

Développeur HTML5





HTML

Présentation HTML & CSS

Présentation

- Extension :.html
- Langages à balises
- Exemple:

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="fr" lang="fr">
<head>
  <title>Make your own Website !</title>
  <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
  <link type="image/x-icon" rel="shortcut icon" href="http://www.w3.org/Icons/aim-small.gif" />
  <meta name="description" content="W3C - World Wide Web Consortium" />
  <meta name="revisit-after" content="7 days" />
</head>
<body onload="getId();" >
  <div id="body">
    <div id="div_header">
      <div id="quickmenu">
        <div class="element">
          <a href="#">Accueil
        </div>
      </div>
    </div>
  </div>
</body>
</html>
```

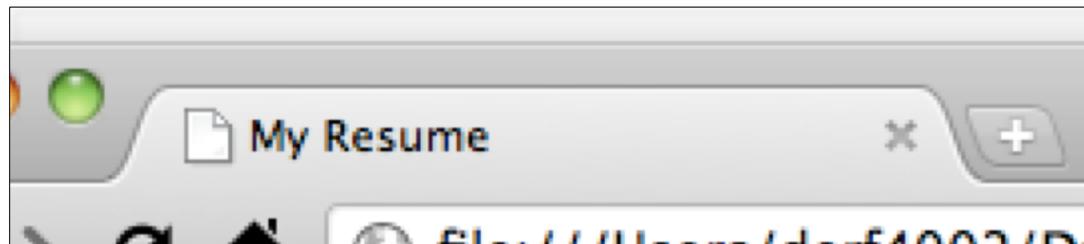
Structure

1. `<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">`
2. `<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">`
3. `<head>`
4. `</head>`
5. `<body>`
6. `</body>`
7. `</html>`

Titre de la page

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
  http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd>
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
  <head>
    <title>My Resume</title>
  </head>
  <body>

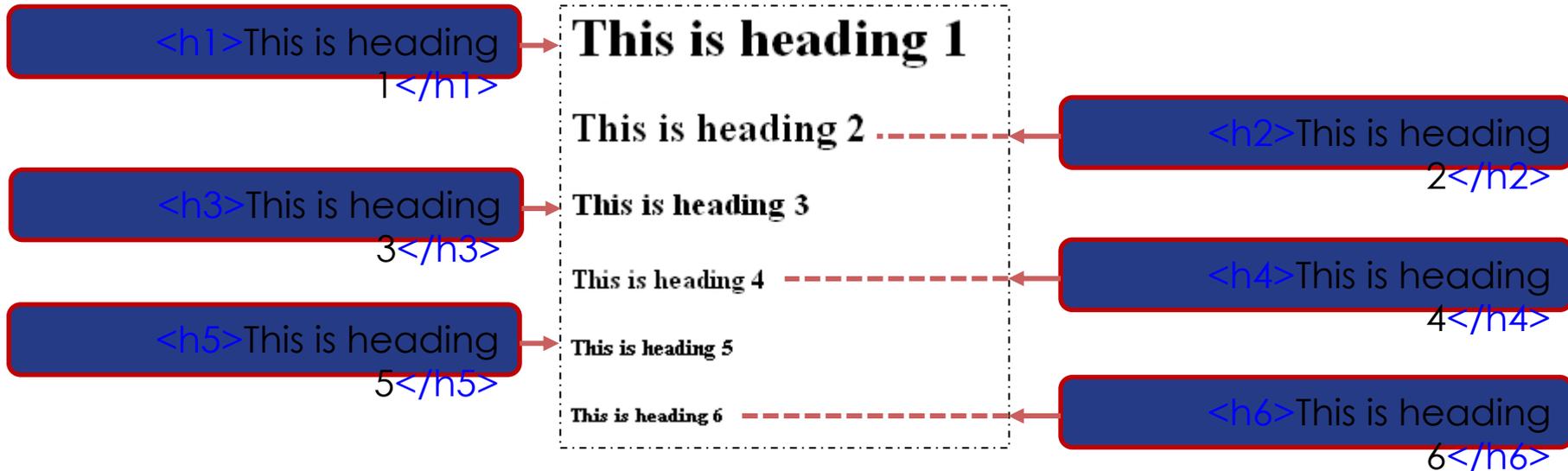
  </body>
</html>
```



