

Tutoriels mirianet.com : **HTML**

1. [Introduction](#)
2. [Structure de base d'un document HTML](#)
3. [Titres et séparateurs de paragraphes](#)
4. [Mots et caractères](#)
5. [Les liens](#)
6. [Listes](#)
7. [Images](#)
8. [Images cliquables](#)
9. [Les tables ou tableaux](#)

1. Introduction

Contrairement à la plupart des langages de programmation, le HTML (HyperText Markup Language) est un langage statique dit *de balises*. Son rôle est de formater la présentation de documents contenant textes, images

Dans un document Html, chaque entité (texte, image ...) est encadrée par 2 balises * (html) : Balise de début et balise de fin. Chaque balise caractérisant un mode bien défini de représentation de l'information et pouvant également avoir des paramètres modifiables (taille, couleur, alignement ...). De plus chaque balise peut être à son tour contenue dans une autre (imbrication).

Le rôle du navigateur est, quant à lui, de décoder la trame html, dans restituer son aspect visuel et de permettre également à l'utilisateur d'interagir avec le serveur Web.

Le langage Html s'apparente donc aux langages de mises en forme comme on les retrouve dans les éditeurs de texte (MS WORD, Word Perfect ...).

Pour créer une page html, un simple éditeur comme le *bloc note* de windows suffit, cependant il existe des éditeurs beaucoup plus sophistiqués comme [DreamWeaver](#), [WebExpert](#), [HotDog](#), ou [FrontPage](#) de Microsoft. Ces éditeurs sont de type WYSIWYG ** et donc souvent très évolués, mais il en existe de beaucoup plus simples, comme par exemple [HTML-Kit](#) qui est bien plus confortable à utiliser qu'un simple *bloc note* et qui en plus est gratuit.

* Il existe cependant des balises de type unaire (une seule) : Exemple la balise `
`, placée après un texte, insèrera un retour à la ligne après celui-ci.

** WYSIWYG (What You See Is What You Get) est un acronyme couramment utilisé en informatique pour désigner les interfaces utilisateur graphiques permettant de composer visuellement le résultat voulu, typiquement pour un logiciel de mise en page, un traitement de texte ou d'image. Il désigne, de fait, une interface « intuitive » : l'utilisateur voit directement à l'écran à quoi ressemblera le résultat final (imprimé). WYSIWYG est l'acronyme de la locution anglaise *what you see is what you get* signifiant littéralement « ce que vous voyez est ce que vous obtenez » ou plus élégamment « tel écran, tel écrit ».

2. Structure de base d'un document HTML

Tout document Html commence par la balise `<html>` pour se terminer par la balise `</html>` (On notera que le "/" est le caractère signifiant "fin de balise").

Entre ces deux balises on trouvera la tête (head) : `<head></head>` et le corps (body) `<body></body>` .

Toutes les informations visibles se retrouveront dans le *corps*, la *tête* ne contenant que des informations *invisibles* * mais utiles au navigateur, aux robots des moteurs de recherche et généralement on y écrit aussi les références à des fichiers externes (feuilles de styles CSS, fichiers de code javascript ...) ou du code javascript.

* La seule exception et le contenu de la balise *titre* `<title></title>` , qui sera visible dans la barre d'état du navigateur.

Ma première page HTML

Code HTML	Affichage HTML
<pre><html> <head> <title>Titre du Navigateur</title> </head> <body> <center>Ma première page HTML</ center> </body> </html></pre>	Ma première page HTML

3. Titres et séparateurs de paragraphes

- [Titres](#)
- [Paragraphes](#)
- [Séparateurs](#)
- [La balise DIV](#)

- **Titres**

Il existe 6 niveaux de titre dans HTML. Un retour à la ligne est automatiquement inséré après un titre.

Syntaxe: `<Hn></Hn>` où n est compris entre 1 et 6 (H1 est le plus grand titre possible).

