ABC création



1st part

Première partie

Introduction
Les balises
Le texte
L'espace insécable
Retour à la ligne
Taille des caractères
Couleur des caractères
Alignement de textes ou de paragraphes
Le séparateur horizontal
Les liens hypertextes et adressages
Les images
La balise d'image
Table des caractères spéciaux

Introduction :

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

- Vous avez donc deux interlocuteurs :
1. le browser de votre « visiteur »
2. et votre « visiteur » 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 ainsi qu'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 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 visiteur » affichera sur l'écran de celui-ci (à moins de tester vos pages
sur tous les principaux browsers).

- 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 format de texte pur et dur, du Bloc-notes ou "Notepad " de Windows par exemple. En revanche vous serez privé de certains caractères spéciaux comme le é pour lesquels il faudra passer par des codes particuliers. (voir table des caractères en annexe à ce dossier)

- En « récompense » le HTML est un langage universel qui s'adapte à toutes les plates-formes que ce soit Windows, Macintosh, Unix, OS/2...

Les outils nécessaires pour produire des pages Web :

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 (SimpleText pour mac).
· d'un browser soit Netscape Navigator, soit Microsoft Explorer que vous pouvez cependant télécharger sur le web (http://home.netscape.com ou http://www.microsoft.com), mais aussi sur les CD de certains mensuels informatiques.

Vous n'avez pas besoin:

· d'être connecté pour écrire, voir et peaufiner vos pages Html.
· d'avoir un éditeur Html comme Claris Home Page, Frontpage de Microsoft, Hotdog, Hotmetal, WebExpert ou Netscape Editor ainsi que des dizaines d'autres éditeurs que vous trouverez sur le Web.

Conseils

Le langage HTML étant un ensemble de balises et d'attributs, il nous paraît utile sinon indispensable de toutes les passer en revue ou 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 ou de la création d'un document avec une mise en page complexe. 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. (c'est de loin la meilleure façon d' apprendre !)
· vous aurez besoin d'une connaissance pointue du HTML pour inclure les codes du Javascript ou du Java dans vos pages.

Les Balises :

En plus du texte adressé à votre « visiteur », 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...).

Les balises de base (obligatoires) : exemple de la page blanche.

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 qui n' apparaîtront jamais dans la page).
</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 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.

Un premier exemple de page web : le document minimum

· Ouvrir l'éditeur de texte.
· Ecrire les codes Html suivants:

```
<HTML>
<HEAD>
<TITLE>La page de base en HTML</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...
· Admirer son premier document Html.

Celui-ci est vide (et c'est normal). Il faudra maintenant lui fournir votre information à l'intérieur des balises <BODY></BODY>.
Remarquez que votre titre (<TITLE>...</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 cliquer dans la barre "Location" et faire "Enter".

Le texte :

Tout document HTML contiendra en majorité du texte.
Le texte tout simple s'écrit sans balises. Il sera repris par le browser de votre « visiteur » avec la police et la taille de caractères choisis dans sa configuration par défaut

Voyons comment l'agrémenter par quelques balises élémentaires :

Gras [Bold] <B>...</B> Début et fin de zone en gras
Italique [Italic] <I>...</I> Début et fin de zone en italique
Souligné [Underline] <U>...</U> Début et fin de zone soulignée
Clignotant [Blink] <BLINK>...</BLINK> Début et fin de zone clignotante

· La balise <U>...</U> souligne le texte.
Comme, par convention, les éléments servant d'hyperlien sont soulignés, on évitera de 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.

· Il n'est pas rare d'utiliser plusieurs balises pour un même élément de texte.
Il faut donc veiller à les imbriquer ainsi :
<B><I> ... </I><B> est correct.
<B><I> ... </B></I> risque de vous créer des ennuis

Exemple :

```
<HTML>
<HEAD><TITLE>Exemple de styles de caract&egrave ;res</TITLE></HEAD>
<BODY>texte simple<BR>
<B>texte en gras</B><BR>
<I>texte en italique</I><BR>
<B><I>texte en gras et en italique</I></B><BR>
</BODY>
</HTML>
```

L'espace insécable [non-braking space] :   ;

Quel que soit le nombre d'espaces que vous mettrez dans votre code source le HTML ne comprend qu'un seul espace entre deux mots, si vous souhaitez en introduire plusieurs, il faudra utiliser la commande «   ; » que vous taperez directement entre les mots.

Exemple :

