



Le Langage HTML

Hyper Texte Markup Language



Table des matières

AVANT PROPOS.....	3
LA P'TITE HISTOIRE.....	3
LE SGML	3
COMMENTAIRES	5
DOCTYPE	5
HTML.....	5
SÉPARATEURS.....	8
POINTEURS VERS AUTRES DOCUMENTS	9
LISTES	12
STYLE	15
LES COULEURS UTILISÉES DANS BODY	17
IMAGES DANS LE TEXTE	18
FORMULAIRES	21
LES FRAMES	25

Avant propos

Première chose importante à savoir sur **HTML** est la signification de ces quatre initiales :

Hyper **T**ext **M**arkUp **L**anguage

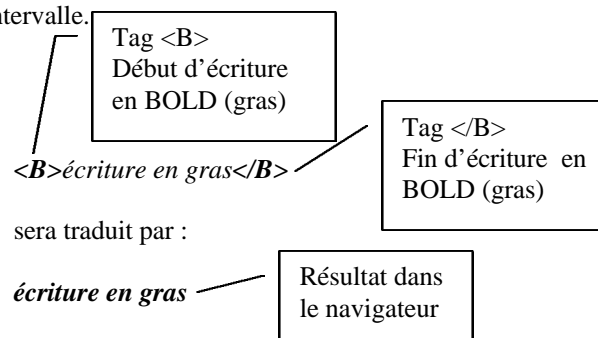
HTML est d'un langage de description (et non pas d'un langage de programmation) qui va nous permettre de décrire l'aspect d'un document, d'y inclure des informations variées (textes, images, sons, animations etc.) et d'établir des relations cohérentes entre ces informations grâce aux liens hypertextes.

Les avantages du langage HTML sont nombreux :

- ☞ peu coûteux en effet un simple éditeur de texte suffit à écrire ses premiers documents HTML
- ☞ relativement facile à aborder,
- ☞ il représente en outre un bon moyen de dépasser les problèmes de compatibilité entre des systèmes et des formats informatiques différents.

La description d'un document HTML passe par l'utilisation de **BALISES** (ou "TAGS" en anglais). Une balise est délimitée par les signes "<" et ">" entre lesquels figure le nom de la balise. Par exemple, la balise de retour à la ligne est
 La plupart du temps, on utilise une balise de début et une balise de fin, qui définissent les propriétés de l'intervalle.

Exemple :



La p'tite histoire

Le terme markup se réfère aux marques, aux annotations manuscrites placées par l'auteur sur un document pour préciser à l'imprimeur comment il doit être présenté. Avec l'apparition des ordinateurs et des photocomposeuses, ces marques ont été intégrées dans le texte mais chaque matériel de photocomposition réclamait son propre langage "markup" Au début des années 80, le CGA (Graphics Communications Association) a mis au point le premier langage markup généralisé baptisé GenCode. Au même moment, un comité de normalisation ANSI publiait le standard Generalized Markup Language (GML). En décembre 1986, les deux comités ont unis leurs efforts pour définir le standard SGML (Standard Generalized Markup Language) accepté par l'ISO (International Standard Organization) sous le numéro 8879.

Jusqu'à 1990 les principales applications Internet étaient le courrier électronique, listserv, telnet et FTP. En 1990 l'université de Mac Gill (Montréal) introduisit ARCHIE, un outil de recherche FTP pour l'Internet. En 1991 l'université du Minnesota proposa GOPHER.

En 1992, Tim Berners-Lee un physicien du CERN, développa les protocoles du World Wide Web. Il créa le HTML, issu d'un sous ensemble du SGML en s'appuyant sur les travaux de Ted Nelson qui furent à l'origine du terme *hypertexte* (1965).

C'est en 1993 que l'explosion de l'Internet eu lieu avec la sortie du navigateur (en anglais "browser") MOSAIC du NCSA.

Le SGML

Le SGML est un standard ouvert qui n'est lié ni à une plate-forme, ni à un constructeur. Les fichiers SGML sont stockés sous forme de simples textes ASCII et peuvent donc être utilisés par n'importe quelle machine.

Le langage SGML est extrêmement pratique pour réaliser des publications à la demande. Le but est de pouvoir, à partir d'un document de base unique, publier différentes versions s'adressant à des publics différents.

SGML est un langage de description de données qui divise un document en deux parties : le DTD (Data Type Definition) et les données elles-mêmes. Le DTD est une sorte de dictionnaire qui décrit les différents "tags"

acceptés dans le document et les relations qui les unissent. Le corps du document contient les données délimitées par les "tags" définis dans le DTD. Le langage HTML n'est qu'une instance du langage SGML définie par un DTD spécifique. Un document HTML est un simple fichier texte ASCII qui contient des "tags" HTML. Sur un serveur Unix, un document HTML porte généralement l'extension .html tandis que sur les PC, limités à des noms de fichiers de type 8.3, l'extension est .htm. La notion essentielle à bien comprendre avant de se lancer dans la création de pages HTML est que comme tous les langages "markup", HTML ne se soucie pas de l'apparence d'un document mais bien de sa structure. Chaque navigateur World Wide Web représentera différemment des objets HTML comme la ligne horizontale ou le titre. L'HTML n'imposera jamais une police de caractères, une taille de police ou une largeur d'écran. C'est au navigateur de choisir en fonction de la station d'accueil, de la résolution d'écran, des polices disponibles. Il faudra donc toujours veiller à essayer les pages HTML nouvellement réalisées avec plusieurs navigateurs.

Une autre notion à garder en tête est que le langage HTML est en constante évolution, chaque navigateur s'adaptant plus ou moins rapidement à cette évolution.

Chaque commande du langage HTML débute par un "<" et se termine par un ">". Les commandes peuvent s'écrire en minuscules ou majuscules bien qu'on utilise généralement les caractères majuscules par souci de lisibilité.

Allure d'un fichier HTML ou squelette d'un fichier HTML

```
<!-- commentaire -->
<!DOCTYPE>
<HTML>
  <HEAD>
    [ ]
  </HEAD>
  <BODY>
    [ ]
  </BODY>
</HTML>
```



Commentaires

Les commentaires peuvent être placés n'importe où dans un document HTML à condition de ne pas être imbriqués. Ils sont placés entre les chaînes de caractère `<!--` et `-->`.

Exemple :

```
<!-- Ceci est un commentaire -->
```

DocType

DocType est une directive SGML permettant d'identifier un document comme étant de type HTML. Cette commande est facultative. La plupart des navigateurs n'en ont pas besoin même s'il est de bon ton pour un éditeur HTML de placer en tête de document une commande de type.

Exemple :

Avec Netscape 3.0 GOLD

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 3.2//EN">
```

HTML

C'est le premier marqueur ou tag que l'on trouve dans un document HTML. Tout le document qui sera écrit par la suite (texte ou marqueur) sera compris entre le tag `<HTML>` et `</HTML>`.

Exemple :

```
<HTML>
  document à écrire
</HTML>.
```

En-tête <HEAD>

L'en-tête du document est réservée aux méta-informations (les informations relatives au document) comme son titre. Ces méta-informations doivent être placées entre les tags `<HEAD>` de début et `</HEAD>` de fin. Même si

l'en-tête est optionnelle, il est important de l'inclure dans un document pour éviter que le logiciel serveur n'ait à balayer tout le document pour y retrouver l'information recherchée.

Dans l'en-tête, on trouvera les 6 commandes suivantes :

- ☞ **TITLE** (la seule véritablement essentielle- voir ci-dessous)
- ☞ **ISINDEX** indique au logiciel client que le document est un écran permettant une saisie pour interroger un programme externe (Programme CGI), (Servait à l'origine du HTML aux browser à envoyer de l'info.-Les formulaires ont simplifiés tout),
- ☞ **BASE** permet d'indiquer une adresse de base qui complétera tous les chemins relatifs,
- ☞ **LINK** (Indique un lien entre le présent document et un autre) ; très rarement utilisé.
- ☞ **NEXTID** (Indique le document suivant – considéré actuellement comme obsolète-)
- ☞ **META** donne des informations sur le document au serveur. Ces informations sont généralement :
 - <META NAME=...> le nom,
 - <META AUTHOR=....> le nom de l'auteur,
 - <META CONTENT=.....> ,
 - <META NEXTID=....>un identifiant.

☞ <BASFONT SIZE = n> ou n prend les valeurs 1 à 7.Cette commande fixe la taille de la police de caractères (par défaut : 3) 1, petits caractères ... 7, gros caractères.

☞ **PROMPT=texte**, permet d'afficher le texte comme message pour <ISINDEX>.

Titre <TITLE>

<TITLE> Titre </TITLE>

Chaque document HTML, pour être correct, doit disposer d'un titre. Le titre apparaît le plus souvent dans la barre de titre du navigateur Web. Son but est de pouvoir identifier le document dans un ensemble plus large comme un index WAIS par exemple. Le titre d'un document HTML sert aussi lorsqu'un navigateur World Wide Web dispose d'une fonction "hotlist" ou "bookmark" pour fournir un accès rapide à vos documents favoris.

Le titre est limité à **64 caractères** y compris les espaces et ne doit normalement contenir que les 128 premiers caractères ASCII (pas de caractères accentués)

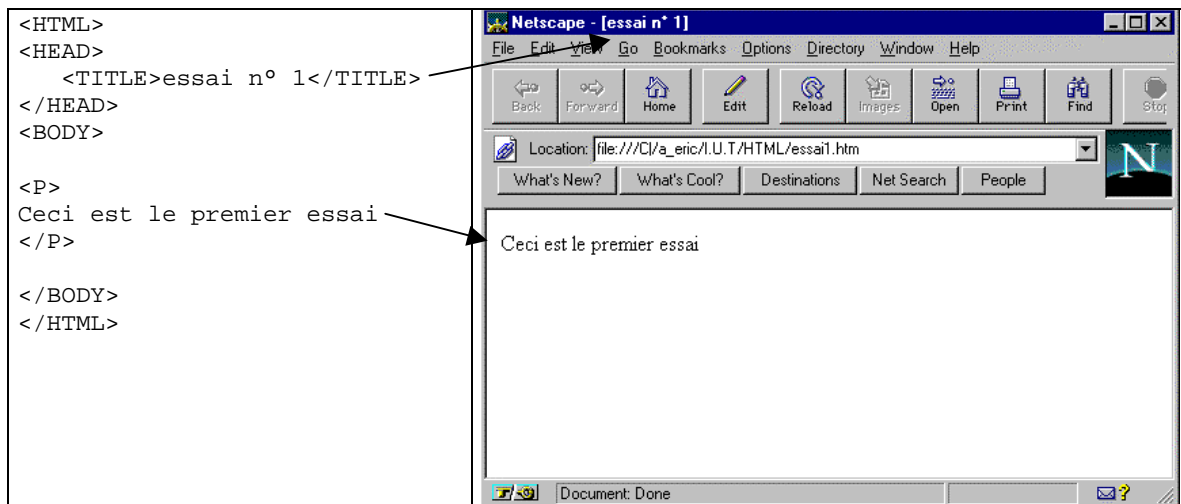
Corps du document <BODY>

Le corps du document, ce qui sera effectivement affiché par le navigateur Web, est balisé par les commandes : <BODY> et </BODY>

Exemple :