Titre

`<hn>...</hn>`

- 2 balises
- Exemple :



Paragraphes

`<p>...</p>`

- Définir un paragraphe
- Avec saut de ligne

`
`

- Saut de ligne

```
<body>
  <p>First Paragraph</p>
  <p>Second <br /> Paragraph</p>
</body>
</html>
```

Listes

Ordonnée:

```
<ol></ol>
```

Non orconnée :

```
<ul></ul>
```

■ Élément : ``

```
<body>  
  <ol>  
    <li>HTML</li>  
    <li>Javascript</li>  
  </ol>  
</body>
```

1. HTML
2. Javascript

```
<body>  
  <ul>  
    <li>HTML</li>  
    <li>Javascript</li>  
  </ul>  
</body>
```

- HTML
- Javascript

Liens

`<a>...`

- Liens vers une autre ressource

- 3 types :

- Lien absolu

```
<a href="http://web.com/Path/file.html">My link</a>
```

- Liens relatifs

```
<a href="file.html">My link</a>
```

- Ancre

- Saut à l'intérieur de la page

```
<a href="file.html#myAnchor">My link</a>
```

Tables

- `<table>...</table>`
- Lignes : `<tr>...</tr>`
- Colonnes :
 - Headers (« *table header* » `<th>...</th>`)
 - Datas (« *table data* » `<td>...</td>`)

Tables

□ Exemple :

```
<table>
  <tr>
    <th>Line 1, Header 1</th>
    <th>Line 1, Header 2</th>
  </tr>
  <tr>
    <td>Line 2, Cell 1</td>
    <td>Line 2, Cell 2</td>
  </tr>
  <tr>
    <td>Line 3, Cell 1</td>
    <td>Line 3, Cell 2</td>
  </tr>
</table>
```

Line 1, Header 1	Line 1, Header 2
Line 2, Cell 1	Line 2, Cell 2
Line 3, Cell 1	Line 3, Cell 2

Tables

- Fusion de cellules
 - Colspan
 - rowspan

Header	
1	Fusionning 2 lines
2	
Fusionning 2 cells	

```
<table>
  <tr>
    <th colspan="2">Header</th>
  </tr>
  <tr>
    <td>1</td>
    <td rowspan="2">Fusionning 2 lines</td>
  </tr>
  <tr>
    <td>2</td>
  </tr>
  <tr>
    <td colspan="2">Fusionning 2 cells</td>
  </tr>
</table>
```

Tables

...

```
<h3><a name="Foreign">Foreign Languages</a></h3>
```

```
<table>
```

```
  <tr>
```

```
    <th colspan="2" rowspan="2"></th>
```

```
    <th colspan="5">Level</th>
```

```
  </tr>
```

```
  <tr>
```

```
    <th>1</th> <th>2</th> <th>3</th> <th>4</th> <th>5</th>
```

```
  </tr>
```

...

Des questions ?

CSS

Présentation HTML & CSS

Découpe de spages

<div>...</div>

- Un document = en ensemble de division.

Bannière

Menu

Main

Footer

```
<body>
  <div id="banner">
</div>
  <div id="menu">
    <ul>
      <li><a href="home.html">Home</a></li>
      <li><a href="forum.html">Forum</a></li>
    </ul>
  </div>
  <div id="main">
    <h1>Welcome on my Website</h1>
    <p>This is the main section where you can find the
most valuable information.</p>
  </div>
  <div id="footer">
    Copyright WebDev Laboratory © 2007-08
  </div>
</body>
```

CSS : où?

