

HTML-CSS

LES BASES DU CSS 3

V0

SOMMAIRE

Les bases du CSS 3	3
1/9 : Formatage du texte : taille, gras, italique, souligné, aligné.....	3
Syntaxe CSS.....	3
Ecrire du CSS dans un fichier .css séparé : CE QU'IL FAUT FAIRE !	4
2/9 : Ciblage.....	7
Objectif	7
Technique 1 : le chemin.....	7
Technique 2 : attribut id : pour cibler une balise et une seule	8
Technique 3 : attribut class : cibler quelques balises situées n'importe où dans le code	9
Technique 4 : balises de structuration : div et autres (header, footer, etc.) : pour cibler un bloc	10
Technique 5 : balises « en ligne » pour cibler des mots : span, strong, em, mark	11
Construction des sélecteurs	12
Cascading, spécialisation, héritage.....	14
3/9 : Formatage du texte : taille, gras, italique, souligné, aligné.....	16
taille du texte : font-size	16
gras, italique, souligné : font-style, font-weight, text-decoration	17
alignement : text-align : center, etc.	18
Rappel : formatage de texte avec des balises : à éviter	18
4/9 : Couleurs, fond, image de fond	19
Couleur du texte : color	19
Couleur de fond : background-color	19
background-image	19
Principes des couleurs	20
5/9 : Margin et padding.....	21
margin, padding, border	21
6/9 : Bordures.....	23
border : width style color	23
border-radius : bordures arrondies	25
7/9 : listes à points et a href customisés.....	26
list-style-type : none – text-decoration : none	26
8/9 : Apparences dynamiques	27
Principes	27
: hover -> pour changer du couleur quand on passe la souris	27
9/9 : Polices et fontes.....	28
Les différents types de police	28
font-family : pour définir une police pour un sélecteur	29
Charger une police avec Google Fonts	30
A : Début de mise en page : flex : mise en colonnes de certaines parties.....	31

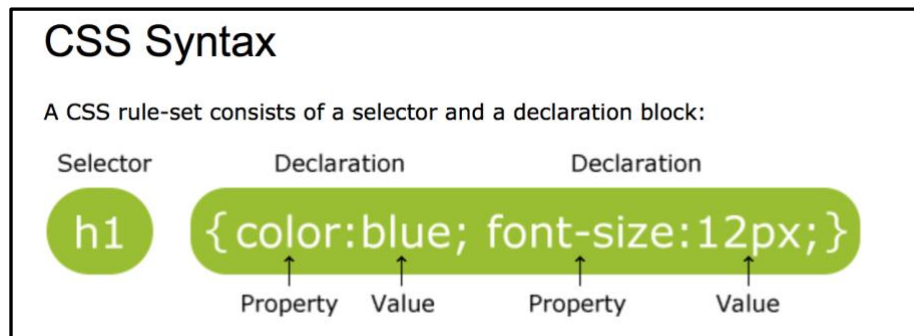
display : flex ; flex : 1;	31
Principes :	32
B : Début de mise en page : images flottantes	33
Image flottante : float, clear	33
TP	34

LES BASES DU CSS 3

1/9 : Formatage du texte : taille, gras, italique, souligné, aligné

Syntaxe CSS

Principes



https://www.w3schools.com/css/css_syntax.asp

```
selecteur {  
    propriété : valeurs ; /* commentaire */  
    propriété : valeurs ;  
    etc.  
}
```

Sélecteur

Le sélecteur de base c'est une balise.

Les commentaires sont entre /* et */

Exemples

```
p {  
    color : green ;  
}
```

```
h1, em {  
    color : blue ; /* red */  
}
```

On peut mettre plusieurs sélecteurs séparés par des virgules pour un même style.

Ecrire du CSS dans un fichier .css séparé : CE QU'IL FAUT FAIRE !

- Code HTML :

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

- Code CSS:

```
h1{
  background-color:yellow
}
```

- **Le bon usage :**

Plus c'est factorisé, mieux c'est. La bonne manière : le fichier CSS séparé.

Le reste est à éviter sauf cas particulier.

Le fichier CSS s'appelle style.css. On peut le mettre dans un dossier css, ou pas.

- **Facilités Sublime Text :**

On tape « link » et 2 tabulations : il écrit toute la ligne, sauf le nom du fichier CSS.

