

Chapitre I :

Le langage Html

Section 1 : Introduction

HTML est le langage universel utilisé pour communiquer sur le Web. Votre information sera ainsi transportée sur cette gigantesque toile de réseaux interconnectés qu'est Internet, pour aboutir sur l'ordinateur de votre lecteur grâce à un programme appelé navigateur ou browser.

- Vous avez donc deux interlocuteurs :
 1. le browser de votre lecteur
 2. et votre lecteur lui-même.

Ce logiciel, que l'on appelle un browser, vous permet de surfer sur le Net et d'afficher sur votre écran les "pages" qu'il a interceptées. Il y a, hélas, beaucoup de marques et de types de browsers différents. Des simples, des archaïques ou des sophistiqués... Les plus connus sont Netscape dans sa version 2, 3 et 4 ainsi Internet Explorer de Microsoft mais il en existe beaucoup d'autres.

- Chaque browser a sa propre façon de travailler.

A la différence de votre traitement de texte préféré qui restitue exactement votre document sur une feuille de papier avec votre police de caractères et votre mise en page, vous ne saurez jamais exactement ce que le browser de votre lecteur du bout du monde affichera sur l'écran de celui-ci.

- En HTML, vous n'avez pas la maîtrise totale de votre document.

Pour transiter le plus rapidement possible sur les lignes téléphoniques, on a adopté un format de texte très compact mais aussi (par conséquence) peu sophistiqué. C'est le bon vieux format de texte pur et dur, sans fioritures du Bloc-notes ou Notepad de Windows par exemple. Et de plus ce format ASCII a été amputé d'un bit (7 bits au lieu de 8) ! Vous serez donc privé de certains caractères spéciaux comme le é pour lesquels il faudra passer par des codes particuliers.

- Mais récompense suprême... Html est un langage universel qui s'adapte à toutes les plate-formes que ce soit Windows, Macintosh, Unix, OS/2...

En plus du texte adressé à votre lecteur, il vous faudra inclure des instructions pour le browser de celui-ci. Ces instructions seront différenciées de votre texte par les signes < et > par exemple <html>. Ces "instructions" s'appellent des tags ou des balises.

Quand vous écrirez les balises de votre page HTML, il faudra garder à l'esprit :

- qu'une balise marque une action pour le browser (ce qu'il doit faire...).
- que les attributs précisent les modalités de cette action (comment il doit le faire...).

Section 2 : Vos premiers outils

2.1 Vous avez besoin:

- d'un éditeur de texte tout simple comme par exemple le Bloc-notes ou Notepad de Windows ou tout autre équivalent dans votre système d'exploitation.
- d'un browser soit Netscape Navigator (payant) que vous pouvez cependant télécharger gratuitement au site de Netscape (<http://home.netscape.com>) ou sur les CD de certains mensuels informatiques soit Microsoft Explorer (gratuit également) à télécharger (<http://www.microsoft.com>) ou dans les revues.

2.2 Vous n'avez pas besoin:

- d'être connecté pour écrire, voir et peaufiner vos pages Html.
- d'avoir le dernier éditeur Html sans doute performant mais coûteux qu'il sera toujours temps d'adopter lorsque vous aurez usé vos petits doigts sur vos premières pages. Nous pensons à Claris Home Page, Frontpage de Microsoft, Dreamweaver, Hotdog, Hotmetal, WebExpert ou Netscape Editor ainsi qu'aux dizaines d'autres éditeurs que vous rencontrerez sur le Web.

2.3 Conseils

Le langage Html étant un ensemble de balises et d'attributs, il nous paraît utile sinon indispensable de les passer en revue et surtout de les visionner au moins une fois car :

- si les éditeurs Html vous faciliteront grandement la tâche, ils ne sont pas toujours parfaits surtout lors des modifications, annulations ou suppressions en cours de travail. Il vous faudra bien alors vous plonger dans le code source pour corriger les dysfonctionnements.
- les codes source de vos pages préférées sont disponibles (et sans copyright). Il est alors possible de s'en inspirer pour reprendre le procédé sans avoir à réinventer le monde.
- ces mêmes éditeurs Html vous proposeront des termes comme " En-tête, Heading, Cell spacing, Numered List..." qui sont propres au langage Html.
- vous aurez besoin d'une connaissance pointue du Html pour inclure les codes du Javascript ou du VBscript dans vos pages.

Section 3 : Le document Html minimum

3.1 Voici vos premières balises ou tags :

<HTML>	Ceci est le début d'un document de type HTML.
</HTML>	Ceci est la fin d'un document de type HTML.
<HEAD>	Ceci est le début de la zone d'en-tête (prologue au document proprement dit contenant des informations destinées au browser).
</HEAD>	Ceci est la fin de la zone d'en-tête.
<TITLE>	Ceci est le début du titre de la page.
</TITLE>	Ceci est la fin du titre de la page.
<BODY>	Ceci est le début du document proprement dit.
</ BODY>	Ceci est la fin du document proprement dit.

- Vous aurez remarqué qu'à chaque balise de début d'une action, soit <...>, correspond (en toute logique) une balise de fin d'une action </...>.
- Vous noterez aussi que les balises ne sont pas "case sensitive". Il est donc équivalent d'écrire <HTML>, <html>, <Html>, etc.

3.2 Faisons ensemble notre premier document Html:

- Ouvrir l'éditeur de texte.

- Ecrire les codes Html suivants:

```
<HTML>
<HEAD>
<TITLE>Document Html minimum</TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

- Enregistrer le document avec l'extension .html ou .htm.

- Ouvrir le navigateur.

- Afficher le document via le menu File/Open file...

- Admirez votre premier document Html.



Celui-ci est vide (et c'est normal) mais tout a fait opérationnel! Il faudra maintenant lui fournir votre information à l'intérieur des balises <BODY></BODY>. Remarquez que votre "TITLE" est présent dans la fenêtre de Netscape.

Pour vos éventuelles modifications, il n'est pas nécessaire de rouvrir à chaque fois le navigateur.

- Retourner dans l'éditeur de texte (sans fermer le navigateur).
- Modifier les codes Html.
- Enregistrer le fichier.
- Utiliser la commande Reload du browser ou si celui-ci est paresseux cliquer dans la barre "Location" et faire "Enter".

Section 4 : Le texte

Tout document Html contiendra en majorité du texte. Voyons comment l'agréments par quelques balises élémentaires.

Gras	[Bold]	Début et fin de zone en gras
Italique	[Italic]	<I>...</I> ...	Début et fin de zone en italique
Taille de caractère	[Font size]	...	Début et fin de zone avec cette taille
Couleur de caractère	[Font color]	 	Début et fin de zone en couleur
A la ligne	[Line break]	 	Aller à la ligne
Commentaires	[Comments]	<!-- *** -->	Ne pas afficher
Centrage	[Center]	<CENTER></CENTER>	Centrer

Ouvrir l'éditeur de texte

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>texte simple<BR>
<B>texte en gras</B><BR>
<STRONG>texte en gras</STRONG><BR>
<I>texte en italique</I><BR>
<EM>texte en italique</EM><BR>
<B><I>texte en gras et en italique</I></B><BR>
<FONT SIZE=5>texte</FONT>
<FONT COLOR="#0000FF">en bleu</FONT>
<!--C'est fini-->
</BODY>
</HTML>
```

