color: #0000FF;font-weight=bold;}  
<machin id='toto'>le machin toto est toujours bleu et gras  
</machin>
```

regroupement d'éléments par span et div

span et **div** sont des éléments structurels classiques de regroupement en HTML.

L'usage des classes CSS et des styles associés à des identifiants permettent des regroupements élégants de spécification de présentations.

Exemple de style CSS

```
span.rouge {color:red;}
```

```
#ID123{color:blue;}
```

Exemple d'usage dans un document

```
<span class='rouge'><p>paragraphe de l'espace  
rouge</p></span>
```

```
<div id='ID123'><p>toute la division 123 est bleue</p></div>
```

Présentation des liens

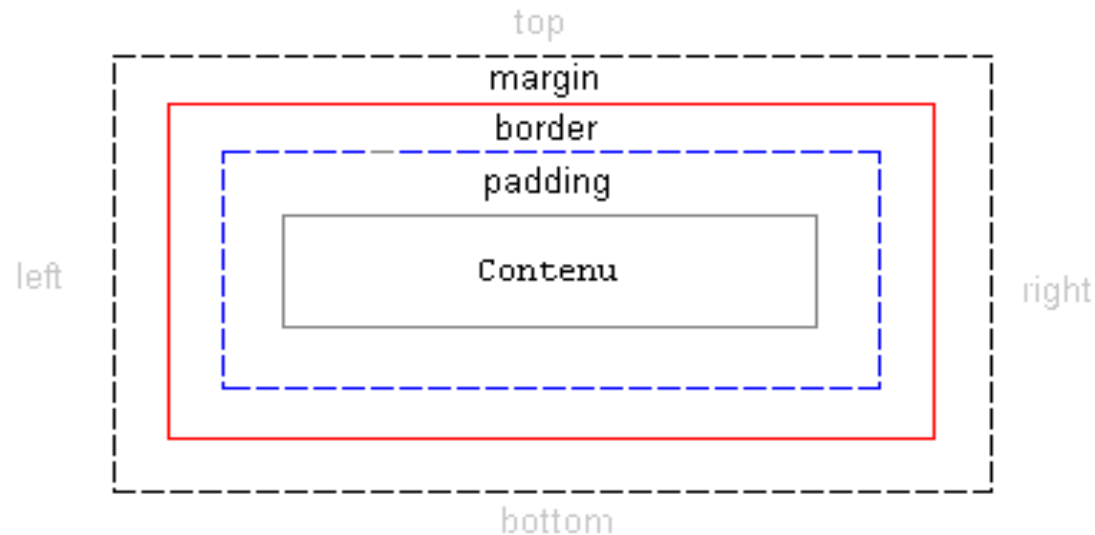
Pour présenter l'histoire d'utilisation des liens, CSS définit des pseudo-classes invoquées selon l'état d'un objet :

- **a:link** présente un lien non encore visité.
- **a:visited** présente un lien déjà visité.
`a:link {color: blue;}`
`a:visited {color: red;}`
- **a:active** présente à un lien activé,
`a:active {text-decoration : underlined;}`
- **a:hover** définit le comportement de survol du lien
`a:hover {color: orange;font-style: italic;}`

Les emboîtements

Les CSS encadrent dans une boîte d'encombrement graphique tous les éléments qu'ils représentent, avec des marges externes, une bordure et une marge interne, et ce de tous côtés.

Aux emboitements du balisage correspondent les emboitements de présentation



- bord de la marge
- bord de la bordure
- - - bord de l'espace
- bord du contenu

Marges et remplissages de boîtes

Les marges se spécifient explicitement, ou par une convention implicite de parcours (dans l'ordre top, right, bottom left) :

Exemple de spécification :