Exemple

Dans la balise <head> on ajoute une balise <link> avec un attribut « href » qui fait référence à un fichier .css. Il y a aussi un attribut « rel » pour préciser qu'il s'agit d'une feuille de style.

http://www.w3schools.com/tags/tag_link.asp

➤ *Code html*

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title>Style</title>
  <link rel="stylesheet" href="style.css">
</head>

<body>
  <h2> Bienvenue sur ma page stylée dans un fichier css</h2>
  <p> </p>
</body>
</html>
```

Syntaxe CSS

➤ *Code CSS*

```
body{
  background-color:aqua;
}
h2{
  background-color:rgb(255,255,0);
}
img{
  border:10px solid yellow;
  width:1000px;
}
```

Test à faire : utilisation pratique de Firefox

http://bliaudet.free.fr/Cours/HTML/21_CSS3_Bases/html_21_01_css_style.html

➤ ***alt-cmd-i pour voir le code***

On peut afficher le code CSS avec l'onglet Editeur de Style

On peut modifier les valeurs de la feuille de style : ici le width qui est à 1000 ! On peut le réduire.

A noter qu'on peut enregistrer la feuille de style qu'on est en train de modifier.

➤ ***cmd u dans l'inspecteur***

Pour afficher le code HTML dans une page. A noter que les liens sont cliquables.

A partir du fichier HTML, on peut ouvrir le fichier CSS.

2/9 : Ciblage

Objectif

Avec le `p{ }`, on cible toutes les balises `<p>`.

On veut pouvoir **cibler les balises individuellement** ou **par petits groupes** et non pas tous les « h1 » ou tous les « p ».

On a plusieurs techniques pour faire ça :

Technique 1 : le chemin

On peut préciser le chemin dans l'arbre des balises en listant les balises.

Exemple HTML

```
<body>
  <header>
    <h1>header</h1>
  </header>
  <main>
    <h1>main</h1>
    <ul>
      <li>li1</li>
      <li>li1</li>
    </ul>
  </main>
</body>
```

Exemple CSS

```
header h1{
  color : red;
}
main li{
  background-color: yellow;
}
```

header h1 : les h1 dans le header, mais pas ceux dans le main.

Main li : les li dans le main.

Technique 2 : attribut id : pour cibler une balise et une seule

Principe

Pour le moment, on cible tous les paragraphes, p, ou tous les h1.

On veut par exemple **pouvoir ne cibler qu'un et un seul paragraphe.**

Syntaxe : attribut id

```
<p id="introduction">
```

On ajoute l'attribut « id » avec une valeur au choix dans l'unique balise qui nous intéresse.

Avec l'attribut « id », il ne peut y avoir qu'une seule balise avec la même valeur pour l'id.

Exemple HTML

```
<p id="introduction">  
  Mon texte  
</p>
```

Exemple CSS

#introduction est un sélecteur qui concerne les balises avec un attribut id="introduction"

```
#introduction {  
  color : green ;  
}
```

#introduction : sélecteur qu'on met à la place d'un nom de balise.

Le # veut dire qu'on fait référence à un attribut id dans le HTML : il ne peut donc y avoir qu'une seule balise concernée. En fait, avec plusieurs ça marche aussi mais ça posera des problèmes avec les langages serveurs ou le JavaScript.

Présentation W3schools

http://www.w3schools.com/cssref/sel_id.asp

Technique 3 : attribut class : cibler quelques balises situées n'importe où dans le code

Principe

Pour le moment, on cible tous les paragraphes, p, ou tous les h1, etc.

On veut par exemple **pouvoir ne cibler que certains paragraphes.**

Syntaxe : attribut class

```
<p class="introduction">
```

On ajoute l'attribut « class » avec une valeur au choix dans les balises qui nous intéressent.

Avec l'attribut « class », on peut avoir plusieurs balises avec la même valeur.

Exemple HTML

On a deux balises avec le même attribut class :

```
<p class="introduction">
    Mon texte
</p>
<p class="introduction">
    Mon autre texte
</p>
```

Exemple CSS

.introduction est un sélecteur qui concerne les balises avec un attribut class="introduction"

```
.introduction {
    color : green ;
}
```

W3schools

http://www.w3schools.com/cssref/sel_class.asp

http://www.w3schools.com/tags/att_global_class.asp

Technique 4 : balises de structuration : div et autres (header, footer, etc.) : pour cibler un bloc

Présentation

La **balise div** est une balise dite « universelle » (comme la balise span, cf. § suivant).

C'est une **balise « bloc »**. Elle est de la **famille des balises** <p>, <h1>, etc.

La balise <div> permet de **regrouper plusieurs balises** et ensuite de le cibler en CSS.

On commence par un <div> et on finit par un </div>

http://www.w3schools.com/tags/ref_byfunc.asp

Les robots d'analyse ne font rien de la balise div.

Exemple HTML

On crée un div pour regrouper 3 balises <p> :

```
<div id= "encadrement">
  <p> bla bla bla</p>
  <p> bla bla bla</p>
  <p></p>
</div>
```

Exemple CSS

#encadrement permet de cibler les 3 paragraphes.