- Style « inline » NON

```
<span style="color: red;">Red text... </span>
```

- Interne NON

```
<head>  
  <style type="text/css">  
    p { color: blue; }  
  </style>  
</head>
```

- Externe OUI

```
<link href="main.css" media="all" rel="stylesheet" type="text/css" />
```

Syntaxe

- ▣ Sélecteur : balise, class ou id

- ▣ selector {
 property: value;
 ...
}

```
p {  
  background-color: white;  
  color: black;  
  font-family: Verdana, "Sans serif";  
}
```

```
h1 { background-color: white; color: red; }  
p, h1 { font-family: Verdana, "Sans serif";}
```

Class

- Un groupe de balise

```
.section { background-color: red; }
```

- Exemple

```
<h1 class="section">My Title</h1>
```

id

- Sélection d'une seule balise

```
#MyObjectId { background-color: red; }
```

- Exemple :

```
<div id="MyObjectId">My Text</div>
```

Descendance

- h1, p, h2, p sont enfants de div
- a, strong et em sont enfant du p dans lequel ils sont contenus (mais pas de l'autre p)
- a, strong et em (et h1, p, h2 et p) sont descendants de div



Descendance

- ▣ div est parent de h1 p, h2, p
- ▣ Le 2ème p est parent de a, strong et em.
- ▣ div est ancêtre de a, strong et em (et de h1, p h2 et p)



Descendance

- Pour sélectionner le a descendant de p, nous allons pouvoir écrire :
 - `pa{}`
- Pour sélectionner plusieurs éléments et leur appliquer la même valeur, on les sépare par une virgule.
 - `h1, h2 { }`



Résumé

CSS	HTML ciblé
p { }	<p> </p>
.maclasse { }	<element class="maclasse"> ... </element>
#monid { }	<element id="monid"> ... </element>
p.maclasse { }	<p class="maclasse"> ... </p>
p .maclasse { }	<p> <element class="maclasse"> </element></p>
p, div { }	<p> ... <p> <div> ... </div>
https://www.w3schools.com/cssref/css_selectors.asp	

Pseudo classe

- `a:link` désigne les liens hypertextes non visités.
- `a:visited` désigne les liens hypertextes déjà visités
- `a:hover` désigne un lien survolé (disponible sur tout les éléments HTML)
- `a:active` désigne un lien cliqué.
- `a:focus` désigne un lien qui a le focus (disponible sur certains éléments HTML)
- `a:hover { }`

Pseudo élément

- Sélectionne une partie de l'élément
- after: Insérer quelque chose avant l'élément
- before : Insérer quelque chose après l'élément
- first-letter: sélectionne la première lettre
- first-line: sélectionne la première ligne
- selection: : sélectionne la portion sélectionné par l'utilisateur
- ```
p::first-line {
 color: #ff0000; }
```

# Important /default

- Un propriété mise dans un sélecteur de type classe aura moins d'importance que la propriété dans un sélecteur de type id
- Si 2 propriétés ont le même sélecteur c'est l'ordre d'apparition qui importe, le dernier à la priorité
- 2 propriétés déclarés au même niveau peuvent être mise en priorité avec
  - `maProp: vaValeur !important;`
  - `maProp: vaValeur !default;`

# Couleur

- ▣ **colorname**
  - ▣ Exemple: red
  
- ▣ **rgb(x, x, x)** **[0 ≤ x ≤ 255]**
  - ▣ Exemple : rgb(255, 0, 0)
  
- ▣ **rgb(x%, x%, x%)** **[0 ≤ x ≤ 100]**
  - ▣ Exemple : rgb(100%, 0%, 0%)
  
- ▣ **#rrggbb** **[00 ≤ (rr, gg, bb) ≤ FF]**
  - ▣ Exemple : #FF0000 = #F00

# Taille

- Mesure:
  - %
  - in, cm, mm
  - em, ex
  - pt, pc
  - px

- Exemple :

```
body { font-size: 1.2em; }
```

# Font-family

- font-family : indiquer la (ou les) polices à utiliser
- L'ordre des polices appliquées est celui de la propriété de gauche à droite
- Il faut que la police soit présente sur le poste de l'utilisateur, sinon le navigateur passe à la suivante.
- [www.cssfontstack.com](http://www.cssfontstack.com) : liste des polices websafe par système d'exploitation

```
body {
 font-family: Arial, Helvetica, sansserif;
}
```

# Font-size

- font-size : valeur en unité permet de définir la taille d'une police (=la « hauteur »)
- On utilise les unités vues précédemment (px, em, etc.)

