

Secteur Tertiaire Informatique Filière étude - développement.

Développer des pages web en lien avec une base de données

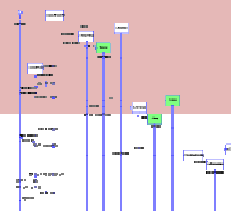
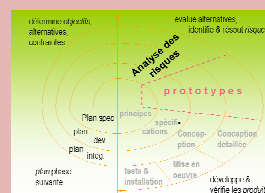
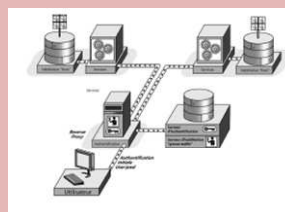
HTML - XHTML - (X)HTML5

Accueil

Apprentissage

Période en
entreprise

Evaluation



CODE BARRE

SOMMAIRE

Sommaire

I	Introduction	5
I.1	Qu'est-ce que le Web ?	5
I.2	Les navigateurs	5
I.3	Les éditeurs	6
I.4	Qu'est-ce que HTML ?	6
I.5	Qu'est-ce que XHTML ?	7
I.6	Qu'est-ce que HTML5 ?	8
I.7	Accessibilité des pages Web	9
I.8	Comment utiliser ce document ?	11
II	Structure d'une page HTML.....	12
II.1	HTML : un langage de balises	12
II.2	Structure d'un document HTML	12
II.3	DOCTYPE	13
II.4	Structure de page en HTML5	14
II.5	Structure et en-tête du document (<head>)	14
II.6	Le corps du document (<body>)	14
II.7	Le corps du document en HTML5.....	15
II.8	Les éléments en ligne	15
II.9	Les éléments de type bloc	16
II.10	Commenter le code HTML.....	16
II.11	Les couleurs de A à Z.....	16
III	Mise en forme des textes et paragraphes.....	18
III.1	Les balises <div> et 	18
III.2	Les paragraphes.....	18
III.3	Les titres	19
III.4	Les lignes horizontales	20
III.5	Mise en forme des caractères	20
III.6	Les polices de caractères HTML4/XHTML	22
III.7	Caractères spéciaux	23
IV	Les listes	26
IV.1	Liste ordonnée (listes à numéros).....	26
IV.2	Liste non ordonnée (listes à puces)	26
IV.3	Liste de définitions	27
IV.4	Combiner les types de listes	27
IV.5	Créer vos propres puces	28
V	Les images : la balise 	28
V.1	Les différents formats d'image.....	28

V.2	Afficher une image.....	29
V.3	Préciser la taille de l'image au navigateur	29
V.4	Afficher un texte de remplacement	30
V.5	Modifier la présentation de l'image	30
VI	Liens hypertextes	31
VI.1	La balise <a>	31
VI.2	Créer un lien à l'intérieur d'une même page	32
VI.3	Créer un lien entre deux pages	32
	Les liens relatifs.....	32
	Les liens absolus	33
VI.4	Créer un lien E-mail	33
VII	Multimedia	34
VII.1	Video	34
VII.2	Audio	35
VIII	Images réactives.....	36
IX	Les Tableaux.....	37
IX.1	Créer un tableau.....	38
IX.2	Alignement des données	39
IX.3	Regroupement (fusion) de lignes et de colonnes	40
IX.4	Présentation des cellules, mise en page	41
X	Nouveaux éléments HTML5	42
XI	Les cadres flottants : les iframes	43
XII	Les balises <meta>	45
XII.1	Principe.....	45
XII.2	Jeu de caractères	45
XII.3	Recharger une page régulièrement	46
XII.4	Faire une redirection vers une autre page	46
XII.5	Définir une date limite à votre page	46
XII.6	Interdire de mettre en cache votre page	46
XII.7	Indexer et documenter votre site	46
XIII	Referencement	48
XIII.1	Comment faire pour que Google connaisse l'existence de votre site ?	48
XIII.2	Comment être premier dans les résultats de Google ?.....	48
XIII.3	Comment propulser son site en tête des classements en 26 étapes?.....	49

I Introduction

I.1 QU'EST-CE QUE LE 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, le plus largement possible :

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

Pour créer un site Web, on n'utilisera pas un mais deux langages :

- HTML ou XHTML qui est le langage de base. Il va permettre essentiellement de structurer le contenu des pages à afficher.
- CSS (Cascading Style Sheets) : son rôle est d'assurer la présentation du site web. On utilise le CSS pour réaliser la mise en page du site, pour définir la police, la taille du texte, la couleur du texte et du fond, etc....

Ces 2 langages nous permettent de séparer le fond et la forme. Le fond est l'affaire du HTML ou XHTML et le CSS s'occupera de la forme. Cela a pour avantages de pouvoir changer facilement le design de son site, d'homogénéiser la mise en page du site (aspect plus professionnel), de rendre la navigation plus rapide (si le CSS est commun à chaque page, il n'est chargé qu'une fois).

I.2 LES NAVIGATEURS

En l'espace de quelques années, le navigateur est devenu le programme le plus utilisé sur un ordinateur. C'est lui en effet qui permet de « surfer » sur le Web.

Le rôle du navigateur est d'analyser le code HTML et CSS des pages web et d'en produire un résultat visuel. Le problème est que les navigateurs interprètent parfois le HTML et le CSS différemment. Il faut donc tester votre site sur plusieurs navigateurs.

Les navigateurs les plus utilisés sont :

- Chrome 42 %
- Safari 17 %
- Internet Explorer 13 %
- Firefox 20 %
- Android 6 %

Conseil : installez plusieurs navigateurs sur votre poste pour tester le fonctionnement de vos sites sous différents navigateurs.

I.3 LES EDETEURS

Il existe un grand nombre d'éditeurs de sites web. Nous pouvons les regrouper en 2 catégories :

- Les éditeurs WYSIWYG
- Les éditeurs de texte

Les éditeurs WYSIWYG (Why You See Is What You Get : ce que vous voyez est ce que vous obtenez) permettent de rédiger le contenu d'un site directement sans avoir à taper la moindre ligne de Xhtml et CSS. Parmi les plus connus, nous trouvons Nvu (gratuit et Open Source), Word, FrontPage, Adobe GoLive.... tous payants. La qualité du code produit par ces éditeurs est rarement bonne et le site est souvent difficile à maintenir. Les éditeurs sont « une solution de facilité » mais ils ne permettent pas toute la souplesse et les possibilités que l'on a en rédigeant soi-même le code.

Des éditeurs de texte peuvent être inclus dans des outils de développement (comme CodeLobster, VisualStudio, Aptana, etc...) ou être des éditeurs de texte « bruts » (comme bloc-Notes, Notepad++, PSPad, jEdit. Notepad++ ...).

I.4 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 (*Hyper Text Markup Language*) standardisé par le W3C (organisme de régulation du Web).

HTML est un langage qui permet donc de présenter des informations sur Internet. Ce que vous voyez quand vous regardez une page sur Internet est l'interprétation par votre navigateur du code HTML. Pour voir le code HTML d'une page sur Internet, choisissez « *afficher la source* ».

HTML va nous permettre de :

- publier des documents en ligne possédant des titres, du texte, des tableaux, des listes, des photos, etc.
- accéder à ces informations en ligne via des liens hypertextes en un clic de souris (ou en un effleurement d'écran tactile depuis HTML5).

- concevoir des formulaires pour conduire des transactions avec des services distants de gestion de données, dans le but de rechercher des informations, de faire des réservations, de prendre des commandes, etc.
- intégrer des documents bureautiques (comme des feuilles de calculs, des fichiers pdf) ou des documents multimédia (comme des séquences vidéo ou sonores) et même d'autres applications, directement dans les documents publiés.

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

HTML a bien évolué depuis les années 1990. On peut considérer que la version minimale actuelle est HTML4 (stable depuis 1999).

I.5 QU'EST-CE QUE XHTML ?

XHTML (*Extensible HyperText Markup Language*) est une évolution du langage HTML apparue en 2000 qui respecte la syntaxe et la rigueur du langage XML (*Extensible Markup Language*). XML permet de décrire des données quelconques organisées en arbre, indépendamment de leur représentation graphique.

XHTML reste un langage de présentation de données mais il est plus rigoureux que HTML : il exige que les données soient bien structurées en arbre, et sépare la structure du document contenu dans la page Web des détails de présentation qui seront définis par des « styles ». Il convient de séparer le fond de la forme !

Le fond est le document XHTML :

Structuration des éléments d'un document : titre, paragraphe, liste, tableau, ...

Formatage de données textuelles de nature sémantique : emphase, citation, code, définition,

La forme est contenue dans une feuille de style associée à la page XHTML :

Aspects de rendu visuel : police, taille et couleur des caractères, ...

Espacement et placement sur la page des différents éléments

Les outils de structuration du contenu d'un document doivent être utilisés en relation avec leur sémantique :

Les listes ne sont pas des tableaux et inversement ...

Les tableaux ne sont pas un moyen de placement dans une page XHTML, ils ne servent qu'à structurer des éléments tabulaires

Voici quelques règles simples pour passer d'HTML à XHTML. Les cinq premières seront mises en œuvre de suite, la 6ème exige l'utilisation des styles :

1. n'omettre aucune des balises qui définissent la structure en arbre du document (html, body...)
2. fermer toutes les balises
3. les balises et les attributs sont toujours en minuscules
4. les valeurs des attributs sont entre guillemets

5. tous les attributs ont des valeurs explicites
6. les balises indiquent la structure du document et non pas sa présentation. Les balises et les attributs de présentation sont remplacés par l'application de styles.
7. les éléments vides doivent avoir un "terminateur". Par exemple <hr> devient <hr/>.
8. toujours utiliser & pour & dans des attributs, y compris les URL par exemple
9. le chevauchement d'environnements n'est pas autorisé.

Syntaxe correcte :

```
<h1><a href= " .... ">Votre titre      </a></h1>
```

Syntaxe incorrecte :

```
<h1><a href= " .... ">Votre titre</h1></a>
```

10. Les outils de structuration du contenu d'un document doivent être utilisés en relation avec leur sémantique :
 - Les listes ne sont pas des tableaux et inversement ...
 - Les tableaux ne sont pas un moyen de placement dans une page XHTML, ils ne servent qu'à structurer des éléments tabulaires

I.6 QU'EST-CE QUE HTML5 ?

HTML5 est la toute nouvelle version du langage HTML. Il a comme objectif de remédier à certains désagréments des versions précédentes d'HTML, et a été motivé par quelques constats simples :

- Les navigateurs ont besoin de plugins pour gérer le multimédia (audio et vidéo Flash, Silverlight, etc...),
- HTML ne structure pas correctement les documents ce qui rend l'accessibilité difficile,
- Les API JavaScript varient d'un navigateur à l'autre ce qui pose problème pour la portabilité des documents,
- HTML ne permet pas de développer des sites qui se rapprochent des applications de bureaux (de type Desktop).

HTML5 a donc comme objectif de répondre à ces problématiques et de favoriser l'émergence du web sémantique. Pour cela, il va permettre de structurer la mise en page par des éléments plus sémantiques, et harmoniser l'usage des médias en intégrant de nombreuses fonctionnalités auparavant disponibles sous forme de plugins.