```
#encadrement {
  background-color: yellow;
}
```

W3schools

http://www.w3schools.com/tags/tag_div.asp

Technique 5 : balises « en ligne » pour cibler des mots : span, strong, em, mark

Présentation

La **balise span** est une balise dite « universelle » (comme la balise div, cf. § précédent).

C'est une **balise « inline »**. Elle est de la **famille des balises** , et <mark> mais elle n'a pas d'effet par défaut.

Elle ne crée **pas de nouveau bloc** ni de retour à la ligne.

La balise span est **utilisée pour cibler un ou plusieurs mots** dans un texte.

http://www.w3schools.com/tags/ref_byfunc.asp

Les robots d'analyse ne font rien de la balise span.

Exemple HTML

On veut marquer les mots « super site » dans le texte :

```
<p> Bienvenue dans mon <span class="super"> super site</span> de  
super trucs! </p>
```

Exemple CSS

.super permet de cibler notre span et toutes les class « super ».

```
.super {  
    color : green ;  
}
```

W3schools

http://www.w3schools.com/tags/tag_span.asp

Différences entre span, strong, emp et mark

strong sert à marquer des mots très importants (forts).

em sert à montrer une emphase

mark sert à marquer des mots.

La distinction entre les 3 n'est pas très claire et assez subjective. C'est pourquoi ces balises sont finalement peu utilisées : on continue à utiliser le span qui nous permettra de faire ce qu'on veut sans comportement par défaut.

Construction des sélecteurs

Usages de base

- p { /* ou n'importe quelle balise */
- h1, em { /* plusieurs balises à la fois */
- .class { /* pour cibler plusieurs balises */
- #id { /* pour cibler une balise en particulier */
- * { /* Sélecteur universel : tout ! Rare ! */

Usages avec plusieurs sélecteurs

- h2 strong { /* Une balise dans une autre : les strong qui se trouvent dans des h2 */
- div > p { /* les p dont le parent direct est une div */
- ul .intro { /* Espace entre les 2. Les classes intro dans un ul (des li par exemple) */
- h1.intro { /* Collé. Les h1 avec une classe intro */
- h2 + p { /* Une balise qui en suit une autre : les p qui suivent h2 */

Autres usages

Il faut s'y intéresser pour se faciliter la tâche dans la mise en page.

Par exemple :

a[title] : récupère les balises <a> avec un attribut title.

p::first-letter : récupère la première lettre.

p::first-line : récupère la première ligne, quelle que soit la largeur de l'écran.

p:first-child : récupère le premier enfant.

li:last-child : récupère le dernier enfant.

::after : permet d'ajouter du code HTML après un élément.

Etc.

Tous :

http://www.w3schools.com/cssref/css_selectors.asp

<http://www.w3.org/Style/css3-selectors-updates/WD-css3-selectors-20010126.fr.html>

Cascading, spécialisation, héritage

Principes

- **Cascading (c'est-à-dire héritage)** : si j'applique un style à body (ou un div par exemple), ça vaut pour tout ce qui est contenu dans body. Par exemple le color ou le font-size valent pour les balises dans le body. Les balises contenues dans le body « héritent » du style appliqué au body.
- **Spécialisation** : si j'applique un nouveau style à une balise dans le body, c'est ce nouveau style qui prime. La balise a alors un style « spécialisé ».
- **Bloquer l'application d'un style : inherit**. Si un selecteur a un style, par exemple p{color:green;}, on peut bloquer l'application du style en écrivant <p style="color:inherit;">. Dans ce cas, si le style est défini dans une balise qui englobe p, par exemple le body, le style du body s'appliquera.

Test

http://bliaudet.free.fr/Cours/HTML/21_CSS3_Bases/html_21_04_heritage_et_specialisation.html

Code du test

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>test</title>
  <style type="text/css">
    body{
      color : blue;
    }
    header h1{
      color : red;
    }
    header h1::first-letter{
      color : blue;
    }
    main li{
      background-color: yellow;
    }
    main ul li:last-child{ /* le dernier li dans main ul
      color: red;
    }
  </style>
</head>

<body>
  <header>
    <h1>header</h1>
  </header>
  <main>
    <h1>main</h1>
    <ul>
```

```
        <li>li1</li>
        <li>li2</li>
        <li>li3</li>
    </ul>
</main>
</body>
</html>
```

3/9 : Formatage du texte : taille, gras, italique, souligné, aligné

taille du texte : font-size

{ font-size : 14px ; } px ce sont des pixels. C'est absolu : ne pas en abuser.

