

Le langage HTML

Le World Wide Web (W.W.W.) est constitué de pages Web, ces pages sont conçues avec le langage HTML ou *HyperText Markup Language* ou encore Langage dit à balises. Le HTML est donc un simple "langage à balises" comme les premiers traitements de texte. Ce langage permet de coder une page à l'aide de commandes de mise en forme. Ces dernières sont ensuite interprétées par un navigateur (ou browser en anglais) et apparaissent sur l'écran. Les pages HTML sont aujourd'hui le système de base d'Internet. Les sites peuvent inclure du texte ainsi que des images fixes ou animées, du son, de la vidéo et même des programmes interactifs (à l'aide de Java ou Javascript). Le langage HTML ne se rencontre pas exclusivement sur le web, celui-ci est également utilisé pour fournir une Interface aux CD-ROM multimédia et maintenant aux DVD ROM.

L'architecture d'un document.

Les balises

Toutes les instructions HTML sont exprimées dans des balises (ou "tags"), délimitées par des crochets (signe inférieur < et signe supérieur >). Cette structure est toujours la même, cependant chaque balise doit être ouverte et fermée (sauf à de rares exceptions), à noter que la balise de fermeture est toujours identique à la balise d'ouverture sauf qu'elle comporte une barre oblique après le signe <.

ex:

```
<TABLE> et </TABLE>
```

Les majuscules et minuscules dans les instructions n'ont aucune importance <TABLE> est équivalent à <table>. Attention quand même de bien respecter les ordres d'ouverture et de fermeture sinon on risque de perdre en lisibilité.

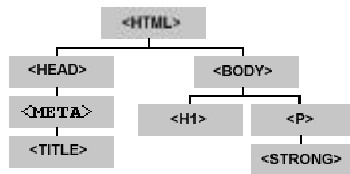


Schéma des relations conteneur - contenu entre les éléments d'un document HTML.

L'en-tête du document (<HEAD>)

Avant de commencer à écrire l'en-tête, le document principal (ou chaque page du projet, mais ce n'est pas obligatoire) doit comporter une instruction **DOCTYPE** qui permet de définir le standard HTML qu'on va utiliser.

Ex:

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
```

Ensuite on doit entrer la balise <HTML> qui indique le début du fichier, puis la balise <HEAD> dans laquelle sont résumées des informations générales, c'est à dire le titre de la page (c'est aussi le titre qui apparaîtra dans les favoris du visiteur s'il l'ajoute à sa liste de favoris) et les balises qui permettent d'indexer (référencer) le site (ces balises seront étudiées dans la rubrique suivante).

Ex:

```
<TITLE>Ma première page</TITLE>
```

La balise <META>.

La balise <META> permet normalement de fournir une description du site par le biais de mots clés et de phrases décrivant le site, ceci est donc l'utilisation la plus courante de cette balise mais, elle a d'autres fonctions comme notamment, la redirection automatique vers une URL.

Les principales propriétés de balise <META>

L'attribut HTTP-EQUIV.

Avec les attributs **HTTP-EQUIV** et **CONTENT** on peut recharger une page toutes les **n** secondes. Cette fonction est identique à l'action actualiser (reload) de votre navigateur.

ex :

```
<META HTTP-EQUIV="refresh" CONTENT="60">
```

Dans cet exemple la page qui comporte cette en-tête se rechargera toutes les 60 secondes.

Rediriger le visiteur.

Avec les fonctions ci-dessus et l'attribut URL on peut charger un autre document après **n** secondes.

ex :

```
<META HTTP-EQUIV="refresh" CONTENT="60; URL="http://www.yahoo.fr">
```

Définir le cadre de destination par défaut.

Avec l'attribut **Window-TARGET**, il est possible de définir le cadre (frame) de destination par défaut de la page, ceci évite de spécifier à chaque lien **TARGET="la_destination"**.

ex :

```
<META HTTP-EQUIV="Window-TARGET" CONTENT="votre_cadre-par_défaut">
```

A noter que si l'attribut **CONTENT** a comme valeur "**_top**", ceci aura pour effet d'afficher la page dans la totalité de la fenêtre du navigateur, utile si la page d'accueil est appelée d'un site où le multifenêtrage est présent.

Définir une date limite pour les pages.

Avec la valeur **expires**, il est possible de définir une date limite des pages et donc d'indiquer au *serveur proxy* qu'il doit impérativement recharger la page originale à la date indiquée.

ex :

```
<META HTTP-EQUIV="expires" CONTENT="Wed, 30 Sept 1999 12:15:00 GMT">
```

Interdire de mettre "en cache" les pages.

Avec les valeurs **pragma** et **no-cache** il est possible d'interdire de mettre "en cache" les pages.

ex :

```
<META HTTP-EQUIV="pragma" CONTENT="no-cache">
```

Indexer le site.

Avec les attributs **Keywords**, **Description**, **Author**, et **Lang** (nouveau dans HTML 4.0), on peut indexer un site, ou plus justement, on facilite la tâche du robot qui viendra visiter la page.

Voici un exemple d'indexation, commenté. A noter que ces balises <META> doivent être placées entre les balises <HEAD> et </HEAD>, juste après le titre de la page (balise <TITLE>).

Note : Si le site est en plusieurs langues on doit créer autant de balises <META> que de langues (juste celles qui comportent l'attribut LANG).

```
<META NAME="Author" LANG="fr" CONTENT="PRENOM NOM">
```

```
<META NAME="Author" LANG="bg" CONTENT="PRENOM NOM">
```

C'est l'auteur du site. L'attribut LANG spécifie la langue utilisée pour le site.

```
<META NAME="Description" CONTENT="La description du site">
```

Description du site en une phrase de préférence.

```
<META NAME="Keywords" LANG="fr" CONTENT="Mots clés">
```

Une liste de mots clés correspondant au mieux à ce site.

```
<META NAME="Identifler-URL" CONTENT="http://refg.vmei.acad.bg">
```

L'URL complète du site.

```
<META NAME="Publisher" CONTENT="PRENOM NOM">
```

Celui qui publie, c'est donc encore vous sauf si vous travaillez pour une société.

```
<META NAME="Copyright" CONTENT="COPYRIGHT NOM ...">
```

Inscrire les copyrights.

```
<META NAME="Generator" CONTENT="WebExpert, FTP Expert, PSP 4.1">
```

Les outils de développements utilisés .

```
<META HTTP-EQUIV="Content-Type" CONTENT="text/html; charset=ISO-8859-1">
```

Cette balise identifie le code ISO en vigueur.

Diriger les robots de référencement.

Avec la valeur robots on peut indiquer aux robots de référencement comment référencer nos pages. Cette valeur possède plusieurs attributs dont :

- **All** : Le robot indexe tout (par défaut).
- **None** : Stoppe le robot.
- **Index** : Votre page est indexée.
- **Noindex** : Votre page n'est pas indexée mais il suivra les liens de cette page.
- **Follow** : Le robot récupère les liens de votre page pour les référencer plus tard.
- **Nofollow** : Le robot ne récupère pas les liens de votre page.

ex :

```
<META NAME="robots" CONTENT="index">
```

Un exemple complet:

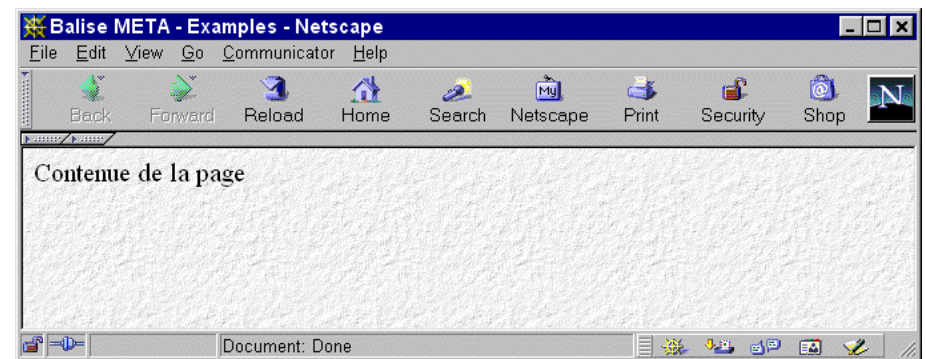
```
<!doctype html public "-//w3c//dtd html 4.0 transitional//en">
```

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1">
<meta name="GENERATOR" content="Mozilla/4.7 [en] (WinNT; I) [Netscape]">
<meta name="author" LANG="fr" content="Ivan Madjarov">
<meta name="Copyright" content="COPYRIGHT Ivan Madjarov">
<META NAME="robots" CONTENT="All">
<meta name="description" content="Cours programmation en Internet">
<meta name="keywords" content="Programmation, Internet, HTML, Tags, META">
<title>Balise META - Exemples</title>
</head>
```

```
<body text="#000000" bgcolor="#FFFFFF" link="#0A3990" vlink="#606060" alink="#FF0000"
background="bgdgray.gif" COLOR="WHITE">
```

Contenu de la page

```
</body>
</html>
```



Le corps du document (<BODY>)

Il est à noter que pour l'instant le navigateur n'affichera rien, car les véritables données qu'il va afficher vont se trouver dans la balise <BODY>. Cette balise détermine le corps du document, c'est ici qu'on va intégrer titres, textes, images, tableaux, liens, ...

```
<BODY ALINK=#rrbbgg BACKGROUND=URL BGCOLOR=#rrbbgg LINK=#rrbbgg TEXT=#rrbbgg
VLINK=#rrbbgg>
```

Le tag <BODY> a de nombreux attributs

- **BGCOLOR** : Définit la couleur du fond de l'écran.
- **TEXT** : Définit la couleur du texte.
- **LINK** : Définit la couleur des liens.
- **VLINK** : Définit la couleur des liens déjà visités.
- **ALINK** : Définit la couleur des liens activés, c'est à dire quand le pointeur de la souris passe au-dessus.
- **BACKGROUND** : Définit l'image (gif ou jpeg) à utiliser comme fond d'écran.
- **BGPROPERTIES** : Quand elle est définie à **FIXED**, l'image de fond ne défile pas (fonctionne qu'avec Internet.Explorer).
- **LEFTMARGIN** : Définit la largeur de la marge de gauche en pixels (fonctionne qu'avec Internet.Explorer).
- **TOPMARGIN** : Définit la largeur de la marge du haut en pixels (fonctionne qu'avec Internet.Explorer).

Ex:
<HTML>
<HEAD>

```
<TITLE>Ma première page</TITLE>
</HEAD>
<BODY BGCOLOR="WHITE" TEXT="BLACK" LINK="BLUE" VLINK="GREEN"ALINK="RED">
Voici une première page
</BODY>
</HTML>
```

Commenter un document

On peut commenter le code source sans que le navigateur affiche ce commentaire, cela peut s'avérer pratique si vous êtes plusieurs à travailler sur le même document. Cette instruction est aussi utilisée lors de l'écriture d'un script Javascript pour cacher le code aux navigateurs non compatibles.

Ex:

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

Mise en forme des textes et paragraphes

Les paragraphes <P>

Un document lu c'est avant tout un document aéré, c'est pour cela qu'il faut diviser le contenu du texte en plusieurs paragraphes. Il existe deux balises qui permettent d'obtenir, soit un saut de ligne soit, le commencement d'un nouveau paragraphe.

La balise <P> exprime le début d'un paragraphe, c'est à dire un double espace, tandis que la balise
 effectue un retour à la ligne (simple espace) identique à la touche Entrée de votre clavier (sous traitement de texte).

```
<P ALIGN=left|center|right> ... </P>
```

La balise
 possède l'attribut **CLEAR** (trois valeurs : **LEFT**, **RIGHT**, **ALL**) permettant d'insérer un retour à la ligne dès que la marge (gauche, droite, ou les deux) est libre.

```
<BR CLEAR=left|all|right|none>
```

ex :

```
<BR CLEAR=RIGHT>
```

(un retour s'effectue à la ligne dès que la marge de droite est libre).

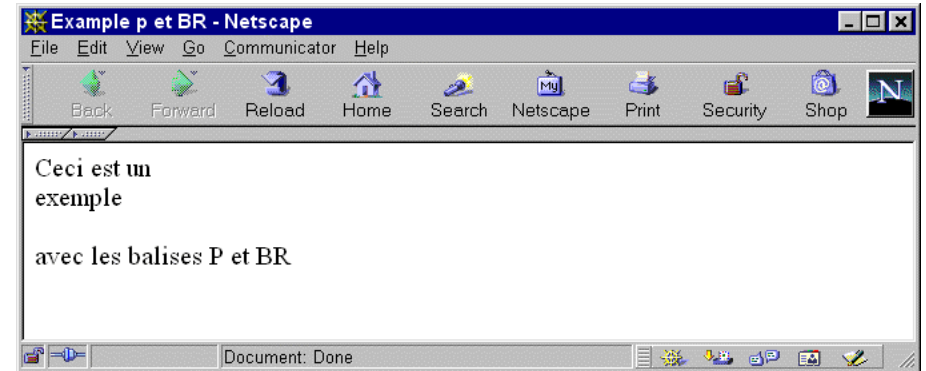
Les balises </P> et </BR> de fermetures sont facultatives.

La balise <P> possède l'attribut **ALIGN** qui sert à aligner le texte, il prend 3 valeurs :

1. **LEFT** : Aligne le texte sur la marge de gauche.
2. **RIGHT** : Aligne le texte sur la marge de droite.
3. **CENTER** : Centre le texte entre les marges.

Exemple et code source.

```
<html>
<head><title>Exemple p et BR</title>
</head>
<body>
<P>Ceci est un <BR> exemple <P> avec les balises P et BR</P>
</body>
</html>
```



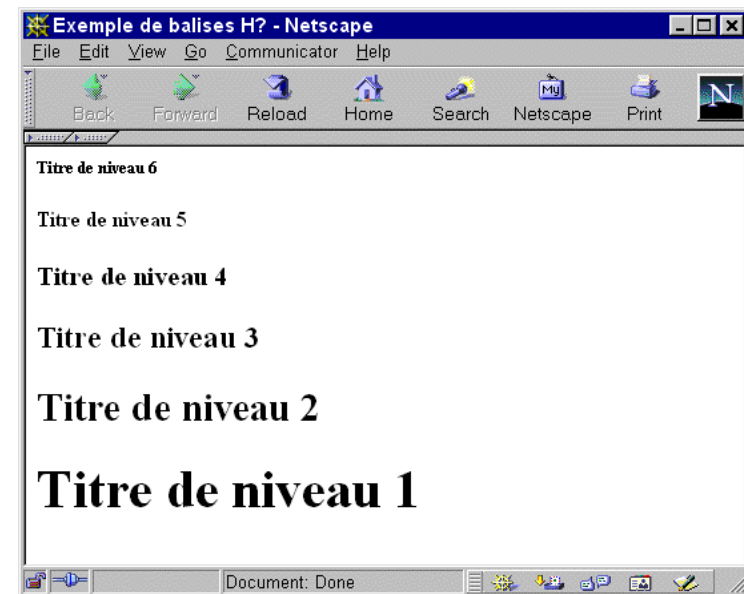
Tout le texte compris entre les balises <NOBR> et </NOBR> ne contiendra pas de passages à la ligne intempestifs, à utiliser avec parcimonie car le visiteur sera peut être obligé de faire défiler horizontalement la fenêtre de son navigateur, ce qui n'est pas souvent apprécié ...

Les titres <H?>.

L'élément TITRE permet de mettre en valeur une portion de texte. Pour cela, le texte doit être encadré avec les balises d'ouverture et de fermeture choisies, un retour à la ligne se fera automatiquement. Il y a six niveaux de TITRE de <H1> (le plus important) à <H6> (le plus petit). Il possède lui aussi l'attribut **ALIGN** (**RIGHT**, **LEFT**, **CENTER**).

```
<H1|2|3|4|5|6 ALIGN=LEFT|CENTER|RIGHT>Texte de titre</H1|2|3|4|5|6>
```

Exemple et code source avec chaque balise TITRE.



Lignes horizontales (<HR>).

Utiliser des lignes horizontales ou filets horizontaux sont très pratiques pour séparer les différentes parties de votre document. Cette méthode s'obtient avec la balise <HR>, elle ne nécessite pas de balise de fermeture. Elle possède plusieurs attributs :

```
<HR ALIGN=LEFT|RIGHT|CENTER NOSHADE SIZE=n WIDTH=n|n% COLOR="#rrggbb">
```

1. **ALIGN** : Permet d'aligner la ligne horizontale. Valeurs : **RIGHT**, **LEFT**, ou **CENTER**.
2. **WIDTH** : Spécifie la largeur de la ligne (en pixels ou % de la fenêtre).
3. **SIZE** : Epaisseur de la ligne (de 1 à 10 en pixels).
4. **NOSHADE** : S'il est présent, l'effet d'ombre (3D) est annulé.
5. **COLOR** : Spécifie la couleur de la ligne (uniquement Internet.Explorer).

Exemple de plusieurs filets ainsi que leur code source.