Exemple :

Code HTML	Affichage HTML
<pre><h1>Titre de type h1</h1> <h2>Titre de type h2</h2> <h3>Titre de type h3</h3> <h4>Titre de type h4</h4> <h5>Titre de type h5</h5> <h6>Titre de type h6</h6></pre>	<p>Titre de type h1</p> <p>Titre de type h2</p> <p>Titre de type h3</p> <p>Titre de type h4</p> <p>Titre de type h5</p> <p>Titre de type h6</p>

Paramètre :

ALIGN sert à spécifier l'alignement horizontal d'un titre.

Syntaxe: `ALIGN="LEFT ou CENTER ou RIGHT"`

Exemple :

Code HTML	Affichage HTML
<pre><h1 align="center">Texte du titre centré</h1></pre>	<p>Texte du titre centré</p>

[En haut](#)

- **Paragraphes**

-Paragraphe normal

Le marqueur de paragraphe "p" produit un double retour de ligne.

Syntaxe: `<p> ... </p>`


```
<address>
This document was created by Patrick.Jermann@tecfa.unige.
ch
Last modified: 2/7/95
</address>
```

This document was created by Patrick.
Jermann@tecfa.unige.ch
Last modified: 2/7/95

[En haut](#)

• Séparateurs

-Retour de ligne

Le marqueur "br" effectue un simple retour de ligne.

Syntaxe: `
` ou `
`

Exemple :

Code HTML	Affichage HTML
<pre>ligne 1
 ligne 2
</pre>	<pre>ligne 1 ligne 2</pre>

- Ligne horizontale

Le marqueur "hr" insère une barre horizontale.

Syntaxe: `<hr >` ou `<hr />`

Paramètres :

-SIZE :

spécifie l'épaisseur de la barre horizontale en pixels. La valeur par défaut est 1.

Syntaxe: `SIZE="n"`

-WIDTH

spécifie la longueur de la ligne, soit en pixels, soit en pourcentage de la largeur de la fenêtre.

Syntaxe: `WIDTH="n"`

-ALIGN

spécifie l'alignement de la barre

Syntaxe: `ALIGN="LEFT ou RIGHT ou CENTER"`

-NOSHADE

ne prend pas de valeur. Lorsqu'il est présent dans le marqueur "hr" l'effet est une ligne pleine sans ombrage.

Syntaxe: `NOSHADE`

Exemple :

Code HTML	Affichage HTML
<pre><hr size="10"> <hr width="200"> <hr width=60%> <hr align="center"> <hr NOSHADE></pre>	

[En haut](#)

- **La balise DIV**

-Permet de regrouper des éléments

-On peut associer des attributs communs aux éléments contenus entre les balises

Syntaxe: `<div>Liste d'éléments</div>`

Exemple :

Code HTML	Affichage HTML
<pre><div align="center"> <h1> Introduction </h1> <p> Bonjour </p> </div></pre>	<p style="text-align: center;">Introduction</p> <p style="text-align: center;">Bonjour</p>

Cette balise est surtout utile pour associer des styles, cf. CSS (Cascading Style Sheet)

Cours CSS débutant : <http://mammothland.free.fr/cours/css/>

Exemple :

```
<DIV class="cours">
.....
</DIV>
```

4. Mots et caractères

- [Mise en forme de mots](#)
- [Caractères accentués](#)

- **Mise en forme de mots**

Voici quelques marqueurs de mise en forme de mots les plus utilisés en HTML.

Code HTML	Affichage HTML
<pre><div align="center"> Bold <I>Italic</I> <U>Underline</U> <S>Strikethru</S> <SUP>Superscript</SUP> <SUB>Subscript</SUB> Emphasised Strong <CODE>Code</CODE> <Q>Quote</Q> <CITE>Citation</CITE> </div></pre>	<p>Bold <i>Italic</i> <u>Underline</u> Strikethru Superscript Subscript <i>Emphasised</i> Strong Code Quote <i>Citation</i></p>

[En haut](#)

- **Caractères accentués**

Certains éditeurs prennent en charge la conversion des caractères accentués spécifiques à une langue. Sinon le HTML ci-dessous permet d'obtenir le caractère souhaité à l'affichage.