{ font-size : medium ; } small, large, x-large, xx-small. C'est relatif. Donc c'est mieux.

{ font-size : 1em ; } 2.3em, 0.8 em. C'est relatif et lisible. Donc c'est mieux.

{ font-size : 80% ; } C'est relatif mais vite lourd.

1em <=> 100% <=> medium

Le bon usage : les « em » : c'est une taille relative qu'on peut faire varier.

http://www.w3schools.com/cssref/pr_font_font-size.asp

<http://www.w3schools.com/cssref/default.asp#font>

gras, italique, souligné : font-style, font-weight, text-decoration

{ font-style : normal ; }

{ font-style : italic ; }

http://www.w3schools.com/cssref/pr_font_font-style.asp

{ font-weight: bold; } /* ou un nombre de 100 à 900. 400= normal, 700= bold */

http://www.w3schools.com/cssref/pr_font_weight.asp

{ text-decoration : underline ; } /* ou bien line-through (barré); overline (ligne au dessus) ; */

<http://www.w3schools.com/cssref/default.asp#textdecor>

http://www.w3schools.com/cssref/pr_text_text-decoration.asp

alignement : text-align : center, etc.

{ text-align : center ; } /* ou bien left ; right ; justify ;

http://www.w3schools.com/cssref/pr_text_text-align.asp

<http://www.w3schools.com/cssref/default.asp#text>

Rappel : formatage de texte avec des balises : à éviter

On peut faire certaines choses simples avec des balises : mettre en gras, italique, etc.

 passage à la ligne,

<hr> tirer un trait,

 gras,

<i> italique,

<ins> souligné,

<sub> bas,

<sup> haut,

<small> petit

Dans des pages graphiques, il faut éviter cette technique qui ne permet pas de généraliser (définir un style général pour plusieurs balises) et rend les modifications longues et complexes à réaliser.

Pour des pages simples, on peut les utiliser.

https://www.w3schools.com/html/html_formatting.asp

4/9 : Couleurs, fond, image de fond

Couleur du texte : color

La couleur est une propriété qui s'applique à tout le document (selecteur * ou balise body) ou à n'importe quel balise (un titre par exemple).

```
p {  
    color : blue ;  
}
```

ou encore

```
body {  
    color : #FFC305;  
}
```

L'attribut s'applique à toutes les balises incluses, donc à toute la page sauf spécialisation.

Couleur de fond : background-color

La couleur de fond est une propriété qui s'applique à tout le document (selecteur * ou balise body) ou à n'importe quel balise (un titre par exemple).

```
h1 {  
    background-color : turquoise ;  
}
```

http://www.w3schools.com/cssref/pr_background-color.asp

background-image

L'image de fond est une propriété qui s'applique à tout le document (balise body) ou à n'importe quel balise (un titre par exemple).

Quelle que soit la taille de l'image, l'image est répétée dans toute la balise, dans tout le document pour une balise body.

```
body {  
    background-image : url("neige.png") ;  
    background-color : whitesmoke;  
}
```

Si le fichier neige.png n'existe pas la couleur whitesmoke sera appliquée.

➤ Exemple et application

http://www.w3schools.com/cssref/pr_background-image.asp

Dans l'exemple ci-dessus, ajoutez un f à gif : giff pour que le fichier ne soit pas trouvé. Regardez le résultat.

Principes des couleurs

<http://www.w3schools.com/colors/>

Principes : le RGB - rgb(255, 195, 23)

Chaque couleur est un mélange de 3 couleurs de base : rouge, de vert et de bleu (RGB : Red, Green, Blue).

La quantité de chaque couleur de base peut valoir de 0 à 255 : 256 valeurs possibles car c'est codé sur 1 octet (1 octet = 8 bits : soit 2 puissance 8 possibilités).

Il y a donc 16 millions de couleurs : 256 au cube.

On peut écrire les couleurs avec la valeur de chaque couleur codé de 0 à 255.

Exemple : rgb(255,0,0) c'est 255 quantité de rouge (red), 0 quantité de vert (green), 0 quantité de bleu (blue), donc c'est du rouge.

https://www.w3schools.com/colors/colors_rgb.asp

Couleurs en Hexa : #FFC305

On peut écrire les couleurs avec la valeur de chaque octet écrit en hexadécimal.

