Introduction au HTML

Présentation du HTML

Le **HTML** ("*HyperText Markup Language*") est un langage dit de "marquage" chargé de formaliser l'écriture d'un document avec des balises de formatage indiquant la façon dont doit être présenté le document et les liens qu'il établit avec d'autres documents.

Il permet notamment la lecture de documents sur Internet à partir de machines différentes grâce au protocole HTTP, permettant d'accèder via le réseau à des documents repérés par une adresse unique, appelée URL.

On appelle **World Wide Web** (noté *WWW*) ou tout simplement **Web** (mot anglais signifiant *toile*) la "toile virtuelle" formée par les différents documents (appelés pour l'occasion **pages web**) liés entre-eux par des hyperliens.

Les pages web sont généralement **organisées** autour d'une page d'accueil, jouant un point central dans la navigation des visiteurs parmi les pages HTML à l'aide des liens hypertextes. Cet ensemble cohérent de pages web liées par des liens hypertextes et articulées autour d'une page d'accueil commune est appelée **site web**.

Le Web est ainsi une énorme archive vivante composée d'une myriade de sites web proposant des pages web pouvant contenir du texte mis en forme, des images, des sons, des vidéo, etc.

Le principe du Web

Le web est composé de pages web stockées sur des serveurs web, c'est-à-dire des machines connectées à Internet en permanence et chargées de fournir les pages web demandées. Chacune des pages web, et plus généralement toute ressource en ligne (image, vidéo, musique, animation, etc.), est repérée par une adresse unique appelée URL.

L'élément clé pour la navigation au sein des pages web est le **navigateur** (en anglais **browser**, parfois également appelé *fureteur*, *fouineur*, *feuilleteur* ou *butineur* au Québec), c'est-à-dire le logiciel client capable d'interroger les serveurs web, d'exploiter leurs résultats et de mettre en page les informations grâce aux instructions contenues dans la page HTML.

Parmi les principaux navigateurs utilisés sur Internet, citons notamment les suivants :

- Mozilla Firefox,
- Microsoft Internet Explorer,
- Netscape Navigator,
- Safari.

Le HTML est un standard

Il est important de comprendre que le langage HTML est un standard, c'est-à-dire qu'il s'agit de recommandations publiées par un consortium international, : le World Wide Web Consortium (**W3C**).

Les spécifications officielles du HTML décrivent donc les "instructions" HTML mais en aucun cas leur implémentation, c'est-à-dire leur traduction en programmes d'ordinateur, afin de permettre la consultation de pages web indépendamment du système d'exploitation ou de l'architecture de l'ordinateur.

Toutefois, aussi étoffées les spécifications soient-elles, il existe toujours une marge d'interprétation de la part des navigateurs, ce qui explique qu'une même page web puisse s'afficher différemment d'un navigateur Internet à l'autre.

De plus, il arrive parfois que certains éditeurs de logiciels ajoutent des instructions HTML propriétaires, c'est-à-dire ne faisant pas partie des spécifications du W3C. Ainsi des pages web contenant ce type d'instruction pourront être parfaitement affichées sur un navigateur et totalement illisibles sur les autres, d'où la nécessité de créer des pages web respectant les recommandations du W3C afin de permettre leur consultation par le plus grand nombre.

Versions du langage HTML

Le langage HTML a été mis au point par *Tim Berners-Lee*, alors chercheur au CERN, à partir de 1989. Celui-ci annonça officiellement la création du web sur Usenet en août 1991. Ce n'est cependant qu'à partir de 1993 que l'on considère l'état du HTML suffisamment avancé pour parler de langage (HTML est alors baptisée symboliquement *HTML 1.0*). Le navigateur internet utilisé à l'époque était nommé *NCSA Mosaïc*.

Le RFC 1866, daté de novembre 1995 représente la première version officielle de HTML, c'est-à-dire le HTML 2.0.

Après la brêve apparition d'un HTML 3.0, qui ne vit jamais officiellement le jour, le HTML 3.2 devint le standard officiel le 14 janvier 1997. Les apports les plus marquants du HTML 3.2 étaient la standardisation des tableaux ainsi que d'un grand nombre d'éléments de présentation.

Le 18 décembre 1997, le HTML 4.0 a été publié. La version 4.0 du langage HTML standardise notamment les feuilles de style et les cadres (frames). La version HTML 4.01, apparue le 24 décembre 1999 apporte quelques modifications mineures au HTML 4.0.

Introduction aux balises HTML

Le HTML est un langage à balises

Le HTML n'est pas un langage de programmation, Une page HTML est donc un simple fichier texte contenant des **balises** (parfois appelées *marqueurs* ou *repères*) permettant de mettre en forme le texte, les images, etc.

Une balise est une commande (un nom) encadrée par le caractère inférieur (<) et le caractère supérieur (>) par exemple "*<H1>*".

Comment utiliser les balises HTML ?

Les balises HTML sont généralement présentes par paire afin d'agir sur les éléments qu'elles encadrent. La première est appelée "**balise d'ouverture**" (parfois *balise ouvrante*) et la seconde "**balise de fermeture**" (ou *fermante*). La balise fermante est précédé du caractère /) :

```
<marqueur> Votre texte formatté  </marqueur>
```

Ainsi les balises et permettent de mettre en gras le texte qu'elles encadrent :

```
<b> Ce texte est en gras </b>
```

Les balises HTML peuvent parfois être uniques : la balise <br> représente par exemple un retour à la ligne. Pour autant, afin d'être le plus proche possible du standard XHTML (beaucoup plus stricte que le standard HTML), il est conseillé d'utiliser la notation suivante : *
*.



Les balises ne sont pas sensibles à la casse, c'est-à-dire qu'elles peuvent être indifféremment en minuscules ou en majuscules !

Imbrication

Les balises HTML ont la particularité de pouvoir être imbriquées de manière hiérarchique afin de permettre le cumul de leur propriétés. En contrepartie le chevauchement de balises n'est pas toléré par le standard HTML.

```
<i><u>Comment ça Marche</u>, encyclopédie informatique libre</i>
```

L'exemple ci-dessus donne le résultat suivant :
Comment ça Marche, encyclopédie informatique libre
En contrepartie l'exemple ci-dessous n'est pas correct :
~~Comment ça Marche, encyclopédie informatique libre~~

Notion d'attribut

Un attribut est un élément, présent au sein de la balise ouvrante, permettant de définir des propriétés supplémentaires à la balise à laquelle il appartient. Les attributs se présentent la plupart du temps comme une paire clé-valeur, sous la forme *cle="valeur"*, mais certains attributs sont parfois une seule clé.

Voici un exemple d'attribut pour la balise <p> (balise définissant un paragraphe) permettant de spécifier que le texte doit être aligné sur la droite :

```
<p align="right">Exemple de paragraphe</p>
```

Chaque balise peut comporter un ou plusieurs attributs, chacun pouvant avoir (aucune,) une ou plusieurs valeurs.

Espaces, saut de ligne et tabulations

La langage HTML ne tient pas compte des espaces, des tabulations et des sauts de ligne ou plus exactement considère une suite d'un ou plusieurs espaces/tabulations/saut de ligne comme un seul espace. Cela est notamment très utile afin de permettre d'indenter le code HTML pour plus de lisibilité.

