

# Apprendre HTML Pour Les Zéros

- Mis en page par KoraS.

- Mis en partage sur le réseau eDonkey le mercredi 2 juin 2004 par Koras
  - Ce cours de HTML viens du site : <http://www.Mcours.com/>

Détestant lire sur mon PC et aimant le travail bien fait, j'ai mis en page a fin d'impression ce cours très complet et si j'ai décider de le mettre en partage sur le réseau pour les gens qui ne connaîtrais pas le site du zéro (a visiter ABSOLUMENT) et/ou qui voudrais l'imprimer CORRECTEMENT pour le lire tranquillement dans leurs pieux.

Aller, bonne lecture les gars !!!!

- Programmez, crackez, scannez, rippez et mettez tout ça en partage sur le réseau eDonkey, on va leurs apprendre a nous vendre des logiciels a 200 Euros, des jeux a 60 Euros, des albums a 20 Euros, des livres a 50 Euros, des magazines a 8 Euros a tous ces empafés de capitalistes !
- A propos du réseau, j'ai remarqué depuis quelques temps une certaine tendance de la communauté eDonkey Française a prendre et a ne pas laisser en partage, cela est sans doute du au fait de la démocratisation d'Internet et des connections haut débits et donc a l'émergence de nombreux newbies (débutants), qui prennent sans se demander pourquoi, comment, par qui.....
- Vous vous devez de laisser en partage quelques temps ce que vous venez de prendre pour qu'il y est plus de « full sources » (personne possédant le fichier en entier), c'est **OBLIGATOIRE** !!!! Qu'est ce que vous risquez en n'appliquant pas cette loi ? C'est simple vous risquez un jours ou l'autre de devoir acheter tous ce que vous prenez **GRATUITEMENT** avec votre Mule les gars !!!! Et oui, déjà qu'avec les lois pourrisses qui ont étaient votées ces derniers temps de nombreuses personnes luttent contre nous, les FAI's (Fournisseurs d'Accès à Internet) collaborent avec les keufs et autres R.I.I.A, les espions sont partout sur le réseau et même sur nos PC's (et je suis pas parano, ce que je dit est prouver), alors si les utilisateurs se mettent a partir en couilles aussi, je ne donne pas 2 ans au Roi des réseau Peer To Peer (eDonkey bien surs) ! Contre ça, partageons le plus et le plus longtemps possible.
- J'ai également remarquer une montée en flèche du nombres de fichiers incomplets, quand vous décidez de mettre un fichier en partage, vous endossez une responsabilité, définissez la priorité de ce fichier au maximum (release ou très haute selon les programmes utilisés) en cliquant droit dessus dans la partie partage de votre logiciel, surtout, surtout, surtout, assurez vous avant de l'enlever qu'il y est au moins 20 personnes qui l'ont en entier et mieux encore, ne l'enlever pas, assumez jusqu'au bout votre rôle.
- La survie du réseau est entre nos mains les gars, faites tourner l'info, **IL FAUT PARTAGER !!!**

**Longue vie a eDonkey et a tous les Hackers, Crackers, Rippers, Suppliers, Webmestres, etc, qui se décarcasse pour notre plaisir sans rien y gagner... au fait si, ils y gagnent notre estime et nos éternels remerciements !!!!!**

# Koras

[www.Mcours.com](http://www.Mcours.com)

Site N°1 des Cours et Exercices Email: [contact@mcours.com](mailto:contact@mcours.com)

---

# Les Bases Du HTML

Nous commencerons dans ce premier chapitre à préciser ce qu'est le html et ce que vous pouvez faire avec.

Plus bas, vous devrez retenir quelques mots de vocabulaire indispensables à la compréhension de tout ce qui suivra.

Vous aurez à la fin quelques petits mots sur la syntaxe. Rien ne vous permettra donc ici de commencer à créer votre page en html, mais tout vous sera indispensable si vous êtes un zéro !

## Qu'est-ce que le html ?

Vous connaissez sûrement le web, et vous aimez certains sites. Peut-être que vous les envie, même. Sachez que dans la plupart des cas, votre fournisseur d'accès (ou provider) tels que Aol, Infonie, Wanadoo, Club Internet, Compuserve... vous donne gratuitement de la place sur le web pour déposer votre page perso. Cette place peut occuper maximum 5 à 15 Mo, car on vous limite celle-ci.

Alors, vous avez fait tout ce qu'il fallait pour savoir comment utiliser cette place sur le web. Oui mais voilà : comment allez-vous faire pour créer ces pages que vous envie tant ? Eh bien, vous allez devoir utiliser le seul moyen que tout le monde utilise : le langage html.

Le html est un langage de description de documents, pas de programmation. En plus clair, c'est un langage informatique qui sert à éditer des pages (virtuelles), mais qui ne permet pas de créer des programmes.

Enfin, si vous trouvez une bonne idée pour votre site perso, vous pourrez voir le nombre de visiteurs augmenter considérablement. Tout est donc dans l'idée, mais si vous ne savez pas exploiter le html, cela ne vous servira à rien (ce qui explique la présence de mon site). Le prochain paragraphe vous explique où taper ce langage html.

## Où écrire en html ?

Quel est donc l'environnement magique dans lequel vous allez écrire votre page perso ? Il faut quand même savoir que ce n'est pas en tapant du texte n'importe où, et en l'enregistrant en .html que ça va marcher ! C'est tout de même plus compliqué que ça ! Toutefois, c'est un langage destiné à tous les internautes, donc il n'est pas impossible à utiliser.

Je vais vous le dire de suite, vous avez tout ce qu'il faut pour le faire. N'importe quel éditeur de texte ASCII suffit. Et parmi eux, il y a edit, sous Dos. Sous Windows, vous pouvez utiliser bloc-notes (c'est celui que j'ai choisi pour ma part).

La deuxième solution, c'est d'utiliser un éditeur WYSIWYG, soit : What You See Is What You Get. En bon français : ce que vous voyez est ce que vous obtenez. Avec ça, un enfant de 5 ans peut créer sa page perso. L'interface ressemble à un programme de traitement de textes ordinaire. Il vous suffit de cliquer sur italique, rouge, centré... et vous voyez illico le résultat !

Mais tout n'est pas si beau. En effet, le résultat que vous aurez devant les yeux sera très approximatif. De plus, si vous choisissez cette méthode, je ne pourrai pas vous aider, car ici, je parle de html "pur", sous un éditeur de texte ASCII.

Citons, en éditeurs WYSIWYG, Netscape Composer, livré avec Netscape, et Front Page Express, livré avec Internet Explorer.

En bref, la meilleure solution qui vous offre le plus de choix, c'est un éditeur ASCII. Pour ouvrir Bloc-notes, cliquez sur :

- Démarrer
- Programmes
- Accessoires
- Bloc-notes

Vous voilà maintenant devant un grand vide blanc. C'est là que nous allons opérer...

## Vocabulaire essentiel

Je vous l'ai déjà dit, il ne suffit pas de taper ce que vous voulez voir s'afficher pour avoir un résultat. Cela ne marche pas. Qu'y a-t-il d'autre, alors ? La SEULE et unique chose qui vient troubler vos esprits, ce sont les **TAGS**. C'est tout ce que vous verrez en plus par rapport à votre traitement de texte habituel.

Un tag ? C'est une information qui s'adresse au browser (Netscape ou Internet Explorer). Elle ne sera pas affichée. Le tag donne une information sur la taille, la couleur du texte, le lien...

Comment le distingue-t-on du texte ? Tout tag commence par un < et se termine par un >. Par exemple (j'invente) : le tag météo. Il s'écrit :

```
<météo>
```

Et des tags, je vous en donnerai des tonnes, mais petit à petit.

Il existe un deuxième mot à connaître : **ATTRIBUT**.

Un attribut se trouve à l'intérieur même d'un tag. Il donne des informations plus précises sur le tag. Ce dernier peut avoir un ou plusieurs attributs, selon votre choix.

Revenons à notre tag météo. On peut lui donner les attributs été et température. Vous voyez bien qu'ils précisent ce tag. Regardez plutôt :

```
<météo été température="34°">
```

Tous les attributs doivent avoir un espace entre eux pour les séparer. Mais il s'agit toujours du tag météo, dont on doit écrire le nom en premier. Je vous en dirai plus dans le chapitre suivant.

# Syntaxe

## La présentation

Voilà quelque chose de très important à savoir : la présentation de votre page web, lorsque vous l'écrivez en html sous bloc-notes, ne donnera strictement rien lorsque les internautes la verront avec leur browser. Vous pouvez vous amuser à mettre 13 tabulations, 10 entrées, mais vous ne verrez rien de spécial lorsque vous la regarderez avec Internet Explorer ou Netscape.

Je vous donne 2 exemples pour être plus clair. J'ai mis des entrées n'importe où, fait des espaces n'importe où dans l'un, et pourtant le résultat sera le même dans les 2 cas. Attention : tout ceci se passe lorsque vous écrivez du texte qui sera affiché, comme dans un traitement de texte. Ne jouez pas à ça avec les tags !

Bien sûr, le browser va à la ligne quand le texte arrive au bout de celle-ci.

Exemple 1, où je n'ai pas mis des entrées n'importe où :

```
Julien marchait dans une sombre forêt, où régnait un froid
glacial. Il se demandait combien de temps il errerait encore
dans ces lieux sinistres. Il voulait retrouver la douce
chaleur du foyer...
```

Exemple 2 :

```
Julien marchait dans une sombre
forêt, où      régnait un froid
glacial. Il se      demandait combien
de temps
il errerait     encore
dans ces lieux sinistres. Il voulait
retrouver la douce
chaleur  du      foyer...
```

Vous n'avez droit qu'à un espace entre 2 mots. Vous avez vu dans l'exemple 2 que j'en avais mis partout, mais le résultat sera le même qu'avec l'exemple 1. C'est comme ça.

A quoi ça sert ? C'est pour vous qui écrivez votre page. C'est juste pour vous, étant donné que l'internaute ne saura pas si vous avez mis des entrées et des espaces ou pas.

Ca vous permet "d'éclaircir" votre page sous bloc-notes. Ainsi, la prochaine fois que vous reviendrez agrandir ou modifier votre page, vous localiserez plus vite un endroit précis. Il vaut mieux quelque chose de clair pour vous plutôt qu'un gros pâté de texte... Evidemment, j'ai un peu exagéré dans le deuxième exemple.

Résumons : logiquement, vous n'aurez pas à faire plusieurs espaces entre 2 mots car ce n'est pas ça qui éclaircira votre texte sous bloc-notes et qui le placera où vous le voulez dans le browser. Mais pour vous, mettre une ou deux entrées entre 2 paragraphes donnera quelque chose de plus clair. Seulement pour vous, car pour faire des entrées, il existe un tag.

Si je vous ai dit de ne pas le faire avec les tags, c'était pour que vous ne fassiez pas d'entrées au milieu d'un tag ! Mais vous verrez, le troisième exemple en partant d'ici (avec gras et souligné) vous montre qu'il est tout à fait possible de mettre des entrées entre 2 tags.

## Les tags

Vous savez comment écrire un tag. Mais, en général, un tag ne peut exister seul. Il faut le fermer. Pour cela, recréez le même tag plus loin, sans les attributs s'il y en avait. Vous ajouterez un / au début du nom du tag. Avec le tag météo :

```
<météo été température="34°"></météo>
```

Des tags interdisent leur fermeture, ils ne doivent pas être fermés. Question de logique : si vous dites qu'il y a une image ici, pas besoin de fermer le tag...

Entre les 2 tags, vous écrirez du texte qui sera affiché, en respectant les indications des tags. J'abandonne momentanément le tag météo, et je le remplace par un autre de mon invention :

```
<souligné>Ce texte est souligné</souligné>
```

Ainsi, la phrase "ce texte est souligné" apparaîtra souligné (n'essayez pas, c'est un faux tag !).

Il est possible que vous utilisiez plusieurs tags en même temps. Mon prochain exemple illustre ce que je vous ai dit à propos de la présentation.

```
<gras><souligné>Gras et souligné</souligné>Seulement  
gras</gras>
```

Comme indiqué, une partie du texte sera en gras et souligné, une autre seulement en gras. Vous pourrez ainsi vous amuser à "jongler" avec les tags. Cependant, cela corse quelque peu la chose... Il va falloir faire attention à l'ordre de fermeture. Voici le mauvais exemple à ne pas suivre :

```
<gras><souligné>Ca ne marche pas...</gras></souligné>
```

Voici le bon exemple :

```
<gras><souligné>En gras et souligné</souligné></gras>
```

C'est compris ? Il faudra faire pareil, même avec 10 tags ! Attention donc.

