



# Html pour les Newsletters

# PRÉAMBULE

Aborder le « vieil » HTML 4.0 alors qu'on parle classiquement d'XHTML voire même déjà de HTML5 peut sembler un peu inutile mais ce vieux langage de balisage a encore une place essentielle dans la vie de nombreux webdesigners.

En effet, on écrit encore en HTML 4.0 pour toutes les newsletters que vous recevez dans vos boîtes mail. C'est le seul langage qui est compris et interprété correctement par la majorité des clients mails classiques (outlook, thunderbird, ...) et webmails (hotmail, gmail, ...) du marché.

Et puis commencer par HTML 4.0 a finalement quelque chose de logique puisqu'il s'agit d'un langage basique et assez facile à comprendre qui vous donnera les premières clés qui vous aideront dans la conception de vos sites en XHTML et HTML5.

Nous verrons par la suite qu'il est néanmoins possible de combiner le vieil HTML4 pour la structure de vos newsletters avec CSS2 pour la mise en forme de certains éléments (mais pas tous).

# TABLE DES MATIÈRES

<b>PRÉAMBULE</b>	<b>2</b>
<b>INTRODUCTION</b>	<b>5</b>
• Le Web parle l'HTML .....	5
• Techniquement .....	5
• Les logiciels employés pour créer un fichier html .....	5
• Principes généraux du balisage .....	6
• Nomenclature.....	6
• Pourquoi l'HTML et pas XHTML ou HTML5 directement ?.....	7
<b>LA CRÉATION DES PREMIÈRES PAGES</b>	<b>8</b>
• Les outils nécessaires.....	8
• Le document HTML minimum .....	9
<b>FORMATAGE DE TEXTE</b>	<b>11</b>
• Les balises élémentaires.....	11
• Un exemple récapitulatif.....	12
• Le formatage par défaut.....	13
• Attributs multiples.....	13
• L'imbrication des balises.....	14
• Parents et enfants .....	14
• Les balises uniques (ou auto-fermantes).....	15
• Le codage des caractères spéciaux.....	15
• Les balises « en ligne » et les balises « blocs ».....	15
• Les autres balises de texte .....	16
• Les titres.....	17
• Les listes (à puce).....	17
• Les listes imbriquées.....	18
• Les filets horizontaux .....	18
<b>LES IMAGES</b>	<b>20</b>
• Principes.....	20
• En pratique .....	20
• Exemple récapitulatif.....	22
• L'adressage relatif et absolu .....	23
Chemin relatif	23
Chemin absolu	23
<b>LES ARRIÈRES-PLANS</b>	<b>24</b>
• Couleur et image d'arrière-plan.....	24
<b>LES LIENS HYPertextES ET HYPERMÉDIAS</b>	<b>25</b>
• Principe .....	25
• Les cibles des hyperliens .....	25
• En pratique .....	25
• Les liens hypermedia.....	26
• Les ancres.....	27
• Les liens vers une adresse mail .....	27

<b>LA COULEUR DU TEXTE ET DES HYPERLIENS</b>	<b>28</b>
• La couleur du texte .....	28
• La couleur des liens, des liens visités et des liens actifs .....	28
<b>LES TABLEAUX</b>	<b>29</b>
• Principes .....	29
• Comportements par défaut .....	30
• Les attributs facultatifs des tableaux.....	30
• Les attributs facultatifs des colonnes de tableaux .....	32
• Les attributs facultatifs des colonnes ET des lignes de tableaux .....	33
• Les attributs facultatifs des tableaux, des colonnes ET des lignes .....	34
<b>TABLEAU RÉCAPITULATIF DES BALISES, DE LEURS ATTRIBUTS ET DE LEUR TYPE</b>	<b>35</b>

# INTRODUCTION

## ● Le Web parle l'HTML

Créé par Tim Berners Lee et Robert Caillau, l'HTML est le langage universel utilisé pour **communiquer sur le Web**.

HTML signifie **HyperText Markup Language**.

Il est directement inspiré du SGML (Standard Generalized Markup Language).

L'HTML n'est pas un langage de programmation, mais un langage de formatage (càd, de mise en page). "Markup" signifie littéralement "marquage" !

L'une des spécificités de l'HTML est qu'il permet de créer des hypertextes en établissant des liens entre plusieurs documents.

Le langage HTML est simple à comprendre et à utiliser, d'où son immense succès !

## ● Techniquement ...

Afin de transiter le plus rapidement possible sur Internet et afin de créer un langage universel :

- L'HTML exploite un format de texte très compact et très peu sophistiqué (format de texte pur et dur)
- c'est un format ASCII ! Il est donc impossible de coder certains caractères spéciaux comme le "é" (qu'il faudra remplacer par des codes particuliers uniquement dans le cas de création des newsletters).

## ● Les logiciels employés pour créer un fichier html

Pour créer un fichier HTML, le plus simple des éditeurs de texte suffit donc :

SublimeText, Brackets, Atom, Bloc-note, Notepad, TextEdit, Cyberdoc ...

Des logiciels WYSIWYG existent et permettent un rendu immédiat des balises que vous écrivez pour vous faciliter la vie.

Le plus connu de ces logiciels est issu de la suite Adobe : Dreamweaver.

Et entre ces logiciels simples et les WYSIWYG se trouvent des logiciels hybrides qui vont vous faciliter la vie dans l'écriture de votre code mais sans vous offrir d'aperçu :

Notepad++, BBedit, ...

# ● Principes généraux du balisage

Les documents HTML sont hybrides.

C'est-à-dire qu'ils comportent divers éléments en leur sein :

- Du texte adressé à l'internaute
- Des instructions destinées à formater le texte (couleurs, grasse, police, etc.)
- Des instructions destinées à ajouter des éléments graphiques (images, listes à puce, etc.)

Ces instructions seront différenciées du texte par les signes < et >, par exemple <html>.

Ces instructions s'appellent des **tags** ou des **balises**. L'HTML est donc un langage de « balisage ».

## **Gardez toujours à l'esprit :**

Qu'une balise marque une action pour le navigateur (ce qu'il doit faire).

Que certaines balises possèdent des attributs (facultatifs ou obligatoires) qui précisent les modalités de cette action (comment il doit le faire).

# ● Nomenclature

La nomenclature (ou la façon d'écrire) des balises est assez simple :

```
<balise> ... </balise>
```

Ou bien avec un attribut :

```
<balise attribut="valeur">...</balise>
```

Ou encore avec plusieurs attributs :

```
<balise attribut1="valeur" attribut2="valeur">...</balise>
```

## **Mise en garde**

L'HTML est d'abord téléchargé, ensuite lu et finalement interprété par le parseur du navigateur Web (Chrome, Firefox, Internet Explorer, Safari, Opera, ...).

Le problème est que chaque parseur (gecko pour Firefox, webkit pour safari et chrome, trident pour Internet Explorer et Presto pour Opera) a sa propre façon d'interpréter le code que vous écrivez et c'est là que les choses vont se compliquer évidemment.

Avec HTML, on n'a donc presque jamais la maîtrise totale de l'affichage de son document sur les écrans des utilisateurs.

## ● Pourquoi l'HTML et pas XHTML ou HTML5 directement ?

Les éditeurs HTML facilitent la tâche mais ils ne sont pas toujours parfaits. Il faut parfois plonger dans le code source pour corriger les dysfonctionnements.

Les codes source de toutes les pages Web sont disponibles. On peut toujours s'en inspirer... mais il faut le comprendre.

Il faut une connaissance pointue du HTML pour inclure les codes du Javascript ou créer des pages dynamiques.

À l'heure actuelle, la version 4.0 d'HTML est encore utilisée structurellement pour un seul et unique produit Web : les newsletters. Simplement parce que les parseurs des clients et webmails sont nettement moins évolués que ceux des navigateurs même si des progrès sont constatés année après année. Mais malheureusement pas encore assez de progrès pour se passer définitivement de notre bon vieux HTML 4.0.

L'XHTML et l'HTML5 ne sont finalement que des produits plus aboutis et plus rigoureux que l'HTML 4.0.

