

Module : Multimédia et développement Web

Chapitre 1 : L'Internet

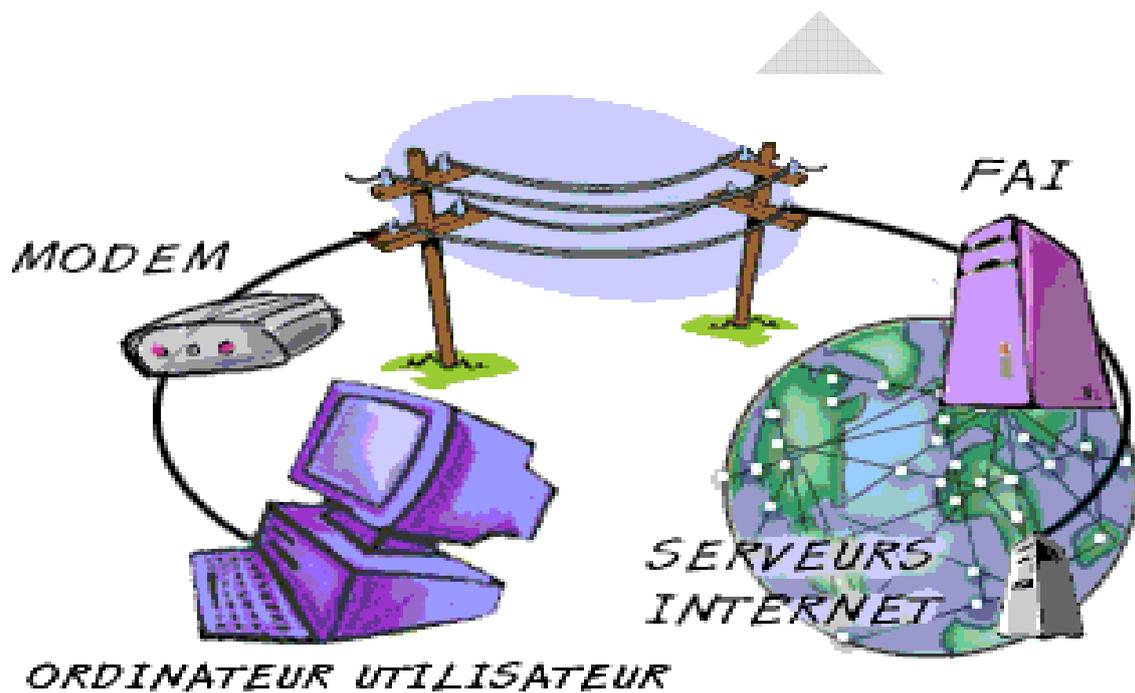
Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadji

Attention !

Ce produit pédagogique numérisé est la propriété exclusive de l'UVT. Il est strictement interdit de la reproduire à des fins commerciales. Seul le téléchargement ou impression pour un usage personnel (1 copie par utilisateur) est permis.

Partie I : L'INTERNET



Après avoir étudié cette partie, l'étudiant sera en mesure de :

- Connaître la définition et l'historique de l'Internet.
- Connaître le matériel nécessaire pour la mise en place d'Internet.
- Identifier les différents services d'Internet.

I.1. Présentation

I.1.1. Qu'est-ce que l'Internet ?

L'Internet est un système de communication qui permet aux ordinateurs autour du monde de communiquer et de s'échanger des informations entre eux. Cette communication entre ordinateurs permet plusieurs possibilités. Ceci permet d'accéder aux ressources de ceux-ci, que ce soit des fichiers, des services ou mêmes d'autres personnes.

I.1.2. L'histoire de l'Internet

C'est dans les années 60's, en plein milieu de la "guerre froide" entre le bloc de l'est et les pays occidentaux qu'on retrouve les origines de l'Internet. Les militaires américains avaient besoin d'un moyen de coordonner leurs armes nucléaires en cas de guerre atomique. Ils avaient besoin d'un système de communication qui pourrait survivre même si plusieurs bases étaient détruites.

- **1964**: Projet ARPANET: Le début du projet pour créer le premier réseau de communication distribué. L'avantage de cette technologie est qu'elle est décentralisée. Le système n'est pas dépendant d'un ordinateur principal pour coordonner les communications.
- **1969**: Mise en opération du projet ARPANET qui utilise un protocole de communication appelé NCP (Network Control Protocol). C'est la première fois que des ordinateurs communiquent entre eux malgré de longues distances. Le réseau survivrait même aux ravages d'une guerre atomique puisque chaque site est connecté physiquement à quatre autres.
- **1973**: Les scientifiques développent leur propre réseau à travers les universités et en utilisant un protocole de communication amélioré nommé TCP/IP (Transfert Control Protocol/ Internet Protocol). Le protocole IP s'assure que chaque ordinateur relié au réseau a une adresse unique et que chacun soit capable de communiquer avec les autres. Ce protocole est utilisé sur plusieurs plates-formes pour permettre une communication entre ceux-ci. L'accès est devenu avec le temps international.
- **1989**: Tim Berners-Lee conçoit la base de ce qui allait devenir le World Wide Web.
- **1995**: C'est cette année que le terme "Internet" est entré dans le vocabulaire de la plupart du monde.

I.1.3. Définitions

- L'Intranet



Fig.1.1. : Un réseau intranet

Il s'agit en fait d'un Internet interne, conçu pour être utilisé partout dans une société, une université ou une organisation. Ce qui distingue un intranet de l'Internet auquel on accède librement, c'est son caractère privé (voir Fig.1.1.).

L'utilisation des outils et des ressources nécessaires pour travailler sur l'Internet mais pour concevoir et pour utiliser pour des besoins internes de l'entreprise. Par exemple, de plus en plus d'entreprises affichent des documents internes en utilisant les mêmes outils que sur l'Internet. Cependant, ces documents sont disponibles seulement à l'intérieur de l'entreprise.

- L'Extranet

C'est l'ancien terme pour décrire ce qu'on appelle aujourd'hui les transactions commerciales d'entreprise à entreprise, business-to-business ou B2B. Ceci consiste à l'achat de fournitures et de matériaux par des transactions sur l'Internet.

- L'adresse IP

Chaque ordinateur connecté à l'Internet à une adresse IP (Internet Protocol). Elle sert à savoir quel ordinateur a fait une requête d'information et de la transmettre au bon endroit

Cette adresse est composée d'une série de quatre chiffres entre 0 et 255 (256 possibilités). Toutes les adresses se retrouvent donc entre 0.0.0.0 et 256.256.256.256 . Cela donne un peu plus que 4,2 milliards adresses. Les adresses sont allouées en

segments de 256 adresses aux fournisseurs d'accès à l'Internet (ISP ou Internet Service Provider en anglais). C'est ceux-ci qui allouent une adresse à ses utilisateurs. Le nombre d'adresses utilisées est en progression. C'est rendu à un point où on envisage une pénurie d'adresses. Une nouvelle version des adresses IP devrait sortir bientôt (IP version 6 ou IPv6) offrant plus d'adresses ainsi que d'autres avantages.

I.2. Mise en place d'Internet

Pour la mise en place d'Internet, il faut disposer :

- d'un ordinateur (PC, Mac, etc.) et son système d'exploitation installé avec les logiciels adaptés
- d'une ligne téléphonique (RTC ou RNIS) ou d'une liaison par câble
- d'une interface (interne ou externe) téléphonique : MODEM ou carte R.N.I.S.
- d'un accès chez un fournisseur d'accès (provider)

I.2.1. Connexion d'un réseau par un routeur électronique

Voici un exemple simple de connexion d'un petit réseau à Internet (voir fig.12.). Les machines, reliées par leur carte réseau et pouvant fonctionner avec des systèmes d'exploitation différents, utilisent ici un **routeur électronique** pour appeler le fournisseur d'accès quand cela est nécessaire.

L'intérêt évident est l'utilisation d'une seule ligne téléphonique pour plusieurs internautes.

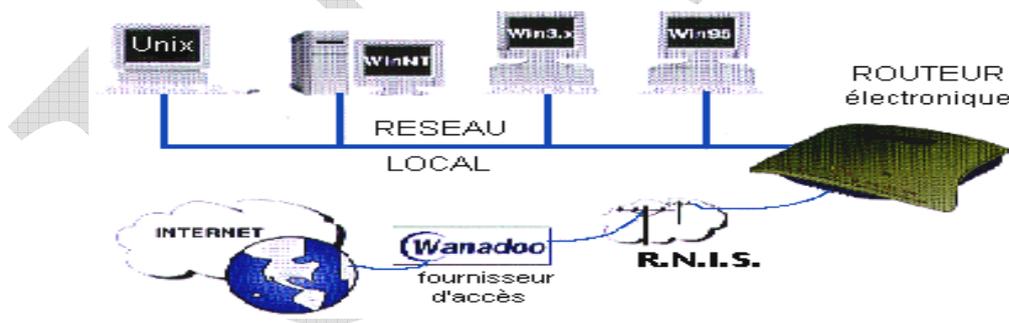


Fig.1.2. : La connexion à l'Internet

I.2.2. Connexion d'un réseau par un serveur PROXY

On peut aussi connecter plusieurs réseaux ou sous-réseaux sur la même ligne téléphonique.

L'utilisation d'un **"switch"** (commutateur électronique) devient nécessaire pour connecter plusieurs **hubs** du même domaine.

Le **serveur d'accès** sur lequel est installé un **logiciel PROXY** (voir Fig. 1.3.) permet de contrôler les accès (dans les deux sens) et offre aux clients une **mémoire-cache** générale (les pages visitées restent sur le disque dur du proxy) ce qui évite bien souvent des connexions inutiles.

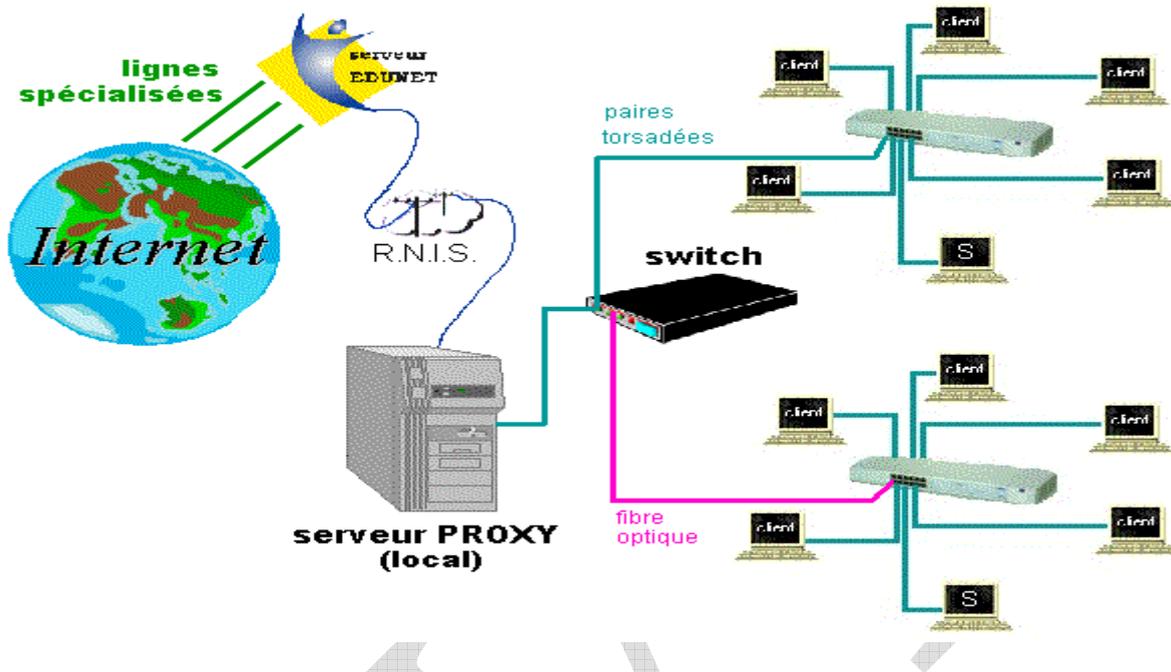


Fig.1.3. : Utilisation d'un serveur PROXY

Pour un nombre encore plus important de clients, même la ligne numérique devient insuffisante. Pour passer au stade supérieur, il faut demander à un prestataire de services l'installation de **lignes spécialisées** (câble, transpac, transfix ou satellitaires). Le réseau local est alors connecté à l'Internet 24/24.

Il est nécessaire pour établir une connexion en ligne que l'ordinateur soit équipé d'un MODEM (MOdulateur DEModulateur). Ce périphérique convertit les signaux numériques de l'ordinateur en signaux analogiques, à même de voyager sur une ligne téléphonique classique, comme indiqué par la figure 1.4.

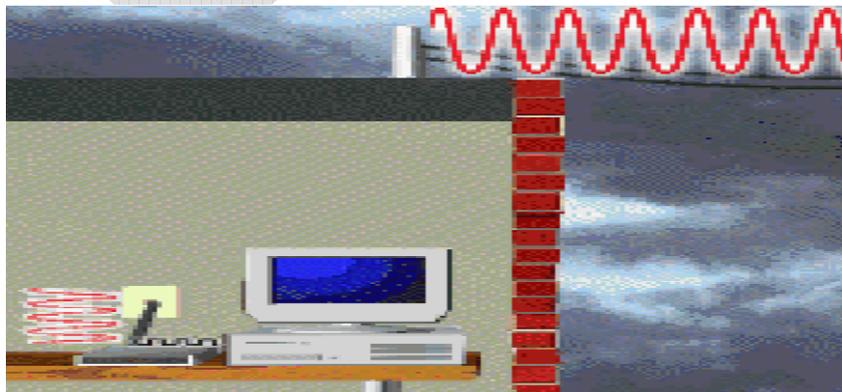


Fig.1.4. : Le MODEM

Les MODEMS sont dotés de vitesses différentes, et leurs performances sont évaluées en bps ou bits par seconde. Il existe des modems de vitesses différentes qu'on peut installer directement dans l'ordinateur (modèle interne) ou connecter au port série de ce dernier (modèle externe). Aujourd'hui, la grande majorité des ordinateurs sont équipés d'un modem interne.

Aussi, un câble de téléphone standard pour connecter le modem à la ligne téléphonique. Le connecteur classique d'un câble de téléphone est de format RJ-11. L'arrière du modem est certainement équipé de deux prises au format RJ-11. L'une sert à connecter le modem à une prise murale et l'autre au téléphone.

I.3. Les services d'Internet



Fig.1.5. : L'Internet

L'Internet (voir Fig.1.5.) offre plusieurs services à savoir :

I.3.1. La navigation WEB



Fig.1.6. : WWW

A l'origine, le World Wide Web (voir Fig.1.6.) a été développé en 1990 au [CERN](#), le Centre Européen pour la Recherche Nucléaire. A présent, il est géré par le [Consortium du World Wide Web](#), aussi connu sous le nom de World Wide Web Initiative.

Un navigateur Web est un programme logiciel utilisé pour accéder au World Wide Web, l'environnement graphique d'Internet. NCSA Mosaic est le premier navigateur à avoir été développé au début des années 90, au National Centre for Supercomputing Applications. De par sa simplicité d'utilisation, l'interface de type pointer et cliquer a contribué à mettre le Web à la portée du grand public.

Parmi les nombreux navigateurs disponibles, **Microsoft Internet Explorer** et **Netscape Navigator** sont les deux plus connus.

A/ Le nom du domaine

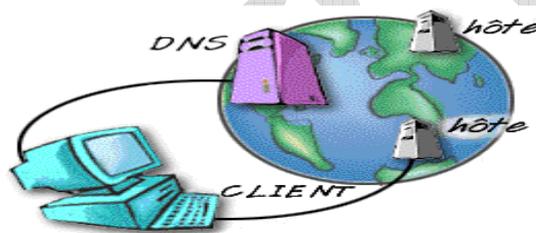


Fig.1.7. : Le système DNS

Chaque ordinateur hébergeant des données sur Internet dispose d'une adresse numérique unique. Cependant, il est difficile de se mémoriser une longue suite de numéros. Le système **DNS** : Domain Name System (voir Fig.1.7.) a été créé en associant une adresse numérique à un nom.

En règle générale, le suffixe du nom de domaine, composé de deux ou trois lettres, s'appelle le domaine de premier niveau. Il décrit le type d'entité concernée. En voici quelques exemples :

.com : pour les activités commerciales

.edu : pour les établissements d'enseignement et les universités

.org : pour les organisations caritatives

.net : pour les fournisseurs d'accès

.gov : pour les organismes gouvernementaux

Un nom de domaine est toujours constitué de deux ou plusieurs parties séparées par des points, et correspond généralement à un nom de société accompagné d'un suffixe de trois lettres.

Outre les noms de domaine de premier niveau, 244 noms de domaine nationaux ont été créés pour les pays et les territoires, dont :

.tn : Tunisie

.ca : Canada

.fr : France

.de : Allemagne

.uk : Royaume-Uni

Il est possible de se représenter le **World Wide Web** comme un réseau de fichiers électroniques enregistrés sur des ordinateurs du monde entier. Les liens **hypertextes** relient ces ressources entre elles. Les **URL** (Uniform Resource Locator) correspondent aux adresses utilisées pour localiser ces fichiers. L'information contenue dans un URL permet de sauter d'une page Web à une autre à l'aide d'un simple clic de souris. Lorsqu'on entre un URL dans le navigateur ou qu'on clique sur un lien hypertexte, le navigateur demande à un ordinateur distant le téléchargement d'un fichier.

Exemple :



La première moitié de l'URL (celle placée avant les deux barres obliques) renseigne sur le type de ressource ou la méthode d'accès de cette adresse. Tels que :

- **http** : un répertoire ou un document hypertexte.
- **ftp** : un fichier téléchargeable ou un répertoire abritant des fichiers de ce type.
- **news** : un groupe de discussion.
- **telnet** : un système informatique auquel on peut s'y connecter via Internet.
- **file** : un fichier situé sur une unité locale (le disque dur par exemple).

La seconde moitié correspond la plupart du temps à l'adresse de l'ordinateur abritant les données ou le service. Des informations supplémentaires peuvent s'y ajouter comme les noms des fichiers, le port de connexion ou le texte à rechercher dans une base de données.

B/ La recherche des informations

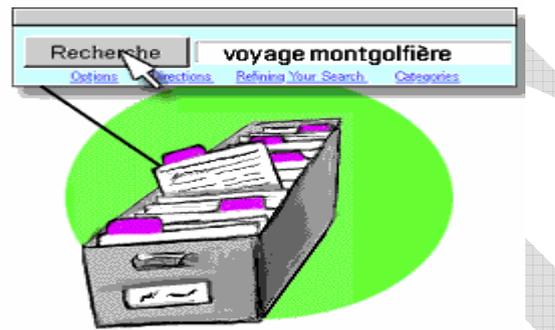


Fig.1.8. : La recherche des informations

Les moteurs de recherche se divisent en deux catégories, les **annuaires** et les **index**. Les annuaires permettent une identification efficace de l'information générale. Comme le catalogue d'une bibliothèque, ils regroupent les sites Web sous des catégories identiques. Au terme de la recherche (voir Fig.1.8.), on parvient à une liste de sites Web relatifs au terme de départ. Mais si on veut une recherche d'informations plus spécifique, les index seront très utiles, car ils parcourent l'intégralité du texte d'un site Web. Les indexes utilisent des logiciels appelés spiders ou robots "collecteurs". Ces derniers explorent Internet, analysant des millions de pages Web et de messages postés dans les groupes de discussion, et indexent chaque mot rencontré.

Exemples :

- **AltaVista** crée des index exhaustifs de chaque mot apparaissant dans les pages Web ou les groupes de discussion Usenet parcourus, et permettent ainsi d'effectuer des recherches parfaitement ciblées. AltaVista effectue une recherche par mot-clé, à partir du texte d'une page Web. C'est ainsi que des millions de pages Web et d'articles en provenance des groupes de discussion Usenet sont indexés. AltaVista actualise en permanence ces informations, et chaque page issue d'une recherche est accompagnée d'une date et d'une heure correspondant à la dernière mise à jour effectuée.

- **Excite** présente une fonctionnalité intéressante, appelée « taux de réussite ». Il s'agit d'un pourcentage accompagnant le résultat d'une recherche. Plus il est élevé, et plus la réponse sera fiable par rapport à la question d'origine. En outre, Excite autorise à étendre la consultation à

d'autres documents, similaires à ceux compris dans le résultat de la recherche. Comme la plupart des annuaires et des index, Excite permet la recherche en plusieurs langues.