## Les attributs

Comme les attributs précisent un tag, ils ne sont normalement pas obligatoires. Mais certains tags ne sont utiles qu'avec un ou plusieurs attributs.

Vous savez les insérer, en les séparant chacun d'un espace. A vous de voir si vous en mettez, ou si vous n'en mettez pas. Ce que vous devez savoir, c'est qu'il y a 2 types d'attributs. On précise certains, d'autres pas.

Vous vous souvenez des attributs été et température. Ils font partie chacun d'une catégorie différente. Été n'est pas précisé. C'est tout simple.

Mais température l'est. Voici ce que l'on doit faire : sans espaces (puisqu'ils servent à séparer les attributs), vous écrivez après l'attribut un = et deux ". Entre les 2 " vous préciserez l'attribut, par un nombre ou par un mot. Je vous redonne cet ancien exemple :

```
<météo été température="39°">
```

# Structure D'Une Page HTML

Maintenant nous allons commencer à écrire pour de bon. Comme je vous l'ai déjà expliqué, ouvrez bloc-notes. Vous allez à partir de ce moment précis (et ce jusqu'à la fin) "jongler" entre les 2 fenêtres : celle de bloc-notes et celle de votre browser dans laquelle vous lisez ceci.

Vous écrirez la page en langage html sous bloc-notes, puis vous regarderez le résultat avec votre browser pour vérifier (après avoir enregistré, bien sûr).

## En-tête et corps

Notre tout premier tag sera le tag <html>. Comme la plupart, il s'ouvre et se ferme. Il dit où commence la page html et où elle se termine. Donc vous n'écrirez rien en dehors de ce tag html. Commencez donc à écrire :

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

Ca va jusque là ?

Dans une page html, on distingue 2 parties différentes : l'en-tête et le corps que l'on met dans cet ordre respectif. Je ne détaillerai pas ici ce qu'il contiennent, mais je vous en donnerai une idée. Commencez d'abord par écrire ces 2 tags qui délimitent ces parties.

Le tag de l'en-tête est <head>. Celui du corps est <body>. Essayez d'imaginer dans votre tête et sans continuer à lire, à quoi va ressembler votre page html. Voilà la réponse :

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

Le tag head sera minuscule par rapport au tag body à la fin. Mais head contient des informations importantes sur la page qui s'adressent plus particulièrement aux moteurs de recherche. Le prochain chapitre lui est consacré.

Quant à body, il risque d'être gigantesque ! C'est lui qui contiendra l'énorme majorité des tags qui existent. C'est entre les 2 body que vous écrirez du texte qui sera affiché, que vous mettrez en gras, italique, rouge, barré... Vous mettrez aussi des tableaux, des images... Bref, tous les prochains chapitres sauf le 3° lui sont consacrés.

Je commencerai déjà ici à vous en parler...

Tout d'abord, enregistrez votre page. Comme c'est la structure de base de la page html, vous serez obligé d'écrire ça pour toutes les pages. La bonne idée serait de la conserver comme modèle. Effectuez les opérations suivantes :

Fichier  
Enregistrer  
Choisissez un répertoire  
Type : "Tous"  
Nom : "nom.html"  
Enregistrer

Remplacez le "nom" par le nom de votre choix.

Vous pouvez essayer de regarder à quoi ressemble votre page dans le browser. A la ligne "Adresse", tapez le chemin d'accès complet de votre fichier. Par exemple :

```
c:\Mes documents\web\exemple.html
```

Et que voyez-vous ? Il y a deux possibilités, soit vous avez un navigateur ancien (pensez à télécharger le dernier) et la page est entièrement grise, soit il est récent et elle est blanche. Rien de plus.

## Vos débuts avec <body>

Vous pouvez essayer maintenant de taper du texte entre les 2 body :

```
<html>

<head>
</head>

<body>
Bonjour, bienvenue sur la page personnelle de Jean Dupont,
qui a été créée il y a peu. Vous m'excuserez du manque
d'informations et de rubriques que je ne tarderai pas à
vous proposer... En attendant, vous trouverez quelques
adresses susceptibles de vous intéresser.
</body>

</html>
```

Et ce texte sera affiché ! Il est vrai qu'il faut un minimum d'efforts avant d'arriver à un résultat. Oui, mais quel résultat ? Ce que vous venez de voir, c'est ce que vous devriez taper sous bloc-notes (n'hésitez pas à changer le texte pour en avoir un plus approprié à ce que vous souhaitez dire).

Je vous proposerai souvent d'essayer l'exemple que vous avez sous vos yeux. Vous pourrez comparer celui-ci avec le votre, pour vérifier et comparer s'il n'y a pas d'erreurs. Dans ces cas-là, vous aurez en dessous de l'exemple un bouton "Essayer !". Je vous conseille d'imaginer dans votre tête ce que cela donnera, pour savoir si vous avez bien compris. Voici le premier bouton (qui s'applique au précédent exemple) :

Normalement, une nouvelle fenêtre s'est affichée, sans la plupart des menus. Vous pouvez l'agrandir si vous le voulez, mais ce n'est pas une obligation. Cela vous permet de différencier ces pages-ci de celles qui servent d'exemples.

Vous avez pu remarquer que le résultat est loin des pages web que vous visitez, mais patience ! Vous serez bientôt surpris de ce que vous aurez fait. Le premier problème de présentation auquel nous allons remédier de suite, est cet effet de "gros pâté" sur cette page. Vous aurez eu beau faire des entrées, mais je vous ai déjà dit que cela ne donne rien.



Il existe deux tags qui vous permettent cela. Le premier donne un retour à la ligne, et le second la création d'un nouveau paragraphe. Il y a ici un mini-tableau qui les présente :

Tag	Introduit	Valeur
 	Retour à la ligne	1 entrée
<p>	Nouveau paragraphe	2 entrées

Mais c'est toujours comme les espaces, vous ne pouvez pas en faire deux. 2 tags <br> à la suite rendront l'effet d'un seul tag <br>.

Les deux tags <p> et <br> ne doivent pas être fermés.

Cela donnera quelque chose de bien plus clair par rapport à notre ancien exemple. Faites de même en parallèle avec votre page perso.

```
<html>
<head>
</head>
<body>
Bonjour, bienvenue sur la page personnelle de Jean Dupont,
qui a été créée il y a peu.<p>Vous m'excuserez du manque
d'informations et de rubriques que je ne tarderai pas à
vous proposer...<br>En attendant, vous trouverez quelques
adresses susceptibles de vous intéresser.
</body>
</html>
```

Notez que j'aurais pu faire une entrée avant chaque tag d'entrée, mais cela n'aurait changé en rien le résultat. C'aurait été pour que je me retrouve plus facilement dans ma page la prochaine fois que je serais venu l'éditer.

Vous pouvez dès maintenant créer une page html.

## Commentaires

Les commentaires peuvent vous être utiles dans certains cas, mais comme ils ne sont pas très importants, je ne m'attarderai pas longtemps dessus.

Un commentaire se trouve dans un tag assez spécial, et le browser ne l'affiche pas. C'est du texte qui peut vous permettre de vous situer (sous bloc-notes) dans une grosse page web.

Le commentaire est le **SEUL** tag qui peut se trouver n'importe où, même dans head ou en-dehors. Comment créer un commentaire ? Comme d'habitude, ouvrez le tag par un <. Tapez juste après un ! et deux -. Tapez votre commentaire après. A la fin de celui-ci, refaites deux - puis fermez le tag par un >. Ouf ! Voyons ce que cela donne :

```
<!-- Rappel : petit problème ici -->
```

En imaginant qu'hier j'ai eu un problème pour créer ma page web, ce commentaire est là pour me rappeler le lendemain d'essayer d'y remédier.



# Le Tag : <head>

Vous venez de voir comment insérer du texte et des entrées. Mais cela se passe dans le tag <body>... Et tout le reste, vous verrez, se trouvera entre le tag d'ouverture et celui de fermeture body !

Mais à quoi peut bien servir ce mystérieux tag <head>. Je vous ai dit qu'il s'adresse plus particulièrement aux moteurs de recherche. C'est vrai. Mais il ne faut pas l'oublier, même si l'internaute ne verra rien de ce qui se trouve dedans.

Assez de bavardages, voyons un peu les tags que l'on trouve dans l'en-tête...

## Les titres

L'élément le plus connu et le plus utilisé de l'en-tête est sans aucun doute le titre. Le titre s'affiche généralement en haut de la fenêtre du navigateur. On le retrouve aussi dans l'historique. Et les moteurs de recherche s'en servent pour donner un titre aux pages qu'ils trouvent.

Bref, ces quelques raisons devraient, je l'espère, vous faire réfléchir un bon moment au titre de votre page.

Chaque page doit avoir un titre pas trop long et si possible différent.

Le tag qui indique un titre est <title>. Ecrivez votre titre après <title> puis fermez le tag par </title>. Par exemple :

```
<html>

<head>
<title>La page perso de Jean Dupont</title>
</head>

<body>
Bonjour, bienvenue sur la page personnelle de Jean Dupont,
qui a été créée il y a peu.<p>Vous m'excuserez du manque
d'informations et de rubriques que je ne tarderai pas à
vous proposer...<br>En attendant, vous trouverez quelques
adresses susceptibles de vous intéresser.
</body>

</html>
```

Vous avez sans doute pu remarquer que le titre s'est affiché en haut de la fenêtre. Pensez aux titres, c'est souvent eux qui déterminent si le site est intéressant ou pas, et par conséquent, ce peut être grâce à lui que vous verrez le nombre de visiteurs augmenter !

## Les tags <meta>

Les tags <meta> sont un autre moyen (plus complexe) qui permet aux moteurs de recherche d'indexer (référencer) votre site. Ils donnent des informations sur l'auteur, sur le programme avec lequel vous avez créé vos pages ou bien des mots clés...

Vous devrez mettre un tag <meta> par type d'information. Ce qui donnera plusieurs même tags à la fin.

Commençons par l'auteur, c'est-à-dire vous. Ajoutez l'attribut name au tag <meta>. Et, comme je vous l'ai déjà expliqué, précisez l'attribut en indiquant "author". Le tag devrait ressembler à ça : <meta name="author">. Ajoutez l'attribut : content. Et précisez cet attribut par votre nom. Jean Dupont écrira :

```
<meta name="author" content="Jean Dupont ">
```

Le moteur de recherche saura et fera savoir que l'auteur de la page web est Jean Dupont.

---

Un deuxième tag <meta> important est celui avec l'attribut name="description". Précisez cette fois-ci content par un petit paragraphe qui résume votre page. Exemple :

```
<meta name="description" content="Voici la page personnelle de Jean Dupont. Je suis vétérinaire. Ma page personnelle vous propose des conseils pour votre animal et vous pouvez m'envoyer des messages en cas d'urgence. Je saurai vous conseiller.">
```

Le moteur de recherche affichera ce texte en guise de présentation de votre site.

Vous pouvez aussi indiquer avec quel programme vous avez créé votre page web. Ici, il s'agit de Bloc-notes. L'attribut name doit avoir pour valeur : "generator". Exemple :

```
<meta name="generator" content="Microsoft Bloc-notes">
```

Un dernier <meta> non moins important : celui qui donne des mots clés. Précisez name="keywords" puis donnez dans l'attribut content une liste de mots clés se rapportant à votre site. Espacez chaque mot par une virgule :

```
<meta name="keywords" content="Dupont, vétérinaire, chien, chat, oiseau, tortue, blessure, urgence, conseils">
```

N'hésitez pas à en mettre bien plus (pas trop quand même) ! Si un internaute tape vétérinaire dans un moteur de recherche, il verra à coup sûr la page de Jean Dupont parmi les autres (il ne reste plus qu'à espérer qu'il choisisse de voir celle-ci).

Enfin, un dernier tag <meta>, qui ne fonctionne que sous Internet Explorer. Il permet d'obtenir des effets de transition entre les pages du site (essayez vous verrez). Voici un exemple :

```
<meta http-equiv="Page-Enter" content="RevealTrans(Duration=2,Transition=12)">
```

Vous pouvez modifier "Duration" : c'est le temps que met la transition pour s'effectuer, en secondes. Quant à "Transition", c'est le type de transition que vous avez choisi : vous pouvez mettre une valeur de 0 à 22. Si vous mettez 23, l'effet sera aléatoire.