Les deux codes HTML suivants donnent ainsi le même résultat :

```
Exemple      de
code
                              HTML
Exemple de code HTML
```

Le langage HTML possède par contre des éléments permettant expressément de définir chacun de ces éléments de mise en forme :

- **Espace insécable** : il s'agit d'un espace ne pouvant être brisé par une fin de ligne. Sa représentation en HTML est * *.
- **Saut de ligne manuel** : il s'agit d'un saut de ligne explicite. Sa représentation en HTML est *
* (*
* pour être conforme au XHTML).



A noter tout de même : il existe une exception pour le code contenu dans des balises <PRE>, dont l'objectif est justement de conserver le formatage du texte (espaces, sauts de lignes, etc.) ! !

Commentaires

Il est possible d'ajouter des éléments d'information dans une page web sans que ceux-ci soient affichés à l'écran grâce à un jeu de balises spécifique, appelé *balises de commentaires*.

```
<!-- Voici un commentaire -->
```

Les balises de commentaires permettent de mettre en commentaire du texte mais peuvent également servir à commenter du code HTML.

Structure d’un document HTML

Notion de document HTML

Une page HTML est un simple fichier texte contenant des balises HTML. Par convention l'extension donnée à une page HTML est *.htm* ou *.html* mais une page web peut potentiellement porter n'importe quelle extension notamment les suivantes :

- **.asp** pour une page générée dynamiquement en ASP (*Active Server Pages*) ;
- **.cgi** pour une page générée dynamiquement avec des CGI (*Common gateway Interface*) ;
- **.php**, **.php3** ou **.php4** pour une page générée dynamiquement en PHP ;
- **.pl** pour une page générée dynamiquement en Perl (*Practical Extraction and Report Language*) ;
- etc.

La page HTML minimum

Une page HTML est un fichier texte commençant par la balise *<HTML>* et finissant par la balise *</HTML>*. Elle contient également un *en-tête* décrivant le titre de la page, puis un *corps* dans lequel se trouve le contenu de la page.

L'en-tête est délimité par les balises *<HEAD>* et *</HEAD>*.
Le corps est délimité par les balises *<BODY>* et *</BODY>*.

Ainsi la page HTML minimum peut être représentée comme suit :

```
<HTML>

        <HEAD>
                <TITLE> Le titre </TITLE>
        </HEAD>

    <BODY>

                Contenu de la page

        </BODY>

</HTML>
```

Déclaration du type de document

Il est nécessaire d'indiquer dans la page HTML le *prologue du type de document*, c'est-à-dire une référence à la norme HTML utilisée, afin de spécifier le standard utilisé pour le codage de la page en HTML. Cette déclaration se fait par une ligne du type :

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0//EN">
<HTML>
        <HEAD>...</HEAD>
        <BODY>Contenu de la page</BODY>
</HTML>
```

Éditeurs HTML

Une page web peut être construite à partir du plus basique des éditeurs de texte (une application *bloc-note* par exemple), mais il existe des éditeurs beaucoup plus évolués.

Les éditeurs *WYSIWYG* («*What You See Is What You Get*», littéralement «*ce que vous voyez est ce que vous obtenez*») sont des éditeurs graphiques permettant de travailler sur une page web telle qu'elle sera affichée sur un navigateur à quelques détails près. Grâce à ce genre d'éditeurs il est possible d'ajouter des balises par simple clic et d'en modifier les attributs en éditant leur propriétés dans un formulaire. Pour autant, afin d'utiliser au mieux ce genre d'éditeur, une connaissance préalable du HTML est tout de même très utile.

Il existe également des éditeurs permettant d'éditer le code HTML en affichant les balises, les attributs et leurs valeurs avec différentes couleurs pour une meilleure lecture et proposant parfois des outils pour vérifier la validité du code HTML.

Les caractères spéciaux

Codage des caractères spéciaux

Les normes HTML demandent de respecter le codage des caractères ASCII 7 bits, c'est-à-dire que les caractères accentués ne sont pas autorisés.

Pourtant, les navigateurs actuels reconnaissent les caractères accentués, ainsi vous pouvez entrer des caractères accentués directement sous votre éditeur de texte, mais votre page sera vraisemblablement illisible dans la plupart des pays du monde ...

Pour coder un caractère accentué, on entre une combinaison précédée du caractère & et terminée par un point-virgule (;). Voici la liste représentations HTML des caractères ASCII de 128 à 255:

Caractère	Code ISO	Code HTML
"	"	"
&	&	&
€	€	€
•		
‚	‚	
ƒ	ƒ	
„	„	
...	…	
†	†	
‡	‡	
ˆ	ˆ	
‰	‰	
Š	Š	
‹	‹	<
Œ	Œ	
•		
Ž	Ž	
•		
•		
‘	‘	
’	’	
“	“	
”	”	
•	•	
–	–	
—	—	
˜	˜	
™	™	
š	š	
›	›	>
œ	œ	œ
•		
ž	ž	
Ÿ	Ÿ	Ÿ
espace		
¡	¡	¡
¢	¢	¢
£	£	£

¤	¤	¤
¥	¥	&yen
¦	¦	¦
§	§	§
¨	¨	¨
©	©	©
ª	ª	ª
«	«	«
¬	¬	¬
®	®	®
¯	¯	&masr;
°	°	°
±	±	±
²	²	²
³	³	³
´	´	´
µ	µ	µ
¶	¶	¶
·	·	·
¸	¸	¸
¹	¹	¹
º	º	º
»	»	»
¼	¼	¼
½	½	½
¾	¾	¾
¿	¿	¿
À	**À**	**À**
Á	Á	Á
Â	Â	Â
Ã	Ã	Ã
Ä	Ä	Ä
Å	Å	Å
Æ	&#198	&Aelig
Ç	**Ç**	**Ç**
È	**È**	**È**
É	**É**	**É**
Ê	**Ê**	**Ê**
Ë	Ë	Ë
Ì	Ì	Ì
Í	Í	Í
Î	Î	Î
Ï	Ï	Ï
Ð	Ð	ð
Ñ	Ñ	Ñ
Ò	Ò	Ò
Ó	Ó	Ó
Ô	Ô	Ô
Õ	Õ	Õ
Ö	Ö	Ö
×	×	×
Ø	Ø	Ø
Ù	Ù	Ù
Ú	Ú	Ú

Û	Û	Û
Ü	Ü	Ü
Ý	Ý	Ý
Þ	Þ	þ
ß	ß	ß
à	**à**	**à**
á	á	á
â	**â**	**â**
ã	ã	ã
ä	ä	ä
å	å	å
æ	æ	æ
ç	**ç**	**ç**
è	**è**	**è**
é	**é**	**é**
ê	**ê**	**ê**
ë	**ë**	**ë**
ì	ì	ì
í	í	í
î	**î**	**î**
ï	ï	ï
ð	ð	ð
ñ	ñ	ñ
ò	ò	ò
ó	ó	ó
ô	**ô**	**ô**
õ	õ	õ
ö	ö	ö
÷	÷	÷
ø	ø	ø
ù	**ù**	**ù**
ú	ú	ú
û	**û**	**û**
ü	ü	ü
ý	ý	ý
þ	þ	þ
ÿ	ÿ	ÿ

Codage des signes HTML propres

Certains caractères ont une signification particulière en HTML, ayant une incidence forte sur le formatage de la page; il s'agit des caractères <, >, " et &. Le caractère < possède une importance particulière, dans la mesure où il représente pour le navigateur l'ouverture d'une nouvelle balise HTML.

Lorsque ces caractères sont utilisés dans un texte contenu dans une page web, il est donc impératif de les coder en HTML, au risque sinon de provoquer une erreur d'affichage dans le navigateur.

Caractère	Code ISO	Code HTML
"	"	"
&	&	&
‹	‹	<
›	›	>

Les balises de style

Qu'est-ce qu'une balise de style?

Les balises de style modifient la typographie du texte. Elles peuvent être imbriquées dans d'autres balises de style de la même façon qu'on le ferait avec un traitement de texte.
Voici une liste de balises de style reconnues par la plupart des navigateurs (chacun à sa façon si bien que certains ne font pas la différence entre ces styles).

Balise de style	Effet Visuel
<ABBREV>	Abréviation
<ACRONYM>	Acronyme
<AU>	L'auteur
<B>	**Met la police en gras**
<BIG>	Police plus grande
<BLINK>	Clignote (propre à Netscape)
<CITE>	*Citation*
<CODE>	`Instruction`
<DEL>	
<DFN>	*Définition d'instance*
<EM>	*Emphase*
<I>	*Italique*
<INS>	Nouveau texte inséré a cet endroit
<KBD>	`Clavier - Suite de caractères devant être tapés tel quel`
<PERSON>	Accentuation du nom d'une personne
<Q>	Encadre le texte par des guillemets
<S>	~~Comme strike (barré)~~
<SAMP>	`Exemple`
<SMALL>	Police plus petite
<STRONG>	**Forte accentuation rendue par du gras**
<STRIKE>	~~Texte barré (comme S)~~
<SUB>	Texte en $_{\text{Indice}}$
<SUP>	Texte en $^{\text{Exposant}}$
<TT>	`Caractère de machine à écrire`
<VAR>	*Nom d'une variable*

Niveaux de titres en HTML

Niveaux de titre

Le langage HTML définit 6 niveaux de titre (en anglais *heading*), afin de définir une structuration hiérarchique des paragraphes dans un texte :

Balise	Effet Visuel
H1	Test
H2	Test
H3	Test
H4	Test
H5	Test
H6	Test

Les attributs

Les attributs suivants peuvent être placés dans les balises de structure afin de permettre une disposition plus précise des éléments HTML :

Attribut	Valeur	Effet Visuel
ALIGN	LEFT	Texte aligné à gauche
	RIGHT	Texte aligné à droite
	CENTER	Texte centré
	JUSTIFY	Texte justifié
NOWRAP		Interdit au navigateur d'effectuer un saut de ligne.
ID		Attribue un identifiant à la balise. Cet attribut est notamment utile pour la programmation en JavaScript
LANG		Spécifie un langage différent
CLASS		Assigne une classe au contenu (pour les feuilles de style).
CLEAR		Utilisé dans le cas d'un texte entourant une image. Il permet de descendre dans la page aussi loin qu'il est nécessaire pour atteindre une marge libre

Voici un exemple d'en-tête :