- **HotBot** conduit des recherches rapides et performantes dans les sites Web et les groupes de discussion. Les résultats sont triés en fonction de leur degré de pertinence. Il est possible de rechercher des images, des vidéos, des fichiers MP3, et ce dans plusieurs langues.

- **Lycos** est un robot d'indexation des documents rencontrés sur le Web. Ce qui signifie que ses logiciels « collecteurs » explorent quotidiennement le Web à la recherche de nouveaux sites Web, gophers et FTP. Lycos conduit ses recherches via des mots-clés, et attribue à chaque information trouvée un degré de pertinence. Il présente par ailleurs une fonctionnalité intéressante, consistant à localiser des images et des sons sur le Web.

- **Yahoo** est le plus connu des annuaires organisés en catégories, et fournit de ce fait un point de départ intéressant. La recherche s'effectue soit par catégorie, ou comme pour les autres moteurs de recherche, par saisie d'un terme spécifique. Yahoo est très utile pour la recherche d'informations générales, mais son mode d'indexation ne permet pas d'obtenir de très bons résultats pour des données spécifiques ou récentes.

- La méta recherche

Plutôt que de parcourir individuellement chaque annuaire ou index, il est possible de soumettre la requête à plusieurs moteurs de recherche simultanément en conduisant une méta recherche.

I.3.2. Le courrier électronique



Fig.1.9. : Le courrier électronique

Un E-mail (Electronic mail) est un message électronique envoyé d'un ordinateur à un autre (voir Fig.1.9.). On peut envoyer ou recevoir des messages personnels ou professionnels.

Tout comme une lettre traverse différents centres postaux, le courrier électronique passe d'un ordinateur appelé le **serveur de messagerie** à un autre, au fur et à mesure de son transfert via Internet. Une fois parvenu au serveur de messagerie de destination, il est enregistré dans une boîte aux lettres électronique, jusqu'à ce que son destinataire le récupère. L'ensemble du processus peut s'effectuer en quelques secondes permettant ainsi une communication rapide avec des personnes du monde entier à tout moment.

Pour recevoir un courrier électronique, on a besoin d'un compte sur un serveur de messagerie. Celui-ci s'apparente à une adresse à laquelle on reçoit des lettres. Contrairement à une adresse postale classique, ce système présente l'avantage de pouvoir récupérer le courrier à partir de n'importe quel endroit. Une fois connecté au serveur de messagerie, on peut télécharger les messages.

Pour envoyer un message, on a besoin d'une connexion à Internet et d'un accès à un serveur de messagerie qui puisse transférer le courrier. Le protocole standard utilisé pour envoyer des courriers électroniques via Internet est appelé **SMTP**, abréviation de Simple Mail Transfer Protocol. Il fonctionne conjointement avec les serveurs **POP** (Post Office Protocol).

Lorsqu'on envoie un message électronique, l'ordinateur le transfère à un serveur SMTP. Le serveur prend en considération l'adresse électronique (comme l'adresse inscrite sur une enveloppe), puis l'utilise pour transférer le message vers le serveur de messagerie du destinataire. Lorsque le message est reçu par le serveur de messagerie concerné, il est sauvegardé jusqu'à ce que le destinataire le récupère. Il est possible d'envoyer un message électronique à toute personne dans le monde disposant d'une adresse électronique. Il faut noter que la quasi-totalité des fournisseurs d'accès à Internet (ISP) et les principaux services en ligne offrent au moins une adresse électronique pour chaque compte.

I.3.3. Le téléchargement



Fig.1.10. : Le téléchargement

Pour télécharger les fichiers (voir Fig.1.10.), on peut également cliquer à l'aide du bouton droit de la souris sur le lien du fichier et sélectionner **Enregistrer la cible sous** dans le menu en incrustation. Dans certains cas, un message demande d'enregistrer le fichier sur le disque dur ou bien le fichier est téléchargé automatiquement sur le bureau.

Lorsqu'on télécharge des fichiers du Web, différents formats de fichiers électroniques peuvent se présenter. Pour identifier le type de fichier, il suffit de regarder son extension.

De nombreux sites mettent à disposition des fichiers destinés au téléchargement. La plupart de ces fichiers sont soit des fichiers texte, graphiques, audio ou vidéo. Ils peuvent être ou non compressés.

Les fichiers compressés les plus courants portent une extension **.ZIP**, **.SIT** ou **.TAR**. Ces extensions représentent des formats de compression standard. Que ces formats abritent un seul fichier ou en regroupent plusieurs, ils constituent une archive unique. Un fichier archive peut contenir des fichiers vidéo ou graphiques et comporte souvent des logiciels accompagnés de leur documentation.

Remarques :

- La majeure partie de ces logiciels est en shareware, ce qui signifie qu'on peut les télécharger et les essayer gratuitement, mais si on a l'intention de les conserver, il faut les payer.
- Certains fichiers se téléchargent rapidement, d'autres nécessitent plus de temps, selon la taille des fichiers, l'occupation du serveur et la vitesse de la connexion.
- De nouvelles versions des programmes sortent régulièrement. On peut généralement retourner sur le même site pour trouver la version la plus récente. Il faut vérifier les dates de fichier dans le répertoire pour se renseigner sur la version la plus récente.

I.3.4. Internet Relay Chat (IRC)



Fig.1.11. : L'IRC

Depuis l'ordinateur, on peut entrer en communication avec des gens du monde entier qui partagent les mêmes centres d'intérêt dans le but de bavarder : to chat (Fig.1.11.). La discussion est possible à tout moment. On peut même lire les messages sans y répondre.

Plusieurs sites répandus pour la conversation tels que :

- Cykosmos
- ICQ
- IrCQNET
- Hells Geriatrics

I.3.5. Les Newsgroups



Fig.1.12. : Les Newsgroups

Les news sont des forums fédérés par thème, où, pendant une durée de temps donnée, tous les courriers envoyés sont conservés. Ainsi sur un forum qui traite un sujet donné, les questions des uns sont envoyées sous forme de message et quelques heures plus tard les réponses des autres trouvent leur place. Les news sont des réservoirs d'informations vivantes sur un sujet (voir Fig.1.12.).



Module : Multimédia et développement Web

Chapitre 2 : Le Multimédia

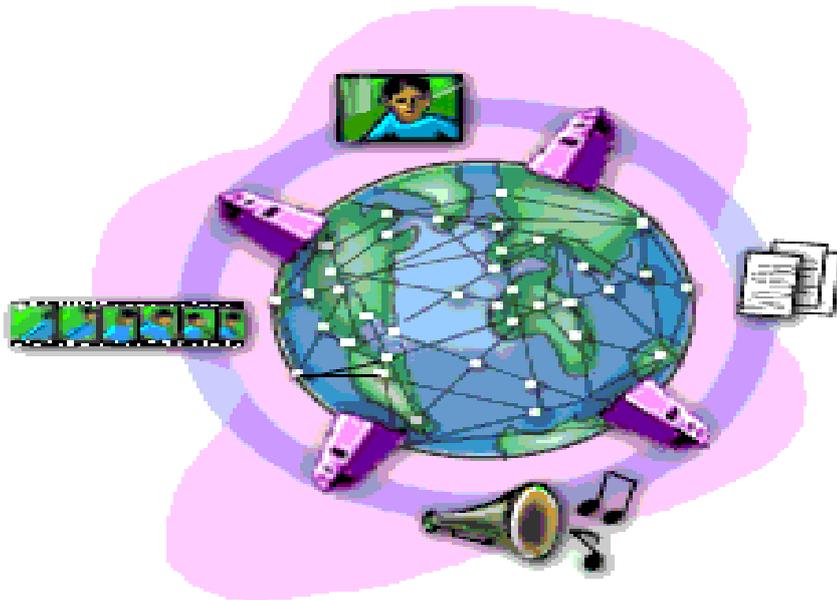
Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Attention !

Ce produit pédagogique numérisé est la propriété exclusive de l'UVT. Il est strictement interdit de la reproduire à des fins commerciales. Seul le téléchargement ou impression pour un usage personnel (1 copie par utilisateur) est permis.

LE MULTIMEDIA



Après avoir étudié cette partie, l'étudiant sera en mesure de :

- Connaître la définition et l'objectif du Multimédia.
- Connaître l'architecture de base d'un ordinateur Multimédia.
- Identifier les formats d'un objet Multimédia.

II.1. Présentation du Multimédia

II.1.1. Historique

Le Multimédia est une technologie qui date depuis les années 80. A cette époque, la vidéo interactive représente le meilleur outil Multimédia. Les grandes avancées technologiques du Multimédia ont eu lieu dans les années 90 avec les projets des autoroutes de l'information.

Certaines avancées technologiques ont contribué au développement du Multimédia :

- Le développement des technologies optiques (CD-ROM, DVD, etc.) pour des volumes importants de données.
- Le développement des méthodes de compression et de compréhension pour les objets Multimédia par exemple pour le :
 - Format texte : .pdf → le logiciel Acrobat Reader/ Writer
 - Format image : .gif, → le logiciel ACDSee
 - Format son : mp3 → le logiciel CoolEdit
 - Format vidéo : .mpeg → le logiciel DIVX
- La croissance de la fréquence du processeur.
- Le développement des technologies de transfert des données volumineuses : les réseaux à haut débit.

II.1.2. Définition et objectif

Le multimédia est *l'intégration sur un même support de données de différents types en vue de leur manipulation (éventuellement interactive) à l'aide de l'outil Informatique.*

Le Multimédia, c'est aussi :

- L'acquisition du son, l'image, la vidéo.
- La numérisation du son, l'image, la vidéo.
- La sauvegarde et la diffusion d'importants volumes de données.
- La communication et l'échange de données.

L'objectif du multimédia est à la fois d'acquérir, de gérer et d'utiliser des valeurs numériques, du texte, des graphiques, des images, du son, de l'image animée et de la vidéo. Ceci implique encore le pilotage d'outils très divers pour gérer des domaines très divers en utilisant des logiciels très différents et spécifiques pour chaque média.

II.1.3. Exemples d'applications Multimédia

Le Multimédia a bénéficié des avancées technologiques déjà citées et il a pu s'intégrer dans plusieurs domaines tels que :

- Systèmes d'édition et de production : Traitements de texte, Montage numérique de vidéo.
- Systèmes de présentation : Logiciels de présentation (ex.: PowerPoint), les bornes interactives (leurs principes est d'utiliser des écrans tactiles au lieu des claviers et des souris), les catalogues (texte, voix, images, vidéo, interactivité, etc.).
- Systèmes d'apprentissage : Cours interactifs, tutoriaux, réalité virtuelle, simulateurs (permis de conduire, aéronautique, jeux vidéo, etc.).
- Communication : télévision interactive, visio-conférence, vidéophonie.

II.2. Ordinateurs Multimédia

II.2.1. Architecture de base d'un ordinateur Multimédia

Un ordinateur multimédia est composé :

- d'une unité d'acquisition de l'information : *périphériques d'entrée* : stylo optique, appareil photo, scanner, webcam, caméra numérique, etc.
- d'une unité de traitement de l'information : le processeur de la carte mère ainsi que les processeurs spécialisés de carte (carte graphique, carte son, carte satellite, etc.).
- d'une unité de stockage de l'information : disque dur, flash disk, lecteur zip, mémoire graphique, etc.
- d'une unité de restitution de l'information : *périphériques de sortie* : datashow, imprimante, écran, etc.
- d'une unité d'échange de l'information : *périphériques de communication* : modem.

II.2.2. La norme MPC

Selon la norme MPC, un ordinateur Multimédia :

- est un ordinateur puissant,
- dispose d'une certaine capacité de stockage,
- peut lire des CD-ROM, des DVD,
- manipule le son et
- avec MPC-3 (inventé en 1995 est une amélioration de MPC), manipule la vidéo.

II.3. Les formats d'un objet Multimédia

II.3.1. Formats du texte et les logiciels correspondants

Il s'agit sans doute de la catégorie de fichiers la moins bien normalisée. Il se dégage tout de même certaines tendances qui permettent aux internautes de s'y retrouver quelque peu.

■.DOC

[Types binaire ou ASCII]. Cette extension de fichier est utilisée pour représenter beaucoup de formats de textes. Le logiciel *Microsoft Word*, en version Windows, utilise l'extension *.doc* pour ses fichiers. Ces derniers sont assez répandus dans Internet.

Utiliser : [Wordviewer](#) (Windows) ; Logiciels de traitement de texte.

■.HTML .HTM .SHTM .SHTML (*Hypertext Markup Language*)

[Type ASCII]. Utilisé spécifiquement pour la rédaction de page Web. Il s'agit d'une application de la norme SGML. Note. Les fichiers se terminant par *.SHTM* et *.SHTML* désignent des fichiers qui contiennent des instructions dynamiques de type SSI (Server Side Include). Ces instructions dynamiques pourraient être l'affichage de la date courante, le nombre de fois que la page a été accédée, ou même une information extraite d'une base de données.

Utiliser : Navigateurs Web.

■.PDF (*Portable Document Format*)

[Type binaire]. Format du logiciel *Acrobat* de Adobe. Très répandu pour la diffusion de documents dans Internet.

Utiliser : [Acrobat Reader](#).

■.PPT .PPS (*PowerPoint Presentation, PowerPoint Slide Show*)

[Type binaire]. Format du logiciel *PowerPoint* de Microsoft (suite *Office*), utilisé pour la production de diapositives électroniques généralement projetées sur un écran lors de présentations. Ce logiciel permet aussi de produire des « séances de projection » (*slide shows*) où les diapositives, qui contiennent souvent des images encapsulées, défilent automatiquement.

Utiliser : [Microsoft PowerPoint Viewer](#).

■.PS .EPS (*PostScript*)

[Type ASCII]. Également développé par Adobe. Langage de typographie et de mise en page. Utilisé avec des imprimantes laser conçues pour interpréter les codes PostScript.

Utiliser : [GostScript](#) ; Envoyer directement à une imprimante laser compatible.

■.RTF (*Rich Text Format*)

[Type ASCII]. Format de documents textuels normalisé favorisant l'échange entre différents logiciels et plateformes.

Utiliser : Logiciels de traitement de texte.

■.SGML .XML (*Standard Generalized Markup Language, Extensible Markup Language*)

[Type ASCII]. SGML est une norme internationale pour la représentation de documents textuels en format électronique. Largement utilisé dans l'industrie de l'édition et les bibliothèques numériques pour la production et l'archivage de documents électroniques (périodiques, monographies, etc.). XML est une version « allégée » de SGML. Consulter le site [The XML Cover Pages](#) pour de plus amples informations

Utiliser : Navigateur Web : en général, les documents XML et SGML qui sont diffusés dans le Web sont convertis en HTML. Les versions récentes des navigateurs peuvent afficher directement les fichiers XML.

■.TEX

[Type ASCII]. Se prononce comme « tek ». Utilisé pour la rédaction de textes scientifiques. Il faut un compilateur (TeX ou LaTeX) pour générer le document formaté. À la sortie du compilateur, un fichier *.dvi* (*Device Independent*) est produit. Ce dernier devra être converti pour l'affichage ou l'impression.

Utiliser : Voir le site de l'American Mathematical Society pour une liste de [sources pour gratuits et partagés de TeX](#)

■.TXT

[Type ASCII]. Par convention, on donne cette extension aux fichiers de texte en simple format ASCII. Documents non formatés.

Utiliser : Navigateurs Web ; Logiciels de traitement de texte ; Éditeurs de texte variés (ex. Bloc-note pour *Windows*, SimpleText pour *Mac*).

II.3.2. Formats du Son

Le son est une sensation provoquée par des vibrations. Il existe 3 types de sons : parole, bruit et la musique.

A/ Caractéristiques du son

- *Hauteur : c'est la fréquence de vibration de l'air. Elle s'exprime en Hertz (Hz) : nombre de vibrations par seconde.*

- *Volume ou intensité sonore : c'est la hauteur de l'amplitude du signal. Elle s'exprime en décibel (db).*

Exemples :

- *70db : conversation normale.*
- *140db : coup de feu.*
- *160db : avion à réaction au décollage.*

- *Timbre : c'est la qualité de la sensation auditive qui différencie deux sons de même hauteur et de même volume.*

B/ Les types de formats

On distingue deux types de formats pour stocker du son :

- *Les formats numériques sont constitués d'une suite d'échantillons et sont stockés sous forme de sons analogiques.*

- *Les formats musicaux sont utilisés par les instruments de musiques.*

C/ La numérisation

La numérisation consiste en la conversion de signaux analogiques continus en données numériques, c'est-à-dire sous forme de valeurs discrètes.

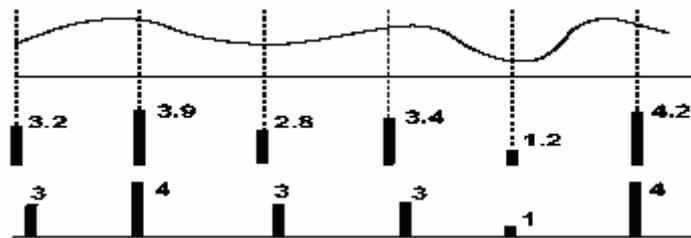
La numérisation d'un signal sonore suit deux étapes :

- **Echantillonnage** : c'est le découpage en "tranches" et permet de Choisir le nombre d'échantillons par seconde.

Exemple : Si l'échantillonnage est égal à 11 KHz alors le son est analysé 11000 fois par seconde.

- **Quantification** : c'est le choix de la précision de représentation d'un échantillon.

Exemple : Si la représentation est sur 1 octet alors l'échantillon est codé sur 256 niveaux.



D/ La qualité de la numérisation

Elle dépend de 3 paramètres :

- Fréquence d'échantillonnage (nombre d'échantillons/seconde) doit être égale à 44,1 KHz pour une qualité CD.

- Le nombre de bits par échantillon doit être égale à 16 bits pour une qualité CD.

- Le nombre de canaux :

- 1 canal : son mono.
- 2 canaux : son stéréophonique.
- 4 canaux : son quadriphonique.

E/ Formats de fichiers audio et les logiciels correspondants

■ **.AIFF .AIF** (*Audio Interchange File Format*)

[Type binaire]. Relativement répandu dans Internet.

Utiliser : Netscape, Internet Explorer. Aussi Windows : [Lecteur multimédia](#). Mac : [lecteur QuickTime](#)

■ **.AU**

[Type binaire]. Développé pour Unix. Très répandu dans Internet.

Utiliser : Windows : [Lecteur multimédia](#). Mac : [lecteur QuickTime](#)

■ **.MID** (*Music Instrument Digital Interface*)

[Type binaire]. Norme pour transmettre des informations musicales entre instruments électroniques et ordinateurs.

Utiliser : Windows : [Lecteur multimédia](#). Mac : [lecteur QuickTime](#)

■.MP3 (*Moving Pictures Expert Group*)

[Type binaire]. Format public pour la numérisation audio en haute-fidélité (qualité comparable au disque compact).

Utiliser : Windows : [Lecteur multimédia](#). Mac : [lecteur QuickTime](#).

■.RA .RAM (*RealAudio*)

[Type binaire]. Format propriétaire pour la diffusion en direct de séquences sonores et vidéo par Internet.

Utiliser : [RealAudio](#)

■.WAV (*Waveform Audio File Format*)

[Type binaire]. Format utilisé par Windows pour stocker des séquences sonores.

Utiliser : Windows : [Lecteur multimédia](#). Mac : [lecteur QuickTime](#)

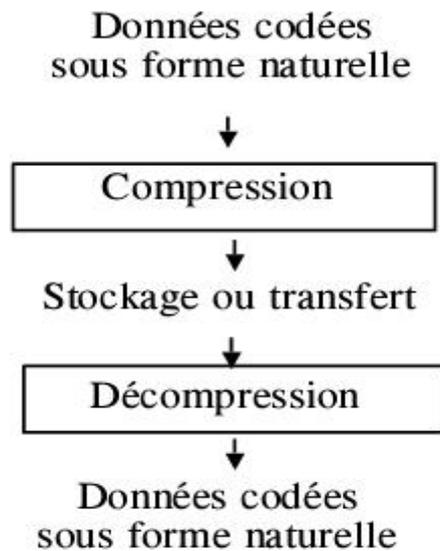
F/ La compression du son

La compression consiste en la réduction de la taille d'un fichier en une taille plus réduite.