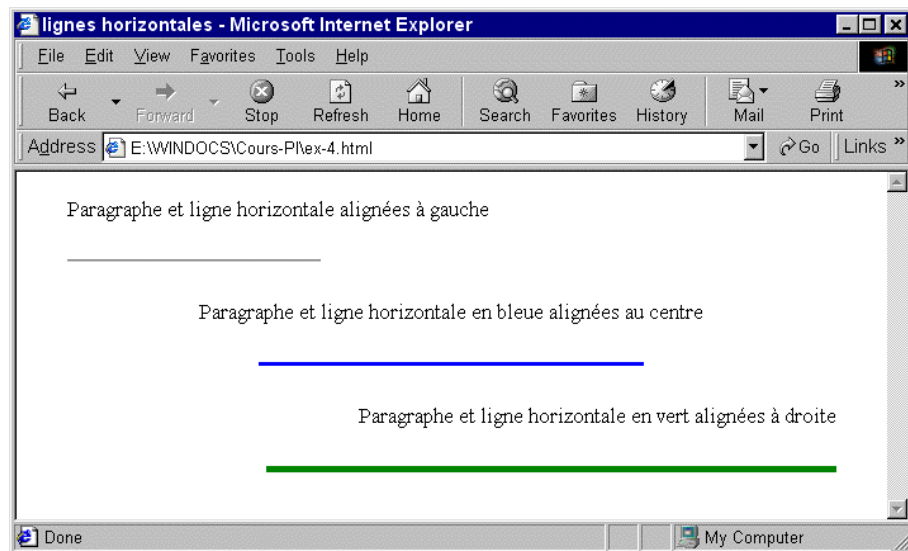
```
<HR WIDTH=100 SIZE=3 ALIGN=LEFT>
```

```
<HR WIDTH=150 SIZE=5 COLOR="blue" ALIGN=CENTER>
```

```
<HR WIDTH=200 SIZE=1 COLOR="#008000" NOSHADE ALIGN=RIGHT>
```

Un exemple plus complet, paragraphes et lignes horizontales et leurs attributs:

```
<html>
<head><title>lignes horizontales</title>
</head>
<body bgcolor="white" text="black" link="blue" vlink="green" alink="red"
leftmargin=40 topmargin=20>
<p align=left>Paragraphe et ligne horizontale alignée;es à gauche</p>
<hr width=200 size=2 align=left>
<p align=center>Paragraphe et ligne horizontale en bleue alignée;es au
centre</p>
<hr width=50% size=3 color="blue" align=center>
<p align=right>Paragraphe et ligne horizontale en vert alignée;es à droite</p>
<hr width=450 size=5 color="#008000" noshade align=right>
</body> </html>
```



La balise <DIV>.

La balise <DIV> permet de définir l'alignement de plusieurs éléments (titres, textes, images, filets, ...). Elle possède l'attribut **ALIGN** qui prend 3 valeurs (**RIGHT**, **LEFT**, **CENTER**), elle peut donc remplacer la balise <CENTER>, qui elle permet de centrer plusieurs éléments sur une page.

```
<DIV ALIGN=LEFT|RIGHT|CENTER ID|CLASS=name STYLE=style-data CLEAR=LEFT|RIGHT|ALL
NOWRAP> ... </DIV>
```

ex :

```
<DIV ALIGN=LEFT>
```

A noter que depuis la version 4.0 du HTML, la balise <DIV> se diversifie, en effet elle accepte l'attribut **STYLE** (feuilles de styles, sujet qui sera abordé plus tard).

ex:

```
<DIV STYLE="color:green">
```

La balise <BLOCKQUOTE>.

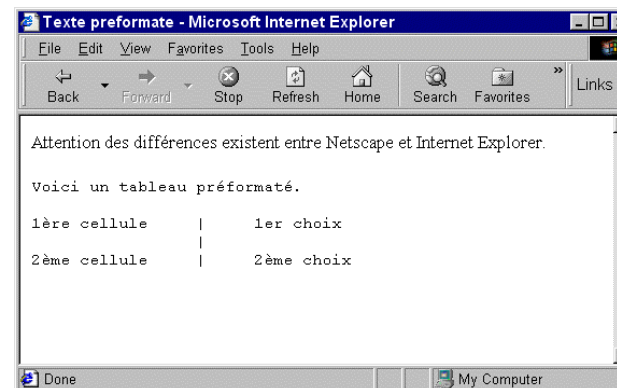
La balise permet de créer un retrait dans la marge de gauche. Comme les balises <CENTER> et <DIV>, tout élément compris entre les balises d'ouverture et de fermeture bénéficiera de ce retrait.

Texte préformaté <PRE>

Si pour une raison quelconque, vous ne voulez pas vous servir des balises d'alignements et de paragraphes, la balise <PRE> est recommandée, en effet tout texte compris entre les balises <PRE> et </PRE> sera transcrit de la même façon. Toutefois l'utilisation de TITRE, et de mise en forme de paragraphe n'est pas autorisée. Attention des différences existent entre Netscape et I.Explorer.

Ex :

```
<html><head><title>Texte preformate</title> </head>
<body>
<p>Attention des différences existent entre Netscape et Internet Explorer.</p>
<pre>
Voici un tableau préformaté;
1ère cellule | 1er choix
2ème cellule | 2ème choix
</pre>
</body> </html>
```



Les principaux caractères accentués.

Voici une petite liste des principaux caractères accentués. Il suffit d'insérer le code du caractère pour l'obtenir dans votre page avec cette manipulation on peut garantir que l'accent sera reconnu correctement par tous les navigateurs.

Caractère	Abréviation
à	à
â	â
ä	ä
æ	æ
é	é
è	è
ê	ê
ë	&euuml;
ù	ù
ú	ú
û	û
ü	ü
ö	ö
ï	ï
î	î
ç	ç

Les principaux caractères spéciaux.

Caractère	Abréviation
<	<
>	>
Espace insécable	
"	"quot;
Á	Á
Â	&Aacirc;
À	À
Å	Å
Ã	Ã
Ä	Ä
Ç	Ç
É	É
Ê	Ê
È	È
Ë	&Euuml;
Î	Î
Ó	Ô
Ö	Ö
Æ	Æ
®	®

©	©
™	™
«	«
»	»
±	±
°	°
¹	¹
²	²
³	³
§	§
¼	¼
½	½
¾	¾
¶	¶
·	·
œ	œ
‰	‰
Æ	Æ

La mise en forme logique.

Ces instructions dites logiques, laissent le navigateur interpréter le formatage du texte. Ces balises sont souvent utilisées dans un environnement non graphique (sous DOS par exemple). A noter qu'elles sont de moins en moins utilisées.

Pour utiliser ces balises, il suffit d'entourer le texte avec la balise d'ouverture choisie et la balise de fermeture correspondante.

Voici toutes les balises de mise en forme logique et le code source.

`<CITE>CITE : Pour effectuer une citation.</CITE>`

`<CODE>CODE : Pour indiquer du code informatique.</CODE>`

`EM : (emphase) Similaire à l'italique.`

`<KBD>KBD : (Keyboard, clavier) Utilisé généralement pour indiquer une réponse d'utilisateur.</KBD>`

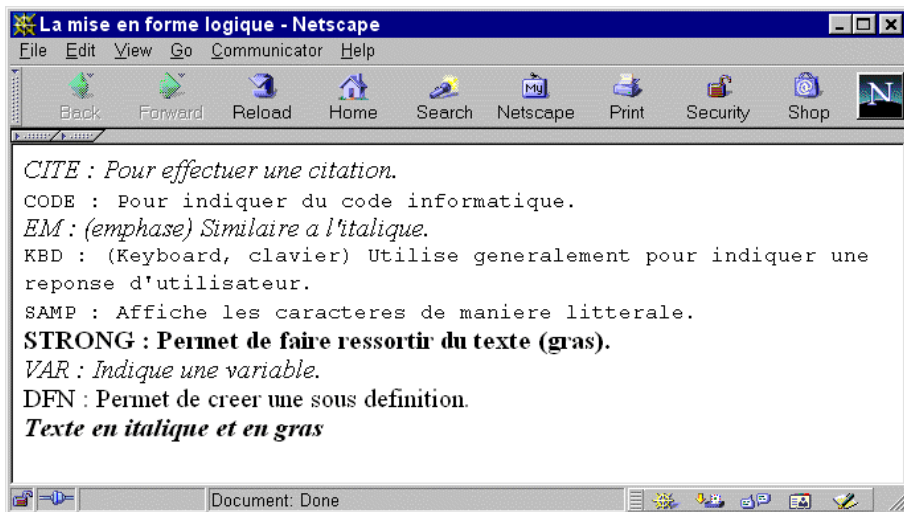
`<SAMP>SAMP : Affiche les caractères de manière littérale.</SAMP>`

`STRONG : Permet de faire ressortir du texte (gras).`

`<VAR>VAR : Indique une variable.</VAR>`

`<DFN>DFN : Permet de créer une sous définition.</DFN>`

Vous pouvez bien sûr imbriquer plusieurs balises attention tout de même à respecter l'ordre de fermeture.



```
<EM><STRONG>Texte en italique et en gras</STRONG></EM>
```

La mise en forme physique.

Ces instructions dites physiques sont plus proches de l'auteur, en effet, avec ces balises ci-dessous c'est vous qui décidez ce que le navigateur affichera, même s'il y a certaines limites.

Pour utiliser ces balises, il suffit d'entourer votre texte avec la balise d'ouverture choisie et la balise de fermeture correspondante.

Voici toutes les balises de mise en forme physique et le code source.

```
<B>B : Permet de faire ressortir du texte gras.</B>
```

```
<I>I : Met le texte en Italique (ou <ITALIC>).</I>
```

```
<TT>TT : Affiche le texte en police style machine à écrire.</TT>
```

```
<U>U : Souligne le texte.</U>
```

```
<STRIKE>STRIKE : Barre le texte.</STRIKE>
```

```
<BIG>BIG : Permet d'utiliser une police plus grande.</BIG>
```

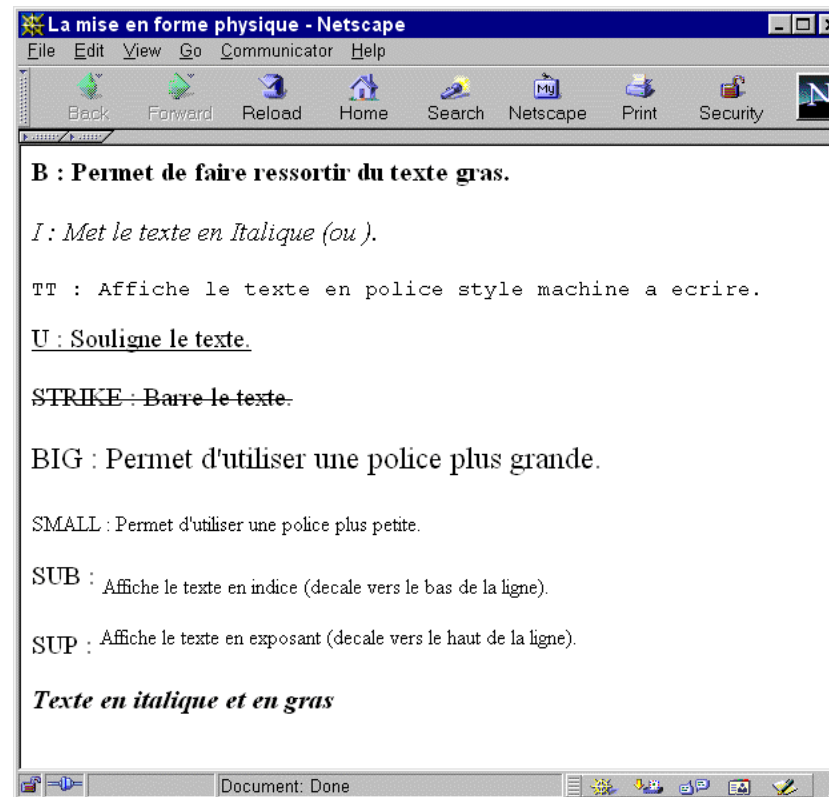
```
<SMALL>SMALL : Permet d'utiliser une police plus petite.</SMALL>
```

```
SUB : <SUB>Affiche le texte en indice (décalé vers le bas de la ligne).</SUB>
```

```
SUP : <SUP>Affiche le texte en exposant (décalé vers le haut de la ligne).</SUP>
```

Vous pouvez bien sûr imbriquer plusieurs balises attention tout de même à respecter l'ordre de fermeture.

```
<I><B>Texte en italique et en gras</B></I>
```



Texte en italique et en gras

Les polices de caractères .

Il n'y a rien de plus facile à spécifier une police, la méthode à utiliser par le HTML s'appuie sur la balise , cependant pour que votre document se visualise correctement (c'est à dire avec la police définie dans votre document), essayer d'employer une police standard (arial, courier ...) à toutes les plates-formes, sinon la police paramétrée par défaut sur le navigateur de chaque visiteur sera affichée.

La balise possède plusieurs attributs :

- **FACE** : Permet de définir par ordre de priorité, les noms des polices.
- **SIZE** : Permet de définir la taille de la police de 1 à 7 (valeur par défaut : 3).
- **COLOR** : Permet de définir la couleur du texte (soit avec le nom de la couleur ou a valeur hexadécimale correspondante).

A noter qu'il peut être utilisé de deux manières : Absolue ex : **SIZE=2** Relative ex : **SIZE= +2** .

La balise <BASEFONT>

La balise <BASEFONT> est utilisée pour définir la police standard, elle ne possède pas de balise de fermeture, et elle emploie les mêmes attributs que la balise . En réalité il n'y a aucune différence importante avec la balise .

Quelques exemples d'utilisation de la balise et leur code source.

```
<FONT FACE="Comic Sans Ms, arial" > Police Comic Sans Ms - Taille par défaut (3)
</FONT>
```

```
<FONT FACE="Impact, verdana" SIZE=2 COLOR="green"> Police Impact - Taille 2 - Couleur
: Verte </FONT>
```

```
<FONT FACE="Times New Roman" SIZE=4 COLOR="red"> Police Times New Roman - Taille 4 -
Couleur : Rouge </FONT>
```



L'accentuation.

Pour inclure des caractères accentués, vous devez remplacer la lettre accentuée par l'entité correspondante (code ISO-8859-1 ou ISO Latin-1). Ceci avant tout pour permettre une harmonisation entre les différents systèmes d'exploitations et surtout entre les différentes particularités de chaque langue. Le principe est le même pour insérer des caractères spéciaux. Voir les tableaux plus-haut.

Les liens hypertextes.

Liens internes (même page).

Les liens internes permettent de se déplacer à l'intérieur d'un fichier HTML sans que le visiteur soit obligé de faire "scroller" la fenêtre. Le visiteur va donc cliquer sur un lien et il sera amené vers l'ancre. Pour cela deux étapes sont nécessaires:

Définir la cible d'un lien (l'ancre).

ex:

```
<A NAME="top"></A>
```

Il n'est pas nécessaire dans ce cas ci de mettre un texte ou une image, entre les balises d'ouverture et de fermeture. (Le nom ne doit pas contenir d'espaces, ni d'accents, ni de caractères spéciaux).

Associer un lien à une ancre.

ex:

```
<A HREF="#top">top</A>
```

Liens internes (page différente).

Il est très facile de créer des liens entre des fichiers HTML.

Lien vers une page située dans le même répertoire.

ex :

```
<A HREF="nom.htm">Cliquez ici</A>
```

Lien dans un répertoire différent.

ex :

```
<A HREF="sujet/nom.htm">Cliquez ici</A>
```

Lien dans un même répertoire et vers une ancre d'une autre page.

ex :

```
<A HREF="nom.htm#top">Cliquez ici</A>
```

Lien dans un répertoire différent et vers une ancre d'une autre page.

ex :

```
<A HREF="sujet/nom.htm#top">Cliquez ici</A>
```

Liens externes.

Les liens externes commencent comme les autres liens, c'est à dire avec la balise <A HREF="...", par contre suivant le protocole, le reste de la syntaxe varie.

Lien externe vers un site http.

ex :

```
<A HREF="http://www.yahoo.fr">Découvrez Yahoo France</A>
```

Lien externe vers un serveur FTP.

ex :

```
<A HREF="ftp://ftp.multimania.com">FTP Multimania</A>
```

Lien externe vers un serveur de news. Attention pour que ce lien soit valide il faut que le visiteur soit abonné à ce serveur de news.

ex :

```
<A HREF="news:news.mygale.org">Les news Mygale - Multimania!!</A>
```

Avec un lien externe on peut également proposer un fichier (.EXE ou .ZIP) en téléchargement.

ex :

```
<A HREF="util.exe">Cliquez ici pour télécharger le fichier</A>
```

Lien e-mail.

Vous pouvez créer un lien e-mail pour que la messagerie paramétrée par défaut du visiteur se lance automatiquement. Attention tout de même ce lien ne marche pas avec les premières versions de Netscape et d'Explorer. Ce lien possède

plusieurs attributs dont : ?subject= pour définir un sujet (ou objet) automatiquement, ce qui est très pratique pour trier votre courrier.

Lien e-mail sans sujet.

```
<A HREF="mailto:docs@multimania.com">docs@multimania.com</A>
```

Lien e-mail avec sujet (ici renseignement).

```
<A HREF="mailto:help@multimania.com?subject=renseignement">help@multimania.com</A>
```

Lien e-mail avec sujet, et texte dans le corps du message.

```
<A HREF="mailto:help@multimania.com?subject=renseignement&BODY=Votre message">help@multimania.com</A>
```

Lien e-mail avec sujet, envoyé à 2 personnes (dont une en CC).

```
<A HREF="mailto:help@multimania.com?cc=jean@inext.fr&subject=renseignement">help@multimania.com</A>
```

Lien e-mail avec 3 personnes (dont deux en CCI).

```
<A HREF="mailto:help@multimania.com?cc=help2@multimania.com&subject=renseignement">help@multimania.com</A>
```

Récapitulatif et exemples.

