

Langage HTML

- **HTML** (*HyperText Markup Language*).
- C'est un langage qui permet de définir l'habillage d'un document.
- C'est à dire : la façon dont il doit s'afficher à l'écran d'un navigateur.
- Le HTML n'est pas un langage de programmation.

Fichier HTML :

- Une page HTML est un fichier texte enrichi de codes et de commandes appelées balises.
- Le fichier source porte généralement l'extension :
 - .htm ou .html
- On peut écrire du code HTML à l'aide de n'importe quel éditeur de texte.

Code HTML :

- Il existe des éditeurs WYSIWYG : What You See Is What You Get (Ce que vous voyez est ce que vous obtenez)
- Le code source doit être sauvegardé sur le serveur Web.
- Il est envoyé vers les clients à l'aide du protocole HTTP

Source HTML :

➤ Le code HTML est interprété par le navigateur et l'affiche sur écran sous forme de :

- textes mis en forme,
- images,
- animations,
- multimédias, ...

➤ Le source est visible aux utilisateurs mais ils ne pourront pas le modifier (facilement) sur le serveur.

Notion de Balise :

➤ C'est un mot clé encadré par < et >

➤ Pour la plupart des balises, il existe une balise de début et une balise de fermeture précédée par /

<HTML> </HTML>

➤ Certaines balises n'ont pas de balises fermantes :

**
**

➤ Pour d'autres, la balise fermante est facultative :

<P>, <TD> ...

Attribut de balise:

- Pour la majorité des balises, il existe un ou plusieurs attributs.
- Les attributs sont des mots réservés.
- A chaque attribut, on affecte une valeur comme suit:

```
<Nom_Balise att_1="val_1" att_2="val_2" ...  
             att_n="val_n">
```



Remarques :

- Certains attributs (cas rares) n'ont pas de valeurs.

<HR noshade>

- Chaque balise a ses propres attributs.
- Certains attributs sont communs entre plusieurs balises : **Color**, **Align**, ...
- Mais aussi, certaines balises n'ont pas d'attributs.



Valeurs des attributs :

- Certaines valeurs doivent appartenir à un ensemble fini :

`align="left"`

- Certaines valeurs doivent obéir à une syntaxe bien précise :

`color = "#FF0080"`

- D'autres sont libres :

`name = "Master IL"`

Valeurs particulières : Les couleurs

➤ Par nom : `white black blue green red ...`

➤ Par code RGB : `#RRGGBB`.

- RR, GG ou BB est une valeur hexadécimale entre 00 (correspond à 0) et FF (255 décimal).

- RR est le dosage de la couleur rouge,

- GG est le dosage de la couleur verte,

- BB est le dosage de la couleur bleue.

`color = "#FF0080"`

Valeurs particulières : Noms de fichiers

➤ Relatif

- Le fichier doit être disponible sur le même serveur Web que le fichier contenant le code HTML.
 - Sur le même répertoire \Rightarrow le nom du fichier directement : **"image.jpg"**
 - Sur un répertoire fils : **"images/image.gif"**
 - Sur un répertoire parent : **"../page.html"**
 - Combinaison : **"../dossier/page.htm"**
-
-

Valeurs particulières : Noms de fichiers

➤ Absolu:

- Que le fichier soit sur le même serveur Web ou sur un autre.
- On donne l'adresse complète du fichier.

`http://www.nomsite.org/doss/.../fich.ext`

➤ Il faut faire très attention à l'écriture dans le cas des fichiers et surtout à la casse (majuscules et minuscules).

Valeurs particulières : La taille

➤ Absolu :

- On donne une valeur entière positive ou nulle n et ça correspondra à n pixels.

➤ Relatif :

- On donne une valeur entière positive ou nulle suivi du symbole % ($n\%$) et ça correspondra à $n\%$ de l'espace disponible.

Valeurs particulières : L'alignement

➤ Horizontalement :

- **"Left"** : gauche.
- **"Right"** : droite
- **"Center"** : centré

➤ Verticalement :

- **"Top"** : haut.
 - **"Middle"** : milieu.
 - **"Bottom"** : bas.
-
-

Structuration d'une Page HTML

```
<HTML>
  <HEAD>          <!-- en-tête -->
    <TITLE>      Titre de la page  </TITLE>
  </HEAD>
  <BODY>          <!-- Corps de la page -->
    ...
    ...
  </BODY>
</HTML>
```

Remarque : L'indentation facilite la lecture du code.