Ainsi se termine ce petit chapitre sur le tag <head>. Il existe peut-être un ou deux autres tags qui se mettent là-dedans, mais je vous le ferai savoir en temps voulu.

# Créer Des Liens

Les liens sont très importants en html. On en voit de partout. Et c'est sans doute cela qui différencie le web d'un journal.

Question toute bête : qu'est-ce qu'un lien ? Un lien est le plus souvent un texte assez spécial. Lorsque l'on pointe dessus, la souris devient une main. C'est un des moyens de le reconnaître. Mais ce n'est pas tout ! Un texte qui fait lien est en général bleu et souligné.

Lorsque l'on clique sur un lien, on arrive sur une autre page ou sur une autre partie de la même page.

Et c'est par cela que l'on représente le web.

## Concept relatif - absolu

Pour indiquer où se trouve un fichier cible (celui que l'Internaute voit après avoir cliqué sur un lien), vous aurez 2 moyens...

Le premier est appelé : "Adresse en absolu". Ce mot vous fait peut-être peur, mais il n'y a rien de sorcier là-dedans.

L'adresse du fichier doit être écrite en entier et vous devrez aussi préciser le type de lien (voir plus bas). Sur Internet, on écrira par exemple :

```
http://perso.provider.fr/jdupont/cible.html
```

Si bien sûr, le fichier de destination s'appelle cible.html. Idem sur votre disque dur, cela serait :

```
c:\Site perso\cible.html
```

Ce moyen est précis, mais assez long. Utilisez-le si vous voulez qu'un lien amène à un endroit précis.

Deuxième solution : utiliser une "Adresse en relatif". Ce doit être de loin le moyen le plus utilisé. Il est un peu plus compliqué et plus vague que l'absolu, mais il est plus utilisé car il a quelques avantages (l'adresse devient plus courte). Accrochez-vous aux exemples suivants...

Supposons que le fichier qui contient le lien se trouve dans :

```
c:\Site perso\sommaire.html
```

Et que le fichier cible se trouve dans le même dossier, c'est-à-dire :

```
c:\Site perso\cible.html
```

Alors, dans ce cas-là, on écrira que le lien amène vers :

```
cible.html
```

Tout court ! Maintenant, corsons la chose : sommaire.html ne change pas de dossier, mais cible.html, lui, se trouve dans un sous-dossier. Disons :

c:\Site perso\Dossier\cible.html

En relatif, on a :

Dossier/cible.html

---

**Je vous coupe ici pour vous faire une remarque très importante :**  
**Sur le web, pour séparer les dossiers, on utilise les /. Contrairement à votre P.C. (surtout avec dos) qui utilise les \. Ceci dit, Windows comprend les /.**

---

Reprenons. Autre cas de figure : sommaire.html ne bouge toujours pas, mais cible.html se trouve dans le dossier précédent :

c:\cible.html

On utilise ici les deux points ..

../cible.html

Pensez aussi que vous pouvez tout mélanger : les Dossier/ et les ../ ! Mais en général un site n'est pas aussi compliqué que ça !

C'est donc un concept à connaître. Je vous conseille d'utiliser les "Adresses en relatif". Pourquoi en relatif ? Eh bien, si votre site n'est pas très complexe, cela vous permettra de pouvoir visualiser les pages sur votre disque dur aussi bien que sur le web ! Parce que si l'adresse commence par http://, il faudra que vous soyez connecté pour voir les pages, et réciproquement...

## Types de liens

Vous l'avez vu, les types de liens se rapportent à l'absolu. Des fois, il est nécessaire de créer des liens en absolu, car on n'a pas le choix. Vous avez ici une liste des types de liens (les plus importants) que vous pourrez utiliser, avec leur description.

http://

Ce type de lien amène vers une page html quelque part sur le web.

ftp://

Il est utilisé pour donner l'adresse d'un fichier à télécharger.

telnet://

Adresse d'un serveur Telnet (il faut cependant que le navigateur reconnaisse le Telnet)

gopher://

Donne l'adresse de fichiers sur un serveur gopher.

news :

Amène vers un groupe de news.

mailto:

Sert à envoyer une adresse e-mail à quelqu'un.

Notez qu'il s'agit là de préfixes, et qu'il faut donc donner l'adresse en entier ! Par exemple :

mailto:jdupont@sonprovider.fr

## Créer un lien

Assez bavardé, on y va ! Pour créer un lien, vous devez d'abord écrire le texte sur lequel l'Internaute cliquera. Entourez-le ensuite du tag, on-ne-peut-plus-simple : <a>. Ce qui donne :

```
<a>Cliquez ici !</a>
```

On croirait que l'on a fait exprès de choisir un nom si simple, pour ne pas l'oublier... Cependant, le tag <a> ne peut exister sans attributs ! Il faut au moins préciser l'adresse ou l'endroit de la page.

C'est donc ici se séparent les 2 moyens d'utiliser un lien, le premier amène à une autre page, le second à un autre endroit de la page.

### Le lien amène à une autre page

Ajoutez l'attribut href et précisez-le de l'adresse que vous voulez (vous allez maintenant réutiliser ce que vous avez appris). Par exemple :

```
<a href="http://perso.sonprovider.fr/jdupont/cible.html">Cliquez ici !</a>
```

Ou bien en relatif :

```
<a href="cible.html">Cliquez ici !</a>
```

Vous pouvez tout aussi bien proposer d'envoyer un mail :

```
<a href="mailto:jdupont@sonprovider.fr">M'envoyer un mail</a>
```

Vous voyez toutes les possibilités qui s'offrent à vous !

Ici se séparent les 2 moyens d'utiliser un lien, le premier amène à une autre page, le second à un autre endroit de la page.

### Le lien amène à un autre endroit de la page

Si vous voulez qu'un lien amène à un autre endroit de la page (c'est surtout intéressant si vous avez une grande page), vous n'avez pas besoin de changer l'attribut href. Mais il vous faut d'abord définir l'endroit où le lien amènera. On appelle cet "endroit" une ancre.

Pour faire une ancre, placez-vous à l'endroit désiré et entourez-le du tag <a>. Ajoutez-lui l'attribut name, puis précisez name par le mot de votre choix. Essayez quand même de ne pas faire un grand

mot, et de ne pas insérer de caractères spéciaux.

On obtient alors quelque chose comme ça :

```
<a name="bobo">Vous vous êtes fait mal ?</a>
```

Par contre, on ne voit pas qu'il s'agit d'une ancre, car, contrairement aux liens, il ne se différencie pas du texte normal. Mais votre navigateur en a besoin pour connaître l'endroit où aller.

2° étape : créer le lien. Je vous ai dit que l'on n'a pas besoin de changer le tag href, et c'est vrai. Mais ce qui va changer sera la façon dont vous écrirez l'adresse...

Après le ", mettez directement un #. Juste après, écrivez le nom de l'ancre :

```
En cas de bobo, <a href="#bobo">cliquez ici pour lire le
chapitre traitant de cela.</a>
```

Mieux, vous pouvez combiner les 2 moyens d'utiliser un lien. C'est à dire que le lien amènera à un endroit précis d'une autre page. Exemple :

```
<a href="sommhtml.html#bas">Go !</a>
```

Reprenons un exemple concret avec Jean Dupont. Observez particulièrement le lien vers une adresse e-mail. Après un ? (sans espaces), on ajoute subject=, et on écrit alors le sujet de l'e-mail. Voyez plutôt :

```
<html>
<head>
<title>La page personnelle de Jean Dupont</title>
<!--Quelques tags meta-->
</head>
<body>
Bonjour, bienvenue sur la page personnelle de Jean Dupont,
qui a été créée il y a peu.<p>Vous m'excuserez du manque
d'informations et de rubriques que je ne tarderai pas à
vous proposer...<br>En attendant, vous trouverez quelques
adresses susceptibles de vous intéresser :<p>
<a href=" ../sommhtml.html#bas">Apprenez le html !(bas de
la page)</a>
<br>Vous pouvez <a href="mailto:jdupont@sonprovider.fr
?subject=Je veux te parler">
m'envoyer un mail.</a>
</body>
</html>
```

J'ai mis quelques entrées pour faciliter la lecture de l'exemple.

Vous avez peut-être remarqué que l'adresse : "../sommhtml.html", contient des .. Si vous voulez savoir pourquoi, voilà la réponse : les exemples se trouvant dans un sous-dossier appelé "Exemples", il faut écrire l'adresse avec .. car le sommaire se trouve dans le dossier précédent.

Fin du chapitre sur les liens. Vous allez à partir de maintenant apprendre à embellir votre page, car, pour l'instant, le résultat est plutôt loin des pages web que vous visitez !

# Mettre En Forme Du Texte

Pour le moment, il faut dire que nos pages html ne sont pas très belles, et qu'elles mériteraient mieux. C'est le but de ce chapitre : embellir vos pages html par tous les moyens...

Nous verrons tout d'abord comment agrandir les caractères, puis ensuite vous verrez des attributs de texte (gras, souligné...). Après, vous pourrez aligner du texte et à la fin, vous saurez choisir une police de caractère (mais ce n'est pas très conseillé, vous verrez pourquoi).

Voilà en gros ce que vous saurez faire, et vous verrez que votre page donnera alors bien plus envie d'être lue.

## Taille des caractères

On a 2 moyens de donner la taille des caractères. Ces 2 moyens s'appliquent :

- Soit aux **titres**
- Soit au **texte**

### Taille des titres

On utilise ce moyen pour les titres et son utilisation n'en est pas très complexe.

On a le choix entre 6 tailles pour les titres, et on utilise chaque fois un tag différent. Le tag commence toujours par "h", qui est suivi d'une taille entre 1 et 6 (1 étant la plus grande, 6 la plus petite). Par exemple : <h4>. A partir de maintenant, pour les exemples, je ne mettrai plus tout le corps de la page html. Vous savez que tout se mettra entre les 2 <body> (sauf indication contraire). Regardez cet exemple-ci et essayez-le :

```
<h1>Taille 1</h1><br>
<h2>Taille 2</h2><br>
<h3>Taille 3</h3><br>
<h4>Taille 4</h4><br>
<h5>Taille 5</h5><br>
<h6>Taille 6</h6>
```

### Taille du texte

Pour donner la taille du texte, on utilise le tag <font> (que nous aurons l'occasion de revoir). Ajoutez-lui l'attribut size et précisez-le par la taille de votre choix de 1 à 7 (par contre, ici, 5 est plus petit que 6).

Exemple :

```
<font size="1">Taille 1</font><br>
<font size="2">Taille 2</font><br>
<font size="3">Taille 3</font><br>
<font size="4">Taille 4</font><br>
<font size="5">Taille 5</font><br>
<font size="6">Taille 6</font><br>
<font size="7">Taille 7</font>
```



Si vous désirez définir une taille pour tout le texte (celle par défaut est 2), utilisez le tag <basefont> juste après <body>. Précisez <basefont size="x">, et remplacez x par la valeur de votre choix.

## Explications concernant les attributs de texte

Ci-après seront développés les attributs de texte logiques et physiques. A priori, ces mots ne devraient pas vous évoquer grand-chose. Et peut-être vous font-ils peur (il n'y a pas de honte !) ? N'ayez crainte, voici quelques explications.

Un *attribut de texte logique* laisse au navigateur le choix de la manière avec laquelle il va présenter le texte. On ne dit pas qu'un texte est gras, mais qu'il est "fort". Le navigateur interprètera à sa manière le tag. C'est une vieille méthode un peu démodée, à éviter si vous ne voulez pas avoir de surprises...

Elle peut être cependant intéressante, vous verrez cela plus tard.

Un *attribut de texte physique*, vous l'aurez compris, **impose** au navigateur une certaine mise en forme. Vous serez ainsi certain que votre texte s'affichera en italique par exemple.