En octobre 2014, la standardisation du langage est enfin stabilisée mais, fait unique dans l'histoire des Techniques, l'industrie a anticipé depuis longtemps sur ce que serait le langage et toutes les versions récentes des navigateurs Web (en particulier sur les appareils mobiles -smartphones et tablettes-) supportent plus ou moins bien

les nouveautés du langage. Il faudra encore quelques années pour que HTML5 puisse être totalement stabilisé et considéré comme LE langage standard du Web. HTML5 apporte principalement :

- Une simplification de la syntaxe (oubliée la rigueur de XHTML -sous prétexte d'alléger le code-!)
- Le support des nouvelles interfaces graphiques (écrans tactiles) et des ergonomies modernes (glisser/déplacer)
- Un enrichissement des *contrôles graphiques* permettant le dialogue avec l'utilisateur (curseurs, saisie de dates, d'adresse email, de couleur...) et des fonctions de contrôle de saisie automatique sans programmation JavaScript
- Une meilleure structuration de la page Web favorisant son auto-adaptation à l'écran utilisé pour l'affichage (la même page peut présenter des rendus différents mais cohérents sur un écran de PC, de tablette ou de smartphone ; on parle de *responsive design*)
- Un support direct et standardisé de données audio, vidéo et animation (adieu les plug-in comme FlashPlayer)
- Une utilisation systématique des composants graphiques des machines pour alléger le processeur principal, ce qui autorise par ailleurs le *dessin vectoriel* de bien meilleure qualité que le *dessin bitmap*
- Une capacité à fonctionner temporairement hors connexion au réseau en stockant des (lots de) pages en local sur le poste utilisateur
- Une capacité à gérer en local des lots de données (recherches, filtrages, tris) sans nécessité d'allers-retours au serveur
- Des fonctionnalités standardisées pour l'aide au référencement par les moteurs de recherche
- L'intégration des fonctions de géo-localisation et de niveau de batterie des appareils mobiles -ces derniers points nécessitant de la programmation JavaScript-

Et **XHTML5** ? Des travaux sont en cours pour standardiser HTML5 selon la rigueur et la syntaxe XML ; on peut s'attendre à ce que XHTML5 prévu en 2014 ou 2015 rectifie les faiblesses de HTML5 ; *c'est dans cette logique que ce document est présenté : on s'efforcera dès maintenant d'adopter de bonnes pratiques dans la forme du code HTML5.*

I.7 ACCESSIBILITE DES PAGES WEB

Le respect des recommandations du XHTML assure non seulement l'accessibilité d'un site sur la majorité des navigateurs actuels, mais garantit surtout sa pérennité. En effet, il est risqué d'utiliser du code approximatif et des balises obsolètes ou propriétaires, car ces techniques ne sont plus reprises dans les normes actuelles du langage HTML telles que définies par le W3C.

L'universalité suppose que tous les êtres humains devraient pouvoir accéder aux sites web dans la mesure de leurs moyens. Cela concerne aussi les personnes souffrant de handicaps, quelle que soit leur nature (physique, auditif, visuel ou moteur). Il existe à cet effet des standards et des normes d'accessibilité du Web, dites WCAG : <http://www.w3.org/WAI/>

On peut classer les handicaps affectant l'accès au Web en quatre catégories :

- Les déficiences visuelles rassemblent les aveugles, malvoyants, daltoniens et porteurs de lunettes. Les illustrations dépourvues d'intitulé ou de texte de remplacement, les polices de caractères trop petites ou les couleurs peu contrastées limitent alors l'accès à l'information.
- Les déficiences auditives des sourds ou malentendants les empêcheront de profiter de certaines informations si le créateur du site n'a pas prévu à leur effet des dispositions d'accessibilité.
- Des handicaps physiques gênent certains utilisateurs dans le maniement du clavier ou de la souris. Ceux-ci seront donc exclus de tout site exigeant de pointer avec précision de petits éléments ou imposant la souris pour interagir avec leurs scripts et menus.
- Les déficiences mentales ou neurologiques ralentiront certains en l'absence de repères clairs et précis. Un système de navigation non intuitif pourra lui aussi troubler de nombreux utilisateurs. L'abus d'effets visuels comme les clignotements ou les animations de fréquence élevée peut quant à lui avoir de graves conséquences sur des sujets sensibles ou épileptiques.

Tout lieu public se doit d'être accessible à tous, handicapés ou non. Les cahiers des charges concernés comportent désormais ces obligations. Pourquoi en irait-il autrement du Web ?

Pour un Web accessible à tous, le W3C a publié en 1997 une initiative (dite WAI) visant à garantir la prise en charge des questions d'accessibilité par les nouvelles technologies.

C'est dans cette optique qu'ont été rédigées les 14 directives du **WCAG** (Web Content Authoring Guidelines). Elles décrivent les principes généraux d'accessibilité et les points à respecter lors de l'élaboration d'un site web.

Quelques points à respecter pour assurer une meilleure accessibilité :

- Ne jamais poser de limites à la navigation : un menu exclusivement graphique, utilisant des outils propriétaires, reposant sur des scripts ou nécessitant des plug-ins (Java, Flash, JavaScript) est à éviter, à moins de proposer une autre solution, fonctionnelle sans ces outils.
- Éviter les structures de pages utilisant des cadres (frames, iframes), véritables écueils pour les visiteurs malvoyants.
- Toujours proposer des solutions équivalentes au contenu visuel et sonore : texte de remplacement pour les images (alt), navigation sans souris ou sans clavier (accesskey, tabindex), liens hypertextes explicites (title), etc.
- Ne pas s'en remettre exclusivement aux couleurs et permettre d'augmenter la taille du texte (c'est utile aux malvoyants).
- Utiliser un balisage sémantique pour offrir une structure cohérente, même si l'aspect visuel est absent ou dégradé. Respecter les normes de langage en vigueur proposées par le W3C et séparer le contenu (HTML) de la mise en forme (CSS).

I.8 COMMENT UTILISER CE DOCUMENT ?

Ce document tente de répondre à deux objectifs :

- comprendre le langage HTML de base pour maintenir et faire évoluer des sites Web existants.
- Apprendre à construire les nouveaux sites Web conformes aux standards XHTML et HTML5 ou aussi près que possible de ces normes, tout en étant conscient que les navigateurs supportent très partiellement les nouveautés.

Nous avons, dans ce document, essayé dans la mesure du possible, de nous limiter aux balises et attributs utilisés en HTML5. Certaines balises ou attributs qui traitent des aspects présentation, et qui sont avantageusement remplacés par des styles CSS, ont néanmoins été gardées, un commentaire signalant dans ce cas les écarts avec les normes HTML4 ou XHTML

Ce document se limitera à l'étude des fonctionnalités purement HTML ; mais aujourd'hui, HTML seul ne suffit pas car il est fait pour être utilisé conjointement avec CSS pour la présentation graphique et JavaScript pour les interactions avec l'utilisateur (réactions événementielles, contrôles de saisie) ou la bonne exploitation du matériel (géo-localisation, stockage en local). Des supports de formation spécifiques à ces techniques traiteront donc de certains aspects du langage HTML non abordés ici.

II STRUCTURE D'UNE PAGE HTML

II.1 HTML : UN LANGAGE DE BALISES

HTML est un langage de balises. Toutes les instructions sont exprimées dans des balises (ou "tags"), qui sont encadrées par des crochets (signe inférieur < et signe supérieur >). En général, chaque balise doit être ouverte et fermée de manière à encadrer le contenu sur lequel doit s'appliquer la balise. Les balises de fermeture sont les mêmes que les balises d'ouverture, mais sont précédées d'un caractère slash (/).

Exemple : **<head> ceci est le contenu de la balise head </head>**

*Une paire de balises ouvrantes et fermantes définit un **élément** de la page Web ; les deux termes sont bien souvent confondus ; on emploie plutôt le terme **d'élément** quand on se préoccupe du **contenu**, de la portion de page, et on préfère le terme **balise** quand on se focalise sur la **syntaxe** du langage.*

En HTML (4 ou 5), la casse (minuscule ou majuscule), n'a aucune importance. Il est cependant conseillé d'écrire les balises en minuscules afin de faciliter le respect du standard XHTML. De même, lorsque vous écrivez un document HTML, pensez à respecter les niveaux d'indentation, et veillez aux balises d'ouverture et de fermeture afin d'obtenir des documents lisibles.

En HTML, jamais d'erreur de syntaxe, que des erreurs d'affichage !

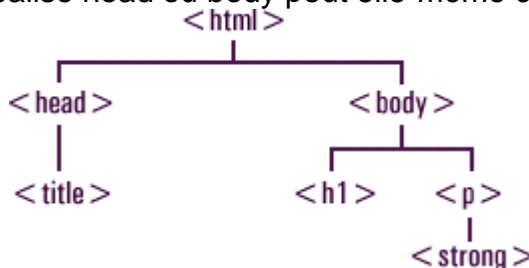
Pour assurer la compatibilité avec les nouvelles versions du langage, un principe de base de HTML est **qu'une balise non reconnue est systématiquement ignorée**. Ainsi, il ne faut pas hésiter à employer dès maintenant les nouvelles balises HTML5, c'est une assurance pour le futur qui ne coûte rien dans le présent.

II.2 STRUCTURE D'UN DOCUMENT HTML

Un document HTML4 se compose de plusieurs parties :

- une ligne DOCTYPE (facultative) contenant les informations de la version HTML.
- puis un élément <html>, qui comprend :
 - ✓ une section en-tête déclarative (balise <head>)
 - ✓ un corps, qui comporte le contenu effectif du document. Le corps est enclos par l'élément <body>.

Chaque balise head ou body peut elle-même contenir des balises incluses.



Voici un exemple de document HTML4 simple :

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN">
<html>
  <head>
    <title>Mon premier document HTML</title>
  </head>
  <body>
    <p>Bonjour tout le monde !</p>
  </body>
</html>
```

II.3 DOCTYPE

Avant de commencer à écrire l'en-tête, votre document principal (ou chaque page de votre projet, mais ce n'est pas obligatoire) doit comporter une instruction doctype qui permet de définir le standard HTML que vous allez utiliser ; ceci est utile pour renseigner le navigateur car il est compatible avec de nombreuses versions du langage.

Commande	Version HTML
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">	HTML 4.0.1 Transitional
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">	HTML 4.0.1 Strict
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">	HTML 4.0.1 FrameSet
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">	XHTML1.0 Transitional
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">	XHTML1.0 Strict
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">	XHTML 1.0 FrameSet

Ensuite, vous devez entrer la balise **<html>** qui indique le début de votre fichier, puis la balise **<head>** dans laquelle sont résumées des informations générales, c'est à dire le titre de votre page affiché en barre de titre du navigateur (c'est aussi le titre qui apparaîtra dans les favoris du visiteur, s'il ajoute cette page à sa liste de favoris) et les balises qui permettent d'indexer (référencer) votre site (ces balises seront étudiées plus loin).

II.4 STRUCTURE DE PAGE EN HTML5

En HTML5, la syntaxe est simplifiée à l'extrême :

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

II.5 STRUCTURE ET EN-TETE DU DOCUMENT (<HEAD>)

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

II.6 LE CORPS DU DOCUMENT (<BODY>)

Il est à noter que, jusqu'à cette balise, le navigateur ne visualisera rien, car les véritables données qu'il va afficher vont se trouver dans la balise <body>. Cette balise détermine le corps du document : c'est ici que vous allez intégrer titres, textes, images, tableaux, liens...

Depuis HTML4, la présentation n'est plus à définir dans le code source HTML mais doit être plutôt spécifiée dans les feuilles de style CSS selon la syntaxe CSS. Les attributs suivants sont donc donnés à titre d'information

Le tag <body> a de nombreux attributs :

- **bgcolor** : Définit la couleur du fond de l'écran
- **text** : Définit la couleur du texte
- **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
- **bgproperties** : Si définie comme "fixed", l'image de fond ne défile pas (uniquement avec Internet Explorer)
- **leftmargin** : Définit la largeur de la marge de gauche en pixels
- **topmargin** : Définit la largeur de la marge du haut en pixels

II.7 LE CORPS DU DOCUMENT EN HTML5

En HTML5, le corps d'un document (élément <body>) peut être structuré en différentes parties afin de *donner du sens* aux éléments de la page et de *faciliter l'auto-adaptation de la page à la taille et à la résolution du navigateur* qui effectue la présentation (on parle de « *responsive design* ») ; attention toutefois que le rendu final correct nécessitera des informations complémentaires grâce aux styles CSS :

- **<section>** : une section est un groupe thématique de contenus incluant généralement des titres (présentation générale, news, informations de contact...). Attention : cette balise ne sert pas à regrouper des éléments à des fins de présentation. Une page peut contenir une ou plusieurs sections ;
- **<article>** : un article est une portion de page qui se suffit à elle-même, c'est-à-dire qu'elle conserve son sens même si elle est déplacée ou séparée des éléments situés autour d'elle (comme par exemple un article de journal, de blog ou de forum) ; il est à noter qu'une <section> peut contenir un ou plusieurs <article> mais qu'un <article> peut aussi contenir une ou plusieurs <section> ! tout dépend du *sens* que l'auteur donne à ces portions de document ;
- **<header>** : une entête <header> représente un ensemble pouvant inclure une introduction et/ou des éléments de navigation. Elle contient en général un titre <hxx> et peut contenir une table des matières, un formulaire de recherche ou des logos ;
- **<footer>** : cet élément contient typiquement des informations sur la section ou la page à laquelle il appartient, comme par exemple l'auteur ou des liens vers d'autres documents ou encore des éléments de copyright ;
- **<nav>** : barre de navigation. Cet élément correspond à des liens vers d'autres pages ou vers les différentes parties de la page actuelle. Une balise <nav> doit contenir un bloc de navigation majeur tel qu'un menu ou un « fil d'Ariane » et pas seulement une suite de liens ;
- **<aside>** : cet élément représente une portion de document dont le contenu a une relation avec le contenu principal, sans pour autant être essentiel (tout comme un classique « encadré » dans une page de magazine). Si la balise <aside> est située dans une section particulière, elle traitera du même sujet que la section ; au contraire, si elle est située à l'extérieur d'une section, elle traitera du même sujet que le site (elle peut alors contenir par exemple des publicités ou des éléments de navigation).

HTML5 ne spécifie pas de règles strictes pour l'imbrication de ces balises de structure : par exemple une <section> peut contenir un <article> et vice-versa, une <section> peut contenir un <header> contenant un <nav>, en plus d'un <aside> contenant lui aussi un <nav>.

II.8 LES ELEMENTS EN LIGNE

Que l'on soit en HTML 4 ou HTML5, le corps du document va contenir un certain nombre de balises. Ces éléments peuvent avoir un affichage en ligne, ou par bloc.

Les éléments en ligne sont affichés au fil du texte (dans le flux courant). Ils peuvent être des portions de phrase ``, des mises en exergue ``, des liens `<a>`, des citations `<q>` et des images ``. Nous allons voir l'ensemble de ces balises plus loin.

Ils ne sont pas prévus pour un positionnement correct au sein du document mais plutôt pour apporter du sens.

Ils ont des marges internes et externes nulles par défaut.

II.9 LES ELEMENTS DE TYPE BLOC

Le rendu visuel de ces éléments forme un bloc. Cette structure leur permet de prévoir des dimensions (hauteur, largeur) et une profondeur dans le plan.

Ils peuvent contenir d'autres éléments et posséder des marges internes « padding ». Ils font office de conteneur.

La propriété la plus importante est la possibilité de les placer (positionner) hors du flux courant du document

Ils ont des marges internes et externes par défaut non nulles en dehors de la balise `<DIV>`.

Il est donc parfois nécessaire de préciser explicitement ces marges (voire les rendre nulles) pour assurer un rendu visuel correct.

Parmi ces éléments, nous trouvons :

Les balises `<div>` : conteneur générique elle permet de structurer la présentation des autres éléments. Elle est souvent assimilée à la notion de calque. Il est vrai que l'on peut jouer avec ces balises pour simuler plusieurs couches dans l'espace (profondeur) du document avec des possibilités de chevauchement.

Les titres `<h1>` à `<h6>` : contrairement aux autres éléments de type bloc ils ne sont pas des conteneurs.

Les listes `` ordonnées ou ``

Les paragraphes `<p>`

Les tables `<table>`

II.10 COMMENTER LE CODE HTML

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.

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

II.11 LES COULEURS DE A A Z

Les trois couleurs de base RGB (rouge, vert, bleu) permettent de composer les millions de couleurs affichables par l'écran. Les proportions varient selon un nombre

exprimé en hexadécimal, pour chacune des trois couleurs, de "00" à "FF" ; il y a donc 256 nuances pour chacune des couleurs. La combinaison des 3 couleurs peut ainsi créer plus de 16 millions de teintes différentes. Certaines couleurs sont nommées et on peut aussi bien utiliser le nom HTML que le code de couleur en hexadécimal comme dans l'extrait suivant :

Couleur	Valeur RGB
black	#000000
silver	#C0C0C0
gray	#808080
white	#FFFFFF
maroon	#800000
red	#FF0000
purple	#800080
fuchsia	#FF00FF
green	#008000
lime	#00FF00
olive	#808000
yellow	#FFFF00
navy	#000080
blue	#0000FF
teal	#008080
aqua	#00FFFF

NB : HTML5 introduit encore de nombreux noms de couleurs supplémentaires mais on doit bien constater que tout cela est très mal supporté par les navigateurs ; il est donc toujours préférable de privilégier les valeurs RGB.

Voir le site :

<http://www.code-couleur.com/>



mise en application :

Reproduire le programme suivant en mode texte (utiliser NOTEPAD++ ou tout autre éditeur).
Exécuter le programme avec un navigateur.
Rechercher la signification et les attributs des balises.
Ajouter et tester quelques attributs de présentation à la balise body.