Voici un tableau sur tous les liens existants à l'heure actuelle ainsi que des exemples.

Liens Exemples

Lien interne départ (même page) ``

Lien interne arrivée (même page) ``

Lien interne (page différente) ``

Lien externe ``

Lien e-mail ``

Lien vers news ``

Lien vers serveur ftp ``

Lien vers serveur gophe ``

Lien vers Telnet ``

Lien vers serveur Wais ``

Les Tableaux

Présentation des tableaux.

La balise `<TABLE>` permet d'insérer un tableau, c'est l'une des balises les plus utilisées, en effet les tableaux permettent une meilleure présentation des informations, nombreux sont les sites qui utilisent les tableaux pour obtenir une présentation de type "journal".

Création des tableaux.

La création des tableaux se passe autour de 4 balises :

1. La balise `<TABLE>` qui indique au navigateur la création d'un **tableau** :

```
<TABLE [ALIGN=left|center|right] [BORDER=n] [CELLPADDING=n] [CELLSPACING=n] [WIDTH=n]>
```

...

```
</TABLE>
```

La balise `<TH>` définit l'**en-tête** de chaque colonne :

```
<TH [ALIGN=left|center|right] [COLSPAN=NUMBER [NOWRAP] [ROWSPAN=NUMBER] [VALIGN=top|middle|bottom|baseline] [WIDTH=value|percent]>
```

La balise `<TR>` définit une **ligne**.

```
<TR [ALIGN=left|center|right] [VALIGN=top|middle|bottom|baseline]>
```

La balise `<TD>` définit une **cellule**.

```
<TD [ALIGN=left|center|right] [COLSPAN=NUMBER] [NOWRAP] [ROWSPAN=NUMBER] [VALIGN=top|middle|bottom|baseline] [WIDTH=value|percent]>
```

La balise `<CAPTION>` qui affiche le **titre** du tableau possède 2 attributs :

```
<CAPTION [ALIGN=top|bottom|left|right|center] [VALIGN=top|bottom] Text </CAPTION>
```

- **TOP** : Titre placé au dessus du tableau (par défaut).
- **BOTTOM** : Titre placé en dessous du tableau.

Toutes ces balises doivent bien entendu être fermées pour indiquer la fin du tableau, d'une colonne, et d'une cellule. Pour définir une police de caractère pour le tableau, on est obligé d'en définir pour chaque cellule même si elle est identique.

Ex:

```
<TD<FONT FACE="Verdana, arial" SIZE=1>Cellule 1</FONT></TD>
```

Exemple d'un tableau avec son code source.

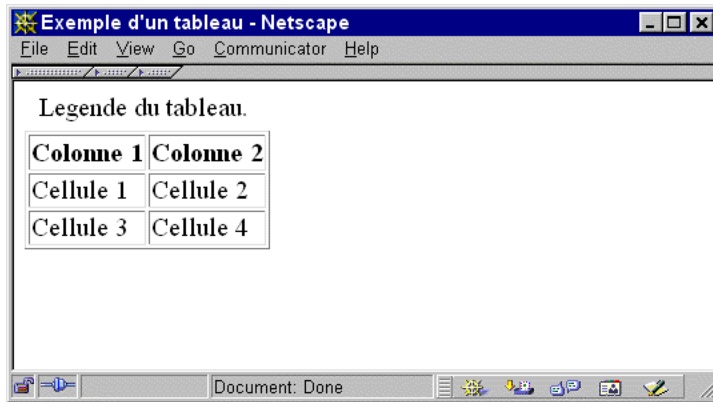
```
<HTML>
<HEAD><TITLE> Exemple d'un tableau</TITLE></HEAD>
<BODY>
<TABLE BORDER=1>
<CAPTION> Légende du tableau. </CAPTION>
<TR>
<TH>Colonne 1</TH>
<TH>Colonne 2</TH>
```



```

</TR>
<TR>
<TD>Cellule 1</TD>
<TD>Cellule 2</TD>
</TR>
<TR>
<TD>Cellule 3</TD>
<TD>Cellule 4</TD>
</TR>
</TABLE>
</BODY>
</HTML>

```



Aligner les éléments du tableau.

1. L'alignement des données s'obtient, en jonglant avec les attributs **ALIGN** pour l'alignement horizontal et **VALIGN** pour l'alignement vertical. Ces attributs ont plusieurs valeurs :
2. **ALIGN=LEFT** : (défaut) Les données sont alignées à gauche de la cellule (pour **TD**) ou de toutes les cellules (pour **TH** ou **TR**).
3. **ALIGN=RIGHT** : Les données sont alignées à droite de la cellule (pour **TD**) ou de toutes les cellules (pour **TH** ou **TR**).
4. **ALIGN=CENTER** : Les données sont centrées à l'intérieur de la cellule (pour **TD**) ou de toutes les cellules (pour **TH** ou **TR**).
5. **VALIGN=MIDDLE** : (défaut) Les données sont centrées verticalement à l'intérieur de la cellule (pour **TD**) ou de toutes les cellules (pour **TH** ou **TR**).
6. **VALIGN=TOP** : Les données sont en haut de la cellule (pour **TD**) ou de toutes les cellules (pour **TH** ou **TR**).
7. **VALIGN=BOTTOM** : Les données sont en bas de la cellule (pour **TD**) ou de toutes les cellules (pour **TH** ou **TR**).
8. **VALIGN=BASELINE** : Toutes les données des cellules sont alignées sur une ligne commune (uniquement pour **TH** ou **TR**).

Exemple d'un tableau où les cellules sont alignées différemment.

```

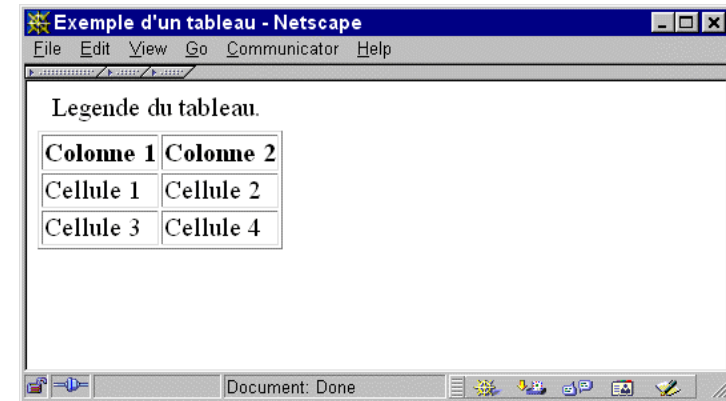
<HTML>
<HEAD><TITLE> Exemple d'un tableau</TITLE></HEAD>
<BODY>
<TABLE BORDER=1>
<CAPTION> Légende du tableau.</CAPTION>
<TR>
<TH ALIGN=RIGHT>Colonne 1 alignée à droite</TH>
<TH ALIGN=CENTER>Colonne 2 centrée</TH>
</TR>
<TR>
<TD ALIGN=LEFT VALIGN=TOP>Cellule 1 Alignée à gauche

```

```

<BR>et en haut</TD>
<TD ALIGN=CENTER VALIGN=BOTTOM>Cellule 2
<BR>centrée et en bas</TD>
</TR>
<TR VALIGN=BASELINE>
<TD>Cellule 3 basée sur la ligne</TD>
<TD>Cellule 4 basée
<BR>sur la ligne</TD>
</TR>
</TABLE>
</BODY>
</HTML>

```



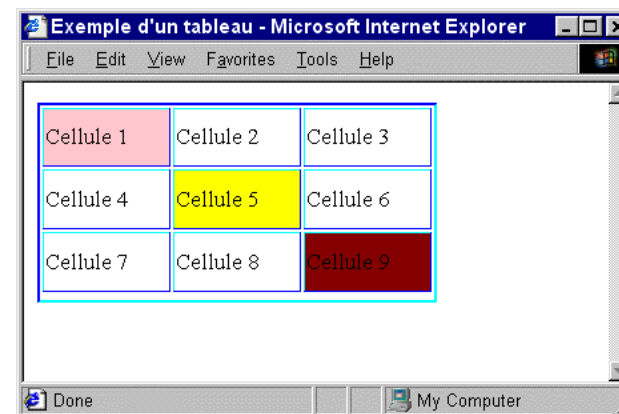
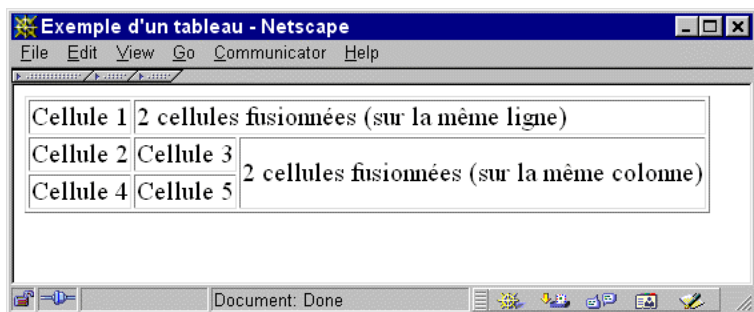
Fusionner les lignes et colonnes.

Les lignes et colonnes peuvent étre fusionnées afin d'obtenir des cellules plus ou moins grandes pour accueillir des données. Ce fusionnement est réalisé avec l'attribut **ROWSPAN** et **COLSPAN**. Pour fusionner 2 cellules adjacentes sur une ligne **COLSPAN** est requis (pour **TH** ou **TD**). Pour fusionner 2 cellules adjacentes sur une colonne **ROWSPAN** est requis (pour **TH** ou **TD**). Exemple d'un tableau avec des fusions.

```

<HTML>
<HEAD><TITLE>Exemple d'un tableau</TITLE></HEAD>
<BODY>
<TABLE BORDER=1>
<TR>
<TD>Cellule 1</TD>
<TD COLSPAN=2>2 cellules fusionnées (sur la même ligne)</TD>
</TR>
<TR>
<TD>Cellule 2</TD>
<TD>Cellule 3</TD>
<TD ROWSPAN=2>2 cellules fusionnées (sur la même colonne)</TD>
</TR>
<TR>
<TD>Cellule 4</TD>
<TD>Cellule 5</TD>
</TR>
</TABLE>
</BODY>
</HTML>

```



Arrière plan, mise en page, espacement.

En principe la hauteur et la largeur d'une cellule s'adaptent automatiquement aux données (texte ou image). Néanmoins vous pouvez définir une hauteur et une largeur pour un tableau ou une cellule, l'espacement des cellules ainsi que la taille de la bordure peuvent être aussi paramétrés.

Attributs de mise en page et d'espacement (valable pour `<TABLE>`, `<TD>` ou `<TH>`).

- **WIDTH** : Spécifie la largeur du tableau ou d'une cellule, en pixels ou pourcentage de la fenêtre du navigateur.
- **HEIGHT** : Spécifie la hauteur du tableau ou d'une cellule, en pixels ou pourcentage de la fenêtre du navigateur.
- **BORDER** : Spécifie la taille en pixels (par défaut 1) de la bordure du tableau.
- **CELLPADDING** : Spécifie l'espace en pixels entre la bordure et le contenu de la cellule du tableau.
- **CELLSPACING** : Spécifie l'espace en pixels entre les cellules du tableau.
- **BGCOLOR** : Spécifie la couleur d'une cellule.
- **BORDERCOLOR** : Spécifie la couleur de la bordure du tableau.
- **BORDERCOLORLIGHT** : Spécifie la couleur aux points culminants de la bordure du tableau (effet 3d).
- **BORDERCOLORDARK** : Spécifie la couleur de l'ombre de la bordure du tableau (effet 3d).
- **BACKGROUND** : Spécifie l'URL de l'image à afficher comme image d'arrière plan du tableau (Internet Explorer seulement).

A noter que pour la balise `<TD>` l'attribut **NOWRAP** permet d'éviter les retours à la ligne (inutile si vous avez défini votre tableau en pixels)

Exemple d'un tableau avec certains de ces attributs.

```
<HTML>
<HEAD><TITLE>Exemple d'un tableau</TITLE></HEAD>
<TABLE BORDER=2 BORDERCOLOR="BLUE" BORDERCOLORDARK="AQUA"
WIDTH=300 HEIGHT=150>
<TR>
<TD BGCOLOR="PINK">Cellule 1</TD>
<TD>Cellule 2</TD>
<TD>Cellule 3</TD>
</TR>
<TR>
<TD>Cellule 4</TD>
<TD BGCOLOR="YELLOW">Cellule 5</TD>
<TD>Cellule 6</TD>
</TR>
<TR>
<TD>Cellule 7</TD>
<TD>Cellule 8</TD>
<TD BGCOLOR="MAROON">Cellule 9</TD>
</TR>
</TABLE>
</BODY>
</HTML>
```

Les attributs RULES et FRAME (HTML 4.0).

Ces attributs sont valables uniquement pour la balise `<TABLE>` ils sont de plus reconnus par les navigateurs de dernière génération qui savent exploiter le HTML 4.0.

L'attribut FRAME.

il fonctionne juste si **BORDER** est présent dans la balise `<TABLE>` il signale les côtés d'un cadre à afficher. Voici la liste des valeurs qu'il peut supporter :

- **ABOVE** : Affiche les bordures externes du haut du tableau seulement.
- **BELOW** : Affiche les bordures externes du bas du tableau seulement.
- **BORDER** : Affiche les bordures sur tous les côtés du tableau.
- **BOX** : Affiche une boîte autour du tableau.
- **INSIDES** : Affiche les bordures sur le dessus et le dessous du tableau seulement.
- **HSIDES** : Affiche les bordures externes du côté horizontal du tableau (haut et bas).
- **LHS** : Affiche les bordures externes du côté gauche du tableau seulement.
- **RHS** : Affiche les bordures externes du côté droit du tableau seulement.
- **VOID** : Permet de retirer toutes les bordures externes du tableau.
- **VSIDES** : Affiche les bordures externes des côtés droit et gauche du tableau.

L'attribut RULES.

il fonctionne juste si l'attribut **BORDER** est présent dans la balise `<TABLE>` Il permet de spécifier des règles supplémentaires à l'intérieur du tableau. Les valeurs acceptées sont les suivantes:

- **NONE** : Pour retirer toutes les bordures internes du tableau.
- **GROUPS** : Affiche une bordure horizontale entre les sections.
- **ROWS** : Affiche une bordure horizontale entre toutes les lignes du tableau.
- **COLS** : Affiche une bordure horizontale entre toutes les colonnes du tableau.
- **ALL** : Affiche des bordures dans le tableau.

A noter que ces deux attributs doivent aussi être utilisés avec les balises :

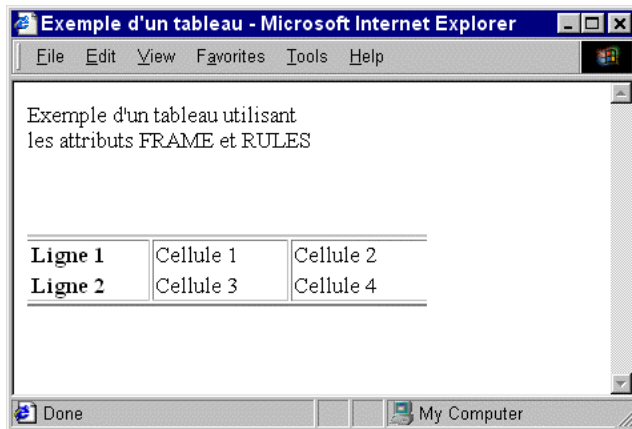
1. **THEAD** : Définissant l'en-tête du tableau.
2. **TBODY** : Définissant le corps du tableau.
3. **tfoot** : Définissant le pied de page du tableau.

Exemple d'un tableau utilisant ces attributs.

```

<HTML>
<HEAD><TITLE>Exemple d'un tableau</TITLE></HEAD>
<BODY>
<P>Exemple d'un tableau utilisant <BR>les attributs FRAME et RULES</P>
<BR>&nbsp;
<TABLE BORDER=2 WIDTH=300 FRAME=HSIDES RULES=COLS>
<THEAD>
<COLGROUP SPAN=3 ALIGN=LEFT VALIGN=TOP>
</THEAD>
<TBODY>
<TR>
<TD><B>Ligne 1</B></TD>
<TD>Cellule 1</TD>
<TD>Cellule 2</TD>
</TR>
<TR>
<TD><B>Ligne 2</B></TD>
<TD>Cellule 3</TD>
<TD>Cellule 4</TD>
</TR>
</TBODY>
</TABLE>
</BODY>
</HTML>

```



Les listes.

Liste ordonnée.

Une liste ordonnée s'obtient avec la commande `` qui permet d'avoir une liste numérotée qui s'incrémente automatiquement. La balise `` est définie avec `` qui représente un item.

