

Les feuilles de style CSS3

(cascading style sheets)

1) QU'EST-CE QUE C'EST CSS ?

Le CSS est un langage informatique (coté client¹) utilisé pour gérer la présentation du contenu d'un document HTML (la mise en forme de tous ces éléments HTML); il est recommandé par le l'organisation W3C (World Wide Web Consortium²).

Le but du CSS est de séparer la structure d'un document HTML de sa présentation; le langage HTML sert à structurer le contenu du document HTML, et CSS sert à mettre en forme plus facilement ce contenu structuré.

2) Déclaration d'un style CSS3 :

Les styles peuvent être incorporés au sein du document HTML de trois manières différentes :

- **Style en ligne**, c'est-à-dire en tant qu'attribut des balises

Dans ce cas l'attribut style déclarée dans la balise ouvrante permet de mettre en forme l'élément HTML en question (dans lequel l'attribut style est déclaré) ; ce style ne concerne que cet élément HTML

Exemples :

`<p style= "color :red "> j'ai une couleur rouge</p>(p1)`

`<p style= "background :blue" > j'ai un remplissage blue
</p>(p2)`

- **Style du document**: déclaré dans l'en-tête, c'est-à-dire au sein des balises `<HEAD>` et `</HEAD>`

¹ Un langage coté client est un langage qui s'exécute au niveau du poste client ; à la demande de ce dernier le serveur envoie une page web sous la forme d'un document HTML qui s'affiche sur le poste client ; au contraire d'un langage coté serveur qui est un langage qui s'exécute au niveau du serveur et ce dernier envoie la réponse sous la forme d'un document HTML.

² C'est un organisme international qui développent les standards (les plateformes web) pour le web pour assurer une communication facile entre les utilisateurs du Web.

Leur site web validator.w3.org à pour but la validation des documents HTML pour les rendre conformes à ces normes (voir notre cours HTML5).

Dans la balise <head> on ajoute la balise <style></style> dans laquelle on déclare les styles des éléments HTML de la page.

Exemple :

```
<head>
  <title>CSS3</title>
  <style>
    p
    {
      color:red;
    }

  </style>
</head>

<body>

<p> j'ai une couleur rouge</p>
  <p> j'ai une couleur rouge</p>

</body>
```

Le code HTML et CSS :

```
<head>
  <title>CSS3</title>
  <style>
    p
    {
      color:red;
    }

  </style>
</head>

<body>

<p style= »color :blue > j'ai une couleur bleu</p>
  <p> j'ai une couleur rouge</p>

</body>
```

Résultat :

j'ai une couleur rouge
j'ai une couleur rouge

- Le style déclaré dans la partie <head> concerne tous les éléments HTML en question.
- Le style en ligne a la priorité qu'un style déclaré dans la partie <head>.

Exemple :

j'ai une couleur rouge
j'ai une couleur rouge
j'ai une couleur blue

Le code HTML et CSS :

```
<head>  
  <title>CSS3</title>  
  <style>  
    p  
    {  
      color:red;  
    }  
  </style>  
</head>  
<body>  
  <p> j'ai une couleur rouge</p>  
  <p> j'ai une couleur rouge</p>  
  <p style="color:blue">j'ai une couleur blue</p>  
</body>
```

- **Style externe**, déclaré dans un fichier à part dont l'extension est **.css**

Un fichier de style est un fichier d'extension *.css, il peut contenir des styles qui seront appliqués sur un document HTML ou plusieurs documents.

C'est la méthode la plus pratique et la plus souple.

Un style en ligne a la priorité par rapport au style du document et ceci a la priorité par rapport aux styles déclarés dans un fichier de style.

Exemple :

Soit le fichier styles.css qui contient les styles suivants :

```
p
{
    color:red;
}
body
{
    background: #17CCAB;
}
```

Pour relier ce fichier avec le document index.html, on ajoute le lien suivant :

<link rel="stylesheet" href="styles/styles.css"/>

Ou

@IMPORT URL("styles/styles.css");

La balise **<link />** permet au navigateur de retrouver l'emplacement du fichier de style en utilisant l'attribut **href**.

L'attribut **rel** : indique que le type du fichier est une feuille de style CSS.

Remarque :

Les styles d'un fichier de style peut être appliqué à plusieurs document HTML, en revanche les style de document ou en ligne.

