



## Le langage HTML

Formation

**Développeur Web et Web Mobile**

Module : 03

**Développer la partie front-end d'une application web**

Séquence : 01

**Réaliser une interface utilisateur web statique et adaptable**

Séance : 01

**Développer des pages web statiques en utilisant Html et Css**

Libellé réduit :	HTML 5
Type de document :	Support de cours
Version :	1
Auteur :	Didier Bonneau
Centre afpa :	Créteil
Date de création :	26/01/2016
Date de mise à jour :	06/09/2022



## Sommaire

## Contenu

I	Introduction .....	4
A.	Qu'est ce que le « world wide web » ? .....	4
B.	Qu'est ce que « html » ? .....	4
C.	Architecture html .....	5
1.	Structure d'un document .....	5
2.	Les commentaires .....	7
3.	Les couleurs .....	7
4.	Le rendu des balises .....	8
II	Les métas balises .....	9
A.	Pourquoi les métas balises ? .....	9
B.	Quelques balises meta .....	9
1.	Balise meta relative au document .....	9
2.	Des infos sur votre site .....	10
3.	Des infos sur vous .....	10
4.	Des infos sur la création .....	10
5.	Orienter les robots .....	11
6.	Encore plus .....	12
III	Les balises de base .....	13
A.	Les balises conteneurs <div> (block) et <span> (inline) .....	13
1.	La balise <DIV> (type block) .....	13
2.	La balise <span> (type inline) .....	13
3.	Les nouvelles balises html5 (type block) .....	13
D.	Les paragraphes : <p> (type block) et   .....	13
E.	Les titres : <h1> à <h6> (type block) .....	14
F.	Les lignes horizontales : <hr> (block) .....	14
G.	Mise en forme de caractères .....	15
1.	Mise en forme logique (inline) .....	15
2.	Mise en forme physique (inline) .....	16
3.	Les polices de caractères .....	16
4.	Les caractères accentués .....	16
5.	Les caractères spéciaux .....	17
H.	Les abréviations et acronymes .....	17
IV	Les listes .....	18
A.	Liste non ordonnée <ul> (block) .....	18

B.	Liste ordonnée <ol> (block) .....	18
C.	Liste avec définition <dl> (block).....	19
D.	Combiner les types de listes .....	19
E.	Créer vos propres puces .....	19
V	Les images <img> (Type inline).....	21
A.	La source de l'image.....	21
B.	Préciser la taille .....	21
C.	Renseigner l'image .....	22
D.	Mettre un bord .....	22
E.	Positionner les images.....	22
1.	Espacer les images .....	22
2.	Aligner les images .....	22
F.	Prévisualisation.....	23
VI	Les liens .....	24
A.	Introduction.....	24
B.	La balise de base <a> (INLINE) .....	24
1.	Créer un lien à l'intérieur d'une même page.....	24
2.	Créer un lien externes vers un autre site web .....	24
C.	Les infos-bulles.....	25
D.	Les e-mails .....	25
E.	Les autres protocoles .....	25
F.	Précision sur l'attribut « target » .....	25
VII	Les animations vidéos, zones réactives.....	27
A.	Animations vidéos.....	27
1.	Le défilement de texte .....	27
2.	Intégration d'une vidéo .....	27
3.	Zones réactives .....	28
VIII	Les tableaux.....	30
A.	Création d'un tableau.....	30
1.	Les balises de base.....	30
2.	Légende du tableau.....	30
3.	Légende des colonnes .....	30
4.	Nouvelles balises sémantiques .....	30
B.	Taille et Alignement .....	31
1.	Taille du tableau et des cellules .....	31
2.	Alignement dans les cellules .....	31
C.	Bordures, Espacements et Marges.....	32
1.	Bordures du tableau .....	32

2.	Espacement entre les cellules.....	33
3.	Marges internes aux cellules .....	33
4.	Récapitulatif de mise en page .....	33
D.	Fusionner lignes et colonnes .....	34
E.	Couleur et Image de fond .....	34
IX	Les cadres .....	35
A.	Introduction.....	35
B.	Les balises de base .....	35
1.	Architecture des cadres.....	35
2.	Définition des cadres .....	36
C.	Imbrication de cadres .....	37
D.	Mise en page .....	37
1.	Bordures.....	37
2.	Marges internes.....	37
E.	Récapitulatif.....	37
X	Les cadres flottants : les iframes .....	39

# I INTRODUCTION

## A. QU'EST CE QUE LE « WORLD WIDE WEB » ?

Le World Wide Web (Web) est un réseau de ressources et d'informations. Le Web repose sur trois mécanismes pour rendre ces données facilement disponibles et le plus largement possible :

1. un système de nommage uniforme pour leur localisation sur le Web (par exemple, les URL<sup>1</sup>) ;
2. des protocoles, pour accéder à des ressources nommées dont on possède l'adresse (par exemple, HTTP<sup>2</sup>) ;
3. l'hypertexte, pour faciliter la navigation entre ces ressources (par exemple, HTML<sup>3</sup>).

Toutes les technologies du web s'appuient sur des recommandations du W3C<sup>4</sup> qui est un organisme de standardisation à but non-lucratif fondé par le MIT <sup>5</sup>(USA) en 1994. En 1995 l'INRIA<sup>6</sup> (France) est désignée pour la partie Européenne remplacée en 2003 par l'ERCIM<sup>7</sup> (Europe).

## B. QU'EST CE QUE « HTML » ?

Publier de l'information en vue d'une distribution mondiale nécessite l'usage d'un langage universellement compréhensible, d'une sorte de « langue maternelle » acceptée par tous les ordinateurs. Le langage de publication utilisé sur le World Wide Web est le langage de balisage hypertexte HTML.

HTML donne aux auteurs les moyens de :

- publier des documents en ligne possédant des titres, du texte, des tables, des listes, des photos, etc.
- ramener des informations en ligne via des liens hypertextes avec un clic ;
- concevoir des formulaires pour conduire des transactions avec des services distants, dans le but de rechercher des informations, de faire des réservations, de prendre des commandes, etc.
- inclure des feuilles de calculs, des séquences vidéos ou sonores et d'autres applications, directement dans les documents.

C'est un langage de balisage, qui a comme objectif de représenter de l'information. Ces balises sont interprétables par les différents navigateurs (Internet Explorer, Firefox, Chrome, Safari, Opera ...) et permettent l'affichage de l'information désirée.

Le langage a évolué depuis sa création et aujourd'hui il en est à sa version 5.

C'est un langage très permissif mais on s'efforcera de respecter les recommandations du W3C en fonction du type de document html que l'on est en train d'écrire.

<sup>1</sup> URL : Uniform Resource Locator

<sup>2</sup> HTTP : Hyper Text Markup Language

<sup>3</sup> HTML : Hyper Text Meta Language

<sup>4</sup> W3C : World Wide Web Consortium

<sup>5</sup> MIT : Massachusetts Institute of Technology

<sup>6</sup> INRIA : Institut National de Recherche en Informatique et en Automatique

<sup>7</sup> ERCIM : European Research Consortium for Informatics and Mathematics

## C. ARCHITECTURE HTML

### HTML : un langage de balises :

HTML est un langage description de pages à l'aide de balises. Toutes les instructions sont exprimées dans des **balises** (ou **tags** ou **marker**), qui sont encadrées par des délimiteurs (signe inférieur < et signe supérieur >). Chaque balise doit être **ouverte** et **fermée** (sauf rares exceptions). Les balises de fermeture sont les mêmes que celles d'ouverture, mais sont précédées d'un **slash** (/).

Exemple : `<h1> contenu de la balise </h1>`

Une balise peut accepter un ou plusieurs **attributs** qui se placent à l'intérieur de la balise d'ouverture. Ils sont placés après le nom de la balise et séparés les uns des autres par un espace. Un attribut comporte un nom et une valeur séparés par le signe **égal** « = ».

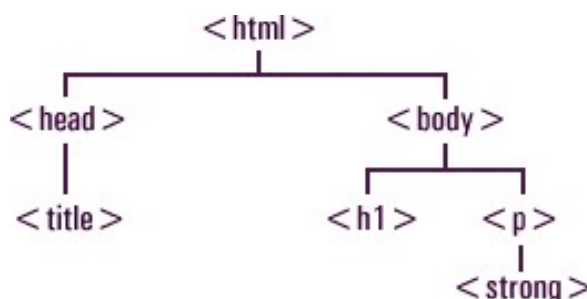
Exemple : ``

En HTML, la casse (minuscule ou majuscule) n'a aucune importance. **Il est cependant conseillé d'écrire les balises en minuscules et les valeurs d'attribut entre guillemets droits double (double quotes)** afin de respecter le standard XHTML<sup>8</sup> de votre code HTML. De même, lorsque vous écrivez un document HTML, pensez à respecter l'indentation et veillez aux balises d'ouverture et de fermeture afin d'obtenir des documents lisibles pour le programmeur. En effet, le navigateur ne prend pas en compte les retours à la ligne ni une suite d'espaces qu'il remplacera par un espace unique.

### 1. Structure d'un document

Un document HTML se compose de plusieurs parties :

- ✓ une première ligne DOCTYPE contenant les informations de la version HTML et précisant le fichier de définition DTD<sup>9</sup> correspondant.
- ✓ puis un élément racine **<html>**, qui comprend :
  - une section en-tête déclarative (enclose par la balise **<head>**)
  - un corps, qui comporte le contenu effectif du document (enclos par l'élément **<body>**).



Les balises html sont incluses les unes dans les autres sans chevauchement.

<sup>8</sup> XHTML : eXtended Hyper Text Markup Language

<sup>9</sup> DTD : Document Type Definition

Voici un exemple de document HTML simple :

```
<!DOCTYPE html>
<html>
  <head>
    <title>Mon premier document HTML</title>
  </head>
  <body>
    <p>Bonjour tout le monde !</p>
  </body>
</html>
```

#### a) Balise « DOCTYPE »

Avant de commencer à écrire le contenu de votre page html, chaque page de votre projet doit comporter une balise « doctype » qui permet au navigateur de définir le standard HTML que vous avez utilisé.