`font-size: 18px;`

# Font weight

- font-weight: valeur permet de déterminer l'épaisseur de la police.
- Principales valeurs possibles : normal, bold (gras), etc...
- Permet de mettre en gras d'autres éléments

`font-weight: bold;`

# Font-style

- font-style : valeur permet de préciser le style de police
- Principales valeurs possibles : normal, italic, oblique.
- Permet de mettre en italique des éléments

`font-style: italic;`

# Line-height

- line-height: valeur change l'interlignage
- Les valeurs possibles sont des valeurs numériques en unité de mesure (px, em, %) mais il vaut mieux ne pas mettre d'unité du tout pour garder plus de flexibilité

`line-height: 1.5px;`

# Color

- color: valeur permet de donner la couleur du texte
- Les valeurs possibles sont toutes les couleurs présentées précédemment

`color: red;`

# Text-align

- text-align : valeur permet d'aligner le texte à gauche, droite ou milieu
- Les valeurs possibles sont : left, right, center, justify
- L'alignement se fait sur le contenu des balises de type bloc (paragraphes, titres, divs, etc.)

```
text-align: left;
```

# Text-transform

- text-transform permet de transformer le texte en majuscule, minuscule
- Valeurs possibles :
  - none
  - capitalize
  - uppercase
  - Lowercase

```
text-transform: capitalize;
```

# Font-variant

- font-variant : valeur permettant de changer la casse de la police
- Principales valeurs possibles :
  - normal
  - small-caps

`font-variant: small-caps;`

# Text-decoration

- text-decoration : valeur permet de changer différents valeurs de « décoration »
- Valeurs possibles :
  - none: permet d'empêcher l'héritage de la propriété et plus particulièrement de supprimer le soulignement par défaut des liens
  - overline : surlignement
  - underline : soulignement
  - line-through: texte barré

`text-decoration: underline;`

# Text-indent

- text-indent : valeur (négative ou positive) permet de spécifier un décalage de la première ligne d'un texte
- Valeurs : unités de mesure de police, positif ou négatif
- Ne fonctionne que sur des éléments de type bloc (ou inline-block)

`text-indent: 10 px;`

# Word-wrap

- Permet de couper les mots long en fin de ligne
  - normal: le retour à la ligne ne se fait qu'entre les mots
  - break-word: le retour à la ligne peut se faire au milieu d'un mot

# Texte

```
1. p {
2. color: orange;
3. text-align: center;
4. text-decoration: underline;
5. text-transform: uppercase;
6. letter-spacing: 10px;
7. font-style: italic;
8. font-weight: bold;
9. line-height: 1.4em;
10. font-family: "Serif";
11. font-size: 1.2em;
12. }
```

CSS  
IS NOT  
COUNTER STRIKE SOURCE !

# Les bordures

- La propriété border permet d'ajouter une bordure à un élément
- Elle a 3 « sous propriétés » possibles : style, color et width.
- On regroupe souvent comme ceci :
  - `border: width style color;`

```
border: 2px solid #009860;
```

# Les bordures

- On peut décomposer les 4 bordures de la manière suivante :
  - `border-top: ...;`
  - `border-left: ...;`
  - `border-right: ...;`
  - `border-bottom: ...;`
- Si on ne précise pas à quelle bordure on applique la propriété, elle s'applique aux 4.

`border-top: 1px solid gray;`

# Style de bordure

- none : pas de bordure (par défaut);
- dotted : pointillés;
- dashed : tirets;
- solid : un trait simple;
- double : bordure double;
- groove : en relief;
- ridge : autre effet relief;
- inset : effet 3D global enfoncé;
- outset : effet 3D global surélevé.