- **Style importé**, déclaré dans un fichier à part dont l'extension est **.css**

Les recommandations du W3C offrent une dernière façon d'inclure des feuilles de style dans un document: en important des feuilles de style. Il est en effet possible d'importer des feuilles de style externes au niveau de la déclaration du style de document, en insérant la commande *@IMPORT* immédiatement après la balise style :

```
<style>

    @IMPORT URL(url de la feuille à importer);

</style>
```

Remarque :

Il est possible de définir plusieurs styles en utilisant les différents moyens qu'offrent les CSS. Ainsi, lorsque plusieurs feuilles de style externes sont appelées, on obtient ce que l'on appelle une cascade de styles, c'est-à-dire une combinaison de styles pour divers éléments HTML. Si plusieurs styles concernent le même élément, seul le dernier style sera conservé.

```
<LINK rel=stylesheet href="style1.css">
<LINK rel=stylesheet href="style2.css">
<LINK rel=stylesheet href="style3.css">
```

3) Les sélecteurs CSS3 ?

Définition :

Un **sélecteur** CSS est code Css de base qui permet de sélectionner un ou plusieurs éléments HTML sur lesquels s'applique un style Css.

```
sélecteur
{
    Propriété1: valeur1;
    Propriété2: valeur2;
    Propriété3: valeur3;
    Propriété4: valeur4;
}
```

Les sélecteurs simples.

Le sélecteur universel.

Ce sélecteur permet de cibler tous les éléments HTML du document HTML.

```
* {  
  Margin :0px ;  
}
```

Pour supprimer les marges de tous les éléments HTML

Les sélecteurs de type

Ce sont les sélecteurs qui permettent de cibler un ou plusieurs éléments HTML par le nom de la balise (tag).

Exemple :

```
h1 {  
  font-size: 200%;  
  color: navy;  
  text-align: center;  
}  
h2 {  
  font-size: 150%;  
  color: red;  
  padding-left: 15px;  
}
```

Remarque :

Les styles Css3 s'exécutent ligne par ligne pour une balise, et le dernier qui sera appliqué si plusieurs styles concernent cette balise.

Exemple :

```
<head>  
<title>Hello!</title>  
<style>  
p  
{  
  color: red;  
  font-size :18px ;  
}  
p  
{  
  color:blue;  
}  
  
</style>  
</head>
```

```

<body>
  <p>Je suis le premier paragraphe</p>
  <p>Je suis le deuxième paragraphe</p>
  <p>Je suis le troisième paragraphe</p>
  <h1>Je suis le titre de niveau 1</h1>
  <h2>Je suis le titre de niveau 2</h2>
  <h3>Je suis le titre de niveau 3</h3>

</body>

```

La couleur **bleu** qui sera appliquée sur les paragraphes.

NB : **N'oublier pas la remarque de priorité des styles**

Remarque : On peut regrouper des sélecteurs de même contenu pour minimiser le code.

```

p, h1, h2
{
  color:blue;
}

```

Exemple d'application :

Soit le code Html suivant :

```

<body>
  <div>
    <p>paragraphe 1<span> div1</span></p>
    <p>paragraphe 2<span> div2</span></p>
    <p>paragraphe 3<span> div3</span></p>
    <span> div4</span>
  </div>
  <p>paragraphe 4</p>
  <p>paragraphe 5</p>
  <p>paragraphe 6</p>
</body>

```

- 1) Changer le remplissage de votre page en rouge.
- 2) Modifier la couleur des paragraphes en bleu.
- 3) Changer le remplissage des div en vert.
- 4) Changer le remplissage des span en Jaune.
- 5) Modifier la couleur du paragraphe 4 en utilisant la couleur : #CC558F.
- 6) Modifier la couleur des paragraphe 5, 6 en utilisant la couleur: #1C22CC.
- 7) Que remarquer Vous ?

Les sélecteurs d'identifiant.

Ce sélecteur permet de cibler un élément HTML en fonction de la valeur de l'attribut ID de cet élément HTML, cette valeur est unique dans un document HTML ; cela veut dire qu'on ne peut pas trouver deux éléments HTML qui ont la même valeur de leurs attributs ID dans un même document HTML.

```

<div id="wikipedia">
  <h1> Nom du site </h1>

  <div id="contenu">
    <h1> Titre 1 </h1>
    <p>Paragraphe...</p>

    <h2> Titre 1.1 </h2>
    <p>Paragraphe...</p>
  </div>
  <div id="contenu1">
    <h1> Titre 11 </h1>
    <p>Paragraphe...</p>

    <h2> Titre 1.1.1 </h2>
    <p>Paragraphe...</p>
  </div>
</div>

```

```

#contenu1
{
  background:yellow;
}

```

8) Répondre aux questions 5,6 en utilisant les sélecteurs d'identifiants.

Remarque :

Lorsqu'on applique un style sur un élément HTML dont une propriété a plusieurs valeurs la priorité est de la gauche vers la droite selon la prise en charge de la valeur.

Exemple :

```

#contenu1
{
  background:yellow;
  font-family: Koufi serif Arial ;
}

```

Les sélecteurs de classe.