Il y a deux types de compression :

- La compression conservative: La plupart des codages "naturels" de l'information sont extrêmement redondants nécessitant moins de bits pour définir la même information.
- La compression non conservative: une partie de l'information sera filtrée soit par les dispositifs de restitution soit par les organes de perception humaines.

Les étapes de la compression sont les suivants :



Le mode de compression utilisé pour un traitement en temps réel de la voix sur un canal à faible débit (GSM, RealAudio...) est basé sur un modèle acoustique de la gorge qui permet de caractériser le son initial et l'effet de résonance (formants).

Les formants sont prédits par un filtre linéaire adaptatif qui induit le formant courant par rapport aux précédents. Une fois le formant calculé il est soustrait du signal initial ce qui donne le résidu qui doit en théorie contenir le son direct. En pratique il faut transmettre une partie du formant ou une référence à un formant voisin (CELP : Coded Exited Linear Prediction).

- Cas de MPEG 3(MP3)

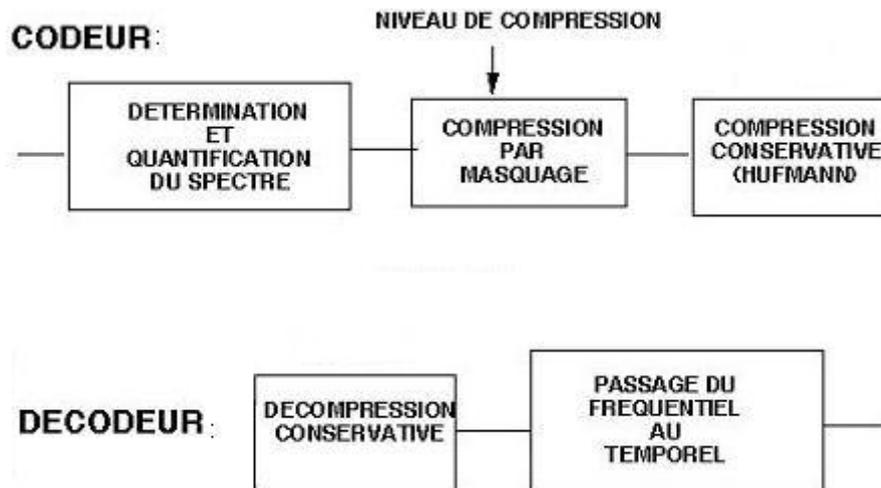
MPEG3 est une norme de transmission de vidéo numérique (image animée+ son) intégrant différentes qualités selon les types d'application (visiophonie, vidéo à la demande, ...).

Le son est échantillonné à 32, 44,1 Khz (fréquence d'échantillonnage d'un CD audio) ou 48 Khz.

Pour obtenir une qualité CD il faudrait une quantification sur 16 bits soit un débit de l'ordre de 700 kb/s par canal.

Pour obtenir un débit plus faible on utilise une compression non conservative.

Le codeur MPEG3 est le suivant :

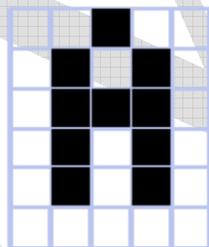


II.3.3. Les images

A/ Définition

Une image numérique est composée d'unités élémentaires appelées pixels. Pixel est une abréviation de *PICTure ELeMent*, c'est-à-dire qu'il s'agit du plus petit élément constitutif d'une image) qui représente chacun une portion de l'image.

L'ensemble des pixels est contenu dans un tableau à deux dimensions :



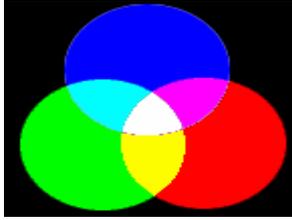
Etant donné que l'écran effectue un balayage de gauche à droite et de bas en haut, on désigne par les coordonnées $[0,0]$ le pixel situé en haut à gauche de l'image, cela signifie que les axes de l'image sont orientés de la façon suivante :

- L'axe X est orienté de gauche à droite.

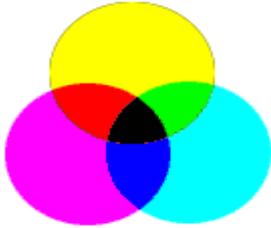
- L'axe Y est orienté de haut en bas, contrairement à ce à quoi on est habitué, car en mathématiques, par convention, l'axe Y est orienté vers le haut.

B/ Les synthèses de couleur

Il existe deux types de synthèse de couleur:



- **La synthèse soustractive** est le fruit de l'ajout de composantes de la lumière. Les composantes de la lumière sont directement ajoutées à l'émission, c'est le cas pour les moniteurs ou les télévisions en couleur. Lorsque l'on ajoute les trois composantes, on obtient du blanc.



- **La synthèse additive** résulte de l'ajout de trois matières de couleurs différentes. Les composantes de la lumière sont donc ajoutées après réflexion sur un objet, c'est le cas pour l'imprimerie. Les trois couleurs primaires sont alors le cyan, le magenta et le jaune, et leur ajout donne du noir. On parle alors de quadrichromie, ou CMYK (Cyan, Magenta, Jaune, Noir pur).

Remarque :

La lumière est issue de deux composantes:

- Une onde électromagnétique ondulatoire.
- Les photons.

Elle a une vitesse de déplacement d'environ $3 \cdot 10^8$ m/s, et une fréquence d'environ $6 \cdot 10^5$ GHz.

C/ Le codage de la couleur

Pour représenter informatiquement une image, il suffit de créer un tableau à deux dimensions dont chaque case contient une valeur. La valeur stockée dans une case est codée sur un certain nombre de bits, le nombre de bits déterminant la couleur ou l'intensité du pixel.

Il existe plusieurs codages possibles:

- **Bitmap noir et blanc** : en stockant un bit dans chaque case, il est possible de définir deux couleurs (noir ou blanc).

- **Bitmap 256 niveaux de gris** : en stockant un octet dans chaque case, il est possible de définir 2^8 intensités de pixels, c'est-à-dire 256 dégradés de gris allant du noir au blanc

- **Palette de couleurs (colormap)** : grâce à cette méthode, on peut définir une palette contenant l'ensemble des couleurs pouvant être contenues dans l'image. A chacune de ces couleurs est associé un indice. Ainsi en codant ces indices sur 8 bits il est possible de définir 256 couleurs utilisables, c'est-à-dire que chaque case du tableau à deux dimensions représentant l'image va contenir un nombre indiquant l'indice de la couleur à utiliser

- **True color**: cette représentation permet de représenter une image en définissant chacune des composantes (RGB: rouge, vert, bleu). Chaque pixel est représenté par un entier comportant les trois composantes, chacune codée sur un octet, c'est-à-dire au total 24 bits (16 millions de couleurs). Il est possible de rajouter une quatrième composante permettant d'ajouter une information de transparence ou de texture, chaque pixel est alors codé sur 32 bits.

D/ La taille d'une image

Pour connaître la taille (en octets) d'une image, il suffit de compter le nombre de pixels que contient l'image. Cela revient à calculer le nombre de cases du tableau, soit la hauteur de celui-ci que multiplie sa largeur. La taille de l'image est alors le nombre d'éléments que multiplie la taille (en octets) de chaque élément. Pour connaître la taille en Ko il suffit de diviser par 1024.

Voici quelques exemples :

Dimensions de l'image	Image en noir et blanc	Image en 256 niveaux de gris	Image en true color
320x200	7.8Ko	62.5Ko	187.5Ko
640x480	37.5Ko	300Ko	900Ko
800x600	58.6Ko	468.7Ko	1.4Mo
1024x768	96Ko	768Ko	2.3Mo

Cela explique la mémoire vidéo que nécessite la carte graphique en fonction de la résolution de l'écran (nombre de points affichés) et du nombre de couleurs. On voit sur l'exemple qu'il faut une carte ayant 4Mo de mémoire vidéo pour pouvoir afficher une résolution de 1024x768 en true color, par exemple.

L'**infographie** regroupe de nombreux savoirs, parmi lesquels la représentation des éléments graphiques (texte, image ou vidéo), ainsi que leurs transformations (rotation, translation, zoom, etc.) par l'intermédiaire d'algorithmes.

E/ Les types d'une image

On distingue deux types d'images :

a) Les images matricielles "bit-map"

Une image numérique est en fait une matrice (un tableau) de pixels.

Avantages :

- Format standard reconnu par les navigateurs.
- Idéal pour des images complexes.

Inconvénients :

- Volume de stockage très élevé.
- Détérioration par agrandissement.
- Perte de qualité par réduction de la résolution de l'affichage.

b) Les images vectorielles

Il existe un autre type d'images numériques, les images vectorielles, dont le principe est de représenter les données de l'image par des formes géométriques qui vont pouvoir être décrites d'un point de vue mathématique. L'usage de ce type d'images concerne les schémas qu'il est possible de générer avec certains logiciels de dessin. On décrit l'image comme l'association de plusieurs objets graphiques simples (droites, courbes, etc.).

Avantages :

- Codage riche prenant en compte la sémantique.
- Codage compact : fichiers de taille réduite.
- Redimensionnement facile, sans perte de qualité.
- Facilité des retouches puisque les différents éléments de l'image sont indépendants.

Inconvénients :

- Inutilisables pour des photographies.
- Pas adaptées aux images complexes avec un grand nombre d'objets de petites tailles.
- Format non standardisé, non reconnu par les navigateurs Internet.

F/ Les formats d'images et les logiciels correspondants

■. Format **BMP** (Windows Bitmap) : est défini par Microsoft pour Windows et possède les caractéristiques suivantes:

- 1, 4, 8 ou 24 bits : jusqu'à 16 millions de couleurs.
- Fichiers de taille importante.
- Reconnu par une majorité de logiciels.

Utiliser : Windows : Paint

■. **GIF** (*Graphics Interchange Format*)

[Type binaire]. Format développé par CompuServe basé sur l'algorithme de compression LZW d'Unisys qui en détient les droits. Fichiers assez compacts mais images limitées à un maximum de 256 couleurs. Utilisé pour produire de petites images peu texturées, des icônes, des boutons dans les pages Web, etc. Permet la transparence et les images animées.

Utiliser : Navigateurs Web

■. **JPEG .JPG** (*Joint Photography Experts Group*)

[Type binaire]. Très fréquent dans Internet. Fichiers très compacts grâce à son algorithme de compression efficace mais qui provoque des pertes d'information. Utilisé pour produire des photos et images texturées dans les pages Web.

Utiliser : Navigateurs Web

■. **PNG** (*Portable Network Graphics*)

[Type binaire]. Appelé à remplacer les formats GIF et JPEG. Norme publique.

Utiliser : Windows : [IrfanView](#). Mac : [SimpleImage](#)

■. **TIFF .TIF** (*Tagged Image File Format*)

[Type binaire]. Utilisé pour l'archivage d'images de qualité. Utilise un algorithme de compression sans perte. Fichiers de relativement grandes dimensions.

Utiliser : Windows : [IrfanView](#). Mac : [SimpleImage](#)

■.Autres formats : PCX (PiCture eXchange), PICT, FAX, TGA, ANI(ANImated cursor).

- DXF (Standard en CAO), PIC (SoftImage PIC), WMF(Windows Metafile Format) : des formats pour les images vectorielles.

Tableau récapitulatif

	Compression des données	Nombre de couleurs supportées	Affichage progressif	Possibilité d'animations	Transparence
JPEG	Oui, réglable (avec perte)	16 millions	Oui	Non	Non
JPEG 2000	Oui, sans perte	16 millions	Oui	Non	Non
GIF	Oui (sans perte)	256 (palette)	Oui	Oui	Oui
PNG	Oui (sans perte)	de 256 (palette) jusqu'à 16 millions (et +)	Oui	Non	Oui (couche alpha)
TIFF	Oui (sans perte) ou non (au choix)	16 millions	Non	Non	Non

G/ Les matériels de numérisation

a) Scanner à main

Ce type de scanner permet de scanner manuellement un document en le passant sur le document en question. Ces scanners peuvent soit scanner une ligne à la fois (texte uniquement) ou une certaine largeur du document (texte et images). Ils sont le plus souvent livrés avec un logiciel de reconnaissance de caractères (OCR : Optical Character Recognition en anglais) afin de pouvoir récupérer le texte scanné dans un traitement de texte. Le majeur inconvénient de ces scanners provient du fait que le scannage se fait manuellement (orientation et vitesse de scannage peuvent varier), ce qui donne parfois des résultats peu satisfaisants.

b) Scanner à plat

Ces scanners permettent de scanner des documents plats de différents types (feuille, revue, livres, photos, etc.). Leur champ d'application va du scannage de textes (avec OCR) au scannage de photos en 16 millions de couleurs (et même plus pour certains).

c) Scanner à diapositives

Ces scanners sont spécifiquement dédiés au scannage des diapos. La dimension de leur fenêtre de scannage est donc de 24 mm sur 36 mm.

d) Appareils photos numériques

Il s'agit d'appareils photos renfermant un système de numérisation interne. Pour la plupart, ils sont les équivalents au point de vue optique de leurs homologues classiques. Les capteurs arrivent maintenant à un nombre de pixels de l'ordre de 3 millions, ce qui permet une bonne qualité d'image.

II.3.4. Les images animées

A/ Définition

Il existe deux types d'images animées :

a) La vidéo

C'est une suite rapide d'images, généralement obtenue à partir d'une source vidéo analogique (caméra, magnétoscope, etc.). Donc c'est une séquence audio synchronisée avec des images.

b) L'animation

C'est une suite d'images de débit moins important que la vidéo. Les images ont généralement été produites grâce à un logiciel de dessin. L'aspect visuel d'une animation se rapproche donc de celui d'un dessin animé. Une animation peut comporter du son.

B/ Les formats de fichiers vidéo et les logiciels correspondants

Il ne faut pas s'attendre à trouver des longs métrages en Dolby stéréo dans Internet (même si, de manière marginale, il est en fait possible d'en trouver). Les réseaux informatiques actuels ne permettent pas des débits de données suffisants. Néanmoins, la vidéo y est bien présente. Il existe deux méthodes de visualisation de séquences vidéo : la méthode en différé et la diffusion en direct. Avec la première méthode, il faut que tout le fichier ait été téléchargé sur son disque dur avant que l'on puisse le visionner. La seconde méthode, au contraire, permet la visualisation pendant le téléchargement. On comprendra que la qualité des séquences en direct sera médiocre. De plus, cette dernière nécessite une connexion très rapide à Internet, sinon les « projections » risquent d'être saccadées. Les séquences vidéo transmises en différé pourront être de meilleure qualité car le temps de téléchargement n'a pas à être synchronisé avec le temps de visualisation. (Selon le débit de la connexion Internet, il faudra peut-être une heure pour télécharger une séquence de quelques minutes.)

Concernant les logiciels de visualisation de vidéos, la plupart de ceux-ci prennent maintenant en charge les formats courants dans le Web. Dans la liste ci-dessous, nous proposons le logiciel « natif », lorsque applicable, mais sachez qu'il n'est pas nécessaire d'avoir chacun de ces logiciels pour faire jouer des vidéos de ces différents formats. En général, un seul suffit.

■ **AVI** (*Audio Video Interleave : Video For Windows*)

[Type binaire]. Format vidéo générique pour Windows. Généralement visualisation en différé. Ce format peut faire appel à différentes techniques de compression de données grâce à l'utilisation de codec (codeur-décodeur) spécifiques à ces méthodes. (De ce fait, un fichier AVI particulier peut ne pas être lisible si le codec auquel il fait appel n'est pas présent sur son ordinateur.)

Utiliser : Windows : [Lecteur multimédia](#). Mac : [lecteur QuickTime](#)

■ **MOV .QT** (*QuickTime Movie*)

[Type binaire]. Développé par Apple. Commun sur Internet. Excellente qualité d'image. Généralement visualisation en différé.

Utiliser : [QuickTime](#)

■ **MPEG .MPG** (*Moving Pictures Expert Group*)

[Type binaire]. Norme publique. Très répandu dans Internet. Généralement visualisation en différé.

Utiliser : Windows : [Lecteur multimédia](#). Mac : [lecteur QuickTime](#)

■.RA (*RealAudio*)

[Type binaire]. Format propriétaire pour la diffusion en direct de séquences sonores et vidéo par Internet.

Utiliser : [RealAudio](#)

■.VIV (*VivoActive*)

[Type binaire]. Format propriétaire pour la diffusion en direct de séquences vidéo par Internet. Qualité d'image médiocre mais fichiers très compacts.

Utiliser : [VivoActive Player](#)

UNIVERSITY

Module : Multimédia et développement Web

Chapitre 3 : Développement Web

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Attention !

Ce produit pédagogique numérisé est la propriété exclusive de l'UVT. Il est strictement interdit de la reproduire à des fins commerciales. Seul le téléchargement ou impression pour un usage personnel (1 copie par utilisateur) est permis.

DEVELOPPEMENT WEB



Après avoir étudié cette partie, l'étudiant sera en mesure de :

- Connaître les terminologies Web.
- Identifier les étapes pour la conception d'un site Web.
- Comprendre l'architecture du Web.
- Connaître les éditeurs de développement des pages Web.
- Assimiler quelques solutions existantes pour le Web dynamique.

III.1. Terminologie

- **Site Web** : Serveur ou hôte sur lequel se trouve le fichier recherché. Il désigne aussi l'ensemble des documents hypertextes.

- **Page Web** : Elle présente une partie d'un site Web. C'est un document électronique écrit dans un langage informatique appelé **HTML** (Hypertext Markup Language). Chaque page Web dispose d'une adresse unique, appelée **URL** ou Uniform Resource Locator, permettant d'identifier son emplacement sur le réseau.

- **Hypertexte** : Ensemble de textes et d'autres documents qui peuvent être consultés à partir d'un système d'envoi hiérarchisé. La sélection d'un renvoi ou hyperlien (en cliquant avec la souris) permet d'afficher un autre texte, une table, une image, etc.

III.2. Intérêt d'un site WEB

La mise en place d'un site web peut être motivée par plusieurs raisons :

- Le **besoin de visibilité** : un site web, dans la mesure où il fait l'objet d'une bonne campagne de promotion, peut être un moyen pour une enseigne ou une organisation d'augmenter sa visibilité ;
- L'**amélioration de la notoriété** : grâce à un site web institutionnel ou un mini site web événementiel, une enseigne peut développer sa popularité auprès du public ;
- La **collecte de données** : Internet représente pour les entreprises une formidable opportunité de recueillir des données sur leurs clients ou bien de démarcher de nouveaux prospects ;
- La **vente en ligne** : Un site Internet peut représenter pour certaines entreprises une opportunité en terme de commercialisation ;
- La **mise en place d'un support aux utilisateurs** : de plus en plus de sociétés utilisent Internet comme support privilégié pour le service avant-vente ou après-vente. En effet, avec un site web, il est possible de mettre à disposition des internautes un maximum d'informations commerciales ou techniques, à moindre coût.

On distingue habituellement plusieurs catégories de sites web, selon le but poursuivi :

- Les **sites vitrine** (appelés également *sites plaquette* ou *site identité*) sont des sites dont l'objectif est de mettre en avant l'image de marque de la société, en présentant par exemple ses produits ou ses services ;
- Les **sites catalogue** sont des sites visant à présenter l'offre de l'entreprise ;

- Les **sites d'information** sont des sites fournissant une information particulière à un type d'internaute ;
- Les **sites marchands** sont des sites vendant directement des produits aux internautes et permettant éventuellement de payer en ligne ;
- Les **sites institutionnels** sont des sites présentant l'organisation et ses valeurs. Ce type de site décrit généralement l'activité de l'organisation, des chiffres clés et donne les informations nécessaires aux clients ou aux bénéficiaires ;
- Les **sites personnels** (parfois *pages perso*) sont des sites réalisés par des particuliers à titre de loisir, le plus souvent par passion pour un sujet ou une discipline ;
- Les **sites web portail** sont des sites Internet ou Intranet qui offrent une porte d'entrée unique sur un large panel de ressources et de services (messagerie électronique, forum de discussion, espace de publication, moteur de recherche), centrés sur un domaine ou un comité publicitaire. Ce sont des sites Web destinés à accueillir les internautes au début d'un de leurs surfs.
- Les **sites communautaires** sont des sites réunissant des internautes autour d'un intérêt commun.
- Les **sites intranet** sont des sites accessibles de l'intérieur d'une entreprise ou d'une direction, ayant pour objet la mise à disposition et le partage d'informations professionnelles.

