

# Programmation Internet

## Cours 5

[kn@lri.fr](mailto:kn@lri.fr)

<http://www.lri.fr/~kn>

# Plan

1 Systèmes d'exploitation (1/2) ✓

2 Systèmes d'exploitation (2/2) ✓

3 Réseaux, TCP/IP ✓

4 Web et HTML ✓

5 CSS

**5.1 Introduction**

5.2 Boîtes

5.3 Autres propriétés

5.4 Selecteurs

# Cascading Style Sheets (CSS)

**CSS** : Langage permettant de décrire le **style graphique** d'une page HTML

On peut appliquer un style CSS

- À un élément en utilisant **l'attribut style**
- À une page en utilisant l'élément `<style>...</style>` dans l'en-tête du document (dans la balise `<head>...</head>`).
- À un ensemble de pages en référençant un fichier de style dans chacune des pages

# L'attribut style

```
<a href="http://www.u-psud.fr" style="color:red">Un lien</a>
```

Apperçu:

Un lien

Inconvénients :

- il faut copier l'attribut style pour tous les liens de la page
- modification de tous les éléments difficiles

# L'élément style

```
<html>
  <head>
    <title>...</title>
    <style>
      a { color: red; }
    </style>
  </head>
  <body>
    <a href="...">Lien 1</a> <a href="...">Lien 2</a>
  </body>
</html>
```

**Apperçu :**

**[Lien 1](#) [Lien 2](#)**

**Inconvénient : local à une page**

# Fichier .css séparé

**Fichier** style.css:

```
a { color: red; }
```

**Fichier** test.html:

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

**Modifications & déploiement aisé**

# Syntaxe

Une **propriété** CSS est définie en utilisant la syntaxe:

```
nom_prop : val_prop ;
```

■ Si on utilise l'attribut `style` d'un élément:

```
<a href="..." style="color:red;border-style:solid;border:1pt;">Lien 1</a>
```

■ Si on utilise un fichier `.css` ou une feuille de style:

```
a {  
    color : red;  
    border-style: solid;  
    border: 1pt;  
}  
h1 { /* Le style des titres de niveau 1 */  
    text-decoration: underline;  
    color: green;  
}
```

# Plan

1 Systèmes d'exploitation (1/2) ✓

2 Systèmes d'exploitation (2/2) ✓

3 Réseaux, TCP/IP ✓

4 Web et HTML ✓

5 CSS

5.1 Introduction ✓

5.2 Boîtes

5.3 Autres propriétés

5.4 Selecteurs



# Unités de longueur

CSS permet de spécifier des longueurs comme valeurs de certaines propriétés (position et taille des éléments, épaisseur des bordures, ...). Les longueurs **doivent** comporter une unité. Les unités reconnues sont:

**px** : pixel

**in** : pouce (2,54cm)

**cm** : centimètre

**mm** : millimètre

**pt** : point (1/72ème de pouce, 0,35mm)

**pc** : pica (12 points)

**em** : facteur de la largeur d'un caractère de la police courante

**ex** : facteur de la hauteur d'un caractère « x » de la police courante

**%** : pourcentage d'une valeur particulière (définie par propriété)

**vh** : *viewport height* (% de la hauteur de la partie visible de la page) CSS3

**vw** : *viewport width* (% de la largeur de la partie visible de la page) CSS3

# Boîte

Chaque élément de la page HTML possède une **boîte rectangulaire** qui délimite le contenu de l'élément:

**Lien 1**

La **taille t du contenu** est calculée pour que:

$(\text{height}|\text{width}) = \text{padding} + \text{margin} + \text{border} + t$

# Marge, bordure, ajustement

On peut spécifier jusqu'à 4 valeurs:

- 1 valeur: toutes les dimensions égales à cette valeur
- 2 valeurs: haut et bas égal à la première valeur, gauche et droite égale à la deuxième
- 3 valeurs: haut à la première valeur, gauche et droite égale à la deuxième, bas égal à la troisième
- 4 valeurs: haut, droit, bas, gauche

```
span {  
  padding:10pt 20pt 5pt 0pt;  
  margin:10pt 5pt;  
  border-width:3pt;  
  border-color:red blue green;  
  border-style:solid dotted;  
}
```

Du  dans une boîte

# Modes d'affichage

La propriété *display* contrôle le mode d'affichage d'un élément:

**none :** l'élément n'est pas dessiné et n'occupe pas d'espace

**inline :** l'élément est placé sur la ligne courante, dans le flot de texte. La taille du contenu (avec les marges, ajustements et bordures) dicte la **taille de la boîte**, `height` et `width` sont ignorés (`<i>`, `<b>`, `<span>`, `<em>`, ... sont *inline* par défaut).