Tableau 1: Codes HTML pour caractères spéciaux

Code HTML	Affichage HTML
-----------	----------------

â comme château
à comme je vais à la maison
ç comme garçon
é comme été
è comme près
ê comme être ou ne pas être
> comme 10 > 9
< comme 9 < 10
î comme chaîne
 comme 3 espaces: !
ô comme ôrage ôdé
sespoir
" comme "entre guillemets"
û comme août

â comme château
à comme je vais à la
maison
ç comme garçon
é comme été
è comme près
ê comme être ou ne
pas être
> comme 10 > 9
< comme 9 < 10
î comme chaîne
comme 3 espaces: !
ô comme ôrage ô
désespoir
" comme "entre
guillemets"
û comme août

Pour avoir le rendu ci-dessus, mieux vaut insérer dans la partie **<head>** du code html, la balise **<META http-equiv="Content-Type" Content="text/html; charset=ISO-8859-1"> ***

Il faut savoir aussi qu'un affichage correct est aussi dépendant du navigateur.

* : charset ISO-8859-1 aussi appelé charset latin1 comprend les caractères français

5. Les liens

Les liens permettent de construire un hypertexte et peuvent être de différents types:

- externes: un pointeur du document mène vers un autre document
- internes: un pointeur renvoie à une section du même document

Bien sûr il est possible de mixer les deux (aller dans un pointeur contenu dans un autre document)

Un lien se définit par le marqueur `<a ...>` suivi du paramètre `HREF="URL"` qui définit l'adresse du document vers lequel le lien conduit. Il se termine par ``.

Le texte ou l'image qui sont insérés entre les marqueurs de début et de fin sont les parties actives du lien ("mots_sensibles" ci-dessous) et déclencheront le chargement du document lorsqu'on clique dessus.

les liens internes sont caractérisés par le marqueur `#`

Syntaxe: `Texte visible`

Paramètres :

-TARGET :

Spécifie la frame (au niveau html) où l'url appelée doit apparaitre après avoir cliqué sur le lien.

Syntaxe: `TARGET="_self"` ou `_top` ou `_blank` ou `<nom de frame>` "

`_self` affichera l'url dans la même frame, `_top` permet de "sortir" des cadres, `_blank` créera une nouvelle fenêtre, et `<nom de frame>` affichera l'url dans la frame de même nom.

-TITLE :Permet d'afficher un texte sur survol de la souris (ne marche pas avec tous les navigateurs)

Remarques :

-Si target n'a pas de valeur il prendra implicitement la valeur "self".

-Si `<nom de frame>` ne correspond à aucune frame portant ce nom, `target=" <nom de frame>` " aura le même résultat que `target="_blank"`.

Exemples :

Code HTML	Affichage HTML
<pre> Vers Mirianet.com dans une nouvelle fenêtre Haut de la page Renvoi au tutoriel html, page 3 section séparateurs </pre>	<p>Vers Mirianet.com dans une nouvelle fenêtre</p> <p>Haut de la page</p> <p>Renvoi au tutoriel html, page 3 section</p>

Cas particuliers

Liens pour l'envoi d'un message e-mail et la lecture d'unnewsgroup

Un tel lien lance automatiquement l'application de messagerie électronique en incluant lenom du destinataire.