```
<html>
  <head>
    <title>Un essai du langage HTML </title>
  </head>
  <body>
    <p>Ce document HTML est très simple</p>
  </body>
</html>
```


III MISE EN FORME DES TEXTES ET PARAGRAPHES

III.1 LES BALISES <DIV> ET

Les balises **<div>** est un conteneur générique qui permet de structurer la présentation des autres éléments. Elle peut inclure tous les tags html (tels que les paragraphes les tableaux, les listes, les images ..., et également d'autres divisions). Cet élément est l'un des plus utilisé pour structurer les pages html (en HTML4) en plusieurs grandes sections (en tête, partie principale, partie navigation, pied de page ...).

Elle est aussi assimilée à la notion de calque. Il est vrai que l'on peut jouer avec ces balises pour simuler plusieurs couches dans l'espace (profondeur) du document avec des possibilités de chevauchement. Son utilisation entraîne un effet visuel : utilisation de toute la largeur disponible et retour à la ligne précédent et suivant son contenu. Il est possible de supprimer cet effet en appliquant le style CSS "[display : inline;](#)".

Les balises **** permettent d'identifier une portion de phrase. Cet élément s'emploie fréquemment au sein de phrases pour encadrer et mettre en valeurs des lettres, mots ou groupes de mots. A l'inverse de la balise <div>, son utilisation n'entraîne aucun effet visuel (affichage *in-line*).

Ces deux balises <div> et possèdent un certain nombre d'attributs :

Attribut	Utilité
id	Identifiant de la balise.
class	liste de classes séparées par des espaces. Utilisé pour appliquer un style
style	Information locale de style
title	Permet une information supplémentaire concernant l'élément.
align	Permet l'alignement du texte (left, center, right, justify) (cet attribut est déprécié, et remplacé par des styles CSS)

III.2 LES PARAGRAPHES

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

La balise double **<p>...</p>** permet de définir un paragraphe, et la balise **
** effectue un retour à la ligne à l'intérieur du même paragraphe.

Le paragraphe délimité par `<p>...</p>` est devenu l'élément de base de la structuration d'un document HTML4/XHTML.

Nous suivrons donc une « bonne pratique » en fermant toutes nos balises, comme dans le langage XML.

La balise `
` ne comportant jamais de contenu, nous utiliserons la notation XHTML `
` qui ouvre et ferme immédiatement la balise en définissant un *élément sans contenu*.

Le texte d'un paragraphe peut être aligné à l'aide de l'attribut de présentation **align** de la balise `<p>`.

Depuis HTML4, ceci étant de l'ordre de la présentation et non du contenu, il est plutôt conseillé d'utiliser un style CSS pour aligner un texte.

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.

Attention : en XHTML, aucun texte ne doit figurer directement au premier niveau du document, sous la balise `<body>`. Tout contenu (texte, image...) doit être rangé dans un contenant : paragraphe, titre, table, div et span (qui seront vus dans la suite du document).

III.3 LES TITRES

Les balises titre permettent de structurer et mettre en évidence des parties de texte selon un formatage défini par le navigateur. Plusieurs styles de titre sont prédéfinis, identifiés par les balises doubles `<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 (changement de paragraphe) sera fait automatiquement par le navigateur après l'affichage d'un titre `<hxx>`. Les balises titre possèdent aussi l'attribut de présentation **align** (**left**, **right**, **center**).

Exemple et code source avec chaque balise titre :

`<h1>Titre de niveau 1</h1>` **Titre de niveau 1**

`<h2>Titre de niveau 2</h2>` **Titre de niveau 2**

`<h3>Titre de niveau 3</h3>` **Titre de niveau 3**

`<h4>Titre de niveau 4</h4>` *Titre de niveau 4*

`<h5>Titre de niveau 5</h5>` **Titre de niveau 5**

<h6>Titre de niveau 6</h6>

Titre de niveau 6

Attention : en XHTML, les niveaux de titre doivent être utilisés dans l'ordre hiérarchique.

Avec HTML5 apparaît la notion de groupe d'éléments d'entête définie par la balise **<hgroup>** **</hgroup>** ; là encore, il ne s'agit que de donner du sens au contenu et cette balise n'a aucune représentation graphique par défaut.

III.4 LES LIGNES HORIZONTALES

On peut insérer des lignes horizontales ('filets') pour séparer certaines parties de notre document. Ceci se fait à l'aide de la balise **<hr>**. Comme pour **
, en XHTML il faut fermer la balise pour être conforme à la norme : **<hr />

Depuis HTML4, la présentation n'est plus à définir dans le code source HTML mais doit être plutôt spécifiée dans les feuilles de style CSS selon la syntaxe CSS. Les attributs suivants sont donc donnés à titre d'information

Elle possède les attributs suivants :

Attribut	Utilité
align	Permet d'aligner la ligne horizontale. Obsolète en XHTML strict. Alternative : le style <i>text-align</i> .
width	Spécifie la largeur de la ligne (en pixels ou % de fenêtre). Obsolète en XHTML strict. Alternative : le style <i>width</i> .
size	Epaisseur de la ligne (de 1 à 10). Obsolète en XHTML strict. Alternative : le style <i>size</i> .
noshade	S'il est présent, l'effet d'ombre est annulé. Obsolète en XHTML strict. Alternative : les styles <i>border</i> en particulier <i>border-style</i> .
color	Spécifie la couleur de la ligne (uniquement Internet Explorer, non reconnu en HTML4)

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" />
```



III.5 MISE EN FORME DES CARACTERES

1 / La mise en forme logique HTML4/XHTML

En utilisant une balise de **style logique**, le développeur donne un sens au contenu de l'élément sans spécifier de mise en forme particulière ; le navigateur va donc décider du formatage du texte.

Attribut	Utilité
<code><cite> ... </cite></code>	Pour effectuer une citation d'un nom d'œuvre ou d'auteur
<code><code> ... </code></code>	Pour indiquer du code informatique
<code> ... </code>	Emphase (reproduit en italique)
<code><kbd> ... </kbd></code>	(Keyboard, Clavier) utilisé généralement pour indiquer une réponse d'utilisateur
<code><var> ... </var></code>	Indique une variable
<code> ... </code>	Permet de faire ressortir du texte
<code><samp> ... </samp></code>	Affiche les caractères de manière littérale

2 / La mise en forme physique HTML4/XHTML

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

Attribut	Utilité
<code> ... </code>	Affiche le texte en gras. A éviter : il est préférable d'utiliser le style <i>font-weight : bold</i>
<code><i> ... </i></code>	Affiche le texte en italique. A éviter : il est préférable d'utiliser le style <i>font-style : italic</i>
<code><u> ... </u></code>	Souligne le texte. Il est préférable d'utiliser le style <i>text-decoration:underline</i>
<code><strike> ... </strike></code>	Barre le texte. Il est préférable d'utiliser le style <i>text-decoration:line-through</i>
<code><big> ... </big></code>	Affiche le texte dans une police plus grande. N'existe plus en XHTML 2.0
<code><small> ... </small></code>	Affiche le texte dans une police plus petite
<code><sup> ... </sup></code>	mise en indice d'un texte (caractères abaissés)
<code><sub> ... </sub></code>	mise en exposant d'un texte (caractères surélevés)

Depuis HTML4, la mise en forme physique n'est plus à définir dans le code source HTML mais doit être plutôt spécifiée dans les feuilles de style CSS selon la syntaxe CSS.

2 / Le nouveau sens HTML5 des balises de mise en forme

La plupart des balises précédentes sont supprimées du langage HTML5 (car elles ont de bien meilleurs équivalents en CSS) mais *certaines ont changé de sens* :

Attribut	Utilité
<code> ... </code>	Pour faire ressortir visuellement

<code><i> ... </i></code>	Autour des mots étrangers ou techniques et pour inclure les mots d'un autre utiliser un ton différent
<code> ... </code>	Pour donner de l'importance à un texte
<code> ... </code>	Pour prononcer un texte avec emphase
<code><big> ... </big></code>	Permet d'utiliser une police plus grande
<code><small> ... </small></code>	A utiliser pour les petites lignes (de contrats par exemple) ou les notes de bas de page
<code><cite> ... </cite></code>	Pour effectuer une citation d'un nom d'œuvre uniquement

Selon les spécifications du langage HTML5, ces balises n'ont pas de rendu prédéfini et le développeur doit spécifier la mise en forme des éléments par feuilles de styles CSS. Mais on peut imaginer le casse-tête pour les navigateurs que représente la nécessaire compatibilité avec toutes les versions précédentes du langage HTML et il faut s'attendre à ce que, en HTML5, ces balises produisent par défaut les rendus habituels HTML4/XHTML.