## Attributs de texte physiques

Voici un tableau indiquant le nom des tags (que vous ne devez pas oublier de fermer), et l'effet qu'ils donnent :

Tag	Effet
<b>	<b>Gras</b>
<i>	<i>Italique</i>
<u>	<u>Souligné</u>
<sub>	Texte en indice
<sup>	Texte en exposant
<small>	Pour écrire en plus petit que la taille normale
<big>	Pour écrire en plus gros que la taille normale
<s>	<del>Barré</del>

Ce sont les tags les plus employés. Petit exemple :

```
<big>N'oubliez pas !</big><p>
Votre chien peut être <i>malade !</i> Il devrait
être <u>impératif</u> de l'emmener <b>régulièrement</b>
chez un vétérinaire <sub>(comme moi !)</sub>
```

Evitez tout de même de surcharger votre page en attributs de texte. Regardez ce que celle-ci donne :

## Attributs de texte logiques

Voici la liste de la plupart des attributs de texte logiques. Rappelez-vous que le résultat une fois affiché ne sera peut-être pas celui que vous attendez !

Tag	Effet (en général)
<code>&lt;strong&gt;</code>	<b>Mise en valeur forte des caractères (souvent gras)</b>
<code>&lt;em&gt;</code>	<i>Autre mise en valeur (souvent italique)</i>
<code>&lt;cite&gt;</code> ou <code>&lt;blockquote&gt;</code>	<i>Utilisés pour les citations (souvent italique)</i>
<code>&lt;code&gt;</code>	Pour présenter du code source
<code>&lt;samp&gt;</code>	Prévu pour les exemples
<code>&lt;kb&gt;</code>	Présenter des saisies de l'utilisateur
<code>&lt;dfn&gt;</code>	<i>Pour les définitions</i>

Ces tags n'ont plus une grande importance sur le web, alors il vaut mieux les éviter. Mais rien ne vous empêche de les ressusciter !

Autre exemple, qui ne s'affichera sûrement pas de la même manière d'un navigateur à l'autre :

```
Je cite :<cite> 43% des maladies des chevaux sont dûes à
une mauvaise hygiène</cite>
Et c'est<strong> important !</strong><p>
Ajoutez cela sur votre liste des bonnes résolutions
de l'année 3144 : <dfn>Hygiène : Chose importante à
respecter</dfn>
```

## Alignements

Pour aligner le texte, je vous conseille fortement d'utiliser le tag `<div>`, parce qu'il s'applique à tous types d'objets (images, texte, vidéo...). C'est donc pour des raisons de simplicité.

Il faut ajouter au tag `<div>` l'attribut "align" que vous préciserez par le type d'alignement. Les trois types d'alignement sont :

- Gauche : "left"
- Centré : "center"
- Droite : "right"

Exemple :

```
<div align="left">A gauche</div><br>
<div align="center">Centré</div><br>
<div align="right">A droite</div>
```

Noter cependant que les tags `<left>`, `<center>` et `<right>` fonctionnent aussi.

## Polices

Ce doit être votre éternelle question sur le net : "Mais que fait la police ???". Il faut d'abord que vous sachiez ceci :

Tout le monde évite de faire des fantaisies avec les polices. Et c'est pour une raison bien simple : imaginez un seul instant que l'internaute qui lit votre page web ne possède pas les polices que votre site utilise ! Le résultat à l'affichage sera alors tout à fait inattendu car le navigateur choisira une police neutre en remplacement, telle Arial.

Donc, à éviter. Mais le tag existe bel et bien, alors je vous le donne quand même...

On utilise encore le tag <font>, comme pour la taille. Mais cette fois-ci, on ajoute l'attribut "face", que l'on précise par le nom de la police. Exemple :

```
<font face="Mistral">Mauvaise écriture</font><br>
<font face="Signature">TRES mauvaise écriture</font><br>
<font face="Wingdings" size="10">N</font>
```

Vous remarquerez que si vous ne possédez pas une ou plusieurs polices, le texte s'affiche dans une autre police.

Utilisez une de ces 3 polices, beaucoup de monde doit les avoir :

- Arial
- MS Sans Serif
- Times New Roman

Voilà !

Vous venez de voir de nombreux moyens d'embellir votre site. A partir de maintenant, en combinant plusieurs tags, vous devriez pouvoir montrer une page web plus facile à lire et plus claire à tous les internautes. On sentira déjà que vous savez manier le html.

Mais votre site doit manquer de couleurs. Pas de problème ! Le prochain chapitre va colorer votre page web !

## La Couleur Dans Une Page HTML

Quoi ? Votre page est grise ? Il manque des couleurs ? Je vais vous en donner des couleurs ! Et par tous les moyens ! Texte, liens, fond...

Voyez vous-même.

### Comment appeler une couleur ?

C'est une question très importante car on a plusieurs choix. On peut donner une couleur en :

- Donnant la couleur en anglais (vous n'aurez pas beaucoup de choix)
- Indiquant les valeurs rouge-vert-bleu (très précis !)

La première méthode est la plus pratique, mais elle offre moins de choix. Inversement pour la deuxième méthode, elle est moins pratique mais offre énormément de choix (256 X 256 X 256 couleurs, si vous voulez savoir.)

Voici les couleurs en anglais reconnues par la plupart des navigateurs :

Couleur	Aperçu
Black	
Silver	
Gray	
White	
Maroon	
Red	
Purple	
Fuchsia	
Green	
Lime	
Olive	
Yellow	
Navy	
Blue	
Teal	
Aqua	

Pour indiquer une couleur avec les valeurs rouge-vert-bleu, on doit suivre le schéma ci-dessous :

#XXXXXX

Après le #, on donne les valeurs hexadécimales (les X) de la couleur.

En hexadécimal, 0=0, comme d'habitude, 1=1, 2=2, 3=3... 9=9, mais à 10 on met des lettres. Donc 10=A, 11=B, 12=C, 13=D, et ce jusqu'à 15 (F).

Deux X valent une couleur. Par exemple :

#4D708F

Veut dire qu'il y a 4 et 13 de rouge, 7 et 0 de vert, 8 et 15 de bleu.

## Le texte

Pour colorer le texte, on reprend le tag <font> et on lui ajoute le nouveau tag color. Après, vous indiquez la couleur comme décrit plus haut.

```
<font color="#1ADE15">Jolie couleur !</font><br>
<font color="maroon" size="9">Marron !</font>
```

Rien de plus simple, maintenant que vous savez écrire les couleurs !!!

Pour définir une couleur qui sera utilisée sur toute la page, on doit ajouter un attribut au tag <body> (eh oui ! y'en a !) : text. Exemple :

```
<body text="navy">
Bla bla bla...
</body>
```

## Les liens

La couleur des liens doit être définie avec un attribut du tag <body> : link.

```
<body text="navy" link="#0FFF59">
```

```
<a href="page.html">Cliquez ici !</a>
</body>
```

De même, il existe les attributs vlink et alink. vlink donne la couleur des liens visités et alink la couleur des liens activés.

```
<body text="navy" link="#0FFF59" alink="blue" vlink="red">
Re-bla bla bla.
</body>
```

## Le fond

Choisissez bien votre couleur de fond, parce qu'en général, c'est jamais super. Le mieux c'est d'utiliser une image de fond. Mais si vous le voulez, utilisez l'attribut bgcolor du tag <body>.

```
<body bgcolor="#2749E0">
</body>
```

## Exemple récapitulatif

Voici maintenant un exemple qui récapitule tout ce que vous venez de voir.

J'y ai ajouté en plus d'autres tags que vous avez vu, et que vous devriez reconnaître. C'est une page... haute en couleurs !

```
<html>

<head>
<title>Rien que des couleurs !</title>
<meta name="keywords" content="couleurs, flashes, vif, jaune,
vert, rouge, bleu">
<meta name="description" content="Une page très colorée !">
<meta name="generator" content="Bloc-notes">
<meta name="author" content="Mathieu Nebra">
</head>

<body text="green" link="maroon" alink="#14D5FA" vlink="red"
bgcolor="yellow">

<basefont size="4">
<div align="center"><h1><font color="red">FLASH !!!
</font></h1></div>

<p><br>Bienvenue sur la page de Mister color ! Régalez
vos <b>yeux</b> !
<p><br>
<a href=" ../couleur.html">Partir !!!!</a>

</body>

</html>
```

Au moins, on ne se plaindra pas d'une certaine monotonie de votre page !

Maintenant nous allons apprendre à insérer des images dans votre page html, et vous vous demanderez ce que l'on serait sans images !



# Les Images

En général, dans un site web, ce n'est pas l'image qui donne l'information (ou très rarement). C'est plutôt le texte.

Cependant, sans elles, les sites web seraient bien tristes... et bien plus rapides à charger ! C'est pourquoi on a adopté certains formats d'images qui prennent très peu de place (voir "Formats d'images").

Lorsque vous saurez quels formats d'image utiliser, vous pourrez insérer une image (c'est le but de ce chapitre !) mais après, vous pourrez la manipuler à souhait pour en faire un lien.

Je ne vous en dis pas plus...

## Formats d'images

A partir de maintenant, vous pouvez oublier le format bmp ! Certes, le web l'accepte, mais pas les internautes. Sa taille est vraiment énorme comparé aux formats du net. Mais quels sont-ils, au fait ?

Les 2 formats les plus répandus du net sont (retenez-les bien) :

- Le gif
- Le jpeg

Quelles sont leurs particularités ?

### Les gif

Le format gif accepte jusqu'à 256 couleurs seulement, mais c'est le plus utilisé, sauf pour des images nécessitant plus de couleurs (les photos par exemple).

Mais il a des qualités pour être répandu : on peut le rendre transparent (très pratique) ou même l'animer ! De plus, il n'apparaît pas ligne par ligne, mais couche par couche, ce qui veut dire qu'on a l'image tout de suite et que sa qualité s'améliore peu à peu.

### Les jpeg

Les jpeg sont des images utilisées pour les photos, car elles peuvent accepter jusqu'à 16,7 millions de couleurs. On ne peut cependant pas les animer et leur compression diminue nettement la qualité de l'image.

Voilà, c'est tout. Pour animer vos gif, je vous conseille Microsoft Gif Animator (chez [Microsoft](#)), en anglais, mais très simple d'emploi.

# Insérer une image

Retournons à Bloc-notes...

Le tag qui vous permet d'insérer une image est le tag `<img>`. Ce tag ne doit pas être fermé : il n'y en a qu'un qui indique simplement qu'une image est affichée là.

Ajoutez maintenant l'attribut "src".

Précisez src par l'adresse de votre image. Souvenez-vous du chapitre sur les liens. Ici aussi, vous pouvez indiquer l'adresse en absolu ou en relatif (et je vous conseille fortement le relatif, sinon, il faudra que vous ou les internautes soyez connectés au moment du chargement de la page).

Cela donne :

```

```

Rien de plus à dire pour afficher votre image !

Il existe d'autres attributs pour le tag `<img>`, mais facultatifs. Par exemple : "alt".

Précisez alt par le texte de votre choix. Celui-ci sera affiché dans les navigateurs ne lisant pas les images, ou alors lorsque l'on pointe sur l'image.

```

```

Vous pouvez aussi définir la hauteur et la largeur de l'image avec les attributs height et width.

- height définit la hauteur de l'image en pixels.
- width définit la largeur de l'image en pixels.

```

```

Vous pouvez encore donner la taille de la bordure entourant l'image (qui est de 0 par défaut) avec l'attribut border.

```

```

Si vous souhaitez aligner votre image, utilisez comme vous l'avez vu, le tag `<div>`.

```
<div align="center"></div>
```

Enfin, 2 derniers attributs : hspace et vspace.

hspace détermine la distance en pixels entre l'image et le texte à côté.

vspace détermine la distance en pixels entre l'image et le texte au-dessus et en-dessous.

```

```



## Image de fond

Comme pour une couleur de fond, c'est un attribut au tag <body> qu'il faut ajouter. Il s'agit ici de "background". Précisez background par l'adresse situant l'image de fond.

Essayez de créer une petite image, de sorte à ne pas voir la jonction entre chaque copie, puisque le navigateur va copier l'image autant de fois qu'il le faut pour remplir l'écran.

```
<body background="Fond.gif">
```

## Images cliquables

Que peut bien être une image cliquable ? Tout simple : une image qui fait office de lien, on clique dessus et on va sur une autre page.

Pour cela, vous auriez très bien pu le deviner, on prend le tag <a>, que l'on ferme après le tag <img> :

```
<a href="http://www.infonie.fr"></a>
```

Mais le html peut mieux faire : il vous offre la possibilité de définir différents endroits de l'image comme liens. Vous pouvez avoir plusieurs liens dans une image ! Sur la photo d'un paysage, par exemple, le soleil amène au site d'un passionné de l'espace, la poubelle vers un site écolo, les animaux vers la WWF...