- Ouvrir le browser



Quelques commentaires s'imposent;

- Le texte tout simple s'écrit sans balises. Il sera repris par le browser avec la police et taille de caractères choisies dans sa configuration par défaut.
- Le browser affiche le texte qu'on lui "dicte" en passant à la ligne lorsque celui-ci atteint le bord de la fenêtre. Pour le forcer à passer outre à cette règle de conduite et à faire un saut à la ligne comme vous le souhaitez, il faut une instruction particulière. C'est la balise
. Celle-ci représente une action ponctuelle et n'a donc pas besoin de balise de fin.
- Le même browser ne tient compte que d'un seul espace entre les mots. Ainsi pour lui texte est équivalent à texteen bleu
- Il n'est pas rare d'utiliser plusieurs balises pour un même élément de texte. Il faut veiller à bien les imbriquer. Ainsi <I>...</I> est correct et <I>...</I> risque de vous créer des ennuis.
- La taille dans peut être indiquée de deux façons :
 1. avec un nombre de 1 à 7. La valeur par défaut étant 3.
 2. de façon relative par rapport à la valeur par défaut (ici 0). Soit -3 -2 -1 0 +1 +2 +3.
- Pour les puristes, les balises <I> et , et ne sont pas totalement équivalentes. Vous verrez en poussant plus avant votre étude du langage Html que et appartiennent aux définitions structurales (style logique) dont l'apparence dépend des options reprises dans la configuration du browser. Les balises et <I> appartiennent aux formats de présentation (style physique) dont l'apparence dépend de la volonté de l'auteur.

Voici les codes de quelques couleurs basiques.

Bleu	#0000FF	Vert	#00FF00
Blanc	#FFFFFF	Violet	#8000FF
Rouge	#FF0000	Jaune	#FFFF00
Gris clair	#C0C0C0	Noir	#000000

Section 5 : Du texte encore...

Des balises de texte, il y en a plein des valises! Selon les versions du HTML (nous en sommes à la version HTML 3.2) et l'apparition des éditeurs Html évolués, certains tags sont moins utilisés. D'autres aussi sont tout bonnement d'un emploi rare ou pour le moins particulier.

En voici quelques-uns qui compléteront votre panoplie actuelle.

- Le tag <BLOCKQUOTE>...</BLOCKQUOTE> introduit une citation. Son utilité serait douteuse si le texte de cette citation n'était introduit avec un léger retrait à gauche et à droite. Ce qui est bien pratique pour agréments la présentation.
- Voici un texte normal sans blockquote.

Voici un texte avec blockquote. Remarquez le retrait.

Pour la petite histoire ce tag n'était pas prévu dans Netscape Editor de la version Gold 3.0 mais il a été ajouté à la version Netscape Communicator 4.0. Le même effet peut être réalisé par un tableau ou un tag de liste vide.

- Vous savez déjà que les browsers ne reconnaissent qu'un espace entre les mots. Ce qui peut se révéler gênant dans certaines situations.
La balise `<PRE>...</PRE>` affiche un texte dit préformaté. Le browser prend ainsi en compte tous les espaces et sauts de ligne définis à l'écran. **P r a t i q u e**

Avant que les tableaux ne soient reconnus par les browsers, les braves pionniers du Html devaient employer ce tag pour faire des tableaux.

- Le tag `<ADDRESS>...</ADDRESS>` pour indiquer une adresse (généralement en fin de document).
Voici mon adresse avec cette balise :
Van Lancker Luc
Rue des Brasseurs, 22
7700 Mouscron (Belgium)
- La balise `<U>...</U>` souligne le texte.
Comme, par convention, les éléments servant d'hyperlien sont soulignés (c'est le même que dans les fichiers d'aide), on évitera le souligner des éléments de texte pour lui donner de l'importance. On préférera la mettre en gras ou dans un format ou une couleur de police différent. Il ne faut donc pas en abuser.
- En parler ou ne pas en parler, voilà la question! Netscape 3.0 et plus (et pas Microsoft Explorer) permet d'afficher un texte clignotant par la balise `<BLINK>...</BLINK>`. Ce tag est un peu tourné en dérision car il est rapidement ennuyeux. On lui préférera une image animée qui peut être débranchée après quelques clignotements.
Particularité Netscape! (uniquement)
- Les tags `_{...}` et `^{...}` placent le texte respectivement en indice et en exposant.
Ainsi, `_{escalier}escalier^{escalier}` aura comme résultat

escalier
 escalier
 escalier
- Pour aligner du texte, on a utilisé l'attribut ALIGN ou le tag `<CENTER>`. Il existe une balise permettant d'aligner différents éléments. c'est le tag :

```
<DIV align=left>...</DIV>
<DIV align=center>...</DIV>
<DIV align=right>...</DIV>
```

Cette alternative pour l'alignement horizontal du texte est parfois bien utile pour définir l'alignement d'une portion de texte. Il nous est très utile pour parer à certaines "paresse" de Microsoft Explorer 3.0 par exemple.
- Il existe encore bien d'autres balises que je vous invite à découvrir dans vos études suivantes.

Section 6 : Les titres et les listes

Tout document d'une certaine consistance se doit de présenter, par exemple dans la table des matières, les différents niveaux de son exposé. Html a disposé dès son origine d'outils spécialement conçus à cet effet.

En-têtes	[Heading]	<code><Hn></Hn></code> avec n=1 à 6	Afficher une en-tête de niveau n et sauter une ligne
Liste non-ordonnée	[Bullet list]	<code></code>	Afficher le texte sous forme d'une liste non-ordonnée.
Liste ordonnée	[Numbered list]	<code></code>	Afficher le texte sous forme d'une liste ordonnée.
Elément de liste	[List items]	<code></code>	Voici un élément de la liste

Paragraphe	[Paragraph]	<code><P></P></code>	Saut de ligne, insérer une ligne vierge et commencer un paragraphe
------------	-------------	----------------------------------	--

Comme d'habitude...

- On ouvre son petit éditeur de texte

```
<H1>Les mois du printemps</H1>
<UL><LI>avril
<LI>mai
<LI>juin</UL>
<P>
<H3>Les mois d'automne</H3>
<OL><LI>octobre
<LI>novembre
<LI>décembre</OL>
```

- Et on ouvre son petit browser

Les mois du printemps

- avril
- mai
- juin

Les mois d'automne

- octobre
- novembre
- décembre

- Quelques commentaires s'imposent;

- `é` est l'un de ces caractères spéciaux utilisés pour représenter le é. Les éditeurs Html vous déchargeront de cette corvée.
- Les paragraphes et les en-têtes peuvent s'aligner à gauche, au centre ou à droite.

```
<H1 align=left></H1>
<P align=left></P>
<H1 align=center></H1>
<P align=center></P>
<H1 align=right></H1>
<P align=right></P>
```
- Les listes peuvent s'imbriquer:

```
<H4>Les 12 mois</H4>
<UL><LI>Les mois du printemps
<OL><LI>avril
</OL></LI></UL>
```

Les 12 mois

 - Les mois du printemps
 - avril

Section 7 : Les liens

Html (Hyper Text Markup Language) est un langage hypertexte (et hypergraphique) qui vous permet en cliquant sur un mot, généralement souligné (ou une image) de vous transporter;

- vers un autre endroit du document.
- vers un autre fichier Html situé sur votre ordinateur.
- vers un autre ordinateur situé sur le Web.