Exemples :

Code HTML	Affichage HTML
<pre> Nous contacter Newsgroup sur HTML et Cie. </pre>	<p>Nous contacter</p> <p>Newsgroup sur HTML et Cie.</p>

6. Listes

Les listes permettent de structurer l'information, en numérotant les paragraphes, insérant des puces, mettant en retrait les sous-paragraphes

On en distingue de 3 types :

- [Liste à puces](#)
- [Liste numérotées](#)
- [Liste de définitions](#)
- [Imbrication de listes](#)

• Liste à puces

Syntaxe:

```
<ul>
    <li class="tuto">...</li>
    <li class="tuto">...</li>
</ul>
```

 indique le début de la liste

<li class="tuto"> sont les items de la liste

Exemple :

Code HTML	Affichage HTML
<pre>Exemple de liste à puce :
 <lh>Titre de la liste</lh> <li class="tuto">Elément 1 <li class="tuto">Elément 2 </pre>	<p>Exemple de liste à puce :</p> <p>Titre de la liste</p> <ul style="list-style-type: none">• Elément 1• Elément 2

[En haut](#)

• Liste numérotées

Syntaxe:

```
<ol>
    <li class="tuto">...</li>
```

```
<li class="tuto">...</li>
```

```
</ol>
```

`` indique le début de la liste

`<li class="tuto">` sont les items de la liste

Exemple :

Code HTML	Affichage HTML
<pre>Exemple de liste numérotée :
> <lh>Titre de la liste</lh> <li class="tuto">Elément 1 <li class="tuto">Elément 2 </pre>	Exemple de liste numérotée : Titre de la liste 1. Elément 1 2. Elément 2

[En haut](#)

● Liste de définitions

Syntaxe:

```
<dl>
  <dt>...</dt>
  <dd>...</dd>
  <dt>...</dt>
  <dd>...</dd>
</dl>
```

`<dl>` indique le début de la liste

`<dt>` introduit un nouveau terme de définition

`<dd>` introduit une description du terme de définition

Exemple :

Code HTML	Affichage HTML

<pre>Exemple de liste de définitions :
 <dl> <lh>Titre de la liste</lh> <dt>Terme 1</dt> <dd>Définition du terme 1</dd> <dt>Terme 2</dt> <dd>Définition du terme 2</dd> </dl></pre>	<p>Exemple de liste de définitions :</p> <p>Titre de la liste</p> <p>Terme 1</p> <p style="padding-left: 40px;">Définition du terme 1</p> <p>Terme 2</p> <p style="padding-left: 40px;">Définition du terme 2</p>
--	---

[En haut](#)

● **Imbrication de listes**

Il est bien sur possible d'imbriquer plusieurs listes, ce qui donne une arborescence sur plusieurs niveaux.

Exemple :

Code HTML	Affichage HTML
<pre>Exemple de listes imbriquées :
 <li class="tuto">Chapitre 1 <li class="tuto">Chapritre 1.1 <li class="tuto">Chapritre 1.2 <li class="tuto">Chapritre 1.3 <li class="tuto">Chapritre 2.1 <li class="tuto">Chapritre 2.2 </pre>	<p>Exemple de listes imbriquées :</p> <p>1. Chapitre 1</p> <ul style="list-style-type: none"> ○ Chapritre 1.1 ○ Chapritre 1.2 ○ Chapritre 1.3 <p>2. Chapitre 21</p> <ul style="list-style-type: none"> ○ Chapritre 2.1 ○ Chapritre 2.2

7. Images

Les formats d'image affichables par les navigateurs sont *.gif et *.jpg (*.png pour les récents)
Pour avoir des images à l'intérieur d'un document HTML, on utilise la balise html "img":

Syntaxe:

```

ou
</img>
```

Paramètres:

- Le paramètre "src" est obligatoire et contient une URL ou un chemin relatif vers un fichier de format .gif ou .jpg ou .png

Syntaxe: src="fichier.gif" l'adresse peut être relative ou absolue

```
 Chemin relatif
 URL
```

Exemple avec une URL :

Code HTML	Affichage HTML
<pre></pre>	

- Le paramètre "align" permet de spécifier l'alignement de l'image par rapport au texte. Les valeurs possibles sont : top, bottom, middle, right, left

Syntaxe: ALIGN = "TOP ou BOTTOM ou MIDDLE ou RIGHT ou LEFT"

```

```

- Le paramètre "alt" contient le commentaire que les personnes utilisant un navigateur textuel (sans images) voient à la place de l'image.