Exemple: #FF0000 c'est FF quantité de rouge (255), 0 quantité de vert (green), 0 quantité de bleu (blue), donc c'est du rouge.

http://www.w3schools.com/colors/colors_hexadecimal.asp

Couleurs nommées : yellow, aqua, pink, etc.

140 couleurs nommées : white, black, red, marron, yellow, green, blue, aqua, purple, etc.

Dégradé : <http://www.leteileauxsecrets.fr/couleurs/couleurs-web.html>

Autres listes :

http://www.w3schools.com/colors/colors_hex.asp

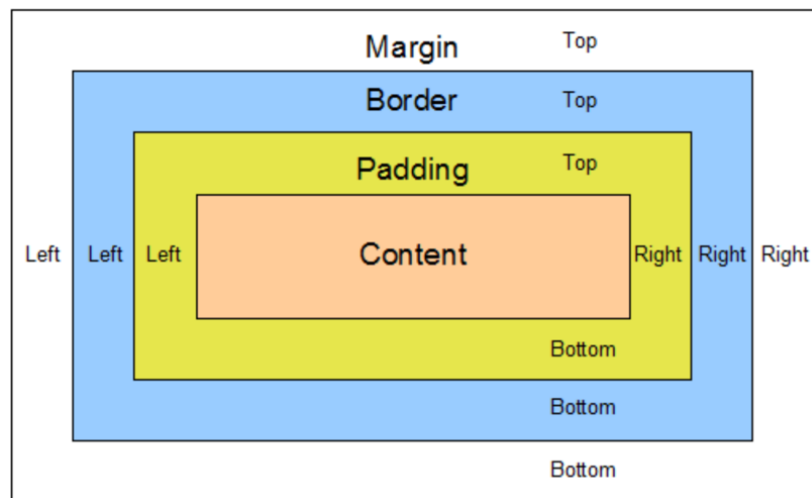
<http://stylescss.free.fr/couleurs.php>

5/9 : Margin et padding

margin, padding, border

Principes

- **padding** : marges à l'intérieur des bordures.
- **border** : les bordures
- **margin** : marges à l'extérieur des bordures.



Valeurs

- pixels, em, pourcentage.
- **margin : auto** ; les marges sont définies automatiquement.

Width et margin auto : centrer

- width et margin : auto : ça centre le bloc.

Usage détaillée :

padding-bottom, -top, -left, -right

margin-bottom, -top, -left, -right

margin : auto ; /* marge extérieure automatique en fonction du contexte */

margin : 3px 0px 1px 4px ; /* top right bottom left : sens des aiguilles d'une montre */

margin : 3px 0px ; /* haut-bas et droite-gauche*/

http://www.w3schools.com/css/css_padding.asp

http://www.w3schools.com/css/css_margin.asp

6/9 : Bordures

border : width style color

border : width style color

L'attribut border permet de définir une bordure : son **épaisseur**, son **style** (trait plein, pointillé, etc.) et sa **couleur**

```
h1 {  
    border : 2px solid black; /* ca fait un encadrement */  
}
```

http://www.w3schools.com/cssref/pr_border.asp

- **Format du width** : px, pt, mm, cm, em, thin, medium, thick.
- **Format du style** : none (pas de bordure), solid, dotted (pointillés), dashed (tirets), double (bordure double), groove (relief 1), ridge (relief 2), inset (3D global enfoncé), outset (3D global surélevé).

http://www.w3schools.com/css/css_border.asp

Contrôles individualisés

border-width

border-style

border-color

border-left, -top, -right, -down : contrôle par côté

border-left

border-left-color

border-bottom

etc.

https://www.w3schools.com/cssref/pr_border-left.asp

border-radius : arrondis des bordures

http://www.w3schools.com/cssref/css3_pr_border-radius.asp

border-radius: 10px; pour avoir **un arrondi, plus ou moins marqué**

border-radius: 10px 5 px, 15 px 20 px ; pour les quatre coins (haut gauche, haut droit, bas droit, bas gauche)

border-radius : 20px / 10 px ; pour un effet elliptique.

