

LE CODE HTML

1- INTRODUCTION

Une page web est écrite nécessairement en code.
 Il en existe beaucoup, aux utilités diverses et variées.
 Le code le plus basique est toutefois le HTML (**H**yper**T**ext **M**arkup **L**anguage)

Définition du HTML par WIKIPÉDIA :

L’Hypertext Markup Language, généralement abrégé HTML, est le format de données conçu pour représenter les pages web.

C’est un langage de balisage permettant d’écrire de l’hypertexte, d’où son nom.

HTML permet également de structurer sémantiquement et de mettre en forme le contenu des pages, d’inclure des ressources multimédias dont des images, des formulaires de saisie, et des programmes informatiques.

Il permet de créer des documents interopérables avec des équipements très variés de manière conforme aux exigences de l’accessibilité du web.

Il est souvent utilisé conjointement avec des langages de programmation (JavaScript) et des formats de présentation (feuilles de style en cascade).

HTML est initialement dérivé du Standard Generalized Markup Language (SGML).

Structure d’un document HTML

Source HTML	Modèle du document
<pre style="font-family: monospace; font-size: 0.9em;"> <!DOCTYPE html PUBLIC "-//IETF//DTD HTML 2.0//EN"> <html> <head> <title> Exemple de HTML </title> </head> <body> Ceci est une phrase avec un hyperlien. <p> Ceci est un paragraphe où il n’y a pas d’hyperlien. </p> </body> </html> </pre>	<div style="border: 1px solid #ccc; padding: 10px;"> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">html <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">head <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">title <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">texte</div> </div> </div> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">body <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">texte</div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">a <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">texte</div> </div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">texte</div> <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">p <div style="border: 1px solid #ccc; padding: 5px; margin-bottom: 5px;">texte</div> </div> </div> </div>

2- LES BALISES

Le langage HTML se construit avec des BALISES

Une BALISE se rédige en minuscule et est entourée de crochets : < >
et fonctionne en paire, c'est-à-dire une balise d'entrée et une balise de fin,
celle-ci se différenciant par un /.

Exemple :

Pour nommer une page web on utilise la balise "title", on trouveras donc en haut d'une page HTML

```
<title>  
le titre de la page  
</title>
```

3- LA MISE EN FORME DE TEXTE

3.1- LES BALISES PRINCIPALES

Pour pouvoir rédiger et mettre en forme un texte en HTML, nous aurons donc besoin
d'un certain nombre de balises.

Voici ci-dessous les premières balises à connaître :

```
<p> </p> Commence et termine un paragraphe  
<br/> permet d'aller à la ligne  
<strong> </strong> écriture en gras  
<i> </i> écriture en italique  
<u> </u> souligné
```

3.2- LA JUSTIFICATION

Pour justifier un texte, il faut indiquer à la balise la façon d'afficher ce qu'elle contient
en utilisant le terme "**align**" :

Exemple :

```
<p align="center">  
le texte  
</p>  
de cette façon, le texte sera aligné au centre ("center")
```

```
right : le texte sera aligné à droite  
left : le texte sera aligné à gauche  
center : le texte sera centré  
justify : le texte sera justifié sur la largeur de la page
```

3.3- LA MISE EN FORME VIA LE CSS

Les balises vues précédemment peuvent être déjà caractérisées par défaut, via un fichier appelé le **CSS**.

Le **CSS** (*Cascading Style Sheets*) est lui aussi un code de programmation.

Il est utilisé pour créer un fichier dont l'extension est .css, et qui est placé dans le site.

Chaque page web peuvent ainsi y être rattachées, et "lire" ce code.

On détermine dans ce fichier la façon dont les éléments d'une page web doivent se comporter.

Par exemple, on y donne la taille, la couleur, le style d'une police d'écriture.

Pour ce faire, on détermine un nom, une "class", et on indiquera dans notre page web de présenter certains éléments suivant ce "code".

Exemple :

Considérons que dans notre fichier.css nous ayons créé une **class** qui se nomme "**gras_orange**".

Cette **class** détermine que les textes seront en gras et de couleur orange.

La syntaxe employée sera alors la suivante :

```
<p class="gras_orange">  
mon texte  
</p>
```

Les caractéristiques qu'il est possible de déterminer dans le CSS sont indénombrables.

La taille, la couleur, le style, la justification, les marges, la réactivité (pour les liens)...

quelque soient la nature de l'objet : une police, une image, une vidéo, un tableau...