Ce sélecteur permet de cibler un ou plusieurs éléments HTML en fonction de la valeur de l'attribut **class** de ces éléments HTML, on peut trouver des éléments HTML qui ont la même valeur de leurs attributs **class** dans un même document HTML.


```

<div id="wikipedia">
  <h1 class="h1"> Nom du site </h1>
  <div id="contenu">
    <h1 class="h1"> Titre 1 </h1>
    <p>Paragraphe...</p>
    <h2> Titre 1.1 </h2>
    <p>Paragraphe...</p>
  </div>
  <div id="contenu1">
    <h1 class="h1"> Titre 11 </h1>
    <p>Paragraphe...</p>
    <h2 class="h1"> Titre 1.1.1 </h2>
    <p>Paragraphe...</p>
  </div>
</div>

```

```

.h1
{
  Color : #396DCC ;
}

```

Remarque :

- Dans un document HTML, le poids d'un sélecteur d'identifiant est plus que celle du sélecteur de classe.
- Un élément HTML peut avoir plusieurs valeurs de l'attribut class.

```

<head>
  <title>Hello!</title>
  <style>
    .h2
    {
      background: red;
    }
    #h1
    {
      background: yellow;
    }
  </style>
</head>

<body>
  <div id="wikipedia">
    <h1 class="h1"> Nom du site </h1>
    <div id="contenu1">
      <h1 class="h1 h2" id="h1"> Titre 11 </h1>
      <p>Paragraphe...</p>
      <h2 class="h1"> Titre 1.1.1 </h2>
      <p>Paragraphe...</p>
    </div>
  </div>
</body>

```

Le code HTML :

```
<head>
  <title>Hello!</title>
  <style>
    .h2
    {
      background: red;
    }
    #h1
    {
      background: yellow;
    }
  </style>
</head>

<body>
  <div id="wikipedia">
    <h1 class="h1"> Nom du site </h1>
    <div id="contenu1">
      <h1 class="h1 h2" id="h1"> Titre 11 </h1>
      <p>Paragraphe...</p>
      <h2 class="h1"> Titre 1.1.1 </h2>
      <p>Paragraphe...</p>
    </div>
  </div>
```

Exercice d'application :

```
<body>
  <div id="div1">
    <p class="p">paragraphe 1<span id="sp1"> div1</span></p>
    <p>paragraphe 2<span> div2</span></p>
    <p class="p" id="p3">paragraphe 3<span> div3</span></p>
    <span> div4</span>
  </div>
  <p>paragraphe 4</p>
  <p>paragraphe 5</p>
  <p>paragraphe 6</p>
</body>
```

- 1) Modifier le remplissage de l'élément (#div1), Sachant qu'il sera jaune.
- 2) Modifier la couleur des paragraphes de classe (.p) en bleue.
- 3) Modifier le texte (#sp1) en rouge, gras.
- 4) Mettre ce dernier texte en majuscule.
- 5) Souligner le paragraphe 4.
- 6) Mettre les paragraphes de (#div1) au centre de la page.
- 7) Modifier le code html de telle façon que les paragraphes (4,5,6) auront la même couleur que les paragraphes (2,3).

Les sélecteurs combinatoires.

Les sélecteurs de voisins directs (+).

Le combinatoire (+) permet de sélectionner les éléments HTML qui suivent directement un élément HTML.

```
<body>
  <div id="div1">
    <p class="p">paragraphe 1<span id="sp1"> div1</span></p>
    <p>paragraphe 2<span> div2</span></p>
    <p class="p" id="p3">paragraphe 3<span> div3</span></p>
    <span> div4</span>
  </div>
  <p>paragraphe 4</p>
  <p>paragraphe 5</p>
  <p>paragraphe 6</p>
</body>
```

Questions :

1) Quel est l'élément HTML qui suit directement la balise<div> ?.

<p>paragraphe 4</p>

2) Quel est l'élément HTML qui suit directement la balise<p> ?.

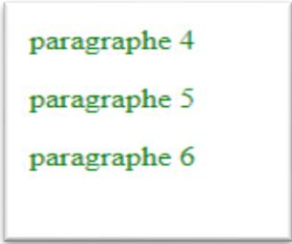
** div4**

Les sélecteurs de voisins (~).

Le combinatoire (~) permet de sélectionner les éléments HTML qui suivent un élément HTML et qui ont le même parent.

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

Le résultat est :



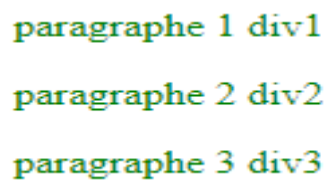
paragraphe 4
paragraphe 5
paragraphe 6

Les sélecteurs des éléments fils(>).

Le combinatoire (>) permet de sélectionner les éléments HTML qui sont des fils direct d'un élément HTML et qui ont le même parent.