```
<BODY>
<P>
    Ceci est le premier essai
</P>
</BODY>
```

Voici donc à quoi ressemblera un document minimal ainsi que sa représentation dans un navigateur Web :



Ces commandes ne doivent pas nécessairement être séparées par des retours de chariot et peuvent être placées n'importe comment. Seule la lisibilité du code source nous fait pencher pour cette présentation.

Tailles de polices :

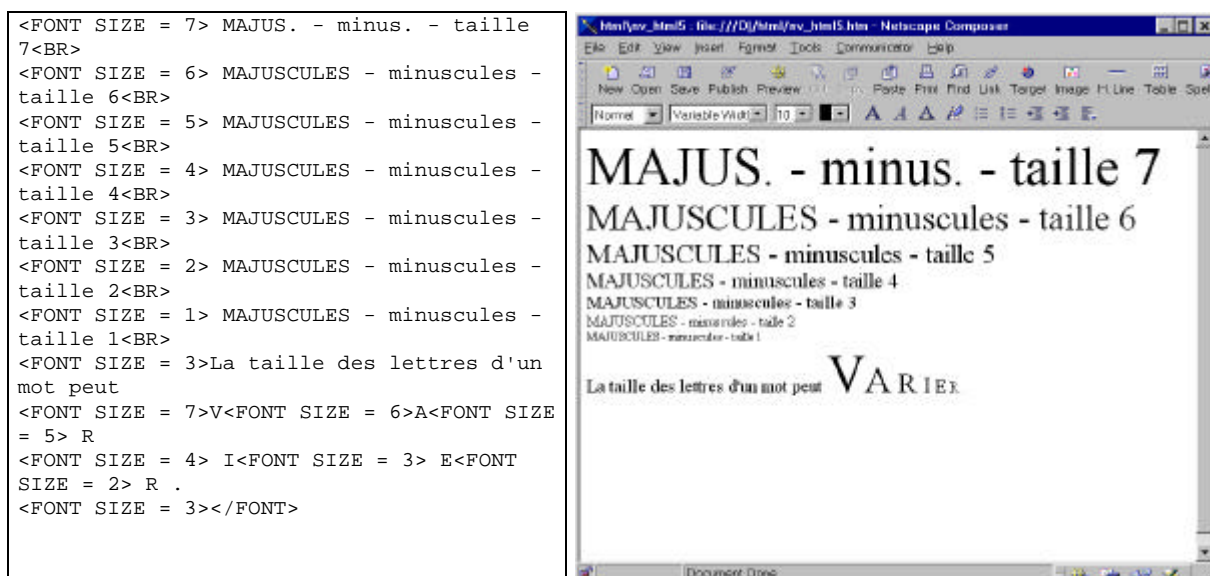
Il existe 7 tailles de polices. La commande est ** ... ** ou n prend les valeurs de 1 à 7.
 1, petits caractères
 7, gros caractères.

La police par défaut possède la taille 3.

La taille de la police peut être fixée dans l'en-tête du document par la commande :

<BASEFONT SIZE = n> ou n prend les valeurs 1 à 7.

Exemple :



Les Entêtes <Hx> avec x : 1→6

Ces entêtes servent à diviser le texte en section de la même manière qu'un livre. Le langage HTML reconnaît six niveaux de d'en-tête numérotés de 1 à 6. Le niveau le plus élevé est le 1, le plus petit est le 6.

Exemple :

```
<HTML>
<HEAD>
  <TITLE>essai n° 2</TITLE>
</HEAD>
<BODY>
<H1> TITRE PRINCIPALE </H1>
<H2> TITRE SECONDAIRE </H2>
<H3> Paragraphe principale</H3>
<H4> Paragraphe secondaire</H4>
<H5> titre Paragraphe 1 secondaire</H5>
<H6> textel, textel, textel, textel </H6>
<H5> titre Paragraphe 2 secondaire</H5>
<H6> texte2, texte2, texte2, texte2 </H6>
</BODY>
</HTML>
```

TITRE PRINCIPALE

TITRE SECONDAIRE

Paragraphe principale

Paragraphe secondaire

titre Paragraphe 1 secondaire

textel, textel, textel, textel

titre Paragraphe 2 secondaire

Séparateurs

Contrairement à tous les autres systèmes de traitement de texte, le retour de charriot n'a aucune valeur en HTML. C'est le navigateur Web lui même qui définira le passage à la ligne en fonction de facteurs comme la taille des polices de caractères, la largeur de la fenêtre de visualisation etc.

De même, plusieurs espaces dans un document HTML seront ramenés à un seul.

<P> Cette commande **<P>** (fin de paragraphe) termine un paragraphe et insère une ligne vide après le paragraphe.

Exemple :

```
Exemple de texte<P>
Avec un autre exemple<P> Fin de l'exemple.
```

**
** Cette commande impose la coupure d'une ligne de texte en rejetant ce qui suit à la ligne suivante.

<NOBR>...</NOBR> Cette commande permet de mettre du texte sans retour à la ligne.

<WBR> Cette commande permet de forcer un retour à la ligne dans un texte encadré par **<NOBR>**

<HR> pour une ligne horizontale de séparation :

WIDTH	fait varier la largeur de la ligne en % soit en pixel (valeur par défaut 100%)
SIZE	fait varier l'épaisseur de la ligne en pixel (valeur par défaut 1)
ALIGN	fait un alignement de la ligne suivant les 3 possibilités :
	CENTER par rapport au centre de la fenêtre,
	LEFT par rapport à la gauche de la fenêtre,
	RIGHT par rapport à la droite de la fenêtre.
NOSHADE	Lorsqu'il est présent dans le marqueur <HR> l'effet est une ligne pleine sans ombrage.

Le paramètre **ALIGN** a un effet que lorsque la longueur de la ligne est inférieure à 100 %.

Exemple :

<HR WIDTH=75%> trace une ligne sur 75% de la largeur de la fenêtre.

<HR SIZE="5"> trace une ligne d'épaisseur 5 pixels.

<HR WIDTH=75% ALIGN=LEFT> trace une ligne sur 75% de la largeur de la fenêtre avec un alignement à gauche.


```

<HTML>
<HEAD>
  <TITLE>essai n° 3</TITLE>
</HEAD>
<BODY>
<H4>
HR<HR>
HR WIDTH="75%"
<HR WIDTH="75%" SIZE=5>
HR WIDTH="25%"
<HR WIDTH="25%">
HR WIDTH="50%" & SIZE=5
<HR WIDTH="50%" SIZE=5>
HR WIDTH="50%" & SIZE=20
<HR WIDTH="50%" SIZE=20>
HR WIDTH="50%" & SIZE=5 ALIGN=LEFT
HR WIDTH="50%" & SIZE=5 <HR WIDTH="50%" SIZE=5
ALIGN=LEFT>
HR WIDTH="50%" & SIZE=5 ALIGN=LEFT NOSHADE
HR WIDTH="50%" & SIZE=5 <HR WIDTH="50%" SIZE=5
ALIGN=LEFT NOSHADE>
</H4>
</BODY>
</HTML>

```