Commande	Version HTML
<!doctype html public "-//ietf//dtd html 3.0//en">	HTML 3.0
<!doctype html public "-//w3c//dtd html 3.2//en">	HTML 3.2 (1997)
<!doctype html public "-//W3C//DTD HTML 4.01Strict//EN" "http://www.w3.org/TR/html4/strict.dtd">	HTML 4.01 strict (1999)
<!doctype html public "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">	HTML 4.01 en transition
<!doctype html public "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">	HTML 4.01 en cadre
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">	XHTML 1.0 strict (2000)
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">	XHTML 1.0 en transition
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">	XHTML 1.0 en cadre
< !DOCTYPE html>	HTML 5 (2008)

#### b) La racine : la balise « html »

Juste après la balise « doctype », vous devez entrer la balise **<html>** qui est la racine de votre page. Toute les autres balises seront contenu entre cette balise et la balise de fermeture **</html>** se trouvant à la fin de votre document.

Afin de préciser la langue du document html, on utilise l'attribut « lang » qui peut se placer sur n'importe quelle balise en html5. (en html 4.01 seules les balises <html>, <base>, <br>, <frame>, <frameset>, <hr>, <iframe>, <param>, and <script> sont autorisées).

Le plus souvent on place cet attribut sur la balise html :

```
<html lang="fr">
```

En xhtml, un attribut supplémentaire « xmlns » permet la déclaration d'un « name space xml » et est obligatoire :

```
<html xmlns="http://www.w3.org/1999/xhtml" lang="en" xml:lang="en">
```

### c) L'entête : la balise « head »

Le premier bloc de votre page est le bloc d'entête qui renfermera toutes les balises de directives pour votre navigateur. Ce bloc est symbolisé par les balises **<head>.....</head>**.

L'élément **<head>** contient des informations sur le document courant, tels que son titre, des mots-clés que les moteurs de recherche peuvent exploiter et qui permettront d'indexer (référencer) votre site, des définitions de style (voir CSS) et d'autres données qui ne sont pas considérées comme faisant partie du contenu du document.

### d) Le corps : la balise « body »

C'est dans ce bloc qu'apparaîtra toutes les balises de contenu de votre page. Il est symbolisé par les balises **<body>... </body>**.

C'est ici que vous allez intégrer titres, textes, images, tableaux, liens...

Le tag **<body>** a de nombreux attributs :

- **bgcolor** : Définit la couleur du fond de l'écran (css : background-color)
- **text** : Définit la couleur du texte (css : color)
- **link** : Définit la couleur des liens
- **vlink** : Définit la couleur des liens déjà visités
- **alink** : Définit la couleur des liens activés, c'est à dire quand le pointeur de la souris passe dessus
- **background** : Définit l'image (gif ou jpeg) à utiliser comme fond d'écran (css : background-image)
- **bgproperties** : Si définie comme "fixed", l'image de fond ne défile pas (css : background-attachment)
- **leftmargin** : Définit la largeur de la marge de gauche en pixels
- **topmargin** : Définit la largeur de la marge du haut en pixels

obsolète

Tous ces attributs ne sont plus admis en html5 et sont remplacés par leurs équivalents dans les feuilles de style css.

## 2. Les commentaires

Vous pouvez commenter votre code source sans que le navigateur affiche ce commentaire. Attention, ici l'abus de commentaires grossit le code source et augmente les temps de téléchargement. Cette instruction est aussi utilisée lors de l'écriture d'un script Javascript pour cacher le code aux navigateurs non compatibles.

Exemple : *<!-- Ceci est un commentaire. -->*

## 3. Les couleurs

Il y a trois façons de positionner la couleur des éléments en html :

1. En précisant la valeur par des mots clés  
Exemples : red, blue, yellow, ...
2. En donnant la valeur en hexadécimal : #rrvvbb  
Les composantes rr, vv, bb vont de 00 à FF



3. En utilisant la fonction `rgb(r, v, b)` qui attend les valeurs des différentes composantes comprises entre 0 et 255.

La combinaison des 3 couleurs peut ainsi créer plus de 16 millions de couleurs.

obsolète

Exemple : `<body text="red">`  
`<body text="#FF0000">`  
`<body text="rgb(255,0,0)">`

Couleur	Valeur RGB
Black (noir)	#000000
Silver (argent)	#C0C0C0
White (blanc)	#FFFFFF
Red (rouge)	#FF0000
Green (vert)	#008000
Lime (citron vert)	#00FF00
Olive (olive)	#808000
Yellow (jaune)	#FFFF00
Blue (bleu)	#0000FF

Une autre fonction `rgba(r, v, b, o)` admet un quatrième paramètre correspondant à l'opacité du composant. Une valeur 0 donne une opacité totale, une valeur 1 donne une transparence totale.

Là aussi les couleurs font partie du style et seront mises en css.

#### 4. Le rendu des balises

Les balises de contenu (présentes entre `<body>` et `</body>`) peuvent être considérées comme des boîtes qui ont deux types de rendus au niveau de l'écran suivant qu'elles sont de type « inline » ou « block ».

Les balises « inline » s'adaptent automatiquement à leur contenu et se positionnent à l'écran les unes derrière les autres. On dit qu'elles sont dans le « flux ».

Les balises « block » peuvent être dimensionnées en hauteur, largeur et se placent dans le flux les unes en dessous des autres. Par défaut leur largeur est la largeur maximum qui leur est autorisé en fonction de leur conteneur (éventuellement la page).

On verra qu'en changeant la valeur par défaut propriété « display », on pourra obtenir le rendu que l'on souhaite pour la page html.

## II LES METAS BALISES

### A. POURQUOI LES METAS BALISES ?

Un site web qui vit, qui bouge, c'est un site qui a de l'affluence et qui interagit avec l'internaute. Pour cela, faut-il encore qu'on « tombe » sur le site en question ! Et là, il n'y a pas de secret : seul un très bon référencement pourra vous permettre d'attirer à vous tous ceux qui se baladent nonchalamment sur le web.

Certaines d'entre elles sont importantes pour le *référencement naturel* ou *SEO* (*SEO* est l'acronyme de Search Engine Optimization).

Mais comme la plupart des outils de recherche sur internet sont des moteurs de recherche d'indexation automatique comme AltaVista (à distinguer des annuaires comme Yahoo), il faut préparer à l'avance ses pages à l'arrivée des robots de ces moteurs de recherche.

Car lorsqu'on soumet l'adresse de son site à un moteur, celui-ci l'enregistre puis envoie des « robots » (encore appelés *spiders*, *crawlers* ou agents) se balader sur vos pages et enregistrer les informations contenues dans les **métas balises** que vous aurez préalablement placées dans l'entête entre les balises <head> et </head> de votre code HTML. Grâce à ces balises vous pouvez commander les robots mais aussi influencer le navigateur. Elles permettent d'informer les robots sur certains détails du site, comme le langage du site, l'auteur, l'éditeur utilisé.

La syntaxe générale de la balise <meta> comporte un attribut **NAME** et un attribut **CONTENT**.

```
<meta name="Nom du tag" content="Attribut">
<meta http-equiv="Nom du tag" content="Attribut">
<meta rel="Nom du tag" content="Attribut">
```

On peut classer les balises méta suivant trois catégories :  
les indispensables, les facultatives et les obsolètes.

### B. QUELQUES BALISES META

#### 1. Balise meta relative au document

##### a) La langue du document : language (facultative)

Cette balise n'est pas prise en compte par les navigateurs.

```
<meta name="language" content="fr">
```

En html 4 et 5 la langue se code comme attribut de la balise html et est indispensable :

```
<html lang="fr-CA">
```

##### b) Le codage des caractères : le charset (indispensable)

Il est indispensable de préciser le codage des caractères du document. La syntaxe est différente suivant la version du html utilisé :

En HTML 4.01 :

```
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
```

En html5 un nouvel attribut « charset » est apparu :

*<meta charset="codage">*

Codage peut prendre tout un tas de valeurs dont les plus courants sont :

UTF-8 : codage des caractères en Unicode

ISO-8859-1 : codage des caractères en alphabet Latin

## 2. Des infos sur votre site

### a) La description de votre site : description (indispensable)

Il faut indiquer au moteur de recherche des informations relatives au contenu de votre site. Cette balise méta permet d'écrire une courte description (160 caractères max) de votre site qui sera affichée lorsque votre site se trouve dans les résultats d'une recherche (snippets) :

*<meta name="Description" content="courte description du site">*

Sachez que si votre description est trop longue (plus de 150 à 200 caractères), elle sera coupée par le moteur de recherche.

### b) Les mots clés : keywords (facultatif)

Un attribut qui était important est « *keywords* », qui permettait de lister quels mots clés utilisés par les moteurs de recherche :

*<meta name="Keywords" content="liste de mots clés séparés par des virgules.">*

A éviter car vous fournissez à tous vos concurrents les mots clés sur lesquels vous cherchez à vous positionner. Les moteurs de recherche ne l'utilisent plus que pour cibler la publicité qu'ils renvoi.

Elle n'intervient plus pour le référencement, il est préférable d'utiliser la balise méta « description ».

## 3. Des infos sur vous

### a) Le ou les noms du créateur du site : author (facultatif)

Il peut être intéressant de trouver l'auteur du site web dans les pages html :


*<meta name="author" content="Prénom NOM">*

## 4. Des infos sur la création

### a) Le copyright : copyright (facultatif)

Des informations de « copyright » peuvent être insérées avec l'attribut « copyright » :

*<meta name="copyright" content="&copy; 2015 Afpa.">*

	Auteur	Nom région	Formation	Date Mise à jour	Page 10 / 40
	Didier Bonneau	GRN164	DWWM	06/09/2022	121-Support-De-Cours_HTML5.docx

### b) Le générateur utilisé : generator (facultative)

De plus, il est de coutume d'indiquer quels ont été les outils de développement du site tels que les logiciels de traitement d'image, les navigateurs, etc... avec l'attribut « *generator* » :

```
<meta name="generator" content="liste des différents logiciels utilisés.">
```

Vous pouvez aussi indiquer la date de création ou de mise en ligne de votre page avec l'attribut « *date* » :

```
<meta name="date" content="date de création de votre page.">
```

## 5. Orienter les robots

### a) Donner des autorisations aux robots : robots (facultatif)

On peut autoriser ou non les robots à indexer notre site (suivre les liens des différentes pages), il faut utiliser l'attribut « robots » :

```
<meta name="robots" content="...", ..., ..." >
```

Il est possible de personnaliser les indications en fonction de chaque robot. Les plus connus : **googlebot** pour Google et **msnbot** pour Bing.

```
<meta name="googlebot" content="...", ..." />
```

La liste des « content » possibles, à espacer d'une virgule dans l'attribut content :

- **noindex** : empêche l'indexation de la page.
- **nofollow** : empêche le robot de suivre les liens de cette page.
- **nosnippet** : empêche l'affichage d'un extrait dans les résultats de recherche.
- **noarchive** : empêche Google d'afficher le lien « En cache » associé à une page.
- **unavailable\_after:[date]** : permet de préciser l'heure et la date exactes auxquelles l'exploration et l'indexation de cette page doivent cesser.
- **noimageindex** : permet d'indiquer que vous ne souhaitez pas faire apparaître votre page comme source d'une image apparaissant dans les résultats de recherche Google.