```
<HTML>
<HEAD>
<TITLE></TITLE>
<BODY>
je place plusieurs espaces  ;   ;   ;ici !
</BODY>
</HTML>
```

Retour à la ligne [Line break] :

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 <BR>.

Celle-ci représente une action ponctuelle et n'a donc pas besoin de balise de fin.

Taille de caractère [Font size] :

<FONT SIZE="$$$">...</FONT> Début et fin de zone avec cette taille

La taille dans <FONT SIZE="$$$"> peut être indiquée de deux façons :

· avec un nombre de 1 à 7. La valeur par défaut étant 3.

· de façon relative par rapport à la valeur par défaut (ici 0), soit -3 -2 -1 0 +1 +2 +3.

La dernière méthode est celle qui est le plus souvent utilisée (même par les éditeurs HTML)

Exemple :

```
<HTML>
<HEAD>
<TITLE> Exemple de tailles de caract&egrave;res</TITLE>
</HEAD>
<BODY>
texte normal taille par defaut<BR>
<FONT SIZE="0"> texte de taille normale</FONT><BR>
<FONT SIZE="1"> texte de tr&egrave;s petite taille </FONT><BR>
 <FONT SIZE="-3"> texte de tr&egrave;s petite taille </FONT><BR>
 <FONT SIZE="+3"> texte de tr&egrave;s grande taille </FONT><BR>
<FONT SIZE="+3"><B> texte de tr&egrave;s grande taille en
gras</B></FONT><BR>
 </BODY>
</HTML>
```

Couleur de caractère [Font color] :

<FONT COLOR="#$$$$$$">... </FONT> Début et fin de zone en couleur

Voici les codes de quelques couleurs basiques :

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

Exemple :

```
<HTML>
<HEAD><TITLE>Un mix des balises de texte</TITLE></HEAD>
<BODY>texte simple<BR>
<B>texte en gras</B><BR>
<FONT SIZE="5"><B><I>texte en gras et en italique de grande taille</I></B></FONT><BR>
<FONT COLOR="#0000FF"><FONT SIZE="5"><B><I>texte en gras et en italique de grande taille en
```

bleu</I></B></FONT></FONT>
</BODY>
</HTML>

Alignement de textes ou de paragraphes [Div align] : <DIV ALIGN="$$$">

Pour définir l'alignement d'un texte, il suffit de le placer entre les balises <DIV ALIGN="$$$"> et </DIV>.
Le paramètre $$$ peut prendre la valeur : LEFT, CENTER ou RIGHT, en fonction de l'alignement que vous souhaitez.

Remarque : la balise <CENTER> est une alternative à <DIV ALIGN="$$$">
lorsqu'on souhaite simplement centrer (le plus souvent utilisé).

Exemple :

```
<HTML>
<HEAD><TITLE>Les alignements de texte</TITLE></HEAD>
<BODY>
<DIV ALIGN="CENTER"> Texte centre </DIV><BR>
<CENTER> texte centre avec la balise CENTER </CENTER><BR>
<DIV ALIGN="LEFT"> Texte aligne &agrace; gauche </DIV><BR>
<DIV ALIGN="RIGHT"> Texte aligne &agrace; droite </DIV>
</BODY>
</HTML>
```

Le séparateur [Horizontal Rule]: <HR>

Les browsers intègrent un outil de mise en forme intéressant pour clarifier la présentation de votre texte. C'est la ligne horizontale. Ce tag fort simple, nous permettra de comprendre un peu mieux le fonctionnement du navigateur.
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.
une largeur de 100% de la fenêtre.

Vous pouvez très bien modifier au gré de votre fantaisie les valeurs de ces différents attributs.

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

Il est donc tout à fait équivalent d'écrire :
<HR> ou <HR size="2" align="center" width="100%">

Remarque : Cette action ponctuelle n'a donc pas besoin de balise de fin (</HR>).

Exemple :