III.3. La construction d'un site WEB

III.3.1. Comment concevoir un site WEB

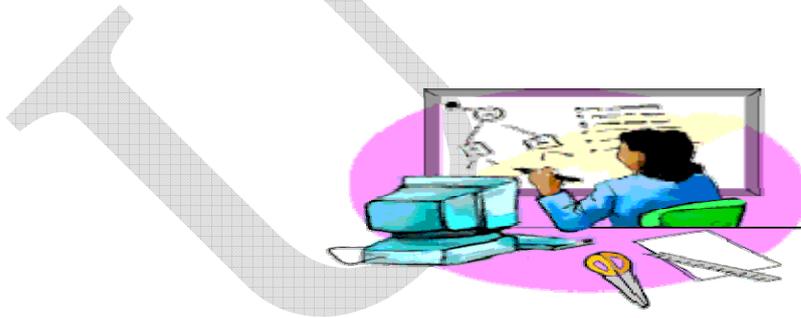


Fig.3.1. : L'étape de la conception d'un site WEB

Avant de concevoir un site Web (voir Fig.3.1.), on doit se poser cette question clé : à qui le site va-t-il s'adresser ? Est-ce que le site sera accédé par autant de visiteurs que possible ou sera-t-il destiné à un groupe en particulier ?

Les pages Web doivent contenir des fichiers graphiques de petites tailles et ne pas faire appel aux fonctions compliquées. Aussi, on doit bien tenir compte du matériel des utilisateurs. Pour le moment, un utilisateur moyen accède au Web à la vitesse de son modem et le site doit donc tenir compte de cette contrainte. L'idéal serait que la taille des fichiers soit suffisamment petite pour qu'ils soient téléchargés en moindre temps.

En parcourant le World Wide Web, on rencontrera bien souvent le terme **page d'accueil**. On doit considérer la page d'accueil comme le point de départ d'un site Web. Tout comme la table des matières d'un livre ou d'un magazine, la page d'accueil présente le contenu du site. Selon sa conception, un site se compose d'une ou plusieurs pages, ou de quelques pages longues. S'il ne contient que peu d'informations, la page d'accueil peut être la seule disponible.

Le design et le contenu des pages Web sont d'une très grande variété, mais nombreuses sont celles qui utilisent un format de magazine classique. Ainsi, le haut de la page comprend la **bannière**, puis une liste d'éléments tels que des articles, souvent accompagnés d'une brève description. Ces éléments sont généralement reliés à d'autres pages du site Web ou même à d'autres sites. Parfois, ces liens s'affichent dans le corps du texte sous forme de mots surlignés ou de liste. Une page Web abrite également des images liées à d'autres informations de contenu.

Les liens ont une couleur différente de celle du reste du texte, et s'affichent la plupart du temps en caractères bleus et soulignés. Lorsqu'on passe avec le curseur au-dessus d'un lien de type texte ou image, l'icône en forme de flèche se transforme en main. Le texte de l'hyperlien renseigne souvent sur l'information à laquelle il conduit.

Lorsqu'on revient à une page abritant un lien qu'on a déjà emprunté, le texte de ce dernier est souvent de couleur différente. De cette manière, on sait qu'on est déjà passé par là. Cela n'empêche pas de le réutiliser.

La conception d'un site web doit être une démarche participative, orientée utilisateurs, c'est-à-dire qu'elle doit se faire, autant que possible, en concertation avec un ensemble de représentants de l'entreprise ou de l'organisation, et de ses bénéficiaires, en vue d'obtenir une satisfaction maximale des utilisateurs.

L'ensemble de la conception doit ainsi principalement être centré sur l'utilisateur et réalisé en concertation étroite avec un panel représentatif d'utilisateurs, par exemple lors de réunion de « brainstorming » (remue-méninges).

L'objet principal de la phase de conception est d'analyser l'ensemble des besoins, puis d'imaginer des contextes d'utilisation. L'analyse des besoins comporte ainsi deux volets :

- Les besoins de l'organisation, correspondant à **une stratégie d'objectifs** :

Avant de commencer tout projet de site web, il est nécessaire de récapituler les objectifs du site web et, dans le cas d'une société ou d'une organisation, en quoi le site web va permettre à l'organisation de remplir ses propres objectifs stratégiques.

Il est conseillé de recenser l'ensemble des objectifs du site web et d'y associer des critères opérationnels, quantifiables, qui permettront de mesurer si l'objectif visé a bien été atteint.

- Les besoins des utilisateurs, correspondant à **une stratégie de cible** :

Un site web est généralement prévu pour une catégorie d'utilisateurs. Un ciblage précis du type d'utilisateurs concerné permettra d'établir un profil type et ainsi de focaliser le contenu sur tel ou tel type d'information.

Il est ainsi nécessaire de s'interroger sur le type d'information que les utilisateurs cibles sont susceptibles de chercher, d'essayer de déterminer leurs principales questions et le type de réponse attendue.

Cette étape peut être faite par le biais d'interview des clients de l'entreprise ou des bénéficiaires de l'organisation, ou, dans le cas d'un site personnel, d'entretiens avec les personnes de l'entourage.

III.3.2. Les étapes de construction d'un site Web

Le développement d'un site Web comprend les étapes suivantes :

- La planification

- En premier lieu, on doit déterminer à qui s'adresse le site. Ceci est un point important, du fait que de nombreuses décisions relatives à la conception et au contenu en dépendent.
- Le site doit être bien organisé pour le rendre plus facile à gérer.
- Il faut tracer les grandes lignes du site sous forme schématique.
- Il faut examiner de près les différents sites Web pour avoir des idées.

- Le développement du contenu

Voici quelques principes généraux à prendre en considération lors de la phase de développement :

- La page d'accueil doit attirer les visiteurs sans les submerger d'informations. Il faut veiller à une présentation simple et à un texte court. En plus, il faut ajouter des illustrations pour transmettre l'image.
- La reproduction de brochures ou autres documents existants ne fonctionne généralement pas bien sur le Web. Les gens ont une attention de courte

durée et n'aiment pas particulièrement lire un texte trop long sur un écran d'ordinateur. Il faut utiliser uniquement les informations les plus indispensables. En plus, il est préférable d'ajouter des titres et du texte en gras, si nécessaire, pour rendre la lecture plus aisée.

- Si on doit inclure des documents longs, il faut les diviser en sections avec des titres. Après on Crée un index ou une table des matières en haut de la page, avec des liens à chaque section. Si on souhaite fournir un accès à de nombreux documents, on doit ajouter une fonction de recherche.
- Il est préférable d'utiliser des graphiques pour améliorer le site, mais on doit minimiser leur taille autant que possible.

- La conception graphique

On peut enrichir le site à travers quelques bons graphiques placés sur la page d'accueil. C'est mieux en répétant certains de ces éléments à travers le site pour créer une impression de continuité.

Les exigences relatives à la taille des fichiers, au choix des couleurs et de la résolution d'écran sont à prendre en considération.

- La programmation

Une fois le site planifié, son contenu et les graphiques créés, on doit convertir les informations dans une forme lisible sur le Web. Pour ce faire, on peut convertir les fichiers textes au format HTML et les graphiques au format GIF ou JPEG.

Après on doit chercher un endroit pour héberger le site. La plupart des gens font héberger leurs sites par un fournisseur de services Internet ou par une société d'hébergement sur le Web.

- L'hébergement

Une société mettant à disposition un serveur web connecté en permanence à Internet est appelée **hébergeur** et offre un service appelé hébergement. On distingue deux principales catégories d'hébergeurs :

- **les hébergeurs gratuits.** Ils prêtent gratuitement un espace disque sur un serveur pour créer les pages web. Ce type de service est généralement totalement gratuit. Ils gagnent de l'argent soit avec un espace publicitaire sur le site, soit uniquement grâce au trafic sur leur propre site.
- **les hébergeurs professionnels.** Ce type d'hébergement garantit un service de qualité (bande passante) et de sécurité (sécurité des données et assurance d'un nombre de pannes réduit). Il s'avère assez indispensable dans le cas d'un site à trafic important et permet l'achat d'un nom de domaine

- La stratégie de sécurité du site

La sécurité est un aspect essentiel des sites Web. La sécurité implique plusieurs disciplines et déterminera souvent si un site Web est réussi. Un site Web sécurisé identifiera correctement les utilisateurs via l'authentification, permettra de s'assurer que ces utilisateurs peuvent accéder uniquement à ce qu'ils demandent via l'autorisation et de protéger les données confidentielles et sensibles via la communication chiffrée et le stockage, et suivra les méthodes conseillées pour l'écriture de code afin d'éviter des problèmes de sécurité.

Par définition, les applications Web permettent aux utilisateurs d'accéder à une ressource centrale, le serveur Web et, via celui-ci, à d'autres ressources, telles que les serveurs de bases de données. En maîtrisant et en implémentant les mesures de sécurité appropriées, il faut :

- protéger les ressources contre un accès non autorisé ;
- limiter les niveaux d'accès par utilisateur ou rôle ;
- établir l'intégrité et la confidentialité des données, ce qui offre un environnement relativement sécurisé dans lequel les utilisateurs peuvent employer aisément le site ;
- contrôler les moyens d'accès du site aux ressources limitées ;
- vérifier que le code de l'application s'exécute comme prévu.

- La mise en ligne

Pour créer un site Internet, il n'est pas nécessaire d'être connecté à Internet, compte tenu du fait qu'il s'agit uniquement dans un premier temps de créer les fichiers HTML et les images.

Toutefois, lorsque le site web est prêt à être vu par des visiteurs, il est essentiel de le mettre en ligne, c'est-à-dire de transférer (copier) les fichiers sur le serveur.

Pour ce faire il est alors essentiel de se connecter à Internet, puis d'envoyer les pages sur le serveur à l'aide d'un client FTP (un logiciel de transfert de fichiers).

- La maintenance

Tout comme une maison, un site Web nécessite un entretien régulier. Il ne faut pas commettre l'erreur de penser qu'une fois en ligne, c'est terminé. On doit ajouter un nouveau contenu et mettre à jour l'existant.

III.4. Notion d'ergonomie

L'**ergonomie** est l'utilisation de connaissances scientifiques relatives à l'homme (psychologie, physiologie, médecine) dans le but d'améliorer son environnement de travail. L'ergonomie se caractérise généralement selon deux composantes :

- L'**efficacité**, consistant à adopter des solutions appropriées d'utilisation d'un produit, au-delà du bon sens du concepteur ;
- L'**utilisabilité**, marquant l'adéquation aux capacités de l'utilisateur. Elle se décline en le **confort d'utilisation**, consistant à réduire au maximum la fatigue physique et nerveuse. La **sécurité**, consistant à choisir des solutions adéquates pour protéger l'utilisateur ;

Appliquée au domaine du web, l'ergonomie d'un site web peut être définie par sa capacité à répondre efficacement aux attentes des utilisateurs et à leur fournir un confort de navigation.

La principale difficulté que tente de lever l'ergonomie est la diversité des profils des visiteurs. Les critères suivants sont généralement déterminants pour un site web :

- **attentes** de l'utilisateur : tous les visiteurs du site ne recherchent pas nécessairement la même information ou n'ont pas nécessairement les mêmes exigences en terme de graphisme.
- **habitudes** de l'utilisateur : elles correspondent à des comportements acquis.
- **âge** de l'utilisateur : il caractérise en général la capacité d'adaptation de l'utilisateur et sa rapidité de navigation.
- **équipements** : il s'agit d'une des difficultés majeures. En effet, l'affichage du site pourra varier d'un équipement à l'autre, en particulier selon le navigateur et la résolution d'affichage.
- **niveau de connaissances** : tous les visiteurs du site ne sont pas nécessairement des de informaticiens. L'ergonomie du site doit être pensée pour l'utilisateur le moins expérimenté.

Les principaux critères d'ergonomie d'un site web sont les suivants (liste non exhaustive) :

Critère	Sous-critère	Description
Sobriété	Simplicité	un site web simple renforcera la crédibilité de l'organisation.
	Peu chargé	Les images animées sont déconseillées. Les animations doivent être privilégiées pour afficher des messages forts car elles attirent le regard de l'internaute.
Lisibilité	Clarté	Il convient de prendre en compte le fait que les informations écrites sont moins faciles à lire sur un écran que sur papier (25% de temps de lecture supplémentaire).

		Ainsi, le texte devra être suffisamment aéré.
	Structuration	Le texte devra être structuré à l'aide de paragraphes et de titres de différents niveaux, afin d'en faciliter la lecture
	Organisation	Les éléments d'informations doivent être hiérarchisés par niveau d'importance. Les éléments les plus importants doivent figurer en haut de page.
Utilisabilité	Facilité de navigation	La « règle des 3 clics », globalement suivie, stipule que toute information doit être accessible en moins de 3 clics
	Repérage	A tout moment le visiteur doit pouvoir être en mesure de se repérer dans le site. Qui plus est, le logo doit être présent sur toutes les pages, au même emplacement, et une charte graphique uniforme doit être appliquée à l'ensemble des pages afin de permettre à l'utilisateur de savoir qu'il est toujours sur le même site. Un plan du site peut constituer un excellent moyen pour permettre au visiteur de savoir où il est.
	Liberté de navigation	Le site doit laisser la possibilité à l'utilisateur de revenir à la page d'accueil et aux principales rubriques par un simple clic, quelque soit la page sur laquelle il se trouve.
	Visibilité de l'adresse	L'URL de la page en cours doit être constamment visible et suffisamment explicite pour permettre à l'utilisateur de se repérer afin de pouvoir revenir facilement.
	Tangibilité de l'information	Les informations présentes sur le site doivent être qualifiées, c'est-à-dire que des éléments permettant d'en connaître la date de mise à jour ou l'auteur qui doivent être présents
	Homogénéité de la structure	Les éléments de navigation doivent être situés au même endroit sur toutes les pages, si possible avec une présentation uniforme d'une page à une autre.
Rapidité	Temps de chargement	Le temps d'affichage d'une page doit être le plus petit possible car la majorité des internautes ne patientent pas plus de 15 secondes lors du chargement d'une page. Ce facteur dépend notamment de la connexion du visiteur, de la taille de la page et de ses images, ainsi que des capacités du serveur web.
	Images optimisées	Il convient d'optimiser au maximum la taille des images, en choisissant un format adapté et un nombre de couleurs

		le plus petit possible.
	Images non redimensionnées	Pour la même raison, les images doivent préférentiellement être dans leur taille réelle, sans redimensionnement.
Interactivité	Liens hypertextes	L'interactivité caractérise les interactions possibles entre l'utilisateur et le site web. Les liens hypertextes offrent de vastes possibilités dans ce domaine et permettent d'offrir au visiteur des parcours multiples, au gré de ses envies. Il est conseillé de mettre suffisamment de liens entre les différentes pages. A l'inverse, trop de liens peuvent rendre difficile la lecture.
	Découpage de l'information	Afin de permettre au visiteur de mieux assimiler l'information et, dans certains cas de susciter sa curiosité, il est préconisé de découper l'information. Le découpage de l'information peut notamment se traduire par un texte d'accroche et un lien hypertexte conduisant à la suite de l'article.
	Facilitation des échanges	Il est souhaitable de tout mettre en place pour faciliter l'échange avec les visiteurs, notamment afin de recueillir leurs impressions et leurs demandes, en vue de faire évoluer le site. A minima, le visiteur doit pouvoir trouver facilement le moyen de contacter une personne, par courrier électronique ou grâce à un formulaire de contact.
Adaptabilité	Adaptabilité	L'adaptabilité caractérise la possibilité de personnalisation du site web sur intervention de l'utilisateur.
	Redimensionnement des polices	Les textes du site doivent préférentiellement utiliser des polices dont la taille n'est pas exprimée en valeur absolue, afin de laisser la possibilité de les redimensionner aux personnes le souhaitant.
Accessibilité	Universalité de l'accès	L'accessibilité est la capacité du site web à être consulté universellement, c'est-à-dire par tout type d'utilisateurs, y compris les malvoyants et non voyants. Il existe un certain nombre de règles d'accessibilité simples, à respecter afin de permettre l'accès au site web au plus grand nombre, quelque soit leur configuration logicielle et matérielle ou leur handicap.
	Interopérabilité	Le respect des standards, permet de garantir un bon niveau d'interopérabilité, c'est-à-dire la capacité du site

	web à être consultable par différents clients logiciels.
Transparence des formats	Les formats utilisés doivent préférentiellement être transparents, c'est-à-dire consultable en mode texte. Ainsi le HTML sera utilisé préférentiellement au Flash. Par ailleurs, les images ou animation ne doivent pas se soustraire aux informations textuelles ou provoquer une gêne pour les non-voyants. Les illustrations graphiques doivent donc uniquement être utilisées en tant que complément visuel au texte.
Légende	Une légende ou un texte de remplacement doivent être prévu en lieu et place des images, afin de permettre aux déficients visuels de comprendre le sens de l'image.
Choix des couleurs	Les couleurs doivent être choisies afin de laisser l'information lisible aux personnes ne distinguant pas les couleurs correctement.
Usage sain des feuilles de style	L'information doit pouvoir être accessible, y compris sans feuille de style.
Contraste adapté	Le niveau de contraste entre la couleur d'arrière-plan et le texte en avant plan doit être suffisant pour en permettre la lecture aux malvoyants.
Taille des polices modifiable	La taille des polices doit être adaptable afin de laisser la possibilité de grossir les textes si nécessaire. Les polices utilisées ne doivent également pas être trop petites, au risque de fatiguer la vue des lecteurs ou d'être illisibles.

III.5. Architecture du Web

La technologie du Web est basée sur l'architecture Client/Serveur de présentation.

On distingue deux types d'architecture du Web :

III.5.1. Web statique

Dans les applications Web statique le serveur Web et le client sont tous les deux « passifs » : le serveur envoie des fichiers HTML qui fournissent du texte mis en forme au client , comme présenté par la figure 3.2.

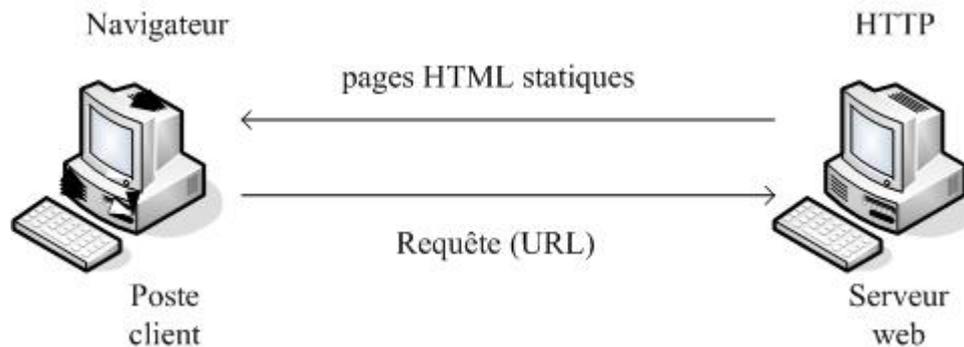


Fig.3.2. : Dialogue entre le serveur et le client

Le principe de fonctionnement est très simple : l'utilisateur envoie une requête de type URL au serveur Web via le protocole HTTP. Le serveur reçoit la demande et renvoie le fichier HTML à l'explorateur du client qui le lit et l'affiche.

Dans une situation de ce type, l'interaction entre l'utilisateur et le serveur Web est limitée. De plus, la mise à jour des fichiers contenant les informations doit être réalisée manuellement. Les informations risquent donc de ne pas être mises à jour en permanence.

III.5.2. Web dynamique

A/ Présentation des différentes architectures Web

a- Présentation de l'architecture d'un système client/serveur

De nombreuses applications fonctionnent selon un environnement client/serveur, cela signifie que des **machines clientes** (des machines faisant partie du réseau) contactent un **serveur**, une machine généralement très puissante en terme de capacités d'entrée-sortie, qui leur fournit des **services**.

Dans un environnement purement Client/serveur, les ordinateurs du réseau (les clients) ne peuvent voir que le serveur, c'est un des principaux atouts de ce modèle.