```
<H1 ALIGN=JUSTIFY> Texte justifié </H1>
```

Paragraphes en HTML

Paragraphes

Le langage HTML considère les paragraphes comme des blocs de texte. Les navigateurs répartissent au mieux leur contenu dans la fenêtre à moins qu'=un attribut NOWRAP ou NOBR soit spécifié explicitement.
A l'intérieur d'un paragraphe, les espaces, tabulations et retours chariot comptent pour un seul espace.

La mise en page par blocs de texte est réalisée par l'intermédiaire de la paire de balises et . Cette balise accepte n'importe lequel des attributs vus précédemment.

Le retour chariot (retour à la ligne simple) est réalisé grâce à la balise

On peut aussi insérer une ligne horizontale grâce à la balise <hr>.

Balise	Attribut	Effet Visuel
 		retour à la ligne
et		paragraphe
<hr size=5 width=20% align=left>	SIZE WIDTH ALIGN	

Blocs d'éléments

Les balises servant à indenter le texte sont appelées des **conteneurs**.

Conteneur	Effet Visuel
<blockquote> et </blockquote>	Texte indenté
<address> et </address>	*Pour écrire une adresse*
<note> et </note>	Permet de rédiger une note
<fn> et </fn>	Permet de réaliser une note de fin de page
<banner> et </banner>	Fixe un bloc de texte par rapport à la page
<pre> et </pre>	Permet d'écrire un texte préformatté en conservant les espaces, les retours à la ligne et les tabulations.

Listes et énumérations en HTML

Les listes

Une **liste** est un paragraphe structuré contenant une suite d'articles. Le langage HTML définit trois types de listes :

- La liste ordonnée ;
- La liste non ordonnée ;
- La liste de définition.

Liste ordonnée

Conteneur	Type de liste	Effet Visuel
  - article 1 - article 2 	Liste ordonnée	1. article 1 2. article 2 3. article 3

Liste non ordonnée

Conteneur	Type de liste	Effet Visuel
  - article 1 - article 2 	Non ordonnée	• article 1 • article 2 • article 3

Liste de définition

Conteneur	Type de liste	Effet Visuel
<dl> <dt>Terme</dt> <dd>Définition</dd> </dl>	De définition	article 1 définition 1 article 2 définition 2

Attributs

Il existe des attributs spécifiques aux listes:

Attribut	Valeur	Effet Visuel
COMPACT		resserre l'interligne
PLAIN		supprime les puces
SEQNUM		définit le premier numéro
START		définit le premier numéro
CONTINUE		repart du numéro où il s'était arrété à la liste précédente
TYPE (pour les listes ordonnées)	1 A a I i	numérotation chiffrée (par défaut) numérotation en capitales numérotation en bas de casse numérotation en chiffres romains (I, II, III, IV ...) numérotation en chiffres romains en bas de casse
TYPE (pour les listes non-ordonnées)	circle square disc	o puce circulaire ▪ puce carrée • puce en disque

Les tableaux en HTML

Utilisation de tableaux

On a souvent besoin de présenter des informations mieux structurées qu'avec des listes. Les tableaux permettent de les afficher en lignes et en colonnes.

Les tableaux sont définis comme étant des suites de lignes.

Le tableau est encadré par les balises *<TABLE>* et *</TABLE>*.
Le titre du tableau est encadré par *<CAPTION> </CAPTION>*
Chaque ligne est encadrée par *<TR> </TR>* (*Table Row*, traduisez par *ligne du tableau*).
Les cellules d'en-tête sont encadrées par *<TH> </TH>* (pour *Table Header* : En-tête de tableau)
Les cellules de valeur sont encadrées par *<TD> </TD>* (Table Data: Donnée de tableau)

Par exemple le tableau:

```
<TABLE BORDER="1">
  <CAPTION> Voici le titre du tableau </CAPTION>
  <TR>
    <TH> Titre A1 </TH>
    <TH> Titre A2 </TH>
    <TH> Titre A3 </TH>
    <TH> Titre A4 </TH>
  </TR>
  <TR>
    <TH> Titre B1 </TH>
    <TD> Valeur B2 </TD>
    <TD> Valeur B3 </TD>
    <TD> Valeur B4 </TD>
  </TR>
</TABLE>
```

donne le résultat suivant:

Voici le titre du tableau			
Titre A1	**Titre A2**	**Titre A3**	**Titre A4**
Titre B1	Valeur B2	Valeur B3	Valeur B4

Les attributs

Attribut	Balises auxquelles il s'applique	Valeur	Effet Visuel
ALIGN	THEAD TBODY TH TR TD	CENTER LEFT RIGHT JUSTIFY	centré gauche droite justifié
	CAPTION	TOP BOTTOM	au-dessus en-dessous
VALIGN (alignement vertical)	THEAD TBODY TH TR TD	TOP MIDDLE BOTTOM	en haut au milieu en bas
BORDER=n	TABLE		taille de la bordure
CELLPADDING=n	TABLE		espacement de n pixels entre le contenu des cellules et son encadrement
CELLSPACING=n	TABLE		Epaisseur de la grille intérieure
FLOAT	TABLE	RIGHT LEFT	Spécifie la position du texte qui suivra </TABLE>
COLS=n	TABLE		Spécifie le nombre de colonnes
FRAME (contrôle les éléments individuels d'encadrement du tableau)	TABLE	NONE TOP BOTTOM TOPBOT SIDES ALL	aucun au-dessus en bas tout en haut sur les cotés tous
RULES (contrôle les éléments de la grille des cellules)	TABLE	NONE BASIC ROWS COLS ALL	aucun basique ligne colonne tous
COLSPAN	THEAD TBODY TH TR TD		Fait déborder les cellules sur les colonnes adjacentes
ROWSPAN	THEAD TBODY TH TR TD		Fait déborder les cellules sur les lignes adjacentes

Les liens hypertextes et ancrages

Présentation des ancrages

Les **ancrages** (*liens hypertextes*) sont des éléments d'une page HTML (soulignés lorsqu'il s'agit de texte) qui emmènent dans un autre endroit lorsqu'on clique dessus. C'est ce qui permet de lier des pages Web entre elles.
Les liens hypertextes permettent de naviguer:

- vers un autre endroit du document
- vers un fichier HTML situé à un emplacement différent sur la machine qui héberge la page
- vers une autre machine

L'attribut principal des ancrages est *href*. Il s'écrit sous la forme:

```
<a href="Adresse ou URL"> .. </a>
```

Lien externe

Il crée un lien vers une page dont on spécifie l'URL
Par exemple:

```
<a href="http://www.commentcamarche.net"> Comment ça marche? </a>
```

Lien local

On peut créer un lien vers une page située sur le même ordinateur en remplaçant l'URL par le fichier cible.

Ce lien peut être fait de façon relative, en repérant le fichier cible par rapport au fichier source. Si le fichier cible est "index.html" situé dans le répertoire parent, son lien s'écrira:

```
<a href="../index.html"> ... </a>
```

Ce lien peut aussi être fait de façon absolue, en écrivant l'adresse du fichier cible de façon locale:

```
<a href="file:///lecteur:/répertoire/index.html"> ... </a>
```

Les signets

On peut créer un signet dans une page c'est-à-dire marquer un endroit précis d'une page pour s'y rendre par hypertexte. Cela se fait grâce à l'attribut NAME ou ID (pour les browsers plus récents)
Par exemple:

```
<p id="signet"> ... </p>
```

On l'appellera grâce au lien suivant:

```
<a href="#signet"> ... </a>
```

On peut ainsi se déplacer à un endroit précis sur une autre page:

```
<a href="url/nom_du_fichier.html#signet"> ... </a>
```

Création des images en HTML

Comment afficher des images sur une page web?

Quelques images sur un site Web peuvent le rendre plus attractif et plus convivial, cependant il est important de ne pas sombrer dans l'excès car les images peuvent impliquer un temps de chargement assez long et peuvent dans certains cas nuire à la lisibilité.

La balise **IMG** du langage HTML permet d'insérer des images dans une page HTML. L'image peut être située sur le même serveur que la page dans laquelle elle est insérée mais également sur un autre serveur en spécifiant son URL complète.

Seuls les formats d'images suivants sont acceptés en standard dans les spécifications du W3C :

- Les images JPEG (.JPG) : les images ayant un grand nombre de couleurs seront bien compressées, c'est-à-dire qu'elles prendront moins de place, donc nécessiteront un temps de chargement moindre)
- Les images PNG : Leur taille est faible dans le cas d'images avec peu de couleurs avec des tons uniformes, ce format permet en outre d'avoir des images entrelacées (qui s'affichent progressivement) avec une profondeur de couleurs de 24 bits et des images dont on définit une couleur comme transparente.
- Les images GIF : Elles possèdent les mêmes atouts que les images PNG, si ce n'est que le format GIF est limité à 256 couleurs maximum et que ce format n'est pas totalement libre.

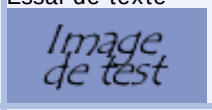
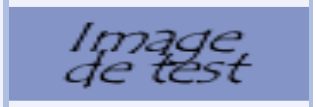
Comment afficher des images sur une page web?

Les principaux attributs de la balise *IMG* sont les suivants :

- **SRC**: Indique l'emplacement de l'image (il est obligatoire)
- **ALIGN**: Spécifie l'alignement de l'image par rapport au texte adjacent. Il peut prendre les valeurs: TOP, MIDDLE, et BOTTOM (au-dessus, au milieu et en-dessous)
- **ALT**: Permet d'afficher un texte alternatif lorsque l'image ne s'affiche pas.
- **TITLE**: Permet d'afficher une infobulle lors du survol de l'image par le curseur.
- **WIDTH**: Permet de spécifier la largeur de l'image.
- **HEIGHT**: Permet de spécifier la hauteur de l'image.

Ainsi pour insérer une image, il faudra saisir une balise du type suivant :