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

Le résultat est :



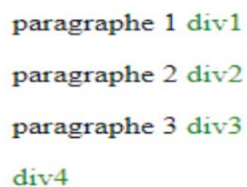
paragraphe 1 div1
paragraphe 2 div2
paragraphe 3 div3

Les sélecteurs des éléments descendants (espace).

Le combinatoire (espace) permet de sélectionner les éléments HTML qui sont des descendants d'un élément HTML et qui ont le même parent.

```
div span
{
  color:green;
}
```

Le résultat est :



paragraphe 1 div1
paragraphe 2 div2
paragraphe 3 div3
div4

Exercices d'application : TP1 CSS3

Les sélecteurs d'attributs :

C'est un sélecteur qui permet de modifier la mise en forme d'un élément HTML, selon leurs attributs ou les valeurs de ces attributs.

```
[attrib]
{
  Propriétés : valeurs ;
}
```

Ou

```
[attrib='valeur']
{
  Propriétés : valeurs ;
}
```

```
<div>
  <p class="p">paragraphe 1<span id="sp1"> div1</span></p>
  <p>paragraphe 2<span> div2</span></p>
  <p class="p" id="p3">paragraphe 3<span> div3</span></p>
  <span> div4</span>
</div>
<p>paragraphe 4</p>
<p>paragraphe 5</p>
<p>paragraphe 6</p>
```

Exemple1 :

```
[id]
{
  color:red;
}
```

Ce sélecteur cible les éléments HTML qui ont l'attribut **ID**.

```
paragraphe 1 div1
paragraphe 2 div2
paragraphe 3 div3
div4
paragraphe 4
paragraphe 5
paragraphe 6
```

```
[id='sp1']
```

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

paragraphe 1 **div1**

paragraphe 2 div2

paragraphe 3 div3

div4

paragraphe 4

paragraphe 5

paragraphe 6

Exemple 2 :

```
balise[attrib]
```

```
{  
  Propriétés : valeurs ;  
}
```

Ce sélecteur cible les éléments HTML qui ont le nom du tag « balise » et qui ont l'attribut **attrib**.

```
p[id]
```

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

Ce sélecteur cible les éléments HTML qui sont des paragraphes et qui ont l'attribut **ID**.

```
p[id='p3']
```

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

Ce sélecteur cible les éléments HTML qui sont des paragraphes et qui ont l'attribut **ID dont la valeur est p3**.

Exercice d'application TP1/Partie3.

Les pseudo-classes :

C'est un type de sélecteur permet de définir le style des éléments HTML selon leur état.

Sélecteur:etat

```
{  
  Propriétés : valeurs ;  
}
```

Exemple 1 :

```
<div>  
  <p class="p">paragraphe 1<span id="sp1"> div1</span></p>  
  <p>paragraphe 2<span> div2</span></p>  
  <p class="p" id="p3">paragraphe 3<span> div3</span></p>  
  <span> div4</span>  
</div>  
<p>paragraphe 4</p>  
<p>paragraphe 5</p>  
<p>paragraphe 6</p>  
<a href="http://www.ofppt.ma">lien vers la site de ofppt</a>  
<a href="http://www.google.ma">lien vers la site de ofppt</a>
```

P:hover

```
{  
  background: #CCCC18;  
}
```

a:visited

```
{  
  background: yellow;  
}
```

La couleur du lien visité sera changée en jaune.

a:link

```
{  
  background: green;  
}
```

La couleur de l'élément HTML sera verte car c'est un lien.

Pour plus d'informations sur les propriétés de styles voir le site :

<https://yesyouweb.com/css3-guide-reference/>

Les pseudo-éléments

Ces sélecteurs représentent les éléments HTML qui ne sont pas décrite en HTML ; ils permettent de parcourir les éléments HTML par leur position dans ce document (par ligne, par caractères.....).

Sélecteur::position

```
{  
  Propriétés : valeurs ;  
}
```

Exemple1 :

```
<div>  
  <p class="p">je suis la première ligne du paragraphe<br/>  
  je suis la deuxième ligne du paragraphe</p>  
</div>
```

```
p::first-line {  
  font-size: 120%;  
  font-weight: bolder;  
}
```

```
p::first-letter {  
  font-size: 120%;  
  font-weight: bolder;  
  color :red ;  
}
```

Résultat :

je suis la première ligne du paragraphe
je suis la deuxième ligne du paragraphe

La propriété **content** permet d'insérer un texte au début ou après une élément HTML dans un document en utilisant le CSS,


```
.p::before {  
  content: "je serai inséré au début de vos paragraphes p."  
}
```

Le texte « je serai inséré au début de vos paragraphes p » sera inséré au début des paragraphes P.