III.6 LES POLICES DE CARACTERES HTML4/XHTML

Depuis HTML4, la présentation n'est plus à définir dans le code source HTML mais doit être plutôt spécifiée dans les feuilles de style CSS selon la syntaxe CSS (la propriété CSS précisant la police est font-family, la propriété CSS font-size gère la taille du texte, et la propriété CSS color permet de changer la couleur du texte).

Les balises suivantes sont donc données à titre d'information uniquement.

Pour spécifier une police de caractères, on peut utiliser la balise ****.

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 standards communes à toutes les plateformes ou de préciser une liste de polices similaires spécifiques à chaque plateforme.

La balise `` possède les attributs suivants :

Attribut	Utilité
face	Permet de définir par ordre de priorité, les noms de polices souhaitées
size	Permet de définir la taille de la police de 1 à 7 (valeur par défaut : 3) Il peut être utilisé de deux manières : <ul style="list-style-type: none"> - Absolue ex : size = 2 - Relative ex : size = +2
color	Spécifie la couleur du texte (avec le nom de la couleur, ou la valeur hexadécimale)

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

A noter que HTML5 introduit avec CSS3 la notion de police de caractère téléchargeable qui sera automatiquement redescendue depuis le serveur par le navigateur si nécessaire.

III.7 CARACTERES SPECIAUX

Le Web a une dimension planétaire et les différentes langues nationales n'utilisent pas le même alphabet ; de plus, certains signes simples comme le symbole de copyright ou les symboles d'unité monétaire sont régulièrement utilisés.

« L'alphabet du Web » va donc plus loin que le code de base de chaque machine ; il permet ainsi de définir et représenter correctement tous nos caractères accentués français.

Il est conseillé de préciser tout d'abord le jeu de caractères utilisé dans une page grâce à une balise adéquate dans l'entête HTML :

- En HTML4/XHTML :
`<meta http-equiv="Content-Type" content="text/html ; charset=utf-8" />`
- En HTML5 :
`<meta charset="utf-8" />`

Attention aussi de toujours bien enregistrer vos codes sources en format UTF-8 (voir les options de l'éditeur utilisé).

Voici une liste des principaux caractères accentués français. Pour que l'accent soit reconnu correctement par tous les navigateurs, il faut le remplacer dans le texte par « l'entité HTML » associée :

Caractère	Entité	Caractère	Entité	Caractère	Entité	Caractère	Entité
À	à	Î	î	Ç	ç	Ù	ù
Á	á	Ï	ï	È	è	Ú	ú
Â	â	Ö	ð	É	é	Û	û
Ã	ã	Ñ	ñ	Ê	ê		
Ä	ä	Ô	ô	Ë	ë		
Æ	æ	Ö	ö	Ü	ü		

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 le code ISO ou l'entité HTML associée.

Caractère	Code ISO	Entité HTML
©	©	©
,	‚	
§	§	§
„	„	
...	…	
£	£	£
¶	¶	¶
.	·	·
¼	¼	¼
½	½	½
¾	¾	¾
€	€	€
‘	‘	
’	’	
“	“	
”	”	
•	•	
–	–	
—	—	
Espace insécable	 	
<	<	<
>	>	>



Exercice d'application

Reproduire le programme suivant en mode texte

Tester sur au moins deux navigateurs Web : par exemple Internet Explorer et FireFox.

Rechercher au besoin la signification de chaque balise ou attribut inconnu.

```

<!DOCTYPE html>
<html>
  <head>
    <title>La mise en page du langage HTML </title>
  </head>
  <body>
    <h2>Ce document</h2>
    <h3>HTML</h3>
    <h4>est encore très simple</h4>
    <p>son objectif est de montrer les principales balises de mise en forme du
      texte. Les lignes précédentes ont mis en évidence l'action des balises
      <hxx> qui permettent de créer et disposer des titres de taille
      plus ou moins grande.
    </p>

    <hr />
    <p>La balise <hr /> permet l'affichage d'une ligne horizontale
      <hr />
    </p>

    <div>La balise <div> : Voici un texte écrit dans une balise
      <div> . La balise div (pour division) peut être utilisée
      pour diviser un document en plusieurs sections qui seront, par exemple,
      présentée chacune de manière différente.
      La présentation (alignement, couleur, marges, etc...) sera défini via des
      feuilles de style CSS.
    </div>

    <hr />

    <p>La balise <span> permet d'identifier un bout de texte afin de lui
      appliquer un <span style="color:red;">style</span> (ici, la couleur
      rouge, à l'aide d'un style CSS dont on verra la syntaxe plus tard.)</p>

    <hr />
    <p>La balise <p> : permet de créer un nouveau paragraphe. La différence
      entre la balise p et div se situe au niveau du type d'enfants acceptés:
      à l'inverse de la division qui n'impose aucune restriction, un paragraphe
      ne peut contenir que des éléments de type inline
    </p>
    <hr />
    <p align="center">On peut utiliser l'attribut align (ici center),</p>
    <p align="center">mais ceci est déconseillé : </p>
    <p align="center">il est préférable de passer par des styles CSS</p>

    <hr />
    <p>La balise <br /> <br /> permet un retour à la ligne</p>

  </body>
</html>

```


IV LES LISTES

HTML propose depuis toujours des balises permettant de reproduire des mises en formes et mises en page usuelles en traitement de texte ; nous avons vu précédemment les lignes de séparation horizontales (<hr />), nous abordons ici les listes (ou énumérations), numérotées ou à puces et nous verrons plus loin la mise en page à l'aide de tableaux.

IV.1 LISTE ORDONNEE (LISTES A NUMEROS)

Les listes ordonnées s'obtiennent avec la balise ** .. **. Cette balise permet de définir la liste numérotée, avec une incrémentation automatique. Chaque élément de la liste sera repéré par la balise **... **.

Exemple de définition d'une liste à numéros :

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

Le résultat est :

1. Rouge
2. Vert
3. Bleu

La balise **** possède les attributs suivants en XHTML :

- **type = "1"** : les items sont 'numérotés' par des nombres (valeur par défaut)
- **type = "a"** : les items sont 'numérotés' par des lettres minuscules
- **type = "i"** : les items sont 'numérotés' en chiffres romains (minuscules)

Anciennes options disponibles en HTML 4 :

- **type = "A"** : les items sont 'numérotés' par des lettres majuscules
- **type = "I"** : les items sont 'numérotés' en chiffres romains (majuscules)
- **start = "n"** : valeur de départ de la numérotation

IV.2 LISTE NON ORDONNEE (LISTES A PUCES)

Les listes non ordonnées s'obtiennent avec la balise ** .. **. 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 **... **.

Exemple de définition d'une liste à puces :

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

```
<li>Bleu</li>
</ul>
```

Le résultat est

- Rouge
- Vert
- Bleu

La balise **** possède les attributs suivants :

- **type = "disc"** : puce en forme de disque plein
- **type = "circle"** : puce en forme de disque vide
- **type = "square"** : puce en forme de carre

compact="compact" : spécifie une liste compacte (est déprécié en XHTML)

IV.3 LISTE DE DEFINITIONS

Une liste de termes et de définitions correspondantes s'obtient avec la balise **<dl>...</dl>**, qui est accompagnée des balises **<dt>... </dt>** et **<dd>... </dd>**. **<dt>** représente le terme, et **<dd>** la définition du terme.

Exemple d'une liste de définition avec 1 terme et 3 définitions :

```
<dl>
  <dt>Terme</dt>
  <dd>def1</dd>
  <dd>def2</dd>
  <dd>def3</dd>
</dl>
```

Le résultat est (l'indentation est automatique) :

```
Terme
  def1
  def2
  def3
```

IV.4 COMBINER LES TYPES DE LISTES

Vous pouvez bien entendu combiner plusieurs listes entre elles.

Exemple d'une liste ordonnée combinée avec une liste de définition :

```
<ol type="i">
  <li>
    item1
    <dl>
      <dt>Terme</dt>
      <dd>def1</dd>
    </dl>
  </li>
</ol>
```

IV.5 CREER VOS PROPRES PUCES

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

- soit vous passez par l'attribut **src=** "puce.gif" de la balise **** (si le fichier graphique de votre puce se nomme 'puce.gif'). Attention, cet attribut n'est pas standard.
- soit vous imbriquez la balise **** dans une balise ****, en définissant votre image comme un simple fichier gif ou jpeg. Cette solution est la meilleure, car valable pour tous les navigateurs.
- Soit vous utilisez les **styles CSS** qui permettent aisément de spécifier le type de fichier image à utiliser (*recommandé depuis XHTML*).

Exemple d'une liste avec un fichier gif :

```
<ol type="i">  
  <li>item1</li>  
  <li>item2</li>  
</ol>
```

V LES IMAGES : LA BALISE

V.1 LES DIFFERENTS FORMATS D'IMAGE

Si les pages Web sont parfois longues à charger ce n'est pas parce qu'elles contiennent beaucoup de texte mais à cause des images. Une photo de 120 Ko correspond à 120 000 caractères, soit environ 24 000 mots !

Avec l'ADSL, nous avons tendance à oublier que certaines personnes se connectent encore en bas débit mais aussi via les satellites dans les régions les plus reculées. Pour toutes ces personnes il est nécessaire de faire attention au poids total des pages web. De manière générale, il faut veiller à ne pas dépasser 150 Ko d'images sur une page si nous ne voulons pas que nos visiteurs fuient le site avant de l'avoir découvert.

L'objectif est donc d'avoir des fichiers images le plus légers possible. Pour cela, nous utiliserons des algorithmes de compression.

C'est au webmaster qu'il incombe de choisir le format d'image approprié. Selon celui que l'on choisit, le poids de l'image et sa qualité peuvent beaucoup varier.

Tous les logiciels de retouche d'image vous laissent le choix du format dans lequel vous voulez enregistrer votre image.

Les 3 principaux formats que l'on utilise sur le Web sont :

- ✓ JPEG
- ✓ GIF

✓ PNG

Il faut oublier les BMP. Ces images sont non compressées donc elles prennent beaucoup de place et sont du coup très longues à charger.

V.2 AFFICHER UNE IMAGE

Les images d'une page Web sont stockées dans des fichiers différents de celui de la page Web elle-même ; elles sont acheminées à part et nécessitent des demandes de transferts spécifiques entre le navigateur et le serveur Web, ce qui ralentit l'affichage pour l'utilisateur.

L'affichage d'une image s'effectue à l'aide de la balise ****. La balise **** a de nombreux attributs, le plus important est **src**, qui permet de spécifier l'url (chemin d'accès) complète du fichier. La balise **** n'a pas de contenu et doit donc être immédiatement fermée, comme **
** et **<hr />** :

Ex : ****

Nom de fichier contient le nom du fichier 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 depuis un autre site. Il faut pour cela donner l'adresse complète du serveur :

Ex : **<img src= "<http://www.afpa.fr/gif/logo.gif>" />**

Attention : en Web pas de barre oblique inverse \ (antislash) « à la Windows ».

V.3 PRECISER LA TAILLE DE L'IMAGE AU NAVIGATEUR

On peut préciser la taille de l'image au navigateur, par l'intermédiaire des attributs **width** et **height**. De ce fait, le navigateur chargera et affichera plus rapidement votre page en « réservant » l'espace nécessaire avant même que l'image ne soit reçue. De plus si l'internaute a désactivé le chargement des images, la mise en page ne sera pas perturbée. Ces dimensions doivent correspondre à la taille en pixels 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 et l'image peut s'en trouver déformée !). Il est donc judicieux de stocker de suite les images sur le serveur Web avec la bonne taille et la bonne résolution, éventuellement en autant de tailles et de définitions que nécessaires (notion de *vignette*).

Attention, comme tous les attributs de présentation, width et height ne doivent plus apparaître dans le code source HTML mais doivent être plutôt spécifiés dans les feuilles de style CSS selon la syntaxe CSS.

V.4 AFFICHER UN TEXTE DE REMPLACEMENT

Cela se fait avec l'attribut **alt**.

Deux avantages : si l'image ne s'affiche pas, l'internaute pourra quand même comprendre le sens de l'image, et de plus, on peut par ce biais fournir une légende à une image, qui s'affiche dans une 'info-bulle' lorsque la souris passe dessus l'image.

Ex : ``

Attention : cet attribut est obligatoire, depuis HTML4 pour une bonne accessibilité (adaptation aux handicaps).