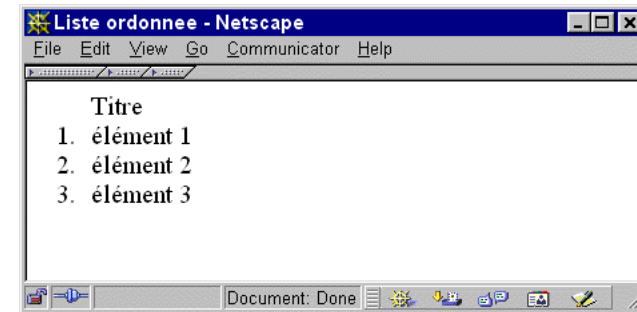
```
<OL COMPACT START=NUMBER TYPE=1|a|A|i|I> ... </OL>
```

La balise `<LH>` indique le titre de la liste. Exemple d'une liste ordonnée avec 3 items.

```

<HTML>
<HEAD>
<TITLE>Liste ordonn&eacute;e</TITLE>
</HEAD>
<BODY>
<OL>
<LH>Titre
<LI>&eacute;l&eacute;ment 1
<LI>&eacute;l&eacute;ment 2
<LI>&eacute;l&eacute;ment 3
</OL>
</BODY>
</HTML>

```



Les autres attributs pour la balise ``

- **TYPE=A** : Marqueurs en majuscules.
- **TYPE=a** : Marqueurs en minuscules.
- **TYPE=I** : Marqueurs en chiffres romains (majuscules).
- **TYPE=i** : Marqueurs en chiffres romains (minuscules).
- **TYPE=1** : Marqueurs en nombres (choix par défaut).
- **START=n** : Valeur de départ du marqueur.

Liste non ordonnée.

Une liste non ordonnée s'obtient avec la commande `` qui permet d'avoir une liste précédée d'une puce. La balise `` est définie avec `` qui représente un item. La balise `<LH>` indique le titre de la liste.

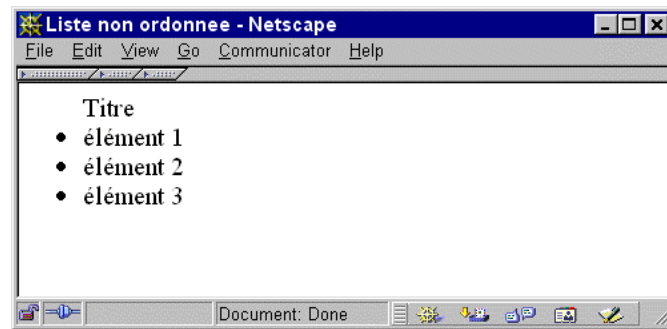
```
<UL COMPACT TYPE=disc|square|circle> ... </UL>
```

Les autres attributs pour la balise ``

1. **COMPACT** : Spécifie une liste compacte.
2. **TYPE=DISC** : Puce en forme de disque plein (valable que sous netscape).
3. **TYPE=CIRCLE** : Puce en forme de disque vide (valable que sous netscape).
4. **TYPE=SQUARE** : Puce en forme de carré (valable que sous netscape).

Exemple d'une liste non ordonnée avec 3 items.

```
<HTML>
<HEAD>
<TITLE>Liste non ordonn&eacute;e</TITLE>
</HEAD>
<BODY>
<UL>
<LI>Titre
<LI>&eacute;l&eacute;ment 1
<LI>&eacute;l&eacute;ment 2
<LI>&eacute;l&eacute;ment 3
</UL>
</BODY>
</HTML>
```



Liste menu.

La balise **<MENU>** permet d'avoir une liste menu. A noter que la balise **<MENU>** n'a que l'attribut **COMPACT**. Exemple d'une liste menu avec 3 items.

```
<HTML>
<HEAD>
<TITLE>Liste menu</TITLE>
</HEAD>
<BODY>
<MENU>
<LI>&eacute;l&eacute;ment 1
<LI>&eacute;l&eacute;ment 2
<LI>&eacute;l&eacute;ment 3
</MENU>
</BODY>
</HTML>
```



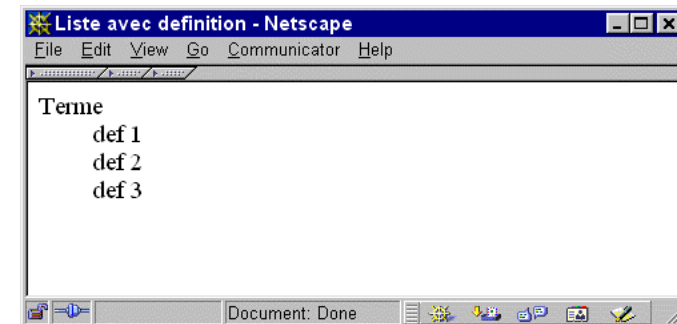
Liste avec définition.

Une liste de termes et de définitions correspondantes s'obtient avec la balise **<DL>** qui est accompagnée des balises **<DT>** et **<DD>**. **<DT>** représente le terme et **<DD>** la définition du terme. A noter que la balise **<DL>** n'a que l'attribut **COMPACT**.

```
<DL COMPACT>
<DT> item
<DD> text
</DL>
```

Exemple d'une liste de définition avec 1 terme et 3 définitions de 3 termes.

```
<HTML>
<HEAD>
<TITLE>Liste avec definition</TITLE>
</HEAD>
<BODY>
<DL>
<DT>Terme
<DD>def 1
<DD>def 2
<DD>def 3
</BODY>
</HTML>
```



Combiner les types de listes.

Vous pouvez bien entendu combiner plusieurs listes. Exemple d'une liste combinant une liste ordonnée avec une liste de définition.

```

<OL TYPE=I>
<LI>item 1
<DL>
<DT>Terme
<DD>def 1
</DL>
</OL>

```

Créer vos propres puces.

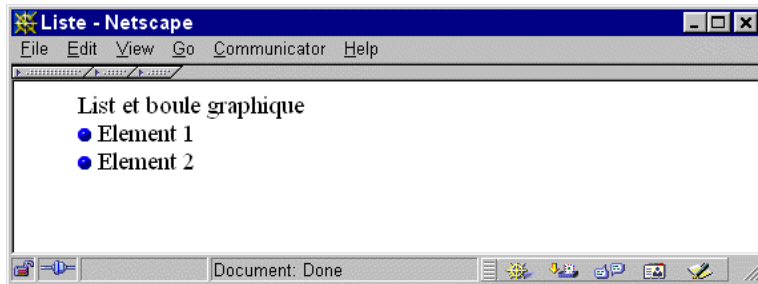
Deux solutions s'offrent à vous pour créer vos propres puces :

1. Soit vous passez par l'attribut `SRC="puces.gif"` si par exemple le fichier graphique de vos puces personnalisées se nomme `puces.gif`, mais cet attribut est spécifique à Internet Explorer.
2. Soit vous remplacez l'attribut la balise `` par la balise `` en définissant votre image comme un simple fichier `gif`. Cette solution est la meilleure car elle est valable pour tous les navigateurs, attention de faire un `break (
)` après chaque ligne. Exemple d'une liste non ordonnée avec un fichier gif.

```

<HTML>
<HEAD>
<TITLE>Liste</TITLE>
</HEAD>
<BODY>
<UL>
<LI>List et boule graphique<BR>
<IMG SRC="blueball.gif" BORDER=0 WIDTH=10 HEIGHT=10> Element 1
<BR>
<IMG SRC="blueball.gif" BORDER=0 WIDTH=10 HEIGHT=10> Element 2
</UL>
</BODY>
</HTML>

```



Les Formulaires.

Les formulaires permettent avant tout de faire réagir votre visiteur, de l'inviter à laisser ses impressions, ses commentaires. Vous pouvez même gérer votre propre livre d'or.

La création des formulaires se réalise grâce à plusieurs champs dans lesquels le visiteur entre une information ou opte pour un choix. L'information est ensuite envoyée à l'aide de scripts sur le serveur qui vous héberge puis renvoyée dans la plupart des cas sur l'adresse électronique de votre choix.

La balise <FORM>.

La balise `<FORM>` vient en début de votre formulaire, (la balise `</FORM>` ferme votre formulaire) elle possède 3 attributs.

```

<FORM [ACTION=URL] [ENCTYPE=cdata] [METHOD=GET|POST]> ... INPUT, TEXTAREA, SELECT
</FORM>

```

1. **ACTION** : Adresse le formulaire vers l'URL (relative) du script CGI (fourni par le serveur qui héberge les pages), qui accepte l'information et vous le retourne suivant les paramètres définis. Les valeurs les plus souvent couramment utilisées sont : "mailto:webmaster@serveur" : Pour l'envoi du formulaire par le logiciel de messagerie de votre visiteur. Attention cette valeur peut poser problème avec certains navigateurs notamment I.Explorer 3.0 "/cgi-bin/mailler" : Pour l'envoi du formulaire par le script CGI de votre hébergeur.
2. **METHOD** : 2 valeurs sont proposées **POST** : Pour un formulaire classique. **GET** : Pour, par exemple définir un moteur de recherche sur votre site.
3. **ENCTYPE** : Cet attribut spécifie le format des données envoyées dans le cas où un protocole n'imposerait pas un format précis. La valeur "text/plain" est normalement utilisée.

Champs de saisie <INPUT>.

La balise `<INPUT TYPE="text" ... >` permet à l'utilisateur de saisir une ligne de texte. Elle a les attributs suivants :

```

<INPUT [ALIGN=top|middle|bottom|left|right] [CHECKED] [MAXLENGTH=NUMBER] [NAME=cdata]
[SIZE=NUMBER] [SRC=URL] [TYPE=TEXT|PASSWORD|CHECKBOX]
RADIO|SUBMIT|RESET|FILE|HIDDEN|IMAGE|RANGE] [VALUE=cdata]>

```

1. **NAME** : Définit le nom de la donnée (obligatoire).
2. **SIZE** : Définit la taille visible du champ en nombre de caractères.
3. **MAXLENGTH** : Taille maximale de caractères autorisés.
4. **VALUE** : Définit le texte par défaut à l'intérieur du champ.

La valeur **TYPE** peut avoir la valeur password (code secret) à la place de text.

Boîte de texte multiligne <TEXTAREA>.

C'est la balise `<TEXTAREA>` qui est utilisée pour réaliser une boîte de texte multiligne. Cette balise est surtout choisie pour permettre aux visiteurs de votre site de laisser un commentaire ou de taper un texte quelconque. Ses attributs sont :

```

<TEXTAREA [COLS=NUMBER] [NAME=cdata] [ROWS=NUMBER]> ...</TEXTAREA>

```

1. **NAME** : Définit le nom de la donnée (obligatoire).
2. **ROWS** : Définit le nombre de lignes dans la boîte de saisie.
3. **COLS** : Définit le nombre de colonnes (en caractères) dans la boîte de saisie.
4. **READONLY** : Spécifie que la boîte multiligne est en mode lecture seulement.

Si du texte se trouve à l'intérieur des balises `<TEXTAREA>` et `</TEXTAREA>` il va alors apparaître comme texte par défaut.

Listes de choix <SELECT>.

La balise `<SELECT>` présente une liste de propositions soit sous forme de liste déroulante (1 seul choix possible), soit sous forme de menu (plusieurs choix possibles), elle doit être couplée avec la balise `<OPTION>` pour définir les choix. Voici ses attributs :

```
<SELECT [MULTIPLE] [NAME=cdata] [SIZE=NUMBER] ... </SELECT>
```

1. **NAME** : Définit le nom de la donnée (obligatoire).
2. **SIZE** : S'il a comme valeur 1 (valeur par défaut), c'est une liste déroulante si **SIZE** supérieur à 1 c'est une liste de choix avec une barre de défilement.
3. **MULTIPLE** : Ce choix est valable uniquement si **SIZE** est supérieur à 1.

La balise `<OPTION>`

```
<OPTION SELECTED VALUE=cdata>
```

1. **VALUE** : Nom de la donnée pour le choix.
2. **SELECT** : S'il est présent dans la balise `<OPTION>` le choix est alors reconnu comme choix par défaut.

Cases à cocher et Boutons radio.

La Balise `<INPUT>` que l'on a vu au-dessus permet aussi de représenter une liste de cases à cocher (plusieurs choix possibles) ou une liste dite "bouton radio" (1 seul choix possible), il suffit pour cela de lui spécifier l'attribut **TYPE="checkbox"** (cases à cocher) ou **TYPE="radio"** (bouton radio). Les autres attributs sont les suivants :

1. **NAME** : Définit le nom de la donnée (obligatoire).
2. **VALUE** : Nom de la donnée pour le choix.
3. **CHECKED** : S'il est présent dans la balise `<INPUT>`, le bouton sera coché par défaut.

Exemple d'une liste de cases à cocher et son code source. Cette liste se positionne verticalement du fait que la balise `
` a été insérée après chaque choix.

Soumettre le formulaire.

La soumission (envoyer) ou la remise à zéro (recommencer) d'un formulaire se fait une fois de plus avec la balise `<INPUT>` **TYPE** ayant ici deux autres valeurs :

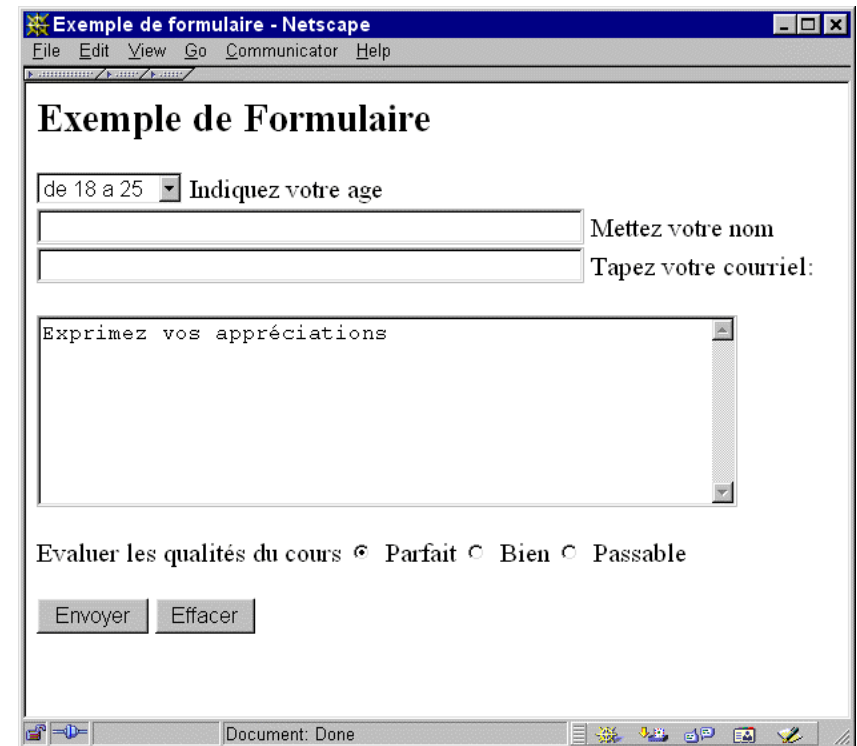
1. **SUBMIT** : Envoi le formulaire vers le serveur.
2. **RESET** : Remet à zéro les choix du formulaire.

Les légendes inscrites sur les boutons peuvent être modifiées avec l'attribut **VALUE**.

Exemple de formulaire :

```
<html> <head><title>Exemple de formulaire</title></head>
<body> <h2>Exemple de Formulaire</h2>
<form action="mailto:ivmad@vmei.acad.bg" method="POST">
<input type="hidden" name="subject" value="Exemple de Form">
<select name="age">
<option value="moins de 18"> moins de 18 </option>
<option selected value="de 18 a 25"> de 18 a 25 </option>
<option value="de 25 a 30"> de 25 a 30 </option>
<option value="de 30 a 40"> de 30 a 40 </option>
<option value="plus de 40"> plus de 40 </option>
</select> Indiquez votre age<br>
<input type="text" name="name" size=40> Mettez votre nom<br>
<input type="text" name="email" size=40> Tapez votre courriel:
<p><textarea name="details" cols=50 rows=7 wrap=physical>Exprimez vos
appréciations</textarea> <BR>&nbsp;<br> Evaluer les qualit&eacute;s du cours
<input type="radio" name="opinion" value="great" checked> Parfait
<input type="radio" name="opinion" value="middling"> Bien
<input type="radio" name="opinion" value="very poor"> Passable
<p>
```

```
<input type="submit" value="Envoyer">
<input type="reset" value="Effacer">
</form> </body> </html>
```



Mise en forme des formulaires.

Il y a mille et une façons de concevoir un formulaire, mais sachez avant tout que plus il sera clair et lisible, plus il sera complété par vos visiteurs. Pour cela vous pouvez opter pour des tableaux, des listes à puces, (pour les cases à cocher notamment), des listes de définition, et bien sur des balises de formatages et d'alignements de texte.

Inclure des données cachées.

Si vous souhaitez envoyer certaines données au script du serveur chargé de traiter votre formulaire, vous devez utiliser la balise `<INPUT>` avec l'attribut **TYPE="hidden"**.

Ex.

```
<INPUT TYPE="hidden" NAME="TO" VALUE="prof@refg.vmei.acad.bg">
```

Cette valeur indique sur quelle adresse électronique le formulaire sera envoyé.

```
<INPUT TYPE="hidden" NAME="FROM" VALUE="etudiant@refg.vmei.acad.bg">
```

Cette valeur indique la provenance du formulaire.

```
<INPUT TYPE="hidden" NAME="SUBJECT" VALUE="FORMULAIRE">
```

Cette valeur indique le nom du formulaire (le sujet du courrier), utile si vous avez plusieurs formulaires, cela permet de trier son courrier.

```
<INPUT TYPE="hidden" NAME="GOTOURL"  
VALUE="http://refg.vmei.acad.bg/cours/reponse1.html">
```

Cette valeur indique l'URL complète de la page html à afficher après l'envoi du formulaire.