Il est donc capital de bien apprendre la base pour pouvoir ensuite décliner ce que vous aurez appris pour les appliquer sur les nouveaux standards tel que HTML5.

# LA CRÉATION DES PREMIÈRES PAGES

## ● Les outils nécessaires

Le premier outil essentiel est celui qui va vous permettre de créer votre fichier HTML et de le modifier.

Comme dit précédemment, un simple éditeur de texte de base est suffisant pour coder n'importe quelle page HTML (ou même XHTML, HTML5, PHP, CSS, XML, etc.).

Ensuite, vous aurez besoin également d'un navigateur (votre navigateur préféré fera l'affaire) mais prenez le réflexe de tester vos pages sur un maximum de navigateurs en n'oubliant surtout jamais les navigateurs principaux :

- **Internet Explorer** (environ 20% des utilisateurs mondiaux)
- **Firefox** (environ 38% des utilisateurs mondiaux)
- **Chrome** (environ 35% des utilisateurs mondiaux)
- **Safari** (environ 4% des utilisateurs mondiaux)
- **Opera** (environ 2,5% des utilisateurs mondiaux)

Les autres % sont partagés entre tous les « plus petits » navigateurs qui peuplent Internet.

À ce stade-ci, on n'a pas encore besoin :

- d'une connexion Internet (et de manière générale, quand vous travaillez en « local » (càd uniquement sur votre disque dur et pas sur un serveur distant) vous n'avez jamais besoin d'une connexion Internet).
- de disposer d'un éditeur HTML performant et puissant (du style Dreamweaver – et on verra d'ailleurs que même pour coder en XHTML/HTML5/CSS on peut toujours tout faire de façon propre sans utiliser Dreamweaver).

**Et c'est tout !**

Vous n'avez besoin de rien d'autre que de ça et des quelques connaissances qui vont suivre évidemment ☺.

## ● Le document HTML minimum

Voici le document HTML minimum que vous pourrez écrire, il reprend les balises de base sans leurs attributs importants, juste les balises et c'est tout.

```
<html>
<head>
<title>Vive le WebDesign !</title>
</head>
<body>
</body>
</html>
```

La balise <html> détermine le début du document HTML.

La balise <head> détermine le début de la zone d'en-tête du document HTML, c'est dans cette partie qu'on va donner un titre à notre page, qu'on va lui insérer des ordres de mise en forme CSS, qu'on va lui lier du javascript ou qu'on va lui préciser certaines meta-informations.

La balise <title> détermine le titre de la page en cours et la balise </title> détermine que ce titre est terminé.

La balise </head> détermine la fin de la zone d'en-tête du document HTML.

La balise <body> détermine le début de la zone du corps du document et </body> la fin de cette zone, c'est exactement entre ces 2 balises qu'absolument TOUT le contenu de votre page DOIT être inscrit.

La balise </html> détermine la fin du document HTML.

### Remarque

Vous aurez évidemment remarqué qu'à chaque balise de début d'action (soit <...>) correspond une balise de fin de cette action (soit </...>).

Ce couple composé d'une balise ouvrante et d'une balise fermante est appelé « élément ». Il est impératif de toujours fermer une balise qui a été ouverte auparavant à moins que celle-ci soit auto-fermante (nous y viendrons plus tard).

### Remarque

Notons également que les balises ne sont pas « case sensitive ». Il est donc équivalent d'écrire <html> ou <HTML> ou <HtMl>, etc. Il est cependant convenu que les balises sont généralement toutes écrites en minuscule, c'est plus confortable pour l'écriture.

Cependant les éléments comme les noms de police ou les URLs peuvent être case sensitive. Pour certains systèmes d'exploitation (Unix, par exemple), Arial n'est pas égal à arial, de même IMAGE.gif n'est pas égal à image.gif (tout dépend évidemment de comment est écrit votre fichier à la base) !

### Remarque

Dernière remarque, notons que le navigateur n'interprète pas les sauts de ligne, les tabulations, les espaces multiples (il en prend 1 et c'est tout) dans le code HTML.

On peut donc aérer et indenter son code pour plus de lisibilité.

## À vous de jouer !

- Ouvrez un éditeur de texte
- Écrivez le code ci-dessus
- Enregistrez le fichier avec l'extension .html
- Lancez votre navigateur préféré
- Localisez et ouvrez la page qui vient d'être créée

### Attention

Le langage HTML comporte une série de règles de nomenclature qu'il faut absolument respecter car ensuite le code est interprété par le parseur.

Le non-respect de ces règles peut causer des problèmes (mauvais affichage ou pas d'affichage du tout) et malheureusement pour vous certains navigateurs sont très tolérants à l'erreur. Ceci veut simplement dire qu'il ne faut pas se contenter d'un affichage dans une version d'un navigateur pour se dire qu'une page ou un site Web est compatible cross-browsers.

Nous allons maintenant apprendre à remplir la zone <body> du document pour que la page affiche quand même quelque chose :

- Du texte
- Des images
- Des listes à puces
- Etc.

### À partir de maintenant ...

À chaque fois que vous allez modifier quelque chose dans votre code, il ne sera pas nécessaire de rouvrir à chaque fois un nouveau navigateur.

Vous devrez simplement modifier votre fichier .html, l'enregistrer, retourner ensuite dans le navigateur que vous avez ouvert auparavant pour faire votre précédent aperçu de page et appuyer sur la touche F5 (ou le bouton « actualiser ») pour voir la transition entre votre résultat précédent et le nouveau résultat, qui affichera les modifications que vous venez d'apporter à votre fichier html.

# FORMATAGE DE TEXTE

## ● Les balises élémentaires

Fort	<code>&lt;strong&gt; ... &lt;/strong&gt;</code>
Définit le texte balisé comme étant important, fort. Le parseur affiche généralement le texte en gras.	<b>Effet de la balise strong</b>
Emphase	<code>&lt;em&gt; ... &lt;/em&gt;</code>
Définit le texte balisé comme étant important, mis en emphase. Le parseur affiche généralement le texte en italique.	<i>Effet de la balise em</i>
Choix typographique du texte	<code>&lt;font attribut="valeur" &gt; ...&lt;/font&gt;</code>
<b>Attribut facultatif</b> : <code>&lt;font size="valeur"&gt;</code> Le texte balisé s'affiche à la taille précisée.	1 2 3 4 5 6 7
<b>Attribut facultatif</b> : <code>&lt;font face="police1, police2, ..."&gt;</code> Le navigateur recherche les polices dans l'ordre spécifié. S'il ne trouve pas la première, il passe à la deuxième, etc. Généralement, on place à la fin de la déclaration une famille de police pour être sûr que le texte profite d'un style typographique. Ces familles peuvent être les suivantes : <ul style="list-style-type: none"><li>- Sans-serif</li><li>- Serif</li><li>- Monospace</li><li>- Cursive</li></ul> Exemple concret : <code>&lt;font face="Verdana, Geneva, sans-serif" &gt; ... &lt;/font&gt;</code>	Verdana Georgia Times New Roman Courier New Courier Arial Tahoma Trebuchet MS Arial Black Palatino Linotype Book Antiqua Lucida Sans Unicode Luci da Consol e Comic Sans MS
<b>Attribut facultatif</b> : <code>&lt;font color="#+ code RVB en hexadécimal ou nom de la couleur en anglais"&gt;</code> Le texte balisé s'affiche dans la couleur spécifiée. ex : <code>&lt;font color="#FF4500"&gt;</code> ou <code>&lt;font color="orangered"&gt;</code> <a href="#">Les couleurs et leurs noms en anglais.</a>	Du texte avec la couleur hexadécimale #FF4500 ou son correspondant en anglais « orangered »