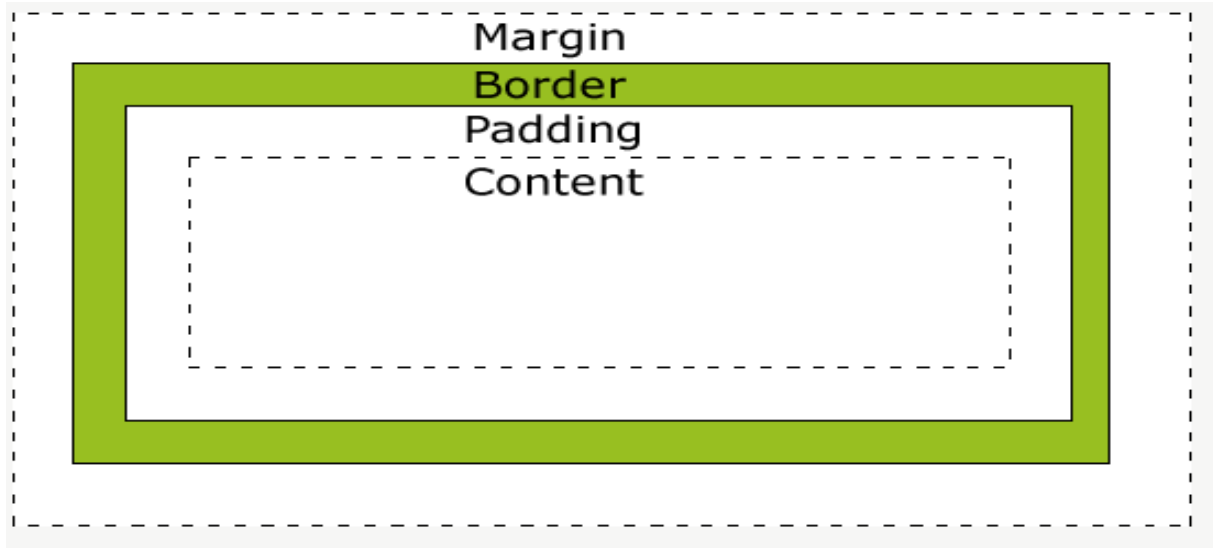
```
.p::after {  
  content: "je serai inséré au début de vos paragraphes p."  
}
```

Le texte « je serai inséré au début de vos paragraphes p » sera inséré au début des paragraphes P.

Manipulation des feuilles de style

1) Les styles d'un document HTML <body>:

Chaque élément HTML est considéré comme une boîte de dialogue qui caractérisée par les propriétés suivantes :



Remplissage du document HTML :

La propriété **Background** permet de définir le remplissage ou l'arrière-plan le document HTML ou un de ces élément Html.

Exemple 1 :

```
body
{
background: #18C9CC;
}
```

Exemple 2 :

```
body
{
background: linear-gradient(to bottom, #5CC3FF, #BF788D);
}
```

Exemple 3 ³:

```
body
{
background: radial-gradient (circle at 100%, #3f87a6, #ebf8e1, #f69d3c);
}
```

³ Utiliser le site : <https://cssgradient.io/> pour personnaliser le remplissage de votre document HTML.

```
}
```

On peut insérer une image en arrière-plan d'un document HTML.

```
body
{
    background-image: url(images/image.jpg);
    background-position: center;
    background-repeat: no-repeat;
    background-size: 800px 700px; /* width height */
    background-attachment: fixed;
}
```

background-image: définit le chemin de l'image dans le serveur.

background-position: définit la position de l'image insérée dans le document HTML ; ses valeurs sont : **right, left, center, bottom, top** ou une combinaison de ces valeurs.

background-position: top right; (coins du document)

background-repeat : permet de répéter ou non l'image au long du document HTML, ses valeurs sont : **repeat repeat-x repeat-y no-repeat.**

background-size : définit la largeur et la hauteur de l'image ; ses valeurs sont : **width height** ou les valeurs **auto cover contain .**

background-attachment: définit l'état de l'image en cas de défilement du contenu du document HTML. Ses valeurs sont : **scroll fixed.**

Les marges d'un document HTML :

On peut regrouper les marges d'un document HTML comme suit :

Exemple 1 :

```
body
{
    margin:5%;
}
```

Cette propriété applique la marge 5% de la largeur et de la hauteur du document HTML sur toutes les marges (marge à droite, marge à gauche, marge en haut – pour la marge en bas cela reste selon le contenu du document HTML).

On peut utiliser les propriétés suivantes pour détailler les marges :

margin-left:5%;	pour la marge à gauche.
margin-right:5%;	pour la marge à droite.
margin-top:5%;	pour la marge en haut.
margin-bottom:5%;	pour la marge en bas.