```
<IMG SRC="url_de_l_image"
    ALT="Texte remplaçant l'image"
    TITLE="Texte à afficher">
```

Attribut	Valeur	Résultat	Effet Visuel
ALIGN	bottom center left middle top right	Alignement de l'image.	
ALT		Texte alternatif au cas où l'image ne s'affiche pas.	
BORDER	Nombre entier	Nombre de pixels de la bordure. La couleur de la bordure peut être définie par l'attribut LINK ou TEXT de la balise <BODY>. Par défaut l'attribut *BORDER* vaut 1 ce qui crée un petit cadre autour de l'image. Pour ne pas avoir ce désagrément pensez à le définir comme étant égal à 0.	
HEIGHT	Nombre entier	Hauteur de l'image (en pixels ou en %). Lorsque cet attribut précise une dimension différente de la dimension originale du graphique, le navigateur la redimensionne dynamiquement, ce qui peut provoquer un affichage de moins bonne qualité.	
HSPACE	Nombre entier	Nombre de pixels d'ajustement entre l'image et le texte adjacent (horizontalement).	 Texte
LONGDESC		URL de la description de l'image.	
LOWSRC	URL	Image alternative (généralemnt plus petite) affichée le temps que la vraie image soit chargée par le navigateur.	
NAME		Permet de définir un nom pour l'image. Cet attribut est notamment utile pour la gestion des images en JavaScript.	
SRC	URL	URL de l'image	
TITLE		Texte alternatif au cas où l'image ne s'affiche pas.	
USEMAP		URL ou nom de l'ancre définissant l'image réactive.	
VSPACE		Nombre de pixels d'ajustement entre l'image et le texte (verticalement).	Essai de texte 
WIDTH	Nombre entier	Largeur de l'image (en pixels ou en %). Lorsque cet attribut précise une dimension différente de la dimension originale du graphique, le navigateur la redimensionne dynamiquement, ce qui peut provoquer un affichage de moins bonne qualité.	

Texte enveloppant une image

Les images s'insèrent dans le texte comme un caractère, ainsi il semble impossible de faire s'écouler du texte le long d'une image.

Il existe en fait plusieurs façons, nous allons en voir deux:

- La première consiste à créer un tableau avec une ligne et deux colonnes, dans lesquelles on met l'image et le texte.
- La seconde (moins précise) consiste à aligner l'image à gauche ou à droite avec l'attribut ALIGN puis de taper son texte. Il suffit, pour arrêter cet enveloppement de mettre l'attribut CLEAR.

Les images réactives ("images MAP")

Il est possible de créer des zones cliquables à l'intérieur même d'une image grâce à l'attribut *USEMAP* utilisé conjointement avec la balise *MAP*.
L'attribut *USEMAP* de la balise <*IMG*> pointe vers une balise <*MAP*> contenant la description du découpage de l'image en zones cliquables.

La balise <MAP> a un attribut *NAME* définissant son nom (*NAME="nom"*) et contient les zones cliquables déclarées grâce à des balises *AREA*.

Balise	Attribut	Valeur	Effet Visuel
MAP	NAME		
AREA	SHAPE	RECT	Rectangle (ses coordonnées sont: "abscisse sup gauche, ordonnée sup gauche, abscisse inf droit, ordonnée inf droit")
		CIRCLE	Cercle (ses coordonnées sont: "abscisse centre, ordonnée centre, rayon")
		POLY	Polygone (ses coordonnées sont: "la suite des coordonnées séparées par des virgules")
	HREF	URL	Lien vers l'URL
	COORDS	"XX,XX,XX,XX"	Contient les coordonnées de la zone cliquable, séparées par des virgules.

Voici ci-dessous un exemple d'image réactive :

```
<IMG SRC="images/image.gif"
    WIDTH=150
    HEIGHT=70
    USEMAP="#Map">
<MAP NAME="Map">
    <AREA SHAPE="rect"
           HREF="debut.html"
           COORDS="0,0,48,28">

    <AREA SHAPE="circle"
           HREF="precedent.html"
           COORDS="50,30,10">

    <AREA SHAPE="poly"
           HREF="suivant.html"
           COORDS="60,50,80,30,100,40,50,100">
</MAP>
```

Les images d’arrière-plan

Insérer une image en fond

L'arrière-plan est défini dans les attributs de la balise <BODY> :

Attribut	Effet Visuel
BACKGROUND="image"	affiche l'image en arrière-plan
BGCOLOR="nom_de_la_couleur ou #XXXXXX"	affiche la couleur demandée en arrière-plan
LINK="couleur"	Couleur des liens hypertexte
ALINK="couleur"	Couleur du lien actif
VLINK="couleur"	Couleur des liens déjà visités
TEXT="couleur"	Couleur du texte par défaut

Les couleurs sont définies par 6 chiffres hexadécimaux représentant 2 à 2 les tons de Rouge, de Vert et de Bleu.
couleur="#RRVVBB" où R, V et B représentent respectivement un chiffre hexadécimal entre 0 et F pour le Rouge, le Vert et le Bleu.

Voir la table des couleurs

Les couleurs en HTML

Le codage des couleurs

Les couleurs en HTML sont définies par 3 nombres hexadécimaux représentant les tons de Rouge, de Vert et de Bleu (selon le codage RGB (*Red Green Blue*, en français : **RVB**) de la couleur choisie. Ainsi la syntaxe de codage d'une couleur en HTML est la suivante :

```
couleur="#RRVVBB"
```

RR, VV et BB représentent respectivement un nombre hexadécimal entre 00 et FF pour le Rouge, le Vert et le Bleu.

Ainsi, plus de 16 millions de couleurs sont disponibles pour colorer les pages web. Toutefois, étant donné que tous les navigateurs ne reconnaissent pas les couleurs de la même façon, le W3C conseille l'utilisation des couleurs ci-dessous, pour lesquelles un nom intelligible a été donné. Il est donc possible (et conseillé) d'appeler une couleur de la façon suivante :