Nouveau paragraphe	<code>&lt;p attribut="valeur" &gt; ...&lt;/p&gt;</code>
Définit le texte balisé comme étant un paragraphe. Le parseur affiche généralement un espace vertical au-dessus et en-dessous du texte.  <b>Attribut facultatif</b> : <code>&lt;p align="valeur"&gt;</code> Cet attribut permet, respectivement, d'aligner à gauche (par défaut), à droite, de centrer ou de justifier le paragraphe.	<code>&lt;p align="left" &gt;  &lt;p align="right" &gt;  &lt;p align="center" &gt;  &lt;p align="justify" &gt; . . . . .</code>
Retour à la ligne	<code>&lt;br&gt;</code>
Retour chariot conditionnel ou retour forcé (Maj + Enter). Normalement, le texte passe à la ligne tout seul lorsqu'il atteint le bord droit de la fenêtre du navigateur. Pour outrepasser cette règle et le forcer à faire un saut à la ligne, il faut utiliser la balise <code>&lt;br&gt;</code> .	Une ligne avec un <code>&lt;br&gt;</code> et la suite à la ligne <code>&lt;br&gt;</code> sans création d'un <code>&lt;br&gt;</code> nouveau paragraphe.
Commentaires	<code>&lt;!-- ... --&gt;</code>
Définit le texte balisé comme étant un commentaire. Le parseur ne l'affiche pas.	<code>&lt;!-- ce texte ne s'affichera pas à l'écran --&gt;</code>

## ● Un exemple récapitulatif

```
<html>
<head><title>Baliser le texte</title></head>
<body>
texte simple<br>
<b>texte en gras</b><br>
<strong>texte fort</strong><br>
<i>texte en italique</i><br>
<em>texte en emphase</em><br>
<b><i>texte en gras et en italique</i></b><br>
<font size="5">texte plus grand</font><br>
<font color="#0000FF">texte en bleu</font>
<font face="Arial,Helvetica">texte en Arial</font>
<!-- C'est fini -->
</body>
</html>
```

## ● Le formatage par défaut

Le texte simple (càd, sans formatage) s'écrit sans balises.

Dès lors, le parseur l'affichera avec la police et la taille définie par défaut : Times ou Times New Roman, +/- 12 points, noir.

Mais attention, l'internaute a la possibilité de modifier ces valeurs par défaut en configurant le navigateur selon ses désirs. Néanmoins, peu d'utilisateurs le font.

En fait, un grand nombre de paramètres sont définis automatiquement, par défaut, par le parseur : taille des marges de la page, crénage, interlignage, disposition des puces, affichage des tableaux, apparence des formulaires, ...

Mais, attention, tous les parseurs n'utilisent pas exactement les mêmes paramètres par défaut.

Pour éviter les différences d'affichage entre les navigateurs, on peut préciser soi-même les paramètres que l'on désire appliquer.

Toutefois, en HTML, tout n'est pas paramétrable : marges, interlignage, disposition des puces, ... sont toujours gérés par le parseur !

### Unités de mesure

La taille dans `<font size="?">` ne peut pas être indiquée de manière très précise avec les unités habituelles (points, mm, pica, ...).

En fait, l'attribut `size` n'accepte que les valeurs suivantes : un nombre de 1 à 7 (la valeur par défaut étant 3) ou une position relative par rapport à la valeur par défaut... soit -2 -1 ou +1 +2 +3 +4

Pour rappel, la valeur 3 correspond par défaut à +/- 12 points. Mais l'utilisateur peut agrandir ou réduire le texte à sa guise.

## ● Attributs multiples

Il est évidemment possible (et vital ☺) de combiner plusieurs attributs facultatifs les uns derrière les autres pour une même balise.

Par exemple, la balise `<font>` peut recevoir les 3 attributs `size`, `color` et `face` vus auparavant :

```
<font size="6" color="#cccc00" face="Tahoma, Geneva, sans-serif">...</font>
```

### Remarque importante

L'ordre des attributs n'a vraiment aucune importance, pour autant qu'ils soient séparés par un espace et que leur valeur soit bien placée entre guillemets.

## ● L'imbrication des balises

L'imbrication des balises est LA règle de base qu'il faut maîtriser quand on commence à coder et la règle à appliquer pour ne jamais se tromper est très simple :

la dernière balise ouverte sera TOUJOURS la première à être fermée.

Veillez donc à ouvrir et à fermer les balises dans l'ordre inversé de manière à respecter l'imbrication des balises.

De cette façon écrire ceci est correct :

```
<p><strong>Coucou</strong> les <em>WebDesigners</em> !</p>
```

Tandis que ceci n'est pas correct :

```
<p><strong>Coucou les <em>WebDesigners</strong></em> !</p>
```

## ● Parents et enfants

Une notion qu'il est important de comprendre est bien celle de la relation de parent et d'enfant entre les balises.

Du fait de leurs imbrications, les éléments de la page sont hiérarchisés selon des relations de parents et d'enfants.

*Exemple :*

```
<p>Bonjour, je m'appelle <strong><em>Arthur</em></strong></p>
```

On peut donc dire ici que l'élément `<p>` est le parent des éléments `<strong>` et `<em>`.

Par ailleurs, l'élément `<em>` est l'enfant de l'élément `<strong>` et `<p>` alors que `<strong>` est uniquement l'enfant de `<p>`. Vous suivez ? 😊

De manière générale, les éléments parents appliquent leur formatage à leurs enfants. On parle de l'héritage des éléments parents. Ici, le mot 'Arthur' apparaîtra en italique et en gras car ces formatages sont hérités de ses balises parentes.

### Hiérarchisation des commandes

On utilise souvent plusieurs balises pour un même élément de texte.  
Dans le cas de commandes multiples et contradictoires...

```
<font size="4"><font size="7">...</font></font>
```

... c'est la commande la plus proche du texte qui l'emporte. Ici, le texte aura la taille 7.

C'est une règle qui sera applicable tout le temps à partir d'HTML 4.0 jusqu'à HTML5 et même dans les feuilles de styles.

## ● Les balises uniques (ou auto-fermantes)

La balise `<br>` représente une action ponctuelle. Elle ne balise pas vraiment une portion de texte. En conséquence, elle n'a pas besoin de balise de fin et il n'est donc pas nécessaire d'insérer de balise `</br>`. On appelle ce type de balise, élément unique ou balise auto-fermante.

Nous verrons qu'il existe d'autres balises du même type.

## ● Le codage des caractères spéciaux

En HTML 4.0, les caractères spéciaux (é è à ê ô ë \ < / = € ...) sont tous codés avec un code chiffré particulier (esperluette + code ISO ;) ou une suite de caractères (esperluette + entity ;). On opte généralement plutôt pour la 2<sup>e</sup> solution car elle est + porteuse de sens que la première.

*Par exemple :*

- ç devient `&ccedil;`
- è devient `&egrave;`
- é devient `&eacute;`

## ● Les balises « en ligne » et les balises « blocs »

Il est très important de comprendre la différence entre les 2 types de balises qui existent.

**Les balises en ligne (ou inline)** ne provoquent pas de rupture du flux du document. Cela signifie que les balises de ce type ne provoquent pas de retour chariot : le texte continue à s'afficher "au kilomètre" de gauche à droite jusqu'à atteindre le bord droit de la fenêtre du navigateur.

*Exemples :* strong, em, font, ...

Les balises blocs (ou block) créent un bloc dans le flux du document. Ce bloc occupe toute la largeur disponible (jusqu'au bord droit de la fenêtre du navigateur) et provoque un retour chariot avant et après lui.

*Exemple :* p

## ● Les autres balises de texte

Voici un florilège des balises de texte les plus utilisées encore de nos jours.

Divisions	<code>&lt;div&gt; ... &lt;/div&gt;</code>
<p>La balise <code>&lt;div&gt;</code> sert à créer des « divisions » dans la page. En clair, elle définit le texte balisé comme un bloc auquel aucun sens particulier n'est associé. Le parseur ne produit aucun affichage particulier.</p> <p>La balise <code>&lt;div&gt;</code> est donc une sorte de conteneur neutre auquel on peut, par exemple, greffer un attribut définissant l'alignement horizontal.</p> <p><b>Cette balise est de type bloc.</b></p> <p>On l'utilisera beaucoup par après, mais peu à l'heure de l'HTML 4.0.</p>	<p>La balise <code>&lt;div&gt;</code> ne produit pas d'effet particulier.</p>
Indice	<code>&lt;sub&gt; ... &lt;/sub&gt;</code>
Affiche le texte balisé en indice (sous la ligne de base).	Un <sub>indice</sub> se place sous la ligne de base.
Exposant	<code>&lt;sup&gt; ... &lt;/sup&gt;</code>
Affiche le texte balisé en exposant (au-dessus de la ligne de base).	Un <sup>exposant</sup> se place au-dessus de la ligne de base.
Bloc de citation	<code>&lt;blockquote&gt;...&lt;/blockquote&gt;</code>
Définit le texte balisé comme étant un bloc de citation. Le parseur augmente généralement les marges gauche et droite du texte (effet d'étau).	Un bloc de citation.
Préformaté	<code>&lt;pre&gt; ... &lt;/pre&gt;</code>
Début et fin de texte préformaté. Habituellement, les navigateurs ne tiennent pas compte des espaces multiples, des tabulations et des sauts de ligne. Avec cette balise, le navigateur interprète le code stricto sensu. <code>&lt;pre&gt;</code> affiche également le texte dans la police Courier.	Un texte dans la balise <code>&lt;pre&gt;</code> .