# Des bords arrondis

- `border-radius` : valeur permet de faire des bords arrondis
- Une valeur en unité de mesure (px, em ou même %)
- Le même arrondi partout : `border-radius: 10px;`
- Un arrondi par angle (en haut à gauche, en haut à droite, en bas à droite, en bas à gauche ) :  
`border-radius: 2px 20px 5px 10px;`

# Des ombres

- On peut mettre des ombres sur nos éléments

```
box-shadow: 5px 10px #888888;
```

# Bordure

```
table {
 border: 5px solid green;
 border-collapse: collapse;
}

th {
 padding: 10 40 10 40;
 border-bottom: 3px dotted red;
 border-left: 1px solid blue;
}

td {
 padding: 20px;
 text-align: right;
 border-left: 1px solid blue;
 border-bottom: 1px solid blue;
}
```

| Nom du jeu    | Nombre de cartes en main |
|---------------|--------------------------|
| Texas Hold'em | 2                        |
| Omaha         | 4                        |

# Background-color

- background-color : valeur permet de donner une couleur de fond.
- Valeurs : n'importe quelle couleur CSS
- Sur un élément bloc (prend toute la largeur) ou inline (prend la largeur du contenu)

```
background-color: red;
```

# Background-image

- background-image : valeur permet de définir une image de fond pour l'élément
- Valeur : none ou url("monimage.png" )
- Par défaut : l'image se répète horizontalement et verticalement (comme une mosaïque)

**background-image:** url("img/bg.png");

- background-size: permet de tailler l'image de fond

# Background-repeat

- background-repeat permet de définir la façon dont l'image va se répéter.
  - no-repeat : l'image ne sera pas répétée
  - repeat-x : l'image sera répétée horizontalement
  - repeat-y : l'image sera répétée verticalement

**background-repeat**: repeat-y;

# Background-position

- background-position : valeur permet de donner la position de l'image de fond par rapport au coin haut gauche. A utiliser avec background-repeat: no-repeat.
- Valeurs possibles :
  - Des valeurs numériques sur des axes x et y : background-position: 10px 30px;
  - Valeurs en toute lettre background-position:center left (en haut à droite):

# Tout dans une ligne !

- Il est possible de combiner toutes ces propriétés dans une seule et même ligne :

- `background : propriete1 propriete2 propriete3;`

`background: #7AAAF url("img/etoile.png") no-repeat center left;`

# Fond

```
body {
 background-color: black;
 background-image: url('kalEidoScope.jpg');
 background-repeat: repeat-x;
 background-attachment: fixed;
 background-position: 0% 50%;
}
```



# Liste

- list-style
- list-style-type
- list-style-position
- list-style-image

```
ul {
 list-style-type: disc;
 list-style-position: outside;
 list-style-image: url('orange.gif');
}
```

-  Pomme
-  Banane
-  Pêche
-  Mirabelle

# Petit rappel inline et block

- Sans altération CSS, les éléments de bloc prennent toute la largeur de leur parent, et sont suivis d'un retour à la ligne (body, html, p, div, h1, etc.)
- Les éléments de type inline prennent la largeur de leur contenu, et ne sont pas suivis d'un retour à la ligne (span, a, strong, img, etc.).