```

HR
-----
HR WIDTH="75%"
-----
HR WIDTH="25%"
-----
HR WIDTH="50%" & SIZE=5
-----
HR WIDTH="50%" & SIZE=20
-----
HR WIDTH="50%" & SIZE=5 ALIGN=LEFT HR
WIDTH="50%" & SIZE=5
-----
HR WIDTH="50%" & SIZE=5 ALIGN=LEFT NOSHADE HR
WIDTH="50%" & SIZE=5
-----

```

Pointeurs vers autres documents

Grâce au langage HTML, vous pouvez créer un hyperlien vers un document local, un document distant ou un signet dans ces deux types de documents.

Anchor

L'hyperlien est créé par l'utilisation du tag <A> pour "anchor" ou ancre.

Les attributs possibles sont :

HREF

La syntaxe complète est texte

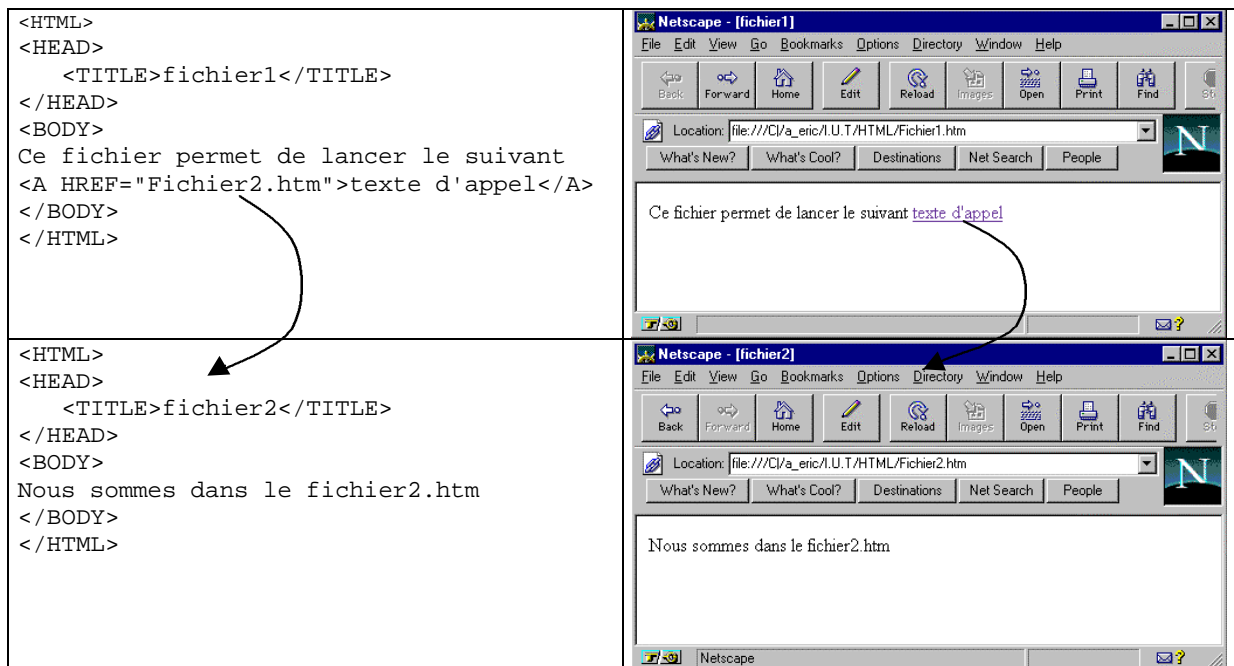
Où **HREF** est un raccourci pour "Hypertexte référence", **document** désigne le document vers lequel on pointe et **texte** représente le texte qui sera affiché pour représenter le lien hypertexte.

Exemple :

```

<A HREF="fichier2.htm"> pour charger le fichier2.htm </A>

```



Ici dans cet exemple les deux fichiers sont forcément dans le même répertoire.

Chemin d'accès relatif :

Dans ce cas, relativement par rapport à l'endroit où vous vous situez dans une arborescence, vous exprimez le chemin d'accès vers le fichier lié.

HREF= "fichier2.htm "	Dans le répertoire courant
HREF= "tempo/version1/fichier2.htm "	Dans le répertoire 2 niveaux en dessous
HREF= "../fichier2.htm "	Dans le répertoire 2 niveaux au dessus

Chemin d'accès absolu :

Dans ce cas, depuis la racine des disques nommés, vous exprimez le chemin d'accès vers le fichier lié.

HREF= "/rep1/fichier2.htm "	Dans le répertoire rep1 de l'unité en cours
HREF= "/d:/html/fichier2.htm "	Dans le répertoire html du disque d
HREF= "~/brassart/fichier2.htm "	Dans le répertoire d'accueil de l'utilisateur brassart sur un système UNIX

Lien vers un autre site distant

Pour cela il suffit de remplacer dans l'exemple précédent *document* par une U.R.L.

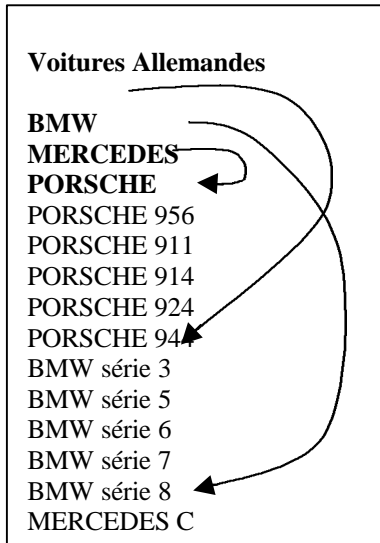
Exemple :

```
<A HREF=http://amiens.com >page d'accueil de la ville d'Amiens</A>
```

Création d'un lien interne vers un endroit précis d'un document

Un lien interne pointe vers une ancre, c'est à dire un endroit à l'intérieur d'un document défini par un nom. Une ancre se définit dans le marqueur <A> en y ajoutant le paramètre **NAME="nom"**. Vous pouvez définir une table des matières au début d'un long document en utilisant ce type de lien.

Exemple :



```
<HTML>
<HEAD>
  <TITLE>voitures allemenades</TITLE>
</HEAD>
<BODY>
<H1>voitures allemandes</H1>
<A HREF="#BMW"><H2>BMW</H2></A>
<A HREF="#MERCEDES"> <H2>MERCEDES</H2></A>
<A HREF="#PORSCHE"><H2>PORSCHE</H2></A>
<BR><BR>
<A NAME="MERCEDES"></A>
MERCEDES C<BR>
MERCEDES E<BR>
<A NAME="PORSCHE"></A>
PORSCHE 956<BR>
PORSCHE 911<BR>
PORSCHE 914<BR>
PORSCHE 924<BR>
PORSCHE 944<BR>
<A NAME="BMW"></A>
BMW série 3<BR>
BMW série 5<BR>
BMW série 6<BR>
BMW série 7<BR>
BMW série 8<BR>
</BODY>
</HTML>
```

Création d'un lien externe vers un endroit précis d'un document

Il est également possible d'utiliser les ancres dans les liens externes. Il faut alors spécifier l'ancre vers laquelle pointe le lien en ajoutant #nom à la fin de l'URL.

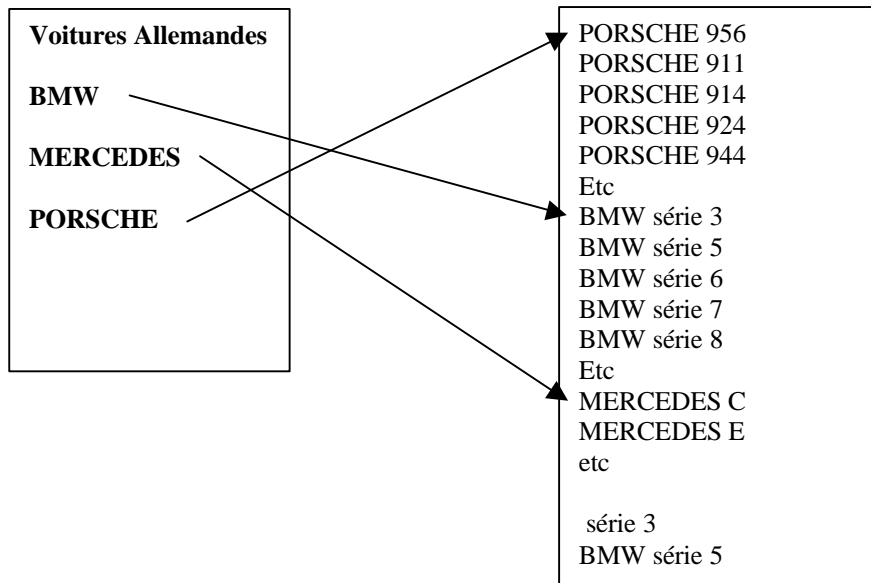
Dans le fichier1.htm

:

```
<A REF="fichier2.htm#BMW">BMW</a>
```

et dans le fichier2.htm à l'endroit ou l'on veut pointer :

```
<A NAME="BMW">
```



Liens externes vers d'autres protocoles/services Internet

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

Pour que la personne qui lit votre document puisse vous répondre sur un simple click de souris vous pouvez insérer un lien qui lance automatiquement l'application de messagerie électronique en incluant le nom du destinataire. De même, il est possible d'emmener le lecteur vers un groupe de discussions en mentionnant le nom de celui-ci dans la définition du lien.

Exemple de lien pour l'envoi d'un E-mail

```
<A HREF="mailto:Eric.Brassart@u-picardie.fr">Envoyez moi un message</A>
```

Exemple de lien pour la lecture d'un newsgroup

```
<A HREF="news:comp.info.systems.www.authoring.html">groupe de discussion sur l'édition en HTML</A>.
```

Liens pour le téléchargement d'un fichier

Ce type de lien s'utilise pour donner accès à un dialogue de téléchargement de fichier. La syntaxe est semblable à celle du premier type de liens que nous avons vu. Le seul changement est le protocole indiqué dans l'URL qui, en l'occurrence, est le protocole de transfert de fichiers *ftp*.

Exemple de lien pour le téléchargement de fichiers