Remarques :

- On peut écrire les balises en majuscules ou en minuscules. (L'interpréteur du HTML ne tient pas compte de la casse)
- Le formatage Manuel: saut de lignes, espaces ..., sont toujours ignorés.
- Les images, animations, sons, vidéos, ... sont insérés à partir de fichiers externes et ne font pas partie du source HTML.



L'en-tête :

- La majorité des éléments n'apparaissent pas.
- Certains sont liés à la page
- Certains sont des informations destinées aux moteurs de recherches
- Certains sont utilisés pour réaliser des traitements sur la machine cliente (petites animations ou défilements, vérifications, ...)



Éléments de l'en-tête :

➤ Le titre :

- Il apparaîtra sur la barre de titre du navigateur.
- On l'insère entre les balises `<title>` et `</title>`

➤ Un son de fond : `<bgsound src="fichier son" loop="nb_repetitions">` `loop à -1` \Leftrightarrow infini

➤ On peut insérer des scripts coté client (en JavaScript par exemple).

Éléments de l'en-tête : les meta

```
<META name="nom_META" content="valeur"  
lang="code_langue">
```

```
<META http-equiv="nom_META"  
content="valeur">
```

➤ Name : Author Title Keywords
Description Date Generator ProgId ...

➤ http-equiv : Content-type Content-
Language ...

➤ La valeur de **content** dépend de la valeur de
name ou de **http-equiv**

Exemples de meta :

➤ L'attribut **lang** (pas pour toutes les **meta**) aura comme valeur le code de la langue : **fr**, **en**, **ar-dz**, ...

```
<META name="keywords" content="html;  
programmation web; php" lang="fr">
```

```
<meta name="description" content="site  
destiné aux étudiants..." lang="fr">
```

```
<meta http-equiv="Content-Type"  
Content="text/html; charset="windows-  
1252">
```

Le corps :

- Défini par la balise `<body>`
- Tout ce qui sera mis (ou presque) dans cette partie s'affichera sur la page
- \Rightarrow Contient tous les éléments à afficher : textes, images, ...



Le Commentaire :

- Il y a aussi des balises de commentaires
- Peut être inséré n'importe où dans la page mais ne sera pas affiché par le navigateur.

<!-- ... le commentaire ... -->

- Peut être sur une ou plusieurs lignes.
- Pas d'imbrication de commentaires

Propriétés de la page (1) :

- A l'aide des attributs de la balise **BODY**. Ces attributs sont facultatifs
 - La couleur de fond de la page \Rightarrow L'attribut **BGCOLOR**.
 - La couleur par défaut du texte \Rightarrow L'attribut **TEXT**.
 - La couleur par défaut des liens \Rightarrow L'attribut **LINK**.
-
-

Propriétés de la page (2):

- La couleur par défaut des liens visités \Rightarrow L'attribut **VLINK**.
- La couleur par défaut des liens actifs \Rightarrow L'attribut **ALINK**.
- Les valeurs de tous ces attributs obéissent à la syntaxe de la couleur (nom ou RGB).

Propriétés de la page (3):

- Image de fond sur la page \Rightarrow L'attribut **BACKGROUND**.
- Sa valeur obéit à la syntaxe des fichiers.
- Par défaut l'image de fond défile avec le texte, pour la fixer on rajoute l'attribut **bgproperties="fixed"**.

Propriétés de la page (4) :

➤ Direction de l'écriture du texte : par défaut de gauche à droite

- Pour modifier la direction \Rightarrow L'attribut **DIR**.

- L'attribut **DIR="rtl"** pour une direction de droite à gauche (texte en Arabe, par exemple).

➤ Les marges de la page :

- La marge du haut \Rightarrow L'attribut **topmargin**

- La marge gauche \Rightarrow L'attribut **leftmargin**

Mise en forme du texte (1) :

➤ Pour définir la police, la taille et/ou la couleur d'un texte en le met entre **** et ****.