Un système client/serveur fonctionne selon le schéma suivant :

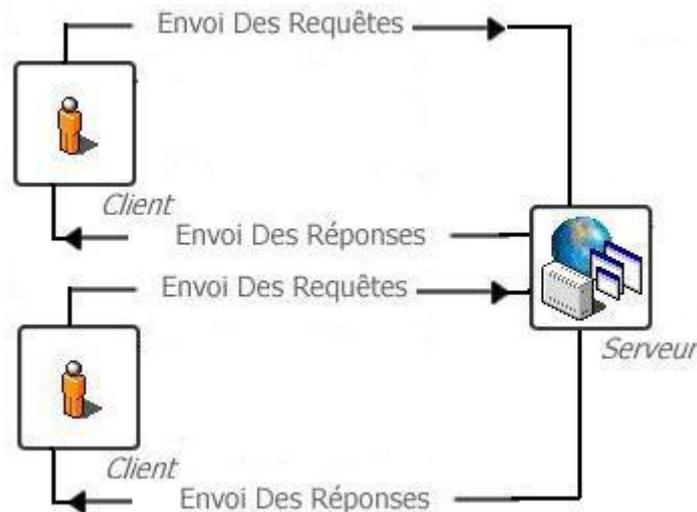


Fig. 3.3. : Modèle client/serveur

- Le client émet une requête vers le serveur.
- Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine client et son port.

b- Présentation de l'architecture à 2 niveaux

L'architecture à deux niveaux (aussi appelée *architecture 2-tier*, *tier* signifiant *étage* en anglais) caractérise les systèmes clients/serveurs dans lesquels le client demande une ressource et le serveur la lui fournit directement (voir Fig. 3.4.). Cela signifie que le serveur ne fait pas appel à une autre application afin de fournir le service.

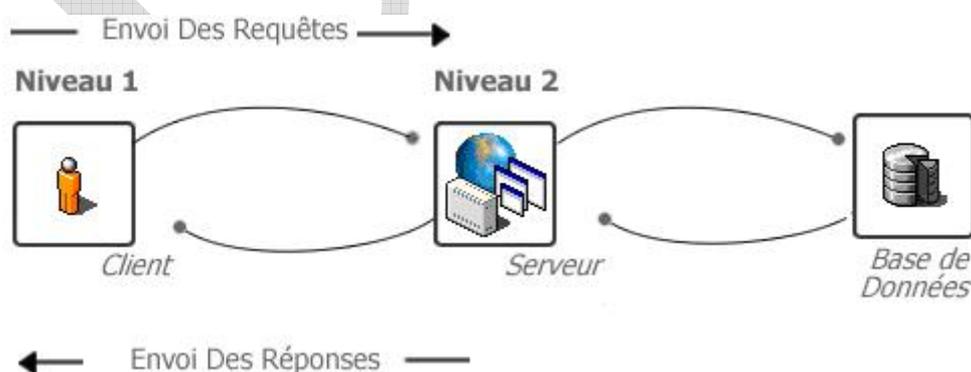


Fig. 3.4. : Architecture à 2 niveaux

c- Présentation de l'architecture à 3 niveaux

Dans l'architecture à 3 niveaux (appelées *architecture 3-tier*), il existe un niveau intermédiaire (voir Fig.3.5.), c'est-à-dire que l'on a généralement une architecture partagée entre :

1. Le client: le demandeur de ressources.
2. Le serveur d'application (appelé aussi **middleware**): le serveur chargé de fournir la ressource mais faisant appel à un autre serveur.
3. Le serveur secondaire (généralement un serveur de base de données), fournissant un service au premier serveur.

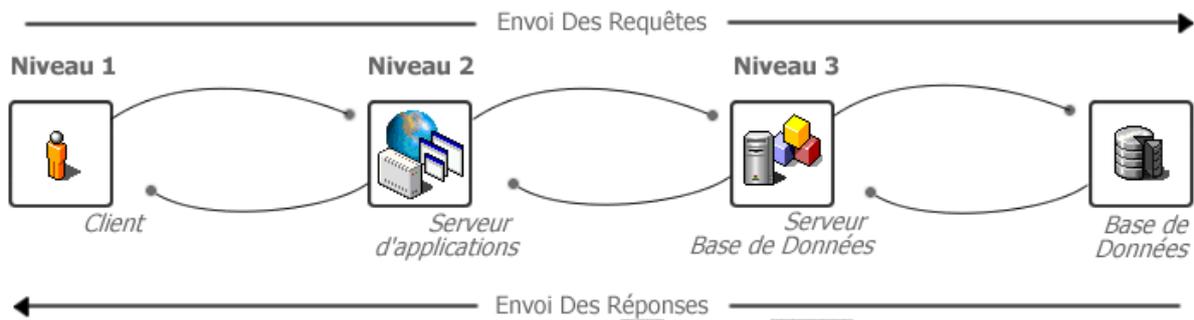


Fig. 3.5. : Architecture à 3 niveaux

d- L'architecture multi-niveaux

Dans l'architecture à 3 niveaux, chaque serveur (niveaux 1 et 2) effectue une tâche (un service) spécialisée. Ainsi, un serveur peut utiliser les services d'un ou plusieurs autres serveurs afin de fournir son propre service. Par conséquent, l'architecture à trois niveaux est potentiellement une architecture à N niveaux comme l'indique la figure suivante :

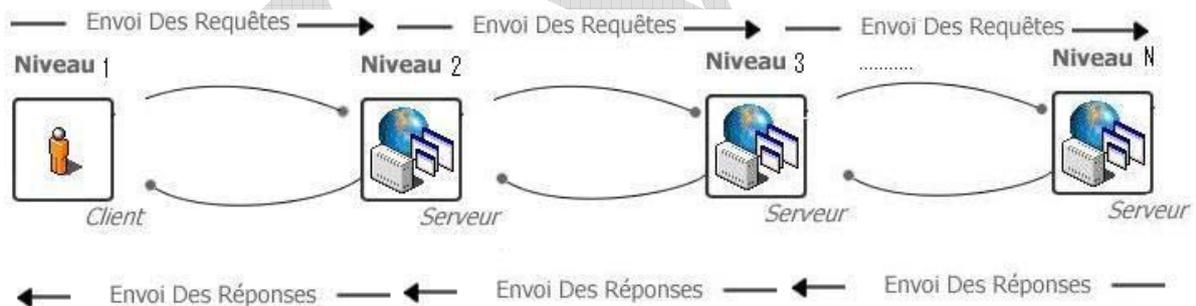


Fig. 3.6. : Architecture à multi niveaux

B/ Fonctionnement d'un site Web dynamique

Pour assurer l'interactivité pour les utilisateurs et la mise à jour des informations dynamiques dans les bases de données, il fallait utiliser des applications Web dynamiques qui font appel à des traitements du côté serveur pour traiter des

formulaires, répondre aux saisies de l'utilisateur et mettre en forme dans des pages HTML générées instantanément, des informations extraites de bases de données.

Dans ce cas, les informations sont structurées au sein d'une base de donnée au tour de laquelle on construit le serveur Web.

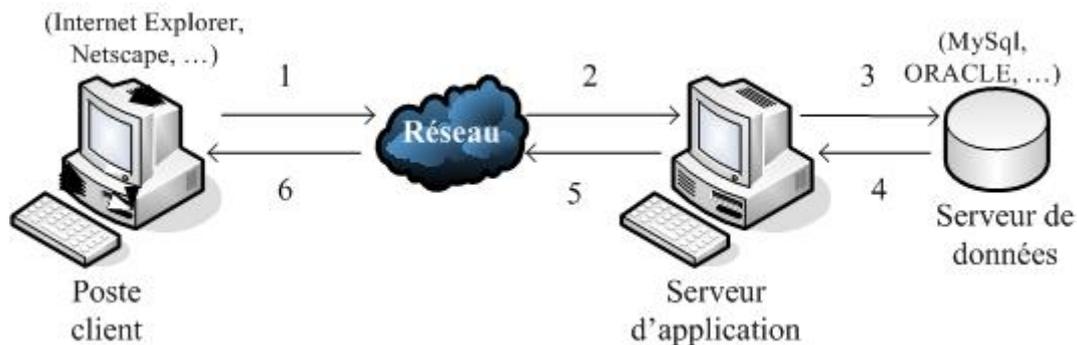


Fig.3.7. : Dialogue entre le serveur, le client et le serveur de données

La figure 3.7. montre les différentes étapes de la communication au sein d'une application dynamique :

1. Le client demande une requête http du serveur d'application.
2. Le serveur d'application reçoit cette requête qui va l'interpréter.
3. Le serveur d'application interroge le serveur de données avec une requête de type SQL, XML, etc.
4. Le serveur de données renvoie la réponse au serveur d'application et la réponse sera formatée au format web.
5. La réponse sera ensuite envoyée via le Web au client.
6. Le client reçoit la réponse qui sera affichée par son navigateur Web.

III.6. Les éditeurs de développement de pages Web

Ecrire une page Web ne nécessite pas de connaître obligatoirement le langage HTML. Ceci peut se faire de plusieurs façons :

III.6.1. Manuellement

Avec l'aide d'un simple éditeur de texte on peut commencer à écrire directement ses pages Web. Cette solution ne nécessite pas d'investissement particulier et permet de faire toutes les opérations que ferait un logiciel spécialisé. Elle offre un autre avantage, car elle permet de créer des pages en écrivant le code HTML. Par exemple, avec *Bloc-notes*, un petit accessoire de Windows, on peut créer des pages Web.

Dans sa forme la plus simple, une page Web correspond à un fichier texte incluant du code au format HTML. On peut créer un fichier HTML dans un éditeur de texte quelconque, en tapant le code HTML et script manuellement, puis enregistrer le fichier avec une extension **.htm**.

III.6.2. En utilisant un éditeur WYSIWYG

Wysiwyg signifie, What You See is What You Get, autrement dit ce que vous voyez est ce que vous obtenez. En effet, ces éditeurs permettent de générer automatiquement les balises HTML à l'aide d'un simple click de souris. Parmi ces éditeurs nous citons : *Frontpage* de Microsoft, *Dreamweaver* de Macromédia et *Webexpert*.

Un éditeur de page HTML doit comprendre certaines entêtes telles que la convivialité de l'interface, la génération automatique du code HTML et l'intégration des éléments multimédia.

III.6.3. En utilisant des convertisseurs de formats bureautiques vers HTML

Un des plus anciens et plus connus convertisseurs se nomme **RTF2HTML**, il est disponible sur micro-ordinateur Mac ou PC et sur systèmes Unix. Il nécessite de convertir au préalable des documents en format RTF.

Les logiciels d'Office 2000 : avec les formats WORD, EXCEL, PowerPoint, on peut convertir immédiatement des documents bureautiques vers le format HTML.

III.7. Quelques solutions existantes pour le Web dynamique

III.7.1. CGI (Common Gateway Interface)

C'est la première technique utilisée pour créer des programmes exécutables sur le serveur Web. Les CGI sont entièrement écrits en langage C mais peuvent être codés en d'autres langages. Cette solution a pour inconvénient que chaque fois qu'un client se connecte à une application CGI, une instance séparée au programme est placée en mémoire. Si des centaines de clients visitent le site, la mémoire sauvegarde des centaines d'instances du programme et chaque appel sera plus long que le précédent.

A cause de cet inconvénient, elles sont apparues d'autres solutions qui ont eu une bibliothèque de lien dynamique DLL (Dynamic Link Library) dans laquelle les ressources du serveur sont situées et ne seront chargées qu'une seule fois quelque soit le nombre d'accès des clients. Cette solution est adaptée par les autres interfaces comme **ASP**, **JSP** et **PHP**.

Un programme CGI peut être écrit dans n'importe quel langage pourvu que celui-ci soit :

- capable de lire le flux de données d'entrée
- capable de traiter des chaînes de caractères
- capable d'écrire sur le flux standard de sortie
- exécutable ou interprétable par le serveur

Les langages de programmation les plus utilisés pour l'écriture des CGI sont:

- Le langage Perl (majoritairement, car il est particulièrement adapté à la manipulation de chaînes de caractères)
- Les langages C, C++
- Java

Lorsqu'une requête est envoyée au serveur, celui-ci examine l'extension du fichier demandé afin de déterminer l'action qu'il va effectuer. Ainsi, par exemple:

- dans le cas d'un fichier HTML (extension .htm ou .html): le serveur va générer des en-têtes de réponse HTTP qu'il va envoyer au navigateur demandant un fichier, puis envoyer le fichier HTML
- dans le cas d'un fichier CGI écrit en Perl (extension .pl) : le serveur va exécuter l'interpréteur Perl, celui-ci va interpréter le code contenu dans le fichier portant l'extension .pl puis envoyer les données au serveur, qui va les envoyer au navigateur, comme dans le cas d'un fichier HTML
- dans le cas d'un fichier CGI écrit en langage C (extension .exe sous Windows puisqu'il s'agit d'un exécutable): le serveur va exécuter le CGI, puis envoyer vers le navigateur les données que le programme lui aura transmis

Ainsi, le rôle d'un programme CGI est de recevoir des données envoyées par le serveur, à les traiter, puis à renvoyer des en-têtes de réponses et des tags HTML valides. En effet, lorsqu'un programme CGI renvoie un fichier, il doit commencer par envoyer un en-tête HTTP permettant de préciser le type de contenu envoyé au navigateur (appelé type MIME).

III.7.2. ASP (Active Server Pages)

Les ASP sont un standard de Microsoft, une page Web ASP est repérable par l'extension .asp. Les ASP sont intégrables au sein d'une page Web en HTML à l'aide de balises spéciales permettant à son serveur Web IIS (Internet Information Server) de savoir que le code compris à l'intérieur de ces balises doit être interprété afin de renvoyer du code HTML au navigateur du client.

ASP est en réalité une technologie, ou plus exactement un environnement de programmation, permettant de représenter sous forme d'objets les interactions entre le navigateur du client, le serveur web, ainsi que les connexions à des bases de données (grâce à ADO, ActiveX Data Objects) ou bien des composants COM (Component Object Model). Les ASP sont donc exécutées du côté du serveur (au même titre que les scripts CGI) et non du côté client.

Ainsi, les Active Server Pages s'inscrivent dans une architecture 3-tiers. Ce terme signifie qu'un serveur supportant les Active Server Pages peut servir d'intermédiaire entre le navigateur du client et une base de données en permettant un accès transparent à celle-ci grâce à la technologie ADO (ActiveX Data Objects), qui fournit les éléments nécessaires à la connexion à un système de gestion de bases de données, à la manipulation des données grâce au langage SQL.

III.7.3. JSP (Java Server Pages)

Les JSP sont un standard de Sun, une page Web JSP est repérable par l'extension .jsp. Il s'agit en réalité d'un langage de script puissant exécuté du côté du serveur et non du côté du client. Les JSP sont intégrables au sein d'une page Web en HTML à l'aide de balises spéciales appelées Servlets permettant à son serveur Web de savoir que le code compris à l'intérieur de ces balises doit être interprété afin de renvoyer du code HTML au navigateur du client.

Les JSP permettent donc d'écrire facilement des servlets, en incluant dans des balises spécifiques le code JSP au sein du fichier HTML. De cette façon, elles fournissent une technologie rapide afin de créer des pages dynamiques.

De plus, les JSP étant basées sur Java côté serveur, elles possèdent toutes les caractéristiques faisant la force de Java :

- les JSP sont multithreadées,
- les JSP sont portables,
- les JSP sont orientées objet,
- les JSP sont sûres.

III.7.4. PHP (Personal Home Pages)

Le PHP est un langage open source, une page Web PHP est repérable par l'extension .php. Il s'agit en réalité d'un langage de script puissant (une syntaxe proche du langage C) exécuté du côté du serveur et non du côté du client.

Malgré la diversité et la multitude des langages de programmation sur Internet, on a choisi de travailler avec PHP avec sa version 5 vu :

- La gratuité et la disponibilité du code source.
- La simplicité d'écriture des scripts.
- La simplicité d'interfaçage avec des bases de données : de nombreux SGBD sont supportés, mais le plus utilisé avec ce langage est MySQL.
- L'intégration au sein de nombreux serveurs Web : Apache, IIS, etc.

III.7.5. AJAX (Asynchronous Javascript And XML)

AJAX est une méthode de développement web basée sur l'utilisation d'un script Javascript pour effectuer des requêtes web à l'intérieur d'une page web sans recharger la page. En effet, le modèle web traditionnel est basé sur une suite de requêtes et de réponses successives, c'est-à-dire une navigation séquentielle de page web en page web. AJAX permet de ne modifier

que la partie de la page web qui nécessite d'être mise à jour en créant une requête HTTP locale et en modifiant tout ou partie de la page web en fonction de la requête HTTP récupérée.

AJAX n'est ni une technologie ni un langage de programmation mais une manière de développer des pages web en se basant sur les technologies suivantes :

- HTML pour la présentation
- DOM (Document Object Model : est un dérivé du langage XML) pour la représentation en objets de la page web
- Javascript et en particulier l'objet XMLHttpRequest pour manipuler des requêtes et des réponses

UVA

Module : Multimédia et développement Web

Chapitre 4 : HTML

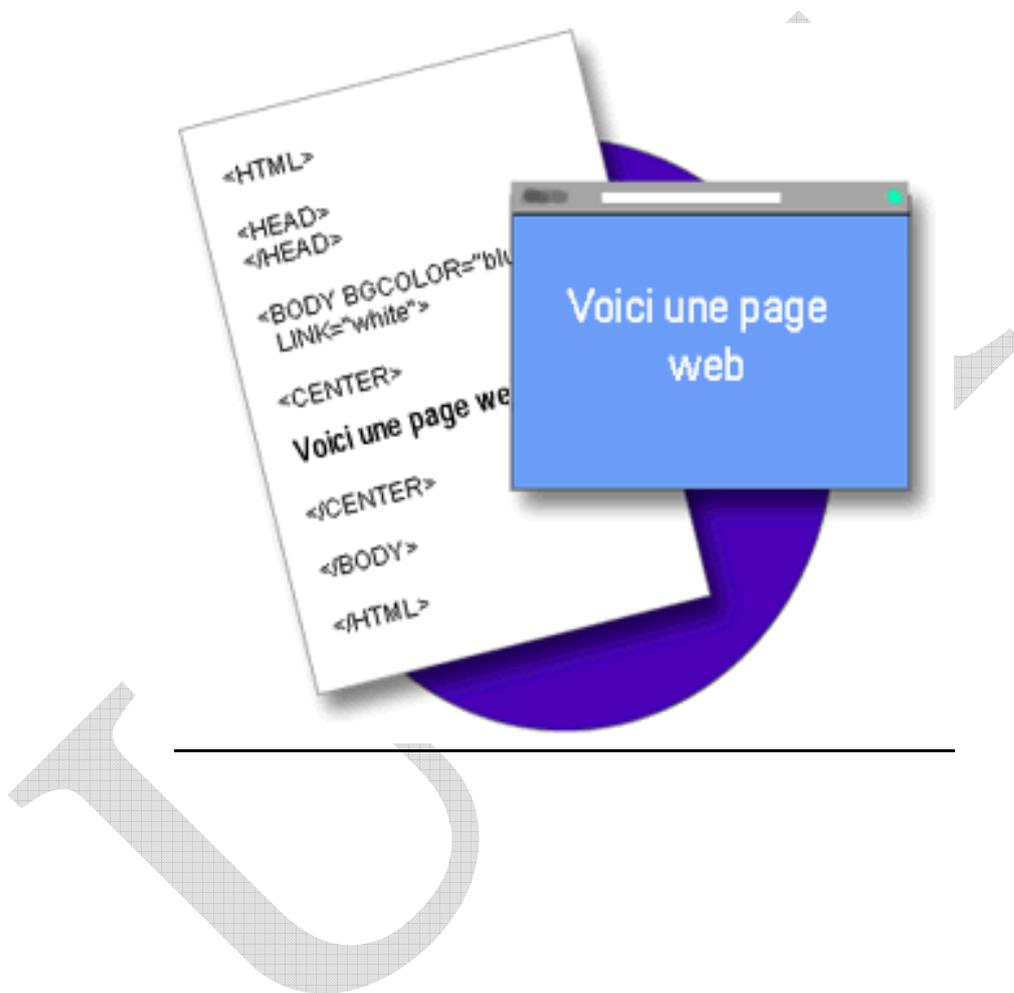
Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Attention !

Ce produit pédagogique numérisé est la propriété exclusive de l'UVT. Il est strictement interdit de la reproduire à des fins commerciales. Seul le téléchargement ou impression pour un usage personnel (1 copie par utilisateur) est permis.

Introduction au langage HTML



Après avoir étudié cette partie, l'étudiant sera en mesure de :

- Connaître la syntaxe du langage HTML.
- Interpréter les séquences en langage HTML.
- Créer des formulaires.