### b) La réindexation : revisit-after (facultative)

Vous pouvez aussi demander aux robots de réindexer automatiquement votre site après **n** jours :


```
<meta name="Revisit-after" content="n">
```

Aujourd'hui ignorée par les robots.

### c) La redirection et le rechargement: refresh (obsolete)

Pouvoir faire recharger une page périodiquement peut s'avérer très utile surtout si l'on affiche des bannières publicitaires, afin d'en changer souvent. Ainsi, il est possible d'ordonner au navigateur de recharger une page toutes les **n** secondes grâce à l'attribut « *http-equiv* » prenant la valeur « *refresh* » :

```
<meta http-equiv="refresh" content="n">
```

	Auteur	Nom région	Formation	Date Mise à jour	Page 11 / 40
	Didier Bonneau	GRN164	DWWM	06/09/2022	121-Support-De-Cours_HTML5.docx

On peut rediriger le navigateur vers un autre site différent, très utilisé lorsqu'on change d'hébergeur :

```
<meta http-equiv="refresh" content="n;URL souhaitée">
```

Non prise en compte par tous les navigateurs et considérée comme trompeuse pour l'internaute, elle est fortement déconseillée par le W3C.

## 6. Encore plus ...

Faire la loi aux caches : vous pouvez interdire aux navigateurs de conserver en mémoire-cache vos pages :

```
<meta http-equiv="pragma" content="no-cache">
```

```
<meta http-equiv="Cache-control" content="public/no-cache">
```

Date d'expiration : vous savez qu'il existe des ordinateurs qui, pour faciliter l'accès aux pages web, conserve dans leurs disques durs les pages déjà visitées et donc susceptibles d'être redemandées. Ainsi, lorsqu'on accède à une seconde reprise au même site, on a des chances de retomber sur les mêmes pages déjà visitées. Ainsi, lorsqu'on veut empêcher cela, on indique une date d'expiration qui, si elle est dépassée, ordonnera au proxy d'aller récupérer les vrais pages et de supprimer celles expirées. Attention, la date est au format anglophone :

```
<meta http-equiv="expires" content="Mon, 30 Jun 1999 14:30:00">
```

### III LES BALISES DE BASE

#### A. LES BALISES CONTENEURS <DIV> (BLOCK) ET <SPAN> (INLINE)

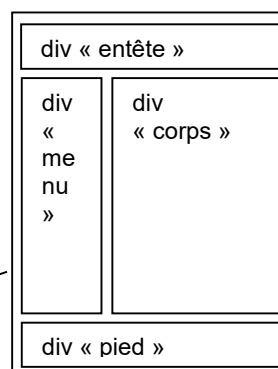
##### 1. La balise <DIV> (type block)

L'élément HTML <div> (signifiant division du document) est un conteneur générique permettant d'organiser le contenu mais qui ne représente rien de particulier. Il peut être utilisé pour grouper d'autres éléments afin de leur appliquer un style (en utilisant les attributs class ou id : voir css). Il doit être utilisé uniquement lorsqu'aucun autre élément sémantique (comme <article> ou <nav>) n'est approprié.

C'est en imbriquant des balises <div> que l'on peut organiser le contenu de nos pages (entête, menu, corps, pied).

Ne pas abuser des div, un bon compromis semble être de faire une div générale (qui aura la taille de votre page web) dans laquelle vous insérerez un « entête », un « menu », un « corps » et un « pied ».

div « générale »



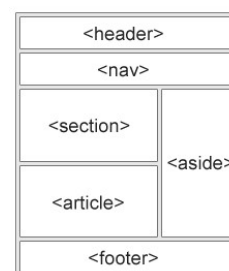
##### 2. La balise <span> (type inline)

L'élément HTML <span> est un conteneur générique de ligne pour les contenus phrasés, qui ne représente rien par lui-même. Il peut être utilisé pour grouper des éléments afin de les styler (en utilisant les attributs class ou id).

##### 3. Les nouvelles balises html5 (type block)

Des nouvelles balises sont apparues avec html5 apportant de la sémantique aux pages html :

- <header> : sert d'entête à la page
- <nav> : utilisée pour les menus de navigation
- <section> : définit une section
- <article> : définit un article de la section
- <aside> : définit un contenu ayant un rapport un peu éloigné de celui de l'article
- <footer> servent à définir le pied de page



#### D. LES PARAGRAPHES : <P> (TYPE BLOCK) ET <BR>

Dans un souci de lisibilité et de structuration, les documents doivent être découpés en paragraphes. En HTML, il existe deux balises qui permettent d'obtenir soit un saut de ligne, soit le début d'un nouveau paragraphe.

La balise **<p>** permet de commencer un nouveau paragraphe (double espacement), et la balise **<br>** effectue un retour à la ligne (simple espacement).

Les balises de fermeture **</p>** et **</br>** ne sont pas obligatoires, mais conseillées. De plus, la balise **br** étant une balise vide (sans contenu), on préférera l'écriture simplifiée **<br />** à l'écriture **<br></br>**.

Le texte d'un paragraphe peut être aligné à l'aide de l'attribut « **align** » de la balise **<p>**. Cet attribut possède les 3 valeurs suivantes :

- ✓ **left** : aligne le texte sur la marge de gauche,
- ✓ **right** : aligne le texte sur la marge de droite,
- ✓ **center** : centre le texte,
- ✓ **justify** : justifie le texte.

Exemple :

```
<p align="justify">Ceci est un texte justifié</p>
<br />
<p align="center">Ceci est un texte centré</p>
```

Remplacés par la propriété css « text-align »

## E. LES TITRES : <h1> A <h6> (TYPE BLOCK)

Les balises « titre » permettent de mettre en évidence des parties de texte. Elles sont très utiles pour les robots de référencement. Plusieurs styles de titre sont prédéfinis, identifiés par les balises **<h1>** (le plus gros) à **<h6>** (le plus petit), ce qui donne 6 niveaux de titres différents. Le texte du titre sera encadré par les balises d'ouverture et de fermeture choisies. Un retour à la ligne sera fait après l'affichage du titre. Les balises titre possèdent aussi l'attribut align (left, right, center, justify).

Exemple et code source avec chaque balise titre :

```
<h6>Titre de niveau 6</h6>    Titre de niveau 6
<h5>Titre de niveau 5</h5>    Titre de niveau 5
<h4>Titre de niveau 4</h4>    Titre de niveau 4
<h3>Titre de niveau 3</h3>    Titre de niveau 3
<h2>Titre de niveau 2</h2>    Titre de niveau 2
<h1>Titre de niveau 1</h1>    Titre de niveau 1
```

## F. LES LIGNES HORIZONTALES : <hr> (BLOCK)

On peut insérer des lignes horizontales pour séparer certaines parties de notre document. Ceci se fait à l'aide de la balise **<hr>**. Cette balise ne nécessite pas de balise de fermeture en html mais là aussi on préférera l'écriture xhtml **<hr />**.

obsolète

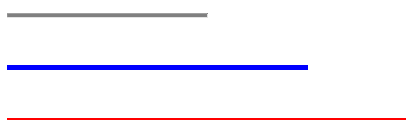
Elle possède les attributs suivants :

Attribut	Utilité
<b>align</b>	Permet d'aligner la ligne horizontale. Valeurs right, left, ou center
<b>width</b>	Spécifie la largeur de la ligne (en pixels ou % de la fenêtre)
<b>size</b>	Epaisseur de la ligne (de 1 à 10)
<b>noshade</b>	S'il est présent, l'effet d'ombre est annulé.
<b>color</b>	Spécifie la couleur de la ligne (uniquement I.E.)

Remplacés par des propriétés css : display, margin, border...

Exemple de plusieurs lignes horizontales et le code associé :

```
<hr width="100" align="left" />
<hr width="150" color="blue" align="left" />
<hr width="200" color="#ff0000" noshade="noshade" align="left" />
```



## G. MISE EN FORME DE CARACTERES

### 1. Mise en forme logique (inline)

A l'aide de ces balises, le navigateur va interpréter la sémantique du texte.

Attribut	Utilité
<b>&lt;cite&gt; ... &lt;/cite&gt;</b>	Pour effectuer une citation ou nommer un travail
<b>&lt;code&gt; ... &lt;/code&gt;</b>	Pour indiquer du code informatique
<b>&lt;em&gt; ... &lt;/em&gt;</b>	Emphase (insistance ou mise en relief) similaire à l'italique
<b>&lt;kbd&gt; ... &lt;/kbd&gt;</b>	(Keyboard, Clavier) utilisé généralement pour indiquer une réponse d'utilisateur
<b>&lt;var&gt; ... &lt;/var&gt;</b>	Indique une variable
<b>&lt;strong&gt; ... &lt;/strong&gt;</b>	Permet de faire ressortir du texte important (en gras)
<b>&lt;samp&gt; ... &lt;/samp&gt;</b>	Utilisé pour afficher la sortie d'une fonction informatique de manière littérale

Exemple :

<b>&lt;em&gt;Texte emphasé&lt;/em&gt;&lt;br&gt;</b>	<i>Texte emphasé</i>
<b>&lt;strong&gt;Texte Important&lt;/strong&gt;&lt;br&gt;</b>	<b>Texte important</b>
<b>&lt;samp&gt;Exemple de sortie d'un programme&lt;/samp&gt;&lt;br&gt;</b>	Exemple de sortie d'un programme
<b>&lt;code&gt;Un bout de code&lt;/code&gt;&lt;br&gt;</b>	Un bout de code
<b>&lt;kbd&gt;Entrée clavier&lt;/kbd&gt;&lt;br&gt;</b>	Entrée clavier
<b>&lt;var&gt;Variable&lt;/var&gt;</b>	<i>Variable</i>



## 2. Mise en forme physique (inline)

A l'aide de ces balises, c'est vous qui décidez ce que le navigateur affichera.