- Les paramètres "width" et "height" se réfèrent à la largeur et à la hauteur de l'image (en pixels). Utilisez cette procédure dans tous les cas, car cela permet de commencer à lire le texte avant que l'image ne soit chargée complètement.



Syntaxe: WIDTH="n" HEIGHT="m" où n et m sont un nombre de pixels

Exemples :

Code HTML	Affichage HTML
<pre> </pre>	

Important :

- Dans le cas où l'on ne connaîtrait pas par avance les dimensions de l'image, il vaut mieux ne pas spécifier les valeurs de "width" et "height", dans ce cas là l'image prendra sa taille optimale
- Si l'on ne respecte pas le rapport des tailles on risque d'avoir une image déformée
- On peut, par contre, ne spécifier qu'une seule dimension, l'autre sera calculée de façon proportionnelle par le navigateur

Code HTML	Affichage HTML
<p>Ici on ne respecte pas le rapport des tailles ...</p> <pre></pre> <p>Là on crée une image de type vignette</p> <pre></pre>	<p>Ici on ne respecte pas le rapport des tailles ...</p>  <p>Là on crée une image de type vignette</p> 

- Le paramètre "hspace" permet de spécifier la distance horizontale en pixels entre le texte environnant et le bord de l'image.

Syntaxe: HSPACE="j" où j est un nombre de pixels



```

```

- Le paramètre "vspace" permet de spécifier la distance verticale en pixels entre l'image et le texte qui l'entoure.

Syntaxe: VSPACE="k" où k est un nombre de pixels

Exemple :

Code HTML	Affichage HTML
<p>Du texte là</p> <pre></pre> <p>Du texte ici</p> <pre></pre>	<p>Du texte là</p>  <p>Du texte ici</p> 

- Le paramètre "border" permet de créer un cadre autour de l'image. La largeur de la bordure est exprimée en pixels.

Syntaxe: BORDER="l" où l est un nombre de pixels

Exemple :

Code HTML	Affichage HTML
-----------	----------------

```

```



8. Images cliquables

Une image cliquable permet de présenter des menus graphiques en plus des liens textuels.

Définir une image cliquable revient à définir des zones sensibles (rectangulaires, circulaires...) et des actions correspondant à chacune d'entre elles.

Il existe plusieurs façons de traiter les actions de l'utilisateur sur une image:

- Le traitement nécessaire est effectué par le client.
- Le traitement nécessaire est une routine du côté du serveur (rarement utilisé nous ne le verrons pas ici)

Principe:

-La définition des zones sensibles de l'image se trouve dans la page HTML.

-C'est le client (e.g. Internet Explorer, Firefox ...) qui se charge de gérer les clicks de l'utilisateur

-La définition d'une image cliquable comporte deux parties:

- 1 La définition des zones et des actions à exécuter (MAP & AREA)
- 2 L'image et son lien vers la définition de zones (USEMAP)

• Le marqueur MAP

Le marqueur "MAP" sert à définir le début et la fin d'une section de définition de zones sensibles.

Paramètre de MAP

Le paramètre NAME sert à définir un nom symbolique pour la définition de zones cliquables

Syntaxe: NAME="nom_de_la_map"

```
<MAP name="fleur" > ... </MAP>
```

• Les marqueur AREA

Le marqueur "AREA" sert à définir des zones à l'intérieur d'une "MAP".

Les paramètres de AREA sont tous obligatoires:

SHAPE

correspond à la forme de la zone sensible.

Syntaxe: SHAPE = "RECT ou POLY ou CIRCLE"

```
<AREA SHAPE="rect" >
```

COORDS

Indique les coordonnées en pixel pour définir la taille des zones.