IV.1. Présentation du langage HTML

IV.1.1. Définition

Le HTML ("HyperText Markup Language") est un langage dit de "marquage" chargé de formaliser l'écriture d'un document avec des balises de formatage indiquant la façon dont doit être présenté le document et les liens qu'il établit avec d'autres documents.

Les pages web sont généralement organisées autour d'une page d'accueil, jouant un point central dans la navigation des visiteurs parmi les pages HTML à l'aide des liens hypertextes. Cet ensemble cohérent de pages web liées par des liens hypertextes et articulées autour d'une page d'accueil commune est appelée site web.

IV.1.2. Versions du langage HTML

- *Le langage HTML a été mis au point par Tim Berners-Lee, alors chercheur au CERN, à partir de 1989. Celui-ci annonça officiellement la création du web sur Usenet en août 1991.*
- *Ce n'est cependant qu'à partir de 1993 que l'on considère l'état du HTML suffisamment avancé pour parler de langage (HTML est alors baptisée symboliquement HTML 1.0).*
- *Le RFC 1866, daté de novembre 1995 représente la première version officielle de HTML, c'est-à-dire le HTML 2.0.*
- *Après la brève apparition d'un HTML 3.0, qui ne vit jamais officiellement le jour, le HTML 3.2 devint le standard officiel le 14 janvier 1997.*
- *Le 18 décembre 1997, le HTML 4.0 a été publié.*
- *La version HTML 4.01, apparue le 24 décembre 1999 apporte quelques modifications mineures au HTML 4.0.*

IV.2. Notions de base du langage HTML

IV.2.1. Les balises du langage HTML

Le HTML n'est pas un langage de programmation, Une page HTML est donc un simple fichier texte contenant des balises permettant de mettre en forme le texte, les images, etc.

Les balises HTML sont généralement présentes par paire afin d'agir sur les éléments qu'elles encadrent. La première est appelée "balise d'ouverture" (parfois balise ouvrante) et la seconde "balise de fermeture" (ou fermante). La balise fermante est précédé du caractère / :

`<marqueur> Votre texte formaté </marqueur>`

Exemple :

Les balises `` et `` permettent de mettre en gras le texte qu'elles encadrent :

` Ce texte est en gras `

La page HTML, structuré d'un ensemble de balises, a la structure suivante :

```

<HTML>
  <HEAD>
    <TITLE> Le titre </TITLE>
  </HEAD>
  <BODY>
    Contenu de la page
  </BODY>
</HTML>
    
```

Voici un tableau comportant quelques exemples de balises de style :

Balise de style	Effet Visuel
<code><ABBREV></code>	Abréviation
<code><ACRONYM></code>	Acronyme
<code><AU></code>	L'auteur
<code></code>	Met la police en gras
<code><BIG></code>	Police plus grande
<code><BLINK></code>	Clignote (propre à Netscape)
<code><CITE></code>	<i>Citation</i>
<code><CODE></code>	Instruction

<DFN>	<i>Définition d'instance</i>
	<i>Emphase</i>
<I>	<i>Italique</i>
<INS>	Nouveau texte inséré a cet endroit
<KBD>	Clavier - Suite de caractères devant être tapés tel quel
<PERSON>	Accentuation du nom d'une personne
<Q>	Encadre le texte par des guillemets
<S>	Comme strike (barré)
<SAMP>	Exemple
<SMALL>	Police plus petite
	Forte accentuation rendue par du gras
<STRIKE>	Texte barré (comme S)
<SUB>	Texte en _{Indice}
<SUP>	Texte en ^{Exposant}
<TT>	Caractère de machine à écrire
<VAR>	<i>Nom d'une variable</i>

IV.2.2. Les commentaires

Il est possible d'ajouter des éléments d'information dans une page web sans que ceux-ci soient affichés à l'écran grâce à un jeu de balises spécifique, appelé balises de commentaires.

Exemple :

```
<!-- Voici un commentaire -->
```

Les balises de commentaires permettent de mettre en commentaire du texte mais peuvent également servir à commenter du code HTML.

IV.2.3. Le codage des caractères spéciaux

Les normes HTML demandent de respecter le codage des caractères ASCII 7 bits, c'est-à-dire que les caractères accentués ne sont pas autorisés.

Pourtant, les navigateurs actuels reconnaissent les caractères accentués, ainsi on peut entrer des caractères accentués directement sous l'éditeur de texte, mais la page sera vraisemblablement illisible dans la plupart des pays du monde.

Pour coder un caractère accentué, on entre une combinaison précédée du caractère & et terminée par un point-virgule ;.

Exemples :

ç çç

è èè

É éé

IV.2.4. Les niveaux de titres

Le langage HTML définit 6 niveaux de titre (en anglais heading), afin de définir une structuration hiérarchique des paragraphes dans un texte. Ainsi H1 représente la plus grand taille et H6 la plus petite, comme le montrant le tableau suivant :

Balise	Effet Visuel
H1	Bonjour
H2	Bonjour

H3	Bonjour
H4	Bonjour
H5	Bonjour
H6	Bonjour

IV.2.5. Les attributs

Les attributs peuvent être placés dans les balises de structure afin de permettre une disposition plus précise des éléments HTML.

Un exemple d'en-tête :

```
<H1 ALIGN=JUSTIFY> Texte justifié </H1>
```

Voici une liste des principaux attributs :

Attribut	Valeur	Effet Visuel
ALIGN	LEFT	Texte aligné à gauche
	RIGHT	Texte aligné à droite
	CENTER	Texte centré
	JUSTIFY	Texte justifié
NOWRAP		Interdit au navigateur d'effectuer un saut de ligne.
ID		Attribue un identifiant à la balise. Cet attribut est notamment utile pour la programmation en JavaScript
LANG		Spécifie un langage différent
CLASS		Assigne une classe au contenu (pour les feuilles de style).
CLEAR		Utilisé dans le cas d'un texte entourant une image. Il permet de descendre

		dans la page aussi loin qu'il est nécessaire pour atteindre une marge libre
--	--	---

IV.2.6. Les tableaux en HTML

Les tableaux sont définis comme étant des suites de lignes.

Le tableau est encadré par les balises **<TABLE>** et **</TABLE>**.

Le titre du tableau est encadré par **<CAPTION>** **</CAPTION>**.

Chaque ligne est encadrée par **<TR>** **</TR>** (Table Row, traduite par ligne du tableau).

Les cellules d'en-tête sont encadrées par **<TH>** **</TH>** (pour Table Header : En-tête de tableau).

Les cellules de valeur sont encadrées par **<TD>** **</TD>** (Table Data: Donnée de tableau)

Exemple :

```
<TABLE BORDER="1">
  <CAPTION> Voici le titre du tableau </CAPTION>
  <TR>
    <TH> Titre A1 </TH>
    <TH> Titre A2 </TH>
    <TH> Titre A3 </TH>
    <TH> Titre A4 </TH>
  </TR>
  <TR>
    <TH> Titre B1 </TH>
    <TD> Valeur B2 </TD>
    <TD> Valeur B3 </TD>
    <TD> Valeur B4 </TD>
  </TR>
</TABLE>
```

Voici le résultat donné par cet exemple :

Voici le titre du tableau			
Titre A1	Titre A2	Titre A3	Titre A4
Titre B1	Valeur B2	Valeur B3	Valeur B4

Exemples d'attributs de tableau :

- *Largeur d'une cellule* `<TD width=?>` en pixels
`<TD width=?>` en pourcentage
- *Fusion de lignes* `<TD rowspan=?>`
- *Fusion de colonnes* `<TD colspan=?>`
- *Ligne de tableau* `<TR align=left/center/right>` horizontalement
`<TR valign=top/middle/bottom>` verticalement
- *Cellule de tableau* `<TD align=left/center/right>` horizontalement
`<TD valign=top/middle/bottom>` verticalement
- *Couleur de la cellule* `<TD BGCOLOR="#$$$$$">`

IV.2.7. Les feuilles de style

IV.2.7.1. Définition

Le concept de **feuilles de style** est apparu en 1996 avec la publication par le W3C d'une nouvelle recommandation intitulée « *Cascading StyleSheets* » (*feuilles de style en cascade*), notée **CSS**.

Les feuilles de style permettent notamment :

- d'obtenir une présentation homogène sur tout un site en faisant appel sur toutes les pages à une même définition de style ;
- de permettre le changement de l'aspect d'un site complet entier par la seule modification de quelques lignes ;
- une plus grande lisibilité du HTML, car les styles sont définis à part ;
- des chargements de page plus rapides, pour les mêmes raisons que précédemment ;
- un positionnement plus rigoureux des éléments.

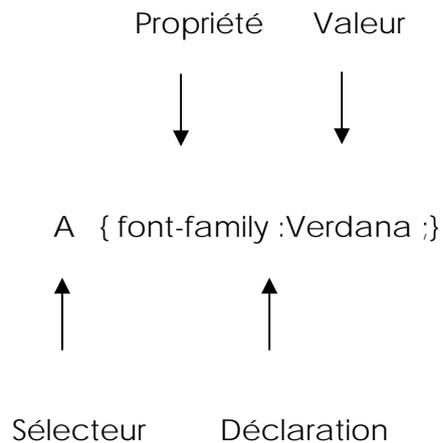
IV.2.7.2. Syntaxe

Une règle CSS est caractérisée par deux principaux éléments :

- Un **sélecteur de balises**, permettant de préciser à quelles balises du document le style s'applique ;
- Une **déclaration** de style, définie entre accolades, permettant de préciser le style à appliquer aux balises sélectionnées. La déclaration est elle-même constituée :

- d'une ou plusieurs **propriété(s)**, suivie du caractère « : » (double point),
- d'une ou plusieurs **valeur(s)** associée(s) à chaque propriété, entourée de guillemets s'il s'agit de plusieurs mots ou séparés par des virgules s'il y en a plusieurs, suivie d'un point virgule.

La syntaxe est ainsi la suivante :



La syntaxe suivante définit par exemple le style à appliquer aux liens hypertextes (balise <A>), c'est-à-dire une police Verdana de taille 18 pixels, en gras et de couleur jaune :

```
A {
    font-family: Verdana;
    font-size: 18px;
    font-style: bold;
    color: yellow
}
```

IV.2.8. Les frames

IV.2.8.1 Division de l'écran

Il est possible de diviser un écran en plusieurs cadres, chacun pouvant afficher un document HTML. Il est déconseillé d'utiliser les cadres car :

- Ce ne sont pas tous les navigateurs qui sont en mesure de les supporter

- Un affichage avec cadres est plus lourd à gérer qu'un affichage sur un seul écran

IV.2.8.2 La commande <FRAME>

L'ensemble des commandes servant à définir les cadres ne doivent pas se retrouver à l'intérieur de <BODY>. C'est l'élément <FRAMESET> qui remplace <BODY> à cet effet.

<FRAMESET> accepte deux attributs, ROWS et/ou COLS qui servent à déterminer la grandeur des cadres (pixels ou pourcentage)

IV.2.8.3 Un exemple de structure

```

<HTML>

  <HEAD>

    <TITLE> ...</TITLE>

  </HEAD>

  <FRAMESET COLS="30%,*">

    <FRAME NAME="NOM1" SRC="URL1">

    <FRAME NAME="NOM2" SRC="URL2">

    <NOFRAME>

      <BODY>

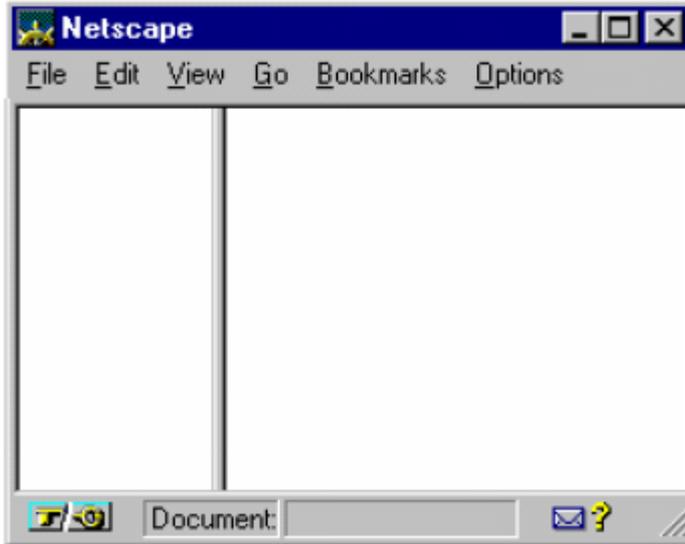
      </BODY>

    </NOFRAME>

  </FRAMESET>

</HTML>
  
```

Dans le navigateur on trouve :



IV.2.8.4 Caractéristiques de <FRAME>

Chaque cadre est défini par la commande FRAME. Un nom est donné à chaque cadre avec l'attribut NAME = "unNom".

À chaque cadre doit correspondre un fichier HTML appelé par SRC = "URL" où URL détermine le document à charger dans le cadre

IV.2.8.5 Ouverture d'un document dans un cadre

Pour créer un lien vers un cadre, il faut utiliser la commande habituelle <A HREF ...

Toutefois, l'option TARGET doit être utilisée pour référer au cadre à l'aide de son nom :

```
<A HREF="URL" TARGET="NomDuCadre"> Texte du lien</A>
```

L'attribut TARGET peut prendre certaines valeurs prédéfinies :

- _blank qui indique au browser qu'il doit créer une nouvelle fenêtre afin d'y afficher le fichier. Dans ce cas, on ouvre en fait un nouveau browser.
- _self qui indique que le fichier sera chargé dans la même fenêtre que celle dans laquelle se trouve le lien.
- _top qui implique l'affichage du fichier sur toute la surface de la fenêtre du browser.

IV.2.9. Les liens hypertextes et ancrages

Les ancrages (liens hypertextes) sont des éléments d'une page HTML (soulignés lorsqu'il s'agit de texte) qui emmènent dans un autre endroit lorsqu'on clique dessus.

C'est ce qui permet de lier des pages Web entre elles.

Les liens hypertextes permettent de naviguer:

- Vers un autre endroit du document.
- Vers un fichier HTML situé à un emplacement différent sur la machine qui héberge la page.
- Vers une autre machine.

L'attribut principal des ancrages est href. Il s'écrit sous la forme:

```
<a href="Adresse ou URL"> .. </a>
```

Exemple :

```
<a href=" ../index.html"> ... </a>
```

IV.3. Les formulaires avec HTML

IV.3.1. Présentation d'un formulaire

Le lecteur saisit des informations en remplissant des champs ou en cliquant sur des boutons, puis appuie sur un bouton de soumission (submit) pour l'envoyer.

*Les formulaires sont délimités par la balise **<FORM>** ... **</FORM>**, une balise qui permet de regrouper plusieurs éléments de formulaire (boutons, champs de saisie,...) et qui possède les attributs obligatoires suivants:*

- **METHOD** indique sous quelle forme seront envoyées les réponses.

- **"POST"** est la valeur qui correspond à un envoi de données stockées dans le corps de la requête
- **"GET"** correspond à un envoi des données codées dans l'URL, et séparées de l'adresse du script par un point d'interrogation

- **ACTION** indique l'adresse d'envoi.

Exemple :

```
<form name="form1" method="post" action="page.htm"> </form>
```

IV.3.2. A l'intérieur de la balise FORM

La balise *FORM* constitue en quelque sorte un conteneur permettant de regrouper des éléments qui vont permettre à l'utilisateur de choisir ou de saisir des données, ensemble de données qui seront envoyées à l'URL indiqué dans l'attribut *ACTION* de la balise *FORM* par la méthode indiquée par l'attribut *METHOD*.

Il est possible d'insérer n'importe quel élément HTML de base dans une balise *FORM* (textes, boutons, tableaux, liens, etc.) mais il est surtout intéressant d'insérer des éléments interactifs. Ces éléments interactifs sont :

- La balise **INPUT** : un ensemble de boutons et de champs de saisie
- La balise **TEXTAREA** : une zone de saisie
- La balise **SELECT** : une liste à choix multiples

A/ La balise INPUT

La balise *INPUT* est la balise essentielle des formulaires, car elle permet de créer un bon nombre d'éléments "interactifs". La syntaxe de cette balise est la suivante:

```
<INPUT type="Nom du champ" value="Valeur par défaut" name="Nom de l'élément">
```

- L'attribut **name** est essentiel car il permettra au script CGI de connaître le champ associé à la paire nom/valeur, c'est-à-dire que le nom du champ sera suivi du caractère "=" puis de la valeur entrée par l'utilisateur, ou dans le cas contraire de la valeur par défaut repéré par l'attribut *value*.

- L'attribut **type** permet de préciser le type d'élément que représente la balise *INPUT* :

- **radio**: il s'agit d'un bouton permettant un choix parmi plusieurs proposés (l'ensemble des boutons radios devant porter le même attribut *name*. Un attribut *checked* pour un des boutons permet de préciser le bouton sélectionné par défaut.
- **reset** : il s'agit d'un bouton de remise à zéro permettant uniquement de rétablir l'ensemble des éléments du formulaire à leurs valeurs par défaut
- **submit** : il s'agit du bouton de soumission permettant l'envoi du formulaire. Le texte du bouton peut être précisé grâce à l'attribut *value*.
- **text** : il s'agit d'un champ de saisie permettant la saisie d'une ligne de texte. La taille du champ peut être définie à l'aide de l'attribut *size* et la taille maximale du texte saisi grâce à l'attribut *maxlength*
- **checkbox**: il s'agit de cases à cocher pouvant admettre deux états: *checked* (coché) et *unchecked* (non coché).
- **hidden**: il s'agit d'un champ caché. Ce champ est non visible sur le formulaire.
- **file**: il s'agit d'un champ permettant à l'utilisateur de préciser l'emplacement d'un fichier qui sera envoyé avec le formulaire.
- **image**: il s'agit d'un bouton de soumission personnalisé, dont l'apparence est

l'image.

- **password**: il s'agit d'un champ de saisie, dans lequel les caractères saisis apparaissent sous forme d'astérisques afin de camoufler la saisie de l'utilisateur.