Comme suit :

```
<font face="nom_police_car"  
color="couleur" size="nb"> ... le texte ...  
</font>
```

➤ On peut définir un, deux ou tous les attributs pour cette balise.

Mise en forme du texte (2) :

- **Face** : on peut attribuer un ou plusieurs noms de polices de caractères
- **Color** : Obéit à la syntaxe de la couleur
- **Size** : définit la taille du caractère et peut prendre les valeurs de 1 à 7.
- 1 correspond à 8pts, 3 à 12pts et 7 à 36pts.

Mise en forme du texte (3) :

- Gras : ` ... ` ou ` ... `
- Italique : `<I> ... </I>` ou ` ... `
- Souligné : `<U> ... </U>`
- Exposant : `^{...}`
- Indice : `_{...}`

Mise en forme du texte (4) :

- Il faut faire attention à la fermeture de balises : ordre « pile »
- En cas de conflit entre balises, la balise la plus proche sera appliquée
- Il y a d'autres balises comme **blink** (clignotant) et **strike** (barré) qui sont déconseillées.

Mises en formes particulières :

➤ Niveau de titres : balise `<H1>` à `<H6>`.

➤ Code : `<code>` ... `</code>`

➤ Clavier : `<kbd>` ... `</kbd>`

➤ Variable : `<var>` ... `</var>`

➤ Citation : `<cite>` ... `</cite>`

➤ Définition : `<dfn>` ... `</dfn>`

➤ Exemple : `<samp>` ... `</samp>`

Mises en formes particulières :

- Bloc en retrait : `<blockquote> ... </blockquote>`
- Adresse : `<address> ... </address>`
- Imprimé (machine à écrire) : `<tt> ... </tt>`
- Texte préformaté : `<pre> ... </pre>`
- Agrandir ou réduire : `<big> ... </big>` ou `<small> ... </small>`

Les paragraphes :

- Un paragraphe est défini par **<P>**
- La balise fermante **</p>** est facultative.
- L'alignement : L'attribut **Align** qui peut avoir les valeurs : **"center"**, **"left"** ou **"right"**.

Les séparateurs :

- Caractère de retour à la ligne **
**
- N'a pas de balise fermante
- Afficher une ligne horizontale **<HR>**
- N'a pas de balise fermante
- Elle définit un nouveau paragraphe indépendant de ce qui la précède et de ce qui la suit.

Les séparateurs : barre horizontale

- **Align** : alignement de la barre : **"left"**, **"center"**, **"right"**
- **Width** : la largeur de la barre,
 - Soit en absolu (valeur entière ≥ 0) qui correspondra à la largeur en pixel.
 - Soit en relatif (valeur entière ≥ 0 suivi de %) qui correspondra au pourcentage de l'espace disponible.

Les séparateurs : barre horizontale

- Size : l'épaisseur de la barre en pixels.
- Noshade : attribut sans valeur pour éliminer l'ombrage 3D. N'est pas reconnu par tous les navigateurs.
- Color : Obéit à la syntaxe de la couleur et définit la couleur de la barre.

Les listes (1) :

➤ Listes numérotés automatiquement : sont insérées à l'aide de `` et ``

- `` marque le début de la liste
- `` marque la fin de la liste

➤ Chaque élément de la liste doit être précédé par la balise ``.

➤ `` est facultative.

Les listes (2):

- Par défaut la numérotation est : 1, 2, ... etc.
 - Pour changer la façon de numérotter \Rightarrow l'attribut **type** dans ****.
 - **Type** prend les valeurs :
 - **"I"** ou **"i"**: grands ou petits chiffres romains,
 - **"A"** ou **"a"**: alphabets majuscules ou minuscules,
 - **"1"**: numérotation ordinaire.
-
-

Les listes (3):

- Par défaut la numérotation commence à partir de 1, a, A, i ou I. (Selon le type)
- Pour changer le numéro du premier éléments \Rightarrow l'attribut **start** dans ****.
- Sa valeur est un entier "*l*" et ça commencera de :
 - *l* Si la numérotation, est ordinaire.
 - *l* en chiffres romains, Si le type est I ou i
 - *l*^{ème} lettre de l'alphabet, Si le type est A ou a.

Les listes (4) :

➤ Listes à puces : sont insérées à l'aide de **** et ****

- **** marque le début de la liste
- **** marque la fin de la liste

➤ Chaque élément de la liste doit être précédé par la balise ****.