Ce système d'hypertexte vous est familier car il est également utilisé par les fichiers d'aide de Windows. Ce sont ces liens qui vous permettent de surfer de page en page et qui constituent l'essence des documents Html.

La syntaxe de ces liens entre plusieurs pages, est simple mais entraînera de nombreux commentaires :

...

7.1 Lien externe

Tout ordinateur situé sur le réseau Internet possède une adresse ou une URL (Universal Ressource Locator). Html permet d'accéder à toutes les machines et toutes les ressources du Net. Pour peu qu'Internet vous soit un peu familier, ce sont les adresses du type :

http://serveur/chemin.../fichier
ftp://serveur/chemin.../fichier
mailto:utilisateur@hôte

7.2 Lien local

L'organisation classique, et plus que conseillée, d'un site Web consiste à regrouper l'ensemble des éléments de celui-ci (fichiers html, images, ...) dans un même répertoire. Vous pourrez ainsi "transporter" aisément votre site pour le présenter sur un autre ordinateur et, but ultime, le charger sur un serveur. Cette façon de procéder est la plus aisée et vous évitera pas mal de problèmes. L'adresse du lien sera alors tout simplement :

fichier.htm

7.3 Lien mixte

Nous entendons par là un lien vers un fichier situé à un autre endroit de votre ordinateur (et donc non situé dans le répertoire de votre site). Attention Danger! En effet, il est peu probable que le serveur Web qui hébergera votre site, possède la même arborescence que votre disque local. L'adresse prendra la forme

file:///lecteur:/répertoire/fichier.htm (en adressage absolu).
.././../fichier.htm (en adressage relatif).

Cette matière d'adressage absolu et relatif dépasse le cadre de cet exposé et je vous invite en cas d'utilisation à vous plonger dans la documentation relative à ce sujet.

7.4 Expérimentons tout ceci.

- Dans l'éditeur de texte, nous allons créer deux fichiers Html.

- le fichier 1.htm:
Aller vers le document 2
- le fichier 2.htm:
Retour au document 1

- On sauve ces deux fichiers dans un même répertoire.



- Le browser vous affichera :



7.5 Les ancrés

Des liens peuvent aussi pointer vers un endroit précis du même document ou d'un autre fichier. C'est ce qu'on appelle les ancrés, ancrages ou pointeurs [Anchor].

Point d'ancrage	...	Ceci est une cible
Lien vers une ancre dans la même page	...	Lien vers la cible *** dans la même page
Lien vers une ancre dans une autre page	...	Lien vers la cible *** dans une autre page

Plusieurs liens à l'intérieur d'un même document supposent que ce document présente une certaine longueur sinon une longueur certaine (et donc un temps de chargement assez long). Ainsi, on préférera généralement à cette technique le découpage d'un long page en un ensemble de plusieurs pages de dimension plus réduite.

Section 8 : Les images

8.1 GIF ou JPEG?

Avant de passer aux balises, il faut savoir ce qui suit.

- Les deux formats d'image (maximum 256 couleurs) reconnus sur le Web, sont le format GIF (version 89a) et le format JPEG. Pour le format GIF, on retiendra la caractéristique "entrelacé" qui permet de charger progressivement l'image lors de l'ouverture de la page. Ces deux formats donnent des résultats assez équivalents bien que JPG soit plutôt recommandé pour des images avec des tons nuancés ou dégradés.
- L'ennemi sur le Web, c'est la taille des images! Plus l'image sera grande, plus le temps de chargement sera long... au risque de décourager vos visiteurs. Si cela est possible, une image en 16couleurs peut très bien faire l'affaire. Présenter une petite image indiquant un lien vers l'image complète est également un bon conseil.
- Il n'est pas inutile de prévoir dans votre panoplie de compositeur Web, un logiciel de retouche d'images. A ce propos "Paint Shop Pro" a la triple qualité d'être disponible en shareware, d'être très intuitif et d'être performant. En outre, un tel logiciel vous permettra de composer vos propres images et ainsi de personnaliser vos pages.
- Dois-je rappeler qu'il est très facile de se faire une bibliothèque d'images tout en surfant sur le Web? Avec Netscape, il suffit de se positionner sur l'image, de cliquer avec le bouton droit de la souris et de suivre les instructions du menu déroulant (Save Image As...).

8.2 Le code Html est :

 Afficher l'image qui se trouve à l'adresse...

La balise image possède de nombreux attributs.

Texte alternatif alt="****"	Pour les browser n'ayant pas l'option "image" activée
Dimensions width=? height=?	Hauteur et largeur (en pixels)
border=? (en pixels)	Bordure
align=top	Alignement
align=middle	
align=bottom	

align=left
align=right

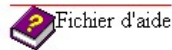
8.3 Commentaires :

- En Html, l'image ne fait pas partie de votre document. Le browser va la chercher à l'adresse indiquée. Généralement, on place les images dans le même répertoire que les pages Html.
- Presque en conséquence logique de ceci, le fait d'utiliser la même image à plusieurs reprises dans un fichier Html ne modifie en rien sa taille.
- Prévoir un texte pour les browsers n'ayant pas l'option image activée, permet au lecteur de ne pas perdre le fil de l'exposé et peut-être d'activer cette option pour découvrir votre oeuvre.
- Il est important pour la fluidité de l'affichage de préciser la taille en hauteur et largeur de l'image car le browser peut ainsi connaissant l'emplacement à réserver pour celle-ci, continuer à afficher le texte. Cette information vous est donnée par Paint Shop Pro.

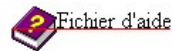
8.4 L'attribut Align

L'attribut align positionne l'image par rapport au texte

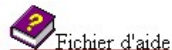
Fichier d'aide



Fichier d'aide



Fichier d'aide



8.5 Lien sur image

Les balises sont :

Remarquons que les images cliquables sont entourées d'une bordure.

Section 9 : Les séparateurs

Les browsers intègrent un outil de mise en forme intéressant pour clarifier la présentation de votre texte. C'est la ligne horizontale.

La syntaxe en est fort simple:

Ligne horizontale [Horizontal Rule] <HR> Insérer une ligne horizontale

Ce tag fort simple, nous permettra de comprendre un peu mieux le fonctionnement du navigateur. D'abord, vous avez déjà découvert au fil de vos erreurs qu'il est plutôt gentil car il s'efforce toujours d'afficher quelque chose. Ensuite, il vous ménage le travail en prenant à sa charge des valeurs par défaut d'un certain nombre d'attributs. Dans le cas présent, les valeurs par défaut de la balise <HR> sont : une épaisseur de trait de 2 pixels, un alignement centré et une largeur de 100% de la fenêtre. Vous pouvez très bien modifier au gré de votre fantaisie les valeurs de ces différents attributs.

