

Créer ses propres pages web

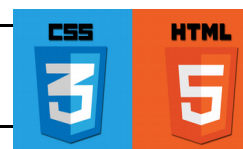


Table des matières

I.Introduction.....	1
II.Créer ses propres pages web.....	1
1)Ma première page.....	2
2)Le langage HTML.....	2
3)Structuration des documents.....	3
4)Travail à faire	4
5)Les images.....	4
6)Les liens.....	5
7)Travail à faire.....	5
8)Les listes énumérées ou pas.....	5
9)Les tableaux.....	6
10)Pour en savoir plus.....	7
III.Le CSS (Cascading Style Sheets).....	7
1)Introduction.....	7
2)Explications.....	8
3)Les balises div et span.....	8
4)La notion de boîte.....	9

I. Introduction

Nous consultons tous les jours des sites Web. Pour les sites les plus connus, on peut s'apercevoir que l'affichage et les fonctionnalités sont identiques quel que soit le navigateur utilisé (ce n'est pas toujours le cas pour des sites moins connus) à quelques détails près.

Les pages Web sont créées à l'aide de langages informatiques. Le respect des règles définies pour ces langages permet de créer des pages WEB qui pourront être lues correctement quel que soit le navigateur.

Pour créer la page Web visualisée, ont été utilisés les langages HTML et CSS. Le langage HTML a été créé en 1991 par Tim Berners-Lee. Ce dernier a aussi créé le World Wide Web Consortium (W3C) qui définit les nouvelles versions et les standards des langages liés au Web.

II. Créer ses propres pages web

Pour créer des pages Web, on peut utiliser :

- Des logiciels WYSIWYG (What You See Is What You Get) ; ce sont des programmes (Mozilla Composer, Dreamweaver et même les traitements de texte) qui permettent de créer des sites sans apprendre de langage particulier. Les pages web sont générés automatiquement. Permettant une création plus rapide au démarrage, ils présentent cependant l'inconvénient de modifications plus laborieuses (le codage n'est pas optimisé, rendant toute modification problématique)
- Des éditeurs de texte, ce sont des programmes dédiés à l'écriture de code (pas seulement HTML ou CSS) plus ou moins évolués (les fonctionnalités de certains facilitent l'écriture du code). Nous utiliserons l'éditeur de texte **Visual Studio Code**

Pour télécharger ce logiciel, il faut se rendre à l'adresse suivante :