Attribut	Utilité
<b>&lt;b&gt; ... &lt;/b&gt;</b>	Pour mettre du texte en gras
<b>&lt;i&gt; ... &lt;/i&gt;</b>	Pour mettre du texte en italique
<b>&lt;u&gt; ... &lt;/u&gt;</b>	Souligne le texte
<b>&lt;strike&gt; ... &lt;/strike&gt;</b>	Barre le texte (en html5 => voir css)
<b>&lt;big&gt; ... &lt;/big&gt;</b>	Police plus grande (en html5 => voir css)
<b>&lt;small&gt; ... &lt;/small&gt;</b>	Police plus petite (en html5 => voir css)
<b>&lt;sup&gt; ... &lt;/sup&gt;</b>	Affiche le texte en exposant
<b>&lt;sub&gt; ... &lt;/sub&gt;</b>	Affiche le texte en indice

## 3. Les polices de caractères

Pour spécifier une police de caractères, on va utiliser la balise **<font>**. Pour que le document s'affiche avec la police voulue, il faut bien sûr que cette police soit connue par la machine de l'internaute. Pour cela, il est conseillé d'utiliser des polices standard à toutes les plateformes.

La balise **<font>** possède les attributs suivants :

Attribut	Utilité
<b>face</b>	Permet de définir par ordre de priorité, les noms de police
<b>size</b>	Permet de définir la taille de la police de 1 à 7 (valeur par défaut : 3) Il peut être utilisé de deux manières : - Absolue ex : size = 2 - Relative ex : size = +2
<b>color</b>	Spécifie la couleur du texte (avec le nom de la couleur, ou la valeur hexadécimale)

La balise **<basefont>** permet de définir une police standard. Elle a les mêmes attributs que la balise **<font>**, et a sensiblement les mêmes utilisations.

En html5 ces deux balises ont disparues, on utilisera des propriétés css.

## 4. Les caractères accentués

Si votre document utilise le codage UTF-8, les caractères accentués sont supportés. Pour les caractères spéciaux, voici une liste des principales entités :

Caractère	Entité	Caractère	Entité	Caractère	Entité	Caractère	Entité
à	&agrave;	î	&icirc;	ç	&ccedil;	ù	&ugrave;
á	&aacute;	ï	&iuml;	è	&egrave;	ú	&uacute;
â	&acirc;	ö	&eth;	é	&eacute;	û	&ucirc;
ã	&atilde;	ñ	&ntilde;	ê	&ecirc;		

obsolète

ä	&auml;	ô	&ocirc;	ë	&euml;		
æ	&aelig;	ö	&ouml;	û	&uuml;		

## 5. Les caractères spéciaux

Voici une liste de quelques caractères spéciaux. Pour que le caractère soit reconnu correctement par tous les navigateurs, il faut le remplacer dans le texte par l'entité associée.

Caractère	Code Iso	Entité
©	&#169;	&copy;
,	&#130;	
§	&#167;	&sect;
„	&#132;	
...	&#133;	
£	&#163;	&pound;
¶	&#182;	&para;
·	&#183;	&middot;
¼	&#188;	&frac14;
½	&#189;	&frac12;
¾	&#190;	&frac34;
€	&#8364;	&euro;
‘	&#145;	
’	&#146;	
“	&#147;	
”	&#148;	
•	&#149;	
–	&#150;	
—	&#151;	
Espace	&#160;	&nbsp;

## H. LES ABREVIATIONS ET ACRONYMES

La balise **<abbr>** et **<acronym>** s'utilise pour définir des abréviations ou des acronymes, au passage de la souris le texte de l'abréviation apparaît :

```
<abbr title="World Wide Web">WWW</abbr>
<abbr lang="fr" title="Abrévié;viation">abr.</abbr>
```

La balise **<acronym>** est obsolète en html5.

## IV LES LISTES

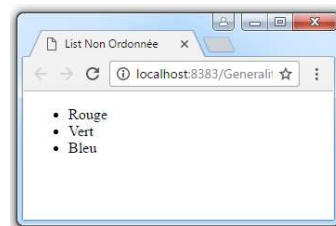
Les listes d'items sont énormément utilisées sur le web au même titre que dans les documents électroniques ou manuscrits. Ils permettent d'énumérer des items en les numérotant ou non. Les listes sont pratiques dans le cadre de sommaires.

### A. LISTE NON ORDONNEE <ul> (BLOCK)

Les listes non ordonnées s'obtiennent avec la balise **<ul>...</ul>** (unordoned list). Cette balise permet de définir une liste dans laquelle chaque élément est précédé d'une puce. Chaque élément de la liste sera repéré par la balise **<li>..</li>** (list item).

Exemple d'utilisation d'une liste :

```
<ul>
<li>Rouge</li>
<li>Vert</li>
<li>Bleu</li>
</ul>
```



La balise <ul> possède les attributs suivants :

**compact** : spécifie une liste compacte  
**type = disc** : puce en forme de disque plein (par défaut)  
**type = circle** : puce en forme de disque vide `<ul type="circle">`  
**type = square** : puce en forme de carre `<ul type="square">`

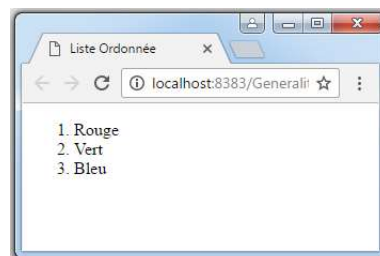
En html5 utiliser les propriétés css : list-style-type.

### B. LISTE ORDONNEE <ol> (BLOCK)

Les listes ordonnées s'obtiennent avec la balise **<ol>...</ol>** (ordoned list). Cette balise permet de définir la liste numérotée, avec une incrémentation automatique. Chaque élément de la liste sera repéré par la balise **<li>...</li>**. Le titre de la liste sera lui repéré par la balise **<lh>...</lh>**.

Exemple d'utilisation d'une liste :

```
<ol>
<li>Rouge</li>
<li>Vert</li>
<li>Bleu</li>
</ol>
```



La balise <ol> possède les attributs suivants :

**type = A** : les items sont 'numérotés' par des lettres  
**type = a** : les items sont 'numérotés' par des lettres minuscules  
**type = I** : les items sont 'numérotés' en chiffres romains (majuscules)  
**type = i** : les items sont 'numérotés' en chiffres romains (minuscules)  
**type = 1** : les items sont 'numérotés' par des nombres (valeur par défaut)  
**start = n** : Valeur de départ de la numérotation `<ol start=4>`

majuscules

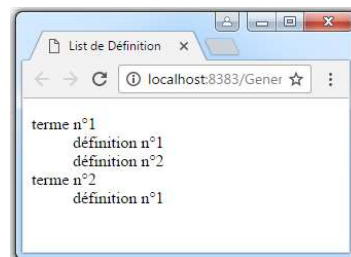
En html5 utiliser les propriétés css : list-style.

### C. LISTE AVEC DEFINITION <DL> (BLOCK)

Une liste de termes et de définitions correspondantes s'obtient avec la balise **<dl>...</dl>** (definition list), qui est accompagnée des balises **<dt>...</dt>** (definition terme) et **<dd>...</dd>** (definition data). <dt> représente donc le terme et <dd> le terme. A noter que la balise <dl> n'a que l'attribut compact.

Exemple d'une liste de définition avec 2 termes et 3 définitions :

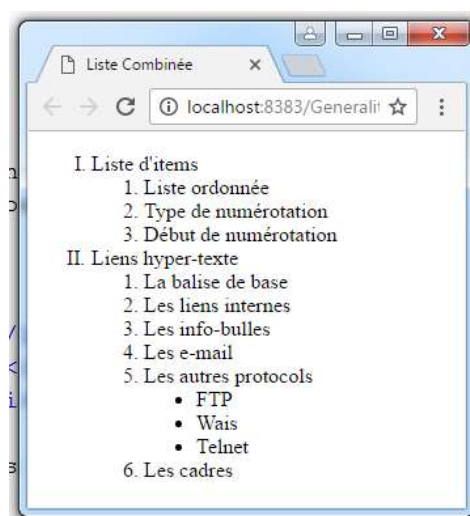
```
<dl>
  <dt>terme n°1</dt>
    <dd>définition n°1</dd>
    <dd>définition n°2</dd>
  <dt>terme n°2</dt>
    <dd>définition n°1</dd>
</dl>
```



### D. COMBINER LES TYPES DE LISTES

Il est possible d'imbriquer les listes entre elles afin d'avoir des listes de listes... pour créer, par exemple des menus détaillés.

```
<ol type="I">
  <li>Liste d'items
    <ol type="1">
      <li>Liste ordonnée</li>
      <li>Type de numérotation</li>
      <li>Début de numérotation</li>
    </ol>
  </li>
  <li>Liens hyper-texte
    <ol type="1">
      <li>La balise de base </li>
      <li>Les liens internes </li>
      <li>Les info-bulles </li>
      <li>Les e-mail </li>
      <li>Les autres protocols
        <ul type="disc">
          <li>FTP</li>
          <li>Wais</li>
          <li>Telnet</li>
        </ul>
      </li>
      <li>Les cadres </li>
    </ol>
  </li>
</ol>
```



### E. CREER VOS PROPRES PUCES

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

- soit vous passez par l'attribut `src="puce.png"` de la balise `<li>`, si le fichier graphique de votre puce se nomme `puce.png`. Attention, cet attribut est spécifique à Internet Explorer

- soit vous insérez la balise <img> dans la balise <li>, en définissant votre image comme un simple fichier. Cette solution est la meilleure, car valable pour tous les navigateurs. Supprimez les puces par défaut.

Exemple d'une liste avec un fichier png :

```
<ul type="none">
  <li>
    element 1
  </li>
  <li>
    element 2
  </li>
</ul>
```

Il est préférable de modifier les images des puces d'une liste par l'attribut « list-style-image » du css :

```
list-style-image: url("puce.png");
```

## V LES IMAGES <IMG> (TYPE INLINE)

Pas de site web sans l'affichage d'images. Pour cela l'emploi de la balise **<img>** est nécessaire. Elle comporte de nombreux attributs que l'on va détailler même si beaucoup d'entre eux ont leur équivalence en CSS. La syntaxe générale en est la suivante :

```

```

### A. LA SOURCE DE L'IMAGE

L'attribut le plus important est « **src** », qui permet de spécifier l'url (chemin) complète du fichier. C'est le seul obligatoire, tous les autres sont optionnels.

```

```

Nom de fichier contient le nom de l'image et son extension :

```
ex : 
```

Si votre fichier se trouve dans un autre répertoire que vos pages HTML, on doit donner le chemin :

```
ex : 
```

On peut aussi charger une image sur un autre site. Il faut pour cela donner l'adresse complète du serveur :

```
ex : 
```

Attention : pas de barre oblique inverse « \ ».