## ● Les titres

Les titres en HTML 4.0 (et par après également) sont essentiels pour séparer sémantiquement vos contenus.

Titres	<code>&lt;h1&gt; ... &lt;/h1&gt;</code>
<p>Le texte balisé est défini comme étant un titre.</p> <p>Le chiffre mentionné dans la balise (représenté ici par la variable n) indique le niveau hiérarchique (sa valeur variant de 1 à 6 ; 1 étant le plus gros titre).</p> <p>Le parseur ajoute généralement un espace vertical au-dessus et en-dessous du titre et combine plusieurs paramètres typographiques (taille, graisse) pour refléter son niveau hiérarchique.</p> <p><b>Attribut facultatif :</b> <code>&lt;h1 align="valeur"&gt;&lt;/h1&gt;</code> Cet attribut permet, respectivement, d'aligner à gauche (par défaut - left), à droite (right), de centrer (center) ou de justifier (justify) le titre.</p>	<pre>&lt;h1&gt;&lt;/h1&gt; &lt;h2&gt;&lt;/h2&gt; &lt;h3&gt;&lt;/h3&gt; &lt;h4&gt;&lt;/h4&gt; &lt;h5&gt;&lt;/h5&gt; &lt;h6&gt;&lt;/h6&gt;</pre>

## ● Les listes (à puce)

De nouveau un élément très important qu'on retrouvera jusqu'à l'HTML5. Les listes à puces numérotées ou non sont cruciales dans l'écriture et la disposition de vos contenus.

Liste à puce non-ordonnée	<code>&lt;ul&gt; ... &lt;/ul&gt;</code>
<p>Définit le texte balisé comme une liste à puces, c'est-à-dire une liste non-ordonnée.</p> <p>Le parseur ajoute généralement un espace vertical au-dessus et en-dessous de la liste.</p> <p><b>Attribut facultatif :</b> <code>&lt;ul type="disc, circle ou square"&gt;&lt;/ul&gt;</code> Affiche les puces, respectivement, sous forme de cercles pleins (par défaut), de cercles vides ou de carrés.</p>	<ul style="list-style-type: none"><li>● Liste à puce avec l'attribut disc.</li><li>○ Liste à puce avec l'attribut circle</li><li>▪ Liste à puce avec l'attribut square.</li></ul>

Liste à puce ordonnée ou numérotée	<code>&lt;ol&gt; ... &lt;/ol&gt;</code>
<p>Définit le texte balisé comme une liste ordonnée. Le parseur ajoute généralement un espace vertical au-dessus et en-dessous de la liste.</p> <p><b>Attribut facultatif :</b> <code>&lt;ol type="A, a, I, i ou 1"&gt;&lt;/ol&gt;</code> Ordonne les éléments de la liste en les faisant précéder d'une lettre majuscules, d'une lettre minuscule, d'un chiffre romain majuscule, d'un chiffre romain minuscule ou d'un chiffre arabe (par défaut) incrémenté dans l'ordre croissant.</p>	<ol style="list-style-type: none"><li>A. Liste à puce avec l'attribut A</li><li>B. Liste à puce avec l'attribut A</li><li>1. Liste à puce avec l'attribut 1</li><li>2. Liste à puce avec l'attribut 1</li></ol>

### Attribut facultatif :

```
<ol start="..."></ol>
```

Définit la valeur de départ (ici représentée par ...) de l'incrément.

Exemple : `<ol start="3">...</ol>` fait débiter la liste ordonnée à 3 (ou C, c, III, iii si un autre type est spécifié).

3. Liste à puce avec l'attribut start à la valeur 3

### Les balises de liste

```
<li> ... </li>
```

Combiné avec une balise `<ul>` ou `<ol>`, le texte balisé avec `<li>` est défini comme étant un élément de la liste.

Le parseur augmente généralement la marge gauche de manière à décaler chaque item.

Mais on ne peut pas paramétrer finement l'apparence des listes (espace, présence d'un point, ...)

Exemple :

```
<ul>
<li>Bruxelles</li>
<li>Namur</li>
<li>Liège</li>
</ul>
```

## ● Les listes imbriquées

Les listes peuvent s'imbriquer l'une dans l'autre :

```
<ul>
  <li>Garçons
    <ul>
      <li>Pierre</li>
      <li>Olivier</li>
    </ul>
  </li>
  <li>Filles
    <ul>
      <li>Ethel</li>
      <li>Maud</li>
    </ul>
  </li>
</ul>
```

Les listes imbriquées peuvent mélanger listes ordonnées et non-ordonnées

Si l'on imbrique deux listes à puces, le navigateur change automatiquement le type de la puce.

## ● Les filets horizontaux

Les filets horizontaux sont des séparateurs horizontaux auxquels vous pouvez appliquer une certaine largeur, une certaine épaisseur et une certaine couleur.

### Les filets horizontaux

```
<hr>
```

Insère un filet horizontal. Il s'agit d'une balise unique ne nécessitant pas de balise de fin.

Le parseur affiche généralement un filet occupant toute la largeur disponible, centré, d'environ 3 pixels

d'épaisseurs et avec un effet 3D (estampage). Cette balise provoque un saut de ligne avant et après.

**Attribut facultatif :** *width="x" ou "x%"*

Définit la largeur du filet. La valeur de l'attribut est exprimée en pixels ou en pourcentage (%) de la largeur disponible. Valeur par défaut = 100%

**Attribut facultatif :** *size="x"*

Définit l'épaisseur du filet. La valeur de l'attribut est exprimée en pixels. Valeur par défaut = 2.

**Attribut facultatif :** *align="left, right ou center"*

Définit l'alignement horizontal du filet, respectivement à gauche, à droite ou au centre (par défaut)

# LES IMAGES

## ● Principes

En HTML, les images ne sont pas incorporées directement dans la page mais sont en quelque sorte « liées » à la page.

Dès lors, pour insérer une image, on utilise une instruction de téléchargement suivie de l'adresse et du nom de l'image.

Une fois l'image téléchargée, celle-ci est placée dans la mémoire cache du navigateur et est affichée par le parseur. En conséquence, le fait d'insérer la même image à plusieurs reprises dans la page n'a aucune incidence sur le temps de téléchargement.

Comparativement au texte, le codage des images nécessite un grand volume d'informations (la couleur de chaque pixel doit être définie). Les images représentent donc souvent la plus grande partie du poids de la page.

Les formats principaux utilisés pour l'enregistrement des images destinée à être affichées sur le Web sont le **GIF**, le **JPEG**, **PNG** et le **SVG**. Ces formats se caractérisent par leur capacité à compresser le poids des images grâce à leur algorithme (avec ou sans perte).

Rappelons que pour générer de la transparence, il est vivement recommandé, pour les images complexes, d'utiliser le **PNG 24bits** plutôt que du **GIF** qui aura un rendu moins bon et moins propre graphiquement.

Les dimensions de l'image (largeur x hauteur), la profondeur de couleur (nombre de bits alloués au codage de la couleur de chaque pixel) et le taux de compression de l'algorithme sont les trois facteurs clés déterminant le poids d'une image.