Commencez d'abord à mettre le tag <map>. Ajoutez-lui l'attribut name que vous préciserez par un petit mot de votre choix. Pensez à fermer le tag plus loin.

```
<map name="endroit">  
</map>
```

Tout ce que vous allez écrire à présent se trouvera entre ces 2 tags.

Ecrivez maintenant le tag <area> que vous ne devez pas fermer et que vous pourrez répéter autant de fois que vous le désirez. Il indique les parties de l'image qui vont servir de lien.

Vous devez ajouter 3 attributs à chaque tag <area> :

- shape
- coords
- href

L'ordre ci-dessus est à respecter.

Shape définit la forme de la surface cliquable. Il peut avoir comme valeur :

Valeur	Forme
shape="rect"	Un rectangle
shape="circle"	Un cercle
shape="polygon"	Un polygone quelconque

En général on utilise un rectangle, ou si on veut être plus précis, un polygone quelconque. Plus rarement des cercles.

L'attribut coords dépendra de la valeur de shape.

## shape="rect"

Entre guillemets, comme d'habitude, vous indiquez :

```
coords="x1, y1, x2, y2"
```

Remplacez x1, y2... par des valeurs en pixels sachant que :

- x1 = angle supérieur gauche, pixel de gauche
- y1 = angle supérieur gauche, pixel du haut
- x2 = angle inférieur droit, pixel de gauche
- y2 = angle inférieur droit, pixel du haut

Par exemple :

```
<area shape="rect" coords="50, 80, 250, 290">
```

## shape="circle"

Voici le schéma à suivre pour un cercle :

```
coords="x, y, r"
```

Remplacez x, y et r par des nombres en pixels sachant que :

- x = centre, pixel de gauche
- y = centre, pixel du haut
- r = rayon en pixels

Exemple :

```
<area shape="circle" coords="130, 150, 60">
```

## shape="polygon"

Avec un polygone, vous pouvez définir autant de coordonnées que vous le voulez. Respectez seulement le schéma suivant :

```
coords="x1, y1, x2, y2, x3, y3, ....etc"
```

Vous devez définir 3 points minimum, cela fera un triangle.

- x(Nombre) = pixel de gauche d'un point
- y(Nombre) = pixel du haut du même point

Pour finir, ajoutez le troisième tag "href", qui donne l'adresse où amène la surface cliquable. Voilà ce que cela pourrait donner :

```
<map name="ClicImage">
<area shape="rect" coords="45, 26, 210, 168"
href="site1.html">
<area shape="circle" coords="120, 154, 34"
href="site2.html">
<area shape="polygon" coords="50, 66, 110, 102, 232, 231"
href="site3.html">
</map>
```

Ensuite, sortez des tags <map></map> et choisissez l'endroit où vous voulez insérer votre image cliquable. Ajoutez en plus l'attribut "usemap" au tag <img>, que vous précisez par l'attribut "name" du tag <map>, précédé d'un # (#ClicImage, par exemple).

Voilà un exemple qui récapitule ce chapitre, un peu lourd, je l'avoue. Si vous voyez une indésirable bordure bleue autour de votre image, ajoutez l'attribut "border" et affectez-lui la valeur 0.

```
<body text="red">

<map name="ClicImage">
<area shape="rect" coords="65, 50, 75, 83"
href="../couleur.html">
<area shape="circle" coords="40, 54, 20"
href="../bases.html">
</map>

<b>

<div align="center">

</div>
Pensez à regarder et à
essayer de pointer la souris sur l'image pour faire
fonctionner l'attribut alt, à cliquer sur l'image
n'importe où...</b>

</body>
```

Utilisez votre programme de dessin pour connaître les coordonnées des points que vous voulez taper.

Ouf ! C'est fini ! Enfin, juste pour les images... Pour le prochain chapitre, le sujet sera, ou plutôt seront, les puces. En bien plus court et bien plus simple, ne vous inquiétez pas.

## Les Listes

Ce chapitre est assez court comparé au précédent.

Vous allez apprendre à créer des listes à puces, et des listes numérotées. Elles vous seront bien pratiques quand vous devrez énumérer des actions à effectuer, par exemple.

Et surtout, elles mettent bien en valeur ce texte-là, puisque il est décalé par rapport au reste du texte.

Vous allez voir, c'est très simple.

## Liste à puces

Pour indiquer au navigateur où se trouvera votre liste à puces, placez le tag `<ul>` où vous le désirez puis refermez-le plus loin.

Entre ces deux tags se trouvera la liste à puces (ne mettez rien d'autre).

Pour indiquer une nouvelle ligne (avec une puce au début), créez un tag `<li>` et fermez-le plus loin. Répétez autant de fois le tag `<li>` que vous le voulez, entre les deux tag `<ul>`.

```
Sur mon site vous trouverez :<p>
<ul>
<li>Les conseils du professionnel du boomerang</li>
<li>La vie de molière en 46 pages</li>
<li>Les liens intéressants</li>
<li>Une adresse super : Apprenez le html !</li>
</ul>
```

Notez que vous n'avez pas besoin d'écrire de tag `<p>` ou `<br>` à la fin de chaque ligne.

Pensez aussi que vous pouvez combiner des tags, ce n'est pas interdit ! Essayez par exemple de faire un lien pour chaque ligne de la liste, ou bien de tout mettre en gras...

Malheureusement, le html n'offre pas autant de puces qu'un traitement de textes...

## Liste numérotée

Si il y a un ordre dans vos actions, une préférence... utilisez une liste numérotée.

A chaque ligne le numéro du début augmente d'un.

Pour ce faire, tapez le tag `<ol>` et refermez-le plus loin. Définissez chaque ligne avec le tag `<li>`, comme tout à l'heure.

```
Ma vie le dimanche :<p>
<ol>
<li>Je me lève</li>
<li>Je mange une tartine</li>
<li>Je retourne au lit</li>
</ol>
```

### Liste numérotée avec des chiffres romains

Ajoutez un attribut au tag `<ol>` : "type". Précisez-le par un i ou un I, c'est tout.

### Liste numérotée avec des lettres

Précisez l'attribut type par un A ou un a, selon votre choix.

### Commencer une liste numérotée à partir de...

Il faut un autre attribut au tag `<ol>` : "start", que vous préciserez par le chiffre de votre choix.

Exemple :

```
Au choix dans le menu dégustation :<p>
<ol type="A">
<li>Sauce de crapaud baveux</li>
<li>Intestins de cheval avec un coulis de sang frais</li>
<li>Zombie du jour (déconseillé aux enfants)</li>
</ol>
```

Et voilà ! Elles vous seront très pratiques, les listes, vous verrez.

Prochain arrêt : les tableaux. Accrochez-vous bien, il y a pas mal de choses à dire dessus.

## Les Tableaux

Quoi ? Vous croyez que les tableaux ne serviront à rien pour votre page ? Détrompez-vous !

Au contraire, ils ont une grande importance. Si vous regardez bien, beaucoup de sites contiennent des tableaux.

Ils offrent une mise en page particulière, et vous pouvez tout mettre dedans !

Du texte, mais aussi des images, des vidéos, du son... tout !

Vous vous rendrez compte de leur utilité lorsque vous ferez votre page.

Ceci dit, si vous voulez faire un petit tableau simplet, ça ne sera pas difficile. Mais dès que vous en voudrez plus, vous devrez combiner bon nombre de tags et d'attributs...

## Créer un tableau

Là, c'est très facile. On utilise le tag `<table>` que l'on referme plus loin avec `</table>`. Ce tag délimite l'emplacement du tableau.

```
<table>
</table>
```

Vous devez savoir qu'en html, un tableau se construit ligne par ligne. Pour indiquer une ligne, on utilise le tag `<tr>`, refermé avec `</tr>`.

C'est dans ce tag que l'on indiquera les cellules (cases du tableau).

Si vous faites un tableau à 4 lignes, le code en html devrait ressembler à ça :

```

<table>

<tr>
</tr>

<tr>
</tr>

<tr>
</tr>

<tr>
</tr>

<tr>
</tr>

</table>

```

Ensuite, entre les `<tr>...</tr>` vous devez définir les cellules.

On distingue 2 sortes de cellules :

- Les cellules d'en-tête, définies avec `<th>...</th>`
- Les autres cellules, contenant les données, définies avec `<td>...</td>`

Ainsi, normalement, la première ligne contiendra des `<th>` et les autres des `<td>`.

Exemple :

```

<table>

<tr>
<th></th>
<th></th>
</tr>

<tr>
<td></td>
<td></td>
</tr>

<tr>
<td></td>
<td></td>
</tr>

<tr>
<td></td>
<td></td>
</tr>

</table>

```

Vous auriez dû comprendre qu'il s'agit d'un tableau 2 X 4 : 2 cellules en largeur et 4 en longueur.

Et c'est entre ces tags `<th>` ou `<td>` que vous devrez écrire les informations qui seront affichées.

Ouf !

Et pourtant c'est le strict minimum, comme vous le verrez plus loin.

Vous n'êtes pas obligé de mettre du texte entre `<td>` et `</td>`. Vous pouvez mettre le tag `img` que vous connaissez, et il y aura une image dans la cellule !

Petit exemple pour résumer :

```

<div align="center">Prix des bus de ville :<p>
<table>

<tr>
<th>Age de la personne</th>
<th>Prix moyen TTC</th>
</tr>

<tr>
<td>Moins de 3 ans</td>
<td>4 francs</td>
</tr>

<tr>
<td>Entre 3 et 14 ans</td>
<td>8 francs</td>
</tr>

<tr>
<td>Plus de 14 ans</td>
<td>10 francs</td>
</tr>

</table>
</div>

```

Et là, Ô surprise(s) ! Il n'y a pas de bordures, pas de cadres ! Pas joli joli...

Par contre vous avez sans doute remarqué que la première ligne était mise en valeur (souvent en gras).

Bon, c'était relativement simple jusqu'ici, mais je crois que vous ne pouvez pas vous arrêter là, il vous faudrait au moins une bordure...

## Bordures et quadrillage

Je vous propose une solution très simple pour avoir une bordure. Il s'agit d'un attribut au tag `<table>`. Il s'appelle "border".

Ce qui donne :

```

<table border>

</table>

```

Pas besoin de préciser border si la taille de la bordure vous convient (en général c'est bon).

Mais si vous en voulez une plus petite ou plus grande, vous pouvez préciser border comme ceci :

```

<table border="5">

</table>

```

Cela ne marche que pour le contour du tableau. Pour modifier l'épaisseur du quadrillage, utilisez un autre attribut avec `<table>` que vous devrez impérativement préciser : "cellspacing".

Exemple :

```

<table border="5" cellspacing="6">

</table>

```



En général on utilise seulement border et on ne le précise pas. Je reprends l'exemple de tout à l'heure et j'y ajoute l'attribut border. Regardez la différence :

## Définir la hauteur et la largeur des tableaux

Si vous désirez vraiment définir une hauteur et une largeur pour votre tableau, ces lignes sont pour vous. Mais elles ne sont pas vitales.

Si vous avez une grande image dans une cellule, alors peut-être que vous désirerez réduire la taille du tableau qui a été soudainement agrandi.

Vous devrez ajouter les attributs height et/ou width au tag <table>.

- Height définit la hauteur du tableau
- Width définit la largeur du tableau

Précisez ces attributs par un pourcentage, comme ceci :

```
<table border height="34%" width="50%">
```

Ce tableau avec bordure utilise 34% de la fenêtre de l'explorateur en hauteur et 50% en largeur.

## Préciser une distance minimale entre le bord d'une cellule et son contenu

Si vous voulez donner un effet de "centré" à votre tableau, rien de tel que de déplacer le texte qui s'y trouve à l'intérieur.

Pour ce faire, vous allez devoir ajouter (encore !) un attribut au tag <table> : cellpadding. Précisez cellpadding par un nombre en pixels (plus question de pourcentage). Exemple :

```
<table border cellpadding="12" width="75%">
<tr>
<th>C'est juste...</th>
<th>... un essai</th>
</tr>
</table>
```

## Ombres et couleurs d'un tableau

### Couleur de fond

Pour définir la couleur de fond d'un tableau, il faut utiliser l'attribut bgcolor du tag <table>.

On le précise par une couleur en hexadécimal ou en anglais (voir le chapitre sur les couleurs).

Choisissez une couleur plutôt claire pour le tableau, de préférence.