### B. PRÉCISER LA TAILLE

On peut préciser la taille de l'image au navigateur par l'intermédiaire des attributs « width » et « height ». De ce fait, le navigateur chargera plus rapidement votre page. De plus si l'internaute a désactivé le chargement des images, la mise en page ne sera pas perturbée. Ces dimensions correspondent à la taille en pixel de l'image.

```
ex : 
```

Vous pouvez agrandir ou rétrécir votre image en jouant sur les attributs width et height, mais attention, l'image ne sera pas chargée plus rapidement (le poids de l'image reste le même, seules les tailles d'affichage sont modifiées). Il est donc judicieux de scanner de suite les images avec la bonne taille et la bonne résolution.

Ces propriétés sont à mettre dans la css en html5.

## C. RENSEIGNER L'IMAGE

Il existe deux solutions permettant de renseigner une image :

- l'attribut « alt » (obligatoire en html5)

Il est utile dans deux cas précis : si l'image ne s'affiche pas, l'internaute pourra quand même deviner le sens de l'image, et de plus, on peut par ce biais fournir une légende à une image, qui s'affiche lorsque la souris passe dessus l'image.

En html5 cet attribut est même rendu obligatoire (accessibilité des malvoyants).

ex : ``

- l'attribut « longdesc »

Il permet de spécifier l'url d'une page html comportant une description détaillée de l'image

ex : ``

## D. METTRE UN BORD

obsolète

Avec l'attribut « **border** », vous pouvez entourer l'image d'une bordure. L'attribut border prend une valeur de 1 à x qui exprime la largeur de la bordure entourant l'image.

ex : ``

La css offre des possibilités beaucoup plus précises (différencier les bords, ...)

## E. POSITIONNER LES IMAGES

### 1. Espacer les images

obsolète

Avec les attributs « hspace=n » et « vspace=m » (n et m étant des nombres de pixels), vous pouvez fixer un espace vertical et horizontal autour de l'image.

``

### 2. Aligner les images

Par défaut, l'image est placée sur une ligne et le texte, s'il est présent est aligné sur le bas de l'image. Si vous voulez encadrer une image avec du texte (style journal), l'attribut « align » est requis. Il a 9 valeurs possibles :

Valeur	Effet
left	Image à gauche
right	Image à droite

<b>middle</b>	Image au milieu
<b>absbottom</b>	Image en bas (absolu)
<b>absmiddle</b>	Image au milieu (absolu)
<b>baseline</b>	Sur la ligne du texte
<b>bottom</b>	Image en bas
<b>texttop</b>	Image en haut
<b>top</b>	En haut

```

```

Là aussi cet attribut est remplacé en html5 par son équivalent css.

## F. PREVISUALISATION

obsolète

Avec l'attribut « lowsrc », vous pouvez accélérer l'affichage de vos pages. En effet, le navigateur affichera en premier lieu ce qu'il trouve dans l'attribut lowsrc. Généralement, l'image contenue dans l'attribut lowsrc doit être en basse résolution ou compressée. L'utilisation de cet attribut est surtout intéressante pour des images supérieures à 20 Ko. Précisez également les dimensions de l'image. Attention, cette balise est spécifique à Netscape !



## VI LES LIENS

### A. INTRODUCTION

Un lien hypertexte est un élément graphique ou textuel d'une page web qui sous l'action d'un clic de souris commande le chargement d'une autre page dans le navigateur. Cette autre page peut faire partie du même site web ou faire partie d'un autre site.

Dans une page web, un lien textuel est souligné et le curseur de la souris devient une petite main. Si c'est un graphique seule le curseur change.

### B. LA BALISE DE BASE <A> (INLINE)

La balise <a> permet d'établir un lien hypertexte. Elle possède les attributs suivants :

Attribut	Utilité
<b>href</b>	Permet d'indiquer l'URL (ou l'endroit de la page courante) où l'on veut se rendre
<b>download</b>	Permet un dialogue avec l'utilisateur pour enregistrer la cible du lien fourni
<b>name</b>	Permet de définir un point d'encrage pour les liens internes (remplacé en html5 par id)
<b>title</b>	Permet de donner une description du lien (sur IE uniquement)
<b>style</b>	Permet de spécifier une feuille de style
<b>target</b>	Spécifie où doit s'ouvrir le document chargé

#### 1. Créer un lien à l'intérieur d'une même page

Les liens internes permettent de naviguer au sein d'une page HTML, sans utiliser les ascenseurs. Depuis un endroit du document (c'est la source du lien), où l'on va cliquer vers la cible définie (c'est l'ancre) où l'on va se débrancher.

Il faut donc définir à la fois la source du lien, et la cible du lien (ancre).

La cible se définit par : <a name="cible" ></a> en html5 <a id="cible" ></a>

La source se définit par : <a href="#cible" >Cliquez ici pour aller vers la cible</a>

**Exemple :**

*En haut de la page html mettre : <a name="hautpage"></a>*

*En bas de la page écrire : <a href="#hautpage">retour en haut de la page</a>*

#### 2. Créer un lien externes vers un autre site web

La syntaxe générale est la suivante :

```
<a href="adresse_destination"
  title="Description sommaire du lien : info bulle"
  style="feuille de style associée"
  target="destination">Texte_ou_image_à_cliquer </a>
```

Exemples :

```
<a href="http://www.afpa.fr" >Découvrez l'AFPA</a>
<a href="http://www.afpa.fr" ></a>
```

### C. LES INFOS-BULLES

L'attribut « title » permet de faire afficher un court descriptif du lien lorsque le curseur de la souris passe sur le lien.

```
<a href=http://www.afpa.fr
title="vers le site AFPA">
</a>
```



### D. LES E-MAILS

- lien email sans sujet :  
**<a href="mailto:didier.bonneau@afpa.fr">Ecrivez-moi</a>**
- lien email avec sujet :  
**<a href="mailto:didier.bonneau@afpa.fr ?subject=absence">Ecrivez-moi</a>**
- lien email avec sujet et personne en copie:  
**<a href="mailto:didier.bonneau@afpa.fr ?cc=michel.coulard@afpa.fr&subject=absence">Ecrivez-moi</a>**
- lien email avec sujet et texte associé:  
**<a href="mailto:didier.bonneau@afpa.fr ?subject=absence&body=en cas d'absence">Ecrivez-moi</a>**

### E. LES AUTRES PROTOCOLES

On vient de voir qu'il est possible d'appeler des pages web, des images mais aussi d'autres documents (.txt, .shtml, .doc) , des films (.avi, .jpeg), des sons (.wav, .mid, .rmi, .ra, aif, .mp3, etc...) et qu'il était possible d'utiliser un autre protocole : le courrier électronique. Il existe bien d'autres protocoles accessibles par un simple lien hypertexte.

Syntaxe	Description
<b>&lt;a href=""&gt;</b>	Protocol par défaut : HTTP
<b>&lt;a href="#"&gt;</b>	Lien interne, protocol HTTP
<b>&lt;a href="http://"&gt;</b>	Protocol HTTP
<b>&lt;a href="pnm://"&gt;</b>	Protocol PNM pour le streaming en RealAudio
<b>&lt;a href="ftp://"&gt;</b>	Protocol FTP pour le transfert de fichiers
<b>&lt;a href="mailto:"&gt;</b>	Protocol SMTP pour le courrier électronique
<b>&lt;a href="news:"&gt;</b>	Protocol NNTP pour les forums de discussion
<b>&lt;a href="telnet://"&gt;</b>	Protocol Telnet pour le contrôle d'un ordinateur distant
<b>&lt;a href="gopher://"&gt;</b>	Protocol Gopher pour discussion et transfert de fichier
<b>&lt;a href="wais://"&gt;</b>	Protocol Wais
<b>&lt;a href="file://"&gt;</b>	Adressage local sur un poste non-distant

### F. PRECISION SUR L'ATTRIBUT « TARGET »

Pour préciser la destination, il existe des constantes pour l'attribut « target » :

Syntaxe	Description
<b>target="_blank"</b>	Affichage du lien dans une autre fenêtre du navigateur
<b>target="_top"</b>	Affichage dans la totalité de la fenêtre courante
<b>target="_parent"</b>	Affichage dans le cadre qui enveloppe la structure courante
<b>target="_self"</b>	Affichage dans le cadre courant (par défaut)
<b>target="nom_du_cadre"</b>	Affichage dans le cadre spécifié (non supporté en html5)

## VII LES ANIMATIONS VIDEOS, ZONES REACTIVES

Ces animations sont maintenant obsolètes.

### A. ANIMATIONS VIDEOS

Le langage HTML est un langage permettant l'affichage de document et la navigation au sein de ces documents. Il permet d'afficher des pages 'statiques'. Il est néanmoins possible de mettre un peu d'animation dans ces pages statiques. Des animations plus élaborées pourront être obtenues par l'intégration d'applets Javascript, ou par l'utilisation d'outils d'animation (de type Flash).

#### 1. Le défilement de texte

La balise **<marquee>** permet de faire défiler du texte (ou des images). Cette balise n'est utilisable que sous Internet Explorer. Elle comporte un nombre important d'attributs :

Attribut	Utilité
<b>align</b>	Permet d'indiquer l'alignement du texte dans l'espace Marquee Vaut top, middle, ou bottom
<b>behavior</b>	Indique le type de défilement. Valeurs scroll (défilement en boucle), alternate (dans un sens puis dans l'autre), slide (défile une seule fois)
<b>direction</b>	Direction du défilement. Défaut left (de gauche à droite), sinon right (de droite à gauche)
<b>bgcolor</b>	Couleur de fond
<b>height</b>	Hauteur du MARQUEE, en pixel (height=10) ou en pourcentage (height=10%)
<b>width</b>	Largeur du MARQUEE, en pixel (width=10) ou en pourcentage (width=10%)
<b>hspace</b>	Marges droites et gauches, en pixel
<b>vspace</b>	Marges basses et hautes, en pixel
<b>loop</b>	Nombre de défilements. Si -1 ou Infinite, ne s'arrête jamais
<b>scrollamount</b>	Nombre de pixel entre deux affichages successifs
<b>scrolldelay</b>	Délai en millisecondes, entre deux affichages successifs

*<marquee>Texte qui défile de gauche à droite doucement</marquee>*  
*<marquee scrolldelay=5 scrollamount=50>Texte très rapide</marquee>*

#### 2. Intégration d'une vidéo

Il est possible d'utiliser la balise **<img>** pour afficher autre chose que des images fixe. On peut notamment afficher des .gif animés, et aussi des vidéos .avi (ceci est vrai sur Internet Explorer).