<HR size=?>	Epaisseur en pixels
<HR width=?>	Largeur en pixels
<HR width="%">	Largeur en % de la fenêtre
<HR align=left>	Alignement gauche
<HR align=right>	Alignement droite
<HR align=center>	Alignement centré

Il est donc tout à fait équivalent d'écrire:

- <HR> ou

- <HR size=2 align=center width="100%">

Je vous propose un petit exercice:

- Avec l'éditeur de texte
<H3 align=center>Riadh BOUSLIMI</H3>
<HR align=center size=8 width="50%">
- Avec le browser

Riadh BOUSLIMI



- Commentaires :
 - On préférera spécifier la largeur en % de la fenêtre plutôt qu'en pixels car l'affichage s'adaptera ainsi à toutes les tailles et résolutions d'écran.
 - On peut préférer l'usage d'images comme séparateur (comme c'est le cas ici) pour personnaliser son ouvrage.

Section 10 : Les arrière-plans

Le langage Html permet d'agrémenter la présentation du document d'un arrière-plan [background] coloré ou composé d'une image. Ce qui apporte un élément "artistique" à votre page.

La balise à utiliser ne pose pas de problème :

Couleur d'arrière-plan <BODY BGCOLOR="#\$\$\$\$\$">

Donc, pas de problèmes! On fait un essai...

```
<BODY BGCOLOR="#000088">
<H1>Bonjour</H1>
</BODY>
```



Joli! Mais la lisibilité n'est pas parfaite. Heureusement, des balises sont prévues pour modifier les couleurs utilisées par défaut par le browser pour le texte et les liens.

Couleur de texte	<BODY TEXT="#\$\$\$\$\$">
Couleur de lien	<BODY LINK="#\$\$\$\$\$">
Lien visité	<BODY VLINK="#\$\$\$\$\$">
Lien actif	<BODY ALINK="#\$\$\$\$\$">

Reprenons notre exemple;

```
<BODY BGCOLOR="#000088" TEXT="#FFFF00">
<H1>Bonjour</H1>
</BODY>
```



On peut aussi prévoir un fond en image. Cette image (petite de préférence) est affichée en mosaïque par le browser. Attention aux raccords... comme lorsque vous posez du papier peint.

Texture d'arrière-plan <BODY BACKGROUND="Adresse">

Ce qui peut nous donner :



```
<BODY BACKGROUND="PAPER.gif">
<H1>Bonjour</H1>
</BODY>
```

Section 11 : Les tableaux

En Html, les tableaux servent non seulement à aligner des chiffres mais surtout à placer des éléments à l'emplacement que vous souhaitez. L'usage des tableaux est donc très fréquent.

Un tableau est composé de lignes et de colonnes qui forment les cellules du tableau. Les balises de base sont donc :

Définition du tableau	[Table]	<TABLE></TABLE>	Début et fin de tableau
Définition d'une ligne	[Table Row]	<TR></TR>	Début et fin de ligne
Définition d'une cellule	[Table Data]	<TD></TD>	Début et fin de cellule

11.1 Un tableau à deux lignes et deux colonnes, et donc à quatre cellules se représente comme suit :

```
<TABLE>
<TR><TD>1</TD><TD>2</TD></TR>
<TR><TD>3</TD><TD>4</TD></TR>
</TABLE>
```



11.2 Si vous souhaitez y adjoindre des bordures :

Bordure de cadre [Border] <TABLE border=?></TABLE>

```
<TABLE border=2>
<TR><TD>1</TD><TD>2</TD></TR>
<TR><TD>3</TD><TD>4</TD></TR>
</TABLE>
```

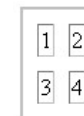


Il y a encore trois éléments (définis par défaut mais modifiables) :

L'espace entre les cellules ou l'épaisseur des lignes du quadrillage	<TABLE cellspacing=?>
L'enrobage des cellules ou l'espace entre le bord et le contenu	<TABLE cellpadding=?>
La largeur de la table	<TABLE width=?>
	<TABLE width=%>

Construisons un exemple :

```
<TABLE border=2 cellspacing=10>
<TR><TD>1</TD><TD>2</TD></TR>
<TR><TD>3</TD><TD>4</TD></TR>
</TABLE>
```



```
<TABLE border=2 cellpadding=10>
<TR><TD>1</TD><TD>2</TD></TR>
<TR><TD>3</TD><TD>4</TD></TR>
</TABLE>
```

1	2
3	4

Section 12 : Les cellules des tableaux

Nous n'en avons pas fini avec les tableaux. Allons plus loin dans notre étude en nous penchant sur les cellules de ces tableaux.

Avant toutes choses, les cellules peuvent contenir tous les éléments Html déjà passés en revue soit :

- du texte
- des images
- des liens
- des arrière-plans
- et même des tableaux (eh oui!)

Bien que l'allure de votre tableaux soit déjà déterminée, chaque cellule est en quelque sorte un petit univers à part qui a ses propres spécifications. Découvrons les balises.

Largeur d'une cellule	<TD width=?> en pixels
	<TD width=%> en pourcentage
Fusion de lignes	<TD rowspan=?>
Fusion de colonnes	<TD colspan=?>

Découvrons ceci par des exemples.

Je veux un tableau centré qui occupe 60% de la fenêtre avec sur une ligne, trois colonnes égales. Essayons ceci :

```
<CENTER><TABLE width=60% border=1>
<TR>
<TD>cellule1</TD>
<TD>cel. 2</TD>
<TD>3</TD>
</TR>
</TABLE></CENTER>
```

cellule 1	cel 2	3
-----------	-------	---

Pas brillant! Et avec la balise de largeur de la cellule?

```
<CENTER><TABLE width=60% border=1>
<TR>
<TD width=33%>cellule1</TD>
<TD width=33%>cel. 2</TD>
<TD width=34%>3</TD>
</TR>
</TABLE></CENTER>
```

cellule 1	cel 2	3
-----------	-------	---

Impeccable! Prenons le même tableau mais avec 2 lignes.

```
<CENTER><TABLE width=60% border=1>
<TR><TD width=33%>cellule1</TD><TD width=33%>cel. 2</TD>
```

```
<TD width=34%>3</TD></TR>
<TR><TD width=33%>cellule1</TD><TD width=33%>cel. 2</TD>
<TD width=34%>3</TD></TR>
</TABLE></CENTER>
```

cellule 1	cel 2	3
cellule 1	cel 2	3

Je souhaite que la première ligne prenne toute la largeur de la ligne. La première cellule doit donc déborder sur 3 cellules horizontales.

```
<CENTER><TABLE width=60% border=1>
<TR>
<TD colspan=3>cellule 1</TD>
</TR>
<TR> <TD width=33%>cellule 1</TD> <TD width=33%>cel 2</TD>
<TD width=34%>3</TD> </TR>
</TABLE></CENTER>
```

cellule 1		
cellule 1	cel 2	3

Dans le même style, je souhaite que la première colonne prenne toute la hauteur de la colonne. La première cellule doit donc déborder sur 2 cellules verticales.

```
<CENTER><TABLE width=60% border=1>
<TR>
<TD width=33% rowspan=2>cellule 1</TD>
<TD width=33%>cel 2</TD>
<TD width=34%>3</TD>
</TR>
<TR>
<TD width=33%>cel 2</TD>
<TD width=34%>3</TD>
</TR>
</TABLE></CENTER>
```

cellule 1	cel 2	3
	cel 2	3

Maintenant, comme le disait un gardien de prison de mes connaissances, il est temps de mettre de l'ordre dans les cellules [Je n'ai pas pu résister...].