V.5 MODIFIER LA PRESENTATION DE L'IMAGE

Depuis HTML4, la présentation n'est plus à définir dans le code source HTML mais doit être plutôt spécifiée dans les feuilles de style CSS selon la syntaxe CSS. Les attributs suivants sont donc donnés à titre d'information

Entourer l'image :

Avec l'attribut **border**, vous pouvez entourer l'image d'une bordure. L'attribut **border** prend une valeur de 0 à x qui exprime la largeur en pixels de la bordure entourant l'image.

Ex : ``

Espacer l'image :

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 (*gouttière*).

Ex : ``

Positionner et aligner l'image :

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 (« habillage », style journal), l'attribut **align** est requis. Il reste 5 valeurs possibles, en HTML4 et

suivant (d'autres attributs peuvent être reconnus par certains navigateurs mais ne sont pas standards) :

Valeur	Effet
left	Image à gauche, texte appuyé à droite
right	Image à droite, texte appuyé à gauche
middle	Le bas du texte s'aligne avec le milieu de l'image.
bottom	Le bas de l'image s'aligne avec le bas des lettres 'non descendantes'
top	Le haut de l'image s'aligne avec le haut de la 1 ^{ère} lettre du texte

Ex :

<p>Texte placé à gauche de l'image</p>

VI LIENS HYPERTEXTES

Les liens hypertextes sont à la base du Web ; ils permettent à l'utilisateur de passer d'une page à l'autre par un simple clic sur un mot, une phrase une image... en fait, sur n'importe quelle portion de page.

VI.1 LA BALISE <A>

La balise <a> permet d'établir un lien hypertexte, c'est-à-dire un lien vers une autre partie de la page courante, vers une autre page de notre site, ou vers une autre page d'un autre site.

Elle possède les attributs suivants :

Attribut	Utilité
href	Permet d'indiquer l'adresse de la page (ou l'endroit de la page) où l'on veut se débrancher - notion d'URL
title	Permet de donner une description du lien (info-bulle)
style	Permet de spécifier le style de présentation à utiliser
target	Utilisé dans le cas des Frames pour définir quel est le cadre (frame) dans lequel on souhaite afficher
name	Permet de définir un signet ou ancre à l'intérieur d'une page HTML

VI.2 CREER UN LIEN A L'INTERIEUR D'UNE MEME PAGE

Les liens internes permettent de naviguer directement au sein d'une même page HTML, sans nécessiter l'utilisation des ascenseurs. Depuis un endroit du document (c'est la source du lien), lorsque l'on va cliquer on va se débrancher vers la cible définie (c'est *l'ancre* ou *signet*).

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

La cible se définit par : `` ou ``

La source se définit par : `Cliquez ici pour aller vers la cible`

Tout ce qui est entre `<a href...>` et `` (texte, image...) devient alors zone sensible.

VI.3 CREER UN LIEN ENTRE DEUX PAGES

On peut aussi naviguer entre différentes pages HTML, qu'elles fassent partie du même site Web ou qu'elles se trouvent dans un autre site ; c'est bien le principe même de Web, la « toile d'araignée » qui relie les pages du monde entier. Un lien se réalise en précisant *l'adresse* de la page cible. Les adresses de pages sont conformes à la notion **d'URL** (*Uniform Resource Locator*) qui définit sans ambiguïté toute ressource d'Internet.

Pour faire un lien d'une page vers une autre, 2 façons d'écrire l'adresse de la page s'offrent à nous : la notation relative ou absolue.

Les textes seront écrits en bleu souligné.

Lorsque vous cliquez sur un lien, la nouvelle page s'affiche dans la même fenêtre du navigateur. Depuis l'apparition du langage XHTML, il n'est plus possible de faire en sorte que la nouvelle page s'affiche dans une nouvelle fenêtre.

Ils ont estimé que c'était au visiteur de décider si oui ou non il souhaitait ouvrir le lien dans une nouvelle fenêtre;

Pour qu'un lien s'ouvre dans une nouvelle fenêtre de Firefox ou IE, il faut maintenir la touche Shift enfoncée lorsque nous cliquons.

Si nous retournons sur la page d'origine, nous constaterons que le lien est désormais en violet. C'est le navigateur qui a changé la couleur. Cela indique au visiteur que c'est un lien qu'il a déjà visité. Il sera possible de changer la couleur à l'aide des feuilles de style.

Il est possible d'ajouter une info bulle (bulle d'aide) sur le lien. Cette bulle est un petit message qui s'affiche lorsque le visiteur pointe avec la souris sur le lien. Pour créer une info bulle sur un lien, il faut ajouter à la balise `<a>` l'attribut *title*.

Le texte de la bulle d'aide doit être assez court (quelques mots). Il est inutile de rédiger un roman car il sera tronqué par le navigateur.

L'attribut *title* peut être utilisé sur la plupart des balises.

Les liens relatifs

Dans le cas d'un lien relatif, nous ne connaissons pas l'adresse complète de la page vers laquelle on amène.

Si tous les fichiers .htm sont dans le même dossier (répertoire), nous écrivons :

< a href="page2.htm" >Cliquez ici pour aller vers la page 2

La page page2.htm se trouve dans un sous-dossier nommé *pages* :

**< a href ="pages/page2.htm" > Cliquez ici **

Si la page pages2.htm se trouve dans le dossier situé avant page1.htm dans l'arborescence :

**< a href ="../page2.htm" > Cliquez ici **

Pour aller vers l'ancree 'cible' définie dans une autre page

< a href =" page2.htm#cible" >Cliquez ici

Pour créer un lien en vue de télécharger un fichier

< a href =" projet.zip" >Télécharger si le fichier projet.zip se trouve dans le même dossier que le fichier .htm.

Les liens absolus

Il faut connaître l'adresse complète de la page cible.

Une adresse de page web commence la plupart du temps par *http://* et quelques fois par *https://* pour les pages sécurisées.

< a href =" http://www.afpa.fr" >Aller sur le site de l'AFPA

Ce type de lien est le seul utilisable pour faire un lien vers un autre site que le sien. La notation relative ne fonctionne que pour les liens internes à son site, tandis que la notation absolue fonctionne dans tous les cas.

Si les liens absolus fonctionnent dans tous les cas, quel intérêt d'utiliser les liens relatifs ?

Si nous ne connaissons pas encore l'adresse de notre site (nous ne sommes pas encore hébergés) ou si notre site change d'adresse, les liens relatifs sont plus commodes car nous n'aurons rien à modifier car l'adresse n'y figure pas.

VI.4 CREER UN LIEN E-MAIL

On peut par ce biais *lancer automatiquement le logiciel de messagerie e-mail de l'internaute*. Les messages peuvent être pré-renseignés, notamment le sujet, les personnes en copie, et le texte du message :

lien email sans sujet :

Ecrivez-nous

lien email avec sujet :

`Ecrivez-nous`

lien email avec sujet et personne en copie :

`<a href=`
`mailto:marc.fayolle@ifrance.com?cc=regis.lecu@ifrance.com&subject=commande>`
`Ecrivez-nous`

lien email avec sujet et texte associé:

`<a href="mailto:marc.fayolle@ifrance.com?subject=commande&body=Suite à votre`
`dernière commande, etc....." >Ecrivez-nous`

VII MULTIMEDIA

Le langage HTML est un langage permettant l’affichage de documents et la navigation au sein de ces documents. Il permet d’afficher des pages ‘statiques’. Il est néanmoins possible depuis toujours de mettre un peu d’animation dans ces pages statiques à l’aide de bandeaux défilants (balise `<marquee>`) ou de fichiers d’animation de type « GIF animé » (limité à 256 couleurs). Des animations plus élaborées peuvent être obtenues par l’intégration d’applets Java, de scripts JavaScript, ou par l’utilisation d’outils d’animation non standards (de type Flash).

HTML 5 dispose de son propre lecteur vidéo ou audio, personnalisable par CSS, ce qui va nous dispenser de plugins pour les intégrer dans une page. La vidéo est aujourd’hui implémentée sur le web grâce à différents plugins (Silverlight, Quicktime, WMP et surtout Flash). HTML5 nous propose une simple balise `<video>`, qui nous permettra d’intégrer une vidéo dans une page, mais aussi d’afficher une image de remplacement, de charger le fichier en tâche de fond et d’aller directement en un point de la vidéo ou la bande son.

Il reste à résoudre des problèmes d’harmonisation entre les différents navigateurs (codec, formats de fichiers) avant de pouvoir généraliser leur usage.

VII.1 VIDEO

L’élément `<video>` est utilisé pour le traitement de contenu multimédia de type vidéo.

Cette balise comporte de nombreux attributs, dont les principaux :

- **src** : URL du fichier vidéo.
- **preload**="auto", qui permet de spécifier au navigateur de débiter le téléchargement de la vidéo tout de suite
- **poster** : affiche une image lorsque le média n’est pas disponible.
- **autoplay**="true" : joue le fichier automatiquement.

- **loop** : joue en boucle.
- **controls** : affiche les contrôles.

Cette balise peut aussi contenir la sous-balise **<source>**, afin de spécifier des contenus alternatifs.

Pour les vidéos, plusieurs formats sont actuellement utilisés : WebM, MP4, Ogg Theora. Attention, tous les navigateurs ne supportent pas tous les formats de vidéo.

VII.2 AUDIO

L'élément **<audio>** est utilisé pour le traitement de contenu multimédia de type audio.

Les principaux attributs de cet élément sont les suivants :

- **src** : URL du fichier audio.
- **autoplay** : joue le fichier automatiquement.
- **loop** : joue en boucle.
- **controls** : affiche les contrôles.

Pour les fichiers audios, plusieurs formats sont actuellement utilisés et supportés par tel ou tel navigateur.

VIII IMAGES REACTIVES

Une image peut être définie comme étant une image réactive côté client, c'est-à-dire une image qui peut être découpée en zones, chaque zone pouvant réagir différemment lors du click.

L'image **** est définie comme étant une image réactive à l'aide de l'attribut **usemap** :

```
<img src = "../images/pic1.gif" usemap = "#nom" />
```

La **map** est définie comme suit :

```
<map id="nom"> (ou <map name="nom">)  
  <area shape="rect" coords="10,10,49,49" href = "page1.htm" />  
  <area shape="rect" coords="60,10,99,49" href = "page2.htm" />  
  .....  
</map>
```

L'identificateur de la **map** est celui qui a été défini dans l'image pour l'attribut **usemap** (sans le signe #). La MAP est découpée en un certain nombre de zones par la balise **<area>**.

Chaque **area** définit une zone sensible et possède les attributs suivants :

Attribut	Utilité
shape	Définit la forme de la zone. Les valeurs possibles sont : rect : rectangle circle : cercle poly : polygone
coords	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 angle pour un polygone.
href	Définit la page qui s'affiche lors du click sur la zone

IX LES TABLEAUX

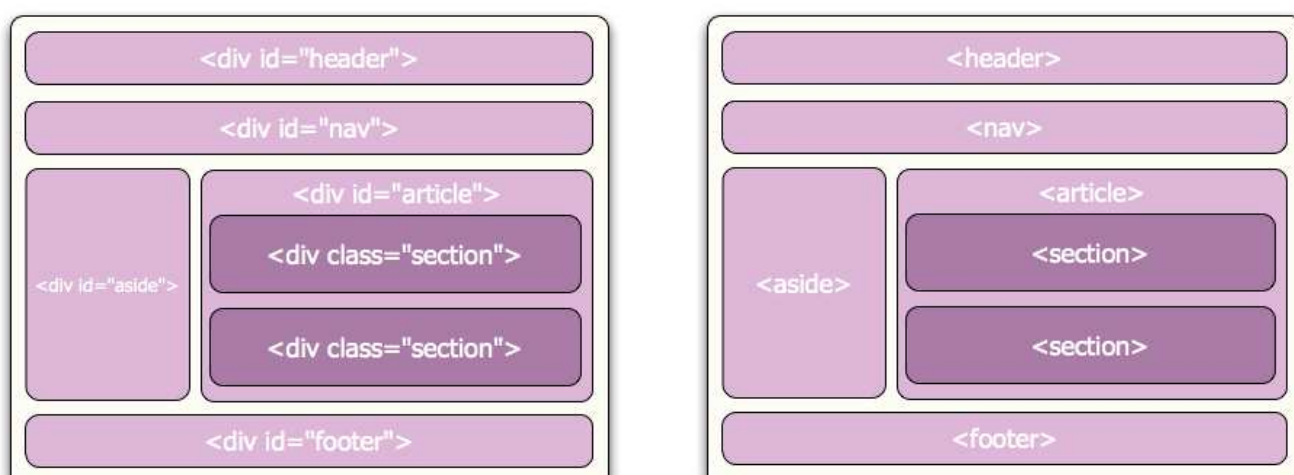
Un principe de base du Web est que le **rendu final de la page est calculé au dernier moment par le navigateur en fonction de la taille de l'écran ou de la fenêtre et de la résolution graphique du périphérique**. Le développeur doit toujours raisonner en proportions et en pourcentages et non en nombre de pixels pour effectuer la mise en page.