```
<HTML>
<HEAD><TITLE>Les s&eacute;parateurs</TITLE></HEAD>
<BODY>
La balise HR sans ajout d'attributs
<HR><BR><BR>
La balise HR avec les attributs par defaut
<HR size="2" align="center" width="100%"><BR><BR>
La balise HR avec differents valeurs d'attributs
<HR size="10" align="center" width="100"><BR><BR>
<HR size="2" align="LEFT" width="30 %"><BR><BR>
<HR size="10" align="RIGHT" width="100"><BR>
</BODY>
</HTML>
```

Les Liens hypertextes et adressages:

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 fichier HTML situé sur votre ordinateur.
· vers un autre fichier HTML 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 :

```
<A HREF="URL ou Adresse">...</A>
```

Les Liens Internes au site web

L'organisation classique et plus que conseillée d'un site Web consiste à regrouper l'ensemble des éléments de celui-ci (fichiers htm, images, ...) dans un même « grand répertoire », organisé en sous-répertoire (séparer les pages des images).

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 ou sous-repertoire/fichier

```
<A HREF="fichier.htm">.</A>   ou <A HREF= "sous-repertoire/fichier.htm">.</A>
```

Attention Danger ! :

Il est peu probable que le serveur Web qui hébergera votre site, possède la même arborescence que votre disque local.

· L'adresse ayant la forme :
file:///lecteur:/répertoire/fichier.htm (adressage absolu).
Vous mène droit à l'erreur (et certains éditeurs HTML utilisent cette forme...).

· La forme de type : ../../../fichier.htm (adressage relatif).
Evite les problèmes de « portabilité » entre votre ordinateur et le serveur qui va héberger votre site, puisque dans ce cas seul compte l'adressage à partir du « grand » répertoire dans lequel tout votre site est contenu.

Je vous conseille d'utiliser l'adressage relatif.
Remarque : « ../ » sert à « remonter » d'un répertoire dans l'arborescence.
Pour « descendre » dans l'arborescence il suffit d'écrire le « chemin » :
Répertoire/ sous-répertoire/ fichier.htm

Exemple :

Dans un répertoire nommé « mon site web », il y a deux sous-répertoires appelés « pages1 » et « pages2 ».
« pages 1» contient la moitié des pages HTML et « pages2 » l'autre moitié.

Dans ce cas les liens entre les différentes pages au sein de « pages1 » seront :
<A HREF= "page cible"> texte du lien </A>

Par contre les liens entre les pages de « pages1 » vers les page de pages2 » seront :
<A HREF="../pages2/page_cible.htm"> texte du lien </A>

Lien externe :

Tout site web sur le réseau possede une adresse URL (Universal Ressource Locator).
Vous pouvez « lier » votre site à d'autres, en indiquant l'adresse URL du fichier à joindre.

Exemple :
Pour relier votre site web au site MCI
<A HREF="http://www.devinci.fr/mci/index.html"> texte du lien</A>

Les Images :

GIF ou JPEG ?

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

· Les deux formats d'images (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 16 couleurs 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.

· 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 droite de la souris et de suivre les instructions du menu déroulant (Save Image As...).

· 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 un sous-répertoire nommé img ou image, contenu dans le « grand » répertoire.

· Le fait d'utiliser la même image à plusieurs reprises dans un fichier HTML n'alourdit pas sa taille mémoire.