➤ **** est facultative.

Les listes (5):

➤ Il est possible de changer la forme de la puce \Rightarrow l'attribut **type** :

- dans **** (pour toute la liste)
- dans **** (pour un élément).

➤ On peut importer une image pour jouer le rôle de la puce \Rightarrow l'attribut **imagesrc** dans ****. Sa valeur obéit à la syntaxe des fichiers.

Les listes (6):

- On peut imbriquer les listes.
- Un élément **** peut contenir une liste **** ou ****.
- On peut insérer un autre type de liste \Rightarrow Les listes de définitions.
- Permet d'afficher des glossaires

Les listes (7) :

- On utilise `<DL>` et `</DL>` pour marquer le début et la fin de la liste.
 - Chaque élément (un titre) doit être encadré par `<DT>` et `</DT>`.
 - La définition de l'élément doit être encadrée par `<DD>` et `</DD>`.
 - `</DT>` et `</DD>` sont facultatives.
-
-

Les liens :

- Il y a deux types de liens (appelés aussi références) qu'on peut définir sur un texte ou une image :
 - **Internes:** Ils désignent un point précis, repéré par une balise, dans le corps même du document.
 - **Externes:** Spécifient simplement une autre page HTML, un document multimédia ou l'accès à un service comme le e-mail ou le FTP.
-
-

Les liens internes :

- On fixe un endroit dans le document (la cible).
- On définit une étiquette à cet endroit (une ancre)

`...`

- On définit le lien (la référence) :

`....`

Les liens externes :

`....`

- La valeur de **HREF** obéit à la syntaxe de fichier.
 - Comme il peut aller vers un autre service si la valeur de **HREF** est :
 - **"ftp://url"** : va vers un le serveur ftp spécifié par l'url.
 - **"mailto:adr_email"** : ouvre l'application mail pour l'envoi d'un email à l'adresse définit par adr_email.
-
-

Insertion d'une image :

- On utilise la balise ****
- L'image ne fait pas partie du code HTML. Elle est importée à partir d'un fichier externe.
- Attribut obligatoire **SRC** : sa valeur obéit à la syntaxe de fichier. Extensions : **.gif .jpg .jpeg .png**
- L'image sera affichée à sa dimension réelle.

Propriétés d'une image (1):

- On peut spécifier la dimension (hauteur, largeur) de l'image à l'aide des attributs **width** et **height**
- Leurs valeurs doivent être en pixels.
- Attention à la déformation de l'image!!!
- On peut définir un texte alternatif à l'aide de l'attribut **alt**. Sa valeur est texte libre.

Propriétés d'une image (2):

- Bordure autour de l'image \Rightarrow Attribut **border** : sa valeur en pixels correspondra à l'épaisseur.
- Alignement par rapport au texte autour de l'image \Rightarrow Attribut **align** : sa valeur **left**, **right**, **center**, **top**, **bottom** ou **middle**.
- Espacement par rapport au texte autour de l'image \Rightarrow Attributs **hspace** (à droite et à gauche) et **vspace** (en haut et en bas) : valeurs en pixels

Insertion d'un tableau :

- Les tableaux sont très importants dans la mise en forme d'une page Web.
 - Un tableau est un ensemble de lignes et de colonnes
 - Un élément d'une ligne donnée i et d'une colonne donnée j est appelé cellule
 - On utilise les balises **<TABLE>** et **</TABLE>**
-
-

Les tableaux :

- Un tableau HTML est organisé en lignes (suite de plusieurs lignes).
- Chaque ligne contient un ensemble de cellules
- Il faut s'assurer d'insérer le même nombre de cellules dans chaque ligne.