Ligne de tableau	<TR align=left/center/right>	horizontalement
	<TR valign=top/middle/bottom>	verticalement
Cellule de tableau	<TD align=left/center/right>	horizontalement
	<TD valign=top/middle/bottom>	verticalement

A titre d'exemple, reprenons le tableau suivant :

cellule 1		
cellule 1	cel 2	3

Avec quelques aménagements, il devient...

```
<CENTER><TABLE width=60% border=1>
<TR>
<TD colspan=3 align=center>Tarif au ...</TD>
</TR>
<TR>
<TD width=33% >Article 1</TD><TD width=33%>Ref 002 </TD>
<TD width=34% align=right>30fr</TD>
</TR>
</TABLE></CENTER>
```

Tarif au ...		
Article 1	Ref 002	30fr

Pour terminer cette leçon, il est aussi possible de changer la couleur de la cellule.

Couleur de la cellule <TD BGCOLOR="#\$\$\$\$\$">

Notre dernier exemple pourrait devenir :

```
<TD colspan=3 align=center bgcolor="#000088">
```

Tarif au ...		
Article 1	Ref 002	30fr

Section 13 : Les frames

Quoi de plus simple que les frames? Mais aussi quoi de plus délicat et dangereux (risque de plantage) que les frames? En outre, tous les browsers n'ont pas la possibilité de les afficher.

Pour diviser l'écran en plusieurs fenêtres, les balises sont peu nombreuses :

<u>Zone avec des fenêtres</u>	
<FRAMESET>	Début de zone avec des fenêtres
</FRAMESET>	Fin de zone avec des fenêtres
<u>Agencement des fenêtres</u>	
<FRAMESET ROWS="...">	Fenêtres horizontales
<FRAMESET COLS="...">	Fenêtres verticales

13.1 Il est impératif de travailler avec des exemples. Pour obtenir un agencement ainsi :

Il faut employer les balises suivantes :

```
<HTML>
<HEAD></HEAD>
<FRAMESET ROWS="30%,70%">
<FRAME>
<FRAME>
</FRAMESET>
</HTML>
```



Attention! <FRAMESET></FRAMESET> remplace <BODY></BODY>

L'attribut ROWS="hauteur1,hauteur2,...,hauteurN" définit la hauteur des différentes fenêtres en cas de division horizontale.

La hauteur s'exprime en pixels ou en %. Dans ce cas, on veillera à ce que le total soit égal à 100%;

13.2 Le même pour un agencement vertical

```
<FRAMESET COLS="30%,70%">
<FRAME>
<FRAME>
</FRAMESET>
```



L'attribut COLS="largeur1,largeur2,...,largeurN" définit la largeur des différentes fenêtres en cas de division verticale.

La largeur s'exprime en pixels ou en %. Dans ce cas, on veillera à ce que le total soit égal à 100%;

13.3 On peut mélanger les deux :

```
<FRAMESET ROWS="30%,70%">
<FRAME>
<FRAMESET COLS="30%,70%">
<FRAME>
<FRAME>
</FRAMESET>
</FRAMESET>
```




13.4 Pour l'instant, nos frames sont vides. On va donc leur donner à manger par des attributs de la balise :
`<FRAME .SRC="URL ou adresse du document à afficher dans la fenêtre">`

On construit 3 fichiers Html élémentaires que l'on place dans le même répertoire que le fichier de frames :

A.htm

```
<HTML><BODY>
<H4>A</H4>
</BODY></HTML>
```

B.htm

```
<HTML><BODY>
<H1>B</H1>
</BODY></HTML>
```

C.htm

```
<HTML><BODY>
<H1>C</H1>
</BODY></HTML>
```

On reprend le fichier de frame précédent que l'on complète.

```
<FRAMESET ROWS="30%,70%">
  <FRAME SRC="A.htm">
  <FRAMESET COLS="30%,70%">
    <FRAME SRC="B.htm">
    <FRAME SRC="C.htm">
  </FRAMESET>
</FRAMESET>
```



Les ascenseurs, comme à la fenêtre A, apparaissent automatiquement. Par l'attribut de la balise `<FRAME>` `SCROLLING="yes/no/auto"` vous pouvez indiquer si la fenêtre doit ou non posséder une barre de défilement.

13.5 Un autre attribut de cette balise `<FRAME>` est `NAME="NOM"`. Name indique le nom de la fenêtre de telle sorte que cette frame puisse être utilisée comme cible d'un lien hypertexte.

Ainsi, je voudrais faire un lien sur B pour afficher le contenu de ce lien (prenons le fichier A.htm pour éviter de l'encodage) dans C .

Le fichier de frames devient :

```
<FRAMESET ROWS="30%,70%">
<FRAME SRC="A.htm">
  <FRAMESET COLS="30%,70%">
    <FRAME SRC="B.htm">
    <FRAME SRC="C.htm" NAME="fenetreC">
  </FRAMESET>
</FRAMESET>
```

Et on met un lien vers A.htm dans le fichier B.htm en désignant comme cible la frame C.

```
<HTML><BODY>
<A HREF="A.htm" TARGET="fenetreC"><H1>B</H1></A>
</BODY></HTML>
```

L'attribut TARGET peut aussi prendre certaines valeurs prédéfinies :

- `_blank` qui indique au browser qu'il doit créer une nouvelle fenêtre afin d'y afficher le fichier. Dans ce cas, vous ouvrez en fait un nouveau browser.
- `_self` qui indique que le fichier sera chargé dans la même fenêtre que celle dans laquelle se trouve le lien.
- `_top` qui implique l'affichage du fichier sur toute la surface de la fenêtre du browser.

VERIFIEZ TOUJOURS VOS DIFFERENTS LIENS AVEC DES FRAMES. Vous éviterez ainsi que votre beau site ne devienne rapidement une "soupe".

13.6 Par défaut, les cadres sont séparés par des bordures. Il est possible de supprimer ces bordures mais les attributs à utiliser diffèrent selon Netscape ou Internet Explorer. Netscape utilise l'attribut `"border=0"` et Explorer, les attributs `"frameborder=no"` et `"framespacing=0"` (pour enlever l'espace entre les cadres). Le tout cohabite sans problème. La balise devient alors par exemple :

```
<FRAMESET COLS="30%,70%" border=0 frameborder=no framespacing=0>
```

13.7 Il y a pour terminer la balise `<NOFRAMES>...</NOFRAMES>` qui est utilisée pour indiquer le texte que doivent afficher les browsers incapables de gérer les frames. Il est même indiqué de prévoir une page sans fenêtres pour que ces visiteurs puissent profiter quand-même de votre site.

Section 13 : Les formulaires

1. Définition d'un formulaire

1.1. Définition

Avant de pouvoir utiliser les différentes sortes de formulaires (ligne de texte, liste déroulante, cases à cocher...), il faut déclarer au browser qu'il devra gérer des formulaires et ce qu'il devra en faire.

```
<FORM method="post" action="URL d'expédition" enctype="text/plain">
... les formulaires proprement dit ...
</FORM>
```

1.2. Commentaires

- L'attribut `"method"` vous offre le choix entre get et post. La différence entre ces deux méthodes repose sur la façon dont les données seront transmises au serveur et exploitées par celui-ci. Avec le temps, la méthode post s'est imposée car elle apparaît plus efficace et permet le traitement d'une quantité plus importante de données.

- L'attribut "action" spécifie l'adresse d'expédition des données.