Pour ce faire, on utilise l'attribut **dynsrc** de la balise **<img>**.

obsolète

`` permettra de jouer le film `film1` si l'on est sur un navigateur I.E., et d'afficher l'image `image.gif` si l'on est sur un autre navigateur. Les différents attributs sont les suivants :

Attribut	Utilité
<b>dynsrc</b>	Donne le chemin du clip vidéo à jouer.
<b>start</b>	Vaut fileopen ou mouseover ; Indique quand a lieu le démarrage de la vidéo.
<b>loop</b>	Nombre de fois que le clip est joué. Si -1 ou infinie, ne s'arrête jamais
<b>controls</b>	Indique qu'un panneau de contrôle est apparent et permet de contrôler la vidéo (pauses, etc...)
<b>loopdelay</b>	Délai en millisecondes, entre deux projections du clip

En htm5 c'est la balise `<video>` qui va permettre de jouer des vidéos de type quelconques.

```
<video width="320" height="240" controls autoplay>
  <source src="movie.mp4" type="video/mp4">
  <source src="movie.ogg" type="video/ogg">
  Votre navigateur ne supporte pas la balise video.
</video>
```

L'attribut « **controls** » ajoute des contrôles comme play, pause, et volume.

L'attribut « **autoplay** » permet de jouer directement la vidéo.

Il est possible de préciser plusieurs sources pour la vidéo grâce à la balise `<source>`

Seuls les formats vidéo MP4<sup>10</sup>, WebM<sup>11</sup>, et Ogg<sup>12</sup> sont supportés par le standard HTML5.

### 3. Zones réactives

Une image peut être définie comme étant une image réactive du côté client, c'est-à-dire une image qui peut être découpée en zones, chaque zone pouvant réagir lors du click souris et pointant vers un lien externe ou interne.

L'image `<img>` est définie comme étant une image réactive à l'aide de l'attribut « `usemap` » :

```

```

La MAP est définie avec la balise `<map>` incluant une ou plusieurs balises `<area>` :

```
<map name=" nom attribué à la carte ">
  <area name="nom_de_la_zone" shape="type_de_zone" coords="coordonnées"
  href="adr_destination" alt="commentaire">
```

Ex : `<map name="maCarte">`  
`<area shape="rect" coords="60,10,99,49" href="page2.htm">`  
`</map>`

<sup>10</sup> MP4 : Développée par le Moving Pictures Expert Group

<sup>11</sup> WebM : Développée par Mozilla, Opera, Adobe, et Google

<sup>12</sup> Ogg : Theora Ogg. Développée par la Foundation Xiph.Org

Le nom de la MAP est celui qui a été défini dans l'image pour l'attribut usemap. La MAP est découpée en un certain nombre de zones par la balise **<area>**. Chaque « area » possède les attributs suivants :

Attribut	Utilité
<b>shape</b>	Définit la forme de la zone. Les valeurs possibles sont : <b>rect</b> : rectangle <b>circle</b> : cercle <b>poly</b> : polygone
<b>coords</b>	Indique les coordonnées des coins supérieur gauche et inférieur droit pour un rectangle, les coordonnées du centre et le rayon pour un cercle et les coordonnées de chaque coin pour un polygone.
<b>href</b>	Définit la page qui s'affiche lors du click sur la zone
<b>alt</b>	Commentaire sur la zone

Exemples :

#### ZONE CIRCULAIRE

```
<map name="bulles">
  <area shape="circle" coords="50,50,40" href="#bulles">
</map>

```



#### ZONE RECTANGULAIRE

```
<map name="menu">
  <area shape="rect" coords="10,3,60,16" href="#home">
  < area shape="rect" coords="10,16,60,30" href="#edito">
  < area shape="rect" coords="10,30,60,43" href="#links">
  < area shape="rect" coords="10,43,60,57" href="#mail">
</map>

<area shape="poly" coords="0,0,34,0,78,20,43,33,46,69,19,54,0,78
,0,0" href="#moi">
< area shape ="poly"
coords="34,0,78,20,43,33,46,69,19,54,0,78,0,93,91,93,123,76,98,49,147,34,131,0,34,0"
href="#toi">
< area shape ="poly" coords="131,0,200,0,200,94,91,93,123,76,98,49,147,34,131,0"
href="#nous">
</map>

```



## VIII LES TABLEAUX

Les tableaux sont très largement utilisés dans le développement de pages web. Ils sont indispensables pour pouvoir présenter l'information de manière structurée et facilement lisible. Un tableau peut contenir du texte, des images, des vidéos, des formulaires, d'autres tableaux...

### A. CREATION D'UN TABLEAU

#### 1. Les balises de base

On crée un tableau à l'aide de la balise `<table> ... </table>`.  
Un tableau est composé de plusieurs lignes repérées par les balises `<tr>...</tr>`.  
Chaque ligne est composée de plusieurs cellules repérées par les balises `<td>...</td>`.

#### 2. Légende du tableau

Le titre du tableau est repéré par les balises `<caption>...</caption>`. Le titre peut être placé en dessus du tableau (attribut « top », valeur par défaut), ou bien en dessous du tableau (attribut « bottom »).

#### 3. Légende des colonnes

L'en tête des colonnes (c'est-à-dire chaque cellule de la première ligne) est repérée par les balises `<th>...</th>`.

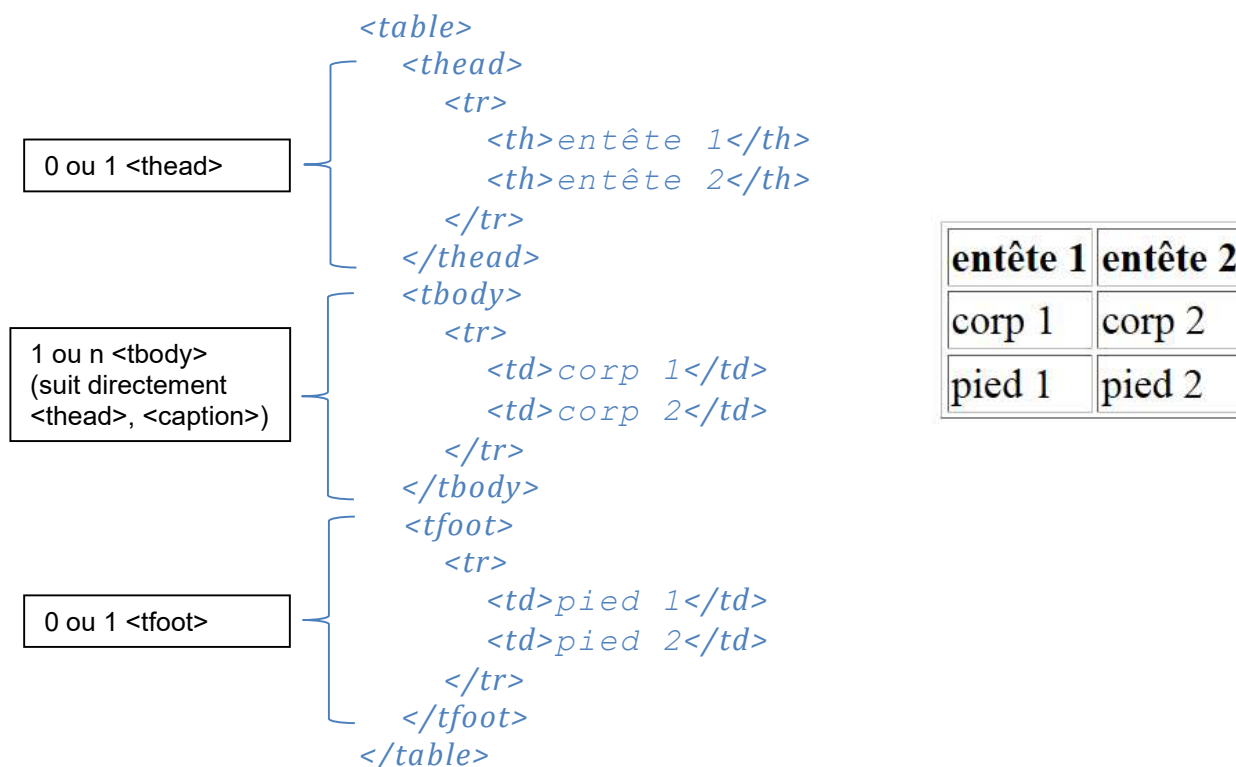
Exemple d'un tableau avec le code associé :

Liste des clients		
Nom	Prénom	Adresse
DUPOND	Léon	3 rue de la paix, 38000 Grenoble
DUBOIS	Claire	3 place de la poste, 38000 Grenoble
DUBUIS	Julie	25 cours Berriat, 38000 Grenoble

```
<html>
<head>
<title>Exemple de tableau</title>
</head>
<body>
<table border="2">
<caption>Liste des clients</caption>
<tr>
<th>Nom</th>
<th>Prénom</th>
<th>Adresse</th>
</tr>
<tr>
<td>DUPOND</td>
<td>Léon</td>
<td>3 rue de la paix, 38000 Grenoble</td>
</tr>
<tr>
<td>DUBOIS</td>
<td>Claire</td>
<td>3 place de la poste, 38000 Grenoble</td>
</tr>
<tr>
<td>DUBUIS</td>
<td>Julie</td>
<td>25 cours Berriat, 38000 Grenoble</td>
</tr>
</table>
</body>
</html>
```

#### 4. Nouvelles balises sémantiques

De nouvelles balises sont apparues apportant de la sémantique aux tableaux.  
Ce sont les balises `<thead>`, `<tbody>` et `<tfoot>`.



Avantages : lors d'un grand tableau, le scroll joue uniquement sur le `<tbody>`. Pour une impression, le `<thead>` et le `<tfoot>` sont imprimés à chaque page.

## B. TAILLE ET ALIGNEMENT

Attention tous les attributs qui suivent ont en html5 leur équivalents css et ne doivent donc plus être utilisés.