Année	Effectifs	Évolution
2000	18000	12%
2002	20000	10%
2004	21000	05%
2006	21500	2.5%

Lignes et Cellules d'un tableau :

- Ligne (Table Row) \Rightarrow Balises **<TR>** et **</TR>** : la balise de fin de ligne n'est pas obligatoire.
 - Cellule (Table Data) \Rightarrow Balises **<TD>** et **</TD>** : la balise de fin de cellule n'est pas obligatoire.
 - Cellules spéciales \Rightarrow Balises **<TH>** et **</TH>** : Titre de colonne. On peut s'en passer!!!
 - Titre du tableau \Rightarrow Balises **<CAPTION>** et **</CAPTION>** : A insérer juste après **<TABLE>**.
-
-

Propriétés du tableau (1) :

- Alignement du tableau : on encadre le tableau par les balises `<DIV>` et `</DIV>` avec comme attribut `Align`.
- Aussi à l'aide de l'attribut `Align` de `<table>`.
- Largeur et Hauteur \Rightarrow attributs `width` et `height` : en absolu (pixels) ou en relatif (%).

Propriétés du tableau (2):

- Bordure \Rightarrow attributs **border** et **bordercolor** : le 1^{er} en pixels pour l'épaisseur de la bordure et le 2^{ème} pour définir sa couleur.
- Pour avoir un effet 3D à la bordure \Rightarrow bordure claire et la bordure sombre avec **bordercolorlight** et **bordercolordark**.
- Leurs valeurs sont des couleurs

Propriétés du tableau (3):

- Fond \Rightarrow attributs **bgcolor** et **background** : le 1^{er} pour définir la couleur de fond et le 2^{ème} pour fixer une image de fond.
- Espacements \Rightarrow attributs **cellspacing** et **cellpadding** : Valeurs en pixels;
- Le 1^{er} pour définir l'espacement entre 2 cellules et le 2^{ème} pour fixer la marge interne des cellules

Propriétés des cellules (1) :

- A l'aide des attributs de **<TD>** (ou **<TH>**)
- Alignement à l'aide des attributs **Align** et **VAAlign**.
- **left**, **right** ou **center** pour **Align** et **top**, **bottom** ou **middle** pour **VAAlign**
- Largeur et Hauteur \Rightarrow attributs **Width** et **Height** : en absolu (pixels) ou en relatif (%).

Propriétés des cellules (2) :

- Le fond \Rightarrow de la même façon que pour `<table>` : attributs `bgcolor` et `background`.
- Couleur bordure \Rightarrow de la même façon que pour `<table>` : attributs `bordercolor`, `bordercolorlight` et `bordercolordark`.
- Éviter le renvoi automatique à la ligne \Rightarrow attribut sans valeur `nowrap`

Fusionnement des cellules (1) :

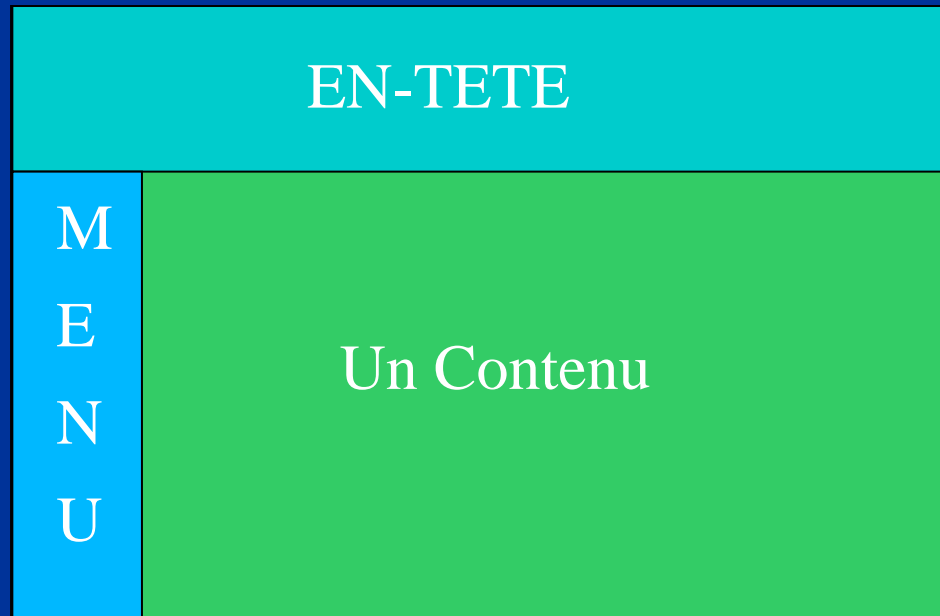
- On peut fusionner des cellules adjacentes horizontalement ou verticalement.
 - On utilise des attributs dans la balise `<TD>`
 - Fusionnement horizontal \Rightarrow attribut `colspan` : sa valeur est un entier $n \geq 2$ qui correspond au nombre de cellules fusionnées.
 - Les $n-1$ autres cellules sur la même ligne doivent être supprimées.
-
-