```
<A HREF="ftp://tecfa.unige.ch/pub/software/">Liste des logiciels disponibles</A>
```

Listes

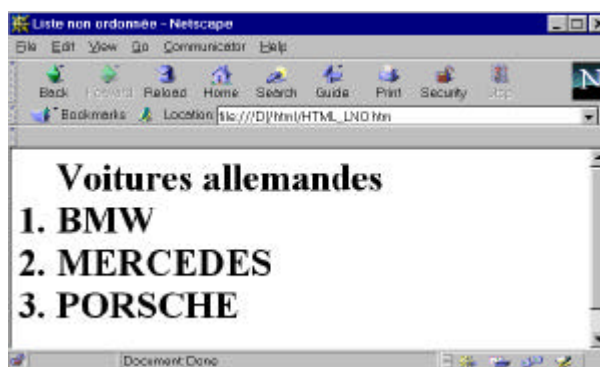
HTML définit 5 types de listes :

- ☞ Les listes numérotées ou ordonnées (ou Ordered (numbered) lists),
- ☞ Les listes non numérotées dites listes à puces dont les entées sont signalées par un signe typographique,
- ☞ Les listes de glossaire,
- ☞ Les listes de menu,
- ☞ Les listes de répertoire.

Listes numérotées ou Ordered (numbered) lists

Exemple :

```
<HTML>
<HEAD>
  <TITLE>Liste non ordonnée</TITLE>
</HEAD>
<BODY>
<H1>
<ol>
  <lh>Voitures allemandes
  <li>BMW
  <li>MERCEDES
  <li>PORSCHE
</ol>
</H1>
<BODY>
</HTML>
```



Les options suivantes sont possibles :

- ☞ TYPE=1 : pour une liste numérotée 1,2,3...
- ☞ TYPE=A : pour un repérage type A,B,C...
- ☞ TYPE=a : pour un repérage type a,b,c...
- ☞ TYPE=I : pour une liste numérotée I,II,III,IV...
- ☞ TYPE=i : pour une liste numérotée i,ii,iii,iv...

START=n fait débiter le repérage (chiffres ou lettres) au rang numéro n.

Remarque : TYPE=1 est utilisé par défaut.

Listes non numérotées ou Unordered lists

Exemple

```
<HTML>
<HEAD>
  <TITLE>Liste non ordonnée</TITLE>
</HEAD>
<BODY>
<H1>
<ul>
  <lh>Voiture allemandes
  <li>BMW
  <li>MERCEDES
  <li>PORSCHE
</ul>
</H1>
</BODY>
</HTML>
```



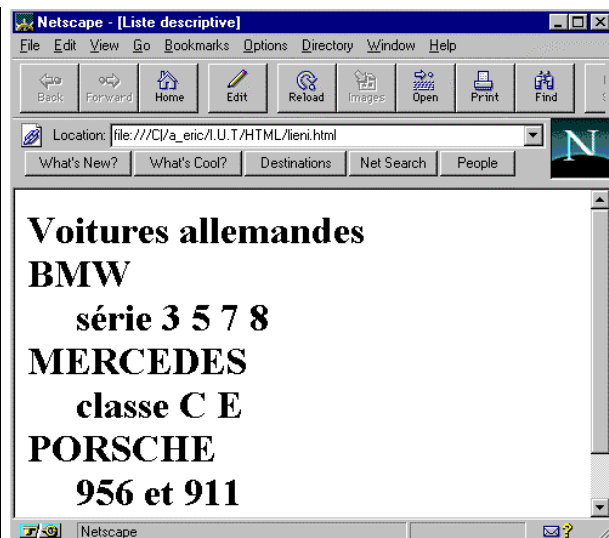
Listes descriptives ou Definition lists <DL> <DT> <DD>

L'environnement <DL> délimite une zone de liste de définition.

La commande <DT> introduit un nouveau terme de définition. C'est en général un élément court.

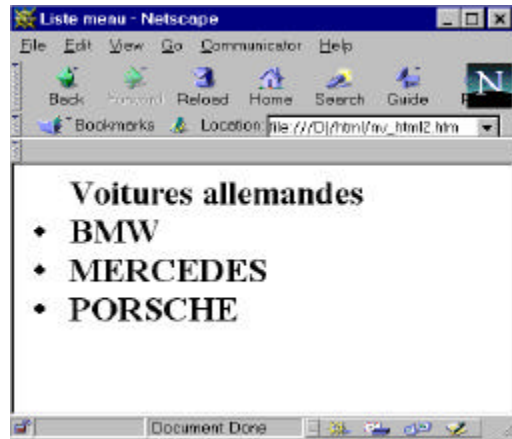
La commande "dd" introduit une description du terme de définition. Le résultat à l'écran est un décalage du texte vers la droite.

```
<HTML>
<HEAD>
  <TITLE>Liste descriptive</TITLE>
</HEAD>
<BODY>
<H2>
<DL>
  <lh>Voitures allemandes
  <DT>BMW
    <DD>série 3 5 7 8
  <DT>MERCEDES
    <DD> classe C E
  <DT>PORSCHE
    <DD> 956 et 911
</DL>
</H2>
</BODY>
</HTML>
```



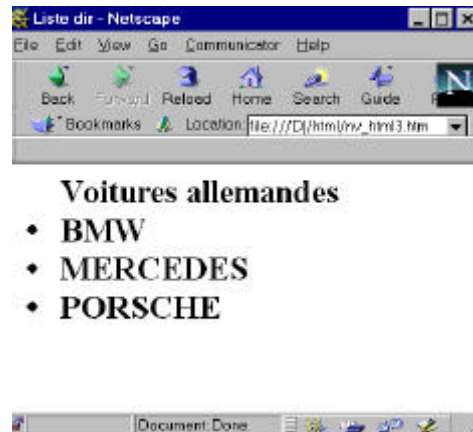
Menus

```
<HTML>
<HEAD>
  <TITLE>Liste menu</TITLE>
</HEAD>
<BODY>
<H2>
<MENU>
  <lh>Voitures allemandes
  <LI>BMW
  <LI>MERCEDES
  <LI>PORSCHE
</MENU>
</H2>
<BODY>
<HTML>
```



Directory List

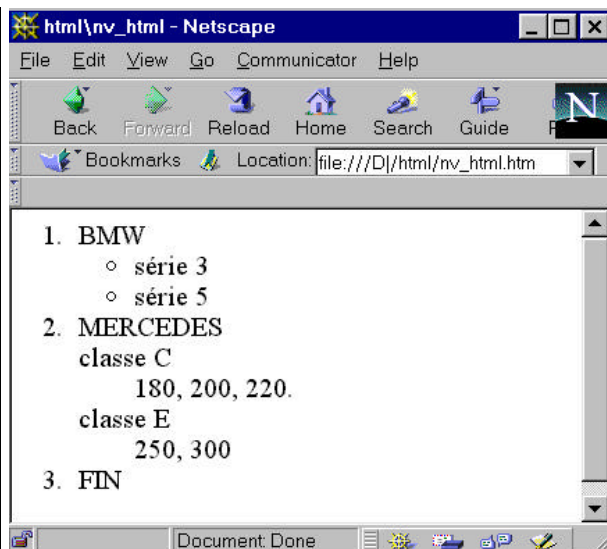
```
<HTML>
<HEAD>
  <TITLE>Liste dir</TITLE>
</HEAD>
<BODY>
<H2>
<DIR>
  <lh>Voitures allemandes
  <LI>BMW
  <LI>MERCEDES
  <LI>PORSCHE
</DIR>
</H2>
<BODY>
<HTML>
</BODY>
</HTML>
```



Deux types de liste peuvent s'emboîter

Exemple d'emboîtement :

