

WWW

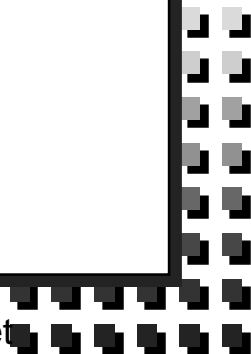
Programmation sur Internet

Cours CSS-2 : stratégie de conception des pages web



Jean-Michel Ilié

IUT Paris 5 - département informatique



Question d'alignement des conteneurs

utiliser une propriété d'alignement pour
les conteneurs à aligner.

Avec ...

display:inline

- ☞ les conteneurs se suivent mais
- ☞ il n'y a pas d'espace au dessus et en dessous
- ☞ pas de gestion des hauteurs

display:inline-block

- ☞ Les conteneurs se suivent mais se comportent en blocs
- ☞ contrôler les dimensions des blocs.

float:left

- ☞ Les conteneurs se suivent en blocs flottants
- ☞ contrôler les dimensions des blocs
- ☞ prévoir d'arrêter le flottement avant le prochain système de blocs en flottement.

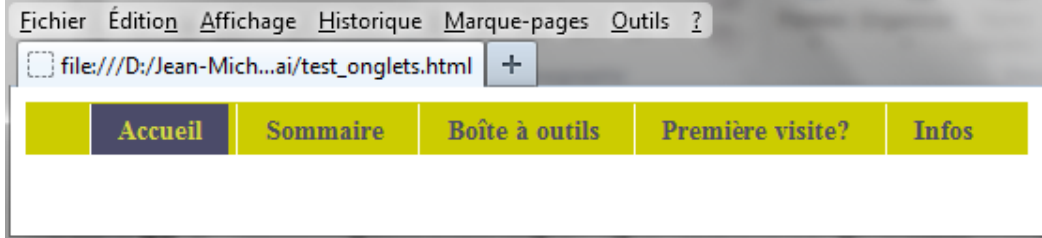
Exemple de menu horizontal redéfinissant une liste comme inline

```
<html><head><style type="text/css">
UL#nav-box {list-style-type: none;
  font-weight: bold; font-size: 0.9em;
  padding-right: 1em; padding-left: 0px;
  padding-bottom: 5px; padding-top: 5px;
  margin : 0px 0px 1.5em;
  color: #4b4b69; background-color: #cc0;
  text-align: right}
```

```
UL#nav-box LI {padding-right: 0px; display: inline; }
UL#nav-box LI A {padding-right: 1em; padding-left: 1em;
  padding-bottom: 5px; padding-top: 5px;
  border-left: #ffffff 1px solid;
  color: #4b4b69;
  text-decoration: none}
```

```
UL#nav-box LI A:hover {border-left: #ffffff 1px solid;
  color: #d9d940;
  background-color: #4b4b69}
```

```
UL#nav-box A:active {border-left: #ffffff 1px solid}
</style></head>
```

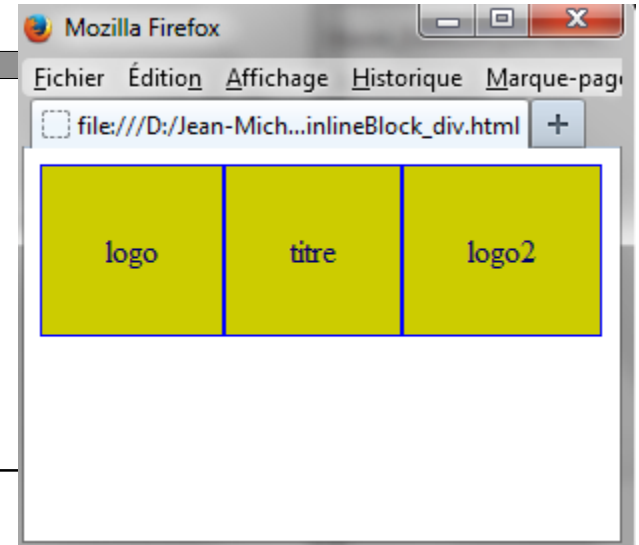


```
<body>
<ul id=nav-box>
  <li><a href="">Accueil</a></li>
  <li><a href="">Sommaire</a></li>
  <li><a href="">Boîte à outils</a></li>
  <li><a href="">Première visite?
    </A></li>
  <li><a href="">Infos</a></li>
</ul>
</body></html>
```