Exemple 2 :

```
body
{
    margin-left:10%; /*pour la marge à gauche.*/
    margin-right:10%; /*pour la marge à droite.*/
    margin-top:5%; /*pour la marge en haut.*/
    /*margin-bottom:5%; pour la marge en bas.*/ (commentaire)
}
```

Comme on peut utiliser les propriétés suivantes :

```
body
{
    margin: 15% 30% 15% 20%; /* Top - Right - Bottom - Left */
}
```

```
body
{
    margin: 15% 30%; /* Top / Bottom - Right /Left */
}
```

```
body
{
    margin: 15% 30%; 15% /* Top - Right / Left - Bottom*/
}
```

Les bordures d'un document HTML :

Pour délimiter la zone d'un élément HTML ou d'un document HTML on peut utiliser la propriété **Border** et ses descendants.

Exemple 1 :

```
body
{
    border-style: solid;
```

```
border-width: 2px;  
border-color: blue;  
}
```

On peut regrouper ces propriétés en une seule : **border :solid 2px blue ;**

Pour détailler les bordures du document HTML, on utilise les propriétés suivantes :

```
border-left-style;;  
border-right-style;;  
border-top-style;  
border-bottom-style;;
```

```
border-left-color;;  
border-right-color;;  
border-top-color;;  
border-bottom-color;;
```

```
border-left-width;;  
border-right-width;;  
border-top-width;;  
border-bottom-width;;
```

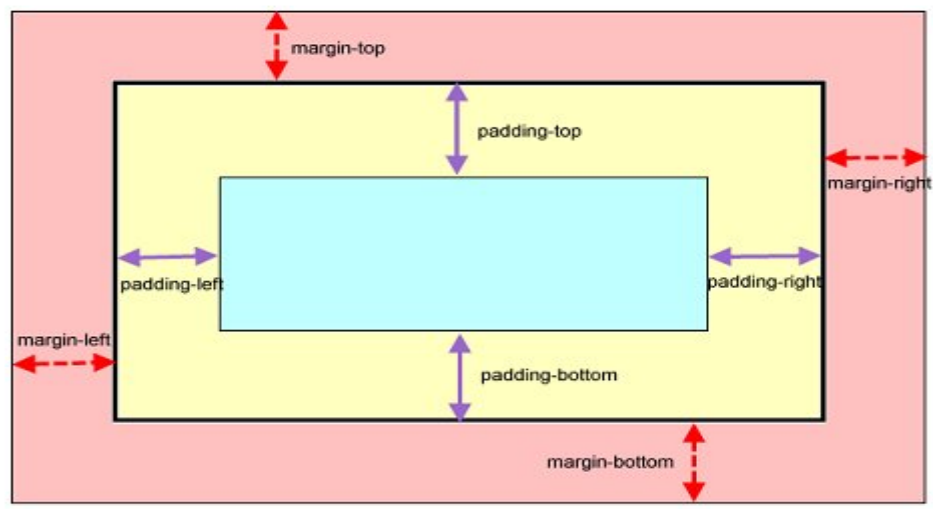
```
border-bottom-left-radius ;  
border-top-left-radius ;  
border-bottom-right-radius ;  
border-top-right-radius ;
```

```
border-left;;  
border-right;;  
border-top;;  
border-bottom;;
```

```
border: solid 2px black;  
border-radius: 25px 25px 25px 25px;  
border-width : 2px 2px 3px 5px; /* top right bottom left */
```

Les marges intérieures ou les padding d'un document HTML :

Ce sont les espaces internes qui séparent un élément HTML par rapport à son contenu.



Exemple :

```
body
{
  border:solid 2px black;
  padding: 5%;
}
```

Comme on a vu pour les marges (les marges extérieures on peut utiliser les descendants de la propriété padding qui sont :

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

De même on peut utiliser les mêmes décompositions et les appréciations de la propriété margin.

```
Padding -left:5%;    pour la marge interne à gauche.
Padding -right:5%;   pour la marge interne à droite.
Padding -top:5%;     pour la marge interne en haut.
Padding -bottom:5%;  pour la marge interne en bas.
```

Exemple 2 :

```
body
{
  Padding -left:10%;    /*pour la marge interne à gauche.*/
  Padding -right:10%;   /*pour la marge interne à droite. */
  Padding -top:5%;      /*pour la marge interne en haut. */
  Padding -bottom:5%;  /*pour la marge interne en bas.  */
}
```

Comme on peut utiliser les propriétés suivantes :

```
body
{
```

```
padding: 15% 30% 15% 20%; /* Top - Right - Bottom - Left */  
  
}
```

```
body  
{  
padding: 15% 30%; /* Top / Bottom - Right /Left */  
  
}
```

```
body  
  
{  
  
padding: 15% 30%; 15% /* Top - Right / Left - Bottom*/  
  
}
```

Les bordures d'un document HTML :

Pour délimiter la zone d'un élément HTML ou d'un document HTML on peut utiliser la propriété Border et ses descendants.