# Largeur et hauteur d'un élément : width et height

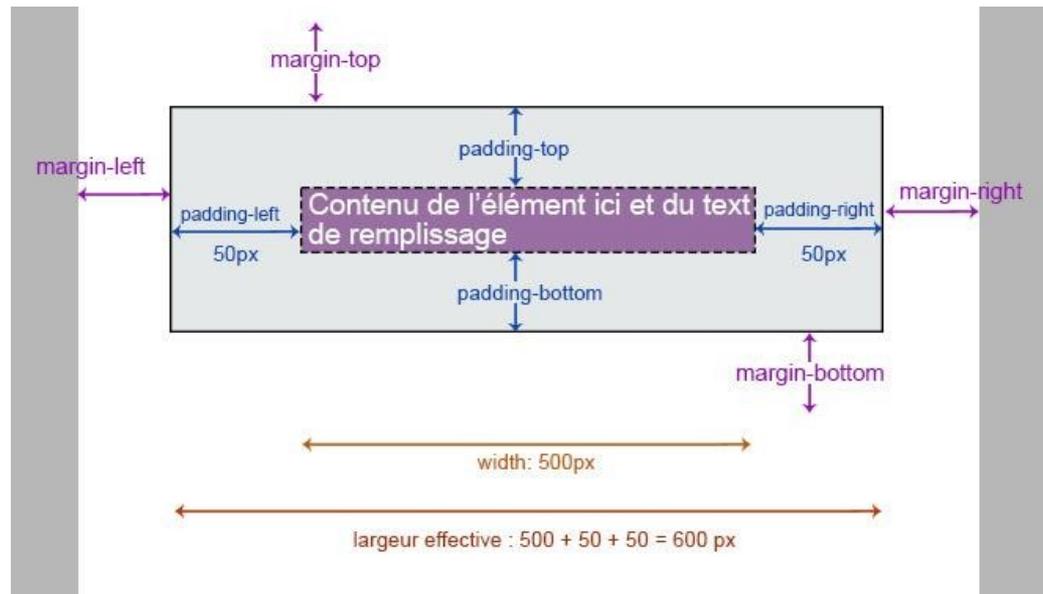
- Valeurs possibles : auto (par défaut, toute la taille du parent), valeur numérique en px, % ou même em

# Max et Min

- On peut utiliser max et min pour déterminer des tailles maximums et minimums (ne fonctionne pas sous IE6) : min-width, min-height, max-width et max-height
- Valeurs : numériques
- Permet par exemple de gérer un débordement d'image :
- `img { max-width: 100%; }` : les images sont limitées à la taille de leur parent

# Marges

- Il existe deux types de marges : intérieure appelée **padding**, et extérieure appelée **margin**



# Marge extérieure « margin »

- La propriété margin détermine l'espace entre le bord de l'élément, et l'élément suivant.
- Par défaut margin: valeur applique la même valeur aux 4 cotés
- Valeur possible : auto, valeur en unité de mesure, positive ou négative
- On peut la décomposer individuellement en
  - margin-top : marge extérieure haute
  - margin-right : marge extérieure droite
  - margin-bottom : marge extérieure basse
  - margin-left : marge extérieure gauche
- Ou rassembler les 4 valeurs :

**margin:** 10px 5px 8px 15px; (dans l'ordre : haut, droite, bas, gauche)

# Padding : la marge « intérieure »

- La propriété padding détermine un espacement entre le bord de la boîte et son contenu
- Par défaut padding : valeur applique la même valeur aux 4 côtés
- Valeur possible : valeur en unité de mesure
- On peut la décomposer individuellement en
  - padding-top : marge intérieure haute
  - padding-right : marge intérieure droite
  - padding-bottom : marge intérieure basse
  - padding-left : marge intérieure gauche
- Ou rassembler les 4 valeurs :