7/9 : listes à points et a href customisés

list-style-type : none – text-decoration : none

On peut choisir ce qu'on affiche pour une liste à points :

http://www.w3schools.com/cssref/tryit.asp?filename=trycss_list-style-type_ex

```
li {  
    list-style-type : square ;  
}
```

⇒ affiche un carré plutôt qu'on point. On peut aussi mettre : upper-alpha ; lower-alpha ; lower-roman ; upper-roman ; etc.

https://www.w3schools.com/cssref/pr_list-style-type.asp

```
li {  
    list-style-type : square ;  
}
```

⇒ n'affiche rien : supprime le point

On peut choisir de ne pas souligner un a href :

```
a {  
    text-decoration : none ;  
}
```

⇒ n'affiche rien : supprime le point

8/9 : Apparences dynamiques

Principes

On peut changer le comportement du site sur le passage de la souris et quand on clique. :

:hover -> pour changer du couleur quand on passe la souris

➤ *Dans CSS*

```
p:hover {  
    background_color: pink;  
}
```

le paragraphe passe sur fond rose quand on passe la souris dessus.

Les différents types de police



Sans-serif



Serif

https://www.w3schools.com/css/css_font.asp

<https://www.w3.org/Style/Examples/007/fonts.fr.html>

- **sans-serif** (pas de petites pattes),
- **avec serif** (avec petites pattes),
- **curvive** (écriture manuelle),
- **monospace** (écartement constant),
- **fantasy**

font-family : pour définir une police pour un sélecteur

Principes

On peut définir une ou plusieurs polices pour chaque sélecteur.

font-family

http://www.w3schools.com/cssref/pr_font_font-family.asp

- L'attribut font-family permet de définir une police pour un sélecteur.

```
p {  
    font-family: "Times New Roman", Georgia, Serif;  
}
```

- En général, on met une série de polices : si la première ne marche pas, on prend la suivante. On finit en général par la police générique : ici Serif (avec petites pattes).

D'abord « Times New Roman ». On met des guillemets si le nom comporte un espace. Ensuite Georgia. Sinon Serif.

- Version sans serif :

```
p {  
    font-family: Arial, Helvetica, sans-serif;  
}
```

- La liste des polices correspond aux polices qu'on trouve classiquement sur les navigateurs.

Charger une police avec Google Fonts

➤ *Principes*

Google Fonts propose des polices : <https://fonts.google.com/>

Choisir une catégorie : on peut sélectionner Serif, Sans Serif, Display, Handwriting et Monospace.

On voit le nom des polices, par exemple « Indie Flower » si on sélectionne Handwriting. Il suffit de mettre ce nom dans le code suivant :

➤ *Dans HTML*

En plus du <link> du CSS, on ajoute le <link> suivant dans le head :

```
<link href="https://fonts.googleapis.com/css?family=Indie+Flower"
rel="stylesheet" type="text/css">
```

➤ *Dans CSS*

```
font-family: 'Indie Flower', 'cursive';
```

Le cursive pour avoir la police cursive par défaut si le « Indie Flower » ne marche pas.

➤ *Bilan*

C'est très simple ! Mais ça nous rend dépendant de l'API google.

➤ *test complet très moche !*

http://bliaudet.free.fr/Cours/HTML/21_CSS3_Bases/html_21_05_exemple_complet.html

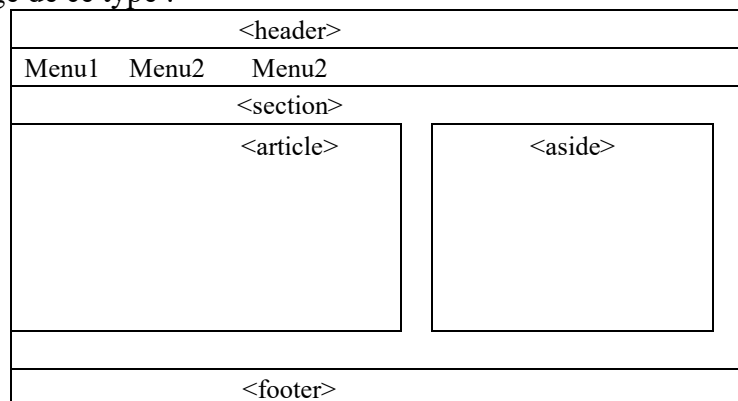
A : Début de mise en page : flex : mise en colonnes de certaines parties

display : flex ; flex : 1;

Exemple

http://b্লাuidet.free.fr/Cours/HTML/31_CSS_structure/01_flex/index.html

On veut une page de ce type :



Pour article et aside

On met uniquement l'aside et l'article en colonne : il suffit d'un display flex dans la section.

Avec « flex : 3 ; » et « flex : 1 ; » on précise le rapport de taille entre article et aside

```
section{
    display: flex; /* aside et article concernés */
}
article{
    flex: 3;
}
aside{
    flex: 1;
}
```

Pour le menu

On met un display flex dans l'ul (nav ul pour cibler).