<https://code.visualstudio.com/>

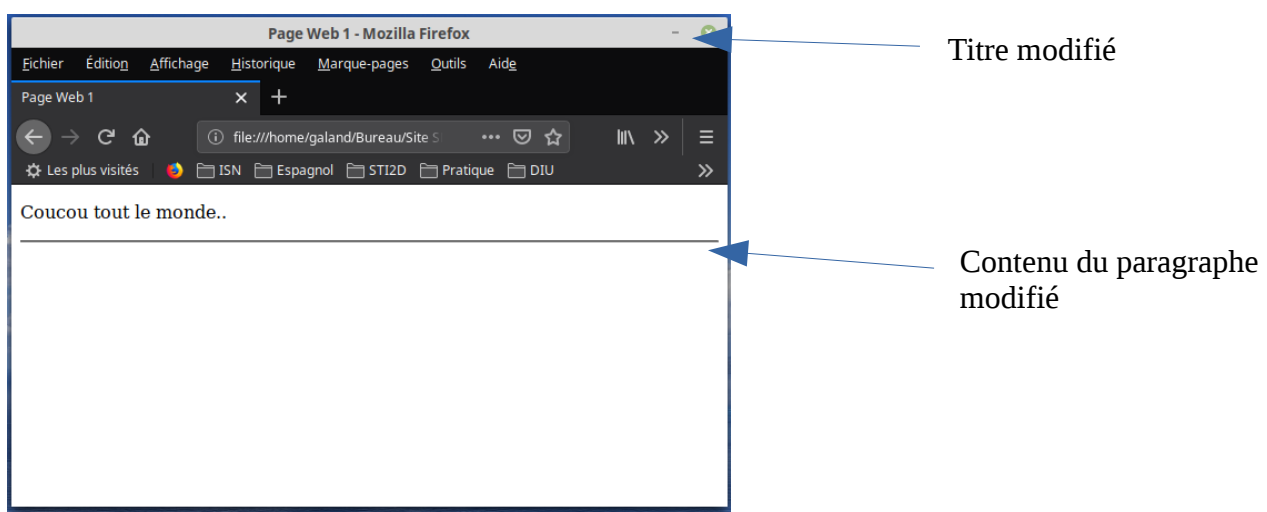
1) Ma première page

- Dans vos documents, créer un dossier nommé *monPremierSite*. Puis dans ce dossier, créer à nouveau un dossier nommé *Img* pour les images.
- Ouvrir *Visual Studio Code* et créer un nouveau document. Pour cela, il faut cliquer sur *File* dans la barre d'outil puis *NewFile*.
- Puis saisir le code suivant (ou bien faites un Copier/Coller à partir du TP) :

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8" />
  <title>Premier essai</title>
</head>

<body>
  <p>
    Bonjour tout le monde...
    <!-- ceci est un commentaire -->
  </p>
  <hr />
</body>
</html>
```

- Enregistrer votre fichier sous le dossier *monPremierSite* sous le nom de : *pageWeb1.html*
Bien vérifier, que l'extension de votre fichier est .html et non .txt
- **Remarque :** observer la coloration syntaxique. Sous *Visual Studio Code*, les balises apparaissent en bleu, les commentaires en vert et le texte visible par l'utilisateur en blanc. Si vous ne voyez pas cette coloration syntaxique, appeler le professeur.
- Un fois enregistré votre travail, ouvrir votre fichier *pageWeb1.html* avec un navigateur internet (Firefox, Chrome, Explorer ...)
- Modifier le code précédent pour obtenir ceci :



2) Le langage HTML

Le langage HTML (HyperText Markup Language) permet de concevoir des documents Web. Ce langage de programmation descriptif est composé d'**éléments**. Un élément HTML est composé

d'une **balise ouvrante** et d'une **balise fermante** afin de délimiter la zone de texte concernée par l'élément en question. Par exemple, l'élément `p` que nous avons utilisé délimite un paragraphe de texte à pour balise ouvrante `<p>` et pour balise fermante `</p>`.

Il y a également la balise `<!DOCTYPE html>`. Le *doctype* (pour Document Type Declaration) permet de renseigner le navigateur Web sur la version de HTML utilisée par le document Web. Le *doctype* utilisé dans l'exemple fait référence à HTML5.

Un fichier HTML est composé de deux parties.

`<head>` et `</head>` : les éléments contenus dans l'entête n'ont pas vocation à être affichés sauf l'élément `title` dont le contenu apparaît dans l'onglet du navigateur Web. Les informations contenues dans l'entête renseignent sur l'auteur et le contenu du document.

`<body>` et `</body>` : l'ensemble des informations contenues entre ces balises sont affichées dans la fenêtre du navigateur Web.

3) Structuration des documents

- **Titre de section** : les balises `h1`, `h2`, `h3`, `h4`, `h5`, `h6` permettent de définir des titres de sections, sous-sections, etc jusqu'à six niveaux.

Exemple :

Code

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8" />
  <title>Premier essai</title>
</head>
<body>
  <h1> Un titre de section</h1>
  <p>
    Bonjour tout le monde...
  </p>
</body>
</html>
```

Résultat

Un titre de section

Bonjour tout le monde...

- **Balises de texte** :

Balises ouvrante et fermante	Rôle
<code></code> et <code></code>	Mise en gras
<code><i></code> et <code></i></code>	Mise en italique
<code><u></code> et <code></u></code>	Soulignement
<code><sub></code> et <code></sub></code>	Texte en indice
<code><sup></code> et <code></sup></code>	Texte en exposant
<code><code></code> et <code></code></code>	Texte en police fixe. Utilisé pour insérer des codes informatiques
<code><mark></code> et <code></mark></code>	Texte surligné

- Structuration du texte . La balise `<p>` représente un paragraphe. Les espaces et les retours à la ligne sont ignorés. Chaque nouvel élément `p` crée un nouveau paragraphe

- la balise `<hr>` sert à tracer une ligne
- Dans la balise `body`, on peut placer l'attribut `bgcolor="# 33CCFF"`. Celui-ci permet de colorer le fond d'écran d'une certaine couleur (ici : le bleu clair). Attention : la couleur est écrite en hexadécimal. Cela nous donne :