Fusionnement des cellules (2) :

- Fusionnement vertical \Rightarrow attribut **rowspan** : sa valeur est un entier $n \geq 2$ correspondant au nombre de cellules fusionnées.
- Une cellule sur les $n-1$ lignes suivantes doit être supprimée.

Frames – Cadres :

➤ Les frames (cadres) permettent de définir des zones d'affichage dans la page Web.



➤ Chaque frame correspond à une page HTML + la page des cadres (exple. ci-dessus nécessite 4 pages)

Page des Cadres (1):

- La syntaxe de la page des cadres est particulières.
 - Le corps de la page est défini par la balise **<FRAMESET>** et **</FRAMESET>** au lieu de **<body>**.
 - La balise **<FRAMESET>** définit une décomposition (horizontale ou verticale) en 2 ou plusieurs cadres.
 - Décomposition horizontale \Rightarrow l'attribut **COLS**.
 - Décomposition verticale \Rightarrow l'attribut **ROWS**.
-
-

Page des Cadres (2):

- Sa valeur est une suite d'éléments séparés par des virgules.
- Chaque élément représente la hauteur d'une ligne ou la largeur d'une colonne (espace réservé).
- Un élément (ligne ou colonne) peut être un cadre défini par la balise **<FRAME>** ou un ensemble de cadres défini par la balise **<FRAMESET>**.

Page des Cadres (3):

➤ L'espace réservé (hauteur ou largeur) est défini par :

- Le nombre de pixels (entier ≥ 0)
- Un pourcentage (entier ≥ 0 suivi de %)
- * (pour utiliser tout l'espace restant).

➤ Par exemple :

<Frameset rows="100,*,150,10%">.

Propriétés des Cadres (1):

- Espacement entre cadres \Rightarrow attribut **framespacing** de la balise **<Frameset>**. Sa valeur est un entier.
- Entre **<Frameset>** et **</Frameset>**, on doit définir tous les cadres.
- Chaque cadre est défini par la balise **<Frame>**. On peut avoir au lieu d'un cadre, une autre décomposition à l'aide de **<Frameset>**.

Propriétés des Cadres (2):

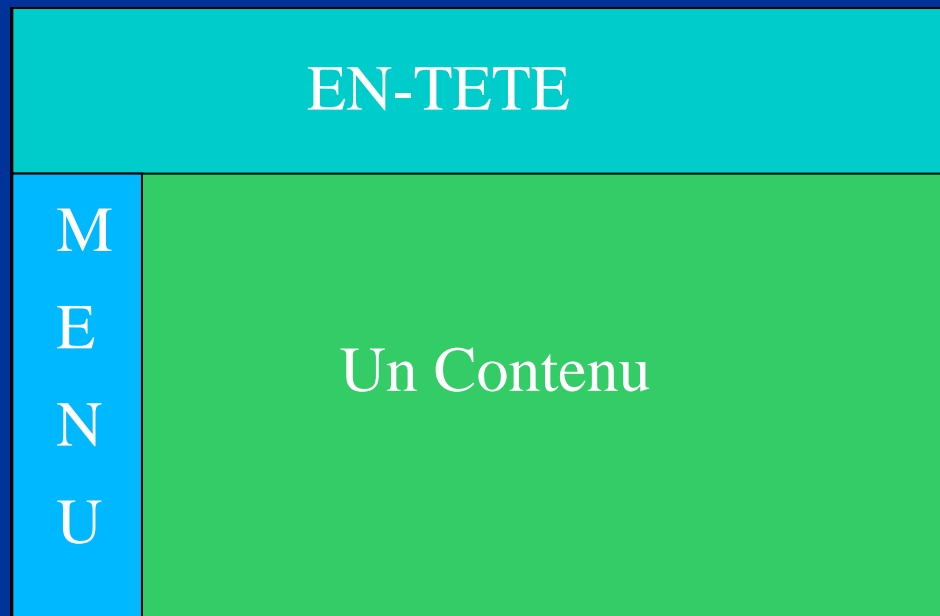
- Les autres propriétés sont fixées à l'aide des attributs de la balise **<Frame>**.
 - Nom du cadre \Rightarrow attribut **Name** : valeur libre.
 - Page HTML associée au cadre \Rightarrow attribut **SRC**.
 - Cadre d'ouverture par défaut des liens \Rightarrow attribut **Target** : sa valeur doit être le nom d'un cadre qui existe.
-
-

