

# HTML CSS ressource numéro 14

Sources :

<https://openclassrooms.com/>

L'ancien site 2minuteslearning.fr qui à malheureusement disparu

<https://www.vectorskin.com/balises-html5/>

## Positionnement en CSS

Avant de nous « lancer » dans l'aventure, un petit retour en arrière avec un rappel sur les différentes balises utilisées

### La balise HEADER

La plupart des sites web possèdent en général un en-tête, appelé header en anglais. On y trouve le plus souvent un logo, une bannière, le slogan de votre site...

### La balise FOOTER

À l'inverse de l'en-tête, le pied de page se trouve en général tout en bas du document. On y trouve des informations comme des liens de contact, le nom de l'auteur, les mentions légales, etc.

### La balise NAV

Généralement, dans cette section on va trouver des liens permettant d'accéder soit à d'autres pages du site, soit à différents contenus de la page.

### La balise SECTION

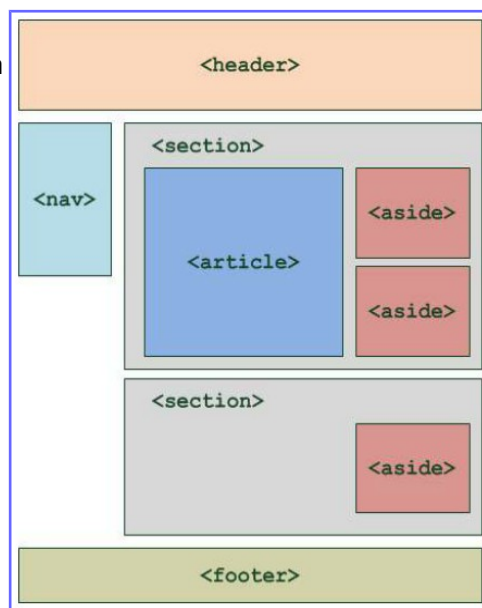
La balise <section> sert à regrouper des contenus en fonction de leur thématique. Elle englobe généralement une portion du contenu au centre de la page.

### La balise ARTICLE

ARTICLE fait partie de la liste des balises sémantiques html5. C'est une section (une zone) dans laquelle on peut mettre du texte, des images, de la vidéo, un formulaire de contact...etc.

### La balise ASIDE

La balise <aside> est conçue pour contenir des informations complémentaires au document que l'on visualise. Ces informations sont généralement placées sur le côté



Nous allons réaliser un exemple simple,

Code HTML

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <link rel="stylesheet" href="style.css" />
    <title>Le site web ecole</title>
  </head>
  <body>
    <header>
      <h1>Joachim</h1>
      <h2>Mes differents liens</h2>
    </header>
    <nav>
      <ul>
        <li><a href="#">Accueil</a></li>
        <li><a href="#">Blog</a></li>
        <li><a href="#">CV</a></li>
      </ul>
    </nav>
    <section>
      <aside>
        <h3>À propos de Joachim du bellay</h3>
        <p>Joachim du Bellay naît à Liré, en Anjou, dans l'actuel département de
Maine-et-Loire</p>
      </aside>
      <article>
        <h3>Je suis un grand écrivain</h3>
        <p>Heureux qui, comme Ulysse, a fait un beau voyage</p>
      </article>
    </section>
    <footer>
      <p>Copyright Joachim - Tous droits réservés
      <a href="#">Nous contacter au chateau de la Turmelière</a></p>
    </footer>
  </body>
</html>
```



Vous constatez sur l'image à droite que nous avons uniquement du HTML et pas encore de CSS.

Nous allons essayer de placer le menu à gauche et le reste du texte à droite. Pour cela, nous allons faire flotter le menu à gauche et laisser le reste du texte se placer à sa droite.

Nous voulons que le menu occupe 150 pixels de large. Nous allons aussi rajouter une bordure noire autour du menu et une bordure bleue autour du corps (à la balise <section>) pour bien les distinguer :

## Code CSS

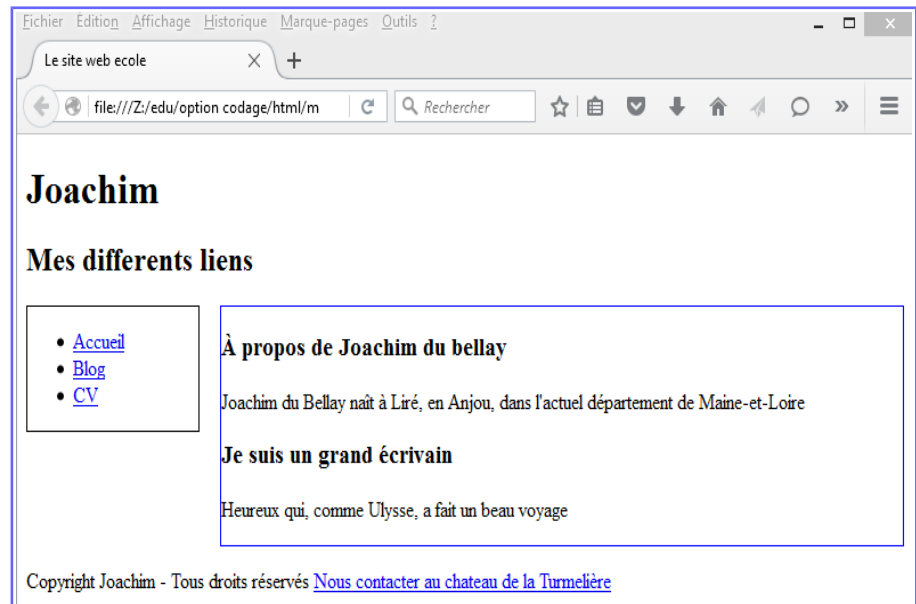
```
nav
{
  float: left;
  width: 150px;
  border: 1px solid black;
}
section
{
  border: 1px solid blue;
}
```



Cela commence à prendre forme mais ce n'est pas encore très « beau », nous allons donc opérer quelques changements

## Code CSS