Les frames.

Auparavant, les navigateurs ne pouvaient afficher qu'un seul fichier HTML mais avec la technologie développée par Netscape, on peut désormais diviser en plusieurs zones la page affichée par le navigateur, ces zones ou fenêtres ou encore cadres se nomment frames. Les frames sont apparues avec la version 3.0. Même si bon nombre de webmaster les utilisaient, elle n'étaient pas une composante officielle du langage.

Le conteneur <FRAMESET>.

La balise <FRAMESET> prend la place du tag <BODY>, c'est elle qui dans un premier temps va définir les cadres, qu'ils soient verticaux ou horizontaux, et leurs dimensions (en % ou en pixels).

```
<FRAMESET> [ROWS="x1,x2,...|x1%,x2%,...|x1*,x2*,..." ]  
[COLS="x1,x2,...|x1%,x2%,...|*" ] [FRAMEBORDER="yes|no|0" ] [BORDER="value" ]  
[BORDERCOLOR="#hexcolor" ] [FRAMESPACING=n]... </FRAMESET>
```

Les attributs ROWS et COLS :

Les attributs **ROWS** et **COLS** définissent le type de cadre, qui peut être vertical (**ROWS** - rangée) ou horizontal (**COLS** - colonnes). Ils prennent une liste de valeurs séparées par des virgules. Ces valeurs sont, sous forme de pourcentage entre 1 et 100 (valeurs relatives) ou sous forme de pixel (valeurs absolues).

Si une valeur manque ou est remplacée par une étoile (*) la colonne ou rangée s'ajustera automatiquement, de plus la hauteur de toutes les rangées ou la largeur de toutes les colonnes doit correspondre à la hauteur ou à la largeur de la fenêtre du navigateur, sinon la dernière valeur ne sera pas prise en compte et la fenêtre ou rangée s'ajustera automatiquement à l'écran.

Exemple de deux colonnes, qui ont comme valeur 20% et 80 %.

```
<FRAMESET COLS="20%,80%">
```

Exemple de deux colonnes, l'une d'une largeur de 50 pixels et l'autre s'ajustant automatiquement à la fenêtre du navigateur.

```
<FRAMESET COLS="50,*">
```

L'attribut FRAMEBORDER.

Cet attribut permet de déterminer si les cadres auront ou n'auront pas de bordure. Il a deux valeurs YES ou NO.

```
FRAMEBORDER=YES OU FRAMEBORDER=NO
```

L'attribut BORDER.

Cet attribut permet de déterminer la taille des bordures entourant les cadres. Il peut prendre plusieurs valeurs ex: BORDER=n n étant une valeur en pixels définissant la taille de la bordure. La valeur 0 indique aucune bordure.

```
BORDER=0 ou par exemple BORDER=5
```

L'attribut BORDERCOLOR.

Cet attribut permet de déterminer la couleur de l'ensemble des bordures des cadres. Il suffit pour cela de spécifier une couleur sous forme de nom ou de sa valeur hexadécimale.

```
BORDERCOLOR="red" ou BORDERCOLOR="#FF0000"
```

L'attribut FRAMESPACING.

Cet attribut permet de déterminer un espace entre les cadres. Il peut prendre plusieurs valeurs.

Ex:

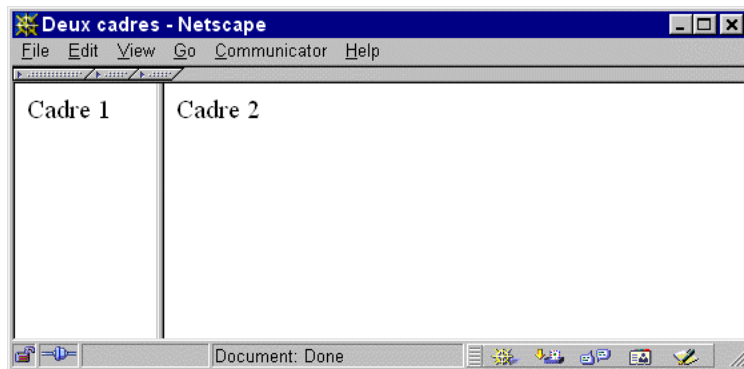
FRAMESPACING=n n étant une valeur en pixels définissant l'espace entre les cadres. La valeur 0 indique aucun espace.

FRAMESPACING=0 ou par exemple **FRAMESPACING=5**

Voici deux exemples (les plus courants) ainsi que le code source.

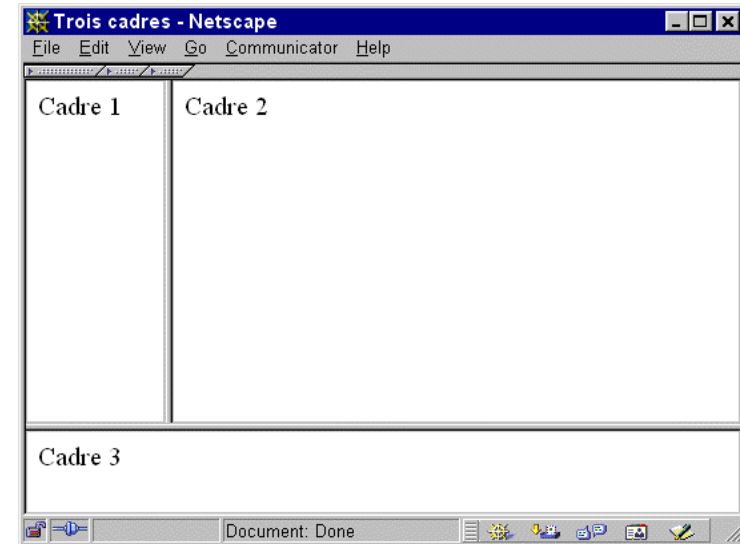
Le premier contient deux cadres horizontaux appelés Cadre 1 et Cadre 2.

```
<html>
<head><title>Deux cadres</title></head>
<FRAMESET COLS="20%,80%">
<FRAME SRC="cadre1.html" NAME="Z1">
<FRAME SRC="cadre2.html" NAME="Z2">
</FRAMESET>
</html>
```



Le deuxième contient aussi deux cadres horizontaux appelés Cadre 1 et Cadre 2 et, un cadre vertical appelé Cadre 3.

```
<html>
<head><title>Trois cadres</title></head>
<frameset rows="80%,20%">
<frameset cols="20%,80%">
<frame src="cadre1.html" name="z1">
<frame src="cadre2.html" name="z2">
</frameset>
<frame src="cadre3.html" name="z3">
</frameset>
</html>
```



La balise <FRAME>.

La commande **FRAME** permet de définir un cadre à l'intérieur du conteneur **<FRAMESET>**. Elle a plusieurs attributs, les plus importants étant **SRC** et **NAME**.

```
<FRAME> [SRC="URL"] [NAME="WINDOW_NAME"] [NORESIZE] [SCROLLING="YES|NO|AUTO"]
[MARGINWIDTH="VALUE"] [MARGINHEIGHT="VALUE"] [BORDERCOLOR="#HEXCOLOR"]
[FRAMESPACING="VALUE"] [FRAMEBORDER="YES|NO"] ... </FRAME>
```

L'attribut SRC.

Cet attribut indique l'URL (chemin) du document HTML qui sera affiché dans un cadre spécifique.

```
<FRAME SRC="cadre1.html">
```

L'attribut NAME.

Cet attribut permet de donner un nom à un cadre, ce qui permettra ensuite de l'appeler avec l'attribut **TARGET**, attention majuscule différent de minuscule.

```
<FRAME SRC="cadre1.html" NAME="Z1">
```

Les attributs MARGINWIDTH et MARGINHEIGHT.

MARGINWIDTH permet de spécifier la grandeur des marges de gauche et de droite du cadre créée, la valeur doit être exprimée en pixels, elle peut avoir comme valeur 0.

MARGINHEIGHT permet de spécifier la grandeur des marges de haut et de bas du cadre créée, la valeur doit être exprimée en pixels, elle peut avoir comme valeur 0.

```
<FRAME SRC="cadre1.html" NAME="Z1" MARGINWIDTH=2 MARGINHEIGHT=2>
```

L'attribut FRAMEBORDER.

Cet attribut permet de déterminer si les cadres auront ou n'auront pas de bordure. Deux valeurs YES ou NO.


```
<FRAME SRC="cadre1.html" NAME="Z1" FRAMEBORDER=NO>
```

L'attribut **BORDER** (Netscape uniquement).

Cet attribut permet d'ajuster l'espace entre les cadres. La valeur doit être en pixels.

```
<FRAME SRC="cadre1.html" NAME="Z1" BORDER=2>
```

L'attribut **NORESIZE**.

En utilisant cet attribut vous interdisez à l'utilisateur de redimensionner les cadres. Par défaut les cadres peuvent être redimensionnés.

```
<FRAME SRC="cadre1.html" NAME="Z1" NORESIZE>
```

L'attribut **SCROLLING**.

Cet attribut permet d'attribuer ou non une barre de défilement (scrollbar) à un cadre. Il possède trois valeurs :

1. **YES** : Indique que la barre de défilement sera toujours visible.
2. **NO** : Indique que la barre de défilement ne sera jamais visible (à tester avant de l'utiliser).
3. **AUTO** : Indique que le navigateur déterminera si la barre de défilement est nécessaire.

```
<FRAME SRC="cadre1.html" NAME="Z1" SCROLLING="AUTO">
```

Les navigateurs non compatibles.

Même si il est de plus en plus rare de trouver des navigateurs non compatibles il faut néanmoins ne pas négliger cet aspect sous peine de se passer de quelques visiteurs. La technique consiste à utiliser les balises **<NOFRAME>** et **</NOFRAME>**. Ces balises permettent de spécifier un texte HTML en version normale. Entre ces balises il faut donc théoriquement développer un deuxième site vous pouvez pour simplifier le travail préciser des regrets polis et insérer des liens vers les sites officiels de Microsoft et Netscape pour que le visiteur opte pour une mise à jour.

Exemple d'utilisation des balises **<NOFRAME>** et **</NOFRAME>**.

```
<FRAMESET ROWS="20%,80%">
<FRAME SRC="cadre1.htm" NAME="Z1">
<FRAME SRC="cadre2.htm" NAME="Z2">
</FRAMESET>
<NOFRAMES>
<BODY>
```

Désolé mais ce site comme vous pouvez le voir utilise la technique des frames. Merci de faire la mise à jour de votre navigateur chez Microsoft ou Netscape.

```
</BODY>
</NOFRAMES>
```

Cadres locaux (**<IFRAME>**).

La technique des cadres locaux s'obtient avec la balise **<IFRAME>** elle permet d'insérer une fenêtre à n'importe quel endroit de votre document. Cette fenêtre fera référence à un autre document HTML, la balise **<IFRAME>** doit être dans le corps de votre document, c'est-à-dire entre les balises **<BODY>** et **</BODY>**. Attention cette balise est spécifique à Internet Explorer malgré qu'elle a été approuvée et par le W3C. Elle a les même attributs que la balise **<FRAME>** c'est à dire:

```
<IFRAME [ SRC="URL" ] [ NAME="WINDOW_NAME" ] [ NORESIZE ] [ SCROLLING="YES | NO | AUTO" ]
[ MARGINWIDTH="VALUE" ] [ MARGINHEIGHT="VALUE" ] [ BORDERCOLOR="#HEXCOLOR" ]
[ FRAMESPACING="VALUE" ] [ FRAMEBORDER="YES | NO" ] [ HEIGHT=n ] [ WIDTH=n ]... </IFRAME>
```

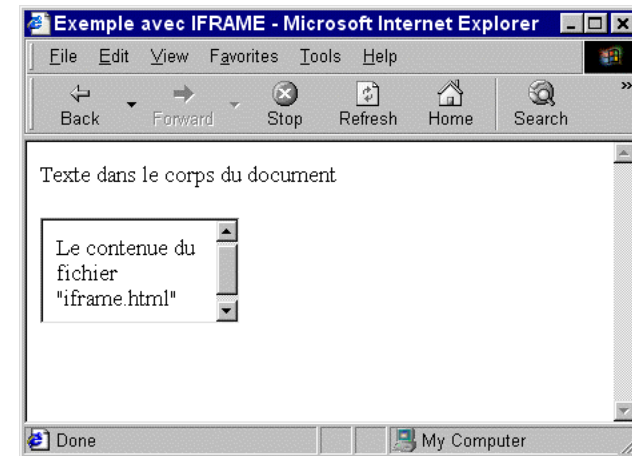
Elle possède en plus les attributs **HEIGHT** et **WIDTH** qui déterminent la hauteur et la largeur du cadre inséré dans votre page. Valeur (n) en pixels. Note : Un texte de remplacement (pour Netscape) peut être mis entre **<IFRAME>** et **</IFRAME>**.

Exemple d'un cadre local.

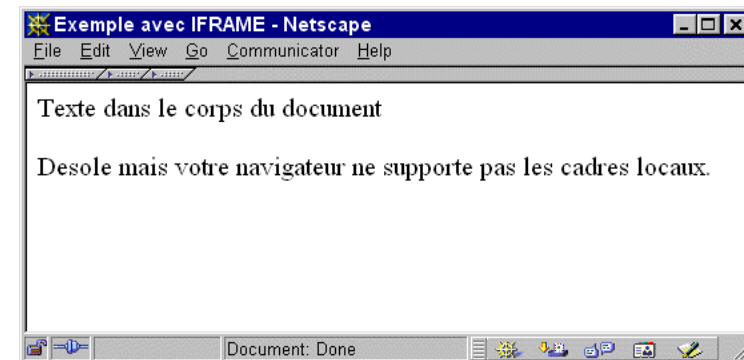
```
<html>
<head><title>Exemple avec IFRAME</title></head>
<body>
<p>Texte dans le corps du document</p>
```

```
<IFRAME SRC="iframe.html" NAME="Z11" HEIGHT="80" WIDTH="150"> Desole mais votre
navigateur ne supporte pas les cadres locaux. </IFRAME>
```

```
</body>
</html>
```



Affichage avec Internet Explorer 5.0.



Affichage avec Netscape Communicator 4.7.

Liens vers d'autres frames.

Normalement, les pages appelées par des liens s'affichent dans la frame où les liens ont été activés. L'attribut **TARGET** permet d'appeler un autre cadre par le nom défini avec l'attribut **NAME** précédemment. Mais il existe des noms réservés de frame cible, voici les différentes valeurs.

Valeur	Signification
<code>_self</code>	L'URL cible va s'afficher dans la même frame que le lien.
<code>_parent</code>	L'URL cible va s'afficher dans la structure externe.
<code>_blank</code>	L'URL cible va s'afficher dans une nouvelle fenêtre ouverte par le navigateur.
<code>_top</code>	L'URL cible va s'afficher dans toute la fenêtre du navigateur et les frames disparaissent.

Exemple de lien vers un autre cadre.

```
<A HREF="page1.html" TARGET="_Z2">
```

Exemple de lien externe avec la valeur `_top`.

```
<A HREF="http://refg.vmei.acad.bg" TARGET="_top">
```

Les feuilles de styles.

Internet Explorer à partir de la version 3.0 (attention la version d'Internet Explorer ne gère pas toutes les propriétés des feuilles de styles) et, Netscape à partir de la version 4 supportent les feuilles de styles.

Cette technologie a été introduite par Microsoft avec la version 3.0 (1996), le but : modifier le contenu d'une page par une grande variété d'effets surprenants (Dynamic HTML) et cela en un clin d'œil. En fait avec les feuilles de styles (**CSS** ou *Cascading Style Sheets*), on contrôle parfaitement la mise en page des différents éléments qui composent un document, par exemple vous pouvez définir la couleur d'une police, sa taille, le positionnement d'un objet (image, texte, ...), l'espacement entre les paragraphes, et même gérer des effets de transition.

La syntaxe.

Normalement, la syntaxe des déclarations **CSS** est toujours identique. La méthode est simple, il s'agit de redéfinir le rôle d'une balise HTML en lui imposant de nouvelles propriétés.

1. Comme les feuilles de styles ne sont pas une composante directe du HTML, vous pouvez définir à l'aide de la balise **META** (dans l'en-tête de la page) la déclaration suivante :

```
<META HTTP-EQUIV="Content-Style-Type" content="text/CSS">
```

2. Comme pour les navigateurs ignorant le *JavaScript*, vous devez cacher le code des feuilles de styles à l'aide des commentaires :

```
<!-- // -->
```

3. Quatre méthodes différentes sont possibles pour intégrer les feuilles de styles en HTML :
 - o L'instruction **STYLE** en tant qu'élément :

```
<STYLE TYPE="text/CSS">
```

- o L'élément **LINK** pour établir un lien vers une feuille de style externe (fichier texte avec l'extension `.css`) :

```
<LINK TITLE="TEST" TYPE="text/CSS" rel="stylesheet"
  HREF="ma_feuille.css">
```

- o L'instruction **STYLE** en tant qu'attribut :

```
<P STYLE="font-size: 12pt; color: green">
```

- o Importer une feuille de style (fichier texte avec l'extension `.css`) :

```
<STYLE TYPE="text/CSS">
@import url(http://refg.vmei.acad.bg/html/feuille.css);
</STYLE>
```

- o **Note : Le fichier `.css` ne doit pas contenir de code HTML.**

4. Pour définir une propriété, on indique d'abord la balise concernée puis entre deux accolades les caractéristiques de cette propriété (s'il y a plusieurs propriétés, on les sépare par un point virgule).