Dans le cas d'un traitement des données par une CGI, on spécifie le répertoire CGI du serveur et le nom de la CGI.

```
<FORM method="post" action="http://www.serveur/cgi-bin/ma_cgi.pl">
```

Dans le cas d'un envoi vers en adresse électronique (e-mail), on utilise le protocole mailto: suivi de l'adresse électronique de destinataire (généralement votre adresse e-mail).

```
<FORM method="post" action="mailto:Vanlancker.Luc@ccim.be">
```

(sans espace entre mailto: et l'adresse e-mail !)

- L'attribut "enctype" (optionnel) spécifie l'encodage utilisé pour le contenu du formulaire. Ce paramètre ne peut être utilisé qu'accompagné par la méthode post. Ainsi enctype="text/plain" encode le contenu du formulaire en format texte lisible par le destinataire. Cette option est particulièrement adaptée à l'action du type mailto.
- Il n'est pas inutile de prévoir l'attribut name="nom" si la page comporte plusieurs formulaires.
- Attention !!! Je suis certain que, emporté par votre impatience à encoder les formulaires, vous allez oublier la balise de fin </FORM>. Dans ce cas, à la visualisation dans le navigateur, rien ne sera affiché.
- Dans le cas de l'utilisation en interne des formulaires par du Javascript, les attributs method, action et enctype sont inutiles car on ne fait pas appel au serveur.

2. Ligne de texte

2.1. Définition

INPUT type="text" indique un champ de saisie d'une seule ligne. C'est assurément le formulaire le plus simple à mettre en oeuvre :

```
<FORM>
<INPUT type="text" name="nom" size="50">
</FORM>
```

2.2. Commentaires

- L'attribut name="nom" est quasiment obligatoire car on n'utilise que rarement un seul formulaire. Le nom va identifier la chaîne de caractères du champ de saisie. De façon schématique, nom = (ce qui est introduit dans la ligne de texte).
- L'attribut size (optionnel) définit la longueur du champ de saisie. Notons que l'on peut introduire un nombre de caractères plus élevé que celui de la longueur.
- Il existe l'attribut maxlength="x" (optionnel) qui limite le nombre réel de caractères que l'on peut introduire dans le champ de saisie. La balise <INPUT> n'a pas de balise de fin.

3. Zone de saisie

3.1. Définition

La balise <TEXTAREA></TEXTAREA> introduit une zone de texte multilignes et non plus une simple ligne de texte. La syntaxe est :

```
<FORM>
<TEXTAREA name="nom" rows=4 cols=40>Valeur par défaut</TEXTAREA>
</FORM>
```

Valeur par défaut

3.2. Commentaires

- L'attribut name permet de donner un nom à la zone de saisie.
- L'attribut rows=x détermine le nombre de lignes de la zone de texte.
- L'attribut cols=y détermine le nombre de caractères visibles sur chaque ligne ou si vous préférez le nombre de colonnes.
- L'attribut wrap (optionnel) détermine la façon dont les sauts de ligne seront traités lors d'un changement de ligne.
 - * Avec wrap=virtual, les changements de lignes sont effectués automatiquement dans la zone de texte mais le tout sera transmis en une seule ligne.
 - * Avec wrap=physical, les changements de lignes sont effectués automatiquement dans la zone de texte et ceux-ci sont également transmis.
 - * Avec wrap=off, il n'y a aucun changement de ligne.
- Attention !!! La balise <TEXTAREA> a une balise de fin, soit </TEXTAREA>.

4. Liste déroulante

4.1. Définition

La balise <SELECT></SELECT> indique au browser l'usage d'une liste déroulante. Les éléments de la liste sont introduits par la balise <OPTION> ... (</OPTION> facultatif).

```
<FORM>
<SELECT name="nom" size="1">
<OPTION>lundi
<OPTION>mardi
<OPTION>mercredi
<OPTION>jeudi
<OPTION>vendredi
</SELECT>
</FORM>
```

lundi

Si vous cliquez sur la petite flèche vers le bas, vous obtiendrez la liste déroulante où on retrouve les éléments de la liste (<OPTION>).

lundi

lundi
mardi
mercredi
jeudi
vendredi

4.2. Commentaires

- L'attribut name="nom" définit le nom de la liste déroulante.
- L'attribut size="x" détermine le nombre d'éléments de liste affichés dans la boîte d'entrée. En fait, size="1" est optionnel car "1" est la valeur par défaut. Le même exemple avec size="3" donne :



- On peut présélectionner l'élément affiché dans la boîte d'entrée (par défaut, le premier élément de la liste sera retenu). On utilise pour ce faire l'attribut selected de la balise <OPTION>. Pour faire afficher d'entrée mercredi au lieu de lundi notre exemple devient :

```
<FORM>
<SELECT name="nom" size="1">
<OPTION>lundi
<OPTION>mardi
<OPTION selected>mercredi
<OPTION>jeudi
<OPTION>vendredi
</SELECT>
</FORM>
```



- La balise <SELECT> a une balise de fin </SELECT> tandis que la balise de fin </OPTION> est facultative.

5. Bouton d'option

5.1. Définition

Il serait plus logique de parler de boutons d'option car ils n'ont de sens que s'ils sont plusieurs. Ainsi on parle d'un groupe de boutons d'options.

Les boutons d'option, aussi appelés boutons radio, ont comme particularité qu'une seule option à la fois peut être activée (le "ou" exclusif).

La syntaxe de base est :

```
<FORM>
<INPUT type="radio" name="nom du groupe" value="valeur du bouton">
</FORM>
```

Ainsi, si on propose un choix entre, ou le tarif de jour ou le tarif de nuit ou le tarif de week-end, l'exemple devient

```
<FORM>
<INPUT type="radio" name="tarif" value="jour"> tarif de jour
<INPUT type="radio" name="tarif" value="nuit"> tarif de nuit
<INPUT type="radio" name="tarif" value="week-end"> tarif de week-end
</FORM>
```

☐ tarif de jour ☐ tarif de nuit ☐ tarif de week-end

5.2. Commentaires

- Vous avez compris que l'attribut name="nom" doit avoir le même nom pour tout le groupe de boutons d'option.
- L'attribut "checked" (optionnel) permet de présélectionner un bouton radio. Ainsi

```
<INPUT type="radio" name="tarif" value="jour" checked> tarif de jour
```

présenterait le bouton radio "tarif de jour" en position présélectionnée.

☒ tarif de jour ☐ tarif de nuit ☐ tarif de week-end

- Le contenu de l'attribut "value" du bouton retenu sera renvoyé par mailto ou utilisé par le Javascript.
- La balise <INPUT> n'a pas de balise de fin.
- Pour la petite histoire le terme radio ferait référence aux anciens postes de radio sur lesquels le fait d'appuyer sur un bouton désactivait le bouton précédemment enfoncé.

6. Case à cocher

6.1. Définition

La philosophie des cases à cocher [checkbox] est assez similaire aux boîtes d'option. Ici, cependant, plusieurs choix simultanés peuvent être réalisés.