Code

Résultat

```
<!DOCTYPE html>
<html>
<head>
    <meta charset="UTF-8" />
    <title>Fond d'écran</title>
</head>
<body bgcolor="#E6E6FA">
    <h1>Hello world!</h1>
    <p><a href="https://www.w3schools.com">Visit
W3Schools.com!</a></p>
</body>
</html>
```

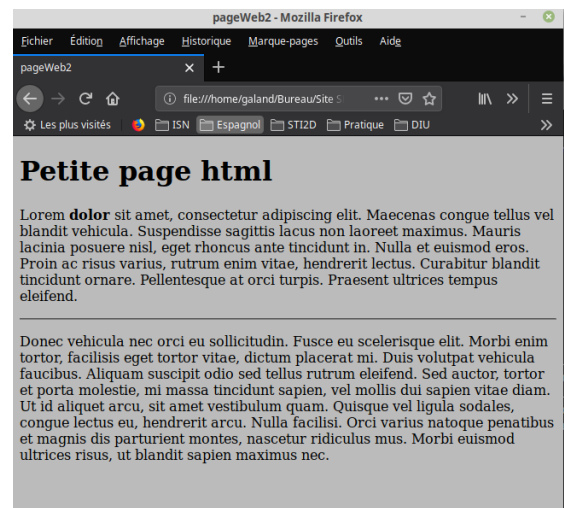
Hello world!

[Visit W3Schools.com!](https://www.w3schools.com)

4) Travail à faire

- Rechercher sur internet et écrire ci-dessous, le code hexadécimal du :
 - Blanc : #
 - Vert : #
 - Rouge : #
 - Jaune : #
 - Magenta : #
 - Cyan : #
 - Gris foncé : #
- Réaliser une page internet qui se nommera `pageWeb2` et qui ressemble à celle ci-dessous avec :
 - un titre
 - deux paragraphes écrits séparé par un ligne
 - un fond d'écran de couleur gris
 - un mot mis en évidence

Remarque : pour créer un paragraphe imaginaire, vous pouvez utiliser le site <http://fr.lipsum.com/>. En effet, on sait depuis longtemps que travailler avec du texte lisible et contenant du sens est source de distractions, et empêche de se concentrer sur la mise en page elle-même.



5) Les images

On peut insérer une image (au format png, jpeg ou gif) qui se trouve dans le dossier *Img* du dossier *monPremierSite* grâce à la balise ``. Donc pour insérer une image (*smiley.gif* préalablement enregistrée dans le dossier *Img*), on peut procéder de la manière suivante :

```
<img src= "Img/smiley.gif" alt="Smiley face" width="42" height="42">
```

Détaillons cette ligne :

- `src="smiley.gif"` donne le nom de la photo à afficher (*smiley.gif*). Cet attribut est obligatoire.
- `alt="Smiley face"` spécifie un texte alternatif si l'image ne s'affiche pas (*Smiley face*). Cet attribut est recommandé.

- `width="42" height="42"` spécifie la largeur (42 pixels) et la hauteur (42 pixels). Ces attributs sont facultatifs.

6) Les liens

HTML (Hyper Text Markup Language) est un langage hypertexte, cela signifie qu'il vous permet en cliquant sur un mot, généralement souligné (ou une image) de vous rendre:

- sur un autre endroit de la page.
- sur un autre fichier html situé sur votre ordinateur.
- sur un autre ordinateur en réseau, y compris sur le Web.

Pour écrire un lien sur une page web, il faut adopter la syntaxe suivante :

```
<a href="lien"> texte </a>
```

avec :

- **texte** : texte qui apparaît en bleu souligné dans la page
- **lien** : adresse d'un site (Exemple : *<https://fr.wikipedia.org/wiki/Informatique>*) , d'une autre page html située dans notre dossier *monPremierSite* (exemple : *pageWeb3.html* à créer ultérieurement) ou bien un mot dans la page web concernée.

Exemple: voir code et résultat précédents

7) Travail à faire

Réaliser deux pages internet qui ont chacune

- Un titre ;
- deux paragraphes ;
- une image de 300 pixels de largeurs entre les deux paragraphes et la même image à la fin ;
- un fond d'écran de couleur ;
- Un lien pour l'autre page et un lien vers un fichier existant.