Le coin en haut à gauche de l'image est le point d'origine.

- Pour les rectangles:

les coordonnées à spécifier sont le coin en haut à gauche et celui en bas à droite.

Syntaxe: COORDS="n,m,i,j"

```
<AREA SHAPE="rect" COORDS="16,17,249,77" >
```

- Pour les cercles:

les coordonnées du centre et la valeur du rayon sont nécessaires (3 nombres).

Syntaxe: COORDS="l,k,p"

```
<AREA SHAPE="circle" COORDS="16,249,10">
```

- Pour les polygones:

on donne les coordonnées pour chaque point

Syntaxe: COORDS="x1,y1,x2,y2,x3,y3,....."

```
<AREA SHAPE="poly" COORDS="10,10, 10,15, 15,15, ....">
```

Remarque:

Pour définir les zones réactives, il est fortement conseillé d'utiliser un éditeur de type [WYSIWYG](#)

HREF

Contient l'URL qui est chargé lorsque l'utilisateur clique sur une des zones sensibles.

Syntaxe: HREF="URL" où URL est une adresse WWW

```
<AREA HREF="http://www.mirianet.com">
```

USEMAP

Le paramètre USEMAP se met à l'intérieur d'une commande IMG et indique au client que l'image en question est cliquable.

La valeur du paramètre USEMAP contient l'adresse d'une section MAP qui peut se trouver dans le même document (auquel cas on le déclare en commençant par #, e.g. "#nom_carte") ou dans un autre document ("URL#nom_de_carte").

Syntaxe:


```

```

Exemple:

Dans l'exemple ci-dessous, essayer de cliquer tour à tour :

Au centre de l'image, en haut à gauche de l'image et en bas à droite de l'image

Code HTML	Affichage HTML
<pre> <map name="FleurMap"> <area shape="poly" coords="55,49,55,25,82,18,85,26, 95,32,108,46,111,62,104,81,94,93,81,87,79,98,63,97, 53,89,51,78,31,64,44,40" href="javascript:alert('Fleur du centre');"> <area shape="circle" coords="137,101,17" href="javascript:alert('Cercle bas droite');"> <area shape="rect" coords="2,2,32,23" href="javascript:alert('Rectangle haut gauche');"> </map></pre>	

9. Les tables ou tableaux

Les tableaux sont des balises HTML complexes, permettant de présenter de façon uniforme une liste de valeurs, ou même de structurer un contenu html. Avec l'évolution des CSS (Cascading Style Sheet), on peut désormais structurer un document html de façon très fiable (et modulable) en appliquant des styles à des balises de type *DIV* (calque), cependant le tableau reste un élément incontournable du langage.

Syntaxe:

```
<table border="n" align="alignement" width="n"
cellpadding="n" cellspacing="n" bgcolor="CodeCouleur">
...
</table>
```

Note : Aucun paramètre n'est obligatoire.

- Le paramètre "align" permet de spécifier l'alignement horizontal du tableau par rapport à son conteneur. Les valeurs possibles sont : center, right, left
- Le paramètre "border" défini en pixels, l'épaisseur de la bordure du tableau

Exemple:

```
<table border="2">
...
</table>
```

- Le paramètre "width" détermine la largeur du tableau qui peut être relative (%) ou en pixel

Exemple:

```
<table width="100%">
...
</table>
<table width="450">
...
</table>
```

- Le paramètre "cellpadding" détermine la marge intérieure des cellules du tableau (pixels)

Exemple:

```
<table cellpadding="5">
...
</table>
```

- Le paramètre "cellspacing" détermine l'espacement entre les cellules du tableau (pixels)

Exemple:

```
<table cellspacing="10">
...
</table>
```

- Le paramètre "bgcolor" détermine la couleur de fond du tableau (code couleur en hexadécimal)