Dans un premier temps, les tableaux HTML ont pendant longtemps été LE moyen d'assurer et de contrôler la mise en page des pages Web : mise en colonnes, juxtaposition de textes et d'images... et les fonctionnalités sont très correctes (titre au tableau, fusion de cellules, encadrements...). Les tableaux HTML sont dotés d'un bon rendu standard par tous les navigateurs mais leur inconvénient principal est une certaine rigidité dans leur structure : les sites Web changent souvent de look et il est bien difficile de faire évoluer la structure d'une page définie par des tableaux HTML ;

Dans un second temps (HTML4/XHTML), la tendance forte a ensuite été de définir les parties d'une page à l'aide d'imbrications de balises <div>, mises en page à l'aide de styles CSS, nécessitant un travail conséquent avant d'obtenir un bon résultat.

Enfin, HTML5 apporte des éléments de solution à ces problèmes de mise en page avec de nouvelles balises de structure définissant des parties fonctionnelles de la page (<section> et <article> pour le contenu, <header> pour les entêtes, <nav> pour les blocs de navigation...). Une mise en forme à l'aide des styles CSS reste toutefois nécessaire (voir plus loin la mise en œuvre).

L'utilisation d'HTML 5 nous permet donc, en terme de structuration de page, d'éviter l'utilisation de balises div :



IX.1 CREER UN TABLEAU

On crée un tableau HTML4 à l'aide de la balise `<table> ... </table>`

Depuis HTML4, on doit définir les différentes colonnes du tableau à l'aide de balises `<col />` (éventuellement regroupées dans des `<colgroup> </colgroup>`).

Depuis HTML4, un tableau doit contenir au moins un groupe de lignes défini par `<tbody> </tbody>`. On peut donner du sens aux groupes de lignes grâce aux variantes `<thead>..</thead>` et `<tfoot> </tfoot>` qui représentent les entêtes et pied de tableau.

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>`

L'entête des colonnes (c'est-à-dire chaque cellule de la ligne de titre) est repérée par les balises `<th> .. </th>` au lieu de `<td>..</td>` qui réalise automatiquement une mise en évidence centrée et en gras.

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

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

```
<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8" />
  <title>Liste des clients</title>
</head>
<body>
  <table border="1">

    <caption>Liste des clients</caption>

    <thead>
      <tr>
        <th>Nom</th>
        <th>Prénom</th>
        <th>Adresse</th>
      </tr>
    </thead>

    <tbody>
```

```

        <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>
    </tbody>

    <tfoot>
        <tr>
            <td></td>
            <td>Nombre de clients</td>
            <td>3</td>
        </tr>
    </tfoot>

</table>
</body>
</html>

```

Attention de bien décrire toujours autant de cellules pour chaque ligne (sous peine d'erreurs ou de refus d'affichage) !

IX.2 ALIGNEMENT DES DONNEES

Attention, ces attributs sont obsolètes : l'alignement des données se spécifie en utilisant des feuilles de style CSS.

On peut influencer sur l'alignement des données dans un tableau par l'intermédiaire des attributs **align** et **valign**. Ces attributs peuvent prendre les valeurs suivantes :

Attribut	Utilité
align = left	Valeur par défaut. Les données sont alignées à gauche de la cellule (pour la balise td), ou de toutes les cellules de la ligne (pour la balise tr)
align = right	Les données sont alignées à droite de la cellule (pour la balise td), ou de toutes les cellules de la ligne (pour la balise tr)
align = center	Les données sont alignées au centre de la cellule (pour la balise td), ou de toutes les cellules de la ligne (pour la balise tr)
valign = top	Les données sont alignées en haut de la cellule (pour la balise td), ou de toutes les cellules de la ligne (pour la balise tr)

valign = bottom	Les données sont alignées en bas de la cellule (pour la balise td), ou de toutes les cellules de la ligne (pour la balise tr)
valign = middle	Les données sont alignées verticalement au centre de la cellule (pour la balise td), ou de toutes les cellules de la ligne (pour la balise tr)
valign = baseline	Valable que pour la balise tr . Les données des cellules sont toutes alignées.

IX.3 REGROUPEMENT (FUSION) DE LIGNES ET DE COLONNES

On peut regrouper plusieurs cellules, aussi bien en largeur qu'en hauteur, afin d'obtenir des cellules plus grandes. Ceci s'effectue à l'aide des attributs **rowspan** et **colspan** des balises `<td>`.

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
DUBUIS	Julie	Grenoble

```
<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8" />
  <title>Liste des clients</title>
</head>
<body>
<table border="1">
  <caption>Liste des clients</caption>
  <col /><col /><col />
  <tbody>

    <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> <!--cette cellule compte
pour 2 sur cette ligne-->
    </tr>

    <tr>
      <td>DUBOIS</td>
      <td>Claire</td>
      <td rowspan="2">même adresse : 25 cours Berriat,38000 Grenoble</td>
    <!--cette cellule compte pour 2, sur 2 lignes-->
  </tbody>
</table>
```

HTML XHTML (X)HTML5

```

</tr>
<tr>
  <td>DUBUIS</td>
  <td>Julie</td>
</tr>
</tbody>
</table>
</body>
</html>

```

IX.4 PRESENTATION DES CELLULES, MISE EN PAGE

Attention, tous les attributs ci-dessous sont obsolète, la présentation et la mise en page étant spécifiée en utilisant des feuilles de style CSS.

Un tableau HTML correctement structuré aura toujours un rendu correct ; différents attributs des balises **<table>**, **<td>**, **<th>** sont disponibles pour influencer sur la présentation des tableaux :

Attribut	Utilité
width	Indique la largeur du tableau ou d'une cellule (en pixel ou en %)
height	Indique la hauteur du tableau ou d'une cellule (en pixel ou en %)
border	Taille de la bordure du tableau (1 par défaut)
cellpadding	Indique l'espace entre la bordure et le contenu de chaque cellule
cellspacing	Indique l'espace entre les cellules du tableau
bgcolor	Spécifie la couleur d'une cellule
bordercolor	Spécifie la couleur de la bordure
bordercolorlight	Spécifie la couleur claire de la bordure (3D)
bordercolordark	Spécifie la couleur sombre de la bordure (3D)
background	Url de l'image à afficher en fond (IE uniquement)
frame	<p>Permet d'afficher des bordures sur un ou plusieurs cotés du tableau. Il faut que l'attribut border soit défini dans la balise <table>. Les valeurs possibles sont :</p> <p>above : Affiche les bordures hautes seulement. below : Affiche les bordures basses seulement border : Affiche toutes les bordures box : Affiche une boîte autour du tableau insides : Affiche les bordures internes hautes et basses hsides : Affiche les bordures externes hautes et basses lhs : Affiche les bordures gauches seulement. rhs : Affiche les bordures droites seulement. void : Retire les bordures externes. vsides : Affiche les bordures externes gauches et droites.</p>

rules	<p>Permet d'afficher des bordures à l'intérieur du tableau. Il faut que l'attribut border soit défini dans la balise <table>. Les valeurs possibles sont :</p> <p>none : Retire toutes les bordures internes</p> <p>rows : Affiche une bordure horizontale entre chaque ligne</p> <p>cols : Affiche une bordure verticale entre chaque colonne</p> <p>groups : Affiche une bordure horizontale entre chaque section</p> <p>all : Affiche toutes les bordures internes du tableau</p>
--------------	---

NB : on conservera la notion de tableau HTML quand les informations à afficher sont véritablement des tableaux de données (tableaux de gestions, statistiques...).

X NOUVEAUX ELEMENTS HTML5

L'élément **<time>** **</time>** permet de donner du sens à une information de type heure ou date et heure ; elle n'implique aucune représentation graphique par défaut.

```
<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8" />
  <title>Date et Heure HTML5</title>
</head>

<body >
<h3> Element time brut : format imposé</h3>
<p> ... nous ouvrons nos portes à <time>10:00</time>.</p>
<hr />
<h3> Element time avec attribut datetime : format libre</h3>
<p> Notre prochaine réunion aura lieu <time datetime="2013-06-23">à la pleine Lune de juin </time>
.</p>
<hr />
</body>
</html>
```

La balise **<figure>** **</figure>** combinée avec **<figcaption>** **</figcaption>** permet grouper et de donner un sens à une image et son titre ou sa légende, comme dans l'exemple suivant :

```
<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8" />
  <title>Figures HTML5</title>
</head>
<body >
<h3>Notre sélection d'images</h3>
<p>
  <figure>
    
    <figcaption>Ampoule éteinte</figcaption>
  </figure>
</p>
<p>
```

```

<figure>
  
  <figcaption>Ampoule allumée</figcaption>
</figure>
</p>
<hr />

</body>
</html>

```

En guise de cerise sur le gâteau, les nouvelles balises **<details>** et **<summary>** permettent de reproduire en standard (quand elle est supportée...) la fonctionnalité de bloc à 2 états condensé/étendu comme dans l'exemple suivant :

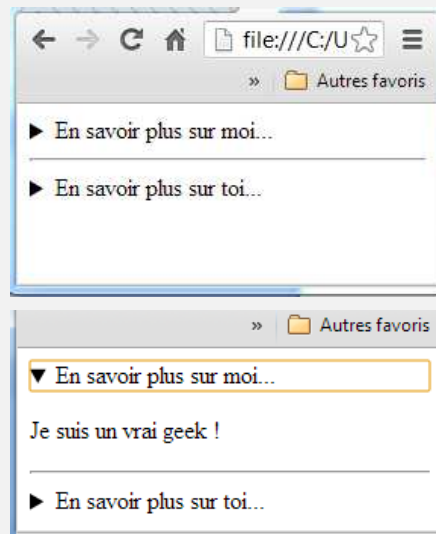
```

<!doctype html>
<html lang="fr">
<head>
  <meta charset="utf-8" />
  <title>details/summary HTML5</title>
</head>
<body >
<details>
  <summary>En savoir plus sur moi...</summary>
  <p>Je suis un vrai geek !</p>
</details>

<hr />
<details>
  <summary>En savoir plus sur toi...</summary>
  <p>Je ne sais rien de toi !</p>
</details>

</body>
</html>

```



XI LES CADRES FLOTTANTS : LES IFRAMES

Une balise **<iframe>...</iframe>** permet d'ouvrir une fenêtre imbriquée à n'importe quel endroit de votre page. Cette portion de page peut accueillir n'importe quelle ressource (article, image, page du même site, page d'un autre site...). Au besoin, le navigateur dotera l'iframe des barres de défilement nécessaire à la consultation de son contenu.

Les attributs de cette balise sont les suivants :

Attribut	Utilité
src	Indique l'url de la page html qui sera affichée dans la iframe.
name	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
marginwidth	Valeur en pixel qui indique la taille des marges droite et gauche.
marginheight	Valeur en pixel qui indique la taille des marges haute et basse.

frameborder	Valeurs yes ou no ; indique si les iframes ont une bordure visible.
border	Valeur en pixel qui indique la taille de la bordure des iframes.
bordercolor	Indique la couleur des bordures des iframes.
noresize	Empêche l'internaute de redimensionner les iframes. Par défaut, les iframes sont redimensionnables.
scrolling	Permet de définir une barre d'ascenseur pour la iframe. Les valeurs possibles sont : yes : un ascenseur est toujours visible no : l'ascenseur n'est jamais visible auto : l'ascenseur apparaît lorsqu' il est nécessaire.
height	Définit la hauteur de la iframe insérée dans la page.
width	Définit la largeur de la iframe insérée dans la page.

Cette balise faisait partie de HTML4 mais elle a été abandonnée en XHTML (on peut reproduire des effets similaires avec les styles CSS). Elle est remise d'actualité en HTML5...



Exercice d'application :

Reproduire les programmes suivant
Exécuter les programmes avec le navigateur.