```
couleur="nom_de_la_couleur"
```

Nom de la couleur		Codage RVB	Aperçu
aliceblue	aliceblue	#F0F8FF	
antiquewhite	antiquewhite	#FAEBD7	
aqua	aqua	#00FFFF	
aquamarine	aquamarine	#7FFFD4	
azure	azure	#F0FFFF	
beige	beige	#F5F5DC	
bisque	bisque	#FFE4C4	
black	black	#000000	
blanchedalmond	blanchedalmond	#FFEBCD	
blue	blue	#0000FF	
blueviolet	blueviolet	#8A2BE2	
brown	brown	#A52A2A	
burlywood	burlywood	#DEB887	
cadetblue	cadetblue	#5F9EA0	
chartreuse	chartreuse	#7FFF00	
chocolate	chocolate	#D2691E	
coral	coral	#FF7F50	
cornflowerblue	cornflowerblue	#6495ED	
cornsilk	cornsilk	#FFF8DC	
crimson	crimson	#DC143C	
cyan	cyan	#00FFFF	
darkblue	darkblue	#00008B	
darkcyan	darkcyan	#008B8B	
darkgoldenrod	darkgoldenrod	#B8860B	
darkgray	darkgray	#A9A9A9	
darkgreen	darkgreen	#006400	
darkkhaki	darkkhaki	#BDB76B	
darkmagenta	darkmagenta	#8B008B	
darkolivegreen	darkolivegreen	#556B2F	
darkorange	darkorange	#FF8C00	
darkorchid	darkorchid	#9932CC	

darkred	darkred	#8B0000	
darksalmon	darksalmon	#E9967A	
darkseagreen	darkseagreen	#8FBC8F	
darkslateblue	darkslateblue	#483D8B	
darkslategray	darkslategray	#2F4F4F	
darkturquoise	darkturquoise	#00CED1	
darkviolet	darkviolet	#9400D3	
deeppink	deeppink	#FF1493	
deepskyblue	deepskyblue	#00BFFF	
dimgray	dimgray	#696969	
dodgerblue	dodgerblue	#1E90FF	
firebrick	firebrick	#B22222	
floralwhite	floralwhite	#FFFAF0	
forestgreen	forestgreen	#228B22	
fuchsia	fuchsia	#FF00FF	
gainsboro	gainsboro	#DCDCDC	
ghostwhite	ghostwhite	#F8F8FF	
gold	gold	#FFD700	
goldenrod	goldenrod	#DAA520	
gray	gray	#808080	
green	green	#008000	
greenyellow	greenyellow	#ADFF2F	
honeydew	honeydew	#F0FFF0	
hotpink	hotpink	#FF69B4	
indianred	indianred	#CD5C5C	
indigo	indigo	#4B0082	
ivory	ivory	#FFFFF0	
khaki	khaki	#F0E68C	
lavender	lavender	#E6E6FA	
lavenderblush	lavenderblush	#FFF0F5	
lawngreen	lawngreen	#7CFC00	
lemonchiffon	lemonchiffon	#FFFACD	
lightblue	lightblue	#ADD8E6	
lightcoral	lightcoral	#F08080	
lightcyan	lightcyan	#E0FFFF	
lightgoldenrodyellow	lightgoldenrodyellow	#FAFAD2	
lightgreen	lightgreen	#90EE90	
lightgrey	lightgrey	#D3D3D3	
lightpink	lightpink	#FFB6C1	
lightsalmon	lightsalmon	#FFA07A	
lightseagreen	lightseagreen	#20B2AA	
lightskyblue	lightskyblue	#87CEFA	
lightslategray	lightslategray	#778899	
lightsteelblue	lightsteelblue	#B0C4DE	
lightyellow	lightyellow	#FFFFE0	
lime	lime	#00FF00	
limegreen	limegreen	#32CD32	
linen	linen	#FAF0E6	
magenta	magenta	#FF00FF	
maroon	maroon	#800000	
mediumaquamarine	mediumaquamarine	#66CDAA	
mediumblue	mediumblue	#0000CD	
mediumorchid	mediumorchid	#BA55D3	
mediumpurple	mediumpurple	#9370DB	
mediumseagreen	mediumseagreen	#3CB371	

mediumslateblue	mediumslateblue	#7B68EE	
mediumspringgreen	mediumspringgreen	#00FA9A	
mediumturquoise	mediumturquoise	#48D1CC	
mediumvioletred	mediumvioletred	#C71585	
midnightblue	midnightblue	#191970	
mintcream	mintcream	#F5FFFA	
mistyrose	mistyrose	#FFE4E1	
moccasin	moccasin	#FFE4B5	
navajowhite	navajowhite	#FFDEAD	
navy	navy	#000080	
oldlace	oldlace	#FDF5E6	
olive	olive	#808000	
olivedrab	olivedrab	#6B8E23	
orange	orange	#FFA500	
orangered	orangered	#FF4500	
orchid	orchid	#DA70D6	
palegoldenrod	palegoldenrod	#EEE8AA	
palegreen	palegreen	#98FB98	
paleturquoise	paleturquoise	#AFEEEE	
palevioletred	palevioletred	#DB7093	
papayawhip	papayawhip	#FFEFD5	
peachpuff	peachpuff	#FFDAB9	
peru	peru	#CD853F	
pink	pink	#FFC0CB	
plum	plum	#DDA0DD	
powderblue	powderblue	#B0E0E6	
purple	purple	#800080	
red	red	#FF0000	
rosybrown	rosybrown	#BC8F8F	
royalblue	royalblue	#4169E1	
saddlebrown	saddlebrown	#8B4513	
salmon	salmon	#FA8072	
sandybrown	sandybrown	#F4A460	
seagreen	seagreen	#2E8B57	
seashell	seashell	#FFF5EE	
sienna	sienna	#A0522D	
silver	silver	#C0C0C0	
skyblue	skyblue	#87CEEB	
slateblue	slateblue	#6A5ACD	
slategray	slategray	#708090	
snow	snow	#FFFAFA	
springgreen	springgreen	#00FF7F	
steelblue	steelblue	#4682B4	
tan	tan	#D2B48C	
teal	teal	#008080	
thistle	thistle	#D8BFD8	
tomato	tomato	#FF6347	
turquoise	turquoise	#40E0D0	
violet	violet	#EE82EE	
wheat	wheat	#F5DEB3	
white	white	#FFFFFF	
whitesmoke	whitesmoke	#F5F5F5	
yellow	yellow	#FFFF00	
yellowgreen	yellowgreen	#9ACD32	

Les formulaires en HTML

Intérêt d'un formulaire

Les formulaires interactifs permettent aux auteurs de pages Web de dialoguer avec leurs lecteurs, un peu comme les coupons-réponse que l'on trouve dans les magazines.

Le lecteur saisit des informations en remplissant des champs ou en cliquant sur des boutons, puis appuie sur un bouton de soumission (submit) pour l'envoyer soit à un URL, c'est-à-dire de façon générale à une adresse e-mail ou à un script CGI (Common Gateway Interface, traduisez "Interface de passerelle généralisée") stocké sur un serveur et écrit dans un langage de programmation comme un shell UNIX, PERL, TCL, Applescript, etc.

La balise FORM

Les formulaires sont délimités par la balise **<FORM> ... </FORM>**, une balise qui permet de regrouper plusieurs éléments de formulaire (boutons,champs de saisie,...) et qui possède les attributs obligatoires suivants:

- **METHOD** indique sous quelle forme seront envoyées les réponses
  "POST" est la valeur qui correspond à un envoi de données stockées dans le corps de la requête, tandis que "GET" correspond à un envoi des données codées dans l'URL, et séparées de l'adresse du script par un point d'interrogation (pour plus de renseignement sur les méthodes POST et GET, consultez l'article sur le protocole HTTP)
- **ACTION** indique l'adresse d'envoi (script CGI ou adresse email (mailto:adresse.email@machine))

La balise FORM possède comme attribut facultatif *ENCTYPE* qui spécifie le codage des données dans l'URL, toutefois il n'est pas nécessaire de le préciser car la valeur attribuée par défaut (application/x-www-form-urlencoded) est la seule valeur valide. L'attribut facultatif *ACCEPT* permet de définir les types MIME des données pouvant être envoyées par le formulaire.

Voici la syntaxe de la balise FORM:

```
<FORM METHOD="POST" ou "GET" ACTION="url" ENCTYPE="x-www-form-urlencoded">
...
</FORM>
```

Voici quelques exemples de balises FORM:

```
<FORM METHOD=POST ACTION="mailto:webmaster@commentcamarche.net">
<FORM METHOD=GET ACTION="http://www.commentcamarche.net/cgi-bin/script.cgi">
```

A l'intérieur de la balise FORM...

La balise *FORM* constitue en quelque sorte un conteneur permettant de regrouper des éléments qui vont permettre à l'utilisateur de choisir ou de saisir des données, ensemble de données qui seront envoyées à l'URL indiqué dans l'attribut *ACTION* de la balise *FORM* par la méthode indiquée par l'attribut *METHOD*.

Il est possible d'insérer n'importe quel élément HTML de base dans une balise *FORM* (textes,boutons,tableaux,liens,...) mais il est surtout intéressant d'insérer des éléments interactifs. Ces éléments interactifs sont:

- La balise *INPUT*: un ensemble de boutons et de champs de saisie
- La balise *TEXTAREA*: une zone de saisie
- La balise *SELECT*: une liste à choix multiples

Envoi des données

Lorsqu'un formulaire est soumis (appui sur le bouton de soumission), les données présentes dans le formulaire sont envoyées au script CGI sous forme de paires nom/valeur, c'est-à-dire un ensemble de données représentées par le nom de l'élément de formulaire, le caractère "=", puis la valeur associée. L'ensemble de ces paires nom/valeur étant séparées entre elles par des esperluettes (caractère &). Les données envoyées ressembleront donc à ceci:

```
champ1=valeur1&champ2=valeur2&champ3=valeur3
```

Dans le cas de la méthode GET (envoi des données à travers l'URL), l'URL ressemblera à une chaîne du genre:

```
http://www.commentcamarche.net/cgi-bin/script.cgi?champ1=val1&champ2=val2
```

La balise INPUT

La balise *INPUT* est la balise essentielle des formulaires, car elle permet de créer un bon nombre d'éléments "interactifs". La syntaxe de cette balise est la suivante:

```
<INPUT type="Nom du champ" value="Valeur par défaut" name="Nom de l'élément">
```

L'attribut *name* est essentiel car il permettra au script CGI de connaître le champ associé à la paire nom/valeur, c'est-à-dire que le nom du champ sera suivi du caractère "=" puis de la valeur entrée par l'utilisateur, ou dans le cas contraire de la valeur par défaut repéré par l'attribut *value*.
L'attribut *type* permet de préciser le type d'élément que représente la balise *INPUT*, voici les valeurs que ce champ peut prendre:

- **checkbox**: il s'agit de *cases à cocher* pouvant admettre deux états: *checked* (coché) et *unchecked* (non coché). Lorsque la case est coché la paire nom/valeur est envoyée au CGI
- **hidden**: il s'agit d'un *champ caché*. Ce champ non visible sur le formulaire permet de préciser un paramètre fixe qui sera envoyé au CGI sous forme de paire nom/valeur
- **file**: il s'agit d'un *champ* permettant à l'utilisateur de préciser l'emplacement d'un fichier qui sera envoyé avec le formulaire. Il faut dans ce cas préciser le type de données pouvant être envoyées grâce à l'attribut *ACCEPT* de la balise *FORM*
- **image**: il s'agit d'un *bouton de soumission personnalisé*, dont l'apparence est l'image situé à l'emplacement précisé par son attribut *SRC*
- **password**: il s'agit d'un *champ de saisie*, dans lequel les caractères saisis apparaissent sous forme d'astérisques afin de camoufler la saisie de l'utilisateur
- **radio**: il s'agit d'un *bouton* permettant un choix parmi plusieurs proposés (l'ensemble des boutons radios devant porter le même attribut *name*. La paire nom/valeur du bouton radio sélectionné sera envoyé au CGI. Un attribut *checked* pour un des boutons permet de préciser le bouton sélectionné par défaut
- **reset**: il s'agit d'un *bouton de remise à zéro* permettant uniquement de rétablir l'ensemble des éléments du formulaire à leurs valeurs par défaut
- **submit**: il s'agit du *bouton de soumission* permettant l'envoi du formulaire. Le texte du bouton peut être précisé grâce à l'attribut *value*
- **text**: il s'agit d'un *champ de saisie* permettant la saisie d'une ligne de texte. La taille du champ peut être définie à l'aide de l'attribut *size* et la taille maximale du texte saisi grâce à l'attribut *maxlength*

La balise TEXTAREA

La balise *TEXTAREA* permet de définir une zone de saisie plus vaste par rapport à la simple ligne de saisie que propose la balise *INPUT*. Cette balise possède les attributs suivants:

- **cols**: représente le nombre de caractères que peut contenir une ligne
- **rows**: représente le nombre de lignes
- **name**: représente le nom associé au champ, c'est le nom qui permettra d'identifier le champ dans la paire nom/valeur
- **readonly**: permet d'empêcher l'utilisateur de modifier le texte entré par défaut dans le champ
- **value**: représente la valeur qui sera envoyée par défaut au script si le champ de saisie n'est pas modifié par une frappe de l'utilisateur

La balise SELECT

La balise *SELECT* permet de créer une liste déroulante d'éléments (précisés par des balises *OPTION* à l'intérieur de celle-ci). Les attributs de cette balise sont:

- **name**: représente le nom associé au champ, c'est le nom qui permettra d'identifier le champ dans la paire nom/valeur
- **disabled**: permet de créer une liste désactivée, c'est-à-dire affichée en grisée
- **size**: représente le nombre de lignes dans la liste (cette valeur peut être plus grande que le nombre d'éléments effectifs dans la liste)
- **multiple**: marque la possibilité pour l'utilisateur de choisir plusieurs champs dans la liste

Un exemple de formulaire

Les formulaires peuvent être mis en page à l'aide de tableaux (cela est même conseillé pour avoir une mise en page soignée). Voici un exemple récapitulant les points ci-dessus et montrant comment mettre en page un formulaire à l'aide d'un tableau:

```
<FORM method=post action="cgi-bin/script.pl">

Enregistrement d'un utilisateur

<TABLE BORDER=0>

<TR>

        <TD>Nom</TD>

        <TD>

        <INPUT type=text name="nom">

        </TD>

</TR>

<TR>

        <TD>Prénom</TD>

        <TD>

        <INPUT type=text name="prenom">

        </TD>

</TR>

<TR>

        <TD>Sexe</TD>

        <TD>

        Homme : <INPUT type=radio name="sexe" value="M">
```

```
    <br>Femme : <INPUT type=radio name="sexe" value="F">
        </TD>
</TR>
<TR>
        <TD>Fonction</TD>
        <TD>
        <SELECT name="fonction">
                <OPTION VALUE="enseignant">Enseignant</OPTION>
                <OPTION VALUE="etudiant">Etudiant</OPTION>
                <OPTION VALUE="ingenieur">Ingénieur</OPTION>
                <OPTION VALUE="retraite">Retraité</OPTION>
                <OPTION VALUE="autre">Autre</OPTION>
        </SELECT>
        </TD>
</TR>
<TR>
        <TD>Commentaires</TD>
        <TD>
        <TEXTAREA rows="3" name="commentaires">
        Tapez ici vos commentaires</TEXTAREA>
        </TD>
</TR>
<TR>
        <TD COLSPAN=2>
        <INPUT type="submit" value="Envoyer">
        </TD>
</TR>
</TABLE>
</FORM>
```

Voici le résultat de ce code HTML

Enregistrement d'un utilisateur

Nom

Prénom

Sexe Homme :

Femme :

Fonction Enseignant

Commentaires Tapez ici vos commentaires

Envoyer

Attributs de la balise FORM et types d'entrées

Balise	Attribut	Valeur	Résultat	Effet Visuel
<FORM> ... </FORM>	METHOD	POST GET		
	ACTION		envoie à l'adresse indiquée	
	ENCTYPE		spécifie le type de codage utilisé	
<INPUT>	TYPE	submit	effectue l'ACTION dans le balise <FORM>	Envoyer
		text	simple ligne de texte dont la longueur est donnée par l'attribut size	
		reset	efface le contenu du formulaire	Rétablir
		radio	bouton radio	
		checkbox	case à cocher	
	NAME		Nom	
	SIZE		Taille du texte	
<TEXTAREA> ... </TEXTAREA>	NAME		Zone de texte	
	ROWS			
	COLS			
<SELECT> <OPTION> ... </OPTION> </SELECT>	NAME			Choix 1
	MULTIPLE		Plusieurs choix possibles	Choix 1
<OPTION> ... </OPTION>	SELECTED	Option par défaut		

	VALUE	Valeur forcée		Choix 1 Choix 2 Choix 3

Les méta-tags

Qu'appelle-t-on "Méta données" ?

Les méta-données sont des informations situées au sein d'un document afin de le décrire.

On appelle ainsi «**méta-tags**» (en français «*méta-balises*») des balises spéciales situées dans l'en-tête du document HTML (c'est-à-dire la balise *HEAD*, située avant la balise *BODY*), afin notamment de fournir des informations permettant aux moteurs de recherche d'indexer la page web.

Il est possible de renseigner plusieurs métas les uns après les autres dans l'en-tête de la page, en utilisant la notation suivante:

```
<META HTTP-EQUIV="Nom du tag" CONTENT="Attribut">
```

Nom du tag	Attribut	Utilité
Pragma	no-cache	Empêche le navigateur de garder la page dans son cache
Refresh	x; URL="adresse"	Appelle une autre page (située à l'URL spécifié) après un délai d'attente de x secondes.
Robots	all follow index nofollow noindex none	Permet au robot de référencer toute la page (par défaut) Permet au robot de suivre les liens de la page Permet au robot d'indexer la page Interdit au robot de suivre les liens Interdit au robot d'indexer la page Empêche le robot d'aller plus loin
Window-target	_blank _top	Force le browser à afficher la page dans une nouvelle fenêtre Affiche la page dans la fenêtre entière (pas dans une frame)

Il en existe d'autres s'écrivant de la manière suivante:

```
<META NAME="Nom du tag" CONTENT="Attribut">
```

Nom du tag	Attribut	Utilité
Author	"Informations sur l'auteur"	Informations sur l'auteur de la page
Copyright	"Informations de copyright"	Permet de référencer des informations de droits d'auteur
Description	"Description de votre site"	Permet de donner une information à afficher lors du résultat d'une recherche.
Expires	never "date à laquelle la page expire"	Indique au robot la date d'expiration de la page.
Generator	Nom du logiciel	Nom de l'éditeur HTML ayant généré la page web.
Keywords	"mot clé, mot clé, ..."	Mots clés décrivant la page Web.
Rating	general	Type de contenu (public visé).
Rev	Votre@email	Mél du Webmestre.
Revisit-after	"x days" ou x représente le nombre de jours	Délai de visite par le robot
Robots	all none	Permet à tous les robots d'indexer Ne permet à aucun robot d'indexer
Subject	"Sujet de la page"	Permet de préciser le sujet de la page

Les frames (cadres)

Présentation des frames

Grâce à la technologie des frames (en français "cadres") il est désormais possible d'afficher plusieurs pages HTML dans différentes zones (ou *cadres*).



Les frames ne font pas partie de la spécification du HTML 3.x, il faut donc déclarer sa page comme étant écrite en HTML 4.0 !

Créer des frames

Afin de créer un site contenant des cadres, il suffit de créer un fichier contenant l'agencement des cadres : ce fichier HTML a pour particularité d'avoir un conteneur <FRAMESET> à la place du jeu de balises <*BODY*>. C'est cette balise qui définit les cadres par leur dimension en pixels ou en pourcentage (%).

Voyons ceci sur 3 exemples:

- 2 cadres verticaux
- 2 cadres horizontaux
- 2 cadres horizontaux et un vertical

Exemple n°1

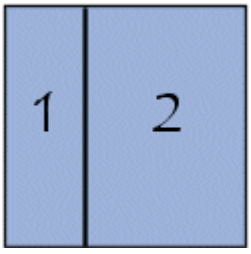
```
<FRAMESET COLS="20%,

80%">

<FRAME SRC="frame1.htm" NAME="gauche">

<FRAME SRC="frame2.htm" NAME="droite">

</FRAMESET>
```



Exemple n°2

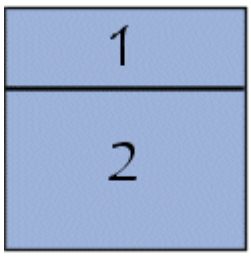
```
<FRAMESET ROWS="20%,

80%">


<FRAME SRC="frame1.htm" NAME="haut">


<FRAME SRC="frame2.htm" NAME="bas">


</FRAMESET>
```



Exemple n°3

```
<FRAMESET COLS="20%,

80%">


<FRAME SRC="frame1.htm" NAME="gauche">


<FRAMESET ROWS="50%, 50%">

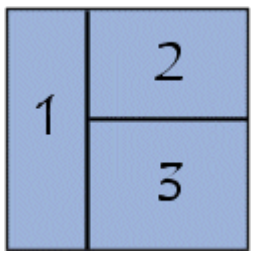

<FRAME SRC="frame2.htm"

NAME="droit_haut">


<FRAME SRC="frame3.htm"

NAME="droit_bas">


</FRAMESET>
```



Les attributs de la balise <FRAMESET>

Attribut	Valeur	Action
Rows	pourcentage (entre 1 et 100) valeur en pixels En fixant une seules des valeurs et en donnant à l'autre la valeur *, la valeur s'ajuste automatiquement	Cadre horizontal
Cols	pourcentage (entre 1 et 100) valeur en pixels En fixant une seules des valeurs et en donnant à l'autre la valeur *, la valeur s'ajuste automatiquement	Cadre vertical
Frameborder	YES NO	Indique si le cadre a une bordure ou non
Border=n	n est une valeur définissant la taille de la bordure	Indique la taille de la bordure
Bordercolor	Nom de la couleur Valeur de la couleur en hexadécimal	Indique la couleur de la bordure
Framespacing=n	n est une valeur définissant l'espace entre les cadres	Indique l'espace entre les cadres

Les attributs de la balise <FRAME>

La balise <FRAME> permet de définir un ou plusieurs cadres au sein de la balise <FRAMESET>

Attribut	Valeur	Action
Src	URL	Définit l'emplacement de la page à afficher dans le cadre
Name	"nom"	Définit un nom qui permettra d'afficher un autre document dans le cadre gâce à l'attribut **Target**
Marginwidth=n	n est un entier spécifiant le nombre de pixels	Taille des marges latérales
Marginheight=n	n est un entier spécifiant le nombre de pixels	Taille des marges du haut et du bas
Frameborder	YES NO	Détermine si les cadres auront ou non une bordure
Border=n	n est un entier spécifiant le nombre de pixels	Taille de l'espace entre les cadres (uniquement pour Netscape)
Noresize	(Aucune)	Interdit à l'utilisateur de redimensionner les cadres (Ce n'est pas la valeur par défaut)
Scrolling	***YES*** ***NO*** ***AUTO***	Permet ou non l'affichage d'une barre de défilement (***Auto*** laisse le navigateur décider de son utilité)

Comment éviter les erreurs avec les navigateurs non compatibles?

On utilisera les balises <NOFRAMES> et </NOFRAMES> permettant de spécifier un texte HTML à afficher en cas de navigateur ne permettant pas d'afficher les frames. Le texte entre les balises <NOFRAMES> et </NOFRAMES> doit donc contenir les balises <BODY> ... </BODY>.

Exemple :

```
<FRAMESET COLS="20%,

80%">


<FRAME SRC="frame1.htm" NAME="gauche">


<FRAME SRC="frame2.htm" NAME="droite">


</FRAMESET>
<NOFRAMES>

<BODY>

Cette page HTML nécessite un navigateur supportant les frames,

veuillez nous en excuser.

</BODY>

</NOFRAMES>
```

Désigner un cadre avec un lien hypertexte

Pour faire afficher des liens dans un des cadres, il suffit d'utiliser l'attribut target dans la balise <A HREF ..> pour spécifier le nom du cadre qui a été spécifié dans la balise <FRAME> par l'attribut NAME.

Par exemple:

```
<A HREF="page.htm" TARGET="gauche">
```

Valeur	Action
_self	Affiche la cible dans le même cadre que le lien
_parent	Affiche la cible dans le cadre de niveau supérieur
_blank	Affiche la cible dans une nouvelle fenêtre
_top	Affiche la cible dans la fenêtre entière du navigateur