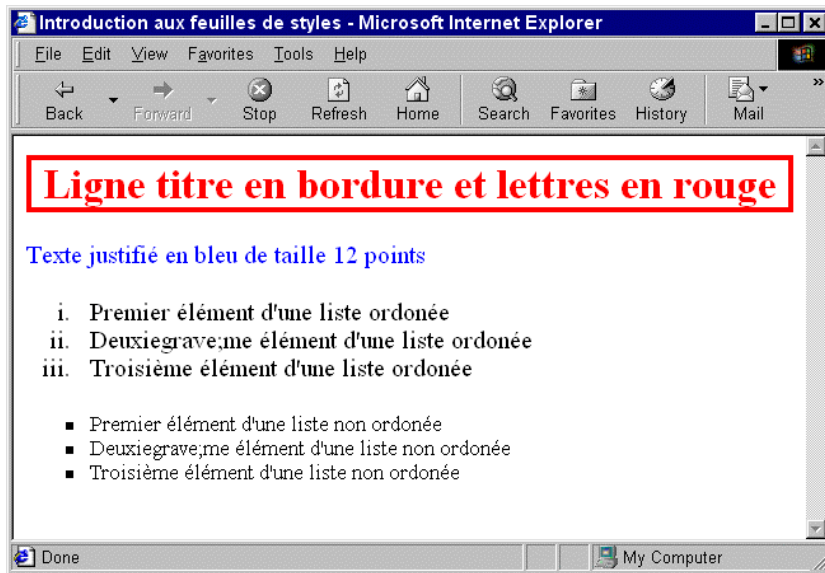
```
<HTML>
<HEAD>
<STYLE TYPE="text/css">
<!--
H1
{
font-size: 12pt;
color: green
}
-->
```

```
//-->
</STYLE>
</HEAD>
```

Si on opte pour la balise <DIV> la syntaxe sera :

```
<DIV STYLE="margin-left: 50px; color: blue">
Ce paragraphe sera placé à 40 pixels de la marge de gauche
et sera de couleur bleu.
</DIV>
```

Ex :



```
<HTML>
<HEAD>
<TITLE>Introduction aux feuilles de styles </TITLE>
<STYLE type="text/css">
<!--
H1 {border-width: 1; border: solid; text-align: center; color: red}
P {text-align: justify; color: blue; font-size: 12pt}
OL LI {list-style-type: lower-roman; font: bold normal 12pt times, serif}
UL LI {list-style-type: square}
<--
</STYLE>
</HEAD>
<BODY>
<H1> Ligne titre en bordure et lettres en rouge </H1>
<P> Texte justifié&eacute;, en bleu de taille 12 points </P>
<OL>
<LI> Premier &eacute;l&eacute;ment d'une liste ordon&eacute;e
<LI> Deuxi&eacute;me &eacute;l&eacute;ment d'une liste ordon&eacute;e
<LI> Troisi&eacute;me &eacute;l&eacute;ment d'une liste ordon&eacute;e
</OL>
<UL>
<LI> Premier &eacute;l&eacute;ment d'une liste non ordon&eacute;e
<LI> Deuxi&eacute;me &eacute;l&eacute;ment d'une liste non ordon&eacute;e
```

```
<LI> Troisi&eacute;me &eacute;l&eacute;ment d'une liste non ordon&eacute;e
</UL>
</BODY>
</HTML>
```

Propriétés et valeurs.

Voici un tableau résumant toutes les propriétés et valeurs possibles les plus courantes pour définir une feuille de style.

Propriété	Valeur(s)	Description	Exemple
font-size	xx-small, x-small, small, medium, large, x-large, xx-large, larger, smaller, ou taille défini en pixels (px) cm (cm), mm (mm), point (pt), pourcentage (%)	Taille de la police	{font-size: 12pt}
font-family	serif, sans-serif, cursive, fantasy, monospace, ou nom de la police (Arial, Verdana, etc...)	Type de police	{font-family: Arial}
font-weight	normal, bold, bolder, lighter, ou valeur numérique de 100 à 900 (par tranche de 100)	Poids de la police	{font-weight: bold}
font-style	normal, italic, oblique	Style de la police	{font-weight: italic}
font-variant	normal, small-caps	Variante de la police	{font-weight: small-caps}
line-height	point (pt), pourcentage (%)	Hauteur de la ligne	{line-height: 5pt}
color	nom ou valeur hexadécimale de la couleur	Couleur de la police	{color: green}
background-image	URL du fichier	Image d'arrière-plan	{background-image: url(image/fichier.gif)}
background-repeat	repeat, no-repeat, repeat-x, repeat-y	Répétition de l'arrière-plan	{background-repeat: no-repeat}
background-attachment	scroll, fixed	Effet de filigrane	{background-attachment: fixed}
background-position	top, center, bottom, left, right	Position de l'arrière-plan (verticale et horizontale)	{background-position: center center}
vertical-align	baseline, sub, super, top, text-top, middle, bottom, text-bottom, ou valeurs en pourcentage	Alignement vertical	{vertical-align: top}
text-align	left, right, center, justify	Alignement du texte	{text-align: justify}
text-indent	valeur en pixel ou pourcentage	Retrait de la première ligne	{text-indent: 10px}
text-decoration	none, underline, overline, line-through	Décoration du texte	{text-decoration: overline}
text-transform	capitalize, uppercase, lowercase, none	Casse du texte	{text-transform: none}
margin-left (ou -right, -top, -bottom)	valeur en pixel ou pourcentage	Valeurs des marges	{margin-right: 10px}
padding-left (ou -right, -top, -bottom)	valeur en pixel ou pourcentage	Remplissage du bloc	{padding-left: 15pt}
border-left (ou -right, -top, -bottom)	medium, thick, ou un style	Type de bordure	{border-top: medium}
border-x-width (x à remplacer par right, left, top, bottom)	medium, thick ou taille.	Taille de la bordure	{border-top-width: 15}

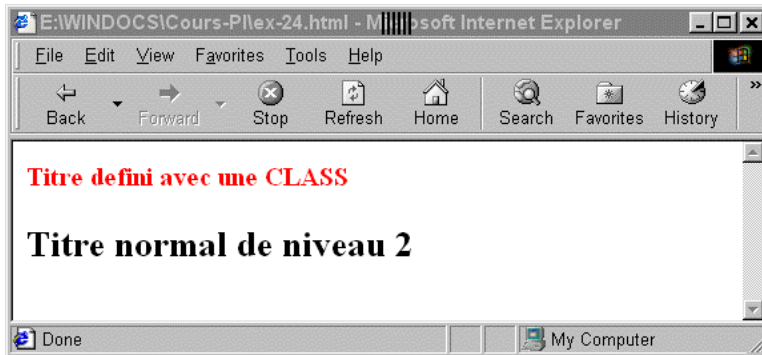
border-style	none, dotted, dashed, solid, double, groove, ridge, inset, outset	Style de la bordure	{border-style: ridge}
border-color	nom ou valeur hexadécimale de la couleur	Couleur de la bordure	{border-color: red}
float	none, left, right, both	Alignement d'un bloc par rapport à un autre	{float: right}
clear	none, left, right, both	Autorise l'alignement d'un bloc par rapport à un autre	{clear: right}
letter-spacing	normal ou valeur en point	Esacement des lettres	{letter-spacing: 2pt}

Les éléments CLASS et ID.

Comme nous l'avons vu plus haut, on peut redéfinir l'utilité d'une balise, mais si vous voulez que cette balise ne soit pas limitée aux seules feuilles de styles l'élément CLASS doit alors être utilisé. L'élément ID (identifiant) peut l'être aussi, sa différence avec CLASS se situe au niveau de l'utilisation, en effet on opte pour une ID de préférence lors d'un script JavaScript.

Exemple avec une CLASS.

```
<HTML>
<HEAD>
<STYLE TYPE="text/css">
<!-- H2.test
{ font-size: 12pt;
color: red }
/-->
</STYLE>
</HEAD>
</BODY>
<H2 class="test">Titre défini avec une CLASS</H2>
<H2>Titre normal de niveau 2</H2>
</BODY>
</HTML>
```



Exemple avec une ID (à noter le #).

```
<HTML>
<HEAD>
<STYLE TYPE="text/css">
<!--
#test
{
font-size: 10pt;

```

```
color: black
}
/-->
</STYLE>
</HEAD>
</BODY>
<H2 id="test">Titre défini avec une ID</H2>
<H2>Titre normal de niveau 2</H2>
<BODY>
</HTML>
```

Le paramètre MEDIA.

On utilise le paramètre **MEDIA** (à l'intérieur de la balise **STYLE**) pour obtenir des présentations différentes selon le type de média utilisé (ordinateur, synthétiseurs vocaux, ...).

Tableau résumant les valeurs disponibles pour le paramètre MEDIA.

Valeur	Information(s)
all	Pour tous les types de périphériques
screen	Type classique, pour tous les moniteurs courants
tty	Type télévision (basse résolution)
projection	Type projecteur
handheld	Type PC de poche (Psion, Windows CE, Pilot ...)
print	Type impression (pour les imprimantes)
braille	Pour les dispositifs de représentation en braille
aural	Pour les synthétiseurs vocaux

Exemple d'utilisation du paramètre MEDIA (ici pour les PC de poche et les PC standards).

```
<HEAD>
<STYLE TYPE="text/css" MEDIA="screen">
<!--
H2
{
font-size: 12pt;
color: black
}
/-->
</STYLE>

<STYLE TYPE="text/css" MEDIA="handheld">
<!--
H2
{
font-size: 10pt;
color: black
}
/-->
</STYLE>
</HEAD>
```

Les balises DIV et SPAN.

Les balises **DIV** et **SPAN** permettent de désactiver certaines instructions de la feuille de style, vous pouvez de cette façon modifier le style appliqué à certaines balises tout en conservant les autres effets définis dans le fichier CSS.

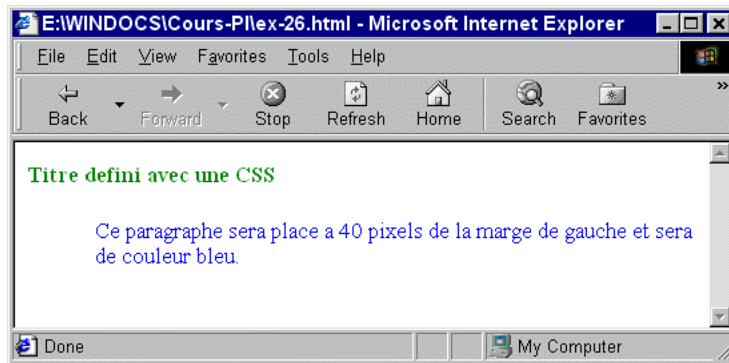
A noter que :

- **DIV** est normalement utilisé pour des blocs de texte entier.
- **SPAN** est lui, utilisé que pour des petites portions de texte (mots ou lettres).

SPAN et **DIV**, prennent en charge les attributs **ID** et **CLASS**.

Exemple avec DIV.

```
<HTML>
<HEAD>
<STYLE TYPE="text/css">
<!--
H1
{
font-size: 10pt;
color: green
}
/-->
</STYLE>
</HEAD>
</BODY>
<H1>Titre défini avec une CSS</H1>
<DIV STYLE="margin-left: 50px; color: blue">
Ce paragraphe sera placé à 40 pixels de la marge de gauche
et sera de couleur bleu.
</DIV>
<BODY>
<HTML>
```

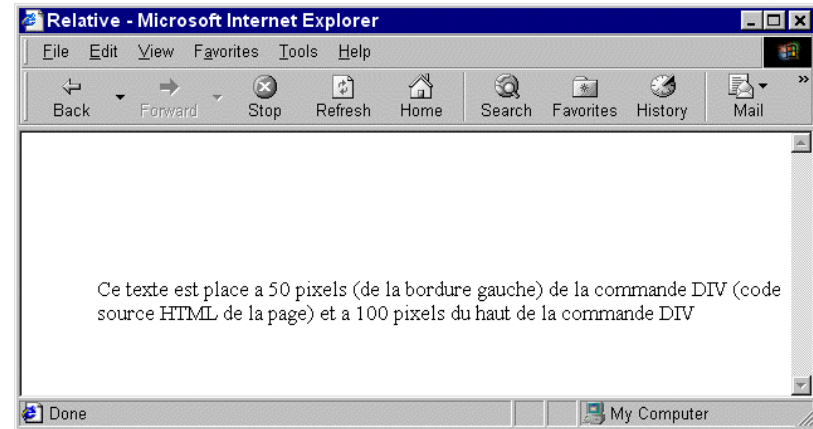


Le positionnement d'élément.

Un des avantages des feuilles de style, c'est le positionnement des éléments (bloc de textes, images ...) au pixels près !! fini les multitudes de balises (surtout les tableaux) pour essayer de faire une mise en page correcte. Pour positionner un élément il existe deux types de positionnement:

- Le positionnement absolu : C'est le plus rencontré, ce type de positionnement est identique aux logiciels de PAO tels que Publisher ou XPress, le placement d'un objet s'effectue à l'aide de boîte indépendante du reste du contenu de la page où l'on insère du texte ou des images.
- Le positionnement relatif : (le moins précis) il est identique au HTML, le navigateur affichera l'élément par rapport à l'emplacement de la balise concernée (dans le code source de la page).

Exemple de positionnement relatif :



`<DIV STYLE="position: relative; left: 50; top:100"> Ce texte est placé à 50 pixels (de la bordure gauche) de la commande DIV (code source HTML de la page) et à 100 pixels du haut de la commande DIV </DIV>`

Exemple de positionnement absolue

`<DIV STYLE="position: absolue; left: 50; top:100"> Ce texte est placé à 50 pixels de la bordure gauche de la fenêtre du navigateur et à 100 pixels du haut de cette même fenêtre </DIV>`

Note : La valeur z-index précise le chevauchement des éléments les uns par rapport aux autres (image en dessous du texte par exemple). Un élément sera affiché au-dessus d'un autre s'il possède un ordre d'affichage supérieur, par exemple si un élément à la valeur : {z-index: 2} il va alors recouvrir l'élément qui a comme valeur {z-index: 1}.

Les polices web (OpenType et TrueDoc).

Principe des polices Web.

L'intégration des polices Web dans une page HTML est un soulagement pour un bon nombre de webmasters. En effet avec ce système, votre page s'affichera avec la police désirée pour tous les navigateurs, car la police est fournie avec la page lors du téléchargement de celle-ci. Cependant Microsoft et Netscape ont une vision différente, pour Microsoft on parle de polices intégrées (OpenType) tandis que pour Netscape se sont des polices dynamiques (TrueDoc).

Intégration d'une police Web.

Cette intégration se réalise à l'aide de l'élément `@font-face` (cet élément existe aussi pour les feuilles de styles). Si la taille de votre police est importante, cela ralentira l'affichage de vos pages.

Schéma type d'une intégration d'une police dans une page Web. A intégrer entre les balises `<HEAD>` et `</HEAD>` de la page HTML.

```
<STYLE TYPE="text/css">
<!--
@font-face
{
propriété: valeur; propriété: valeur;
}
//-->
</STYLE>
```

Voici en exemple obéissant au schéma ci-dessus.

```
<STYLE TYPE="text/css">
<!--
@font-face
{
font-family: "Garamond"; src: url(police/garamond); format: (opentype);
}
//-->
</STYLE>
```

Dans cet exemple l'élément `@font-face` télécharge la police nommée Garamond (depuis le répertoire police), qui est une police OpenType (Microsoft). **Note** : La propriété `src` peut prendre la valeur local pour rechercher la police sur l'ordinateur de vos visiteurs avant de la télécharger depuis le serveur.

ex : `src: local(garamond); url(police/garamond);`

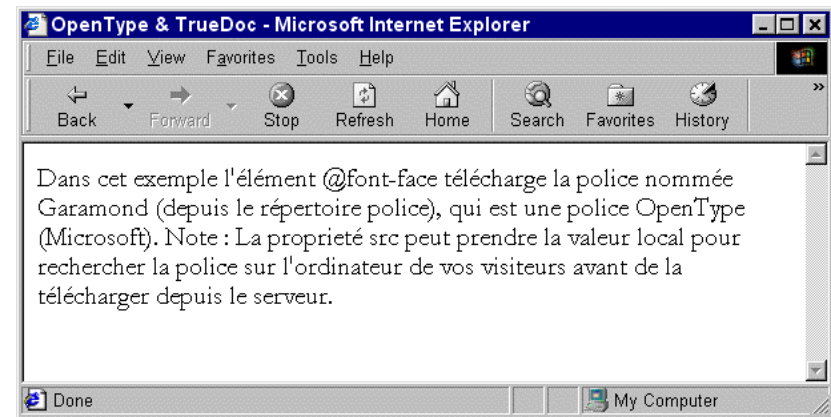
Ex:

```
<html>
<head>
<title>OpenType & TrueDoc</title>
<style TYPE="text/css">
<!--
@font-face
{
font-family: "Garamond"; src: local (Garamond);
url(police/Garamond); format: (opentype);
}
//-->
</style>
</head>
<body>
```

```
<font face="Garamond">Dans cet exemple l'élément @font-face
télécharge la police nommée Garamond (depuis le
```

```
répertoire police), qui est une police OpenType (Microsoft). Note : La
propriété src peut prendre la valeur local pour rechercher la police sur
l'ordinateur de vos visiteurs avant de la télécharger depuis le
serveur.</font>
```

```
</body>
</html>
```



Les propriétés d'une police.