```
<HTML>
<HEAD>
  <TITLE>html\nv_html</TITLE>
</HEAD>
<OL>
  <LI> BMW
  <UL>
    <LI> série 3
    <LI> série 5
  </UL>
  <LI> MERCEDES
  <DL>
    <DT>classe C
    <DD>180, 200, 220.
    <DT> classe E
    <DD>250, 300
  </DL>
  <LI> FIN
</OL>
</BODY>
</HTML>
```



Style

L'utilisation de différents styles de polices de caractères permet de varier la présentation d'un texte. En HTML, on peut utiliser des styles logiques où le nom du style indique la nature du fragment de texte à écrire dans ce style ou des styles physiques où le nom du style indique explicitement le style de police que l'on souhaite voir utiliser.

Styles de caractères :

Styles logiques : à utiliser de préférence car ils sont interprétés par les différents logiciels clients WWW.

Les commandes de styles logiques sont les suivantes :

<code></code> texte <code></code>	met le texte généralement en italique.
<code></code> texte <code></code>	met le texte généralement en gras.
<code><CODE></code> texte <code></CODE></code>	pour l'utilisation d'une police à chasse fixe (encombrement des caractères constant).
<code><SAMP></code> caractères <code></SAMP></code>	séquence de caractères littéraux.
<code><KBD></code> saisie <code></KBD></code>	pour mettre en évidence une saisie d'utilisateur.
<code><VAR></code> variable <code></VAR></code>	pour indiquer le nom d'une variable.
<code><DFN></code> définition <code></DFN></code>	pour mettre en évidence la 1ère utilisation d'un terme.
<code><CITE></code> citation <code></CITE></code>	pour mettre en évidence une citation (généralement en italique).
<code><ADDRESS></code> adresse <code></ADDRESS></code>	cette commande n'est pas utilisée pour une adresse postale. Cette commande est généralement utilisée pour indiquer l'auteur d'un document ainsi que le moyen de le contacter ou bien elle donne l'adresse du document. Elle est souvent placée en fin de document.
<code><BLOCKQUOTE></code> texte <code></BLOCKQUOTE></code>	cette commande constitue avec le texte un texte avec retrait à gauche et à droite.
<code><BLINK></code> texte <code></BLINK></code>	pour faire clignoter le texte.
<code><STRIKE></code> texte barré <code></STRIKE></code>	cette commande permet de barrer du texte.
<code><LISTING></code> code informatique <code></LISTING></code>	cette commande permet d'afficher du code informatique
<code><BIG></code> texte <code></BIG></code>	cette commande permet d'écrire un texte en plus gros caractères que les caractères en cours.
<code><SMALL></code> texte <code></SMALL></code>	cette commande permet d'écrire un texte en plus petits caractères que les caractères en cours.
<code><SUP></code> texte <code></SUP></code>	cette commande permet d'écrire un texte en exposant.
<code><SUB></code> texte <code></SUB></code>	cette commande permet d'écrire un texte en indice.

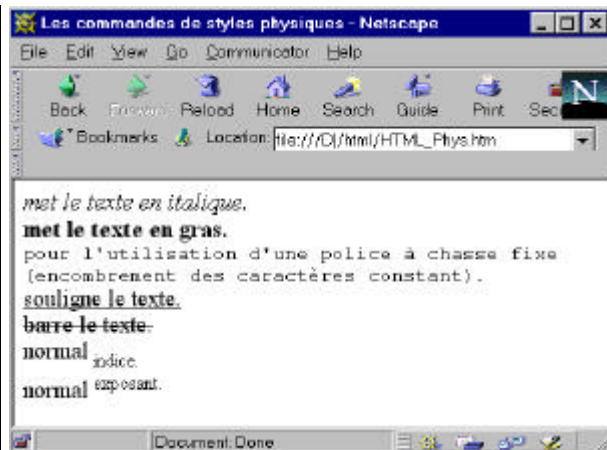
Styles physiques :

Les commandes de styles physiques sont les suivantes :

<code><I></code> texte <code></I></code>	met le texte en italique.
<code></code> texte <code></code>	met le texte en gras.
<code><TT></code> texte <code></TT></code>	pour l'utilisation d'une police à chasse fixe (encombrement des caractères constant).
<code><U></code> texte <code></U></code>	souligne le texte.
<code><S></code> texte <code></S></code>	barre le texte.
<code><SUB></code> texte <code></SUB></code>	indice.
<code><SUP></code> texte <code></SUP></code>	exposant.

Exemple de commandes de styles physiques :

```
<HTML>
<HEAD>
  <TITLE>Les commandes de styles
  physiques</TITLE>
</HEAD>
<BODY>
<I> met le texte en
italique.</I><BR>
<B> met le texte en gras.</B><BR>
<TT> pour l'utilisation d'une police
à chasse fixe (encombrement des
caractères constant).</TT><BR>
<U> souligne le texte.</U><BR>
<S> barre le texte.</S><BR>
normal<SUB> indice. </SUB><BR>
normal<SUP> exposant. </SUP><BR>
</BODY>
</HTML>
```



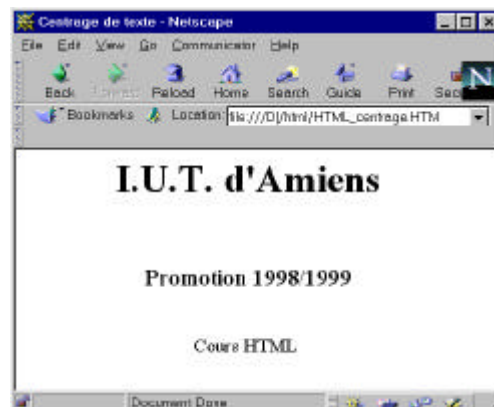
Attention: la correspondance que vous constatez peut-être entre styles logiques et styles physiques n'est pas garantie. Cela dépend du navigateur.

CENTRAGE D'UN TEXTE <CENTER>...</CENTER>

La commande <CENTER> ... </CENTER> permet de centrer toutes les lignes d'un texte.

Exemple :

```
<HTML>
<HEAD>
  <TITLE>Centrage de texte</TITLE>
</HEAD>
<BODY>
  <CENTER>
    <H1>I.U.T. d'Amiens</H1><BR>
    <H3>Promotion 1998/1999</H3><BR>
    Cours HTML
  </CENTER>
</BODY>
</HTML>
```



TEXTE PRÉ-FORMATÉ <PRE>...</PRE>

Les espaces (plusieurs à la suite), tabulations, retour chariot n'ont pas de valeur en HTML.

La commande <PRE> ... </PRE> est utilisée pour inclure un texte pré-formaté dans un document HTML.

Les espacements, tabulations et retour chariot gardent alors leur sens initial.

Attention : les commandes HTML existant dans le texte pré-formaté seront interprétées.

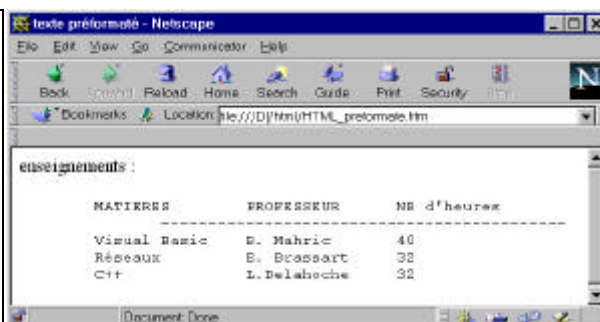
L'option WIDTH=n peut-être utilisée pour limiter la longueur de la ligne.

Exemple :


```

<HTML>
<HEAD>
  <TITLE>texte préformaté</TITLE>
</HEAD>
<BODY>
enseignements : <BR><BR>
<PRE>
  MATIERES          PROFESSEUR      NB d'heures
  -----
  Visual Basic     B. Mahric        40
  Réseaux          E. Brassart      32
  C++              L.Delahoche      32
</PRE>
</BODY>
</HTML>

```



Les couleurs utilisées dans BODY

Par défaut le fond du document est gris clair, les caractères sont noirs, les prises d'hypertextes sont bleues quand elles n'ont jamais été utilisées, violettes dans le cas contraire, rouges à l'instant de la sélection (dans le cas de NETSCAPE).

Il est possible de modifier ces couleurs en rajoutant à la commande les options suivantes :

- ☞ BGCOLOR=c pour le fond du document
- ☞ TEXT=c pour la couleur des caractères
- ☞ LINK=c pour la couleur des prises d'hypertextes non utilisées
- ☞ VLINK=c pour la couleur des prises d'hypertextes utilisées
- ☞ ALINK=c pour la couleur des prises d'hypertextes à l'instant de la sélection

La valeur de c est composée de trois nombres hexadécimaux accolés (codés de 00 à FF) représentant le mélange des trois couleurs primaires RGB (Red, Green, Blue). Le nombre obtenu est précédé d'un dièse (#). Le blanc à pour valeur #FFFFFF ; le noir à pour valeur #000000. Toutes les autres couleurs sont obtenues par des dosages précis dans chacune des composantes RVB.

Exemple de couleur :

avec c = #000000	pour noir
avec c = #FFFFFF	pour blanc
avec c = #C0C0C0	pour gris clair
avec c = #80FF80	pour vert clair
avec c = #FFFFE8	pour jaune clair
avec c = #80FFFF	pour bleu clair
avec c = #FF8080	pour rose
avec c = #FF80C0	pour mauve
avec c = #800000	pour marron
avec c = #000080	pour bleu
avec c = #008040	pour vert
avec c = #FF0000	pour rouge
avec c = #FF8000	pour jaune
avec c = #800080	pour violet

Exemple d'utilisation des couleurs :

```

<HTML>
<HEAD>
  <TITLE> Essai de couleurs </TITLE>
</HEAD>
<BODY BGCOLOR=#80FFFF TEXT=#000080 LINK=#FF0000>
  La couleur du fond est bleu clair, la couleur du texte est bleu. La prise d'hypertexte est
  rouge.
  Retour : cliquer <A HREF="html.doc.htm#para4"> ici </A>
</BODY>
</HTML>

```

☞ BACKGROUND="nom_image.gif"

L'image peut être de type GIF ou JPG. Elle est le plus souvent de petite taille (100x100pixels) qui est reproduite par effet mosaïque sur tout l'écran. Il faut donc choisir soigneusement le motif pour que les bords droit et gauche d'une part et haut et bas d'autre part se raccorde sans trop de discontinuité.

Exemple d'utilisation des images de fond :