### 1. Taille du tableau et des cellules

On peut fixer la taille du tableau ou des cellules. Cette taille est exprimée au choix en pixel ou en pourcentage de l'espace libre dans la fenêtre. Vous pouvez donc spécifier de manière précise la largeur : `width=x` et la hauteur : `height=y` de votre tableau en sachant pertinemment que le navigateur calcule la taille des cellules en conséquence. Il vous est tout à fait possible de ne spécifier qu'une seule de ces dimensions ou même aucune (dans ce cas le navigateur calcul la taille par la technique du juste minimum).

Ces deux attributs sont valables pour les balises `<table>`, `<th>`, et `<td>`.

### 2. Alignement dans les cellules

On peut aligner les données à l'intérieur des cellules grâce aux attributs « `align` » pour l'alignement horizontal et « `valign` » pour la verticale.

Attribut	Utilité
<b>align=left</b>	Valeur par défaut. Les données sont alignées à gauche dans la cellule pour la balise TD, ou dans toutes les cellules de la ligne pour la balise TR
<b>align=right</b>	Les données sont alignées à droite dans la cellule pour la balise TD, ou dans toutes les cellules de la ligne pour la balise TR
<b>align=center</b>	Les données sont alignées au centre dans la cellule pour la balise TD, ou dans toutes les cellules de la ligne pour la balise TR



<b>valign=top</b>	Les données sont alignées en haut dans la cellule pour la balise TD, ou dans toutes les cellules de la ligne pour la balise TR
<b>valign=bottom</b>	Les données sont alignées en bas dans la cellule pour la balise TD, ou dans toutes les cellules de la ligne pour la balise TR
<b>valign=middle</b>	Les données sont alignées verticalement au centre dans la cellule pour la balise TD, ou dans toutes les cellules de la ligne pour la balise TR
<b>valign=baseline</b>	Valable que pour la balise TR. Les données des cellules sont toutes alignées.

Exemple d'alignement dans un tableau :

Liste des clients		
Nom du client	Prénom du client	Adresse du client
DUPOND	Léon	3 rue de la paix, 38000 Grenoble
DUBOIS	Claire	3 bis place de la poste, 38000 Grenoble
DUBUIS	Julie	25 cours Berriat, 38000 Grenoble

```
<table border="1">
<caption>Liste des clients</caption>
<tr>
<th>Nom du client</th>
<th>Prénom du client</th>
<th>Adresse du client</th>
</tr>
<tr align="center">
<td>DUPOND</td>
<td>Léon</td>
<td>3 rue de la paix, 38000 Grenoble</td>
</tr>
<tr>
<td align="right">DUBOIS</td>
<td>Claire</td>
<td>3 bis place de la poste, 38000
Grenoble</td>
</tr>
<tr>
<td valign="top">DUBUIS</td>
<td valign="bottom">Julie</td>
<td align="center">25 cours Berriat, 38000
Grenoble</td>
</tr>
</table>
```

## C. BORDURES, ESPACEMENTS ET MARGES

Attention là aussi tous les attributs qui suivent ont en html5 leur équivalents css et ne doivent donc plus être utilisés.

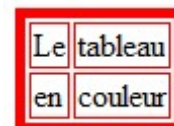
### 1. Bordures du tableau

Dans les exemples précédents nous avons déjà rencontré l'attribut « **border** » de la balise <table>. La valeur par défaut est de 1 mais il est possible de ne plus avoir de bordure en lui attribuant la valeur 0 ou au contraire une bordure plus grosse d'une épaisseur exprimée en pixels 2, 3, ...

Windows attribue automatiquement une couleur de bordure mais il est aussi possible de la modifier en utilisant l'attribut « bordercolor ».

Exemple :

```
<table border="5" bordercolor="red">
<tr><td>Le</td><td>tableau</td></tr>
<tr><td>en</td><td>couleur</td></tr>
</table>
```



Il y a encore bien d'autres possibilités pour jouer sur les bordures externes d'un tableau par l'attribut « frame » ou sur les bordures internes des cellules par l'attribut « rules ».

## 2. Espacement entre les cellules

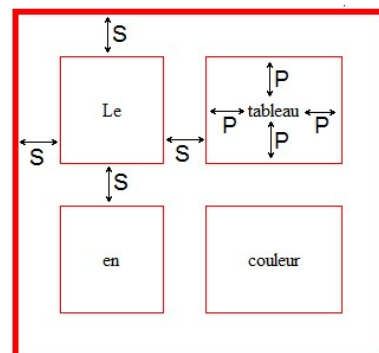
Par souci de clarté, on peut vouloir espacer les cellules grâce à l'attribut « **cellspacing=s** » dont la valeur n définit l'espacement en pixels entre chaque cellule d'un tableau. Par défaut, s=2.

## 3. Marges internes aux cellules

On peut également spécifier des marges à l'intérieur des cellules afin d'espacer les données du bord des cellules grâce à l'attribut « **cellpadding=p** » dont la valeur p est exprimé en pixels. Par défaut, p=1.

Exemple :

```
<table border="5" bordercolor="#ff0000"
      cellpadding="35"
      cellspacing="35">
  <tr><td>Le</td><td>tableau</td></tr>
  <tr><td>en</td><td>couleur</td></tr>
</table>
```



## 4. Récapitulatif de mise en page

Différents attributs des balises <table>, <td>, <th> sont disponibles qui permettent d'influer sur la présentation des tableaux :

Attribut	Utilité
<b>width</b>	Indique la largeur du tableau ou d'une cellule (en pixel ou en %)
<b>height</b>	Indique la hauteur du tableau ou d'une cellule (en pixel ou en %)
<b>border</b>	Taille de la bordure du tableau (1 par défaut, 0 pas de bordures)
<b>cellpadding</b>	Indique l'espace entre la bordure et le contenu de chaque cellule
<b>cellspacing</b>	Indique l'espace entre les cellules du tableau
<b>bgcolor</b>	Spécifie la couleur du tableau ou d'une cellule
<b>bordercolor</b>	Spécifie la couleur de la bordure du tableau
<b>bordercolorlight</b>	Spécifie la couleur claire de la bordure (3D)
<b>bordercolordark</b>	Spécifie la couleur sombre de la bordure (3D)
<b>background</b>	Url de l'image à afficher en fond (IE uniquement)
<b>frame</b>	Permet d'afficher des bordures sur un ou plusieurs cotés du tableau. Il faut que l'attribut FRAME soit défini dans la balise <TABLE>. Les valeurs possibles sont : <b>border</b> : Affiche toutes les bordures <b>above</b> : Affiche les bordures hautes seulement. <b>below</b> : Affiche les bordures basses seulement <b>lhs</b> : Affiche les bordures gauches seulement. <b>rhs</b> : Affiche les bordures droites seulement. <b>box</b> : Affiche une boîte autour du tableau <b>insides</b> : Affiche les bordures internes hautes et basses <b>hsides</b> : Affiche les bordures externes hautes et basses <b>void</b> : Retire les bordures externes. <b>vsides</b> : Affiche les bordures externes gauches et droites.
<b>rules</b>	Permet d'afficher des bordures à l'intérieur du tableau. Il faut que l'attribut RULES soit défini dans la balise <TABLE>. Les valeurs possibles sont : <b>none</b> : Retire toutes les bordures internes <b>rows</b> : Affiche une bordure horizontale entre chaque ligne <b>cols</b> : Affiche une bordure verticale entre chaque colonne <b>groups</b> : Affiche une bordure horizontale entre chaque section <b>all</b> : Affiche toutes les bordures internes du tableau

Les attributs "align", "bgcolor", "cellpadding", "cellspacing", "frame", "rules", "summary", and "width" ne sont plus supportés en HTML5. Utiliser leur équivalents CSS.

## D. FUSIONNER LIGNES ET COLONNES

On peut regrouper plusieurs lignes ou plusieurs colonnes afin d'obtenir des cellules plus grandes. Ceci s'effectue à l'aide des attributs « **rowspan** » et « **colspan** ».

Exemple de fusions dans un tableau :

Liste des clients		
Nom du client	Prénom du client	Adresse du client
DUPOND	Léon / 3 rue de la paix, 38000 Grenoble	
DUBOIS	Claire	même adresse : 25 cours Berriat, 38000 Grenoble
DUBUIS	Julie	

```
<table border="1">
<caption>Liste des clients</caption>
<tr>
    <th>Nom du client</th>
    <th>Prénom du client</th>
    <th>Adresse du client</th>
</tr>
<tr>
    <td>DUPOND</td>
    <td colspan="2">Léon / 3 rue de la paix,
38000 Grenoble</td>
</tr>
<tr>
    <td>DUBOIS</td>
    <td>Claire</td>
    <td rowspan="2">même adresse : 25
cours Berriat, 38000 Grenoble</td>
</tr>
<tr>
    <td>DUBUIS</td>
    <td>Julie</td>
</tr>
</table>
```

## E. COULEUR ET IMAGE DE FOND

Pour html5 voir leurs équivalents css. Ne plus utiliser ces attributs.

Il est permis de mettre une couleur de fond à un tableau ou à une cellule grâce à l'attribut « bgcolor ». Valable pour les balises <table>, <th>, <tr> et <td>.

Il est également permis de mettre une image de fond à un tableau (totalité des cellules) ou à une seule cellule (ou plusieurs) grâce à l'attribut « background » dont la valeur est l'URL du fichier image.

## IX LES CADRES

Les cadres ne sont plus supportés en html5. Seule la balise <iframe> reste possible.

### A. INTRODUCTION

Un site bien fait est un site où il est facile de naviguer entre ses différentes pages. Pour cela il est nécessaire de fournir à l'utilisateur un menu qui reste permanent quelque soit la page du site où l'on se trouve.

La technique la plus utilisée c'est « les cadres ».

Un cadre est une zone de la fenêtre du navigateur associée à une page HTML. On utilise au moins deux zones : une pour le menu de navigation, l'autre au contenu de la page relatif au menu.

### B. LES BALISES DE BASE

Une page utilisant la technique des cadres sert exclusivement à définir les cadres, elle ne peut être utilisée pour afficher du texte, des images. Par contre les pages secondaires appelées depuis cette page principales et qui se partageront la fenêtre du navigateur seront "normales", c'est-à-dire que vous pourrez en faire ce que vous voudrez, y mettre des tableaux, des images, du texte...