Exemple :

```
<input type="text" name="nom" id="nom">
```

B/ La balise TEXTAREA

La balise *TEXTAREA* permet de définir une zone de saisie plus vaste par rapport à la simple ligne de saisie que propose la balise *INPUT*. Cette balise possède les attributs suivants:

- **cols** : représente le nombre de caractères que peut contenir une ligne.
- **rows** : représente le nombre de lignes.
- **name** : représente le nom associé au champ, c'est le nom qui permettra d'identifier le champ dans la paire nom/valeur.
- **readonly** : permet d'empêcher l'utilisateur de modifier le texte entré par défaut dans le champ.
- **value** : représente la valeur qui sera envoyée par défaut au script si le champ de saisie n'est pas modifié par une frappe de l'utilisateur

Exemple :

```
<textarea name="adresse" id="adresse"></textarea>
```

C/ La balise SELECT

La balise *SELECT* permet de créer une liste déroulante d'éléments (précisés par des balises *OPTION* à l'intérieur de celle-ci). Les attributs de cette balise sont :

- **name**: représente le nom associé au champ, c'est le nom qui permettra d'identifier le champ dans la paire nom/valeur.
- **disabled** : permet de créer une liste désactivée, c'est-à-dire affichée en grisée.
- **size** : représente le nombre de lignes dans la liste (cette valeur peut être plus grande que le nombre d'éléments effectifs dans la liste).
- **multiple** : marque la possibilité pour l'utilisateur de choisir plusieurs champs dans la liste

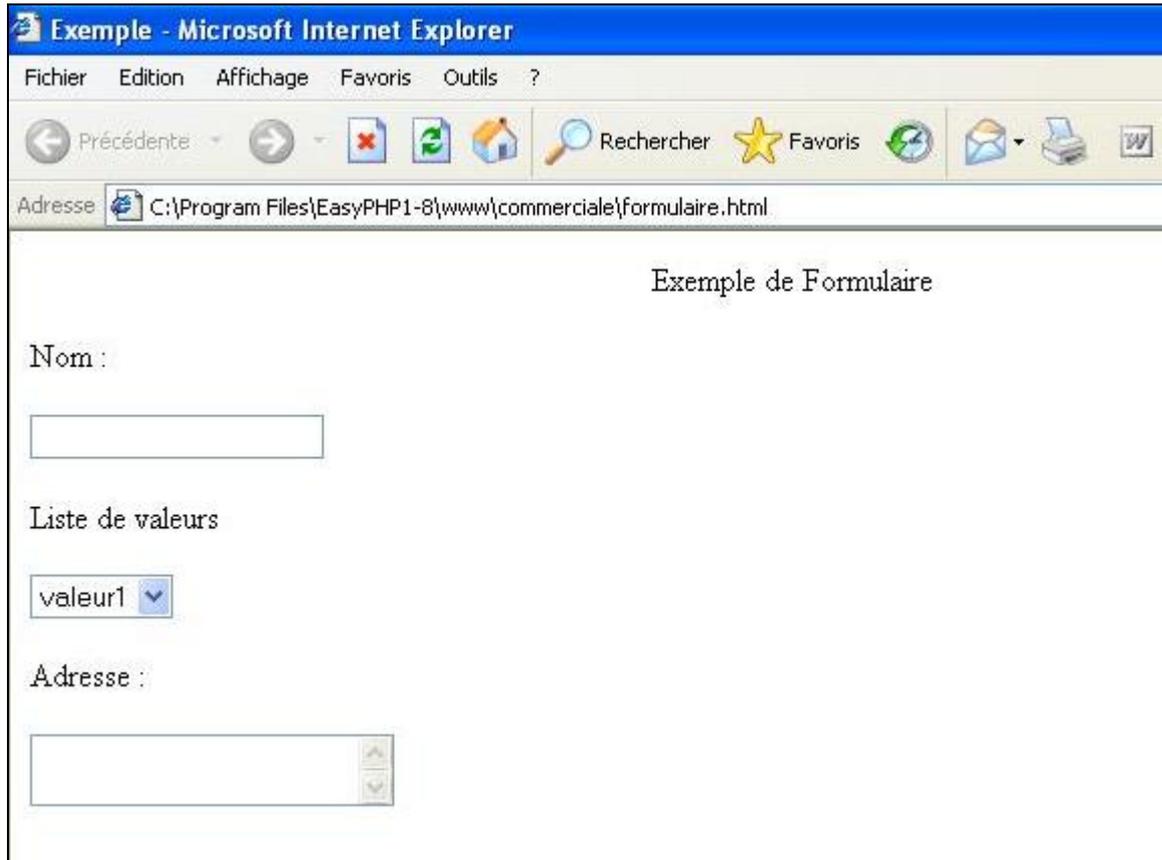
Exemple :

```
<select name="listedevaleurs">
  <option>valeur1</option>
  <option>valeur2</option>
</select>
```

D/ Exemple de formulaire

```
<html>
<head>
<title>Exemple</title>
</head>
<body>
<form name="form1" method="post" action="page.htm">
  <p align="center">Exemple de Formulaire </p>
  <p align="left">Nom :</p>
  <p>
    <input type="text" name="nom" id="nom3">
  </p>
<p>Liste de valeurs</p>
  <p>
    <select name="select">
      <option>valeur1</option>
      <option>valeur2</option>
    </select>
  </p>
  <p>Adresse :</p>
  <p>
    <textarea name="adresse" id="adresse"></textarea>
  </p>
</form>
</body>
</html>
```

Cet exemple donne le résultat suivant :





Module : Multimédia et développement Web

Chapitre 5 : JAVASCRIPT

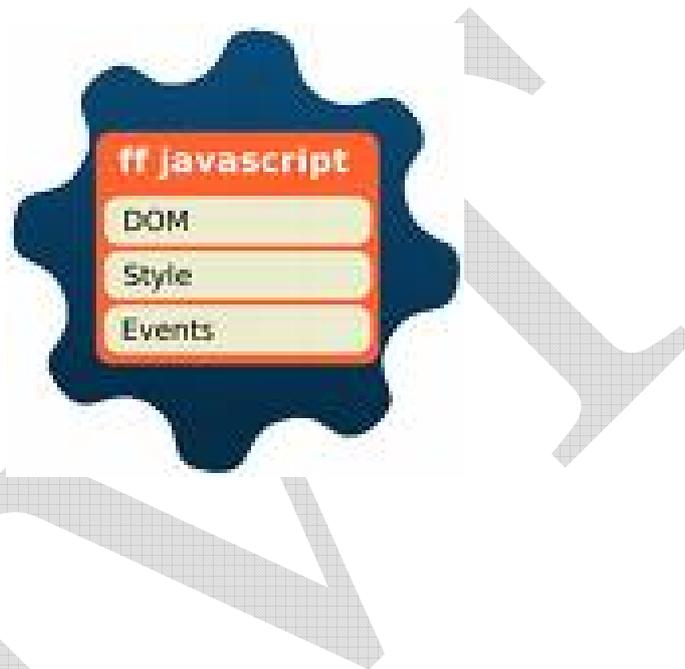
Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Attention !

Ce produit pédagogique numérisé est la propriété exclusive de l'UVT. Il est strictement interdit de la reproduire à des fins commerciales. Seul le téléchargement ou impression pour un usage personnel (1 copie par utilisateur) est permis.

JAVASCRIPT



Après avoir étudié cette partie, l'étudiant sera en mesure de :

- Comprendre les balises du script.
- Connaître l'environnement du langage javascript.
- Comprendre les événements.
- Définir les objets du navigateur.

V.1. Introduction

Le **Javascript** est un langage de script incorporé dans un document HTML. Historiquement il s'agit même du premier langage de script pour le Web.

Ce langage est un langage de programmation qui permet d'apporter des améliorations au langage HTML en permettant d'exécuter des commandes du côté client, c'est-à-dire au niveau du navigateur et non du serveur web.

Ainsi le langage Javascript est fortement dépendant du navigateur appelant la page web dans laquelle le script est incorporé. Mais en contrepartie il ne nécessite pas de compilateur, contrairement au langage Java, avec lequel il a longtemps été confondu.

V.2. Historique

Javascript a été mis au point par Netscape en 1995.

A l'origine, il se nommait LiveScript et était destiné à fournir un langage de script simple au navigateur *Netscape Navigator 2*.

Il a été critiqué pour son manque de sécurité, son développement peu poussé et l'absence de messages d'erreurs explicites rendant dure son utilisation.

Le 4 décembre 1995, suite à une association avec le constructeur Sun, Netscape rebaptise son langage *Javascript*.

A la même époque, Microsoft mit au point le langage *Jscript*, un langage de script très similaire.

Ainsi, un standard a été défini pour normaliser les langages de script, il s'agit de l'**ECMA 262**, créé par l'organisation du même nom (ECMA, *European Computer Manufactures Association*).

V.3. JavaScript Vs Java

Très longtemps Javascript a été confondu avec java. Néanmoins les différences sont nombreuses. Dans le tableau ci-dessous, une comparaison entre les deux langages.

Java Script	Java
Langage interprété	Langage pseudo-compilé (chargement d'une machine virtuelles)
Code intégré au HTML	Code (applet) à part du document HTML, appelé à parir de la page
Langage peu typé	Langage fortement typé (déclaration du type de variable)
Liaisons dynamiques : les références des objets sont vérifiées au chargement	Liaisons statiques : les objets doivent exister au chargement (compilation)
Accessibilité du code	Confidentialité du code
Sûr : ne peut pas écrire sur le disque dur	Sûr : ne peut pas écrire sur le disque dur

V.4. Les balises du script

Les balises annonçant un code Javascript sont les suivantes :

```
<SCRIPT language="Javascript">
```

Placez ici le code du script

```
</SCRIPT>
```

D'anciens navigateurs, créés avant l'arrivée du Javascript, ne connaissent pas ces balises et donc les ignorent. Le code du Javascript risque donc de s'afficher sur la page web et venir gêner le travail.

L'astuce consiste donc à ajouter des balises de commentaires à l'intérieur même des balises de script. Ainsi les anciens navigateurs ignoreront tout simplement l'intégralité du script, tandis que les navigateurs récents l'interpréteront comme du Javascript.

```
<SCRIPT language="Javascript">  
<!-- Placez ici le code du script // -->  
</SCRIPT>
```

Exemple de script :

```
<HTML>  
<HEAD>  
<TITLE>  
Voici une page contenant du Javascript  
</TITLE>  
</HEAD>  
<BODY>  
<SCRIPT language="Javascript">  
<!--  
alert("Attention!");  
// -->  
</SCRIPT>  
</BODY>  
</HTML>
```

Le résultat obtenu est le suivant :



V.5. Les variables

Le Javascript étant très souple au niveau de l'écriture, la déclaration des variables peut se faire de deux façons :

- soit de **façon explicite**, en faisant précéder la variable du mot clé **var** qui permet d'indiquer de façon rigoureuse qu'il s'agit d'une variable :

Exemple :

```
var chaîne= "bonjour"
```

- soit de **façon implicite**, en écrivant directement le nom de la variable suivie du caractère = et de la valeur à affecter. Le navigateur détermine seul qu'il s'agit d'une déclaration de variable :

Exemple :

```
chaîne= "bonjour"
```

V.5.1. Portée de variable

Selon l'endroit où on déclare une variable, celle-ci pourra être accessible (visible) de partout dans le script ou bien uniquement dans une portion confinée du code, on parle de "**portée**" d'une variable.

Lorsqu'une variable est déclarée sans le mot clé *var*, c'est-à-dire **de façon implicite**, elle est accessible de partout dans le script (n'importe quelle fonction du script peut faire appel à cette variable). On parle alors de **variable globale**. Lorsque l'on déclare **de façon explicite** une variable en Javascript (en précédant sa déclaration avec le mot *var*), sa portée dépend de l'endroit où elle est déclarée.

Une variable déclarée au début du script, c'est-à-dire avant toute fonction sera globale, on pourra alors les utiliser à partir de n'importe quel endroit dans le script.

Une variable déclarée par le mot clé *var* dans une fonction aura une portée limitée à cette seule fonction, c'est-à-dire qu'elle est inutilisable ailleurs, on parle alors de variable locale.

Exemple :

```
<SCRIPT language="Javascript">
<!--
var a = 12; var b = 4;
function MultipliePar2(b)
{ var a = b * 2; return a; }
document.write("Le double de ",b," est ",MultipliePar2(b));
document.write("La valeur de a est ",a);
// -->
</SCRIPT>
```

La variable a est déclarée explicitement en début de script, ainsi que dans la fonction. Voici ce qu'affiche ce script :

Le double de 4 est 8

La valeur de a est 12

V.5.2. Déclaration des variables

Le Javascript n'autorise la manipulation que de 4 types de données :

- **des nombres** : entiers ou à virgules
- **des chaînes de caractères** (string) : une suite de caractères entre ' ' ou " " .
- **des booléens** : des variables à deux états permettant de vérifier une condition.

true : si le résultat est vrai représenté par 1.

false : lors d'un résultat faux représenté par 0.

- **des variables de type *null*** : un mot caractéristique pour indiquer qu'il n'y a pas de données.

V.5.3. Conversion de type

Même si Javascript gère de façon transparente les changements de type des variables, il est parfois nécessaire de forcer la conversion du type.

Ainsi Javascript fournit deux fonctions natives permettant de convertir le type des variables passées en paramètre :

- **parseInt** permet de convertir une variable en nombre.
- **parseInt(chaine[, base])**; Si le paramètre base n'est pas précisé (ou s'il est fixé à la valeur 10), la base utilisée sera la base décimale; la base sera 16 si la chaîne commence par 0x, elle sera 8 si la chaîne commence par 0.

Exemple :

parseInt("00100110",2); donne 38

- **parseFloat** permet de convertir une variable en nombre décimal.
- **parseFloat(chaine)** permet de convertir une chaîne en nombre décimal

Exemple :

```
parseFloat("0284");
```

V.6. Les tableaux

La déclaration d'un tableau se fait comme suit :

```
var MonTableau = new Array();
```

Le langage Javascript fournit plusieurs autres façons de créer un tableau :

- **var MonTableau = ["donnée 1", "donnée 2", "donnée 3"];**
- **var MonTableau = new Array("donnée 1", "donnée 2", "donnée 3");**

Exemple :

```
var MonTableau = ["Mohamed", "Saleh", "Hedia"];  
document.write("Le 3ième élément du tableau est "+MonTableau[2]);  
// Affichera "Le 3ième élément du tableau est Hedia"
```

V.7. Les chaînes de caractères

On représente une chaîne de caractère par la suite de caractères encadrée par des guillemets simples (') ou doubles ("), sachant que les deux types de guillemets ne peuvent être mélangés pour une même chaîne.

Il existe des caractères spéciaux à utiliser dans les chaînes pour simuler d'une part des caractères non visuels ou pour éviter au navigateur de confondre les caractères de la chaîne avec ceux du script, ces caractères sont précédés d'un antislash (\) :

\b : touche de suppression

\f : formulaire plein

\n : retour à la ligne

\r : appui sur la touche *ENTREE*

\t : tabulation

\\" : guillemets doubles

\' : guillemets simples

\\ : caractère antislash

L'**objet String**, pour manipuler les chaînes de caractères.

V.8. Les événements

V.8.1. Présentation

Les événements sont des actions de l'utilisateur, qui vont pouvoir donner lieu à une interactivité.

L'événement par excellence est le clic de souris, car c'est le seul que le HTML gère.

Grâce au Javascript il est possible d'associer des fonctions, des méthodes à des événements tels que le passage de la souris au-dessus d'une zone, le changement d'une valeur, etc. Ce sont les gestionnaires d'événements qui permettent d'associer une action à un événement.

La syntaxe d'un gestionnaire d'événement est la suivante:

```
onEvenement="Action_Javascript_ou_Fonction( );"
```

Lorsqu'il est utilisé dans un lien hypertexte, par exemple, la syntaxe sera la suivante :

```
<A href="URL" "onEvenement='Action_Javascript_ou_Fonction()'">Lien</a>
```

V.8.2. Association des événements aux objets

- **onBlur** : se produit quand un texteur, un text ou un select perd la main sur les entrées clavier
- **onChange** : se produit quand un textarea, un text ou un select est modifié par l'utilisateur
- **onClick** : se produit quand un button, checkbox, radio, link, reset ou un submit reçoit un click de la souris
- **onFocus** : se produit quand un textarea, un text ou un select prend la main sur les entrées clavier
- **onLoad** : se produit quand le navigateur a fini de charger une fenêtre ou toutes les frames d'un FRAMESET. L'événement *onLoad* se positionne dans la balise BODY ou dans la balise FRAMESET
- **onMouseOver** : se produit quand la souris passe sur un link ou un layer (version 1.2)
- **onSelect** textarea, un text : se produit quand un textarea, un text ou un text est sélectionné.
- **onSubmit** : se produit quand une form est soumise au serveur par l'appui du bouton *Submit*.
- **onUnload** : se produit quand un document est quitté. L'événement *onUnload* se positionne dans la balise BODY ou dans la balise FRAMESET
- **onAbort** : (*version 1.1*) se produit quand l'utilisateur avorte le chargement d'une image
- **onError** : (*version 1.1*) se produit quand le chargement d'une page ou d'une image produit une erreur.
- **onMouseout** : (*version 1.1*) se produit quand la souris quitte une zone area , un link ou un layer (version 1.2)
- **onReset** : (*version 1.1*) se produit quand on clique sur le bouton *reset* d'un formulaire
- **onDblick** : (*version 1.2*) se produit quand on produit un double click sur la souris (ne fonctionne pas sur Macintosh ou le double clic a sa signification propre)
- **onDragDrop** : (*version 1.2*) se produit lorsqu'on fait un glisser lacher vers le navigateur, par exemple pour ouvrir une page HTML depuis son disque dur.
- **onKeyDown** : (*version 1.2*) se produit quand une touche du clavier est lâchée.
- **onKeyPress** : (*version 1.2*) se produit quand une touche du clavier a été utilisée
- **onKeyUp** : (*version 1.2*) se produit quand une touche du clavier est appuyée
- **onMouseDown** : (*version 1.2*) se produit quand une touche de la souris est lâchée.

- **onMouseMove** : (*version 1.2*) se produit quand le curseur de la souris est bougé
- **onMouseUp** : (*version 1.2*) se produit quand un bouton de la souris est relâché
- **onMove** : (*version 1.2*) se produit quand une fenêtre est dépacée
- **onResize** : (*version 1.2*) se produit quand une fenêtre subit un changement de taille

V.9. Les méthodes

Une méthode est une fonction associée à un objet, c'est-à-dire une action que l'on peut faire exécuter à un objet.

Les méthodes des objets du navigateur sont des fonctions définies à l'avance par les normes HTML, on ne peut donc pas les modifier.

Il est possible de créer une méthode personnelle pour un objet que l'on a créé soi-même.

Une méthode s'appelle un peu comme une propriété, c'est-à-dire de la manière suivante :

```
window.objet1.objet2.methode()
```

Exemple :

Prenons l'objet **document** qui a par exemple la méthode `write()` qui lui est associée et qui permet de modifier le contenu de la page HTML en affichant du texte.

Dans le cas de la méthode `write()`, l'appel se fait comme suit :

```
window.document.write(expression1, expression2, etc.)
```

Exemple :

```
document.write("bonjour");
```

⇒ permet d'écrire le résultat des expressions passées en paramètre dans le document dans lequel elle est utilisée

V.9.1. Définir une méthode pour un objet

Il est possible de définir une méthode pour un type d'objet que l'on a créé en incluant une fonction en tant que propriété d'un objet.

Exemple :

On définit un propriétaire :

```
function Proprietaire(Nom, Prenom)
{ Proprietaire.Nom = Nom;
  Proprietaire.Prenom = Prenom; }
```

Puis on crée une fonction qui affiche le nom du propriétaire d'une fleur ainsi que quelques caractéristiques de celui-ci :

```
function AfficheInfos()
{ alert(this.Proprietaire.Prenom + this.Proprietaire.Nom + 'possede un' +
  this.Type); }
```

Il suffit maintenant de définir l'objet *Fleur* comme suit :

```
function Fleur(Type, Taille, Age, Proprietaire, AfficheInfos)
{ this.Type = Type;
  this.Taille = Taille;
  this.Age = Age;
  this.Proprietaire = Proprietaire;
  this.AfficheInfos = AfficheInfos; }
```

Le fait de taper *Fleur.AfficheInfos()* aura pour effet de produire un message d'alerte du type :

Mohamed Tounsi possède une rose rouge

Car la méthode *AfficheInfos()* est appliquée à l'objet *Fleur*.

V.9.2. Boîte de dialogue

Une boîte de dialogue est une fenêtre qui s'affiche au premier plan suite à un événement.

Javascript en propose trois différentes. Ce sont des méthodes de l'objet *window*.

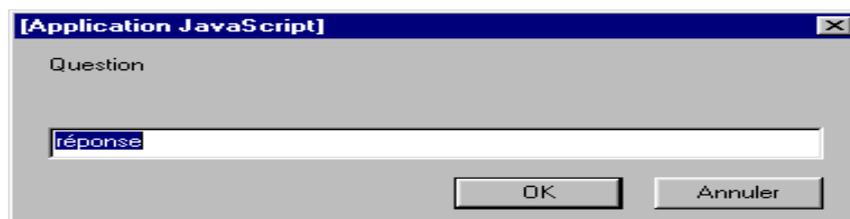
- La méthode *alert()* ex. : `alert('Attention');`



- La méthode *confirm()* ex. : `confirm('Voulez-vous continuer');`



- La méthode *prompt()* ex. : `prompt('Question', 'réponse');`



V.10. La notion d'objet

Le Javascript traite les éléments qui s'affichent dans le navigateur comme des objets, c'est-à-dire des éléments classés selon une hiérarchie pour pouvoir les désigner précisément auxquels on associe des propriétés.

On commence généralement par l'objet le plus grand (celui contenant tous les autres) puis on descend dans la hiérarchie jusqu'à arriver à l'objet voulu.

L'objet le plus grand est l'objet **fenêtre** : *window*. Dans la fenêtre s'affiche une page, c'est l'objet **document**. Cette page peut contenir plusieurs objets, comme des formulaires, des images, etc.

Exemple :



Le formulaire (auquel on a donné le nom *form1*, pour le distinguer des autres formulaires de la page) est repéré par le code suivant :

```
window.document.forms["form1"]
```

Le bouton checkbox (auquel on a donné le nom *checkbox*) est repéré par le code suivant :

```
window.document.forms["form1"].checkbox
```

Le champ de texte (auquel on a donné le nom *champ_text*) est repéré par le code suivant :

```
window.document.forms["form1"].champ_text
```

Le bouton *checkbox* a entre autre une propriété *checked*, qui retourne la valeur 1 si le bouton est coché, 0 dans le cas contraire.