8) Les listes énumérées ou pas

On peut organiser le texte en liste. Une liste énumérée est contenue dans un élément *ol* (abréviation de l'anglais ordered list). Une liste non-énumérée est contenue dans un élément *ul* (abréviation de l'anglais unordered list). Quelque soit le type de liste, chaque entrée de liste est contenue dans un élément *li*. Les listes peuvent évidemment être imbriquées.

Exemple :

Code

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8" />
  <title>pageWeb2</title>
</head>

<body >
<h1>Les listes </h1>
<p>
  <ul>
    <li>Une chose </li>
    <li>Puis une autre </li>
  </ul>
  <ol>
    <li>Une chose </li>
    <li>Puis une autre </li>
  </ol>
</p>
</body>
</html>
```

Résultat

Les listes

- Une chose
 - Puis une autre
1. Une chose
 2. Puis une autre

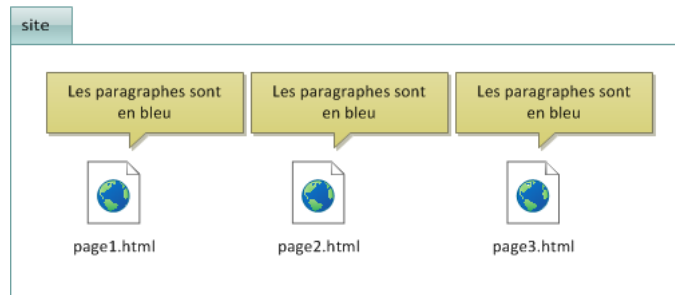
9) savoir plus

Pour connaître d'autres balises et d'autres attributs, consulter le site W3S.

III. Le CSS (Cascading Style Sheets)

1) Introduction

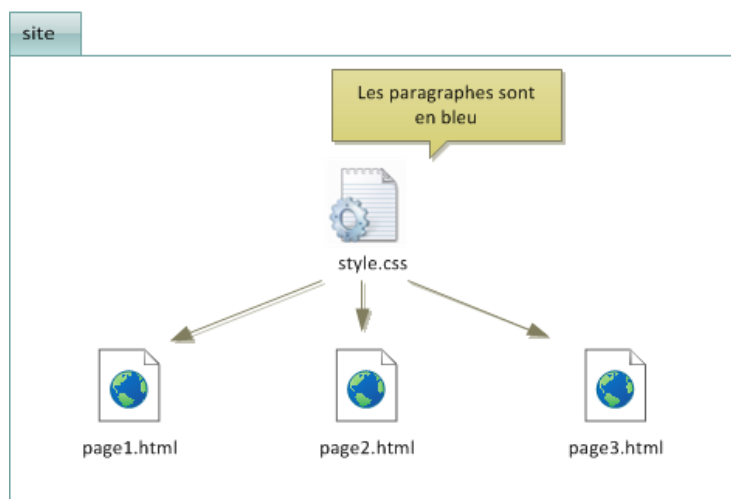
Vous avez remarqué que si nous voulons donner une unité au site, il est nécessaire, pour chaque page, de définir par exemple une police bleue.



Or, nous pouvons aussi utiliser une page de style écrite dans un autre langage : le CSS. Le Langage CSS, apparu en 1996, est complémentaire du langage HTML et permet de gérer la mise en forme (l'apparence) des différentes pages d'un site Web (couleur du texte, police, taille du texte, bordures, fond...)

Ainsi, dans l'exemple précédent, il suffit de créer une page style.css (avec Visual Studio Code par exemple) dans laquelle il écrit que les paragraphes sont en bleu. Toutes les pages html créées appelleront style.css dans l'en-tête (head) de la page grâce à l'instruction :

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



Code Page 1

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>pageWeb3</title>
  <link href="style.css" rel="stylesheet" type="text/css" >
</head>

<body >

<h1>Page Web 3 </h1>
<p>
  Donec laoreet nunc sit amet mauris porttitor,
  sagittis euismod velit tristique.
  Praesent in libero metus. Cras dapibus erat
  nec mauris fringilla, rhoncus ultrices erat
</P>

<a href="pageWeb1.html">Lien vers la page web1</a>
<br>
<a href="pageWeb2.html">Lien vers la page web2</a>

</body>
</html>
```

Insertion :
page de style

Code Page 2

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8">
  <title>pageWeb1</title>
  <link href="style.css" rel="stylesheet"
type="text/css" >
</head>