Voici un tableau résumant les propriétés et valeurs d'une police. A noter qu'il est possible de fournir différentes descriptions pour une famille de police unique.

Propriété	Valeur(s)	Description
font-family	Arial, Verdana, ...	Nom de la Famille
font-style	norma, italic, oblique	Style d'écriture
font-variant	normal, small-caps	Normale, petite majuscule
font-weight	all, normal, bold, et valeurs de 100 à 800	Poids de la police
font-size	all, length	Taille fournie de la police

Intégrer une applet Java.

Le gros avantage de java, c'est l'interactivité, de plus java tourne sur toutes les plates-formes : PC, Macintosh, Station de travail SUN, (il faut néanmoins que le navigateur possède une console java, c'est à dire qu'il puisse comprendre ce langage). L'intégration d'applet java se passe avec la balise `<APPLET>` définie dans la version 3.2 du HTML, et qui a été remplacée depuis par la balise `<OBJECT>` née du HTML 4.0.

Note : Pour créer des applets, plusieurs logiciels sont possibles comme Javasoft disponible gratuitement à www.javasoft.com.

La balise `<APPLET>` possède plusieurs attributs :

- **NAME :** Nom de l'applet.
- **CODE :** Indique le nom de l'applet à intégrer dans la page. (L'extension d'une applet est normalement `.class`). **Attention** il se peut que pour un seul applet JAVA plusieurs fichiers `.CLASS` soient obligatoires.
- **CODEBASE :** Indique le répertoire où se situe l'applet.
- **WIDTH :** Spécifie la largeur de la fenêtre dans laquelle s'affichera l'applet.
- **HEIGHT :** Spécifie la hauteur de la fenêtre dans laquelle s'affichera l'applet.
- **ALT :** Permet de spécifier un texte de remplacement qui sera affiché si le navigateur ne supporte pas java.
- **ALIGN :** Indique l'alignement de l'applet sur la page (`RIGHT`, `LEFT`, `CENTER`).

Après avoir inséré la balise `<APPLET>` vous devez spécifier les propriétés de l'applet, ceci se réalise avec la balise `<PARAM>` qui possède plusieurs attributs dont les plus importants sont :

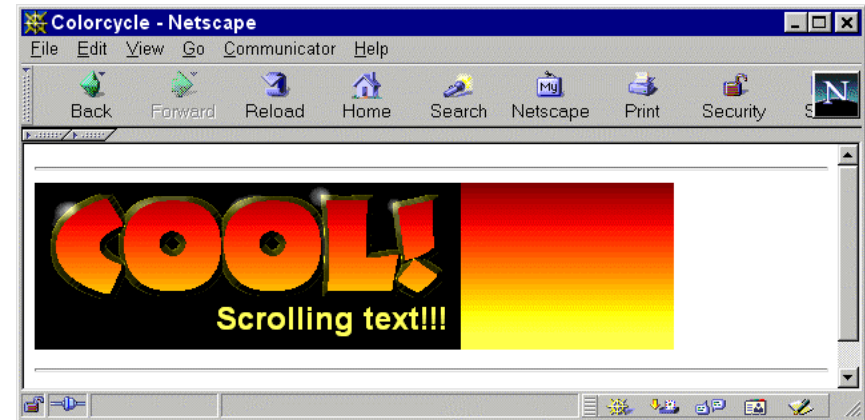
- **NAME :** Spécifie un nom à la propriété.
- **VALUE :** Spécifie les différentes valeurs.

Exemple d'intégration d'un applet (code source).

```
<APPLET CODE="random.class" ALT="Navigateur non compatible Java" WIDTH="50"
HEIGHT="50">
<PARAM NAME="sleeptime" VALUE="2">
<PARAM NAME="reg" VALUE="A00019">
<PARAM NAME="maxitems" VALUE="3">
<PARAM NAME="son0" VALUE="son.au">
<PARAM NAME="son1" VALUE="son1.au">
<PARAM NAME="background" VALUE="0">
</APPLET>
```

Ex:

```
<html>
<head>
<title>Colorcycle</title>
</head>
<body>
<hr>
<applet code=Colorcycle.class name=Colorcycle width=480 height=125 >
<param name=logo value="cool.gif">
<param name=factor value=4>
<param name=scroll value="Scrolling text!!!">
</applet>
<hr>
</body>
</html>
```



Intégrer un contrôle Active X.

En réalité un contrôle Active X (créé par Microsoft en 1990), c'est la technologie OLE (Liaison et Incorporation d'Objets) de Microsoft adaptée au HTML. Normalement les contrôles Active X sont réservés pour Internet Explorer (à partir de la version 3.0), néanmoins à l'aide du plug-in développé par [NCompass Labs](http://www.nccompass.com), vous pouvez les exploiter avec Netscape Navigator (à partir de la version 2.0). Les contrôles Active X peuvent être insérés avec la balise `<OBJECT>` qui remplace dans la norme HTML 4.0 (elle existait déjà dans le HTML 3.2 mais juste pour insérer des contrôles Active X) la balise `<EMBED>` et la balise `<APPLET>`.

Note : Microsoft propose une application qui vous aidera à configurer les contrôles Active X. Vous pouvez télécharger ActiveX Control Pad à l'adresse <http://www.microsoft.com/workshop/author/cpad/setuppad.exe>. Cette balise possède plusieurs attributs :

- **CLASSID :** Valeur fournie avec les contrôles Active X.
- **ID :** Nom unique identifiant le contrôle Active X.
- **WIDTH, HEIGHT, ALIGN, VSPACE, HSPACE,** définissant les propriétés de la fenêtre qui accueillera le contrôle Active X.

La balise `<PARAM>` décrit les paramètres du contrôle Active X comme la balise `<APPLET>` avec java.

Exemple d'intégration d'un contrôle Active X (code source).

```
<OBJECT CLASSID="clsid:4528XD-8956MSN" ID=Test WIDTH=320 HEIGHT=240>
<PARAM NAME="text" VALUE="2">
<PARAM NAME="color" VALUE="#FF0000">
<PARAM NAME="URL" VALUE="http://www.multimania.com/html">
</OBJECT>
```

Les balises propriétaires - Internet Explorer.

Voici une liste des principale balises "propriétaires" à Internet Explorer, je ne vous conseille pas de les utiliser (sauf exception comme un Intranet) car en général les visiteurs n'apprécient guère ...

Balise	Commentaire
<code><BGSOUND></code>	Intègre un arrière plan sonore.
<code><BODY BGGROUP></code>	Effet de filigrane.
<code><!--></code>	Permet de mettre un texte en commentaire..
<code><HR COLOR></code>	Spécifie une couleur pour un filet.

<IFRAME>	Insère un cadre (frame) local.
	Insère une image vidéo.
<MARQUEE>	Balise de défilement de texte.
<OBJECT>	Permet d'inclure des objets dans un document HTML.
<TD BACKGROUND>	Insère une image d'arrière plan pour une cellule.

Les balises propriétaires - Netscape.

Voici une liste des principales balises "propriétaires" à Netscape, je ne vous conseille pas de les utiliser (sauf exception comme un Intranet) car en général les visiteurs n'apprécient guère ...

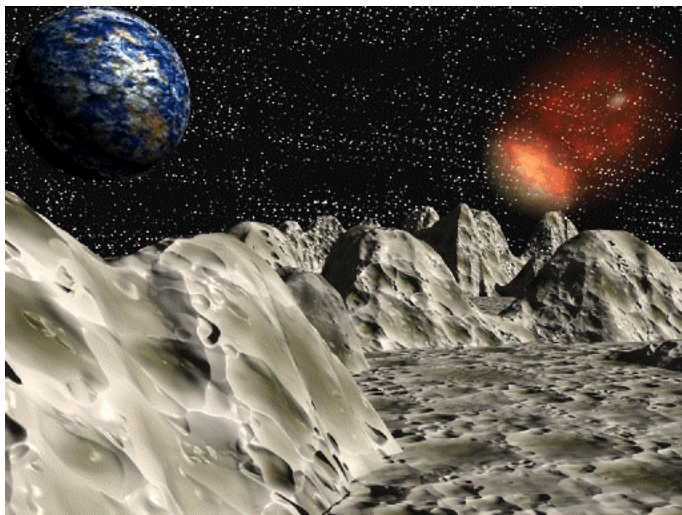
Balise	Commentaire
<BLINK>	Provoque un clignotement du texte.
<OL ou UL TYPE>	Spécifie le type de puce.
	Permet d'accélérer une image.
<LAYER>	Permet de superposer des blocs de textes ou images.
<SPACER>	Permet d'insérer un bloc d'espace vide dans votre document.

Les types de fichiers graphiques.

Il existe trois types de fichiers graphiques sur le Web - GIF, JPEG, PNG.

Le Format GIF.

Le GIF (Graphic Interchange Format) est certainement le format le plus utilisé sur le Web, il est limité à 256 couleurs donc il ne convient pas dans le cas de photographies très colorées ou avec beaucoup de nuances, par contre pour insérer un logo, icône ou même une banderole il est imbattable rapport qualité / taille. Ce format propose deux caractéristiques intéressantes : la possibilité de définir une couleur dite de transparence et l'affichage progressif (ou encore entrelacé). Exemple de gif avec et sans couleur de transparence. (Dans cet exemple la couleur noir est la couleur de transparence).



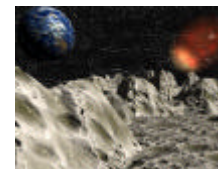
Le cas du GIF entrelacé se rencontre lorsque une image s'affiche progressivement d'abord floue puis de plus en plus nette. Il est utilisé pour charger un gif assez important (plus de 30 ko) sans pour autant pénaliser le lecteur qui explore vos pages pendant que les images se transfèrent petit à petit, sa taille est alors légèrement supérieure.

Pour créer un gif avec une couleur de transparence vous devez choisir le GIF 89a, le GIF 87a ne gère pas cet effet là.

Le Format JPEG.

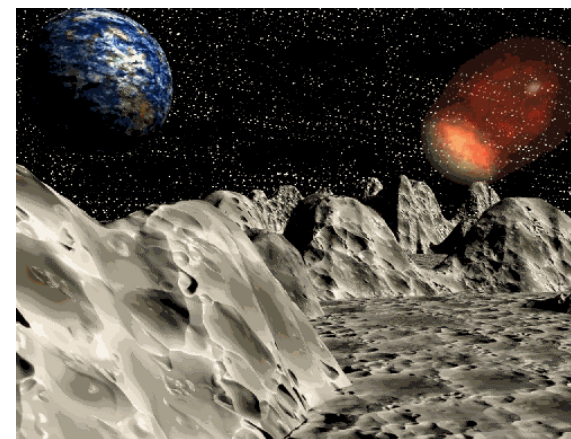
Le JPEG (Joint Photographic Experts Group) est un format particulièrement utilisé pour les photographies scannées riches de milliers de couleurs. En fait le JPEG réduit la taille d'une image en jouant sur sa qualité, il ne gère pas l'effet de transparence comme le GIF mais il dépasse la limite des 256 couleurs. Le taux de compression d'une image peut être déterminé entre 1 et 99 %. Exemples d'images compressées.

Vous remarquerez donc, que plus on augmente le taux de compression plus la qualité est mauvaise. Le meilleur taux se situe certainement dans la fourchette 10-30% Le JPEG gère également l'affichage progressif.



Le Format PNG.

Le Png (Portable Network Graphic) est un format très récent, c'est le format de l'avenir ... Ce format est en fait un "GIF - JPEG" à la fois. Il gère la transparence (plusieurs niveaux de transparence), et il ne se limite pas à 256 couleurs. Malheureusement la taille obtenue demeure trop importante, il se montre moins efficace que le JPEG pour la compression d'image pourtant la qualité est excellente.



Audio.

Note : Le téléchargement des plug-ins concernés dans cette page (RealAudio par exemple) est obligatoire !!

Les différents formats audio.

Voici un tableau récapitulant les différents formats musicaux ainsi que leurs origines.

Extension	Origines - Observations
.AIF ou .AIF	Format son du Macintosh.
.AU	Format son de SUN / NEXT (Unix).
.MID ou .MIDI	Musical Instrument Digital Interface.
.WAV ou .WAVE	Format son de Windows.
.MOD ou .XM ou .S3M	Format Musical (Amiga).
.RPM ou .RA ou .RM	Format RealAudio (ou RealVideo).
.IFF	Format son de l'Amiga.
.MP3	MPEG Layer 3.

Comment intégrer un son MIDI ou WAV.

L'intégration d'un fichier MIDI ou WAV (valable aussi pour .au ou .aiff) peut s'effectuer de différentes façons :

1. Soit avec la balise **<BGSOUND>**
2. Soit avec la balise **<A HREF>**
3. Soit avec la balise **<EMBED>**

La balise <BGSOUND>

Cette balise permet d'intégrer un arrière plan sonore (attention elle est spécifique à I.Eexplorer, elle ne fonctionnera pas avec Netscape), elle possède plusieurs attributs :

- **SRC** : Spécifie l'URL complète (chemin) du fichier à charger.
- **LOOP** : Spécifie le nombre de fois où le fichier sera joué (si la valeur est -1 le fichier sera joué indéfiniment).

ex : `<BGSOUND SRC="dico.mid" LOOP=2>`

La balise <A HREF>

Le plus simple pour insérer un son dans une page est d'appeler ce son par un lien hypertexte.

ex : `Ecoutez !!`

La balise <EMBED>

La balise permet d'inclure n'importe quel son, elle fait appel dans certains cas à un module externe (plugin). Attention elle n'est pas compatible avec certains navigateurs (I.Eexplorer 3.0 par exemple), c'est pour cela que vous pouvez utiliser les balises **<NOEMBED>** et **</NOEMBED>** (ignoré par les navigateurs qui comprennent la balise **<EMBED>**) pour fournir un texte, ou une image de remplacement et même la balise **<BGSOUND>**

Voici les attributs de la balise **<EMBED>** :

1. **ALIGN** : Aligne la console ou le texte (4 valeurs possibles, **RIGHT**, **LEFT**, **TOP**, **BOTTOM**).
2. **BORDER** : Largeur de la bordure du cadre (en pixels).
3. **WIDTH** : Largeur de la fenêtre qui accueillera la console.
4. **HEIGHT** : Hauteur de la fenêtre qui accueillera la console.
5. **PLUGINSOURCE** : URL de la page Web contenant les informations sur le plug-in (facultatif).
6. **SRC** : URL complète (chemin) du fichier.
7. **AUTOSTART** : Spécifie si l'objet multimédia est joué automatiquement (**TRUE**) ou non (**FALSE**) (défaut : **TRUE**).
8. **AUTOLOAD** : Spécifie si le chargement de l'objet multimédia s'effectue automatiquement (**TRUE**) ou non (**FALSE**) (défaut : **TRUE**).
9. **LOOP** : Spécifie le nombre de fois où le fichier sera joué (si la valeur est -1 le fichier sera joué indéfiniment).
10. **TYPE** : Indique le type **MIME** ex: **TYPE="audio/mod"** (facultatif).
11. **CONTROLS** * : Indique l'aspect de la console de contrôle :
 - o **CONSOLE** : Affiche toute la console (choix par défaut).
 - o **SMALL CONSOLE** : Affiche une console réduite.

- o **PLAYBUTTON** : Affiche seulement le bouton lecture.
- o **STOPBUTTON** : Affiche seulement le bouton stop.
- o **PAUSEBUTTON** : Affiche seulement le bouton pause.
- o **VOLUMELEVER** : Affiche seulement le bouton volume.

12. **HIDDEN** : Cache le panneau de contrôle.
13. **VOLUME** * : Volume initial en pourcentage.
14. **STARTIME** * : Indique le temps de départ par rapport au début du fichier (sous forme mm:ss).
15. **STRTIME** * : Indique le temps de fin par rapport au début du fichier (sous forme mm:ss).

A noter que quelques attributs sont spécifiques à certains plug-ins. Exemple d'un fichier MIDI avec la balise **<EMBED>**

```
<EMBED SRC="other/funky2.mid" WIDTH=145 HEIGHT=60 AUTOSTART="FALSE">
```

Comment intégrer du son au format .MP3.

L'intégration d'un fichier .MP3 est identique à l'intégration d'un fichier MIDI ou WAV. Attention tout de même à la taille de votre fichier (il ne doit pas être supérieur à 40-60 Ko, sinon proposez le en téléchargement). Pour encoder un fichier au format .MP3 il faut un encodeur, voilà une petite liste de liens où l'on peut trouver de nombreux utilitaires pour travailler les fichiers .MP3 :

Intégrer du son avec RealAudio.