**padding:** 10px 5px 8px 15px;(dans l'ordre : haut, droite, bas, gauche)

# Overflow : pour éviter que ça dépasse

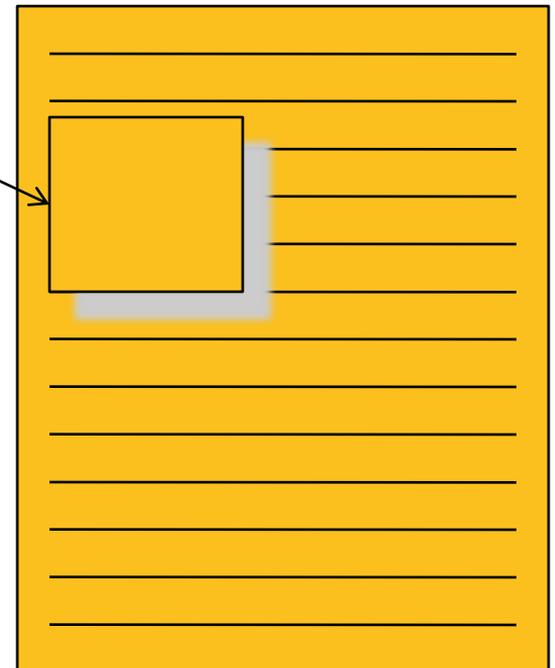
- Il est possible que du contenu texte dépasse l'élément qui le contient (surtout si on définit des hauteurs fixes)
- `overflow: hidden` permet de cacher tout ce qui dépasse.
  - Problème : on perd tout ce qui est caché, impossible de le voir
- `overflow: auto`; permet d'afficher une barre de défilement si nécessaire
- `overflow: visible`; force l'affiche de la barre de défilement

# Float

- Élément flottant
- Valeurs:
  - left
  - right
  - none
  - inherit

```
.menu {
 float: left;
}
```

Floating block



# Position

- clear
- display
- float
- position
- Visibility
- top
- bottom
- left
- right
- position
- overflow
- vertical-align
- z-index

# Les « flottants » et le flux

- La propriété `float: valeur` permet d'extraire des éléments du flux de la page, ce qui signifie que le reste du contenu « coule » autour
- Elle prend 3 valeurs : left, right et none (permet de remettre un élément dans le flux)

`float: left;`

## Les « flottants » et le flux

Images

Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur. Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur. Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur. Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur. Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur.

Images

Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur. Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur. Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur. Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur. Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur. Vivamus sagittis lacus vel augue laoreet rutrum faucibus dolor auctor. Maecenas sed diam eget risus varius blandit sit amet non magna. Aenean lacinia bibendum nulla sed consectetur.

# Clear : bloquer le dépassement des flottants

- La propriété clear : valeur permet à un élément de cesser le contournement des éléments flottants. Il se positionne alors sous les éléments flottants précédents comme si ces derniers étaient restés dans le flux.
- On l'applique sur le premier élément « suivant » dont on veut cesser le contournement.
- Valeurs possibles :
  - clear: left permet d'empêcher le contournement des blocs flottants à gauche
  - clear: right permet d'empêcher le contournement des blocs flottants à droite
  - clear : both permet d'empêcher le contournement des blocs flottants à gauche et à droite

# Clear : bloquer le dépassement des flottants



## Un titre ici

- [Home](#)
- [Service](#)
- [A propos](#)
- [Contact](#)

Soufflé gingerbread cupcake donut tiramisu cookie sweet soufflé cake.  
Candy canes candy canes donut toffee candy canes cheesecake gummies  
apple pie. Pie cupcake cheesecake sugar plum tart donut bear claw caramels

And then... Finally, users can insert a WordPress [gallery], which is kinda ugly and comes with some CSS stuck into the page to style it (which doesn't actually validate, nor does the markup for the gallery). The amount of columns in the gallery is also changable by the user, but the default is three so we'll work with that for our example with an added fourth image to test verticle spacing.

# Clear : bloquer le dépassement des flottants



## Un titre ici

- [Home](#)
- [Service](#)
- [A propos](#)
- [Contact](#)

Soufflé gingerbread cupcake donut tiramisu cookie sweet soufflé cake. Candy canes candy canes donut toffee candy canes cheesecake gummies apple pie. Pie cupcake cheesecake sugar plum tart donut bear claw caramels

And then... Finally, users can insert a WordPress [gallery], which is kinda ugly and comes with some CSS stuck into the page to style it (which doesn't actually validate, nor does the markup for the gallery). The amount of columns in the gallery is also changable by the user, but the default is three so we'll work with that for our example with an added fourth image to test verticle spacing.