<body >

<h1>Page Web 2 </h1>
<p>
  Vestibulum sed ligula libero. Vivamus
vestibulum orci at augue dictum
  lobortis. Nunc cursus nunc et leo tincidunt,
vitae elementum dolor eleifend. Aenean non
placemat sem. Donec finibus lacus mauris,
</p>

<a href="pageWeb1.html">Lien vers la page
web1</a>
<br>
</body>
</html>
```

fichier css

```
body {
  background-color: gainsboro;
}

p{
  color:blue;
  text-align: justify;
}
```

Résultat

Page Web1

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Fusce ultricies aliquet libero ut placerat. Nullam tincidunt felis id nisi vestibulum lacinia. Curabitur rhoncus velit id lacinia pretium. Donec at felis et ex posuere mattis quis eu augue. Suspendisse pretium enim et lorem bibendum tempus. Vivamus a dui vulputate, blandit libero vel, viverra libero. Duis vel sollicitudin nisi, sit amet scelerisque lacus. Nullam eu nisl sem. Vestibulum erat enim, tincidunt vitae ultricies sed, euismod vel nunc.

[Lien vers la page web2](#)
[Lien vers la page web3](#)

2) Explications

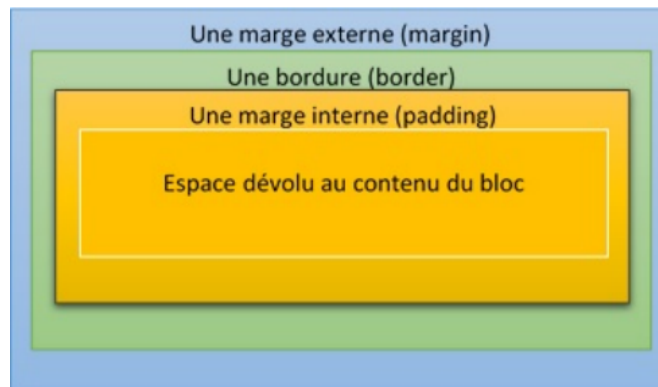
Ici, on indique que l'élément p (paragraphe) contient du texte aligné des deux côtés (justifié) et de couleur bleue. On indique également que l'élément body (corps de la page HTML) aura un fond d'écran couleur gainsboro c'est-à-dire bleu très clair.

En CSS :

- Body et p sont nommés selecteur
- color, text-align, background-color sont nommés propriétés CSS
- gainsboro, blue, justify sont nommés valeurs

3) La notion de boîte

Dans le modèle des feuilles de style, tous les éléments d'une page HTML se coulent dans des blocs (ou « boîtes ») rectangulaires. Un bloc typique est représenté ci-dessous :



A partir de l'extérieur, on rencontre successivement 4 zones :

- Une marge externe (**margin**), toujours transparente, avec des épaisseurs réglables côté par côté ;
- Une bordure (**border**) dont on peut régler l'épaisseur, la couleur et le style côté par côté ;
- Une marge interne (**padding**), ou « espacement », d'épaisseur réglable côté par côté ;
- L'espace dévolu au bloc

Il existe en HTML deux catégories d'éléments textuels : les éléments *block* (les éléments `<p>`, `` sont par défaut des éléments *block*) qui créent une boîte et forcent un retour à la ligne et les éléments de type *inline* (les éléments `<a>`, ``, `<i>`s ont par défaut des éléments *inline*) qui placent leur contenu dans le flot du texte. Il est possible faire passer des éléments *inline* en éléments *block* (et inversement à l'aide de la propriété *display*).

4) Quelques propriétés CSS

De nombreuses propriétés CSS indiquent des longueurs (taille d'une bordure, d'un ajustement, taille des polices). Leur valeur est toujours un nombre suivi d'une unité. Parmi les unités légales en CSS, on retrouve notamment le pixel (px) et le point (pt).

Nous donnons ci-dessous quelques propriétés CSS.

Propriété	Valeur	Description
display	None, block ou inline	mode d'affichage de la boîte
background	couleur	couleur de fond de la boîte
color	couleur	couleur de texte
border	taille motif couleur ou none	Si none, la bordure est masquée. Sinon, on donne la taille, le motif (solid, dotted ou dashed) et la couleur séparés par des espaces.
margin	longueur	taille des marges
padding	longueur	taille des ajustements
text-decoration	none, underline, overline, ou line-through	décoration du texte
text-align	left, right, ou justify	justification du texte

font-family	fixed, serif, ou sans serif	nom de la police
font-weight	normal, light, bold, ou bolder	graisse de la police
font-style	normal ou italic	style de la police
font-size	Longueur ou xx-small, x-small, small, normal, large, x-large, xx-large	taille de la pôlece

5) Les balises div et span et les attributs class

Les éléments div sont des balises de type *bloc* tandis que les éléments *span* sont des balises de types *inline*. Elles sont surtout utiles pour "déconnecter" certains morceaux de paragraphe ou plusieurs paragraphes de cette logique d'écriture avec des feuilles de style. Elles créent ainsi des petits blocs particuliers dans le document sans devoir repasser par les éléments structurels du Html classique.

Les attributs class et id sont des attributs universels ce qui signifie qu'on va pouvoir les utiliser avec n'importe quel élément HTML, et notamment avec les éléments div et span.

En pratique, il va être très courant de préciser des attributs class et id pour nos éléments div et span pour pouvoir appliquer des styles à un div (ou span) ou à un groupe d'éléments div ou span) définis.

Page HTML

Page CSS