Exemple:

```
Pour une couleur de fond rouge :
<table bgcolor="#FF0000">
```

```
...  
</table>
```

Tout seul un tableau n'a aucune existence réelle, en fait un tableau se comporte exactement comme une grille ou matrice, il est composé de lignes (ou rangées) *TR* qui à leur tour contiennent des cellules *TD*

Les rangées (*TR*)

Ce sont donc les lignes horizontales du tableau

Syntaxe:

```
<tr width="n" bgcolor="CodeCouleur">...</tr>  
</table>
```

Les paramètres "width" et "bgcolor" ont le même rôle que pour les tableaux

Exemple:

```
<tr width="100" bgcolor="#0000FF">...</tr>  
</table>
```

Les cellules (*TD* ou *TH*)

Elles sont contenues dans les rangées *TR* et constituent le bout de la chaîne d'un tableau html (même si une cellule peut contenir à son tour d'autres balises html et même des tableaux)

On distingue les cellules de type *TH*, qui constituent les cellules des premières lignes des tableaux ("H" comme "Header"), et les autres cellules de type *TD* (l'emploi des cellules *TH* pour les premières lignes n'est pas obligatoire, mais peut se révéler utile pour donner un aspect différent aux premières lignes d'un tableau)

Syntaxe:

```
<th align="alignement" valign="alignement"  
colspan="n" rowspan="n" width="n" bgcolor="CodeCouleur">  
...  
</th>
```

ou

```
<td align="alignement" valign="alignement"  
colspan="n" rowspan="n" width="n" bgcolor="CodeCouleur">  
...  
</td>
```

- Les paramètres "width" et "bgcolor" ont le même rôle que pour les tableaux
- Le paramètre "valign" détermine l'alignement vertical du contenu de la cellule

Les valeurs possibles sont : middle, top, bottom

- Le paramètre "colspan" détermine l'étendue horizontale de la cellule en nombre de colonnes

Exemple:

```
Une cellule qui s'étend sur 3 colonnes  
<td colspan="3">...</td>
```

- Le paramètre "rowspan" détermine l'étendue verticale de la cellule en nombre de lignes

Exemple:

```
Une cellule qui s'étend sur 2 lignes
```

<td rowspan="2">...</td>

Exemples:

Code HTML	Affichage HTML				
<pre>Un tableau simple <table border="1"> <tr> <td>A</td> <td>B</td> </tr> <tr> <td>C</td> <td>D</td> </tr> </table></pre>	<p>Un tableau simple</p> <table border="1"><tr><td>A</td><td>B</td></tr><tr><td>C</td><td>D</td></tr></table>	A	B	C	D
A	B				
C	D				

Code HTML	Affichage HTML				
<pre>Un tableau avec quelques propriétés et une première ligne qui s'étend sur deux colonnes <table width="150" cellpadding="2" cellspacing="5" border="1"> <tr> <td bgcolor="#FFF000" align="center" colspan="2">Titre</ td> </tr> <tr> <td>A</td> <td>B</td> </tr> </table></pre>	<p>Un tableau avec quelques propriétés et une première ligne qui s'étend sur deux colonnes</p> <table border="1"><tr><td colspan="2">Titre</td></tr><tr><td>A</td><td>B</td></tr></table>	Titre		A	B
Titre					
A	B				

Code HTML	Affichage HTML		
<pre>Un autre exemple <table width="80" cellpadding="2" cellspacing="5" border="1"> <tr> <td bgcolor="#FFF000" align="center" colspan="2">Titre</ td> </tr> <tr> <td rowspan="2">Gauche</td> <td>Droite haut</td> </tr> <tr> <td>Droite bas</td></pre>	<p>Un autre exemple</p> <table border="1"><tr><td colspan="2">Titre</td></tr></table>	Titre	
Titre			

```
</tr>  
</table>
```

Gauche	Droite haut
	Droite bas