## ● En pratique

Image	<img>
<p><b>Attribut obligatoire :</b> <i>src="adresse et nom de l'image"</i></p> <p>L'adresse de l'image peut être exprimée de manière <b>absolue</b> (c:\site\images\icône.gif – chemin complet) ou <b>relative</b> (images\icône.gif – chemin par rapport au fichier dans lequel l'image est appelée) par rapport à l'adresse de la page.</p> <p><b>Attribut facultatif :</b> <i>width="x" height="y"</i></p> <p>Définissent la largeur et la hauteur en pixels ou en pourcentage (%) de l'image.</p> <p>La mention des attributs de taille accélère l'affichage de la page. En effet, le parseur peut prévoir l'espace nécessaire pour l'image avant que celle-ci ne soit complètement téléchargée. De ce fait, pour les pros, l'utilisation de ces attributs prend un caractère obligatoire.</p>	<p>Une image classique sans attribut facultatif :</p> 

Il est déconseillé de réduire ou d'agrandir artificiellement l'image en paramétrant les attributs width et height avec d'autres valeurs que les dimensions réelles de l'image. D'une part, parce que le redimensionnement de l'image par le parseur donne un rendu de faible qualité et, d'autre part, parce que dans le cas d'une réduction, l'internaute télécharge inutilement plus de pixels qu'affichés.

**Attribut facultatif** : Texte alternatif  
*alt= "texte s'affichant à la place de l'image"*  
Définit un texte alternatif

Le texte alternatif sera visible :

- lorsque l'affichage des images est désactivé par l'internaute (dans les options avancées du navigateur)
- durant le temps de téléchargement de l'image
- dans certains navigateurs, sous forme de "bulle" lorsque l'on laisse le curseur immobile au-dessus de l'image
- par le spider des moteurs de recherche
- les textes alternatifs peuvent bien entendu aussi faciliter la vie des mal-voyants dotés d'un système de synthèse vocale...

De ce fait, pour les pros, l'utilisation de cet attribut prend un **caractère obligatoire**.

**alt** doit être présent sur toutes les images, même s'il peut rester vide (pour les images purement décoratives). Il sert à apporter la même information que l'image (en général le texte qu'elle porte) dans le code de la page. Le but de ce report d'information est que celle-ci reste disponible lorsque l'image ne peut être chargée (à cause d'une surcharge de serveur ou d'un problème de connexion...) ou qu'elle ne peut être vue (par un robot de moteur de recherche ou un utilisateur de lecteur d'écran...).

**Attribut facultatif** : Bordures  
*border="x"*

Définit l'épaisseur en pixels de la bordure entourant l'image. Par défaut, border="0" sauf dans le cas où l'image sert de bouton (l'image est associée à un lien hypertexte).

**Attribut facultatif** : Alignement vertical du texte par rapport à l'image.  
*align="top, middle ou bottom (par défaut)"*

Définit la position de la première ligne du paragraphe par rapport à l'image qui précède (respectivement : au-dessus, au milieu ou au bas du côté droit de l'image). Il s'agit donc d'une fonction d'habillage de l'image.



**Attribut facultatif** : Alignement horizontal de l'image par rapport au texte  
*align="left, right"*

Définit la position de l'image par rapport au paragraphe de texte qui la suit (respectivement : à gauche ou à droite du paragraphe texte). Il s'agit également d'une fonction d'habillage de l'image.

**Attention**, <img> n'accepte qu'un seul attribut align et celui-ci ne peut prendre qu'une seule valeur parmi les 5 spécifiées. En conséquence, en dehors des valeurs par défaut (bottom), il est impossible de spécifier en mêmes temps la position verticale du texte et la position horizontale de l'image !

**Attribut facultatif** : espacement vertical  
*vspace="x"*

Définit l'espacement en pixels au-dessus et en-dessous de l'image. Ceci permet de peaufiner l'habillage de l'image.

**Attribut facultatif** : espacement horizontal  
*hspace="x"*

Définit l'espacement en pixels à gauche et à droite de l'image. Ceci permet de peaufiner l'habillage de l'image.

## ● Exemple récapitulatif

```

```



## ● L'adressage relatif et absolu

Souvent source d'erreurs, de pages introuvables, d'images manquantes ou de styles non appliqués, le format d'écriture des liens est très important. Et pourtant, on ne prend pas forcément garde à la façon dont on fait pointer ces derniers vers les différents fichiers cible. Suivant la convention d'écriture utilisée, de nombreux problèmes peuvent survenir.

En effet, deux types d'écritures cohabitent, chacune avec leur spécificité propre :

**le chemin relatif** (absence de slash initial)

**le chemin absolu** (slash en préfixe du chemin)

### CHEMIN RELATIF

**dossier/page.html**

La cible (page.html) va être cherchée par le navigateur à partir de la page html ou de la feuille de style qu'il interprète, dans le sous-répertoire "dossier".

**../dossier/page.html**

../ : Signifie que la page est cherchée à partir du répertoire parent (on remonte d'un niveau).

Il est tout à fait possible de remonter plusieurs répertoires.

Pour cela il faudra cumuler les ../ et ainsi pour remonter de deux niveaux, il sera possible d'écrire ../../dossier/page.html.

### CHEMIN ABSOLU

**/dossier/page.html**

Le slash initial précise que l'on ne se réfère plus à l'emplacement courant mais que l'on remonte directement à la racine pour ensuite préciser le chemin complet.

Dans bien des cas, le chemin absolu est la convention d'écriture la plus sûre, mais aussi la moins souple si l'arborescence est amenée à changer. La page cible sera cherchée par le navigateur à partir de la racine du site.

Dans notre exemple, le fichier page.html sera recherché directement à partir de la racine du domaine : <http://www.mondomaine.fr/dossier/page.html>.

# LES ARRIÈRES-PLANS

## ● Couleur et image d'arrière-plan

Il est possible d'appliquer une couleur d'arrière-plan à différents éléments, l'une des utilisations les plus classiques est de mettre un couleur de fond sur la balise <body> (qui est donc le corps de votre document en entier).

Corps du document	<body> ... </body>
<b>Attribut facultatif</b> : La couleur d'arrière-plan de la page <i>bgcolor="#RRVVBB ou nom de la couleur en anglais"</i>  Modifie la couleur d'arrière-plan par défaut (blanc) de la page. La couleur est codée en RGB hexadécimal ou avec le nom en anglais de la couleur en question.	Une autre couleur d'arrière-plan.

Corps du document	<body> ... </body>
<b>Attribut facultatif</b> : Le motif d'arrière-plan de la page. <i>background="adresse et nom de l'image"</i>  Définit une image comme arrière-plan de la page. La valeur de cet attribut est comparable à src pour la balise <img>.  L'image d'arrière-plan est utilisée comme un motif (pattern). Le parseur la duplique indéfiniment de gauche à droite et de haut en bas de manière à occuper tout l'espace de la page (effet de mosaïque).  Bien entendu, pour un effet esthétique réussi, il faut veiller à ce que les raccords ne se voient pas lorsqu'on en juxtapose plusieurs exemplaires côte à côte.	

# LES LIENS HYPERTEXTES ET HYPERMÉDIAS

## ● Principe

La technique des hyperliens est à la base de la navigation sur le Web et de son interactivité (surf) car les hyperliens permettent de se déplacer rapidement à travers l'information.

Tant une portion de texte (hypertexte) qu'une image (hypermédia) peuvent servir de bouton (c'est-à-dire ce sur quoi on clique).

De manière générale, un hyperlien est signalé dans la fenêtre de navigation par un changement d'apparence du curseur (doigt pointé) lorsqu'on le survole avec la souris.

Par défaut, les liens hypertextes sont [bleus et soulignés](#); les liens hypermédiats (images) sont entourés d'une bordure bleue.

La couleur bleue est remplacée par le violet lorsque la cible a déjà été visitée précédemment (l'url de la cible est enregistrée dans l'historique de navigation).

## ● Les cibles des hyperliens

Les hyperliens peuvent mener à **différents types de cibles** :

- Une page Web
- Un endroit précis dans une page Web (ancrage ou signet)
- un fichier, une ressource ou un service utilisant un autre protocole que le HTTP (ftp://www.yahoo.fr/x.exe, telnet://biblio.polymtl.ca, mailto:moi@net.be)

Les cibles peuvent se trouver :

- Sur le même ordinateur (cible locale) ou sur un ordinateur différent (cible externe)
- Dans le même répertoire que la page d'origine ou dans un répertoire différent

## ● En pratique

La syntaxe générale des liens est très simple.