Remarque : L'adressage d'une image fonctionne exactement comme l'adressage des liens hyperTextes.

La balise d'image [image] : <IMG SRC>

<IMG SRC="Adresse de l'image"> Afficher l'image qui se trouve à l'adresse...

La balise image possède de nombreux attributs :
Texte alternatif [alternative text] : alt="$$$"

Pour les browser n'ayant pas l'option "image" activée, et également visible lors du chargement de l'image, ALT fait apparaître un texte dans l'espace réservé à l'image qui « devait » s'afficher

Prévoir un texte alternatif 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.

Dimensions de l'image [width, Height] :

width="$$$" : Largeur (en pixels)
height="$$$" : Hauteur (en pixels)

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 votre logiciel de production graphique.

Bordure autour de l'image[Border] :

border="$$$" : Epaisseur du « bandeau » qui entoure l'image (en pixels)

Alignement de l'image par rapport au texte [Align] :

align="top" haut des lettres alignées avec le haut de l'image
align="middle" bas des lettres alignées avec le milieu de l'image
align="botton" bas des lettres alignées avec le bas de l'image

Glossaire des balises vues dans ce cours

<HTML>
<HEAD>
<TITLE>
<BODY>
<B>
<I>
<U>
<BLINK>
<HR>
<A HREF="URL ou Adresse">...</A>
<IMG SRC="Adresse de l'image">
<FONT>

Glossaire des attributs vu dans ce cours

ALT="****"
SIZE= "***"
COLOR="***"
WIDTH="***"
HEIGHT="***"
BORDER="***"
ALIGN="***"

Table des caractères spéciaux

Lorsque vous aurez besoin de l'un de ces caractères, il suffira de taper directement le code qui lui correspond à l'endroit même de votre texte où vous le souhaitez

Caractère	Code	Caractère	Code	Caractère	Code
espace		À	À	à	à
¡	¡	Á	Á	á	á
¢	¢	Â	Â	â	â
£	£	Ã	Ã	ã	ã
¤	¤	Ä	Ä	ä	ä
¥	&yen	Å	Å	å	å
¦	¦	Æ	&Aelig;	æ	æ
§	§	Ç	Ç	Ç	ç
¨	¨	È	È	è	è
©	©	É	É	é	é
ª	ª	Ê	&Ecric;	ê	ê
«	«	Ë	Ë	ë	ë
	¬	Ì	Ì	ì	ì
		Í	Í	í	í
®	®	Î	Î	î	î
¯	&masr;	Ï	Ï	ï	ï
°	°	Ð	Ð	ð	ð
±	±	Ñ	Ñ	ñ	ñ
²	²	Ò	Ò	ò	ò
³	³	Ó	Ó	ó	ó
´	´	Ô	Ô	ô	ô
m	µ	Õ	Õ	õ	õ
	¶	Ö	Ö	ö	ö
·	·	´	×	¸	÷
¸	¸	Ø	Ø	ø	ø
1	¹	Ù	Ù	ù	ù
º	º	Ú	Ú	ú	ú
»	»	Û	Û	û	û
¼	¼	Ü	Ü	ü	ü
½	½	Ý	Ý	ý	ý
¾	¾	Þ	Þ	þ	þ
¿	¿	ß	ß	ÿ	ÿ



Seconde partie

La couleur des textes et liens hypertexte
Les fonds d'écran
Les listes
Les tableaux
Les cellules de tableaux

La couleur des textes et liens hypertextes

Il est possible, dans un document HTML, de spécifier en une seule fois et pour tout le document :

· La couleur du texte
· La couleur des liens hypertextes.

Il suffit de placer dans la balise <BODY> (voir le document minimum), les attributs suivants :

TEXT TEXT= "#$$$$$$" Définit la couleur des textes
LINK LINK = "#$$$$$$" Définit la couleur des liens
ALINK [active link] ALINK = "#$$$$$$" Définit la couleur des liens pendant le click
VLINK [visited link] VLINK = "#$$$$$$" Définit la couleur des liens déjà visités

Exemple :

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY TEXT= "#00FF00" LINK= "#FF0000" ALINK= "#FF00FF" VLINK= "#000000">
<A HREF= home.htm> un lien pour voir</A>
</BODY>
</HTML>
```

Les fonds d'écran :

Le langage Html permet d'améliorer la présentation d'un document d'un arrière-plan [background] coloré ou composé d'une image. Ce qui apporte une touche "artistique" à votre page. C'est toujours avec la balise <BODY> qu'on va le définir Pour insérer une Couleur d'arrière-plan, il faut ajouter à cette balise l'attribut BGCOLOR : <BODY BGCOLOR="#$$$$$$">

Exemple :