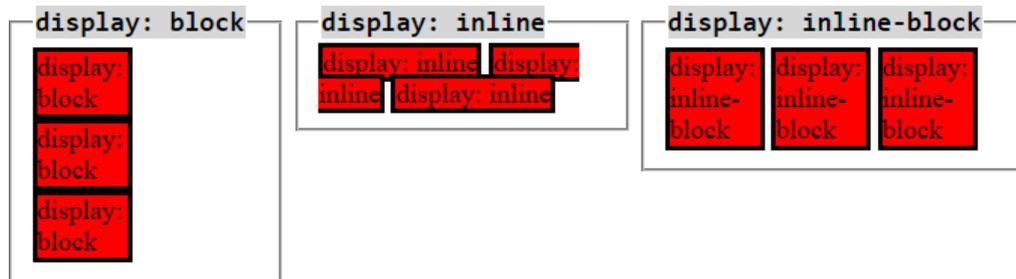
# La propriété « display » pour mise en page

- La propriété `display`: valeur va permettre une mise en forme avancée.
- Quelques unes des valeurs possibles sont : `block`, `inline`, `inlineblock` et `none`
- Ces valeurs sont appliquées par défaut en fonction des éléments
  - `p`, `div`, `h1`, etc. : `display :block`;
  - `span`, `a`, `em`, `strong`, etc. : `display : inline`;
- Il est possible de transformer un élément inline en élément de bloc avec `display: block`
- Il a alors toutes les propriétés d'un élément de bloc : par défaut, il prend toute la largeur de son parent, on peut en changer la dimension et est suivi d'un retour à la ligne.

```
display: block;
```

# Display:inline-block

- `display: inline-block` est une propriété hybride qui permet à un élément d'avoir les propriétés d'un élément en ligne (pas de retour à la ligne après l'élément), mais avec les propriétés d'un bloc (possibilité d'avoir une dimension et des marges)



# Display:none

- `display: none` permet de retirer un élément du flux de la page : il n'est plus visible, et la place qu'il occupait est disponible pour d'autres éléments, c'est comme s'il n'existait pas (contrairement à un `visibility:hidden` ou `opacity:0`)
- Les éléments en `display:none` ne sont pas lus par les lecteurs d'écran
- Pour rendre l'élément à nouveau visible, il faut lui appliquer par exemple `display:block`

# Positionnement CSS

- La propriété `position:valeur` permet de positionner les éléments dans la page
- Les valeurs possibles sont : `static` (valeur par défaut), `relative`, `absolute` et `fixed`

# Position:static;

- static est la valeur par défaut de tous les éléments
- Un élément avec position: static; n'est positionné d'aucune manière spéciale.
- Un élément static est dit non positionné et un élément avec une propriété position ayant une valeur autre que static est dit positionné.

# Position:relative;

- Un élément positionné avec position:relative; se comporte par défaut de la même manière que static.
- On peut utiliser les propriété top, left, right et bottom pour déplacer l'élément
- Le reste du contenu ne sera pas affecté
- top, left right et bottom peuvent prendre des valeurs positives ou négatives (10px, -30px, etc.)

# Position: fixed

- Un élément en position: fixed; ne bougera pas, même si on fait défiler la page. Il sort du flux et se positionne au dessus des autres éléments.
- Il se positionne par rapport à l'espace affichable du navigateur

# Position: absolute;

- Un élément en position: absolute;
  - sort du flux (les autres éléments se réorganisent comme s'il n'était pas là)
  - vient se positionner par rapport à son dernier ancêtre positionné, à l'intérieur de celui-ci.
- Par défaut, si aucun ancêtre n'est positionné (et qu'on a modifié top, left, right ou bottom) il se positionne par rapport à l'élément racine <html>.
- Pour positionner le parent, on donne au parent une position: relative | fixed | absolute;

# Résumé

position: static



position: relative



position: absolute



Des questions ?

# Exercice

- Refaire la page : <http://preprod.up-esi.com/sitethomas/blog.html>. Ne pas essayez de refaire un menu similaire, juste une balise `nav` contenant les liens qui ne reste pas sur la page.
- Pour les icones, APPELEZ LUDIVINE pour utiliser font awesome : <https://fontawesome.com/how-to-use/on-the-web/setup/hosting-font-awesome-yourself>