Page12.htm

```
<html>
<head>
<title>les Iframes</title>

</head>

<body>

<h3 align="center">Deux formations en informatique sont proposées</h3>
<h3 align="center">à l'AFPA de Pont-de-claix.</h3>
<p/>
<hr/>
<iframe src="page12-1.htm" align="left" width="200" height="100" scrolling="yes">
</iframe>

<iframe src="page12-2.htm" align="left" width="200" height="100" scrolling="yes"
>
</iframe>

<p/>
<hr/>
<p> Pour plus d'information contacter
<a href="mailto:ana-prog@alpes-net.fr">ana-prog@alpes-net.fr</a>
</p>

</body>
</html>
```

Page12-1.htm

```
<html>
<head>
<title>DL</title>
</head>
<body bgcolor = "#FFFF00">
<p/>
<h5>Développeur Logiciel</h5>
<p>Détails de la formation :</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
</body>
</html>
```

Page12-2.htm

```
<html>
<head>
<title>CDI</title>
</head>
<body bgcolor =" #FFFF00">
<h5>Concepteur Développeur Informatique</h5>
<p>Détails de la formation :</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
<p>BLABLABLABLABLABLABLABLABLABLA</p>
</body>
</html>
```

XII LES BALISES <META>

XII.1 PRINCIPE

La balise **<meta>** fait partie des *informations de services* placées en entête de la page Web (**<head>**). Elle permet de fournir une description de votre site par le biais de mots-clés et de phrases. De plus, les balises **<meta>** ont d'autres fonctionnalités, comme notamment la redirection automatique vers une autre URL.

XII.2 JEU DE CARACTERES

Le Web est mondial et doit s'adapter à toutes les langues et tous les alphabets. Pour assurer le bon affichage des caractères spéciaux (ne serait-ce que nos lettres accentuées !), il est nécessaire de préciser le jeu de caractères utilisé (c'est à dire la correspondance entre le codage en binaire sous forme de 1 et de 0 et le caractère à afficher). Il est recommandé actuellement d'utiliser le jeu de caractère normalisé "utf-8" en introduisant en début de section **<head>** la balise **<meta>** correspondante :

```
<meta http-equiv="Content-type" content="text/html ;charset=utf-8" />  
<meta charset="utf-8" /> (en HTML5)
```

De plus, il est aussi nécessaire de s'assurer que l'éditeur utilisé enregistre bien le fichier au format utf-8 (voir les options du logiciel).

XII.3 RECHARGER UNE PAGE REGULIEREMENT

L'exemple suivant permet d'actualiser et de rafraîchir une page toutes les 5 secondes (pour obtenir des résultats à jour, une cours de bourse actualisé...) :

```
<meta http-equiv="refresh" content="5" />
```

XII.4 FAIRE UNE REDIRECTION VERS UNE AUTRE PAGE

Permet de charger une autre page html après un certain temps exprimé en secondes :

```
<meta http-equiv="refresh" content="10 ; url=http://www.voila.fr" />
```

Avec un attribut *content* définissant une durée de 0 secondes, la redirection est immédiate (utile en cas de changement d'adresse !).

XII.5 DEFINIR UNE DATE LIMITE A VOTRE PAGE

Avec la valeur '**expires**' de l'attribut **http-equiv**, il est possible de définir une date limite de validité de vos pages, et donc d'indiquer aux routeurs et serveurs proxy qu'il doivent impérativement recharger la page originale à partir de la date indiquée :

```
<meta http-equiv="expires" content="Wed, 19 Nov 2013 14:01:00 GMT" />
```

XII.6 INTERDIRE DE METTRE EN CACHE VOTRE PAGE

Avec les valeurs '**pragma**' et '**no-cache**', il est possible d'interdire la mise en cache de vos pages ; elles seront alors systématiquement rechargées depuis le serveur Web à chaque consultation :

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

XII.7 INDEXER ET DOCUMENTER VOTRE SITE

L'une des principales utilisations des balises **<meta>** est l'indexation des sites Web dans les moteurs de recherche. A l'aide des attributs **keywords**, **description**, **author**, **lang**, vous facilitez la tâche du robot logiciel qui viendra analyser la page.

Voici un exemple d'indexation commenté. Si votre site est en plusieurs langues, vous devez créer autant de balises **<meta>** que de langues.

Pour spécifier l'auteur du site et la langue utilisée :

```
<meta name="author" lang="fr" content="Marc Fayolle" />
```

Pour décrire le site en une phrase de référence :

```
<meta name="description" content="Tapez ici en quelques phrases la description de votre site" />
```

Une liste de mots-clés correspondant au mieux à votre site :

<meta name="keywords" lang="fr" content="Tapez ici les mots-clés séparés par une virgule" />

L'url complète de votre site :

<meta name="Identifier-URL" content="<http://www.afpa.fr>" />

L'adresse électronique qui se rapporte à ce site :

<meta name="reply-to" content="webmaster@afpa.fr" />

Les copyrights :

<meta name="copyright" content="COPYRIGHT NOM" />

Les outils de développement utilisés :

<meta name="generator" lang="fr" content="WebExpert, Gif Animator" />

Cette balise identifie le code ISO définissant le jeu de caractères à utiliser :

<meta name="content-type" content="text/html ;charset=ISO-8859-1" /> Ici, il s'agit du jeu de caractères 'Latin 1' comprenant nos lettres accentuées

Pour diriger les robots de référencement :

<meta name="robots" content="index, nofollow" /> Ici le robot indexe la page, mais n'indexe pas les pages qui ont un lien depuis cette page...

Les différentes valeurs possibles pour l'attribut robots sont :

Attribut	Utilité
all	Le robot indexe tout
none	Stoppe le robot
index	Votre page est indexée
noindex	Votre page n'est pas indexée, mais il suivra les liens de cette page
follow	Le robot récupère les liens de cette page pour les référencer
nofollow	Le robot ne récupère pas les liens de cette page

XIII REFERENCEMENT

C'est l'ensemble d'actions destinées à attirer des visiteurs sur un site. Un référenceur pourra notamment vous proposer :

- l'optimisation de votre site afin de faciliter son indexation par les outils de recherche,
- la soumission du site dans les outils de recherche,
- le positionnement de votre site dans les outils de recherche
- la promotion du site à l'aide d'articles presse, de la publicité, du marketing viral, etc.
- une veille concernant votre positionnement ou celui de vos concurrents.

XIII.1 COMMENT FAIRE POUR QUE GOOGLE CONNAISSE L'EXISTENCE DE VOTRE SITE ?

La réponse est : rien du tout, ça va se faire tout seul. Google explore en permanence la toile pour recenser les sites qui viennent de se créer. S'il existe un seul lien d'un site connu vers le vôtre, Google suivra ce lien et découvrira votre site. Il lui faudra entre une et huit semaines pour ajouter votre site à son index et à ses résultats de recherche.

Si vous voulez tenter d'accélérer ce processus, vous pouvez déclarer votre site vous-même en remplissant le formulaire d'inscription Google, mais le plus efficace reste de créer des liens vers votre site, grâce aux inscriptions dans les annuaires ou aux échanges de liens.

XIII.2 COMMENT ETRE PREMIER DANS LES RESULTATS DE GOOGLE ?

Il faut tout d'abord réaliser que vous n'êtes pas le seul à désirer cela ! Et puisque devenir premier revient à faire reculer tous les autres sites dans les classements, commencez par réaliser un bon site, riche, intéressant, esthétique, etc.

Il vous faudra ensuite comprendre comment Google classe ses résultats et vous débrouiller pour que votre site corresponde aux critères de choix de ce moteur.

Il vous faudra beaucoup lire, beaucoup apprendre, beaucoup réfléchir et beaucoup travailler. Au risque de vous décevoir, et à moins que votre site ne se positionne sur un secteur où vous êtes le seul, il vous faudra compter un minimum de 6 mois de travail acharné pour effectuer un bon référencement et près de 2 années pour que ce travail ait donné tous ses fruits. N'espérez donc pas de résultat instantané grâce à un coup de baguette magique ! Qui plus est, rien ne vous garantit que tout ce travail parviendra à placer votre site dans les premières positions pour les mots clés qui vous intéressent.

Pour gagner du temps, la meilleure solution reste de sous-traiter tout ou partie de ce travail considérable. Chaque étape de la réalisation d'un site peut être confiée à des professionnels :

- La conception générale et le design

- La rédaction

- La réalisation technique

- Le référencement "naturel" (amélioration de la position du site dans les classements des résultats de recherche)

- Le référencement payant (affichage publicitaire dans Google ou sur d'autres sites)

- La promotion au sens large (par mail, par exemple)

Ces prestations, à condition qu'elles soient de qualité, peuvent vous faire gagner un temps considérable.

Le référencement naturel (également appelé "référencement organique)" et le référencement payant sont généralement considérés comme complémentaires :

Le **référencement naturel** est un travail de longue haleine dont les résultats sont souvent durables. Il n'est gratuit que si vous y travaillez par vous-même mais demande beaucoup de patience et de travail. Les 2 techniques les plus utilisées dans ce but sont l'inscription dans les annuaires et les échanges de liens. Les visiteurs cliquant sur un résultat de recherche n'ont pas d'a priori sur votre site car ils ont généralement confiance dans la neutralité et l'intégrité du moteur de recherche.

Le **référencement payant** apporte un résultat immédiat mais qui cesse aussitôt que vous interrompez vos investissements. Les visiteurs cliquant sur une publicité peuvent par contre arriver sur votre site avec l'intention d'y acheter un produit et/ou une méfiance sur la qualité de votre contenu. Vos pages devront vraiment mettre le visiteur en confiance pour que votre investissement publicitaire porte ses fruits.

Si on considère que "le temps c'est de l'argent", et compte tenu de l'énorme travail nécessaire pour obtenir un bon classement dans les résultats de recherche, on peut considérer que le référencement "naturel" coûte beaucoup plus cher que le référencement dit "payant".

XIII.3 COMMENT PROPULSER SON SITE EN TETE DES CLASSEMENTS EN 26 ETAPES?

Ces étapes sont détaillées sur le site Rankspirit.

Ce qui suit devrait vous permettre d'obtenir, en moins d'un an, un excellent classement de votre site dans les résultats affichés par Google ou par les autres moteurs de recherche.

A) Préparation

Préparer la construction de votre site en amassant un contenu aussi riche que possible. Bien avant que votre nom de domaine ne soit déposé, réunissez suffisamment de matière pour pouvoir construire une centaine de pages. Ce sera seulement le début. Nous parlons de cent pages de contenu véritable, sans compter les liens, les index, les bibliographies ou autres pages de ressources.

B) Nom de domaine

Votre nom doit être court et facile à mémoriser. Cela est bien plus important que de représenter ou non le contenu de votre site. En clair, votre nom n'est pas un mot clé, c'est une marque commerciale.

C) *Aspect du site*

Le plus simple sera le meilleur. Partez d'une règle simple : chaque page doit contenir d'avantage de texte que de balises HTML. Chaque page doit être lisible par n'importe quel navigateur, du plus ancien au plus récent. Si votre code est plus proche de la version HTML 3.2 que de la version 4.0, cela n'en est que mieux. Minimisez l'utilisation de Flash, de Java ou des Java scripts. Ce qui peut poser problème à un navigateur peut poser problème à un moteur de recherche. Organisez votre site de façon hiérarchique en nommant chaque chapitre selon les règles suivantes :

- le nom doit résumer et représenter parfaitement le contenu du chapitre
- le nom doit être l'un des principaux mots-clés pour lesquels vous souhaitez être facilement trouvé.

Évitez de pourrir votre site avec des chapitres frivoles qui pourraient parasiter l'accès à son véritable contenu. Évitez les compteurs de visites qui ralentissent le chargement des pages. Restez simple et professionnel. Retenez la clé du succès de Google lui-même : la beauté est dans la simplicité. Les internautes aiment les pages simples, claires et faciles à charger.

La rapidité de chargement n'est pas importante : elle est capitale. Votre site doit répondre immédiatement à une requête.

Un délai, même de quelques secondes, représentera une véritable hémorragie d'internautes. Si votre site met 5 secondes pour passer d'une page à une autre lorsque vous y accédez depuis votre pays, il peut en mettre 10 ou 20 lorsque l'accès provient de l'étranger.

Considérez que chaque seconde d'attente vous fait perdre 10% de vos visiteurs. Ces 10%, 20% ou 30% feront peut-être la différence entre le succès et l'échec.

D) *Taille des pages*

Si vous le pouvez, faites en sorte que chaque page pèse entre 5 et 15Ko. Difficile de parvenir à cet objectif mais vous aurez des résultats. Des résultats avec les moteurs de recherche et des résultats avec les internautes. Tout le monde n'a pas l'ADSL, rappelez-vous-en ! Ce n'est pas en sélectionnant vos visiteurs que vous deviendrez populaire.