Lien	<code>&lt;a&gt;</code> Texte ou image servant de « bouton » <code>&lt;/a&gt;</code>
<b>Attribut obligatoire</b> : La cible du lien <code>href="adresse et nom de la cible"</code>	<a href="#">Un lien classique</a>
Affiche un lien classique, bleu et souligné et vous renvoie vers la cible placée dans l'attribut obligatoire href quand vous cliquez dessus.	

**Attribut facultatif** : Le titre du lien  
*title="informations sur le lien"*

Définit un titre pour le lien. Ce titre apparaît, dans certains navigateurs, sous forme de bulle lorsque l'on laisse le curseur immobile au-dessus du lien.

title peut être utilisé sur les liens, pour apporter une information nécessaire à la bonne navigation du visiteur et supplémentaire à l'intitulé du lien. C'est un attribut facultatif, et très rarement nécessaire. Un titre reprenant à l'identique l'intitulé du lien est totalement inutile et redondant. Il ne doit pas être utilisé sur les images (sauf cas extrêmement spécifiques).

**Attribut facultatif** : L'ouverture d'un lien dans une autre fenêtre  
*target="\_blank, \_self (par défaut)"*

Définit la fenêtre dans laquelle le lien va s'ouvrir, si vous n'indiquez pas cet attribut, le lien s'ouvrira dans la même fenêtre (c'est recommandé) tandis qu'en indiquant \_blank, il s'ouvrira soit dans une nouvelle fenêtre ou dans un nouvel onglet, selon les préférences indiquées dans votre navigateur.

[Ohmygoodweb.com](http://Ohmygoodweb.com)

Le meilleur du Web

## L'adressage des hyperliens

L'adressage des cibles peut être désigné de manière absolue ou relative.

Pour les raisons décrites précédemment, on privilégiera autant que possible l'adressage relatif.

Pour rappel, l'adressage relatif consiste à désigner la localisation du fichier cible sur base de l'adresse du fichier d'origine.

## ● Les liens hypermedia

Quel est le code à utiliser pour qu'une image serve de lien (pour qu'elle soit cliquable) ?

C'est assez simple, vous combinez 2 balises que nous avons vues :

```
<a href="nom et adresse de la cible">  
  
</a>
```

Rappelons que les images cliquables sont entourées, par défaut, d'une bordure bleue... bordure qu'il est possible de supprimer en spécifiant une bordure nulle sur la balise <img> :

```
<a href="nom et adresse de la cible">  
  
</a>
```

## ● Les ancrés

Lorsqu'une page est assez longue (un glossaire par exemple), il peut être utile d'envoyer directement l'internaute à un endroit précis de cette page, c'est-à-dire à hauteur de la position d'une ancre (point d'ancrage ou signet).

Il est possible de créer un lien vers une ancre située dans la page courante ou dans une autre page.

En pratique, pour éviter les erreurs d'adressage, il est préférable de créer l'ancre avant de s'en servir comme cible pour l'hyperlien.

Lien d'ancre	<a> Texte ou image servant de « bouton » </a>
<p><b>Attribut facultatif :</b> La cible du lien sur la même page (mais à un autre endroit sur cette page) ou sur une autre page (à un endroit précis également). <i>id="nom de l'ancre"</i></p> <p>Ensuite, pour créer un lien vers cette ancre, rien de plus simple, vous utilisez la balise &lt;a&gt; avec son attribut obligatoire <i>href</i> qui aura pour valeur le nom que vous aurez donné à votre ancre, mais précédé de # comme ceci :</p> <pre>&lt;a href="#nom de l'ancre"&gt;...&lt;/a&gt;</pre> <p>Vous serez alors amené à l'endroit où vous avez placé l'ancre sur la même page.</p> <p>Vous utiliserez cette syntaxe :</p> <pre>&lt;a href="nom de la page.html#nom de l'ancre"&gt;...&lt;/a&gt;</pre> <p>Pour arriver sur une autre page que la page sur laquelle se trouve le lien, toujours à un endroit précis.</p>	<p><a href="#">Un lien classique</a></p>

## ● Les liens vers une adresse mail

Grâce à la valeur `mailto`, il est possible de déclencher l'ouverture d'un client mail ou un webmail avec l'adresse du destinataire préalablement encodée.

*Exemple :*

```
<a href="mailto:webmaster@domaine.com">Ecrire au webmaster</a>
```

La commande peut éventuellement être complétée pour définir également (au choix) un sujet, le texte du mail ou un destinataire en copie conforme...

```
<a href="mailto:webmaster@domaine.com?subject=sujet du message&body=corps du message&cc=destinataire_en_copie@domaine.com">Ecrire au webmaster</a>
```

# LA COULEUR DU TEXTE ET DES HYPERLIENS

## ● La couleur du texte

Couleur du texte	<code>&lt;body&gt;...&lt;/body&gt;</code> ou <code>&lt;font&gt;...&lt;/font&gt;</code>
<b>Pour la balise &lt;body&gt; :</b> <b>Attribut facultatif :</b> La couleur du texte de toute la page change <i>text="#RRVVBB ou une couleur équivalente en anglais"</i>	Un texte qui a changé de couleur
<b>Pour la balise &lt;font&gt; :</b> <b>Attribut facultatif :</b> La couleur du texte de ce qui se trouve dans la balise <font> change <i>color="#RRVVBB ou une couleur équivalente en anglais"</i>	

## ● La couleur des liens, des liens visités et des liens actifs

Couleur des liens	<code>&lt;body&gt;...&lt;/body&gt;</code>
<b>Attribut facultatif :</b> La couleur des liens de la page est modifiée <i>link="#RRVVBB ou une couleur équivalente en anglais"</i>	<u>Un lien qui a changé de couleur avec link</u>
<b>Attribut facultatif :</b> La couleur des liens visités de la page est modifiée (par défaut elle est mauve) <i>vlink="#RRVVBB ou une couleur équivalente en anglais"</i>	<u>Un lien qui a changé de couleur visitée avec vlink</u>
<b>Attribut facultatif :</b> La couleur des liens cliqués de la page est modifiée <i>alink="#RRVVBB ou une couleur équivalente en anglais"</i>	<u>Un lien qui a changé de couleur au click avec alink</u>

# LES TABLEAUX

## ● Principes

Les balises et attributs destinés à créer des tableaux ont été intégrés dans l'HTML dans le but de pouvoir réaliser des tableaux statistiques, d'aligner des chiffres, ...

Mais rapidement, des webdesigners comme David Siegel ont découvert que l'on pouvait aussi détourner les tableaux de cet usage et s'en servir comme outils de mise-en-page pour :

- positionner des éléments à un emplacement précis sur la page, horizontalement ou verticalement
- créer des colonnes de texte et contrôler leur largeur
- aligner des éléments
- créer des bordures
- ...

**Un tableau est composé de lignes et de colonnes qui forment les cellules du tableau.**

Les tableaux reposent sur la combinaison de seulement trois balises de base.

Néanmoins, la création d'un tableau doit respecter une syntaxe et un ordre précis : en HTML, les tableaux sont décrits, cellule par cellule, de gauche à droite et de haut en bas.