**block :** l'élément est placé seul sur sa ligne. La taille est calculée automatiquement mais peut être modifiée par `width` et `height` (`<div>`, `<h1>`, `<p>`, ... sont *block* par défaut)

**inline-block** positionné comme *inline* mais la taille peut être modifiée comme **:** pour *block*

# Modes d'affichage (exemples)

```
a { display: inline; ... }  
a { display: block; ... }
```

Le [lien 1](#), le [lien 2](#) et le [lien 3](#).

Le

[lien 1](#)

, le

[lien 2](#)

et le

[lien 3](#)

.

Le [lien 1](#), le [lien 2](#) et le [lien 3](#).

```
a { display: inline-block;  
  width: 4em;  
  height: 2em;  
  ... }
```

# Positionnement

Le type de positionnement est donné par la propriété **position**

**static** : positionnement « automatique »

**fixed** : positionnement par rapport à la fenêtre du navigateur (la boîte est supprimée du flot)

**relative** : positionnement « relatif » par rapport à la position normale

**absolute** : positionnement « absolu » par rapport à l'ancêtre le plus proche qui n'est pas *static*

**Pour fixed, relative et absolute, les propriétés top, bottom, left et right dénotent les décalages respectifs.**

# Positionnement (exemple)

```
a { position: static;
... }
a { position: fixed;
right:10pt;
top: 10pt;
}
```

```
a { position: relative;
left: 10pt;
bottom: -5pt;
... }
a { position:absolute;
right:0pt;
bottom: 10pt;
}
```

<ul sty

- Positionnement **static**

- Positionnement

- Positionnement **relative (left:10pt,bottom:-5pt)**

- Positionnement

- absolute (right:10pt,bottom:10pt)**

# Plan

1 Systèmes d'exploitation (1/2) ✓

2 Systèmes d'exploitation (2/2) ✓

3 Réseaux, TCP/IP ✓

4 Web et HTML ✓

5 CSS

5.1 Introduction ✓

5.2 Boîtes ✓

5.3 Autres propriétés

5.4 Selecteurs



# Couleurs

Les couleurs peuvent être données:

- par nom symbolique: **red**, **blue**, **purple**, ...
- en hexadécimal: **#xxyyzz**, avec  $00 \leq xx, yy, zz \leq ff$
- en décimal: **rgb(x, y, z)**, avec  $0 \leq x, y, z \leq 255$
- en décimal avec transparence: **rgba(x, y, z, a)**, avec  $0 \leq x, y, z \leq 255$  et  $0 \leq a \leq 1$  **CSS3**

# Propriétés du texte

Certaines propriétés permettent d'alterer le rendu du texte d'un élément

<b>direction :</b>	<code>ltr</code> ou <code>rtl</code> (orientation du texte)
<b>text-transform :</b>	<code>capitalize</code> , <code>uppercase</code> , <code>lowercase</code>
<b>text-decoration :</b>	<code>underline</code> , <code>overline</code> , <code>line-through</code>
<b>text-align :</b>	<code>left</code> , <code>right</code> , <code>center</code> , <code>justify</code>
<b>text-indent :</b>	longueur du retrait de paragraphe

# Propriétés de la police

**font-family :** liste de nom de polices séparées par des virgules (Helvetica, sans, "Times New Roman")

**font-style :** normal, italic

**font-weight :** normal, lighter, bold, bolder

**font-size :** soit une longueur soit xx-small, x-small, small, medium, large, x-large, xx-large

On peut aussi spécifier un descripteur de police [CSS3](#)

```
@font-face {  
  font-family: Toto;  
  src: url(toto.ttf);  
}  
a { font-family: Toto; }
```

# Plan

1 Systèmes d'exploitation (1/2) ✓

2 Systèmes d'exploitation (2/2) ✓

3 Réseaux, TCP/IP ✓

4 Web et HTML ✓

5 CSS

5.1 Introduction ✓

5.2 Boîtes ✓

5.3 Autres propriétés ✓

5.4 Selecteurs

# Selecteurs

On peut sélectionner finement les éléments auxquels un style s'applique

- x :** tous les éléments dont la balise est x
  - .foo :** tous les éléments dont l'attribut `class` vaut foo
  - #foo :** l'élément dont l'attribut `id` vaut foo (les id doivent être uniques)
  - X Y :** tous les éléments sélectionnés par Y qui sont des descendants d'éléments sélectionnés par X
  - X > Y :** tous les éléments dont sélectionné par Y qui sont des fils d'éléments sélectionnés par X
  - a:visited :** les liens déjà visités
  - a:link :** les liens non visités
  - X:hover :** élément sélectionné par X et survolé par la souris
- ```
div.foo ul li a:visited { color: red; }
```