Exemple d'entête horizontale avec inline-block

```
<html><head><style>
*{ margin:0;
}
div#header { font-size:0;
}
span{ display:inline-block;
padding:2em;
border-left : 1pt solid blue;
color:#000069;
background-color: #cc0;
font-size:15;
}
</style></head>
```

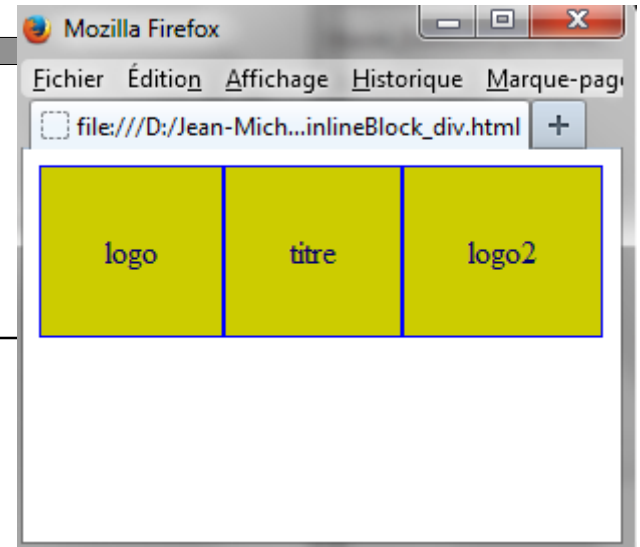
```
<body>
<div id="header">
<span id="G">logo1</span>
<span id="M">titre</span>
<span id="D">logo2</span>
</div>
</body></html>
```



Exemple d'entête horizontale avec div et float

```
<html><head>  
<style>  
  
  div #header div{  
    float:left;  
    width:31%;  
    height:5em;  
  }  
</style></head>
```

```
<body>  
  
  <div id="header">  
    <div id="G">logo1</div>  
    <div id="M">titre</div>  
    <div id="D">logo2</div>  
  </div>
```



Note : penser à dimensionner les blocs floats (width et height).

Exemple flottant

avec 2 ensembles de blocs flottants

```

<html><head>
<style>

div div{
  float:left;
  ...
}

.clear:{
  clear:left;
}

</style></head>

```

```

<body>

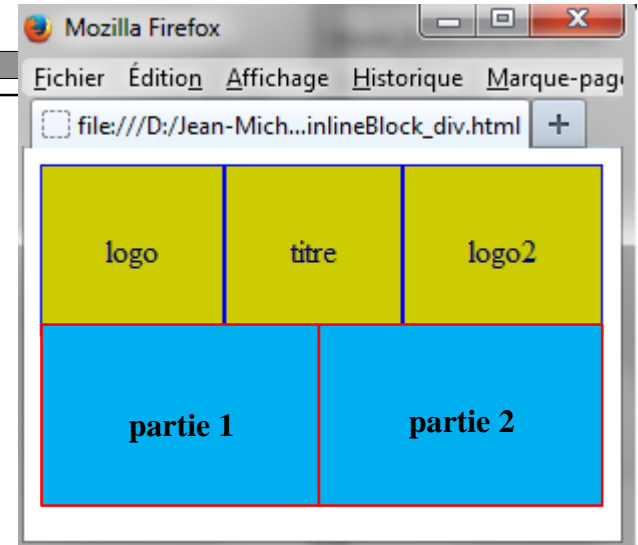
<div id="header">
  <div id="G">logo1</div>
  <div id="M">titre</div>
  <div id="D">logo2</div>
</div>

<div class="clear"></div>

<div id="page">
  <div id="P1">partie1</div>
  <div id="P2">partie2</div>
</div>

</body></html>

```



clear:left nettoie les données de flottement – gauche ce qui permet de réaliser les suivants sans effet de bord.

Question de centrage des blocs

☞ **centrage horizontal :**

margin-left : auto;
margin-right : auto;

*margin-left et margin-right
à la même valeur,
ou à la valeur auto,
force le centrage horizontal*

☞ **centrage vertical :**

*combinaison de 3 propriétés
sur le bloc à centrer :*

position : relative;

*/*sort le bloc du flux courant*/*

top : 50%;

*/*centrage vertical du **haut** du bloc*/*

margin-top:-15px;

*/*remonter le bloc à centrer d'1/2 hauteur
pour en répartir le contenu */*

*nécessite la connaissance
de la taille du bloc à centrer*

Exemple : centrage d'un paragraphe dans un bloc

```
<html> <head> <style>
div#bleu{
  width:8em;height:6em;
  margin:3%;
}
div#bleu p {
  width:50px; height:30px;
  margin : auto; /*pou centrage Hz du paragraphe*/
  position : relative;
  top : 50%; /*centrage V tenant compte de la*/
  margin-top:-15px; /*1/2 hauteur du paragraphe*/
}
</style> </head>
```



```
<body>
  <div id="bleu">
    <p>partie1</p>
  </div>
</body> </html>
```