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY BGCOLOR="#000088">
<FONT COLOR= "#FFFFFF">Un fond d'&eacute;cran bleu</FONT>
</BODY>
</HTML>
```

Remarque : Il ne faut pas oublier de « préserver » la lisibilité de votre texte, par exemple, un texte jaune sur un fond blanc vous garanti d'être illisible

On peut également placer une image en fond d'écran. Cette image est affichée en mosaïque par le browser.
Attention aux raccords... comme lorsque vous posez du papier peint.
Pour, placer une image en fond d'écran, il faut ajouter à la balise <BODY> l'attribut BACKGROUND : BACKGROUND="Adresse"

Exemple :

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY BACKGROUND="fond.gif">
<B>Une image en fond d'&eacute;cran</B>
</BODY>
 </HTML>
```

Les listes

Il peut être utile de placer des puces (points ou numéros) dans sa page HTML afin de créer des listes ou alors mettre en évidence des textes.
Liste non-ordonnée [Bullet list] <UL></UL> Créer une liste avec des puces en forme de gros points
Liste ordonnée [Numbered list] <OL></OL> Créer une liste numérotée.
Elément de liste [List items] <LI> Un élément de la liste

Exemple :

```
 <HTML>
 <HEAD><TITLE></TITLE></HEAD>
 <BODY>
Les jours de la semaine
<UL>
<LI>lundi
<LI>mardi
<LI>mercredi</UL>
Les jours de la semaine (suite...)
<OL>
<LI>jeudi
<LI>vendredi
<LI>samedi
<LI>dimanche
</OL>
 </BODY>
 </HTML>
```

Remarques :

· La couleur des puces rondes est celle definie dans <BODY> pour le texte.
· Les listes peuvent s'imbriquer:

```
<H4>Les 12 mois</H4>
<UL><LI>Les mois
<OL><LI>Les jours de la semaine
</OL></UL>
```

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, et même vivement conseillé.

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

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

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
<TABLE>
<TR><TD>1</TD><TD>2</TD></TR>
<TR><TD>3</TD><TD>4</TD></TR>
</TABLE>
 </BODY>
 </HTML>
```

Les différents aspects « graphiques » des tableaux
les bordures du tableau ont par défaut une épaisseur de 2 pixels, cependant il est possible de les modifier avec l'attribut <BORDER>
Bordure de cadre [Border] <TABLE border="?"></TABLE>

Exemple :

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
<TABLE border="5">
<TR><TD>1</TD><TD>2</TD></TR>
<TR><TD>3</TD><TD>4</TD></TR>
</TABLE>
</BODY>
</HTML>
```

Il y a encore trois éléments définis par défaut (mais modifiables) :
L'espace entre les cellules et le bord du tableau <TABLE cellspacing="?">

Par défaut fixé a 2. l'espace entre le bord de la cellule et son contenu <TABLE cellpadding="?">
Par défaut fixé a 2.
La hauteur de la table <TABLE height="?"><TABLE height="%"> Par défaut de la taille des éléments contenus par la table
La largeur de la table <TABLE width="?"><TABLE width="%"> Par défaut de la taille des éléments contenus par la table

Exemple :

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
<TABLE border="2" cellspacing="10" cellpadding="20">
<TR><TD>1</TD><TD>2</TD></TR>
<TR><TD>3</TD><TD>4</TD></TR>
</TABLE>
</BODY>
</HTML>
```

Exemple 2 :

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
<TABLE border="1" cellpadding="10" WIDTH="50%" HEIGHT="100" >
<TR><TD>1</TD><TD>2</TD></TR>
<TR><TD>3</TD><TD>4</TD></TR>
<TR><TD>5</TD><TD>6</TD></TR>
</BODY>
</TABLE>
</HTML>
```

Remarques :

En général, le seul attribut paramètre fréquemment utilisé est l'attribut BORDER, les attributs CELLSPACING et CELLPADING étant fixés à 0

Pour les mises en page complexes, afin qu'un tableau puisse servir de trame de fond transparente, les paramètres BORDER, CELLSPACING et CELLPADING sont fixés à 0

Les cellules des 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 fonds d'écran