Voici un exemple :

```
<table border bgcolor="aqua">
```



## Bordure

Pour déterminer la couleur des bordures et du quadrillage, on utilise l'attribut bordercolor (toujours du tag table) :

```
<table border bordercolor="red">
```

Ceci dit, l'effet donné n'est plus très joli. Vous pouvez oublier cet attribut...

## Ombrage

Par contre ceci est plus intéressant. Définissez vous-même la couleur claire et la couleur foncée de l'ombre avec bordercolordark et bordercolorlight.

- bordercolorlight définit la couleur la plus claire de l'ombre
- bordercolordark définit la couleur la plus foncée de l'ombre

Ces attributs vont toujours ensemble. Si vous mettez l'un, vous devriez mettre l'autre.

D'autre part, évitez les couleurs farfelues qui ne rimeraient à rien, pour préserver l'effet d'ombrage.

Enfin, et ce n'est pas le moins important, vous devriez préciser l'attribut border dans ce cas, pour faire une grosse bordure, disons border="10". Sinon ces attributs bordercolorlight et bordercolordark ne prendraient pas effet.

Exemple :

```
<table border="9" bgcolor="silver" bordercolorlight="navy" bordercolordark="teal">
```

```
<tr>
<th>Est-ce que cet alliage</th>
<th>de couleurs</th>
<th>vous plaît ?</th>
</tr>
```

```
</table>
```

Voilà pour les tableaux. Vous comprenez maintenant ce que je vous ai dit au début du chapitre : un tableau simple, ce n'est rien, mais quand on veut faire mieux...

Le chapitre suivant traite des formulaires. Autant vous prévenir tout de suite, c'est de loin le plus difficile.

Mais si vous vous accrochez solidement, vous pourrez faire des trucs super !

# Les Formulaires

Tôt ou tard vous serez amené à faire un formulaire pour votre site web. Pourquoi, et dans quel but ?

Ne vous est-il jamais arrivé de remarquer que votre site manque un peu de mouvements ? Parce qu'en fait, un site web, au départ, c'était fait pour lire et obtenir des informations.

Mais on peut faire bien mieux.

On peut ajouter quelques ingrédients qui demandent à l'internaute d' "agir".

Ces ingrédients s'appellent : les formulaires.

Ainsi, votre site aura des éléments des programmes informatiques : cases à cocher, boutons, zones pour écrire, listes déroulantes, boutons d'options...

En général, par la suite, on clique sur un bouton "Envoyer", ce qui déclenche le traitement des données que l'utilisateur a entrées, et leur envoi sur le site, sur votre courrier électronique...

Voilà en gros à quoi consiste un formulaire.

Quelques exemples, pour vous montrer leur utilité :

- Un questionnaire sur l'internaute, ses habitudes et ce qui lui plaît ou pas sur votre site (très fréquent).
- Un vote pour un jeu, une idée, un site ou pour tout autre chose.
- Une commande pour votre logiciel ou un de vos CD dont vous voulez vous débarrasser.

A vous d'imaginer tout ce qu'un formulaire pourrait apporter à votre site.

Cependant, évitez de poser des questions qui ennuieraient l'internaute : votre âge, votre adresse, votre numéro de carte bancaire... Essayez de préserver l'anonymat de cette personne.

Mais bien sûr, tout ce qui est intéressant est difficile à faire. Sachez que, si vous avez bien suivi (et compris) tous les chapitres de Apprenez le HTML, vous n'aurez pas trop de problèmes. Sinon, accrochez-vous un peu, mais je pense que vous pouvez y arriver quand même.

## Créer un formulaire

Définissez l'endroit où se trouvera votre formulaire. Utilisez pour cela le tag `<form>`. Refermez-le plus loin, comme ceci :

```
<form>
</form>
```

Evidemment, ce tag a besoin d'attributs. Le premier est : "action". On le précise par le chemin d'accès du programme qui va traiter le formulaire.

Comme je vous l'ai déjà dit, il faut que ces données soient traitées à partir d'un programme. La meilleure solution consiste à créer un script CGI, comme nous le verrons plus loin. Si vous souhaitez que la réponse au formulaire vous soit directement envoyée par mail, écrivez par exemple (en remplaçant cette adresse par la votre) :

```
<form action="mailto:jdupont@wanadoo.fr">
</form>
```

Le deuxième attribut est "method". On le précise par GET ou POST. Lequel choisir ?

- GET permet d'envoyer des données ne dépassant pas 1Ko, ce qui est peu.
- POST est plus pratique car non limité. Il vous sera utile si vous envoyez les données par mail.

Conclusion, si vous hésitez, optez pour POST, comme ceci :

```
<form method="post">
</form>
```

## Zones de saisie

A l'intérieur du tag `<form>`, vous allez définir votre formulaire. Vous pouvez bien sûr mettre des phrases et utiliser des tags pour mettre en forme du texte.

Vous avez plusieurs moyens pour demander des informations à l'internaute. Un des plus fréquents est la zone de saisie.

Une zone de saisie ressemble à ça :

### Zone de saisie monoligne

Ecrivez le tag `<input>`. Il ne doit pas être refermé. Comme pour les images, il sert juste à dire qu'il y a quelque chose là.

Précisez l'attribut "name" par le nom que vous voulez donner à cette zone de saisie. Cela ne sera pas affiché à l'écran, mais vous en aurez besoin pour traiter les données. Donnez un nom représentatif de préférence.

Exemple :

```
<form action="mailto:jdupont@wanadoo.fr" method="post">
<input name="nomduchat">
</form>
```

Il vaut mieux éviter les majuscules, les espaces et autres caractères.

Vous devez aussi donner d'autres informations obligatoires.

Il y a l'attribut "size" que l'on précise par la longueur de la zone de saisie. "maxlength", quant à lui, indique le nombre maximum de caractères autorisés.

Petit exemple :

```
<form action="mailto:jdupont@wanadoo.fr" method="post">
Cette zone de saisie est de taille 6 et est limitée à 6
caractères : <input name="nomduchat" size="6" maxlength="6">
<p>
Celle-ci est de taille 6, mais d'une longueur maximale de
4 caractères : <input name="nomduchien" size="6"
maxlength="4">
<p>
Celle-là est de taille 6 est limitée à 9 caractères :
<input name="nomduperroquet" size="6" maxlength="9">
</form>
```

Vous pouvez aussi demander à ce que l'internaute respecte certaines conditions. Imaginez que demandiez une adresse e-mail, et qu'on vous renvoie quelque chose du type : "Blabla"...

Heureusement, un attribut, non obligatoire celui-ci, va vous venir en aide. Il s'appelle "type". Ce petit tableau vous explique comment l'utiliser :

Valeur de "type"	Description
int	Un nombre entier doit être entré.
float	Un nombre décimal doit être entré
date	Une date doit être entrée. Exemple : 07/10/96
url	Une adresse e-mail doit être entrée. Exemple : jdupont@wanadoo.fr
password	Le texte entré est masqué par des * : <input type="password"/>

Si vous le désirez, vous pouvez utiliser aussi les attributs "max" et "min". Ils permettent de définir respectivement le plus grand nombre et le plus petit qui puisse être entré. A utiliser avec l'attribut type="int" ou type="float".

## Zone de saisie multiligne

Le tag à utiliser pour une zone de saisie à plusieurs lignes est différent de ce que nous venons de voir.

Il s'agit de <textarea>. Pensez à mettre l'attribut "name" pour le nommer.

Pour la taille, utilisez les attributs "rows" et "cols". Ils indiquent le nombre de lignes et de colonnes.

Mais attention ! Ce tag doit se refermer plus loin ! Qu'écrit-on entre les 2 ? Le texte à afficher par défaut. Ce que l'internaute voit dans la zone de texte au départ. Si vous ne voulez rien y mettre, n'écrivez rien.

Exemple :

```
<textarea name="votrevie" rows="5" cols="13">
Tapez votre texte ici.
</textarea>
```

## Listes d'options

Si l'internaute doit faire un seul choix parmi une liste, je vous conseille d'utiliser les listes d'options. Elles ressemblent à ceci :

Comme une telle liste contient plusieurs choix, on doit indiquer d'abord que l'on fait une liste là et dire où elle s'arrête.

Utilisez le tag <select>, avec l'attribut "name" (c'est obligatoire, il faut le nommer). Refermez ce tag plus loin, comme ceci :

```
<select name="maliste">
</select>
```

Pour créer une option, mettez le tag `<option>` dans le tag `<select>` et refermez-le plus loin. Entre le `<option>` et le `</option>`, tapez le texte à afficher.

Exemple :

```
Ville :  
<select name="maliste">  
  
<option>Paris</option>  
<option>Avignon</option>  
<option>Marseille</option>  
<option>Lille</option>  
<option>Lyon</option>  
  
</select>
```

Vous pouvez aussi définir une taille précise pour la liste d'options, pour donner cet effet :



Ajoutez l'attribut `size` au tag `<select>`, et précisez-le par sa taille, comme ceci :

```
<select name="maliste" size="5">
```

Si vous souhaitez que l'on puisse sélectionner plusieurs options, ajoutez l'attribut `"multiple"` au tag `<select>` :

```
<select name="maliste" size="5" multiple>
```

Ainsi on peut sélectionner plusieurs options en maintenant les touches CTRL ou Maj enfoncées :



## Boutons d'option et cases à cocher

Toutefois, les éléments les plus utilisés dans les formulaires (ils sont plus pratiques) sont :

- Les boutons d'option :  Option 1  Option 2
- Et les cases à cocher :  Choix 1  Choix 2

En cliquant un peu partout vous vous êtes sûrement rendu compte que ces deux éléments fonctionnent différemment. On ne peut sélectionner qu'un bouton d'option, tandis que l'on peut choisir pour chaque case à cocher sa valeur (cochée ou non).

Commençons par les boutons d'option...

### Les boutons d'option

Pour définir un bouton d'option, on utilise aussi le tag `<input>`. Pour le différencier des zones de saisie par exemple, on doit lui ajouter l'attribut `"type"` et le préciser par `"radio"`, comme ceci :

```
<input type="radio">
```

Le navigateur sait alors à quoi il a affaire. Mais vous n'êtes pas encore tiré d'affaire ! Vous devez rajouter encore 2 attributs.

Le premier, c'est "name". Vous le connaissez, mais attention ! Ici ça ne fonctionne pas pareil. Ce n'est pas un nom unique que l'on doit donner, mais le nom du groupe de boutons auquel il est relié. Petite explication...

Monsieur,  Madame

Marié,  Célibataire

Voyez par vous-même. On a le choix entre Monsieur, Madame, Marié ou Célibataire. Mais vous devez sélectionner un bouton d'option par ligne, alors qu'ici si vous avez cliqué sur Monsieur puis sur Marié, Monsieur se désélectionne ! Il vous faut un groupe dans chacune des lignes. D'où l'intérêt de créer des groupes de boutons. Voici le résultat :

Monsieur,  Madame

Marié,  Célibataire

Donnez donc la même valeur à l'attribut "name" pour chaque bouton d'option d'un même groupe.

Cependant, tout ceci n'enlève pas le fait que chaque bouton d'option doit avoir un nom unique. Utilisez l'attribut "value" que vous préciserez par le nom du bouton.

Petit exemple :

```
<form>
<input type="radio" name="Sexe" value="Monsieur">Monsieur,
<input type="radio" name="Sexe" value="Madame">Madame<p>
<input type="radio" name="Situation" value="Marié">Marié,
<input type="radio" name="Situation" value="Célibataire">
Célibataire
</form>
```

Ah ! Et un dernier petit truc... Si vous voulez qu'un bouton d'option soit présélectionné, ajoutez-lui l'attribut "checked". Vous n'avez pas besoin de le préciser.

## Cases à cocher

Ici on va aller plus vite, vous savez déjà le principal.

Une case à cocher se crée avec le tag `<input>` et l'attribut "type=checkbox" :

```
<input type="checkbox">
```

On doit aussi créer des groupes de cases à cocher définis avec l'attribut "name" comme pour les boutons d'option. Cela ne change rien de spécial à la vue de l'internaute, car une case à cocher n'obéit pas aux mêmes lois qu'un bouton d'option. Vous aurez pourtant besoin de connaître les noms des groupes pour classer, organiser les données lorsque vous les recevrez.

Donnez un nom différent à chaque case à cocher avec l'attribut "value", ajoutez l'attribut "checked" si vous voulez que la case à cocher soit cochée dès le départ et c'est gagné !