L'intégration d'un fichier RealAudio est similaire à un fichier **MIDI** ou **WAV**, par contre l'attribut **EMBED** possède certains attributs particuliers, notamment l'attribut **CONTROLS** qui possède des valeurs spécifiques:

- **All** : La fenêtre de contrôle est complète.
- **ControlPanel** : Intègre les boutons play, pause, et stop ainsi que le curseur de position.
- **InfoVolumePanel** : Intègre une fenêtre d'information ainsi que le curseur de volume.
- **InfoPanel** : N'intègre que la fenêtre d'information.
- **Statusbar** : Intègre une barre d'état.
- **Playbutton** : Intègre les boutons play et pause.
- **Stopbutton** : N'intègre que le bouton stop.

L'attribut **CONSOLE=nom** permet, si vous avez plusieurs fichiers sur une même page de les relier par le même nom et donc d'obtenir les mêmes attributs pour **CONTROLS**.

Exemple d'un fichier RealAudio avec l'élément **ControlPanel**.

```
<EMBED SRC="other/exreal.ra" CONTROLS="ControlPanel" WIDTH=200 HEIGHT=35
AUTOSTART="FALSE">
```

Principe du streaming.

Le principe du streaming est simple, il s'agit d'une méthode qui permet d'obtenir un téléchargement fluide : Le fichier sonore (ou vidéo) se chargera petit à petit, ce qui sera transparent et, non pénalisant pour le visiteur. Intégrer un son avec RealPlayer (même principe pour une vidéo), en faisant appel au streaming, se passe en trois étapes.

1. Créer le fichier au format RealAudio.
2. Création du Metafile.
3. Intégrer le fichier final dans votre page.

1 - Créer le fichier au format RealAudio.

Pour créer un fichier RealAudio, vous devez passer par l'utilitaire Real Encoder 3.1 (audio seulement) ou Real Encoder 5 (audio et vidéo) disponible gratuitement à : <http://www.real.com/products/tools.html> Cet utilitaire permet de transformer un fichier .wav en .ra, son utilisation est très simple.

2 - Création du Metafile.

Le Metafile est juste un fichier ayant l'extension .ram et, contenant une ligne qui appellera votre fichier RealAudio. Exemple si votre fichier s'appelle audio.ra, qu'il a été transféré par FTP dans un répertoire se nommant son et que l'adresse de votre site est www.music.com.

<http://www.music.com/son/audio.ra>

La création de ce fichier que l'on nommera meta.ram se fait tout simplement avec un éditeur de texte (le bloc-note par exemple). Attention tout de même aux majuscules et minuscules.

3 - Intégrer le fichier final dans votre page..

Pour relier votre page à ce Metafile, il suffit de créer un lien hypertexte qui appellera celui-ci et lancera RealPlayer. Exemple si vous avez transféré votre Metafile sur la racine de votre site

`Du streaming avec RealAudio`

Il est possible d'opter pour la balise `<EMBED>` à la place du lien.

Compatibilité des navigateurs avec les balises multimédia.

Voici un tableau récapitulatif des différentes balises pouvant accepter des fichiers multimédia ainsi que les navigateurs compatibles avec celles-ci. Comme vous pouvez le voir, il n'est pas de tout repos de proposer du multimédia pour tous les visiteurs.

	Netscape 2.0	IE Explorer 2.0	Netscape 3.0 et 4.x	IE Explorer 3.0, 4.x, 5
<code><BGSOUND></code>	Non	Oui	Non	Oui
<code></code>	Non	Non	Non	Oui
<code><EMBED></code>	Non	Non	Oui	Oui
<code><NO EMBED></code>	Balise non ignorée	Balise non ignorée	Balise ignorée	Balise ignorée
<code><OBJECT></code>	Non	Non	Oui (seulement 4.0)	Oui (seulement 4.0)

Video.

Les différents formats vidéo.

Voici un tableau récapitulatif des différents formats vidéo ainsi que leurs origines.

Extension	Origines - Observations
.AVI	Vidéo for Windows.
.MOV	Fichier Quicktime (Apple - Mac).
.MPEG ou .MPE ou .MPV	Motion Picture Experts (MPEG-1).
.RPM ou .RA ou .RM	Format RealVideo (ou RealAudio).
.ASF ou .ASX	Format Netshow (Microsoft).

Intégrer une vidéo AVI.

L'intégration d'un fichier AVI peut s'effectuer de différentes façons :

1. Soit avec la balise `<A HREF>`
2. Soit avec la balise `<EMBED>`

La balise `<A HREF>`

Le plus simple pour insérer une vidéo dans une page est d'appeler cette vidéo par un lien hypertexte.

ex : `Ouvrez les yeux !!`

La balise `<EMBED>`

Cette balise permet d'inclure n'importe quelle vidéo, elle fait appel dans certains cas à un module externe (plugin). Attention elle n'est pas compatible avec certains navigateurs (IE Explorer 3.0 par exemple), c'est pour cela que vous pouvez utiliser les balises `<NOEMBED>` et `</NOEMBED>` (ignorés par les navigateurs qui comprennent la balise `<EMBED>`) pour fournir un texte, ou une image de remplacement. Voici les attributs standards de la balise `<EMBED>` :

1. **ALIGN** : Aligne la console ou le texte (4 valeurs possibles, **RIGHT**, **LEFT**, **TOP**, **BOTTOM**).
2. **BORDER** : Largeur de la bordure du cadre (en pixels).
3. **WIDTH** : Largeur de la fenêtre qui accueillera la console (en pixels).
4. **HEIGHT** : Hauteur de la fenêtre qui accueillera la console (en pixels).
5. **SRC** : URL complète (chemin) du fichier.
6. **AUTOSTART** : Spécifie si la vidéo est jouée automatiquement (**TRUE**) ou non (**FALSE**) (défaut : **TRUE**).
7. **AUTOLOAD** : Spécifie si le chargement de la vidéo s'effectue automatiquement (**TRUE**) ou non (**FALSE**) (défaut : **TRUE**).
8. **LOOP** : Spécifie le nombre de fois où la vidéo est jouée (si la valeur est -1 le fichier sera joué indéfiniment).
9. **CONTROLS** : Indique si le panneau de contrôle s'affiche (**TRUE**) ou non (**FALSE**) (défaut : **TRUE**).
10. **TYPE** : Indique le type **MIME** ex: **TYPE="audio/mod"** (facultatif).

Exemple d'un fichier AVI avec la balise `<EMBED>`

`<EMBED SRC="other/video.avi." WIDTH=200 HEIGHT=200 AUTOSTART="FALSE" LOOP=1>`

Intégrer une vidéo Quicktime.

Le module **Quicktime** (Apple) permet d'afficher des vidéos de type **.MOV** (mais aussi les formats **AVI**, **MIDI**, **WAV** ... avec Quicktime 3.0). L'intégration de vidéo Quicktime est identique au format **AVI**, cependant la balise `<EMBED>` possède plusieurs attributs particuliers :

1. **AUTOPLAY** : indique si la vidéo est jouée automatiquement (**TRUE**) ou non (**FALSE**) (défaut : **TRUE**).
2. **CONTROLLER** : Affiche la barre d'outils (**TRUE**) ou non (**FALSE**) (défaut : **TRUE**). Si elle est affichée, pensez à augmenter la valeur **HEIGHT** de **24** pixels.
3. **LOOP** : Indique si la vidéo diffusée se fait en boucle.
4. **PLAYEVERYFRAME** : Indique si la vidéo s'exécute au fur et à mesure du téléchargement (**TRUE**) ou non (**FALSE**). Equivalent au streaming mais l'image sera saccadée).
5. **TARGET** : Identique pour un lien vers une frame.
6. **HREF** : Lien avec la vidéo.
7. **PAN**, **TILT**, **FOV**, **NOD**, et **CORRECTION** sont utilisés uniquement pour la diffusion de film **VR** (panoramique).

Note : Attention ce n'est pas parce que vous avez Quicktime installé sur votre machine qu'il se lancera automatiquement, il faut d'abord le définir dans le type **MIME** de Windows sinon, c'est le contrôle **ActiveMovie** qui s'exécutera, ce qui à part l'interface, donnera le même résultat.

Exemple d'un fichier MOV avec la balise `<EMBED>`

`<EMBED SRC="other/video.mov" WIDTH=200 HEIGHT=200 CONTROLLER="FALSE">`

Intégrer une vidéo RealVideo.

L'intégration de Vidéo avec RealVideo est similaire à l'intégration de fichier audio avec RealAudio, elle se fait donc soit avec un lien hypertexte soit avec la balise `<EMBED>` qui possède des attributs identiques à l'insertion d'une vidéo AVI sauf **CONTROLS** qui possède des valeurs spécifiques :

- **ALL** : La fenêtre de contrôle est complète.

- **ControlPanel** : Intègre les boutons play ,pause et stop ainsi que le curseur de position.
- **InfoVolumePanel** : Intègre une fenêtre d'information ainsi que le curseur de volume.
- **InfoPanel** : Intègre seulement la fenêtre d'information.
- **Statusbar** : Intègre une barre d'état.
- **Playbutton** : Intègre les boutons play et pause.
- **Stopbutton** : Intègre seulement le bouton stop.

L'attribut `CONSOLE=nom` permet, si vous avez plusieurs fichiers sur une même page de les relier par le même nom et donc d'obtenir les mêmes attributs pour `CONTROLS`.

Exemple d'un fichier RealVidéo.

```
<EMBED SRC="other/video.ra" CONTROLS="InfoPanel" WIDTH=200 HEIGHT=35
AUTOSTART="FALSE">
```

Principe du streaming.

Le principe du streaming est simple, il s'agit d'une méthode qui permet d'obtenir un téléchargement fluide : Le fichier vidéo (ou sonore) se chargera petit à petit, ce qui sera transparent et, non pénalisant pour le visiteur.

Intégrer une vidéo avec RealPlayer (même principe pour un fichier audio), en faisant appel au streaming, se passe en trois étapes.

1. Créer le fichier au format RealVidéo.
2. Création du Metafile.
3. Intégrer le fichier final dans votre page.

1 - Créer le fichier au format RealVidéo.

Pour créer un fichier Realvidéo, vous devez passer par l'utilitaire Real Encoder 5 (audio et vidéo) disponible gratuitement à : <http://www.real.com/products/tools.html> Cet utilitaire permet de transformer un fichier .avi (ou tout autre fichier vidéo) en .ra, son utilisation est très simple.

2 - Création du Metafile.

Le Metafile est juste un fichier ayant l'extension .ram et, contenant une ligne qui appellera votre fichier RealVidéo. Exemple si votre fichier s'appelle video.ra, qu'il a été transféré par FTP dans un répertoire se nommant film et que l'adresse de votre site est www.tv.com .

<http://www.tv.com/film/video.ra>

La création de ce fichier que l'on nommera meta.ram se fait tout simplement avec un éditeur de texte (le bloc-note par exemple). Attention tout de même aux majuscules et minuscules.

3 - Intégrer le fichier final dans votre page..

Pour relier votre page à ce Metafile, il suffit de créer un lien hypertexte qui appellera celui-ci et lancera RealPlayer. Exemple si vous avez transféré votre Metafile sur la racine de votre site

```
<A HREF="meta.ram">Du streaming avec RealVidéo</A>
```

Il est possible d'opter pour la balise `<EMBED>` à la place du lien.

Compatibilité des navigateurs avec les balises multimédia.

Voici un tableau récapitulant les différentes balises pouvant accepter des fichiers multimédia ainsi que les navigateurs compatibles avec celles-ci. Comme vous pouvez le voir, il n'est pas de tout repos de proposer du multimédia pour tous les visiteurs.

	Netscape 2.0	IE Explorer 2.0	Netscape 3.0 et 4.x	IE Explorer 3.0, 4.x, 5
<BGSOUND>	Non	Oui	Non	Oui
	Non	Non	Non	Oui
<EMBED>	Non	Non	Oui	Oui
<NO EMBED>	Balise non ignorée	Balise non ignorée	Balise ignorée	Balise ignorée
<OBJECT>	Non	Non	Oui (seulement 4.0)	Oui (seulement 4.0)

Les éditeurs HTML Wysiwyg.

Généralités.

Les Webmasters attendaient avec impatience les premiers éditeurs *Wysiwyg* Wysiwyg veut dire, **What You See is What You Get**, autrement dit *ce que vous voyez est ce que vous obtenez*. Nous devrions parler d'éditeur Wymolwyyet (**What You See is more or less What You Get**), en effet, le premier postulat du HTML est de décrire un texte avec ses conventions typographiques. Ainsi le HTML n'impose donc pas de rendu particulier puisque c'est un langage de description de structure de document.

Nous traitons donc les éditeurs qui permettent de saisir les informations affichées sur votre page Web sans se soucier du codage HTML qui n'apparaît plus.

Rappelons pour mémoire que les meilleurs éditeurs HTML de la génération précédente, n'étaient que des éditeurs texte permettant d'ajouter des balises HTML par des macro commandes, ou avec des Wizard.

Nous rappelons les meilleurs éditeurs de cette génération avant d'étudier les nouveaux éditeurs : Webmaster Pro, WebEdit, HTMLed PRO, HTML Assistant Pro, HoTMetaL Pro, Hot Dog Pro, HomeSite, AOLPress/Navipress et WebExpert.

Le meilleur éditeur donné par la presse américaine était généralement HotDogPro, mais le monde francophone préfère WebExpert, logiciel québécois, entièrement en langue française et supportant quelques programmes JavaScript.

Le logiciel WebExpert est disponible à l'adresse <http://www.visic.com/webexpert/index.html>

La liste des éditeurs Wysiwyg

- **Front Page 2000** de Microsoft est un très bon produit, gérant le téléchargement des pages sur le serveur Http (Version française).
- **Netscape Gold** . Netscape Gold 3.04 est une version du navigateur Netscape qui dispose d'un mode de fonctionnement permettant l'édition Wysiwyg. La partie éditeur est moins bien conçue que celle de Front Page. (Version française).
- **Word IA** : est une extension de Word 6.0 qui permet de transformer Word en un éditeur/ Navigateur. Cette version ne supporte que le HTML 2 et ne permet donc pas une mise en page très ergonomique. De plus les efforts de Microsoft vont vers le développement de FrontPage.
- **Incontext Spider** se rapproche d'une version *Wysiwyg* et est un éditeur de bon niveau mais ne supporte pas toutes les balises. Incontext est une société spécialisée dans le domaine SGML et son savoir faire nous laissait en droit de penser que son éditeur serait meilleur.
- **Netscape Composer** : (du package Communicator) encore en version évoluée de Netscape Communicator 4.7. C'est la réponse de Netscape à Microsoft, elle ressemble beaucoup à la version Netscape Gold. Elle devrait intégrer prochainement la gestion de feuille de style, la gestion des profils utilisateurs et un correcteur orthographique. Très bon pour débutants.
- **Dreamweaver** de Macromedia est le meilleur outil testé. Il est rapide, exempt de bogues et est muni d'une interface très ergonomique et très agréable. Seul le prix peut faire réfléchir l'amateur, mais il reste le meilleur investissement pour le professionnel. Dernière version 3.0.
- **Claris Home Page** : premier éditeur *Wysiwyg* sur Macintosh, qui a su gérer le multi-fenêtrage (FRAME). La version 2.0 n'est pour le moment disponible que sur Windows 32 bits. Elle intègre un dictionnaire et une librairie de ClipArt. (Anglais)
- **AOL Press** : dernier né et très remarqué dans sa version 2.0 car la version 1.0 l'avait placé au tout premier plan des éditeurs. Mais cette version est moins aboutie que Front Page et surtout que DreamWeaver. Mais la gratuité de ce logiciel est un argument de poids.

- **PageMill** : d'Adobe dans sa version 3.0 pour Windows , ce logiciel offre une très belle ergonomie et une gestion des fenêtres (*frames*) des plus abouties.
- **MS Word 2000** : Exporte les fichiers doc en format xhtml 1.0.

Les principaux editeurs (total) wysiwyg et non wysiwyg :

Nom Editeurs	WYSIWYG ?	URL
<i>Arachnophilia</i>	<i>Non</i>	<i>http://www.arachnoid.com</i>
<i>Dreamweaver 3.0</i>	<i>Oui</i>	<i>http://www.macromedia.com/fr</i>
<i>FrontPage 2000</i>	<i>Oui</i>	<i>http://www.microsoft.com/france</i>
<i>FrontPage Express</i>	<i>Oui</i>	<i>http://www.microsoft.com/france</i>
<i>Adobe GoLive 1.0</i>	<i>Oui</i>	<i>http://www.adobe.fr</i>
<i>Home Page</i>	<i>Oui</i>	<i>http://www.clarishomepage.com</i>
<i>Allaire Home Site</i>	<i>Non</i>	<i>http://www.allaire.com</i>
<i>Namo WebEditor</i>	<i>Non</i>	<i>http://www.namo.com</i>
<i>Netscape Composer</i>	<i>Oui</i>	<i>http://www.netscape.com/fr</i>
<i>Adobe PageMill 3.0</i>	<i>Oui</i>	<i>http://www.adobe.fr</i>
<i>Visual Page</i>	<i>Oui</i>	<i>http://www.symantec.fr</i>
<i>Web Construction</i>	<i>Non</i>	<i>http://www.pierresoft.com</i>
<i>WebExpert</i>	<i>Non</i>	<i>http://www.visic.com</i>