· et même des tableaux

Les attributs que nous venons de voir concernent le table au sens général, mais il est également possible d'agir sur les cellules.
Largeur d'une cellule <TD width="?"> <TD width="%">
Hauteur d'une cellule <TD height="?"> <TD height="%">
Fusion de lignes <TD rowspan="?">
Fusion de colonnes <TD colspan="?">

Exemples :

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

Essayons ceci :

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
<CENTER><TABLE width="60%" border="1">
<TR>
<TD>cellule1</TD>
<TD>cel. 2</TD>
<TD>3</TD>
</TR>
</TABLE></CENTER>
</BODY>
</HTML>
```

cellule1	cel. 2	3

Pas génial ! essayons maintenant en agissant directement sur les cellules

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
<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>
</BODY>
</HTML>
```

cellule1	cel. 2	3

Impeccable ! Prenons, maintenant, le même tableau mais avec 2 lignes.

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
<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>
</BODY>
</HTML>
```

cellule1	cel. 2	3
cellule1	cel. 2	3

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

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
<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>
</BODY>
</HTML>
```

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.

```
<HTML>
<HEAD><TITLE></TITLE></HEAD>
<BODY>
<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>
</BODY>
</HTML>
```

Resultat :

cellule 1	cel 2	3
	cel 2	3

Les alignements dans les cellules.

Ligne de tableau <TR align=left/center/right><TRvalign=top/middle/bottom>
horizontalementverticalement
Cellule de tableau <TD align=left/center/right><TD valign=top/middle/bottom>
horizontalementverticalement
A titre d'exemple, reprenons le tableau suivant :

Tarif au ../../..		
Article 1	Ref 002	30fr

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 en terminer avec les tableaux, 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">

Remarque il est possible d'appliquer tout ce que nous avons vu sur les cellules a un rang entier en plaçant les attributs dans la balise <TR> (sauf colspan et cellspan).

ABC création



3rd part

Troisième partie
Les formulaires
Définition d'un formulaire
Ligne de texte
Zone de saisie
Liste déroulante
Bouton d'option
Cases à cocher
Les boutons submit et reset
Exemple d'un petit formulaire
Les frames

Les formulaires

Introduction :

Avec les formulaires, HTML vous ouvre les portes de l'interactivité et vous permet de recevoir des informations provenant directement de votre visiteur.

Les formulaires présentent l'avantage, par rapport au simple courrier électronique, de pouvoir prédéfinir ou de structurer l'information qui vous intéresse. Cette prédéfinition des données est très utilisée dans les applications commerciales du Web.

Pour exploiter les données retournées par le formulaire, il y a plusieurs modes de traitements possibles :

· Les CGI :

Un CGI [Common Gateway Interface] est écrit dans un langage de programmation (C, C++, Perl...). Ce programme est chargé sur le disque dur du serveur, dans un répertoire bien déterminé qui (généralement) s'appelle cgi-bin.
L'accès à ce répertoire est limité par l'administrateur du serveur, pour des raisons de sécurité, aux seuls utilisateurs habilités.
Une fois chargé sur le serveur, on peut y accéder de façon classique par l'URL "http:// www.nom du serveur/ cgi-bin/ mon_cgi.pl". Le programme de la CGI s'effectue en utilisant les ressources de l'ordinateur serveur, (avec, si le programme est mal conçu, un risque de plantage complet du serveur).

C'est la solution dite professionnelle car elle permet de bâtir de véritables applications autour des formulaires et du traitement des informations.

· Mailto :
Mailto permet d'utiliser une autre ressource d'Internet disponible pour l'utilisateur, c'est le courrier électronique.
Netscape avec Navigator 3 a imaginé le premier ce moyen d'exploiter un formulaire qui présente l'énorme avantage de pouvoir être utilisé par tout un chacun. Depuis, cette possibilité a été reprise par Microsoft Explorer 4 et bien entendu Netscape 4 (Communicator).

Attention : mailto, ne fonctionne pas sous Microsoft Explorer 3.0.
· Javascript :
le Javascript permet de traiter et de transférer les informations du formulaire à l'intérieur d'une page ou d'un site Web.