Un exemple ?



```

<form>
Cochez les aliments que vous aimez ici :<p>
<input type="checkbox" name="AlimentsBons" value="Frites">
Frites,
<input type="checkbox" name="AlimentsBons" value="Fromage">
Fromage,
<input type="checkbox" name="AlimentsBons"
  value="Hamburger">Hamburger,
<input type="checkbox" name="AlimentsBons" value="Viande">
Viande.
<p>Cochez les aliments que vous n'aimez pas :<p>
<input type="checkbox" name="AlimentsPasBons"
  value="Epinards">Epinards,
<input type="checkbox" name="AlimentsPasBons"
  value="Lézard">Lézard,
<input type="checkbox" name="AlimentsPasBons"
  value="Escargots">Escargots.
</form>

```

## Boutons de commande

Utilisés en général pour commander l'envoi du formulaire ou sa réinitialisation, les boutons de commande ressemblent à ceci :



Pour faire un tel bouton, vous utiliserez (encore !) le tag `<input>`, avec l'attribut "type" que l'on pourra préciser de 4 manières :

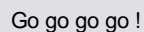
- `type="submit"` : commande l'envoi du formulaire
- `type="reset"` : remet à zéro le formulaire
- `type="button"` : ne fait rien de spécial, attend une autre instruction (script...)

Quelle que soit la valeur choisie, cela donne un joli bouton de commande.

Toujours la propriété "name" à définir, et un attribut "value" qui indique cette fois ce qui est affiché :

```
<input name="BoutonFun" type="submit" value="Go go go go !">
```

Cela donne :



## Envoi des données

Ouf ! Enfin ! On va pouvoir récupérer les données.

Revenons au tag `<form>` et revoyons ses attributs :

```
<form action="mailto:jdupont@wanadoo.fr" method="post">
</form>
```

C'est le schéma que je vous conseille d'utiliser (en remplaçant bien sûr `jdupont@wanadoo.fr` par votre adresse).

Vous recevrez chez vous un mail et son contenu dépendra du browser qui l'a envoyé. Avec IE5 par exemple, vous recevez un mail avec comme objet "Formulaire posté avec Microsoft Internet Explorer." et un fichier joint (postdata.att) que vous pourrez ouvrir avec bloc-notes.

Et vous vous apercevrez alors que tout n'est pas rose. En effet, le message est plutôt... codé. Certains caractères sont remplacés par d'autres.

La réponse fonctionne comme ceci :

NomDeL'objet=Valeur&NomDeL'Objet=Valeur&NomDeL'Objet=....

Vous comprenez maintenant l'intérêt de bien choisir le nom des objets du formulaire que vous intégrez ?

Référez-vous à ce petit tableau qui vous donne quelques correspondances :

Si vous voyez...	... cela signifie :
%2B	+
%3D	=
%26	&
%27	'
+	Espace

Ce n'est pas évident comme vous pouvez le constater. Mais avec un peu d'habitude on s'y fait. L'envoi a été commandé par le bouton et la propriété : type="submit". Le message sera envoyé suivant l'instruction qui se trouve dans l'attribut "action" du tag <form>. Il n'y a rien d'autre à ajouter...

Vous pourrez donc finalement comprendre ce que l'internaute a rempli dans son formulaire. Si vous voulez obtenir quelque chose de plus clair (et de plus complexe évidemment !), vous devez avoir recours au CGI. Voyez plus bas.

## Le CGI

On va finir tranquillement ce gros chapitre, un peu fastidieux je le reconnais.

Tout d'abord, le CGI (Common Gateway Interface) est un petit langage de programmation qui va vous permettre de ranger, classer et organiser les réponses reçues. C'est tout de même quelque chose qui dépasse ce site exclusivement consacré au HTML.

Je ne peux donc que vous renvoyer vers quelques adresses pour que vous puissiez vous renseigner. Il faudra apprendre à le programmer, et ce n'est pas toujours chose facile...

Solution ! Utiliser des scripts CGI tous prêts ! A condition de parler un minimum d'anglais, of course...

Je vous propose ces adresses :

- <http://www.perl-gratuit.com> (français)
- <http://www.cgi-resources.com>
- <http://www.worldwidemart.com/scripts>

Bonne pêche !  
Et bien voilà... Les formulaires n'ont plus de secrets pour vous.

Vous trouverez dans le chapitre suivant tout ce qu'il faut pour faire de jolis frames. Qu'est-ce que c'est ? Surpriiise !

## Les Frames

L'utilité des frames est incontestable. Grâce à eux vous pouvez séparer une fenêtre du navigateur en 2, 3, 4... Même si certains en abusent, je vous recommande vivement d'en mettre un dans votre page perso. L'internaute pourra ainsi accéder plus facilement et plus rapidement aux informations qu'il cherche.

Peut-être que ce terme ne vous dit rien, alors cliquez sur ce bouton pour voir à quoi ressemble un frame :

Sachez qu'il est tout à fait possible de mettre plus de frames que ça, mais cela risque d'encombrer votre site.

Il est possible que votre navigateur soit trop ancien et ne connaisse pas les frames. En effet, les frames ne datent pas du début du HTML, donc si vous avez un vieux navigateur, vous êtes cuit et un texte vous le fera savoir.

La meilleure solution ? Vous procurer une version plus récente de votre navigateur !

Déclarons un frame !

## Insérer un frame

Il vous faudra créer autant de fichiers HTML qu'il n'y a de frames, sans compter le fichier qui fait appel aux frames.

Pour réaliser l'exemple que vous avez vu tout à l'heure, il m'a fallu un fichier pour appeler les frames, puis un fichier par frame. Ce qui fait 3 fichiers en tout !

Ca complique un peu les choses...

Pour commencer, on va s'occuper du fichier qui fait appel aux frames...

Ce fichier ne contiendra pas beaucoup d'informations. Vous pouvez à la rigueur faire l'en-tête de votre page avec <head> et y mettre le tag <title> pour le titre des 2 pages, ainsi que des méta-tags, mais c'est tout.

Nouveauté ! Il n'y a pas de tag <body> ! La page va ressembler à ça :

```

<html>

<head>
<title>Le titre de ma page avec frames</title>
</head>

<!-- Et ici, les appels aux frames, non entourés du tag
body...-->

</html>

```

Bien. Première question : combien de frames voulez-vous, et dans quel sens ? En effet, les frames peuvent être orientés de manière horizontale ou verticale. Dans les deux cas, utilisez le tag `<frameset>`. Ensuite, accompagnez-le d'un de ces deux attributs :

- Pour des frames orientés verticalement, comme dans l'exemple de tout à l'heure, prenez **"cols"**
- Sinon, pour des frames orientés horizontalement, prenez **"rows"**

Vous devrez ensuite donner la taille des frames. Il faut préciser l'attribut que vous avez choisi par une série de pourcentages, séparés pour chaque frame d'une virgule :

```
<frameset cols="18%,51%,31%">
```

Cela donnera 3 frames : le premier d'une taille correspondant à 18% de la fenêtre du navigateur, le second à 51% et le dernier à 31%. C'est simplissime, comme vous pouvez le voir.

Fermez le tag `frameset` plus loin, pour faire :

```
<frameset cols="18%,51%,31%">
</frameset>
```

Nous allons indiquer quelles sont les pages HTML correspondant à chaque frame entre les deux `<frameset>`.

Pour chaque page vous devez écrire un tag `<frame>`.

Puis ajoutez à ce tag l'attribut `"src"`, précisé du chemin d'accès au fichier HTML ; ainsi que l'attribut `"name"` pour donner un nom quelconque à votre frame :

```
<frame src="frame1.html" name="gauche">
<frame src="frame2.html" name="milieu">
<frame src="frame3.html" name="droite">
```

Et voilà qui est fait ! Vous n'avez plus qu'à créer les fichiers `frame1.html`, `frame2.html` et `frame3.html`, et la page sera complète !

Je vais juste vous donner quelques autres infos pour l'attribut `"cols"` ou `"rows"`.

Premièrement, vous n'êtes pas obligé de mettre des pourcentages ; si vous mettez juste des nombres, cela indiquera au browser un nombre de pixels.

Et deuxièmement, si vous mettez une `*` à la place d'un nombre, cela indiquera au browser que ce frame prend toute la place qui reste. Si le premier frame est large de 20% et que vous mettez une `*`, le deuxième frame sera large de 80%.

Pensez aussi à ceux qui ont des navigateurs qui ne connaissent pas les frames. Après le `</frameset>`, écrivez le tag `<noframes>` et refermez-le plus loin. Tapez un petit texte à l'attention de ceux qui ne peuvent pas voir de frames. Ce texte ne sera pas affiché si les frames fonctionnent correctement.

Sachez que les frames peuvent être redimensionnés à la guise de l'internaute. Si vous voulez empêcher ceci, il vous faut ajouter l'attribut "noresize" aux tags `<frame>`. Exemple :

```
<frame src="frame1.html" name="gauche" noresize>
<frame src="frame2.html" name="milieu" noresize>
<frame src="frame3.html" name="droite" noresize>
```

Finissons avec un petit exemple pour récapituler :

```
<html>

<head>
<title>Ma page avec des frames !</title>
</head>

<frameset cols="20%,*">
<frame src="11b1.html" name="gauche">
<frame src="11b2.html" name="droite">
</frameset>

<noframes>
Si vous pouvez lire ces lignes, c'est que votre navigateur
ne supporte pas les frames.
</noframes>

</html>
```

Je n'ai pas mis ici le code des pages 11b1.html et 11b2.html, simplement parce que vous savez déjà les réaliser.

## Les liens dans un frame

Jusqu'ici, les frames ne devraient pas vous avoir causé trop de problèmes, je pense. Ici, nous allons parler des liens dans une page qui contient des frames. En effet, comme vous avez pu le constater dans l'exemple précédent, c'est pas joli-joli si on clique sur un des liens du frame de gauche.

Il est possible d'indiquer vers où et comment le lien doit amener. On aurait préféré que les liens amènent tous dans le grand frame, non ?

Il faut rajouter un attribut au lien dont on veut modifier la cible. Il s'agit de "target" qu'il faut préciser par un de ces mots, selon votre choix :

Si l'on veut que...	Il faut préciser "target" par :
La page s'affiche dans le même frame que le lien	_self
La page s'affiche dans l'autre frame (s'il n'y en a pas une nouvelle fenêtre est créée)	_parent
La page s'affiche dans une nouvelle fenêtre du navigateur	_blank
La page s'affiche dans la même fenêtre du navigateur, cette fois sans frames	_top
La page s'affiche dans un frame précis de votre site	Le nom du frame

Voici 2 exemples. Le premier lien affiche la page cible dans une nouvelle fenêtre, et le second dans le frame appelé "toto".

```
<a href="cible.html" target="_blank">Cliquez</a>
```

```
<a href="cible.html" target="toto">Cliquez</a>
```

Et si tous les liens d'une même page sont destinés à amener au même endroit, vous pouvez écrire dans l'en-tête de votre page, c'est à dire entre les 2 <head>, le tag <base> avec l'attribut "target" comme vous savez maintenant l'utiliser :

```
<head>
<base target="_top">
</head>
```

Concluons ce chapitre avec cet exemple :

Code de la page qui appelle les frames :

```
<html>

<head>
<title>Des frames et des liens</title>
</head>

<frameset rows="50%,*">
<frame src="11c1.html" name="unframe">
<frame src="11c2.html" name="deuxframes">
</frameset>

<noframes>
Si vous pouvez lire ces lignes, c'est que votre navigateur
ne supporte pas les frames.
</noframes>
</html>
```

Code du frame du haut :

```
<html>

<head>
</head>

<body>
<a href="../bases.html" target="_self">Pour revoir les
bases du html dans ce frame</a><p>
<a href="../bases.html" target="deuxframes">Pour revoir les
bases dans l'autre frame.</a><p>
<a href="../bases.html" target="_blank">Pour revoir les
bases dans une nouvelle fenêtre.</a><p><br>
<a href="11c2.html" target="deuxframes">Pour restaurer la
page de départ du frame du bas.</a>
</body>

</html>
```

Code du frame du bas :

```

<html>

<head>
</head>

<body>
<a href="../sommhtml.html" target="_self">Pour voir le
  sommaire dans ce frame</a><p>
  /sommhtml.html" target="unframe">Pour voir le
  sommaire dans l'autre frame.</a><p>
<a href="../sommhtml.html" target="_blank">Pour voir le
  sommaire dans une nouvelle fenêtre.</a><p><br>
<a href="11cl.html" target="unframe">Pour restaurer la
  page de départ du frame du haut.</a>
</body>

</html>

```

Vous devriez avec ce dernier exemple pouvoir manier les frames à la perfection !