Ensuite, on fixe la largeur de chaque menu et on supprime le point et le souligné des menus.

```
nav ul{
    display: flex; /* les li sont concernés */
}
li{
    width: 15%;
    list-style-type: none; /* pour éviter le point */
}
a{
    text-decoration: none; /* pour éviter le souligné */
}
```

Principes :

On a un bloc, par exemple une `<div>` ou un `<main>` qui contient trois sous-blocs, par exemple 2 `<article>` et un `<aside>`.

Par défaut, les sous blocs sont les uns en dessous des autres.

Pour les présenter en colonnes, c'est très simple : il suffit de mettre un :

display : flex ; sur le conteneur

flex : 1 ; sur les sous-blocs. Le 1 définit une taille relative des sous-blocs les uns par rapport aux autres.

Pour les sous-blocs, on peut aussi fixer leurs tailles.

B : Début de mise en page : images flottantes

Image flottante : float, clear

Principes

Une image de base se trouve sur une seule ligne. Le texte avant et après une image est sur une seule ligne, avant et après l'image.

Si le texte qui suit l'image est dans un paragraphe, il est sur une nouvelle ligne.

The float Property

In this example, the image will float to the right in the text, and the text in the paragraph will wrap around the image.

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus imperdiet, nulla et dictum interdum, nisi lorem egestas odio, vitae scelerisque enim ligula venenatis dolor. Maecenas nisl est, ultrices nec congue eget, auctor vitae massa. Fusce luctus vestibulum augue ut aliquet. Mauris ante ligula, facilisis sed ornare eu, lobortis in odio. Praesent convallis urna a lacus interdum ut hendrerit risus congue. Nunc sagittis dictum nisi, sed ullamcorper ipsum dignissim ac. In at libero sed nunc venenatis imperdiet sed ornare turpis. Donec vitae dui eget tellus gravida venenatis. Integer fringilla congue eros non fermentum. Sed dapibus pulvinar nibh tempor porta. Cras ac leo purus. Mauris quis diam velit.



Texte autour d'une image

➤ *Propriété float*

Pour avoir du texte autour de l'image, on parle d'**image flottante** et on utilise la propriété « float », right ou left sur l'image.

http://www.w3schools.com/cssref/pr_class_float.asp

Exercice

Dans cet exemple, faites-en sorte d'avoir deux paragraphes autour de l'image, le second entourant l'image.

➤ *Propriété clear*

Pour reprendre le texte d'un nouveau paragraphe après l'image, on utilise la propriété « clear », right ou left selon le float de départ, ou both pour n'importe quel float.

http://www.w3schools.com/cssref/pr_class_clear.asp

Remarque : Pour décaler tout le paragraphe du texte à droite de l'image, il faudra gérer la mise en page.

Exercice

Reprenez l'exercice précédent (sortez l'image du paragraphe. Faites-en sorte d'avoir deux paragraphes autour de l'image, le second entourant l'image). Ajoutez un clear pour que le second paragraphe n'entoure plus l'image.

TP

0 – W3-School

Reprendre l'exercice W3 school et mettez-y du CSS comme sur le site.

Il faut coder la page ci-dessous qui se trouve ici : https://www.w3schools.com/css/css_intro.asp

(La bordure grise n'est pas à coder, par contre on doit avoir la marge blanche au bord de la page.

Welcome to My Homepage

Use the menu to select different Stylesheets

Stylesheet 1

Stylesheet 2

Stylesheet 3

Stylesheet 4

No Stylesheet

Same Page Different Stylesheets

This is a demonstration of how different stylesheets can change the layout of your HTML page. You can change the layout of this page by selecting different stylesheets in the menu, or by selecting one of the following links:
[Stylesheet1](#), [Stylesheet2](#), [Stylesheet3](#), [Stylesheet4](#).

No Styles

This page uses DIV elements to group different sections of the HTML page. Click here to see how the page looks like with no stylesheet:
[No Stylesheet](#).

Side-Bar

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat. Ut wisi enim ad minim veniam, quis nostrud exerci tation ullamcorper suscipit lobortis nisl ut aliquip ex ea commodo consequat. Duis autem vel eum iriure dolor in hendrerit in vulputate velit esse molestie consequat, vel illum dolore eu feugiat nulla facilisis at vero eros et accumsan et iusto odio dignissim qui blandit praesent luptatum zzril delenit augue duiis dolore te feugait nulla facilisi.

1 – Merveilles-CSS-1

Reprenez le TP 5 sur les merveilles (la dernière version, tableau et vidéo).