Bref, tout y est paramétrable.

La mise en forme par le CSS n'empêche pas pour autant d'utiliser la mise en forme "classique".

Exemple :

```
<p class="gras_orange" align="center">  
mon texte  
</p>
```

Dans ce cas, le texte sera centré et gras en orange, tel que déterminé par le CSS.

Il faut toutefois éviter de trop mélanger les deux méthodes. Certaines caractéristiques données en CSS pourraient entrer en conflit avec d'autres données en "direct" et aboutir à un bug.

Voici les "**class**" utilisées dans le CSS du site de l'UTL :

```
new petit violet  
gras_orange gras orange  
gras_violet gras violet  
txt_gris gris  
txt_grispt gras petit  
txt_grisgras gras gris
```

NB : Tout le site de l'UTL, est, via son .css, par défaut, rédigé en **gris**.

3.4- LES LISTES

Nous pouvons créer avec le HTML, une liste avec puces.
Pour ce faire nous utiliserons 2 balises :

** ** qui commence une liste
** ** qui déterminera le contenu de chaque ligne

Exemple :

```
<p> Voici notre liste :<br/>
  <ul>
    <li> première ligne avec puce</li>
    <li> deuxième ligne avec puce</li>
    <li> troisième ligne avec puce</li>
  </ul>
</p>
```

Ce qui donnera à l'affichage de la page web :

Voici notre liste :

- première ligne avec puce
- deuxième ligne avec puce
- troisième ligne avec puce

3.5- LES TABLEAUX

Nous pouvons également créer des tableaux. Comme nous l'avons vu précédemment, les balises utilisées pour les tableaux sont elles aussi paramétrables via le CSS.

Dans le cadre de nos cours, nous utiliserons les paramètres déterminés dans le site de l'UTL.

Nous utiliserons cette fois 4 balises :

<table></table> ouvre un tableau
<tr> </tr> ouvre une ligne
<th> </th> ouvre la première colonne (cellule de tête)
<td> </td> ouvre les colonnes suivantes

Exemple :

<table>	<i>ouverture du tableau</i>	
<tr>	<i>ouverture de la première ligne</i>	tr (1)
<th> 1ère colonne</th>	<i>colonne1, ligne1</i>	th (1)
<th> 2ème colonne</th>	<i>colonne2, ligne1</i>	th (2)
<th> 3ème colonne</th>	<i>colonne3, ligne1</i>	th (3)
</tr>	<i>fermeture de la première ligne</i>	
<tr>	<i>ouverture de la deuxième ligne</i>	tr (2)
<td>contenu 1</td>	<i>colonne1, ligne2</i>	td (1)
<td>contenu 2</td>	<i>colonne2, ligne2</i>	td (2)
<td>contenu 3</td>	<i>colonne3, ligne2</i>	td (3)
</tr>	<i>fermeture de la deuxième ligne</i>	

`</table>`

fermeture du tableau

Ce qui donnera à l'affichage de la page web :

1ère colonne	2ème colonne	3ème colonne
contenu 1	contenu 2	contenu 3

et qui pourrait s'illustrer ainsi :

tr (1) - th (1)	tr (1) - th (2)	tr (1) - th (3)
tr (2) - td (1)	tr (2) - td (2)	tr (2) - td (3)

4- LES CARACTÈRES SPÉCIAUX

Le HTML est un code qui a aujourd'hui un certain nombre d'années.

Originellement, les caractères spéciaux (accents, tréma, etc..) devaient impérativement être transcrits selon le code ci-dessous :

&nbsp;	espace
&eacute;	é
&Eacute;	É
&egrave;	è
&Egrave;	È
&agrave;	à
&Agrave;	À
&ccedil;	ç
&Ccedil;	Ç
&oelig;	œ
&Oelig;	Œ
&icirc;	î
&Icirc;	Î
&ecirc;	ê
&Ecirc;	Ê
&ocirc;	ô
&Ocirc;	Ô
&euml;	ë
&Euml;	Ë

Aujourd'hui, la plupart des navigateurs internet (Internet Explorer, Mozilla Firefox, Safari, Apache...) acceptent, ou du moins, tolèrent les textes rédigés de façon conventionnelle.

Toutefois, dans un souci d'efficacité, et pour s'assurer que l'internaute aura bien le texte affiché convenablement, il convient de continuer à utiliser ces codes.