E) *Contenu*

Chaque jour, construisez une nouvelle page de 200 à 500 mots et mettez-la en ligne. Si vous avez un doute sur le type de contenu qui vous serait utile revoyez la liste des mots clés que vous avez vous-même placés en tête de vos pages. Si vous souhaitez élargir votre horizon, utilisez l'outil proposé par Google AdWords pour trouver de nouveaux mots clés et de nouveaux sujets d'articles.

Évitez les pages de type « blog » et recherchez des articles capables d'intéresser un public large. Surveillez votre orthographe, évitez les abréviations ou le langage « chat ».

Faites des phrases courtes, multipliez les retour-chariot qui allègent la présentation. Chaque page doit avoir un titre. Placez un sous-titre tous les 4 paragraphes. Les internautes liront vos pages rapidement, n'essayez pas de lutter contre ça, essayez plutôt de tenir compte de ça.

En vérité, la majorité des internautes ne liront pas vos pages, ils les survoleront de façon plus ou moins aléatoire tout en gardant un œil sur les autres pages qu'ils chargent en même temps que la vôtre. Ils choisiront finalement de vous accorder un peu plus de temps si :

- votre page se charge rapidement et sans erreur
- au moins l'un de vos titres est plus attractif que ceux des sites concurrents

Certains sites très populaires ont des en-têtes de pub très longs à charger. Ce n'est pas parce qu'ils peuvent se le permettre que vous le pouvez.

Regardez vos pages en quelques secondes comme elles seront vues par des surfers pressés : qu'est-ce qui retient l'œil ? Est-ce intéressant ?

F) *Utilisation des mots clé et pertinence*

Utilisez vos mots-clés une fois dans le titre, une fois dans la balise « description », une fois dans l'en-tête, une fois dans l'URL de la page, une fois en gras, une fois en italique, une fois dans la balise « ALT » d'une photo et une fois dans les premières lignes de votre texte. Faites-les figurer une fois en phonétique. Les moteurs de recherche utilisent de plus en plus la correction automatique avant indexation et un nombre croissant d'internautes saisissent leurs requêtes en phonétique pure.

G) *Liens sortants*

Chacune de vos pages doit comporter un lien vers une page d'un autre domaine.

- Faites figurer dans le texte de votre lien les mots-clés que vous voulez promouvoir.
- Assurez-vous que le domaine cité figure dans les premiers résultats pour ces mots-clés.

H) *Liens internes*

Créez des liens vers les pages les plus riches de votre site. Faites en sorte que tous les paragraphes dont les sujets sont proches soient liés entre eux. Vous voulez 50 pages qui génère chacune une visite par jour. Vous NE voulez PAS avoir une seule page qui vous génère 50 visites par jour tandis que vos autres pages jouent les inconnues.

Chaque page doit avoir deux liens minimum vers d'autres pages de votre site et doit pouvoir être atteinte en deux clics maximum depuis le sommaire.

Le menu présent dans chaque page doit toujours comporter un lien vers le sommaire.

I) *Mise en ligne*

Évitez les noms de domaine virtuels et les serveurs gratuits. Votre domaine doit être autonome pour être correctement indexé par les moteurs de recherche. Un « vrai » site coûte aujourd'hui environ 10 euros/mois. Soyez prêt à investir au moins ce montant.

Mettez votre site en ligne lorsqu'il est prêt, pas avant. Mieux vaut ne rien publier que publier un site en construction.

J) *Référencement*

Enregistrez votre site dans : Google, MSN, Altavista, DMOZ, Teoma et Hotbot en utilisant le Yooda Submit Center. N'utilisez pas les systèmes de référencement automatiques gratuits. Référenciez toujours votre site vous-même et manuellement. Si vous décidez de vous enregistrer dans des annuaires, choisissez toujours des annuaires dont le thème vous concerne réellement. Ne bâclez pas le résumé que

les annuaires vous demandent de leur fournir. Ces annuaires peuvent vous ramener directement plus de 40 visites/jour si vos résumés sont attractifs. Un bon référencement demande 10 jours de travail. Ne vous attendez pas à un résultat extraordinaire si vous n'y consacrez que 2 heures.

Le résultat de ce travail n'est vraiment visible qu'au bout de 2 à 6 mois. Prenez votre mal en patience et donnez le temps au temps. Ne référencez pas votre site plusieurs fois au même endroit, vous pourriez être considéré comme un spammer et être pénalisé.

K) *Suivi de l'audience*

Utilisez un système d'analyse d'audience performant. Vous devez impérativement pouvoir suivre le parcours des internautes à la trace pour découvrir quelles sont les pages qui les attirent. Le temps moyen de lecture de chaque page est également un critère très important pour savoir si vos visiteurs apprécient le contenu de votre site. Votre objectif est d'atteindre un minimum de 4 pages/vues par visite avec une durée moyenne minimum de 4mn par visite. Tant que ces objectifs ne sont pas atteints, vous devez revoir la conception ou le contenu de vos pages.

L) *Parcours des robots*

Vérifiez le parcours des robots sur votre site. Il leur faut quelquefois 3 à 4 visites pour indexer toutes les pages. Si certaines pages n'ont pas été visitées à l'issue de ces visites, vérifiez vos liens internes : il y a sûrement quelque chose qui cloche.

M) *Sommaire*

Chaque mot clé principal doit figurer dans votre sommaire principal. N'hésitez pas à revoir la structure de votre site plusieurs fois si nécessaire.

N) *Échanges de liens*

Utilisez Google pour rechercher tous les sites de qualité qui abordent des sujets proches du votre. Regardez si ces sites ont des pages de liens et quand c'est le cas, proposez un échange de liens. Ne vous prenez pas la tête si l'échange est refusé : le monde est vaste. Proposez chaque jour un échange à un nouveau site, un simple mail suffit. Placez vos liens dans une page réservée à cet usage et vérifiez vos liens chaque fin de mois. Relancez chaque mois les sites qui ne vous ont pas placé de lien-retour et supprimez-les de la liste au bout de 3 mois si ce n'est pas le cas.

O) *Maîtrise du Web*

Si les termes de PageRank (Ce mot désigne le système utilisé par les moteurs de recherche modernes pour déterminer l'intérêt général d'un site et, d'une certaine façon, sa "popularité"), balises meta ... vous semblent obscurs, il vous reste du pain sur la planche. Vous DEVEZ maîtriser ces notions pour réussir votre mise en orbite. Ce site vous permet d'acquérir les connaissances essentielles à votre réussite, prenez le temps de tout lire attentivement. De très nombreux autres sites et forums vous permettront de parfaire vos connaissances ou de poser les questions qui vont bientôt vous hanter.

P) *Éthique*

Ne faites pas de compromis. Ne vendez pas votre âme au diable. Évitez tout ce qui est discutable, non-éthique, malhonnête. Dans le monde du web, il est plus facile de

réussir en restant intègre. Votre site doit devenir populaire, ne l'oubliez pas. Votre public sera votre miroir. Faites en sorte de pouvoir vous regarder dans le miroir !

Q) *Demande de liens*

Quand VOUS recevez une demande d'échange de liens, vérifiez toujours la qualité du site concerné. Regardez comment ce site est classé dans Google et quel est son PageRank. Ne placez pas un lien retour vers un site pourri simplement parce qu'on vous le demande. Vérifiez que son thème est proche du votre.

R) *Maîtrise du marché*

Parcourez les forums, usez vos doigts sur votre clavier, cherchez, cherchez et lisez. Vous devez tout connaître du sujet dont traite votre site. Quand votre opinion ne coïncide pas avec l'opinion générale, affirmez-là avec force et courage, mais assurez-vous tout d'abord qu'il ne s'agit pas d'une ânerie.

S) *Égocentrisme*

Vous pensez avoir des choses à faire découvrir aux autres ? C'est formidable ! N'oubliez pas, toutefois, de vous demander si vos sujets intéressent quelqu'un d'autre que vous. Les internautes ne viendront pas sur votre site dans l'idée de découvrir ce que vous avez à offrir au monde. Ils viendront avec une idée bien précise en tête : la leur ! Demandez-vous ce que c'est ! Ce n'est pas, a priori, vous ou vos produits qui intéressent les gens. N'en parlez pas trop et proposez aux visiteurs des sujets qui les intéressent. Ne faites pas de votre site une brochure publicitaire sans intérêt.

T) *Ajoutez une nouvelle page chaque jour à votre site*

Vous avez raison, cela figure déjà dans le point E. Et bien, relisez le point E. Google adore le contenu. Un énorme contenu de qualité avec une foule de mots clé. A la fin de l'année vous devriez avoir environ 400 pages sur votre site. Cela va vous permettre de figurer dans les résultats pour une grande variété de recherches. De nombreux sites pointeront vers le vôtre en raison de son contenu. Votre site sera solide et consistant.

U) *Étudiez vos "logs"*

30 à 60 jours après la mise en ligne de votre site, étudiez les mots-clés utilisés dans les moteurs de recherche qui ont permis aux internautes de découvrir votre site. Étudiez attentivement ces mots-clés. Y-a-t-il des combinaisons inattendues ? Pourquoi les gens utilisent-ils ces mots pour trouver un site ? Si l'un des mots figurant dans ces combinaisons ne fait pas partie du vocabulaire de votre site, il est temps de créer une page sur ce sujet. Les combinaisons de mots utilisées vous disent exactement ce que les internautes recherchent. Soyez attentif !

V) *Les sujets dans l'air du temps*

Rien n'amène le succès comme le succès. Soyez à l'affût des nouveautés et des modes de votre secteur. Si vous apprenez qu'une nouveauté va sortir en décembre, rédigez une page sur ce sujet en octobre. Elle sera indexée à temps par les moteurs de recherche, si bien que lorsque tous les internautes lanceront leurs recherches en décembre, votre page figurera parmi les résultats.

W) *Les amis et la « famille »*

Le partage et l'entraide sont les clés de votre réussite. Bâissez-vous un réseau de contacts, d'informateurs et de testeurs. C'est là que les heures innombrables que vous avez passées sur les forums vont porter leurs fruits. Les forums vous permettent de sentir l'air du temps, de tester vos idées avant de les publier sur votre site. Grâce aux forums, votre site ne sera pas le seul résultat de vos réflexions et de votre valeur mais représentera les tendances et les idées du moment.

X) *Notez, Notez, Notez*

Si vous rédigez une page par jour, vous allez devoir penser à votre site jour et nuit, sous la douche, en conduisant ou pendant une soirée. Gardez toujours un calepin sur vous et notez chacune de vos idées sous peine de les voir s'envoler. Ne vous contentez pas d'un simple mot : notez au moins 10 mots décrivant l'idée. Ce conseil peut sembler simpliste mais il vous sauvera la vie quand vous serez en panne sèche.

Y) *Vérification du référencement au bout de 6 mois*

Refaites le tour complet des moteurs et sites auprès desquels vous vous êtes fait référencer. Si vous ne figurez pas dans leurs listes, recommencez le référencement et revérifiez tout six mois plus tard.

Z) *Publiez en anglais*

La France, le Québec, l'Afrique francophone, les pays et territoires d'outre-mer sont des endroits magnifiques. Mais ils n'abritent qu'une petite proportion de la population mondiale. A contenu équivalent, un site anglophone aura au minimum 6 fois plus de visiteurs qu'un site francophone. Ce phénomène s'accroît. Avec le développement de la Chine, de l'Inde et de l'ex URSS, la proportion d'internautes francophones diminue de jour en jour. A vous de choisir votre camp !

Si votre site est intéressant, vous atteindrez au minimum 100 visites/jour pour près de 500 pages vues/jour. Multipliez ces chiffres par 6 si votre site est anglophone.

Etablissement référent

Afpa Pont de Claix

Equipe de conception

J.C. Rigal, Jacques Gereys et R.Lecu

(actualisation B.Héazard – Afpa Nice, M. Fayolle Afpa Grenoble)

Reproduction interdite

Article L 122-4 du code de la propriété intellectuelle.
«toute représentation ou reproduction intégrale ou partielle faite sans le
consentement de l'auteur ou de ses ayants droits ou ayants cause est
illicite. Il en est de même pour la traduction, l'adaptation ou la reproduction
par un art ou un procédé quelconques.»

Date de mise à jour janvier 2016
afpa © Date de dépôt légal janvier 2016



**afpa / Direction de l'Ingénierie 13 place du Générale de Gaulle / 93108 Montreuil
Cedex**

**association nationale pour la formation professionnelle des
adultes**

**Ministère des Affaires sociales du Travail et de la
Solidarité**