Tableaux	<code>&lt;table&gt;...&lt;/table&gt;</code>
Début et fin de tableau	
Définition d'une ligne de tableau	<code>&lt;tr&gt;...&lt;/tr&gt;</code>
Début et fin d'une ligne d'un tableau	
Définition d'une colonne de tableau	<code>&lt;td&gt;...&lt;/td&gt;</code>
Début et fin d'une colonne d'un tableau	
Définition d'une en-tête de colonne de tableau	<code>&lt;th&gt;...&lt;/th&gt;</code>
Début et fin d'une en-tête de colonne d'un tableau	

## ● Comportements par défaut

- A aucun moment, le contenu d'une cellule ne sera caché, masqué, amputé, ... c'est toujours le tableau qui s'adapte.
- Ainsi, la largeur d'une cellule dépend de son contenu. Plus il y a du contenu, plus elle s'élargit. Ce n'est que lorsqu'une cellule a atteint sa largeur maximale (limite de la page) que le contenu passe à la ligne.
- Toutefois, si l'on souhaite empêcher le retour à la ligne même lorsque la largeur maximale du tableau est atteinte, on peut utiliser l'attribut `nowrap="nowrap"` à l'intérieur de la balise `<td>`.
- Il est possible de « bloquer » la largeur minimale d'une cellule en y insérant une image. Une image se trouvant dans une cellule ne pouvant pas être « passée à la ligne », la largeur de la cellule correspondra au minimum à la largeur de l'image.
- La cellule la plus large d'une colonne impose sa largeur aux cellules qui se trouvent à la verticale en-dessous et au-dessus d'elle.
- De même, la hauteur d'une cellule dépend de son contenu. Plus il y a du contenu, plus elle s'agrandit en hauteur.
- La hauteur de la cellule correspondra à la taille de l'image qui y serait placée.
- La cellule la plus haute impose sa hauteur aux cellules qui se trouvent à l'horizontale, à sa gauche et à sa droite.
- Enfin, la largeur d'un tableau correspond à la largeur cumulée de ses cellules.
- Un tableau ne s'élargit jamais au point de dépasser la largeur de la page.
- La hauteur d'un tableau correspond à la hauteur cumulée de ses cellules.
- Si la hauteur d'un tableau dépasse la hauteur de la fenêtre du navigateur, un ascenseur apparaît.

## ● Les attributs facultatifs des tableaux

### Tableaux

**Attribut facultatif** : Bordure du tableau  
`border="xxx"`

Modifie l'épaisseur par défaut (0) de la bordure extérieure du tableau. Mais les filets qui constituent les bordures intérieures ont toujours une épaisseur d'un pixel (sauf dans le cas d'une bordure invisible).

**Attribut facultatif** : Bordure du tableau  
`cellspacing="xxx"`

Modifie l'écartement par défaut (2) entre les cellules du tableau, c'est-à-dire « l'épaisseur » des bordures internes du tableau.

**Attribut facultatif** : Bordure du tableau  
`cellpadding="xxx"`

Modifie l'enrobage par défaut (1) des cellules du tableau, c'est-à-dire l'espace entre le contenu et la bordure interne de chaque cellule.

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

Avec une bordure de 1 :

1	2
3	4

... avec un cellspacing de 10 et un border de 1 :

1	2
3	4

... avec un cellpadding de 10 et un border de 1 :

1	2
3	4

**Attribut facultatif : Alignement du tableau**

*align="left, center ou right"*

Modifie l'alignement horizontal par défaut (gauche) du tableau dans la page. L'alignement ne s'applique pas au contenu des cellules.

**Attribut facultatif : Hauteur du tableau**

*height="xxx"*

Modifie la hauteur par défaut du tableau. La valeur de cet attribut s'exprime en pixels ou en pourcentage (%) de hauteur disponible.

**Attribut facultatif : Largeur du tableau**

*width="xxx"*

Modifie la largeur par défaut du tableau. La valeur de cet attribut s'exprime en pixels ou en pourcentage (%) de la largeur disponible.

Lorsque l'attribut `width` est utilisé pour fixer la largeur d'un tableau, le navigateur effectue un arbitrage entre le comportement général (la largeur d'une cellule dépend de son contenu) et ce paramètre dont il essaye de tenir compte au maximum.

Dans les cas « critiques », le comportement général l'emporte toujours.

Centré à droite :

1	2
3	4

Avec une hauteur de 100 :

1	2
3	4

Avec une largeur de 100 :

1	2
3	4

**Remarque**

Les cellules d'un tableau peuvent contenir tous les éléments Html déjà passés en revue soit :

- du texte
- des images
- des liens
- des arrière-plans
- ... et même des tableaux (eh oui!)

## ● Les attributs facultatifs des colonnes de tableaux

Colonnes (ou cellules) de tableaux

**Attribut facultatif** : Hauteur de colonne (ou de cellule)  
*height*="xxx"

Modifie la hauteur par défaut de la cellule. La valeur de cet attribut s'exprime en pixels ou en pourcentage (%) de la hauteur du tableau.

**Attribut facultatif** : Largeur de colonne (ou de cellule)  
*width*="xxx"

Modifie la largeur par défaut de la cellule. La valeur de cet attribut s'exprime en pixels ou en pourcentage (%) de la largeur du tableau.

Lorsque l'attribut **width** est utilisé pour fixer la largeur d'une cellule, le navigateur effectue un arbitrage entre :

1. le comportement général : la largeur d'une cellule dépend de son contenu
2. la valeur de l'attribut width dont il essaye de tenir compte au maximum
3. le cas particulier des autres cellules
4. le cas du tableau général

**Attribut facultatif** : Fusion de colonnes (horizontalement)  
*colspan*="xxx"

Spécifie le nombre de colonnes (cellules) fusionnées

**Attribut facultatif** : Fusion de lignes (verticalement)  
*rowspan*="xxx"

Spécifie le nombre de lignes (cellules) fusionnées

<td>...</td>

Avec un height de 50 sur la première colonne de la première ligne :

1	2
3	4

Avec un width de 50 sur la première colonne de la première ligne :

1	2
3	4

Avec un colspan de 2 sur les deux premières colonnes :

1	2
3	4

Avec un rowspan de 2 sur les deux premières colonnes :

1	2
1	4

### Remarque

La règle en matière de fusion de cellule est qu'il ne faut jamais décrire deux fois la même cellule, que ce soit dans le cas de fusions horizontales (colonnes) ...

```
<table border="1">
  <tr>
    <td colspan="3">cellule 1</td>
  </tr>
  <tr>
    <td>cellule 1</td>
    <td>cel 2</td>
    <td>3</td>
  </tr>
</table>
```

... ou dans le cas de fusions verticales (lignes)

```
<table border="1">
  <tr>
    <td rowspan="2">cellule 1</td>
    <td>cel 2</td>
    <td>3</td>
  </tr>
  <tr>
    <td>cel 2</td>
    <td>3</td>
  </tr>
</table>
```

## ● Les attributs facultatifs des colonnes ET des lignes de tableaux

Colonnes et lignes de tableaux

```
<td>...</td>
<tr>...</tr>
```

**Attribut facultatif :** Alignement horizontal  
*align="left, center, right ou justify"*

Modifie l'alignement horizontal par défaut (gauche) du contenu d'une cellule...

En l'appliquant à une balise <td>, cela n'aura effet que dans la cellule en question.

En revanche, si vous l'appliquez sur un <tr>, cela prendra effet sur toutes les cellules de la ligne en question.

Avec un align right sur la première cellule (<td>) :

1	2
3	4

Avec un align center sur la première ligne (<tr>) :

1	2
3	4

**Attribut facultatif :** Alignement vertical  
*valign="top, middle ou bottom"*

Modifie l'alignement vertical par défaut (centré) du contenu d'une cellule...

Avec un valign bottom sur la 2<sup>e</sup> ligne :

1	2
3	4

# • Les attributs facultatifs des tableaux, des colonnes ET des lignes

Tableaux, colonnes et lignes

```
<table>...</table>  
<td>...</td>  
<tr>...</tr>
```

**Attribut facultatif :** Couleur d'arrière-plan  
*bgcolor="#RRVVBB ou un code couleur correspondant en anglais"*

Modifie la couleur par défaut (transparent) du tableau, de la ligne ou de la cellule.

La couleur spécifiée pour <table> s'applique sur toute la surface du tableau, y compris entre les cellules.

La couleur spécifiée pour <tr> et <td> ne s'applique pas entre les cellules (cellspacing).

**Attribut facultatif :** Image d'arrière-plan  
*background="URL image"*

Affiche une image d'arrière-plan au tableau, à la ligne ou à la cellule.

Attention, cet attribut n'est pas reconnu par tous les parseurs.

Le motif spécifié pour <table> s'applique sur toute la surface du tableau, y compris entre les cellules.