Question d'alignement des contenus textuels et inline

Alignement horizontal

text-align : left / right / justify

justification horizontale

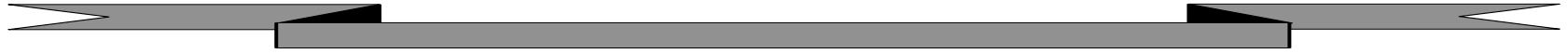
Alignement vertical

*vertical-align : val / val% / nb*font-size* centrage vertical

suivant de nombreux modes

L'alignement vertical est utile quand il s'agit d'aligner des contenus dans des conteneurs contigus, par exemple une image et un bloc contenant un titre pour l'image .

Question du contrôle des débordements de contenu de bloc



*les gros contenus
peuvent outrepasser
les dimensions des blocs.*

2 solutions :

height : auto *force le navigateur à s'adapter sa hauteur au contenu*

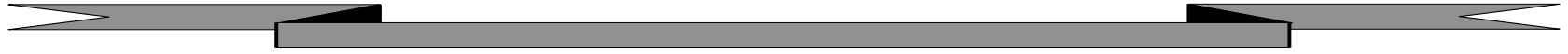
overflow: visible / hidden / *rend visible ou non, le contenu débordant*

scroll / *impose des ascenseurs*

auto *les ascenseurs n'apparaissent qu'en cas de besoin.*

overflow est utile quand les blocs conteneurs sont de taille fixe.

Question du contrôle des dimensions d'un bloc



*les redimensionnements de fenêtre
peuvent endommagés
une présentation graphique.*

min-width

max-width

contrôle de la largeur du bloc

min-height

max-height

contrôle de la hauteur du bloc

*min-width et min-height sont utiles
pour conserver les alignements au sein d'un bloc,
forçant ses ascenseurs à apparaître
en cas de sous-dimensionnement de la fenêtre*

Gestion des contenus textuels et inline

- font-family** *déclaration des polices de caractères*
- font-size** *taille de la police de caractères (px)*
- text-height** *règlage de l'interligne*
- letter-spacing** et **word-spacing** *règlage manuel de l'espacement*
: normal / *val* / *inherit* (*val s'ajoute à la valeur "normal" de la police*)
- white-spacing** : *règlage automatique de l'espacement*
normal = 1 seul espace qqe soit le nb d'espaces, ou
pre = tous les espaces espace qqe
nowrap = idem normal mais les retours à la ligne sont supprimés
- word-wrap** : normal / *break-word* *automatisation de la césure des mots*

Intégration des contenus audios et videos

`<video src="url"> texte si pb </video>`

Mono type

Idem pour
<audio>

`<video> description fichiers formatés et
texte si pb </video>`

Multi-types

Attributs des balises audio et video :

- ☞ `control="controls"` *Affiche une barre de contrôle (lecture/pause...)*
- ☞ `height, width` *Dimensions*
- ☞ `poster="une url d'image"` *Affiche une image fixe*
- ☞ `loop="loop"` *Lecture en boucle*
- ☞ `muted="muted"` *Supprime le son*
- ☞ `preload="none| metadata|auto"` *Préchargement video complète ou
métadonnées seules*

Question des formats (type MIME) pour les fichiers videos

*Chaque navigateur n'accepte qu'un
sous ensemble des formats de fichiers !*

<i>format videos:</i>	.mp4	.ogg	.oga	.webm
Firefox		oui		oui
Chrome	oui		oui	oui
Safari	oui			
Opera		oui		oui
IE Explorer	oui			oui

Prévoir 2 fichiers de formats différents pour couvrir les navigateurs !

Question des formats (type MIME) pour les fichiers audios

*Chaque navigateur n'accepte qu'un
sous ensemble des formats de fichiers !*

<i>Formats audios</i>	.mp3	.wav	.ogg .oga	.acc
Firefox		oui	oui	
Chrome	oui	oui	oui	oui
Safari	oui	oui		oui
Opera		oui	oui	
IE Explorer	oui			oui

Prévoir au moins 2 fichiers de formats différents pour couvrir les navigateurs !