Mettez le tableau avant les références.

➤ *Ajouter du graphisme :*

- Mettez des couleurs de fonds différents pour la partie de début avec le titre et le sommaire, pour chaque partie avec les types de merveilles, pour le tableau (uniquement la surface du tableau) et pour les références. Vous mettrez des couleurs très claires, type « pastel ». Foncez un peu les titres correspondant (on donnera aussi une couleur pour le titre des vidéos). On laisse les vidéos en blanc. On peut donner les mêmes couleurs pour la partie du début, celle des références et celle du tableau.
- Encadrez la zone du titre et la zone des références.
- Changez la typo des nouvelles merveilles.
- Faites-en sorte que les merveilles changent de couleur quand on passe la souris dessus.
- Faites un tableau avec une alternance de couleur de fond. Mettez un cadre à simple ligne autour de chaque case du tableau.
- Remplacez les points de la première liste à points de merveilles par un image au choix (par exemple, une étoile).
- Ajoutez un effet de relief ou d'ombres quand on passe sur un élément du sommaire.
- Dans les pages de détails, limitez la taille de l'image à 50% de l'écran. Placez l'image à droite et entourer l'image par du texte.

Couleurs nommées : <http://www.lettoileauxsecrets.fr/couleurs/couleurs-web.html>

➤ **Exemple de solution**

Les sept merveilles du monde

Connaissez-vous les merveilles du monde ?

Sommaire

- [Les 7 merveilles du monde antique](#)
- [Les 7 nouvelles merveilles du monde](#)
- [tableau des merveilles](#)
- [vidéos des merveilles](#)
- [références](#)

Les 7 merveilles du monde antique

Cette liste nous vient de l'antiquité.

★ La pyramide de Khéops



★ Les jardins suspendus de Babylone



★ La statue de Zéus



★ Le temple d'Artémis



★ Le mausolée d'Halicarnasse



★ Le Colosse de Rhodes



★ Le phare d'Alexandrie



[Retour au sommaire](#)

Nouvelles merveilles du monde

Il existe des centaines de merveilles référencées au **patrimoine mondial** par l'Unesco.

Une liste de 7 nouvelles merveilles a été établie en 2009 à la suite d'un vote par Internet sur une initiative privée.

1. La Grande Muraille de Chine



2. Pétra



• • •



5. La ville de Chichén Itzá



6. Le Colisée



7. Le Taj Mahal

[Retour au sommaire](#)

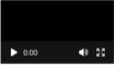
Tableaux des Merveilles

Liste des Merveilles	Pays	Siècle	Drapeau
La pyramide de Khéops	Égypte	2560 av. J.-C.	
Les jardins suspendus de Babylone	Irak	600 av. J.-C.	
La statue de Zeus	Grèce	432 av. J.-C.	
Le Colosse de Rhodes		292 av. J.-C.	
Le temple d'Artémis		560 av. J.-C.	
Le mausolée d'Halicarnasse	Turquie	350 av. J.-C.	
Le phare d'Alexandrie		299 av. J.-C.	
La Grande Muraille de Chine	Chine	800 av. J.-C.	
Pétra	Jordanie	800 av. J.-C.	
Le Christ du Corcovado	Brésil	1926 ap. J.-C.	
Machu Picchu	Pérou	1500 ap. J.-C.	
Chichén Itzá	México	900 ap. J.-C.	
Le Colisée	Italie	70 ap. J.-C.	
Le Taj Mahal	Inde	1631 ap. J.-C.	

[Retour au sommaire](#)

Vidéos

Vidéo mp4 : la grande muraille de Chine



Vidéo youtube : les Merveilles du monde



[Retour au sommaire](#)

Références

- Merveilles antiques : [Wikipedia - Universalis](#)
- Nouvelles merveilles : [Wikipedia - Unesco](#)

2 - Le CV-2

Reprendre le CV sans CSS et y ajouter du CSS.

Ne pas faire de choses trop compliquées dans la structuration de la page : ça viendra avec le chapitre sur la mise en page.

Toutefois, vous pouvez essayer les éléments suivants :

- Placer la photo en haut à droite avec un float.
- Changer la couleur de certains textes
- Changer les alignements de certains textes
- Changer la couleur de fond des différentes parties.
- Appliquer une image de fond sur une partie.
- Utiliser une police personnalisée via @font-face
- Utiliser une police personnalisée via Google Fonts
- Définir une bordure sur une partie du CV.
- Définir une ombre à un élément
- Mettre des apparences dynamiques

On peut changer le fichier HTML si c'est nécessaire.