Propriétés des Cadres (3):

- Barres de défilements \Rightarrow attribut **Scrolling** : valeur **auto**, **yes** ou **no**.
- Interdire de redimensionner le cadre \Rightarrow attribut sans valeur **noresize**.
- Marges internes du cadre \Rightarrow attributs **MarginHeight** et **MarginWidth**.

Exemple de Cadres:

➤ On veut obtenir des cadres comme le montre le schéma ci-dessous :



➤ On a 2 lignes de cadres et la 2ème ligne composée de 2 colonnes de cadres.

Liens dans les Cadres (1) :

- Il faut définir le cadre cible où s'ouvrira le lien.
 - Par défaut, le lien s'ouvrira dans le cadre spécifié par **Target** dans la balise **<FRAME>**, mais on peut le changer.
 - On rajoute à la balise **<a>** l'attribut **TARGET** avec le nom d'un cadre comme valeur.
 - Pour ouvrir le lien dans le même cadre \Rightarrow Valeur **_self** à **Target**.
-
-

Liens dans les Cadres (2) :

- Pour ouvrir le lien dans le cadre père \Rightarrow Valeur **_parent** à **Target**.
 - Pour ouvrir le lien dans la page entière \Rightarrow Valeur **_top** à **Target**.
 - **_parent** et **_top** ne donnent pas la même chose si le cadre est dans une page de cadres qui est cadre.
 - Pour ouvrir le lien dans une nouvelle fenêtre \Rightarrow Valeur **_blank** à **Target**.
-
-

A propos des Cadres :

- Il existe des navigateurs qui ne reconnaissent pas les cadres
- ⇒ La solution est d'insérer juste avant la dernière **</frameset>** le corps de la page qui s'affichera au lieu des cadres entre **<noframes>** et **</noframes>**.
- Les cadres sont déconseillés pour divers raisons :
 - Le nombre de pages utilisées.
 - L'indexation dans les moteurs de recherche.

Les formulaires :

- Un formulaire est un ensemble d'éléments permettant la saisie d'informations.
 - Le HTML s'occupe du côté description seulement, pas du traitement des informations saisies.
 - Pour définir un formulaire on utilise les balises : **<form>** et **</form>**.
 - Permet de délimiter les éléments du même formulaire.
-
-

Propriétés du formulaire (1) :

- Parmi les attribut de **<form>** :
 - Le nom du formulaire \Rightarrow **Name** : très utile dans les traitements coté client (JavaScript par exemple).
 - Le nom de l'application qui traitera le formulaire \Rightarrow **Action** : Sa valeur obéit syntaxe fichiers.
 - **Target** : Utilisé de la même façon que pour la balise **<a>** pour l'affichage de la réponse (quel cadre ou nouvelle fenêtre)
-
-

Propriétés du formulaire (2) :

➤ La méthode d'envoi des données \Rightarrow **Method** : 2 valeurs possibles

- **GET** : Les données sont envoyées avec l'URL. Nombre de caractères limité. Données apparaissent sur la barre d'adresse du navigateur.
- **POST** : Les données sont envoyé dans une zone spéciale sur le serveur. Pas de limites pour les données.

Éléments des formulaires :

➤ Le formulaire est composé d'éléments d'entrée définis par l'une des trois balises :

- **<INPUT>**
- **<TEXTAREA>**
- **<SELECT>**

➤ Ils ont tous le même attribut : **NAME**. Il les identifie de manière unique. Attribut obligatoire et important.