```
<HTML>
  <HEAD>
    <TITLE> Essai de couleurs </TITLE>
  </HEAD>
  <BODY BACKGROUND="image.jpg" TEXT=#000080 LINK=#FF0000>
    L'image du fond s'appelle image.jpg, la couleur du texte est bleu. La prise d'hypertexte
    est rouge.
  </BODY>
</HTML>
```

IMAGES DANS LE TEXTE

Des images peuvent être insérées dans le texte d'un document HTML, et elles sont au format **GIF** ou **JPG**.

Elles peuvent servir de prises d'hypertextes :

☞ soit toute l'image réagit à un clic.

☞ soit l'image réagit en fonction de la zone où le clic s'est produit (on parle dans ce cas là d'IMAGE CLIQUABLE OU REACTIVE.)

Les commandes sont les suivantes :

☞ **** pour insérer une image locale, où nom_image.gif est le nom complet du fichier avec éventuellement son chemin relatif à partir du répertoire du document HTML.

☞ **** pour insérer une image distante. URL étant l'adresse complète du fichier image.

☞ **** pour effectuer le lien sur le point de branchement de nom : *etiquette*, du document courant, en cliquant sur l'image nom_image.gif qui est la prise d'hypertexte. (nom_image.gif est le nom complet du fichier image).

☞ **** pour une image cliquable (ou réactive). Les images cliquables permettent des liens selon l'endroit du "clic" de la souris dans l'image. La requête est transmise à un programme exécutable (souvent IMAGEMAP.EXE) qui associe les coordonnées du "clic" à l'adresse correspondante.

Pour ces commandes les options suivantes sont possibles :

ALIGN=left/right/top/texttop/middle/absmiddle/baseline/bottom/absbottom

☞ left : positionne l'image à gauche de l'écran.

☞ right : positionne l'image à droite de l'écran.

☞ top : positionne le sommet de la ligne de texte au sommet de l'image.

☞ texttop : positionne l'axe horizontal de la ligne de texte au sommet de l'image.

☞ middle : positionne la base de la ligne de texte au milieu de l'image.

☞ absmiddle : positionne l'axe horizontal de la ligne de texte au milieu de l'image.

☞ baseline : positionne la base de la ligne de texte à la base de l'image.

☞ bottom : positionne la base de la ligne de texte à la base de l'image.

☞ absbottom : positionne la ligne de texte au bas de l'image.

WIDTH=n ou n% HEIGHT=m ou m% redimensionnent l'image : n et m valeurs de la largeur et de la hauteur de l'image en pixels ou n% et m% valeurs relatives par rapport à la taille normale de l'image.

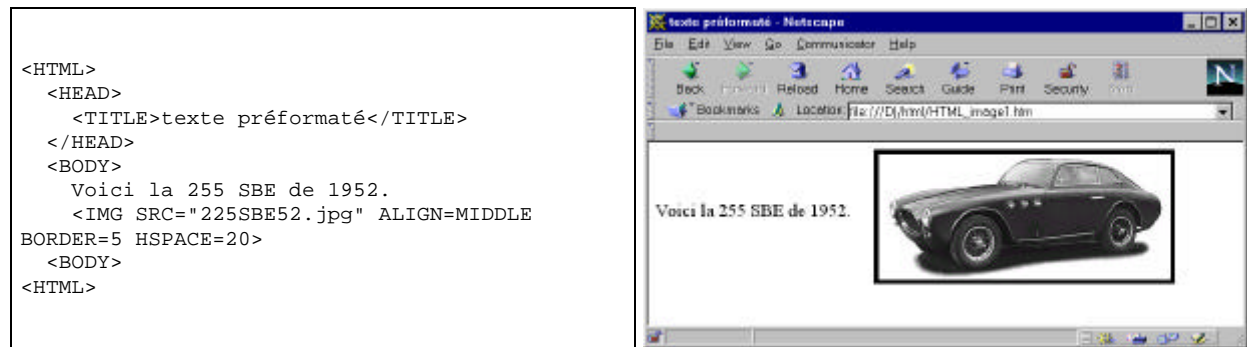
VSPACE=n HSPACE=m éloignent le texte de l'image de n pixels au-dessus et au-dessous de l'image et de m pixels à gauche et à droite de l'image.

BORDER=n trace un cadre autour de l'image avec un trait de n pixels de large.

ALT="texte" le "texte" est une alternative à l'image lorsque le logiciel client WWW utilisé ne peut afficher l'image.

Plusieurs options peuvent être utilisées ensemble.

Exemple :



TABLEAUX **<TABLE> ... </TABLE>**

Les commandes sont les suivantes :

<TABLE> ... </TABLE> pour créer un tableau.

Pour cette commande les options suivantes sont possibles :

BORDER : trace un cadre en trait fin

BORDER=n : trace un cadre en trait de n pixels d'épaisseur

CELLSPACING=n : espacement de n pixels entre les cellules.

CELLPADDING=n : espacement autour de l'écriture dans les cellules.

WIDTH=n ou n% : largeur en pixels ou largeur relative du tableau.

Remarque

cellspacing et cellpadding ont priorité sur width.

<CAPTION> nomtableau</CAPTION> pour indiquer le nom du tableau.

Pour cette commande l'option suivante est possible :

ALIGN=top/bottom : place le nom du tableau au-dessus/au-dessous du tableau.

<TR> ... </TR> encadre une ligne du tableau.

Pour cette commande les options suivantes sont possibles :

ALIGN=left/right/center : position horizontale (gauche/droite/centre) du texte dans les cellules de la ligne.

VALIGN=top/middle/bottom/baseline : position verticale (haut/milieu/bas/bas) du texte dans les cellules de la ligne.

<TH> ... </TH> encadre une cellule d'en-tête du tableau. (cellule pouvant contenir un texte alphanumérique, une image, une liste, un lien, un autre tableau ou rien)(texte en gras).

Pour cette commande les options suivantes sont possibles :

ALIGN=left/right/center : position horizontale (gauche/droite/centre) du texte dans la cellule d'en-tête.

COLSPAN=n : fusionne n cellules horizontalement.

ROWSPAN=n : fusionne n cellules verticalement.

NOWRAP : supprime la césure éventuelle du texte de la cellule d'en-tête.

<TD> ... </TD> encadre une cellule du tableau. (cellule pouvant contenir un texte alphanumérique, une image, une liste, un lien, un autre tableau ou rien)

Pour cette commande les options suivantes sont possibles :

ALIGN=left/right/center : position horizontale (gauche/droite/centre) du texte dans la cellule.

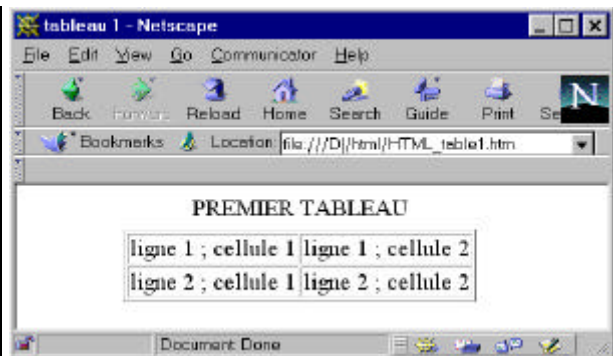
COLSPAN=n : fusionne n cellules horizontalement.

ROWSPAN=n : fusionne n cellules verticalement.

NOWRAP : supprime la césure éventuelle du texte de la cellule.

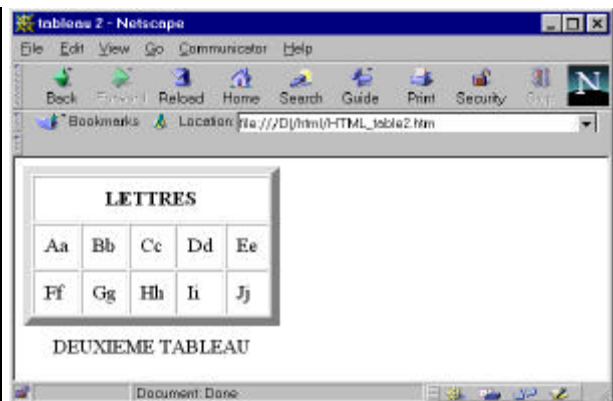
Exemple n°1 :

```
<HTML>
<HEAD>
  <TITLE>tableau 1</TITLE>
</HEAD>
<BODY>
  <CENTER>
    <TABLE BORDER>
      <CAPTION ALIGN=top> PREMIER TABLEAU</CAPTION>
      <TR>
        <TD>ligne 1 ; cellule 1</TD>
        <TD>ligne 1 ; cellule 2</TD>
      </TR>
      <TR>
        <TD>ligne 2 ; cellule 1</TD>
        <TD>ligne 2 ; cellule 2</TD>
      </TR>
    </TABLE>
  </CENTER>
</BODY>
</HTML>
```



Exemple n°2 :

```
<HTML>
<HEAD><TITLE>tableau 2</TITLE></HEAD>
<Body>
  <TABLE BORDER=8 CELLPADDING=10>
    <CAPTION ALIGN=bottom>DEUXIEME TABLEAU
  </CAPTION>
  <TR>
    <TH COLSPAN=5>LETTRES</TH>
  </TR>
  <TR>
    <TD>Aa</TD><TD>Bb</TD><TD>Cc</TD><TD>Dd</TD><TD>Ee</TD>
  </TR>
  <TR>
    <TD>Ff</TD><TD>Gg</TD><TD>Hh</TD><TD>Ii</TD><TD>Jj</TD>
  </TR>
</TABLE>
```



```
<BODY>
<HTML>
```

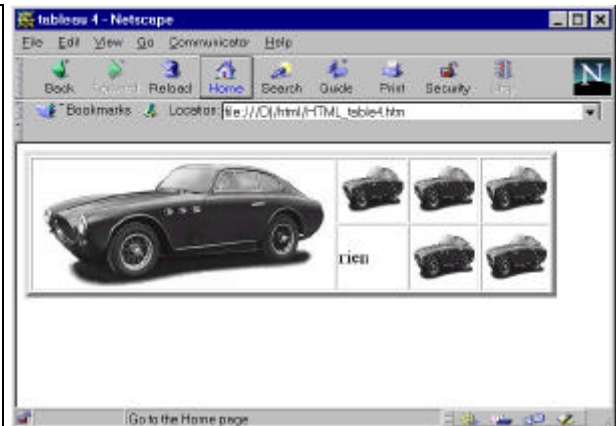
Exemple n°3 :