La syntaxe de base est :

```
<FORM>
<INPUT type="checkbox" name="nom" value="valeur attachée au bouton">
</FORM>
```

Comme exemple :

```
<FORM>
<INPUT type="checkbox" name="choix1" value="1"> glace vanille
<INPUT type="checkbox" name="choix2" value="2"> chantilly
<INPUT type="checkbox" name="choix3" value="3"> chocolat chaud
<INPUT type="checkbox" name="choix4" value="4"> biscuit
</FORM>
```

☐ glace vanille ☐ chantilly ☐ chocolat chaud ☐ biscuit

6.2. Commentaires

- Les règles pour l'attribut name="nom" sont moins précises que pour les boutons d'option. Vous pouvez employer des noms identiques ou des noms différents pour chaque case à cocher. Cependant des noms différents sont nécessaires pour l'utilisation d'un script.
- L'attribut checked (optionnel) permet de présélectionner une case à cocher. Ainsi

```
<INPUT type="checkbox" name="choix1" value="1" checked> glace vanille
```

présenterait la case à cocher "glace vanille" en position présélectionnée.

☒ glace vanille ☐ chantilly ☐ chocolat chaud ☐ biscuit

- Le contenu de l'attribut "value" du ou des boutons retenus seront renvoyés par mailto ou utilisé par le Javascript.

- La balise <INPUT> n'a pas de balise de fin.

7. Bouton de commande

7.1. Définition

Avec l'introduction des langages de scripts (Javascript et VBscript) l'usage du bouton de commande présente un intérêt certain. Simplement à titre d'illustration ou pour vous donner envie d'en savoir plus sur le Javascript, je vous propose d'en découvrir la syntaxe :

```
<FORM>
<INPUT type="button" name="nom" value="texte du bouton" onclick="fonction Javascript">
</FORM>
```

Voyons un petit exemple.

```
<FORM>
<INPUT type="button" name="nom" value="Bouton de test" onclick="alert('Test réussi !')">
</FORM>
```

En cliquant sur le bouton "Bouton de test", on va déclencher une fonction Javascript élémentaire [mon cher Watson] qui consiste à afficher une petite boîte (dite d'alerte) avec le texte "Test réussi!".



7.2. Commentaires

- L'attribut "value" permet d'adapter le texte du bouton à vos souhaits.
- Avec le bouton de commande, il n'est pas nécessaire d'avertir le browser qu'on utilisera du Javascript par une balise du genre <SCRIPT language="javascript">. En effet, le navigateur (compatible Javascript) reconnaît par défaut les fonctions en Javascript. Par contre, si vous utilisez du VBscript, il faudra utiliser l'encodage <INPUT type="button" name="test" value="Pour un test" language="VBscript" OnClick="MsgBox 'Test réussi!'"> (Pour rappel, Javascript par défaut).

8. Submit et Reset

8.1. Submit

Le bouton Submit a la tâche spécifique de transmettre toutes les informations contenues dans le formulaire à l'URL désignée dans les attributs ACTION et METHOD du tag <FORM>.

la syntaxe Html est :

```
<FORM>
<INPUT TYPE="submit" NAME="nom" VALUE="texte du bouton">
</FORM>
```

Soit par exemple :

```
<FORM>
```

```
<INPUT TYPE="submit" NAME="nom" VALUE=" Envoyer ">
</FORM>
```

Envoyer

Commentaires

- Les modifications seront peu nombreuses car le bouton Submit a une fonction Html très spécifique. Seul le texte du bouton pourra être modifié (par défaut Submit).
- La balise <INPUT> n'a pas de balise de fin.

8.2. Reset

Le bouton Reset permet d'annuler les modifications apportées aux contrôles d'un formulaire et de restaurer les valeurs par défaut.

la syntaxe Html est :

```
<FORM>
<INPUT TYPE="reset" NAME="nom" VALUE="texte du bouton">
</FORM>
```

Soit par exemple :

```
<FORM>
<INPUT TYPE="reset" NAME="nom" VALUE=" Annuler ">
</FORM>
```

Annuler

Commentaires

- Les modifications seront peu nombreuses car le bouton Reset a une fonction Html très spécifique. Seul le texte du bouton pourra être modifié (par défaut Reset).
- La balise <INPUT> n'a pas de balise de fin.

9. Un livre d'or

Une des premières applications des formulaires est celle d'un livre d'or [guestbook]. Vous trouverez ci-après un exemple complet de l'utilisation des formulaires par mailto et qui peut être adapté par chacun pour en faire "son" livre d'or.

N'oubliez pas que mailto ne fonctionne pas sous Microsoft Explorer 3.0 mais seulement sous Explorer 4.0 et Netscape 3.0 et plus. Pour être précis, mailto sous Explorer 3.0 n'entraîne pas de message d'erreur mais ouvre simplement une fenêtre de courrier électronique sans tenir compte du formulaire.

Pour ma part, j'ai retiré la page de livre d'or de mon site car à mon avis, pour un site d'amateur ou d'hobbyiste, une prédéfinition de l'information n'était pas forcément adaptée. Il me semble qu'une libre expression du lecteur (concise ou plus étendue) était de loin plus enrichissante. En outre, certains ne tiennent pas à transmettre leur nom ou leur domicile et souhaitent ne s'en tenir qu'à leur adresse email.

Voici donc un exemple de livre d'or que vous pouvez adapter :

```
<FORM method="post" action="mailto:riadh_inform@yahoo.fr">
Votre nom :<BR>
<INPUT type="text" name="nom"><BR>
```

```

Votre adresse :<BR>
<TEXTAREA name="adresse" ROWS=2 COLS=35>
</TEXTAREA><BR>
<INPUT type="submit" value="Envoyer">
<INPUT type="reset" value="Annuler">
</FORM>

```

Votre nom :

Votre adresse :

Vous recevrez dans notre boîte de réception, un truc bizarre du genre :

nom=Riadh+Bouslimi&adresse=Rue+des+Brasseurs+2217OD%OA7700+Jendouba.

- où on retrouve les champs nom= et adresse=, •où les champs sont séparés par le signe &,
- où les espaces sont remplacés par le signe +
- et 17%OD%OA correspond à un passage à la ligne.

Il existe heureusement des programmes comme "mailto: Formatter" pour vous aider à décoder tout ceci.

Liste des principales balises Html

Mise en forme des caractères

```

<B>...</B>
<BIG>...</BIG>
<BLINK>...</BLINK>
<EM>...</EM>
<FONT color="#XXXXXX">...</FONT>
<FONT size=X>...</FONT>
<I>...</I>
<NOBR>...</NOBR>
<PRE>...</PRE>

```

```

<SMALL>...</SMALL>
<STRONG>...</STRONG>
<SUB>...</SUB>
<SUP>...</SUP>
<U>...</U>

```

Texte en gras
 Agrandissement de la taille des caractères
 Texte clignotant (Netscape seul)
 Texte en italique
 Texte en couleur où XXXXXX est une valeur hexadécimale
 Taille des caractères où X est une valeur de 1 à 7
 Texte en italique
 Empêche les ruptures automatiques de ligne des navigateurs
 Texte préformaté, soit avec affichage de tous les espaces et sauts de ligne
 Réduction de la taille des caractères
 Mise en gras du texte
 Texte en indice
 Texte en exposant
 Texte souligné

Mise en forme du texte

```

<!--...-->
<BR>
<BLOCKQUOTE>...</BLOCKQUOTE>
<CENTER>...</CENTER>
<DIV align=center> ...</DIV>
<DIV align=left> ...</DIV>
<DIV align=right> ...</DIV>
<Hx>...</Hx>
<Hx align=center>...</Hx>
<Hx align=left>...</Hx>
<Hx align=right>...</Hx>
<P>...</P>
<P align=center>...</P>
<P align=left>...</P>
<P align=right>...</P>