En outre, Javascript, propose des outils particulièrement adaptés pour la vérification des données introduites par l'utilisateur dans les formulaires avant l'envoi et le traitement de celles-ci.

Définition d'un formulaire

Définition :

Avant de pouvoir utiliser les différents éléments d'un formulaire (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" name="mon_formulaire"> ... les éléments du formulaire ... </FORM>

· 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 un CGI, on spécifie le répertoire CGI du serveur et le nom du CGI.
<FORM method="post" action="http://www.serveur/cgi-bin/mon_cgi.pl">

Dans le cas d'un envoi vers en adresse électronique (e-mail), on utilise le protocole mailto: suivi de l'adresse électronique du destinataire (généralement votre adresse e-mail).
<FORM method="post" action="mailto:raphael.hayem@devinci.fr"> (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 ne faut pas oublier de nommer votre formulaire surtout si il y en a plusieurs sur la même page.

· 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.

Tous les éléments d'un formulaire qui vont suivre sont obligatoirement placés entre <FORM> </FORM> afin de pouvoir fonctionner.

Ligne de texte

INPUT type="text" indique un champ de saisie d'une seule ligne.

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



Commentaires

· L'attribut name="nom du formulaire" est obligatoire car 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.

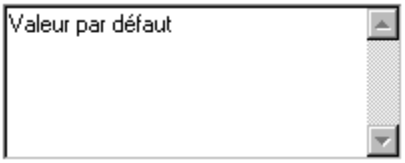
· Il existe l'attribut maxlength="x" (optionnel) qui limite le nombre réel de caractères que l'on peut introduire dans le champ de saisie.

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

Zone de saisie

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>
```



Commentaires :

· 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, mais très utile) détermine la façon dont les sauts de lignes 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 lignes.

· Attention !!! La balise <TEXTAREA> a une balise de fin, soit </TEXTAREA>.

Liste déroulante

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>
```



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



Commentaires :

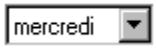
· L'attribut name="nom" définit le nom de la chaine retournée.

· L'attribut size="x" détermine le nombre d'éléments de liste affiché dans la boite 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 boite 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>
```



Bouton d'option [Radio button]

Les boutons d'option, aussi appelés boutons radio, ont comme particularité qu'une seule option à la fois peut être activée.

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

Commentaires :

· 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 le formulaire.

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

Cases à cocher [checkbox]

Les cases à cocher sont assez similaires aux boites d'option, cependant, plusieurs choix simultanés peuvent être réalisés.

La syntaxe de base est :

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

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

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.

· 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 script de traitement.

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

Les boutons Submit et Reset

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>



Commentaires :

· Seul le texte du bouton pourra être modifié (par défaut Submit).

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

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>



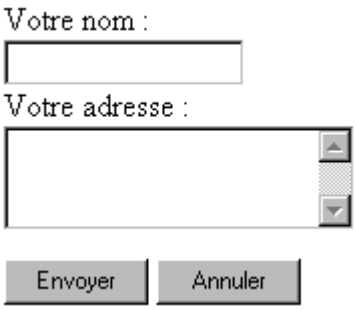
Commentaires :

· 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.

Exemple d'un petit formulaire.