Les éléments (champs) <INPUT> :

➤ La valeur de l'attribut **Type** permet de distinguer les éléments **<INPUT>** :

- **TEXT** : Zone (Champ) de texte simple.
 - **PASSWORD** : Zone de saisie de mot de passe.
 - **HIDDEN** : Champ masqué.
 - **RADIO** : Case d'option (bouton radio).
 - **CHECKBOX** : Case à cocher.
 - **FILE** : Pour sélectionner un fichier à envoyer.
 - **BUTTON** : Bouton de commande (simple).
 - **SUBMIT** : Bouton d'envoi.
 - **RESET** : Bouton de réinitialisation.
-
-

Les champs <INPUT> textuels (1):

- Il s'agit des champs de type **Text**, **Password** et **Hidden**.
 - **Text** : Champ (zone) de saisie de texte simple permet de saisir un texte comme : nom, prénom, ...
 - **Password** : Zone de saisie mot de passe permet de faire apparaître à l'écran chaque caractère en *.
 - **Hidden** : Champ textuel masqué. Permet de transférer des données sans les afficher à l'écran.
-
-

Les champs <INPUT> textuels (2):

➤ Les champs <INPUT> de type **Text** ou **Password** peuvent avoir d'autres attributs facultatifs :

- **Value** : Texte par défaut qui s'affichera.
- **Size** : Largeur du champ en nbre de caractères.
- **Maxlength** : Le nombre max de caractères.

➤ Le champ <INPUT> de type **Hidden** nécessite obligatoirement l'attribut **Value**.

Les champs <INPUT> cases :

- **Radio** : Case d'option (bouton radio). Permet un choix en exclusion mutuelle (ayant le même nom).
- **Checkbox** : Case à cocher. Pour sélectionner plusieurs options à la fois.
- Ont les mêmes attributs :
 - **Value**: La valeur.
 - **Checked**: pour sélectionner ou cocher par défaut.

Les champs <INPUT> boutons (1):

- **Button** : Bouton simple, Utile généralement avec scripts coté client (Javascript, ...)
- **Submit** : Lance l'envoi des données du formulaire à l'application (sur le serveur) définie par **Action**.
- **Reset** : Annule le formulaire en effaçant toutes les données saisies et en remettant les valeurs par défaut.

Les champs <INPUT> boutons (2):

- **Value** : Définit le label du bouton (texte apparaissant sur le bouton).
- L'attribut **Name** n'est pas obligatoire pour **Submit** et **Reset**.

Les champs <INPUT> fichier :

- **File** : Permet d'envoyer dans le formulaire un fichier de la machine utilisateur vers le serveur.
- Zone textuelle + bouton de parcours
- **Size** : La taille de la zone textuelle en nombre de caractères.
- **Accept** : Définit un filtre. Les types de fichiers autorisés.

Champs de saisie multilignes :

- **<TEXTAREA>** et **</TEXTAREA>**.
- Entre les 2, on peut mettre le texte par défaut.
- En plus de **name**, **<textarea>** a d'autres attributs facultatifs :
 - **Rows** : Le nombre de lignes qu'occupera la zone de saisie.
 - **Cols** : Le nombre de caractères qu'occupera la zone en largeur.

Liste déroulante (zone de sélection) :

➤ **<SELECT>** et **</SELECT>** pour délimiter la liste.

Marque le début et la fin de la liste.

➤ En plus de **name**, **<Select>** a d'autres attributs facultatifs :

- **Size** : Le nbre de lignes qu'occupera la zone.
- **Multiple** : Attribut sans valeur, pour autoriser le choix multiple. (En utilisant la touche clavier CTRL)

Éléments de la liste déroulante :

- **<OPTION>** pour de définir un élément de la liste.
- L'attribut **Value** permet d'associer une valeur au choix sélectionné. C'est cette valeur qui sera envoyée au serveur.
- L'attribut **Selected** (sans valeur) permet de sélectionner l'élément par défaut.
- Le texte qui s'affichera sera mis après **<OPTION>**.

Formulaires (Complément) :

- L'attribut **TabIndex** peut être rajouté à toute balise d'un élément du formulaire. Sa valeur est un entier. Définit l'ordre de sélection de cet élément en utilisant la touche du TABULATION du clavier.
- **<fieldset>** et **</fieldset>** permet à l'intérieur du même formulaire de créer des blocs.
- **<Legend>** et **</Legend>** : Pour le titre du bloc.