```
<HTML>
<HEAD><TITLE>tableau 3</TITLE></HEAD>
<Body>
  <TABLE BORDER=12 CELLSPACING=10
  CELLPADDING=10>
  <TR>
  <TH ROWSPAN=2><FONT SIZE=7>LETTRES</FONT></TH>
  <TD><FONT SIZE=6>Aaa</FONT></TD><TD
  VALIGN=top>Bbb</TD>
  <TD VALIGN=bottom>Ccc</TD><TD
  VALIGN=top>Ddd</TD>
  <TD VALIGN=bottom>Eee</TD>
  </TR>
  <TR ALIGN=right>
  <TD ALIGN=left><FONT SIZE=6>F</FONT></TD>
  <TD>G</TD><TD>H</TD><TD>I</TD><TD>J</TD>
  </TR>
  </TABLE>
  <CENTER><H1>Tableau 3</H1></CENTER>
<BODY>
<HTML>
```



Exemple n°4 :

```
<HTML>
<HEAD><TITLE>tableau 4</TITLE></HEAD>
<Body>
  <TABLE BORDER=5>
  <TR>
  <TD ROWSPAN=2><IMG SRC="225sbe52.jpg"></TD>
  <TD><IMG WIDTH=70 HEIGHT=50 SRC="225sbe52.jpg"
  WIDTH=50% HEIGHT=50%></TD>
  <TD><IMG WIDTH=70 HEIGHT=50
  SRC="225sbe52.jpg"></TD>
  <TD><IMG WIDTH=70 HEIGHT=50
  SRC="225sbe52.jpg"></TD>
  </TR>
  <TR>
  <TD>rien</TD>
  <TD><IMG WIDTH=70 HEIGHT=50
  SRC="225sbe52.jpg"></TD>
  <TD><IMG WIDTH=70 HEIGHT=50
  SRC="225sbe52.jpg"></TD>
  </TR>
  </TABLE>
<BODY>
<HTML>
```



Formulaires

La commande FORM

Des formulaires peuvent être préparés afin de saisir des données et les traiter au niveau du serveur.

Pour rédiger un questionnaire, il faut:

- ☞ 1.Établir une zone d'édition (appelée FORM) en utilisant les commandes <FORM></FORM>.
- ☞ 2.Définir la méthode à employer pour transmettre au serveur l'information recueillie dans les champs du formulaire.
- ☞ 3.Identifier l'emplacement et le nom du programme qui devra traiter l'information recueillie.
- ☞ 4.Fournir, s'il y a lieu, les arguments au programme de traitement des données.

Toute cette information se retrouve dans la commande suivante:

```
<FORM METHOD="POST" ACTION="/cgi-bin/questionnaire.cmd?xxx">
```

La méthode utilisée est **POST**, le programme de traitement se nomme questionnaire.cmd et se retrouve dans le dossier cgi-bin, un seul argument est fourni au programme soit xxx.

Il est à noter que le programme de traitement des données (questionnaire.cmd en l'occurrence) doit être fourni par l'administrateur du serveur et créé en fonction de l'application à supporter.

Les commandes INPUT

Syntaxe : **INPUT="TEXT"**

Parmi les choix disponibles en HTML, un des types d'entrée de données est le champ **input type="text"**. Dans ce cas, il faut inscrire le type de champ, le nom du champ et ses dimensions à l'écran.

Ainsi, dans la question ci-dessous, le code à utiliser pour entrer le nom de la personne est :

```
NOM: <input type="text" name="name" size=30>.
```

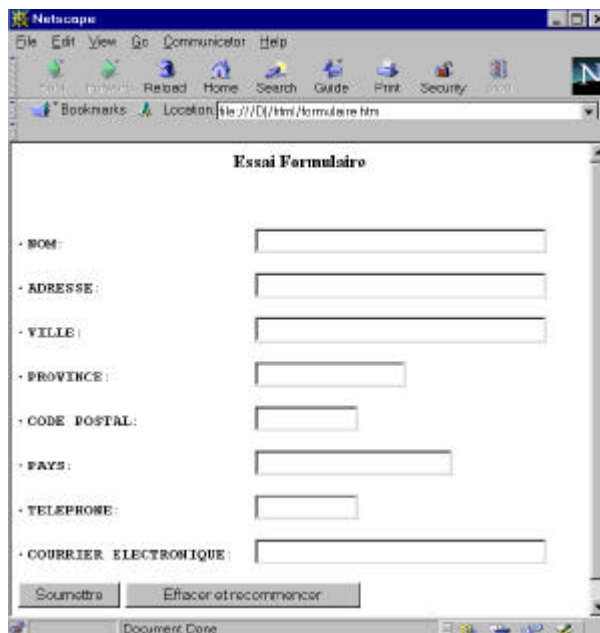
Le type **text** est un champ où l'utilisateur entre de l'information sur son clavier, dans une zone définie à l'écran par la commande **size**.

Ainsi, une commande **size=30** est un champ d'une longueur de 30 espaces de largeur. Pour un champ plus long, entrez une valeur de 50 ou 70, selon l'espace requis pour couvrir toutes les possibilités.

La commande **name="name"** précise que l'on désire enregistrer le contenu du champ dans la rubrique **"name"**.

Exemple :

```
<HTML>
<FORM METHOD="POST" ACTION="/cgi-
bin/questionnaire.cmd?xxx">
<p>
<h4><CENTER>Essai Formulaire
</CENTER></H4>
<pre><b>
<LI>NOM: <input type="text"
name="name" size=30>
<LI>ADRESSE: <input type="text"
name="street.address" size=30>
<LI>VILLE<input type="text"
name="city" size=30>
<LI>PROVINCE: <input type="text"
size=15 name="state">
<LI>CODE POSTAL: <input type="text"
name="zip" size=10>
<LI>PAYS:<input type="text"
name="country" size=20>
<LI>TELEPHONE: <input
type="text" name="phone" size=10>
<LI>COURRIER ELECTRONIQUE: <input
type="text" name="email" size=30>
</b></pre>
<input type="submit"
value="Soumettre"> <input
type="reset" value="Effacer et
recommencer"><P>
</form><p>
</BODY>
</HTML>
```



La commande INPUT="RADIO"

Un autre type de champ est le type "*input type=radio*" qui permet d'afficher une série de boutons radio comme choix de réponses.

Il suffit d'abord de poser la question puis de positionner la commande suivante:

```
<input type="radio" name="info" value="OUI">OUI  
<input type="radio" name="info" value="NON">NON
```

Dans l'exemple ci-dessous, deux choix sont offerts Oui ou Non. Lorsque l'utilisateur répond, il n'a d'autre choix que l'une ou l'autre réponse. Il ne peut pas choisir les deux.

```
<P>  
Comment avez-vous connu l'année spéciale Informatique à Amiens ?<br>  
<DL>  
<DT>A. <input type="radio" name="refer" value="par_hasard."> Tout &agrave; fait par hasard.  
<DT>B. <input type="radio" name="refer" value="Par un ancien."> Par un ancien étudiant d'AS.  
</DL>
```

```
Idem que l'exemple précédent  
...  
Comment avez-vous connu l'année  
spéciale Informatique à Amiens ?<br>  
<DL><DT>A. <input type="radio"  
name="refer" value="par_hasard.">  
Tout &agrave; fait par hasard.  
<DT>B. <input type="radio"  
name="refer" value="Par un ancien.">  
Par un ancien étudiant d'AS.  
</DL>  
<input type="submit"  
value="Soumettre"> <input  
type="reset" value="Effacer et  
recommencer"><P>  
</form><p>  
</BODY>  
</HTML>
```

Remarque :

Dans l'exemple précédent il faut noter que le nombre de choix de réponses n'est pas limité à deux.

La commande INPUT=CHECKBOX

La commande *input Type=checkbox* permet d'afficher une liste où plusieurs choix sont possibles en même temps.

La commande s'écrit:

```
Texte <input name="nom_du_champ" TYPE=checkbox VALUE="texte"><BR>  
ou  
<input name="nom_du_champ" TYPE=checkbox VALUE="texte"> Texte <BR>
```

selon que vous voulez afficher le bouton avant ou après la réponse.

```
Formation d'origine :  
DUT <input name="Formation" TYPE=checkbox VALUE="DUT"><BR>
```

```
BTS <input name="Formation" TYPE=checkbox VALUE="BTS"><BR>
DEUG <input name="Formation" TYPE=checkbox VALUE="DEUG"><BR>
```

Formation d'origine :

DUT

BTS

DEUG

Résultat :

La commande *SELECT NAME* et *OPTION*

La commande *select name* et *Option* permet d'afficher une liste ou un seul choix est possible et qui s'affichent sous la forme d'un menu "*pop-up*". La commande s'écrit:

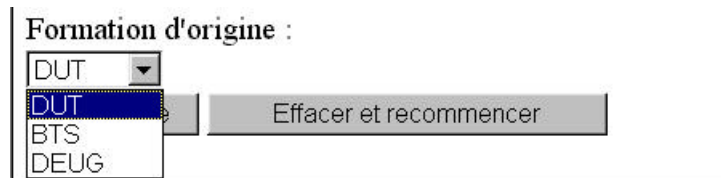
```
<select name><option selected>option1<option> option2<option>
option3</select>
```

Exemple :

```
Formation d'origine : <BR>
<select name><option selected>DUT<option>BTS<option>DEUG</select>
```

Résultat :

Formation d'origine :



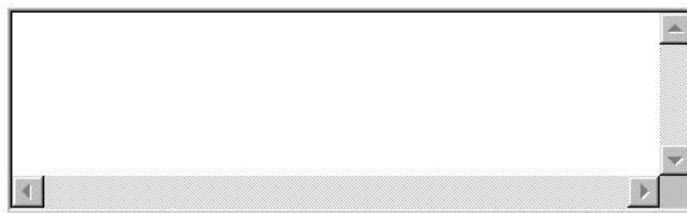
La commande *TEXTAREA*.

La commande HTML utilisée est:

```
<TEXTAREA name="nom_du_champ?" rows=n cols=m></TEXTAREA>
```

Dans ce type, on spécifie d'abord le type soit *textarea*, ensuite le nom de la rubrique soit *name=" nom_du_champ "* puis les paramètres d'affichage de la zone de dialogue en rangées (*n*) et en colonnes (*m*).