```
<FORM method="post" action="mailto:votre_adresse_E-mail">
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>
```



Vous recevrez dans votre boite de réception ::

nom=Raf &adresse=pole+universitaire+léonard +2217OD%OA7700+de+vinci

. 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.

Les frames

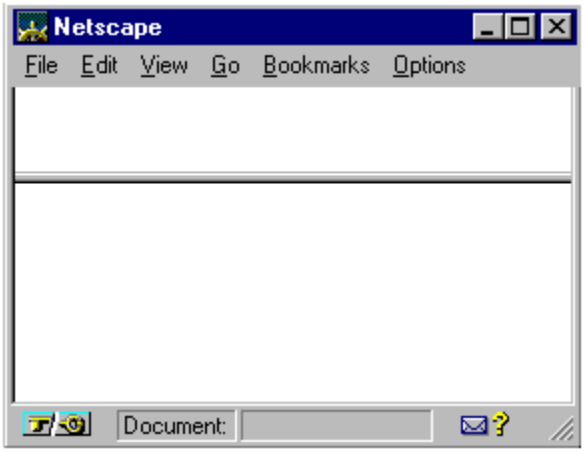
Pour diviser l'écran en plusieurs fenêtres, les balises sont peu nombreuses:

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

Nous allons procéder par exemple :

Pour créer un code frame Il faut employer les balises suivantes :

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

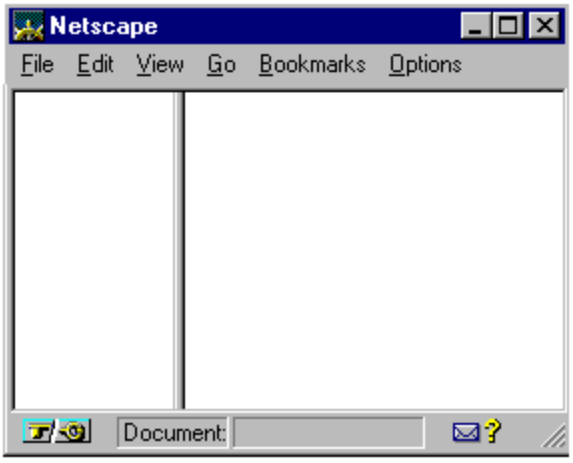


Attention! <FRAMESET></FRAMESET> remplace <BODY></BODY>

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 à100%;

Le même exemple pour un agencement vertical

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

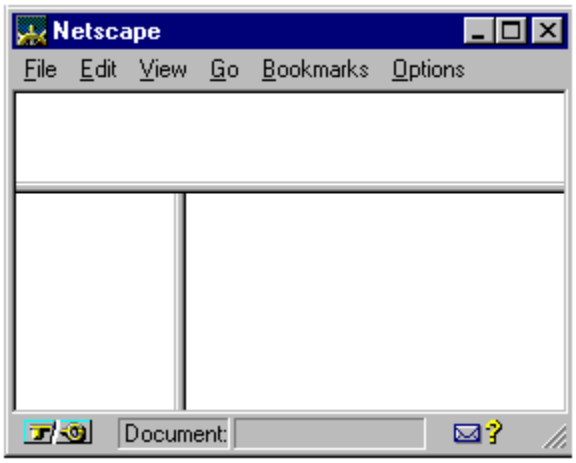


L'attribut COLS="largeur1,largeur2,...,largeurN" définit la largeur des différentes fenêtres en cas de division verticale.
La largeur s'exprime en pixels ou en %. Dans ce cas, on veillera à ce que le total soit égal à100%;

On peut mélanger les deux exemples :

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

</FRAMESET>



Pour l'instant, nos frames sont vides. On va donc ajouter des attributs à la balise <FRAME>:

<FRAME SRC="URL ou adresse du document à afficher dans la fenêtre">

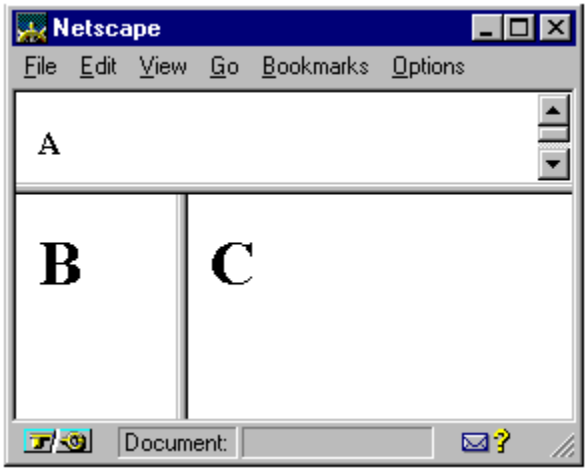
On construit 3 fichiers Html élémentaires que l'on place dans le même répertoire que le fichier de frames :

A.htm B.htm C.htm

<HTML><BODY><B>A</B></BODY></HTML> <HTML><BODY><B>B</B></BODY></HTML>
<HTML><BODY><B>C</B></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.

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 veux 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.

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>
```

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.