La page principale est transparente à l'internaute qui visite votre site puisqu'il verra les pages associées aux cadres. Par contre c'est préférentiellement cette page principale qui devra contenir les méta-informations (voir chapitre <meta>) correspondantes aux pages appelées dans les cadres car il est préférable qu'un moteur de recherche renvoie l'internaute vers la page principale qui propose le sommaire. N'oubliez pas de donner un titre à votre page principale car les balises <title> de vos pages secondaires seront ignorées.

#### 1. Architecture des cadres

L'architecture de base d'une page principale est similaire à celle des autres pages web à cela que la balise <frameset> remplace <body>.

Exemple de page principale :

```
<html>
  <head>
    les méta-informations
    <title>Votre titre</title>
  </head>
  <frameset>
    le contenu de la page principale
  </frameset>
</html>
```

La balise <frameset> définit la liste des cadres parallèles sur l'horizontale ou la verticale. L'attribut « **cols** » a pour valeurs la liste des dimensions respectives en pourcentage ou en pixels des cadres (séparées par des virgules), la séparation des cadres se fera sur l'horizontale, c'est-à-dire qu'une ligne séparatrice verticale (ou plusieurs) coupera la fenêtre

entre chaque cadre. L'attribut « **rows** » a pour valeurs a liste des dimensions respectives en pourcentage ou en pixels des cadres (séparées par des virgules), la séparation des cadres se fera sur la verticale, c'est-à-dire que la ligne séparatrice sera horizontale.

**Syntaxes :**

*<frameset cols="20%, 80%"> pour deux cadres séparées verticalement  
 <frameset cols="20%, 20%, 60%"> pour trois cadres séparées verticalement  
 <frameset rows="200, 600"> pour deux cadres séparées horizontalement  
 <frameset rows="30%, \*"> l'astérisque signifie que la dimension correspondante est automatiquement celle qui reste disponible dans la fenêtre du navigateur*

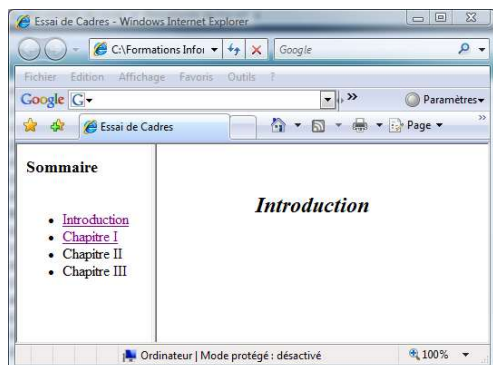
## 2. Définition des cadres

La balise **<frame>** définit chaque cadre individuellement. Chaque cadre doit être associé à une page web grâce à l'attribut « **src** » dont la valeur est l'URL absolue ou relative de la page web associée.

Un nom est donné à chaque cadre par l'intermédiaire de l'attribut « **name** ».

**Exemples :**

```
<html>
<head>
  <title>Essai de Cadres</title>
</head>
<frameset cols="150, *">
  <frame name="sommaire" src="CadreSommaire.html">
  <frame name="contenu" src="CadreIntroduction.html">
</frameset>
</html>
```



```
<!-- CadreSommaire.html -->
<html>
<body>
  <h3>Sommaire</h3>
  <br />
  <ul>
    <li><a href="CadreIntroduction.html"
      target="contenu">Introduction</a></li>
    <li><a href="CadreChapitre1.html"
      target="contenu">Chapitre I</a></li>
    <li>Chapitre II</li>
    <li>Chapitre III</li>
  </ul>
</body>
</html>

<!-- CadreIntroduction.html -->
<html>
<body>
  <br />
  <br />
  <h2 align="center"><i>Introduction</i></h2>
</body>
</html>

<!-- CadreChapitre1.html -->
<html>
<body>
  <br />
  <br />
  <h2 align="center"><i>Chapitre I</i></h2>
</body>
</html>
```

Lorsque l'on utilise un lien `<a href=...>` à l'intérieur d'une frame, la page html appelée va s'afficher à l'intérieur de la frame appelante. On peut néanmoins faire afficher cette page html à l'intérieur d'une autre frame en utilisant l'attribut « **target** ». Il suffit de donner cet l'attribut le nom de la frame dans laquelle on veut faire afficher la page.

Certaines valeurs particulières existent :

`target="_blank"` : dans ce cas le navigateur va ouvrir une nouvelle fenêtre pour afficher la page,  
`target="_top"` : dans ce cas les frames disparaissent et la page s'affiche dans tout l'écran.

## C. IMBRICATION DE CADRES

Il est tout à fait possible d'imbriquer les cadres les uns dans les autres pour obtenir les configurations suivantes (ou toutes autres) :

cadre1
cadre2
cadre3

```
<frameset rows="50%,25%,25%">
  <frame name="cadre1" src="cadre1.html">
  <frame name="cadre2" src="../info/ cadre2.html">
  <frame name="cadre3" src="http://afpa.fr/cadre3.html">
</frameset>
```

Sommaire	cadre1
	cadre2

```
<frameset cols="180,*">
  <frame name="sommaire" src="sommaire.html">
  <frameset rows="50%,50%">
    <frame name="cadre1" src="cadre1.html">
    <frame name="cadre2" src="cadre2.html">
  </frameset>
</frameset>
```

Menu primaire	
Sommaire	Article

```
<frameset rows="15%,*">
  <frame name="menu" src="menu.html">
  <frameset cols="180,*">
    <frame name="sommaire" src="sommaire.html">
    <frame name="article" src="article.html">
  </frameset>
</frameset>
```

## D. MISE EN PAGE

### 1. Bordures

On peut ou non donner une bordure à l'ensemble des cadres grâce à l'attribut « **frameborder** » (valeur **yes** ou **no**) de la balise `<frameset>`. Par défaut, `frameborder="yes"`. Pour donner une taille spécifique à la bordure, utilisez l'attribut « **border** » dont la valeur est la largeur en pixel de la bordure. Par défaut, `border="5"`.

On peut aussi choisir la couleur de la bordure de l'ensemble des cadres à l'aide de l'attribut « **bordercolor** » dont la valeur est soit le nom normalisé (HTML) de la couleur soit le code hexadécimal correspondant. Par défaut, cette couleur est celle de la palette Windows courante.

### 2. Marges internes

On peut aussi mettre des marges sur les côtés gauche et droit du cadre avec l'attribut « **marginwidth** » de la balise `<frame>` et dont la valeur est le nombre de pixel. On peut faire la même chose pour les marges du haut et du bas avec « **marginheight** ».

## E. RECAPITULATIF

Les attributs de la balise `<frameset>` sont les suivants :

Attribut	Utilité
<b>rows ou cols</b>	Indique un découpage en zone horizontales (ROWS) ou verticales (COLS). Les valeurs sont une liste de valeurs séparées par une virgule, représentant la taille de chaque zone, soit en pourcentage, soit en pixel, (* permet d'ajuster la colonne ou la ligne à la taille de la

	fenêtre) Ex : <FRAMESET ROWS= "30%,30%,40%"> découpe l'écran en 3 frames occupant respectivement 30, 30 et 40% de la hauteur. Ex : <FRAMESET COLS= "100,*"> découpe l'écran en 2 colonnes, la première de 100 pixels et la seconde s'ajustant automatiquement à la fenêtre du navigateur..
<b>frameborder</b>	Valeurs <b>yes</b> ou <b>no</b> ; indique si les frames ont une bordure visible.
<b>border</b>	Valeur en pixel qui indique la taille de la bordure des frames.
<b>bordercolor</b>	Indique la couleur des bordures des frames.
<b>framespacing</b>	Valeur en pixel qui permet d'indiquer un espace entre les frames.

Les attributs de la balise <frame> sont les suivants :

Attribut	Utilité
<b>src</b>	Indique l'url de la page html qui sera affichée dans la frame.
<b>name</b>	Permet de donner un nom à la zone. Ce nom sera ensuite utilisé pour identifier la frame, à l'aide notamment de l'attribut TARGET
<b>marginwidth</b>	Valeur en pixel qui indique la taille des marges droite et gauche.
<b>marginheight</b>	Valeur en pixel qui indique la taille des marges haute et basse.
<b>frameborder</b>	Valeurs <b>yes</b> ou <b>no</b> ; indique si les frames ont une bordure visible.
<b>border</b>	Valeur en pixel qui indique la taille de la bordure des frames.
<b>bordercolor</b>	Indique la couleur des bordures des frames.
<b>noresize</b>	Empêche l'internaute de redimensionner les frames. Par défaut, les frames sont redimensionnables ; valeurs <b>yes</b> ou <b>no</b> .
<b>scrolling</b>	Permet de définir une barre d'ascenseur pour la frame. Les valeurs possibles sont : <b>yes</b> : un ascenseur est toujours visible <b>no</b> : l'ascenseur n'est jamais visible <b>auto</b> : l'ascenseur apparaît lorsqu'il est nécessaire.

## X LES CADRES FLOTTANTS : LES IFRAMES

Sur Internet Explorer, depuis la version 3.0, il est possible d'utiliser des cadres flottants, ce qui permet d'ouvrir une fenêtre à n'importe quel endroit de votre page.

On utilise pour cela la balise **<iframe>...</iframe>**. Cette balise doit être définie entre les balises **<body>** et **</body>**.

Les attributs de cette balise sont les suivants :

Attribut	Utilité
<b>src</b>	Indique l'url de la page html qui sera affichée dans la iframe.
<b>name</b>	Permet de donner un nom à la zone. Ce nom sera ensuite utilisé pour identifier la iframe, à l'aide notamment de l'attribut TARGET
<b>marginwidth</b>	Valeur en pixel qui indique la taille des marges droite et gauche.
<b>marginheight</b>	Valeur en pixel qui indique la taille des marges haute et basse.
<b>frameborder</b>	Valeurs <b>yes</b> ou <b>no</b> ; indique si les iframes ont une bordure visible.
<b>BORDER</b>	Valeur en pixel qui indique la taille de la bordure des iframes.
<b>bordercolor</b>	Indique la couleur des bordures des iframes.
<b>noresize</b>	Empêche l'internaute de redimensionner les iframes. Par défaut, les iframes sont redimensionnables.
<b>scrolling</b>	Permet de définir une barre d'ascenseur pour la iframe. Les valeurs possibles sont : <b>yes</b> : un ascenseur est toujours visible <b>no</b> : l'ascenseur n'est jamais visible <b>auto</b> : l'ascenseur apparaît lorsqu' il est nécessaire.
<b>height</b>	Définit la hauteur de la iframe insérée dans la page.
<b>width</b>	Définit la largeur de la iframe insérée dans la page.