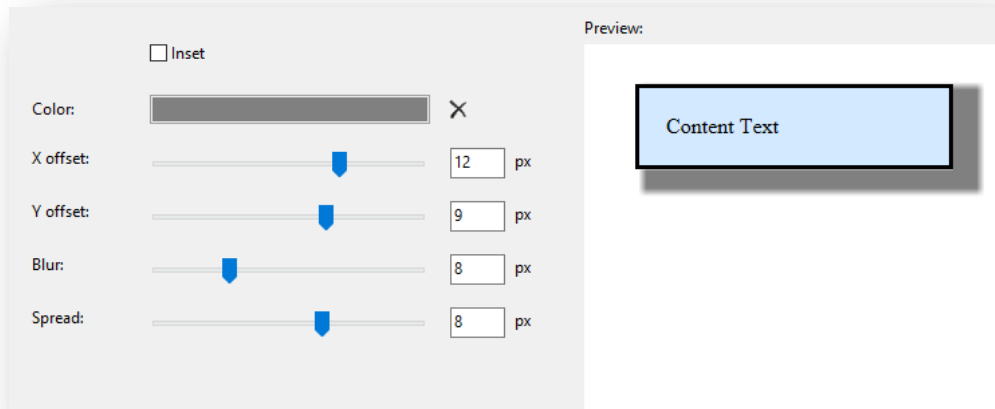
Exemple 1 :

```
body  
{  
border-style: solid;  
border-width: 2px;  
border-color: blue;  
}
```

L'ombre d'un document HTML :

La propriété box-shadow permet d'appliquer une ombre à un document HTML et à une boîte en général.

Ces valeurs sont :



```
body
{
    margin:10px;
    border-style: solid;
    border-width : 15px 18px 30px 5px;/* top right bottom left */
    border-color: blue;
    box-shadow: 12px 9px 8px 8px #808080 ;
}
```

Le résultat est :

L'informatique est un domaine d'activité scientifique, technique, et industriel concernant le traitement automatique de l'information numérique par l'exécution de programmes informatiques par des machines : des systèmes embarqués, des ordinateurs, des robots, des automates, etc. Ces champs d'application peuvent être séparés en deux branches : théorique : concerne la définition de concepts et modèles pratique : s'intéresse aux techniques concrètes de mise en œuvre. Certains domaines de l'informatique peuvent être très abstraits, comme la complexité algorithmique, et d'autres peuvent être plus proches d'un public profane. Ainsi, la théorie des langages demeure un domaine davantage accessible aux professionnels formés (description des ordinateurs et méthodes de programmation), tandis que les métiers liés aux interfaces homme-machine sont accessibles à un plus large public.

2) Les styles ou formatage d'un texte

- **font-family** : Permet de définir la police du texte.

Exemple :

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

Les valeurs de la propriété **font-family** en commençant par la première ; si cette valeur n'est prise en charge le navigateur passe à la deuxième etc....

Remarque : pour charger un font qui n'est pas pris en charge en va voir la règle [@font-face](#)

⁴

- **font-size** : définit la taille de la police qui en px ou em.

```
p
{
  font-family: Arial, Helvetica, sans-serif;
  font-size: 20px ;
}
```

Comme elle peut prendre des valeurs comme :

larger smaller xx-small x-small small x-large xx-large

- **font-style**: définit le style du texte (normal italic oblique).

```
p
{
  font-family: Arial, Helvetica, sans-serif;
  font-size: 20px;
  font-style: italic;
}
```

⁴ @font-face {
 font-family: "Bitstream Vera Serif Bold";
 src: url("/static/styles/libs/font-awesome/fonts/fontawesome-webfont.fdf491ce5ff5.woff");
}
body {
 font-family: "Bitstream Vera Serif Bold", serif;
}

- **font-weight** : permet de définir le style du texte s'il est en gras (ces valeurs sont : bold bolder)
- **text-decoration** : définit le soulignement du texte ; ces valeurs sont: underline dotted; (underline overline double dotted)
- **font-variant**: normal small-caps ;
- **text-transform**: définit la casse du texte (capitalize uppercase lowercase full-width);
- **color** : permet de changer la couleur du texte ; sa valeur peut être le nom de la couleur ou un code précédé par #.

Exemple :

```
p
{
  color:red;
}
```

Ou

```
p
{
  color: #CCCC48;
}
```

Ou

```
p
{
  color: rgb(255,120,180);
}
```

- **text-shadow** : définit l'ombre d'un texte.