```
<h1>Un titre de niveau 1</h1>

<div class="conteneur" id="cp">
  <p>Un premier paragraphe</p>
  <p>Un autre paragraphe</p>

  <ul>
    <li>Un élément de liste</li>
    <li>Un <span class="gras">autre
élément</span> de liste</li>
  </ul>
</div>

<p>Un <span class="gras
fondjaune">troisième</span>
paragraphe</p>

<div>
  <p class="gras">Un dernier
paragraphe</p>
</div>
```

```
/*Couleur de fond du body (donc de la partie visible de la
page) : bleu clair*/
body{
  background-color: lightBlue;
}

div{
  background-color: purple; /*Les div auront un fond violet*/
  color: white; /*Textes des éléments dans les div blancs par
héritage*/
}
.conteneur{
  width: 80%; /*La largeur de l'élément sera égale à 80% de son
parent*/
}
#cp{
  margin: 0 auto; /*Permet de centrer le div dans son élément
parent (body ici)*/
}

.gras{
  font-weight: bold; /*Les textes seront en gras*/
}
.fondjaune{
  background-color: yellow; /*Fond des span jaune*/
}
```

Résultats

Un titre de niveau 1

Un premier paragraphe

Un autre paragraphe

- Un élément de liste
- Un **autre élément** de liste

Un **troisième** paragraphe

Un **dernier** paragraphe

6) Les tableaux

On peut enfin organiser du texte dans des tableaux au moyen de l'élément *table*, qui contient l'ensemble des lignes du tableau, chacune d'elle consistant en un élément *tr* (abréviation de l'anglais table row). Enfin chaque ligne décrit les cases qu'elle contient au moyen de l'élément *td*.

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="UTF-8" />
  <title>pageWeb2</title>
</head>

<body >

<h1>Les tableaux </h1>

<hr>

<table>
  <tr>
    <td>U (V)</td>
    <td>1</td>
    <td>2</td>
    <td>3</td>
  </tr>
  <tr>
    <td>I (A)</td>
    <td>5</td>
    <td>10</td>
    <td>15</td>
  </tr>
</table>

</body>
</html>
```

Les tableaux

U (V)	1	2	3
I (A)	5	10	15

Pour mettre en forme les bordures, alignement du texte, les fonds d'écrans, il faut passer par le CSS.

Note : pour avoir toutes les bordures, il faut appliquer le code CSS suivant :

```
table{
  border-collapse: collapse
}

td{
  border: 1px solid black;
}
```

7) Pour en Savoir plus

Vous pouvez aller sur le site W3S où il existe un validateur HTML5 et sur le site de Pierre Giraud (<https://www.pierre-giraud.com/html-css-apprendre-coder-cours/introduction/>).

8) Travail à faire

Tenter de faire à l'identique le compte-rendu de TP situé sur l'ENT dans le dossier *Espace des classes, spécialité NSI*, [Dossiers partagés](#) > [COURS](#) > [Exercice HTML CSS](#) puis *TP_PS.pdf*.

Remarque : les images se trouvent dans le même dossier