Objet du prochain chapitre : les scripts, ou comment faire des trucs et des bidules dignes d'un programme dans une page web.

## Les Scripts

Toutes les bonnes pages ont maintenant recours aux scripts, pour améliorer le confort des visiteurs et pour ajouter un petit "plus" interactif qui fera que votre page ne sera vraiment pas comme les autres.

Il n'est, certes, pas évident de manier ces choses-là, mais dès qu'on maîtrise un peu les scripts, on peut aller très très loin...

Assez bavardé. Vous ne voyez peut-être pas ce que les scripts peuvent donner à votre page... Et pourtant leurs possibilités sont infinies !

Je vous laisse donc imaginer toutes les possibilités !!! Voyons voir maintenant comment insérer un script dans sa page HTML.

## Insérer un script

Tout d'abord, il faudrait savoir où insérer votre script. Ce n'est pas toujours la même chose, ça dépend du script que vous utilisez. Dans la majorité des cas, vous pouvez le mettre n'importe où dans votre code HTML (entre les 2 <body>, bien sûr).

Le tag pour insérer un script n'est pas dur à retenir, c'est <script>. Vous devez le refermer plus loin, de manière à délimiter la zone du script, comme ceci :

```

<script>
<!--Ici se trouvera votre script-->
</script>

```

Vous devrez par la suite ajouter obligatoirement l'attribut "language". Cet attribut devra être précisé par le langage que vous utiliserez dans votre script.

Là, je pense que ça se corse. Je m'explique.

Les scripts sont des minis-langages de programmation. Oui, c'est de la programmation miniature. Ce qui veut dire que ce n'est pas à la portée de tout le monde. Si vous n'avez jamais programmé, ce qui risque fort d'arriver, vous pouvez vous rabattre sur des scripts tout prêts que proposent certaines pages web.

Quoiqu'il en soit, il faut que vous sachiez que les langages les plus connus pour les scripts sont :

- Les JavaScripts : du langage Java
- Les VbScripts : du langage Visual Basic

Il me semblerait que les JavaScripts soient les plus utilisés, pour leur puissance, mais les VbScripts sont un peu plus faciles.

Il est hors de question de vous enseigner ici un de ces langages, ce serait vraiment trop long.

Votre tag ressemblera donc à ça :

```
<script language="JavaScript">
```

Ou à ça :

```
<script language="VbScript">
```

## JavaScripts

### Le texte défilant

Voilà un script sympa, qui risque quand même de ralentir un petit peu votre PC (rien de bien grave).

Insérez ce code n'importe où dans votre page HTML :

```
<script language="JavaScript"><!--  
// debut du script  
var msg = "Votre texte ici !";  
var espaceur = "      ";  
pos=0;  
function defilmsg(){  
window.status=msg.  
substring(pos,msg.length)+  
espaceur+  
msg.substring(0,pos);  
pos++;  
if (pos > msg.length)  
pos=0  
window.setTimeout("defilmsg()", 200)  
}  
// fin du script  
// --></script>
```

Vous pouvez remplacer "Votre texte ici !" par le texte de votre choix (c'est en haut du script).

Pour faire fonctionner ce script, vous devez ajouter au tag <body> l'attribut "onload", précisé par "defilmsg()" : <body onload="defilmsg()">.



Il ne vous reste plus qu'à admirer le résultat.

## Le dégradé de couleur

Un script très fun, mais plus complexe que le précédent (plus gros aussi).

Vous devez mettre ce script à l'endroit précis où vous voulez voir votre dégradé de couleurs :

```
<script language="JavaScript"><!--
    <!--
    // Fading JavaScript (C)1999 Webmaster-fr - des tas de services gratuits !
-
    // web : http://www.webmaster-fr.net
    // e-mail : webmaster@webmaster-fr.net
    // Ce script est librement utilisable. Merci de ne pas effacer ces
commentaires.

    var hexa = "0123465789ABCDEF";

    function DecToHexa(DecNb) {
        x = Math.floor(DecNb / 16);
        h = hexa.charAt(x);
        x = DecNb % 16;
        h += hexa.charAt(x);

        return h;
    }

    function Degrade(dr,dg,db,fr,fg,fb,texte) {
        steps = texte.length;
        cr = dr; cg = dg; cb = db;
        sr = (fr - dr) / steps;
        sg = (fg - dg) / steps;
        sb = (fb - db) / steps;

        for (var x = 0; x <= steps; x++) {
            document.write('<FONT COLOR="#" + DecToHexa(cr) +
DecToHexa(cg) + DecToHexa(cb) + '>');
            document.write(texte.charAt(x));
            document.write('</FONT>');
            cr += sr; cg += sg; cb += sb;
        }
    }
    //-->
Degrade(0,0,255,0,255,0,"Votre texte ici !");
// --></script>
```

Pour personnaliser le dégradé, il faut vous pencher sur l'avant-dernière ligne (qui commence par Degrade).

Il y a 6 nombres séparés par des virgules. Les trois premiers représentent la couleur du début et les trois derniers la couleur de fin.

Pour écrire vos couleurs, il faut savoir qu'elles sont encodées en RVB (Rouge, Vert, Bleu). La première des trois représente donc le nombre de rouges (jusqu'à 255), la seconde le nombre de verts et la troisième le nombre de bleus.

Ici, c'est assez simplifié, comme vous pouvez le voir, les couleurs vont du bleu au vert. Simple, non ? A vous de créer maintenant vos propres couleurs (et vous avez le choix !).

Il ne vous reste plus qu'à remplacer le texte entre guillemets ("Votre texte ici !") par le texte de votre choix et vous aurez tout compris !

## Le roll-over

Le roll-over est une technique qui permet de changer une image lorsqu'on pointe dessus. Essayez par exemple de pointer sur cette image :

Cette technique est très utilisée et permet de faire des effets sur votre page facilement. Voici comment cela fonctionne :

```
<script language="Javascript">
function rollover(image,fichier)
{
document.images[image].src=fichier
}
```

Dans un premier temps, copiez quelque part entre les 2 <body> le code suivant :

```
<a onmouseover=rollover('MonRollOver','image2.jpg');
onmouseout=rollover('MonRollOver','image1.jpg');>
```

Ce code ne doit être écrit qu'une seule fois par page.

Ensuite, à l'endroit où vous souhaitez placer votre image, insérez le code HTML suivant :

- MonRollOver est le nom de votre roll-over. Si vous mettez plusieurs roll-over dans votre page, vous devez impérativement changer ce nom.
- image1.jpg et image2.jpg sont les noms des images à insérer. Ici, image1.jpg est l'image qui du début, et image2.jpg est l'image qui apparaît lorsqu'on pointe dessus.

## VbScripts

### Surprise !

Vous allez voir qu'on a moins de tracas avec un VbScript. Celui qui fait apparaître une fenêtre au survol par exemple. Insérez ce script où vous voulez dans votre page :

```
<script language="vbscript">
Sub move()
MsgBox "Votre texte ici !"
End Sub
</script>
```

Remplacez "Votre texte ici" par le texte de votre choix. Ensuite, entourez les mots qui vont faire apparaître la boîte de dialogue par des <a>. Oui, le tag de lien va nous servir. Il faut ajouter l'attribut "onmousemove" à préciser par "move()".

Exemple :

Pointez sur le mot `<a onmouseover="move()">BOUM</a>` !

Et le tour est joué !

## La date

C'est un script ultra-simple qui montre à quel point il est facile de programmer avec VbScript. Il affiche la date et l'heure où vous voulez dans votre page. Regardez, ça fait de suite plus "pro" :

*Aujourd'hui nous sommes le 02/06/2004. L'horloge de votre ordinateur indique qu'il est 02:15:23 !*

Pour cela, il suffit d'insérer le code suivant à l'endroit voulu :

```
<script language="vbscript">
document.write("Bienvenue sur ma page, nous sommes le " & Date & ", et il est actuellement
" & Time)
```

Voilà. Ce n'était peut-être pas simple, mais ce n'est pas pour rien que ce chapitre est parmi les derniers.

Vous pourriez faire une recherche sur [Yahoo !](#) par exemple, et rechercher des sites qui pourraient vous passer gratis des scripts... Désolé de ne pas pouvoir vous en dire plus, parce que ça devient un peu vaste à partir de là...

Vous pouvez aussi essayer cette adresse, vous y trouverez sûrement ce que vous cherchez en matière de scripts :

- <http://www.editeurjavascript.com>

Le prochain chapitre vous parlera des successeurs du HTML, du web du futur (qui existe déjà !) et de ses possibilités. De quoi bien clôturer cet apprentissage du HTML !

# L'Avenir Du HTML

Eh oui ! Comme tout, le HTML a évolué, évolue et évoluera. Il vous faudrait peut-être un peu d'histoire, pour voir où en est précisément ce langage HTML qui vous en a fait tant baver...

Et si je vous disais que par rapport à avant, c'est un langage simple, très simple ? Voici un ordre chronologique du langage de description pour page web, qui est actuellement HTML :

1. Le premier langage de description de pages web a été SGML (Standard Generalized Markup Language). Il est puissant et bourré d'options, si bien qu'il n'est pas à la portée de tout le monde, mais il convient parfaitement aux grandes organisations qui ont des exigences élevées.
2. Depuis peu, disons vers le début des années 90, on ressentit le besoin d'un langage plus simple (donc moins puissant) qui puisse être à la portée de tout le monde car le web commençait sa grande expansion. De cette idée est né le HTML (HyperText Markup Language)
3. Et maintenant, on se plaint de la puissance limitée de HTML ! On regrette la puissance du bon vieux SGML. On a donc eu l'idée de créer une variante simplifiée de SGML, une sorte

de SGML-light. De là est né récemment XML, un langage extensible, dont les possibilités sont bien plus nombreuses que HTML (d'où son nom : eXtensible Markup Language).

Difficile de vous en dire plus sur XML quand on connaît sa complexité. Mais je dois vous donner une petite idée de ses possibilités...

XML invente un nouveau type de tags. Mais pas n'importe lesquels. On connaissait déjà avec le HTML des tags servant à la disposition des éléments et à leur formatage (mise en forme). Ces nouveaux tags sont des tags sémantiques. Ils "expliquent" le texte qu'ils entourent...

Bizarre ? Pas tant que ça... En fait, cela veut dire que vous pouvez créer vos propres tags ! Par exemple, voici ce qu'on peut trouver :

```
<ville>Paris</ville>
```

Cela signifie en clair que Paris est une ville, oui, on l'aura tous compris. Sauf que vous savez que les tags ne s'affichent pas et l'internaute ne les voit pas.

Là, ça devient plus compliqué. Il faut avoir fait de la programmation pour comprendre l'utilité des tags sémantiques. On peut réutiliser ces informations et les afficher de différentes manières dans une page.

La page web du futur est donc une page différente à chaque fois que vous la visitez. Il y a plus de programmation, mais moins de pages différentes à créer. Ce qui rend les choses plus simples.

Mais, me direz-vous, quelle est l'utilité de ce langage, vous qui n'êtes que de simples créateurs de pages perso ? La réponse dépend de vos connaissances et ambitions.

- Si vous êtes quelqu'un qui aime bien faire des pages web sans aller chercher trop loin, alors ce n'est pas la peine de s'aventurer dans XML. Cela ne vous sera d'aucune utilité.
- Par contre, si vous programmez un peu, et si vous voulez impressionner votre entourage ou créer une page pour une entreprise, XML est le langage idéal et vaut bien ses longues heures d'apprentissage. Les résultats seront conséquents.

Par contre, je suis désolé mais je n'ai pas de page vers laquelle vous renvoyer. Il est sûr qu'une recherche dans AltaVista ou Yahoo de XML donnera de très nombreuses réponses. Il existe aussi de bons gros bouquins sur le XML. Comme je vous l'ai expliqué plus haut, c'est à vous de voir si ça vaut le coup de s'aventurer là-dedans !

Et voilà ! Notre apprentissage sur le HTML se termine ici (déjà ?). Vous avez acquis de nombreuses (et précieuses) connaissances sur le HTML, qui vous permettront de créer sans problème votre page perso !

Je vous ai ouvert la porte vers de nouvelles voies, comme le XML. A vous de voir si vous voulez approfondir vos connaissances là-dessus...