```

Commentaire ignoré par le navigateur
 A la ligne
 Citation (introduit un retrait du texte)
 Centre tout élément compris dans le tag
 Centre l'élément encadré par le tag
 Aligne l'élément à gauche
 Aligne l'élément à droite
 Titre où x a une valeur de 1 à 6
 Titre centré
 Titre aligné à gauche
 Titre aligné à droite
 Nouveau paragraphe
 Paragraphe centré
 Paragraphe aligné à gauche
 Paragraphe aligné à droite

Listes

```

<UL>
<LI>
</UL>
<OL>
<LI>
</OL>
<DL>
<DT>...</DT>
<DD>...</DD>
</DL>
<HR>
<HR width="x%">
<HR width=x>
<HR size=x>
<HR align=center>
<HR align=left>
<HR align=right>
<HR noshade>

```

Liste non numérotée (dite à puces)
 Élément de liste

Liste numérotée
 Élément de liste

Liste de glossaire
 Terme de glossaire (sans retrait)
 Explication du terme (avec retrait)
 Ligne de séparation
 Trait horizontal (centré par défaut)
 Largeur du trait en %
 Largeur du trait en pixels
 Hauteur du trait en pixels
 Trait centré (défaut)
 Trait aligné à gauche
 Trait aligné à droite
 Trait sans effet d'ombrage

Hyperliens

...
 ...
 ...
 ...
 ...
 ...

Lien vers une page Web
 Lien vers une adresse Email
 Lien vers la page locale fichier.htm située dans le même dossier
 Définition d'une ancre
 Lien vers une ancre

Images

Insertion d'une image au format Gif ou Jpg
 (voir liens pour l'adressage)
 Mise à l'échelle de l'image en pixels
 Définition de la bordure d'une image avec lien
 Texte alternatif lorsque l'image n'est pas affichée
 Aligne l'image en bas
 Aligne l'image au milieu
 Aligne l'image en haut
 Aligne l'image à gauche
 Aligne l'image à droite
 Espacement horizontal entre l'image et le texte
 Espacement vertical entre l'image et le texte

Tableau

<TABLE>...</TABLE>
 <TABLE width="x%">
 <TABLE width=x>
 <TABLE border=x>
 <TABLE cellpadding=x>
 <TABLE cellspacing=x>
 <TR>...</TR>
 <TD>...</TD>
 <TD bgcolor="#XXXXXX">
 <TD width="x%">
 <TD width=x>
 <TD align=center>
 <TD align=left>
 <TD align=right>
 <TD valign=bottom>
 <TD valign=middle>
 <TD valign=top>
 <TD colspan=x>
 <TD rowspan=x>

Définition d'un tableau
 Largeur du tableau en %
 Largeur du tableau en pixels
 Largeur de la bordure
 Espace entre la bordure et le texte
 Epaisseur du trait entre les cellules
 Ligne du tableau
 Cellule du tableau
 Couleur d'une cellule de tableau
 Largeur de colonne en %
 Largeur de colonne en pixels
 Texte dans la cellule centré
 Texte dans la cellule aligné à gauche
 Texte dans la cellule aligné à droite
 Alignement vers le bas du contenu d'une cellule
 Centrage vertical du contenu d'une cellule
 Alignement vers le haut du contenu d'une cellule
 Nombre de cellules à fusionner horizontalement
 Nombre de cellules à fusionner verticalement

Frames

<FRAMESET>...</FRAMESET>
 <FRAMESET rows="x%,y%,...">
 <FRAMESET cols="x%,y%,...">
 <FRAME src="fichier.htm">
 <NOFRAMES>...</NOFRAMES>

Définit une structure de frames(remplace alors le tag BODY)
 Division horizontale de la fenêtre en %
 Division verticale de la fenêtre en %
 Fichier affiché dans une fenêtre de frames
 Contenu pour les browsers non prévus pour les frames

Fichier Html

<HTML>...</HTML>
 <HEAD>...</HEAD>
 <TITLE>...</TITLE>
 <BODY>...</BODY>
 <BODY bgcolor="#XXXXXX">
 <BODY background="xyz.gif">

Début et fin de fichier Html
 Zone d'en-tête d'un fichier Html
 Titre affiché par le browser (élément de HEAD)
 Début et fin du corps du fichier Html
 Couleur d'arrière-plan (en hexadécimal)
 Image d'arrière-plan

Les caractères spéciaux

Voici ces caractères en code ISO 5589-1 (appelée aussi ISO-Latin 1), ainsi que leur abréviation ENTITY.

Caractère	Code ISO	ENTITY	Caractère	Code ISO	ENTITY	Caractère	Code ISO	ENTITY
,			«	«	«	Ö	Õ	Õ
f	‚		¬	¬	¬	Ï	Ö	Ö
„	ƒ		-	­	­	×	×	×
...	„		®	®	®	Ø	Ø	Ø
†	…		—	¯	&masr;	Ù	Ù	Ù
‡	†		°	°	°	Ú	Ú	Ú
§	‡		±	±	±	Û	Û	Û
^	ˆ		²	²	²	Ü	Ü	Ü
%	‰		³	³	³	Ý	Ý	Ý
§	Š		´	´	´	Þ	Þ	Þ
‹	‹		µ	µ	µ	ß	ß	ß
Œ	Œ		¶	¶	¶	à	à	à
			·	·	·	á	á	á
	Ž		,	¸	¸	â	â	â
			1	¹	¹	ã	ã	ã
			º	º	º	ä	ä	ä
‘	‘		»	»	»	å	å	å
’	’		¼	¼	¼	æ	æ	æ
“	“		½	½	½	ç	ç	ç
”	”		¾	¾	¾	è	è	è
•	•		¿	¿	?	é	é	é
—	–		À	À	À	ê	ê	ê
—	—		Á	Á	Á	ë	ë	ë
~	˜		Â	Â	Â	ì	ì	ì
™	™		Ã	Ã	Ã	í	í	í
§	š		Ä	Ä	Ä	î	î	î
›	›		Å	Å	Å	ï	ï	ï
œ	œ		Æ	Æ	&Aelig;	ð	ð	ð
			Ç	Ç	Ç	ñ	ñ	ñ
	ž		È	È	È	ò	ò	ò
ÿ	Ÿ		É	É	É	ó	ó	ó
espace	 	 	Ê	Ê	Ê	ô	ô	ô
¡	¡	¡	Ë	Ë	Ë	õ	õ	õ
¢	¢	¢	Ì	Ì	Ì	ö	ö	ö
£	£	£	Í	Í	Í	÷	÷	÷
¤	¤	¤	Î	Î	Î	ø	ø	ø
¥	¥	¥	Ï	Ï	Ï	ù	ù	ù
	¦	¦	Ð	Ð	Ð	ú	ú	ú
§	§	§	Ñ	Ñ	Ñ	û	û	û
”	¨	¨	Ò	Ò	Ò	ü	ü	ü
©	©	©	Ó	Ó	Ó	ý	ý	ý
ª	ª	ª	Ô	Ô	Ô	þ	þ	þ
						ÿ	ÿ	ÿ