Exemple :

```
p
{
  text-align: center;
  line-height: 20px;
  text-shadow: 4px 5px 5px #808080;
}
```

Mise en forme d'un paragraphe en CSS3 :

- **text-align** : permet de définir l'alignement du texte d'un paragraphe, ces valeurs sont : **left right center justify start end**
- **line-height** définit l'interligne d'un paragraphe ; sa valeur est en px.

```
p
{
  text-align: center;
  line-height: 20px;
}
```

Exercices d'application :

Les Modèle de boites en CSS3

Définition :

Chaque élément HTML est considéré comme une boite rectangulaire quel que soit cet élément en block ou inline.

- Les boites en inline :

La boîte ne crée pas de retour à la ligne; et les autres seront sur la même que la première. Certaines propriétés ne s'appliquent sur ces types de boites.

Exemple de boite inline :<label> <a>

Exemples de propriétés non applicables : width, height.

```
<style>
  label
  {
    width:70px;
    height:40px;
    background:red;
  }
  span
  {
    width:70px;
    height:40px;
    background:green;
  }
</style>
</head>

<body>
<label>Bonjour</label><span>Mes amis</span>
</body>
```

- Les boites en block ;

Ces types de boites appliquent des retours à la ligne ; on peut appliquer toutes les propriétés CSS ses Boites.

Exemple :<p><h1>....

Chaque boite est caractérisé par :

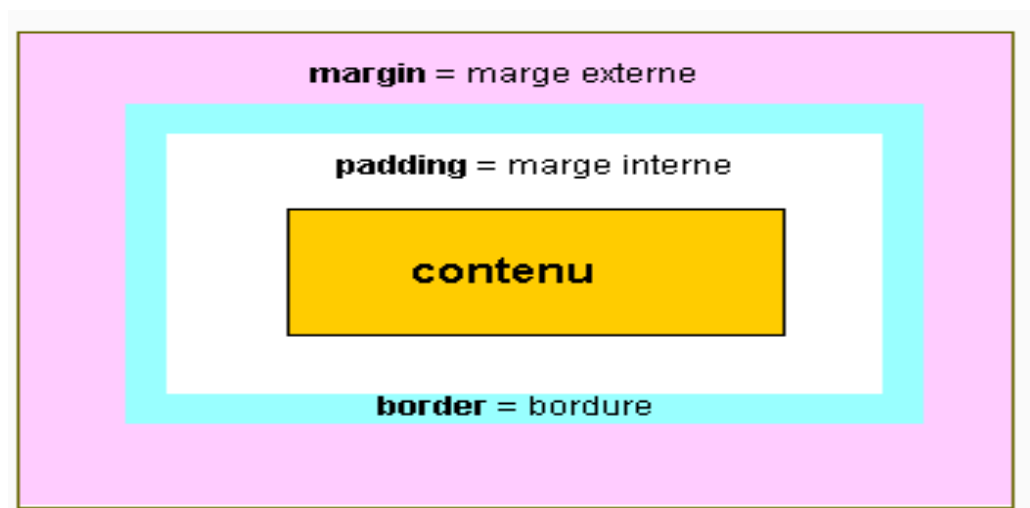


Table des matières

1) QU'EST-CE QUE C'EST CSS ?	1
2) Déclaration d'un style CSS3 :	1
• Style en ligne,	1
• Style du document:	1
• Style externe.....	3
• Style importé	4
3) Les sélecteurs CSS3 ?	5
Définition :	5
Les sélecteurs simples.	5
Le sélecteur universel.....	5
Les sélecteurs de type	6
<i>Les sélecteurs d'identifiant</i>	7
Les sélecteurs de classe.....	8
Les sélecteurs combinatoires.	11
Les sélecteurs de voisins directs (+).	11
Les sélecteurs de voisins (~).	11
Les sélecteurs des éléments fils(>).	12
Les sélecteurs des éléments descendants (espace).	12
Les sélecteurs d'attributs :	13
Les pseudo-classes :	15
Les pseudo-éléments.....	16
Manipulation des feuilles de style.....	18
1) Les styles d'un document HTML <body>:.....	18
Remplissage du document HTML :	18
Les marges d'un document HTML :	19
Les bordures d'un document HTML :	20
Les marges intérieures ou les padding d'un document HTML :	21
Les bordures d'un document HTML :	23
L'ombre d'un document HTML :	24
2) Les styles ou formatage d'un texte	25
- text-shadow : définit l'ombre d'un texte.....	27
Mise en forme d'un paragraphe en CSS3 :	27
Définition :	28

<http://css.mammothland.net/feuille-de-style-css-debutant.php>

<https://www.commentcamarche.net/contents/229-declaration-d-une-feuille-de-style-css>

<https://yesyouweb.com/css3-guide-reference/#:~:text=La%20colonne%20CSS%20dans%20nos,%3A%20CSS1%2C%20CSS2%20ou%20CSS3.>

<https://developer.mozilla.org/fr/docs/Apprendre/CSS>