```
body{
margin-top: 100px;
margin-right: 40px;
margin-bottom: 10px;
margin-left: 70px; background-color:blue;}
h1 { margin: 10px 4px 10px 20px; background-color:red;
padding: 20px 20px 20px 80px;}
h2 { margin: 10px 4px 10px 20px ; background-color:yellow;
padding-left: 80px;}
p { margin: 10px 4px 10px 20px; background-color:green;}
```

Bordures de boîtes

Les boîtes peuvent présenter des bordures visibles d'encadrement, dont peuvent être spécifiées individuellement la couleur, l'épaisseur et le style de trait, ou tout à la fois :

border-width :

```
body{border-width: thick;}
```

border-color

```
h1{border-color: gold;}
```

border-style

```
h2{border-style: dotted;}
```

border

```
p{border: 1px solid blue;}
```

Hauteur et largeur de boîte

Les CSS donnent la possibilité de fixer la hauteur et la largeur d'une boîte, par les propriétés `height` et `width`. Exemple à essayer :

```
div.box1 {  
width: 300px ;  
margin: 10px 10px 10px 10px;  
border: 1px solid black;  
background: orange;  
}  
div.box2 {  
height: 500px;  
width: 200px ;  
margin: 10px 10px 10px 10px;  
border: 1px solid black;  
background: yellow;  
}
```

les flottants

Une boîte peut être placée, « flotter » pour s'insérer dans d'une autre de sorte d'être entouré par le contenu de celle-ci.

```
<div id="rose"></div>
<p class="coule">Mignonne, allons voir si la rose<br/>Qui se matin
avoit desclose<br/>Sa robe de pourpre au soleil,<br/>A point perdu
ceste vesprée<br/>
Les plis de sa robe pourprée<br/>Et son teint au vostre pareil<br/>
Las ! voyez comme en peu d'espace,<br/>Mignonne, elle a dessus la
place<br/>Las ! las ses beautez laissé cheoir !<br/>Ô vraiment
marastre Nature,<br/>Puis qu'une telle fleur ne dure<br/>Que du
matin jusques au soir !<br/>Donc, si vous me croyez, mignonne,<br/>
Tandis que vostre âge fleuronne<br/>En sa plus verte nouveauté,<br/>
Cueillez, cueillez vostre jeunesse :<br/>Comme à ceste fleur la
vieillesse<br/>Fera ternir vostre beauté.</p>
```

float insère une boîte avec les valeurs "left", "right" ou "none".

```
#image {float:left;width: 100px;}
```

Clear contrôle la présentation des éléments suivant les flottants.

```
.coule{clear:both;}
```


Positionnement, absolu ou relatif

Les propriétés de positionnement CSS servent à placer un élément exactement où l'on veut dans la page. Ils s'associent de façon élégante aux flottants. Les positions sont soit :

- Absolues par rapport à l'affichage courant de la fenêtre
`#box1 {position:absolute;top:50px;left:50px;}`
- Relatives à la page initiale
`#box2 {position:relative;left:350px;bottom:150px;}`

Les couches

Les CSS gèrent les superpositions d'éléments en couches, les une cachant éventuellement les autres via la propriété **z-index**, avec des valeurs entières, les plus grandes se plaçant dans l'ordre au dessus des autres.

Exemple :

```
#couche1 {  
    position: absolute;  
    left: 100px;  
    top: 100px;  
    z-index: 1;  
}  
#couche2 {  
    position: absolute;  
    left: 115px;  
    top: 115px;  
    z-index: 2;  
}
```

Utilitaire de validation de CSS du W3C

Le W3C a associé à sa recommandation sur les CSS un outil de validation, destiné à vérifier une feuille de style indépendamment des capacités du navigateur que l'on emploie. Cet outil est accessible à l'adresse :

<http://www.html.net/site/style/default.screen.css>

Pour valider une feuille de style, il suffit de remplacer l'adresse URL qui apparaît ici par celle de cette feuille et de cliquer pour la valider. Le site Web du W3C vous informe alors des erreurs qui s'y trouvent.

Pour des exemples riches sur les possibilités des CSS il est suggéré de visiter les exemples proposés à l'adresse :

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

Fin du module