Intégration multisources audios et videos

<video>

<source src="url" type="typeMIME" media="(règles)"/> *

</video>

idem pour

<audio> </audio>

<video controls="controls" height="250" width="400">

<source src="url" type="video/mp4">

<source src="url2" type="video/webm">

problème audio

</video>

*2 sources aux formats
mp4 et webm*

*message à afficher
en cas de problème*

Adaptation de la présentation aux navigateurs

*Attachement conditionnel
de fichier de style
défini pour IE*

```
<!--[if IE]>  
  <link ..... href="styles-ie.css" />  
<![endif]-->
```

*pb : les navigateurs
ne respectent pas tous et toutes
les normes du W3C.
Comment adapter sa présentation
à différents navigateurs et
à leurs différentes versions ?*

*Utilisation de règles ou propriétés spécifiques pour différents navigateurs
(possible car les navigateurs son permissifs)*

Autres exemples d'attachement conditionnel

Faire un commentaire

```
<!--[if IE]>  
  <link ..... href="styles-ie.css" />  
<![endif]-->
```

Autres :

```
<!--[if !IE]>
```

Avec versions :

```
<!--[if (gt IE 5)&(lt IE 7)]>
```

Opérateurs des conditions :

=

gt pour > «greater than»

gte pour ≥ «greater than equal»

lt pour < «less than»

lte pour ≤ «less than equal».

/ *ou*

& *et*

! *non*

Adaptation de la présentation aux navigateurs

* `html {}` *seul IE n'ignore pas cette règle, qui indique une balise parent de html*

```
* html h1 {  
    float: left;  
}
```

!important *cet argument css valide est un indicateur de priorité dans une règle
mais IE ignore cet argument.*

*cas où la propriété
doit être spécifiée
pour tous les navigateurs*

```
body {  
    background-color : green !important;  
    background-color : red;  
}
```

Adaptation de la présentation aux navigateurs

**Certaines propriétés sont nouvellement introduites
mais sous des noms différents dans les navigateurs :**
souvent seul le préfixe de l'appellation change.

```
div.maclass {  
  background : linear-gradient (left, red:30%, yellow:30%) ; --navigateur CSS3  
  background : -moz-lineargradient (...) ;      -- Mozilla-Firefox  
  background : -o-lineargradient (...) ;        -- Opera  
  background : -webkit-linear-gradient (...) ; -- Chrome-Safari-Android  
  background : -ms-linear-gradient (...) ;     -- Microsoft IE  
}
```

Adaptation aux caractéristiques physiques des medias

Enrichissement de la balise link

```
<link rel="stylesheet" type="text/css"
href = "une url.css"
media="terminal [and (expression composite)]" />
```

Media queries :

contraintes permettant de cibler des styles suivant les caractéristiques physiques d'une machine.

aural
braille
print
screen
tv
all

min-width / *max-width* / *dimension navigateur (px)*
min-height / *max-height* /
min-device-height / *max-device-width* / *dimension écran (px)*
min-color_index / *max-color-index* / *nb de couleurs (noir=0)*
min-resolution / *max-resolution* ... *Finesse d'affichage*
(dpi, dpcm : points par pouce ou par cm)

Exemple

*styles communs
déclarés en premier*

```
<link rel="stylesheet" type="text/css" href="/style.css" />  
<link rel="stylesheet" type="text/css" href="/stylel.css"  
media="only screen and (min-width: 1024px)" />  
<link rel="stylesheet" type="text/css" href="/styleh.css"  
media="screen and (max-device-width: 800px)" />
```

*redéfinition
de certains styles
pour un affichage
suivant largeur
d'écran
ou résolution.*

Directives css

Directives css :
Instructions données

Enrichissement de la balise link

```
<link rel="stylesheet" type="text/css"
href = "une url.css"
media="terminal [and (expression composite)]" />
```

aural
braille
print
screen
tv
all

min-width / max-width / dimension navigateur (px)
min-height / max-height /
min-device-height / max-device-width / dimension écran (px)
min-color_index / max-color-index / nb de couleurs (noir=0)
min-resolution / max-resolution ... Finesse d'affichage
(dpi, dpcm : points par pouce ou par cm)

Exemple

*styles communs
déclarés en premier*

```
<link rel="stylesheet" type="text/css" href="/style.css" />  
<link rel="stylesheet" type="text/css" href="/stylel.css"  
media="only screen and (min-width: 1024px)" />  
<link rel="stylesheet" type="text/css" href="/styleh.css"  
media="screen and (max-device-width: 800px)" />
```

*redéfinition
de certains styles
pour un affichage
suivant largeur
d'écran
ou résolution.*