```
nav
{
  float: left;
  width: 150px;
  border: 1px solid black;
}
section
{
  margin-left: 170px;
  border: 1px solid blue;
}
```



Nous allons maintenant découvrir une nouvelle fonctionnalité de CSS.

Il existe en CSS une propriété très puissante : **display**. Elle est capable de transformer n'importe quel élément de votre page d'un type vers un autre.

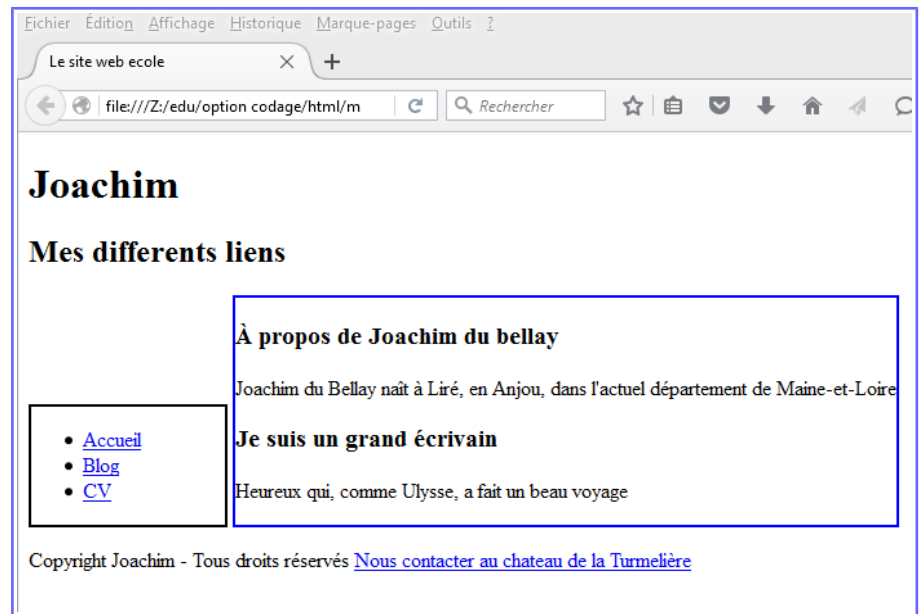
Voici quelques-unes des principales valeurs que peut prendre la propriété **display** en CSS

Valeur	Exemples	Description
inline	<code>&lt;a&gt;</code> , <code>&lt;em&gt;</code> , <code>&lt;span&gt;</code> ...	Eléments d'une ligne. Se placent les uns à côté des autres.
block	<code>&lt;p&gt;</code> , <code>&lt;div&gt;</code> , <code>&lt;section&gt;</code> ...	Eléments en forme de blocs. Se placent les uns en-dessous des autres et peuvent être redimensionnés.
inline-block	<code>&lt;select&gt;</code> , <code>&lt;input&gt;</code>	Eléments positionnés les uns à côté des autres (comme les inlines) mais qui peuvent être redimensionnés (comme les blocs).
none	<code>&lt;head&gt;</code>	Eléments non affichés.

Passons à l'exemple

Code : CSS

```
nav
{
  display: inline-block;
  width: 150px;
  border: 2px solid black;
}
section
{
  display: inline-block;
  border: 2px solid blue;
}
```



On souhaite maintenant remonter le bloc « nav » à la hauteur du bloc « section »  
Pour cela nous allons utiliser la propriété **vertical-align**. Cette propriété permet de modifier l'alignement vertical des éléments.

Voici quelques-unes des valeurs possibles pour cette propriété :

- baseline** : aligne de la base de l'élément avec celle de l'élément parent (par défaut) ;
- top** : aligne en haut ;
- middle** : centre verticalement ;
- bottom** : aligne en bas ;
- (valeur en px ou %)** : aligne à une certaine distance de la ligne de base

Code : CSS

```
nav
{
  display: inline-block;
  width: 150px;
  border: 2px solid black;
  vertical-align: top;
}
section
{
  display: inline-block;
  border: 2px solid blue;
  vertical-align: top;
}
```



## Le positionnement absolu

Le positionnement absolu permet de placer un élément (réellement) n'importe où sur la page

Code : CSS

```
element
{
  position: absolute;
}
```

Pour ce faire, on va utiliser quatre propriétés CSS  
left : position par rapport à la gauche de la page  
right : position par rapport à la droite de la page  
top : position par rapport au haut de la page  
bottom : position par rapport au bas de la page

## En résumé

La mise en page d'un site web s'effectue en CSS. Plusieurs techniques sont à notre disposition.

**Le positionnement flottant** (avec la propriété float) est l'un des plus utilisés à l'heure actuelle. Il permet par exemple de placer un menu à gauche ou à droite de la page. Néanmoins, cette propriété n'a pas été initialement conçue pour cela et il est préférable, si possible, d'éviter cette technique.

**Le positionnement inline-block** consiste à affecter un type inline-block à nos éléments grâce à la propriété display. Ils se comporteront comme des inlines (placement de gauche à droite) mais pourront être redimensionnés comme des blocs (avec width et height). Cette technique est à préférer au positionnement flottant.

**Le positionnement absolu** permet de placer un élément où l'on souhaite sur la page, au pixel près. Le positionnement fixe est identique au positionnement absolu mais l'élément restera toujours visible même si on descend plus bas dans la page.