```
<TEXTAREA name="zone_libre" rows=5 cols=40></TEXTAREA>
```



Les formulaires doivent être complétés avant fermeture par une commande permettant d'envoyer le contenu des champs remplis au serveur HTTP.

Cette commande s'écrit:


```
<input type="submit" value="Soumettre">
```

On ajoute également une deuxième commande qui permet à l'utilisateur de reprendre le questionnaire s'il s'est trompé.

La commande en question s'écrit:

```
<input type="reset" value="effacer et recommencer">
```

Les FRAMES

Cette fonctionnalité est apparue avec NETSCAPE 2.0 ce qui a permis de créer des nannières, des notes de bas page, des fenêtres latérales etc. Ce partage de la fenêtre en "cadres" ou "cellules" autonomes, revient à diviser la fenêtre en plusieurs zones appelées frames. Chaque zone est gérée indépendamment des autres par un document HTML différent. Chaque zone porte un nom.

Attention le corps du fichier n'est plus inclus dans `<BODY>...</BODY>` mais dans : `<FRAMESET>...</FRAMESET>`. Ces deux marqueurs s'excluent mutuellement.

Trois commandes permettent de gérer cette fonctionnalité.

`<FRAMESET ROWS COLS> </FRAMESET>`

permet de diviser une zone en cellules horizontales ou verticales.

ROWS="n,n%,*,..."

divise la zone en cellules horizontales.

n = hauteur en nombre de pixels,

n% = hauteur de la cellule en pourcentage de l'écran,

***** = hauteur restante.

COLS="n,n%,*,..."

divise la zone en cellules verticales.

n = largeur en nombre de pixels

n% = largeur de la cellule en pourcentage de l'écran

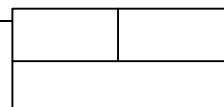
***** = largeur restante

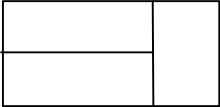
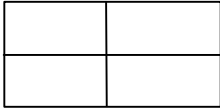
.

<pre><FRAMESET ROWS="80,50%,*" <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"></pre>	80 pixels de haut pour la frame 1 50% de la hauteur de l'écran pour la frame2 le reste de la hauteur pour la frame 3
<pre><FRAMESET ROWS="*,200,50,50" <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"></pre>	Le reste en hauteur pour la frame 1 200 pixels de haut pour la frame2 50 pixels de haut pour la frame3 50 pixels de haut pour la frame4
<pre><FRAMESET COLS="*,200,50,50" <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"></pre>	Le reste en largeur pour la frame 1 200 pixels de largeur pour la frame2 50 pixels de largeur pour la frame3 50 pixels de largeur pour la frame4

Des combinaisons entre COLS et ROWS peuvent être réalisées pour diviser l'écran en frame horizontale et verticale.

<pre><FRAMESET ROWS="*,*"> <FRAMESET COLS="*,*"></pre>	Division de l'écran en 2 parties égales dans la hauteur Division de la largeur de la première case en 2.
--	---



<pre> <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"> </FRAMESET> <FRAME SCR="doc.htm"> </FRAMESET> </pre>	
<pre> <FRAMESET COLS="80%,*"> <FRAMESET ROWS="*,*"> <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"> </FRAMESET> <FRAME SCR="doc.htm"> </FRAMESET> </pre>	<p>Division de l'écran avec 80% de l'espace de la largeur dans la première frame, Et division de la première frame en deux parties égales dans la hauteur.</p> 
<pre> <FRAMESET ROWS="*,*"> <FRAMESET COLS="*,*"> <FRAME SCR="doc.htm"> </FRAMESET> <FRAMESET COLS="*,*"> <FRAME SCR="doc.htm"> <FRAME SCR="doc.htm"> </FRAMESET> <FRAME SCR="doc.htm"> </FRAMESET> </pre>	<p>Division de l'écran en 4 parties égales</p> 

<FRAME NAME SRC SCROLLING MARGINWIDTH MARGINHEIGHT NORESIZE>

caractérise la cellule

NAME="nom_de_la_zone"

SRC="URL"

SCROLLING="yes" ou "no"

MARGINWIDTH= nombre de pixels de la marge verticale

MARGINHEIGHT= nombre de pixels de la marge horizontale

NORESIZ évite la modification de la taille par l'utilisateur

<NOFRAME> </NOFRAME>

à partir de cette commande, le texte ne sera affiché que par les Browsers ne sachant pas gérer les frames.

<pre> <HTML> <HEAD> <TITLE>html\frame1</TITLE> </HEAD> <FRAMESET COLS="80%,*"> <FRAMESET ROWS="*,*"> <FRAME SCROLLING="no" NAME="doc1" SRC="doc1.htm"> <FRAME SCROLLING="no" NAME="doc3" SRC="doc3.htm"> </FRAMESET> <FRAME SCROLLING="yes" NAME="doc2" SRC="doc2.htm"> </FRAMESET> </HTML> </pre>	<p>Frame1.htm: Document d'appel avec formatage des FRAMES</p>
<pre> <HTML> <BODY> <H1>Document1</H1> <CENTER>Amiens</CENTER> </BODY> </HTML> </pre>	<p>Doc1.html</p>
<pre> <HTML> <BODY> c 'est le Document2 </pre>	<p>Doc2.htm</p>

<pre></BODY> </HTML></pre>	
<pre><HTML> <BODY> c 'est le document3 </BODY> </HTML></pre>	Doc3.htm

Pour utiliser les "frames", un nouvel attribut "TARGET" de la commande <A> permet de donner le nom de la cellule (cf attribut "NAME" de la commande <FRAME>) dans laquelle le document sera affiché

lien

- ☞ TARGET="Cellule-1" affichage dans la cellule "Cellule-1"
- ☞ TARGET="_self" affichage dans la même cellule
- ☞ TARGET="_blank" affichage dans une nouvelle fenêtre
- ☞ TARGET="_top" suppression de toutes les frames, retour à un affichage classique



Liste des codes de caractères spéciaux en HTML Code HTML

Les caractères accentués sont remplacés par les séquences d'échappement suivantes :

A a
Á pour Á Â pour Â À pour À Å pour Å Ã pour Ã Ä pour Ä á pour á â pour â å pour å à pour à ã pour ã ä pour ä

E e
É pour É Ê pour Ê È pour È Ë pour Ë é pour é ê pour ê è pour è ë pour ë

I i
Í pour Í Î pour Î Ì pour Ì Ï pour Ï í pour í î pour î ì pour ì ï pour ï

N n
Ñ pour Ñ ñ pour ñ

O o
Ó pour Ó Ô pour Ô Ò pour Ò Õ pour Õ Ö pour Ö ó pour ó ô pour ô ò pour ò õ pour õ ö pour ö
U u

Ú pour Ú
Û pour Û
Ù pour Û
Ü pour Ü
ú pour ú
û pour û
ù pour ù
ü pour ü

Y y
Ý pour Ý
ý pour ý
ÿ pour ÿ

Exemple :

Les caractères accentués sont remplacés par des séquences d'échappement <P>

Les caractères spéciaux sont remplacés par des séquences d'échappement :

Les caractères < , > et & sont interprétés par le langage HTML, pour apparaître sans interprétation ils doivent être remplacés par :

<	<
>	>
&	&

Autres caractères spéciaux

"	"
Æ	Æ
æ	æ
Ç	Ç
ç	ç
Ð	Ð
ð	ð
Ø	Ø
ø	ø
Þ	Þ
þ	þ
ß	ß
®	®
©	©

¡ pour ¡ - ¢ pour ¢ - £ pour £ - ¤ pour ¤ - ¥ pour ¥
 ¦ pour ¦ - § pour § - ¨ pour ¨ - © pour © - ª pour ª
 « pour « - ¬ pour ¬ - ­ pour ® - ® pour ¯ - ¯ pour ¯
 ° pour ° - ± pour ± - ² pour ² - ³ pour ³ - ´ pour ´
 µ pour µ - ¶ pour ¶ - · pour · - ¸ pour ¸ - ¹ pour ¹
 º pour º - » pour » - ¼ pour ¼ - ½ pour ½ - ¾ pour ¾
 ¿ pour ¿ - × pour × - Þ pour Ð - ÷ pour ÷

Exemple :

La leçon du jour porte sur les tubes de Ø 200 mm <P>

En HTML la commande de fin de paragraphe est <P>
INDEX

CGI : Common Gateway Interface (que l'on peut traduire par "Script de Passerelle" en français)