Le motif spécifié pour <tr> et <td> ne s'applique pas entre les cellules (cellspacing).

Avec un align right sur la première cellule (<td>) :

1	2
3	4

Avec un align center sur la première ligne (<tr>) :

1	2
3	4

Avec un valign bottom sur la 2<sup>e</sup> ligne :

1	2
3	4

## Comportements par défaut

Il est impossible de spécifier une police, une taille ou une couleur de texte de manière globale pour toutes les cellules d'un tableau. Ces caractéristiques typographiques doivent être spécifiés pour chaque cellule séparément.

# TABLEAU RÉCAPITULATIF DES BALISES, DE LEURS ATTRIBUTS ET DE LEUR TYPE

I = de type In-Line | B = de type Block

**Valeur par défaut de la balise**

A-F = balise auto-fermante (O = oui, N = non (ce choix nécessite donc une balise de fermeture))

I	B	A-F	Balises	Attributs	Valeurs admises	Description
		N	<html>			Définit le début et la fin d'un document.
		N	<head>			Définit la zone d'en tête de votre page.
		N	<title>			Définit le titre de la page en cours.
		N	<body>	Bgcolor	#RRVVBB ou Couleur anglais	Modifie la couleur d'arrière-plan par défaut (blanc).
				Background	Adresse et nom de l'image	Définit une image comme arrière-plan de la page.
				Text	#RRVVBB ou Couleur anglais	Modifie la couleur par défaut du texte de la page.
				Link	#RRVVBB ou Couleur anglais	Modifie la couleur par défaut de tous les liens de la page.
				Vlink	#RRVVBB ou Couleur anglais	Modifie la couleur par défaut de tous les liens visités de la page.
				Alink	#RRVVBB ou Couleur anglais	Modifie la couleur par défaut de tous les liens activés de la page.
		N	<!-- ... -->			Insertion d'un commentaire
<b>Les balises de texte</b>						
x		N	<strong>			Le texte balisé s'affiche en « gras ».
x		N	<em>			Le texte balisé s'affiche en « italique ».
x		N	<font>	size	1, 2, 3, 4, 5, 6, 7 +1 à +4 -1 à -4	Le texte balisé s'affiche à la taille précisée.
				face	Verdana Georgia <b>Times New Roman</b> Courier New Courier Arial Tahoma Trebuchet MS Arial Black Palatino Linotype Book Antiqua Lucida Sans Unicode Lucida Consolas Comic Sans MS	Le texte balisé s'affiche dans la première police disponible.
				color	#RRVVBB ou Couleur anglais	Le texte balisé s'affiche dans la couleur spécifiée.
	x	N	<p>	align	<b>left</b> right center justify	Cet attribut permet, respectivement, d'aligner à gauche (par défaut), à droite, de centrer ou de justifier le paragraphe.
	x	O	 			Retour chariot après le  .
I	B	A-F	Balises	Attributs	Valeurs admises	Description

I	B	A-F	Balises	Attributs	Valeurs admises	Description
	x	N	<div>	align	left right center justify	La balise <div> sert à créer des « divisions » dans la page.
x		N	<sub>			Affiche le texte balisé en indice.
x		N	<sup>			Affiche le texte balisé en exposant.
	x	N	<blockquote>			Définit le texte balisé comme étant un bloc de citation.
	x	N	<pre>			
	x	N	<h1> <h2> <h3> <h4> <h5> <h6>	align	left right center justify	Le texte balisé est défini comme étant un titre. <h1> est le titre le plus fort. <h6> est le titre le moins fort.
<b>Les listes à puce</b>						
	x	N	<ul>	type	disc circle square	Définit le texte balisé comme une liste à puces, c'est-à-dire une liste non-ordonnée. L'attribut type donne le style de la puce.
	x	N	<ol>	Type	A a I i <b>1</b>	Définit le texte balisé comme une liste ordonnée. L'attribut type donne le style de la puce.
				Start	...	Définit la valeur de départ de la liste.
	x	N	<li>			Le texte balisé avec <li> est défini comme étant un élément de la liste.
<b>Les filets horizontaux</b>						
	x	O	<hr>	Width	... ou ...%	Insère un filet horizontal. L'attribut width en définit la largeur.
				Size	...	Définit l'épaisseur du filet.
				Align	left right <b>center</b>	Définit l'alignement horizontal du filet.
<b>Les images</b>						
x		O	<img>	Src	Adresse et nom de l'image	Permet d'insérer une image dans une page. L'attribut src est obligatoire.
				Width	...	Définit la largeur de l'image
				Height	...	Définit la hauteur de l'image
				Alt	Texte s'affichant à la place de l'image	Définit un texte alternatif, au cas où l'image ne se chargerait pas dans le navigateur.
				Border	...	Définit l'épaisseur en pixels de la bordure entourant l'image.
				Align	top middle <b>bottom</b> left right	Définit la position de la première ligne du paragraphe par rapport à l'image qui précède.
				Vspace	...	Définit l'espacement en pixels au-dessus et en-dessous de l'image.
				Hspace	...	Définit l'espacement en pixels à gauche et à droite de l'image.
I	B	A-F	Balises	Attributs	Valeurs admises	Description

I	B	A-F	Balises	Attributs	Valeurs admises	Description
<b>Les hyperliens</b>						
x		N	<b>&lt;a&gt;</b>	Href	Adresse et nom de la cible	Pour créer un lien, l'attribut href est obligatoire.
				Title	...	Définit un titre pour le lien.
				Name	Nom de l'ancre	Création d'une ancre
<b>Les tableaux</b>						
	x	N	<b>&lt;table&gt;</b>	Border	...	Modifié l'épaisseur par défaut de la bordure extérieure du tableau.
				cellspacing	...	Modifie l'écartement par défaut entre les cellules du tableau.
				cellpadding	...	Modifie l'enrobage par défaut des cellules du tableau, c'est-à-dire l'espace entre le contenu et la bordure interne de chaque cellule.
				Align	<b>left</b> right center	Modifie l'alignement horizontal par défaut du tableau dans la page.
				Height	...	Modifie la hauteur par défaut du tableau. La valeur de cet attribut s'exprime en pixels ou en pourcentage (%) de hauteur disponible.
				Width	...	Modifie la largeur par défaut du tableau. La valeur de cet attribut s'exprime en pixels ou en pourcentage (%) de la largeur disponible.
				Bgcolor	#RRVVBB ou Couleur anglais	Modifie la couleur par défaut (transparent) du tableau.
				background	URL de l'image	Affiche une image d'arrière-plan au tableau.
	x	N	<b>&lt;tr&gt;</b>	Align	<b>left</b> right center	Modifie l'alignement horizontal par défaut (gauche) du contenu d'une ligne complète.
				Valign	top <b>middle</b> bottom	Modifie l'alignement vertical par défaut (centré) du contenu d'une ligne complète.
				Bgcolor	#RRVVBB ou Couleur anglais	Modifie la couleur par défaut (transparent) de la ligne.
				background	URL de l'image	Affiche une image d'arrière-plan sur toute la ligne.
x		N	<b>&lt;td&gt;</b>	Height	...	Modifie la hauteur par défaut de la cellule. La valeur de cet attribut s'exprime en pixels ou en pourcentage (%) de la hauteur du tableau.
				Width	...	Modifie la largeur par défaut de la cellule. La valeur de cet attribut s'exprime en pixels ou en pourcentage (%) de la largeur du tableau.
				Colspan	...	Spécifie le nombre de cellules (colonnes) fusionnées
				Rowspan	...	Spécifie le nombre de cellules (lignes) fusionnées
				Align	<b>left</b> right center	Modifie l'alignement horizontal par défaut (gauche) du contenu d'une cellule.
				Valign	top <b>middle</b> bottom	Modifie l'alignement vertical par défaut (centré) du contenu d'une cellule.
I	B	A-F	Balises	Attributs	Valeurs admises	Description

I	B	A-F	Balises	Attributs	Valeurs admises	Description
				Bgcolor	#RRVVBB ou Couleur anglais	Modifie la couleur par défaut (transparent) d'une cellule.
				background	URL de l'image	Affiche une image d'arrière-plan sur la cellule.