Ainsi le formulaire *form1* a comme code HTML :

```
<form name="form1">  
  
<br><input                type="checkbox"                name="checkbox"  
onClick="ModifChamp();return true;">  
  
<br><input type='TEXT' name='champ_text' value='Essai du javascript'  
size='24'>  
  
</form>
```

Et la fonction javascript associée au bouton *checkbox* est :

```
<script language="Javascript">  
  
<!--  
function ModifChamp()  
{ if (document.forms["form1"].checkbox.checked)  
{document.forms["form1"].champ_text.value='Bouton coché'}  
Else  
{document.forms["form1"].champ_text.value='Bouton non coché'} }  
// -->  
</script>
```

V.11. Les objets du noyau javascript

Javascript fournit en standard un certain nombre d'objets proposant des propriétés et des méthodes permettant d'effectuer simplement un grand nombre de manipulations.

Ces objets sont appelés *objets du noyau Javascript*, car ils ont été standardisés par l'ECMA (*European Computer Manufacturer Association*) et constituent la base du langage Javascript.

Les objets du noyau sont indépendants du navigateur du client contrairement aux objets du navigateur.

Voici la liste des objets du noyau Javascript :

Nom de l'objet	Description
Array	L'objet Array permet de créer des tableaux. Il possède de nombreuses méthodes permettant d'ajouter, de supprimer ou d'extraire des éléments d'un tableau ainsi que de les trier.
Boolean	L'objet Boolean permet de créer des valeurs booléennes, c'est-à-dire des éléments possédant deux états : vrai et faux.
Date	L'objet Date permet de créer des dates et des durées. Il propose des méthodes permettant de les manipuler.
Function	L'objet Function permet de définir des fonctions personnalisées.
Math	L'objet Math permet de manipuler des fonctions mathématiques, comme par exemple les fonctions trigonométriques.
Number	L'objet Number est un objet permettant de faire des opérations de base sur les nombres.
RegExp	L'objet RegExp permet de créer des expressions régulières, c'est-à-dire des éléments permettant de faire des opérations avancées sur les chaînes de caractères.
String	L'objet String propose une grande variété de méthodes permettant de manipuler des chaînes de caractères.

V.11.1. L'objet Array

L'objet Array est un objet du noyau Javascript permettant de créer et de manipuler des [tableaux](#).

Voici la syntaxe à utiliser pour créer une variable tableau :

```
var x = new Array(element1[, element2, ...]);
```

Si aucun élément n'est précisé en paramètre, le tableau est vide à la création. Dans le cas contraire, il sera initialisé avec les valeurs des éléments passés en paramètres.

L'objet Array possède deux propriétés caractéristiques: les propriétés `input` et `length`.

Le tableau suivant décrit les propriétés de l'objet Array :

Propriété	description
constructor	Cette propriété contient le constructeur de l'objet Array.
input	Cette propriété permet de faire une recherche dans le tableau à l'aide d'une expression régulière
length	Cette propriété contient le nombre d'éléments du tableau.
prototype	Cette propriété permet d'ajouter des propriétés personnalisées à l'objet.

Le tableau suivant décrit les méthodes de l'objet Array :

Méthode	description
<code>concat(tab1, tab2[, tab3, ...])</code>	Cette méthode permet de concaténer plusieurs tableaux, c'est-à-dire de créer un tableau à partir des différents tableaux passés en paramètre.
<code>join(tableau)</code> ou <code>Tableau.join()</code>	Cette méthode renvoie une chaîne de caractères contenant tous les éléments du tableau.
<code>pop(tableau)</code> ou <code>Tableau.pop()</code>	Cette méthode supprime le dernier élément du tableau et retourne sa valeur.
<code>Tableau.push(valeur1[, valeur2, ...])</code>	Cette méthode ajoute un ou plusieurs éléments au tableau.
<code>Tableau.reverse()</code>	Cette méthode inverse l'ordre des éléments du tableau.
<code>Tableau.shift()</code>	Cette méthode supprime le premier élément du tableau.
<code>Tableau.slice()</code>	Cette méthode renvoie un tableau contenant une partie (extraction)

	des éléments d'un tableau.
Tableau.splice()	Cette méthode ajoute/retire des éléments d'un tableau.
Tableau.sort()	Cette méthode permet de trier les éléments d'un tableau.
Tableau.unshift(valeur1[, valeur2, ...])	Cette méthode renvoie le code source qui a permis de créer l'objet Array.
Tableau.toString()	Cette méthode renvoie la chaîne de caractères correspond à l'instruction qui a permis de créer l'objet Array.
Tableau.unshift()	Cette méthode permet d'ajouter un ou plusieurs éléments au début du tableau.
Tableau.valueOf	Cette méthode retourne tout simplement la valeur de l'objet Array auquel elle fait référence.

V.11.2. L'objet date

L'objet Date permet de travailler avec toutes les variables qui concernent les dates et la gestion du temps. Il s'agit d'un objet inclus de façon native dans Javascript, et que l'on peut toujours utiliser.

La syntaxe pour créer un objet-date peut être une des suivantes:

1. Nom_de_l_objet = new Date()
cette syntaxe permet de stocker la date et l'heure actuelle
2. Nom_de_l_objet = new Date("jour, mois date année heures:minutes:secondes")
les paramètres sont une chaîne de caractères suivant scrupuleusement la notation ci-dessus
3. Nom_de_l_objet = new Date(année, mois, jour)
les paramètres sont trois entiers séparés par des virgules.
Les paramètres omis sont mis à zéro par défaut
4. Nom_de_l_objet = new Date(année, mois, jour, heures, minutes, secondes[, millisecondes])
les paramètres sont six entiers séparés par des virgules.
Les paramètres omis sont mis à zéro par défaut

La date est stockée dans une variable sous la forme d'une chaîne qui contient le jour, le mois, l'année, l'heure, les minutes, et les secondes. Il est donc difficile d'accéder à un seul élément d'un objet date avec les fonctions de manipulation de chaînes de caractères, étant donné que chacun des éléments peut avoir une taille variable. Heureusement, les méthodes de l'objet

Date fournissent un moyen simple d'accéder à un seul élément, ou bien de le modifier. Leur syntaxe est la suivante :

La syntaxe : *Objet_Date.Methode()*

Les méthodes dont le nom commence par le radical get (mot anglais qui signifie récupérer) permettent de renvoyer une partie de l'objet Date :

Méthode	Description	Type de valeurs retournées
getDate()	Permet de récupérer la valeur du jour du mois	L'objet retourné est un entier (entre 1 et 31) qui correspond au jour du mois:
getDay()	Permet de récupérer la valeur du jour de la semaine pour la date spécifiée	L'objet retourné est un entier qui correspond au jour de la semaine: <ul style="list-style-type: none"> • 0: dimanche • 1: lundi ...
getFullYear()	Permet de récupérer la valeur de l'année sur 4 chiffres pour la date passée en paramètre	L'objet retourné est un entier qui correspond à l'année (XXXX)
getHours()	Permet de récupérer la valeur de l'heure	L'objet retourné est un entier (entre 0 et 23) qui correspond à l'objet Date.
getMilliseconds()	Permet de récupérer le nombre de millisecondes	L'objet retourné est un entier (entre 0 et 999) qui correspond aux millisecondes de l'objet passé en paramètre.
getMinutes()	Permet de récupérer la valeur des minutes	L'objet retourné est un entier (entre 0 et 59) qui correspond aux minutes de l'objet Date.
getMonth()	Permet de récupérer le numéro du mois	L'objet retourné est un entier (entre 0 et 11) qui correspond au mois : <ul style="list-style-type: none"> • 0: janvier • 1: février ...
getSeconds()	Permet de récupérer le nombre de secondes	L'objet retourné est un entier (entre 0 et 59) qui correspond aux secondes de l'objet passé en paramètre.

getTime()	Permet de récupérer le nombre de millisecondes depuis le 1er janvier 1970	L'objet retourné est un entier. Cette méthode est très utile pour convertir des dates, soustraire ou ajouter deux dates, etc.
getTimezoneOffset()	Retourne la différence entre l'heure locale et l'heure GMT (Greenwich Mean Time)	L'objet retourné est un entier, il représente le nombre de minutes de décalage
getFullYear()	Permet de récupérer la valeur de l'année sur 2 chiffres pour l'objet Date.	L'objet retourné est un entier qui correspond à l'année (XX)

Les méthodes dont le nom commence par le radical *set* (mot anglais qui signifie *régler*) permettent de modifier une valeur :

Méthode	Description	Type de valeur en paramètre
setDate(X)	Permet de fixer la valeur du jour du mois	Le paramètre est un entier (entre 1 et 31) qui correspond au jour du mois
setDay(X)	Permet de fixer la valeur du jour de la semaine	Le paramètre est un entier qui correspond au jour de la semaine: <ul style="list-style-type: none"> • 0: dimanche • 1: lundi ...
setHours(X)	Permet de fixer la valeur de l'heure	Le paramètre est un entier (entre 0 et 23) qui correspond à l'heure
setMinutes(X)	Permet de fixer la valeur des minutes	Le paramètre est un entier (entre 0 et 59) qui correspond aux minutes
setMonth(X)	Permet de fixer le numéro du mois	Le paramètre est un entier (entre 0 et 11) qui correspond au mois: <ul style="list-style-type: none"> • 0: janvier • 1: février ...
setTime(X)	Permet d'assigner la date	Le paramètre est un entier représentant le nombre de millisecondes depuis le 1er janvier 1970

V.11.3. L'objet String

Les méthodes de l'objet string permettent de récupérer une portion d'une chaîne de caractère, ou bien de la modifier.

Le tableau suivant décrit les méthodes de l'objet String :

Méthode	Description
Chaine.anchor("nom_a_donner");	Transforme le texte Chaine en ancrage HTML.
Chaine.big()	Augmente la taille de la police.
Chaine.blink()	Transforme la chaîne en texte clignotant.
Chaine.bold()	Met le texte en gras (balise).
Chaine.charAt(position)	Retourne le caractère situé à la position donnée en paramètre
Chaine.charCodeAt(position)	Renvoie le code Unicode du caractère situé à la position donnée en paramètre
concat(chaîne1, chaîne2[, ...])	Permet de concaténer les chaînes passées en paramètre, c'est-à-dire de les joindre bout à bout.
Chaine.fixed()	Transforme la Chaine en caractères de police fixe (balise <TT>)
Chaine.fontcolor(couleur)	Modifie la couleur du texte (admet comme argument la couleur en hexadécimal ou en valeur littérale)
Chaine.fontSize(Size)	Modifie la taille de la police, en affectant la valeur passée en paramètre
Chaine.fromCharCode(code1[, code2, ..])	Renvoie une chaîne de caractères composée de caractères correspondant au(x) code(s) Unicode donné(s) en paramètre.
Chaine.indexOf(sous-chaîne, position)	Retourne la position d'une sous-chaîne (lettre ou groupe de lettres) dans une chaîne de caractère, en effectuant la recherche de gauche à droite, à partir de la position spécifiée en paramètre.
Chaine italics()	Transforme le texte en italique (balise <I>)
Chaine.lastIndexOf(sous-chaîne, position)	La méthode est similaire à indexOf(), à la différence que la recherche se fait de droite à gauche: Retourne la position d'une sous-chaîne (lettre ou groupe de lettres) dans une chaîne de caractère, en effectuant la recherche

	de droite à gauche, à partir de la position spécifiée en paramètre.
Chaine.link(URL)	Transforme le texte en hypertexte (balise <A href>)
Chaine.small()	Diminue la taille de la police
Chaine.strike()	Transforme le texte en texte barré (balise <strike>)
Chaine.sub()	Transforme le texte en indice (balise <sub>)
Chaine.substr(position1, longueur)	La méthode retourne une sous-chaîne commençant à l'index dont la position est donnée en argument et de la longueur donnée en paramètre.
Chaine.substring(position1, position2)	La méthode retourne la sous-chaîne (lettre ou groupe de lettres) comprise entre les positions 1 et 2 données en paramètre.
Chaine.sup()	Transforme le texte en exposant (balise <sup>).
Chaine.toLowerCase()	Convertit tous les caractères d'une chaîne en minuscule.
Chaine.toSource()	Renvoie le code source de création de l'objet.
Chaine.toUpperCase()	Convertit tous les caractères d'une chaîne en majuscule.
Chaine.valueOf()	Renvoie la valeur de l'objet String.

V.12. Les objets du navigateur

Lorsqu'on ouvre une page Web, le navigateur crée des objets prédéfinis correspondant à la page Web, à l'état du navigateur.

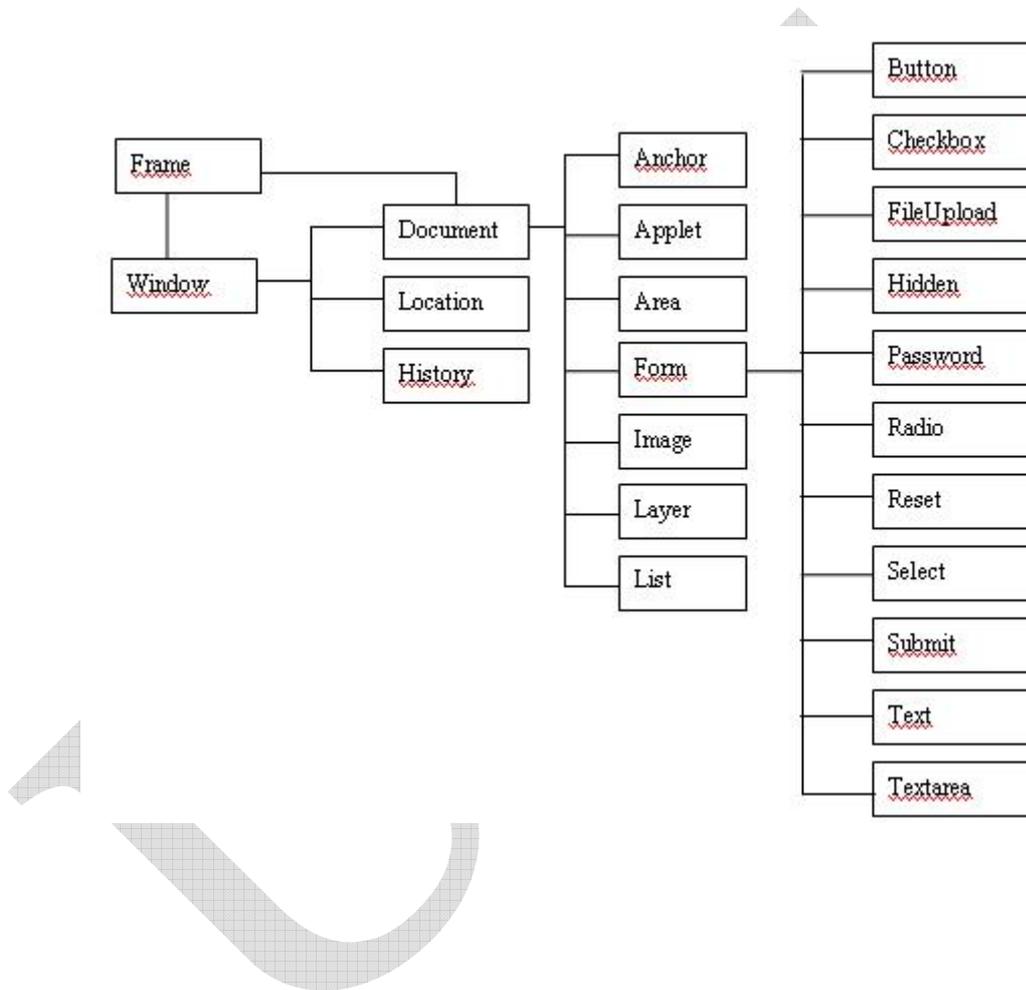
Les objets de base du navigateur sont les suivants :

- navigator : qui contient des informations sur le navigateur de celui qui visite la page.
- window : c'est l'objet où s'affiche la page, il contient donc des propriétés concernant la fenêtre elle-même mais aussi tous les objets-fils contenus dans celle-ci.
- location : contient des informations relatives à l'adresse de la page à l'écran.

- History : c'est l'historique, c'est-à-dire la liste de liens qui ont été visités précédemment.
- document : il contient les propriétés sur le contenu du document (couleur d'arrière plan, titre, ...)

Ces objets sont largement dépendants du contenu de la page.

Mis à part des objets tels que navigator qui sont figés pour un utilisateur donné, le contenu des autres objets variera suivant le contenu de la page.



Pour accéder à un objet du navigateur, il faut parcourir la hiérarchie du navigateur, en partant du sommet (l'objet *window*), puis en parcourant tous les maillons jusqu'à atteindre l'objet désiré. La syntaxe est *window.objet1.objet2.objet3.objet_vise* (ici il y a trois objets intermédiaire *objet1 objet2 objet3* mais ce nombre peut varier de 0 à un très grand nombre d'objets, suivant l'imbrication de vos objets dans la page).

Module : Multimédia et développement Web

Chapitre 6 : PHP

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Attention !

Ce produit pédagogique numérisé est la propriété exclusive de l'UVT. Il est strictement interdit de la reproduire à des fins commerciales. Seul le téléchargement ou impression pour un usage personnel (1 copie par utilisateur) est permis.

- Etablir des connexions avec MySQL.

UNIVERSITY

VI.1. Introduction

PHP (Personal Home Page Hypertext Preprocessor) est un langage de scripts qui s'intègre aux pages HTML et qui permet de réaliser des pages dynamiques.

- Il s'exécute sur le serveur et permet d'accéder facilement aux bases de données.
- C'est un produit "Open Source" c'est-à-dire que le code est accessible à tout développeur.
- Il est gratuit. Combiné au système d'exploitation Linux, au serveur Apache et à la base de données MySQL (eux-mêmes gratuits), il permet de créer des sites Web à des coûts très réduits.

PHP comporte plus de 500 fonctions. Il est fourni avec des bibliothèques offrant des fonctionnalités diverses :

- accès aux bases de données,
- fonctions d'images,
- sockets,
- protocoles Internet divers...

VI.2. Historique

PHP a été créé en 1994 par Rasmus Lerdorf pour les besoins des pages Web personnelles. A l'époque, PHP signifiait Personal Home Page.

C'est un langage incrusté au HTML et interprété (PHP3) ou compilé (PHP4) côté serveur. Il dérive du C et du Perl dont il reprend sa syntaxe. Il est extensible grâce à de nombreux modules et son code source est ouvert. Comme il supporte tous les standards du Web et qu'il est gratuit, il s'est rapidement répandu sur Internet.

En 1997, PHP devient un projet collectif et son interpréteur est réécrit par Zeev Suraski et Andi Gutmans pour donner la version 3 qui s'appelle désormais PHP : Hypertext Preprocessor.

VI.3. Principe de fonctionnement

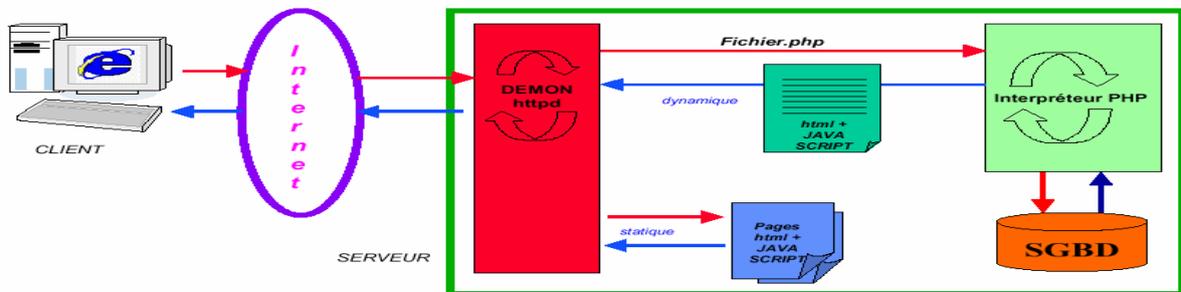


Fig. 6.1. : Le serveur PHP

La figure 4.1. montre lorsqu'une requête HTTP est soumise au serveur Web pour une page dont l'extension est «.php», comme pour un fichier HTML, le serveur commence par rechercher dans son arborescence le fichier d'extension «.php». Il va ensuite passer la main à un sous processus (une dll -Dynamic Link Library- bien particulière) qui va interpréter le script PHP et produire dynamiquement du code HTML. Ce code HTML est alors envoyé au travers du réseau au navigateur client. De plus, aucune ligne de code PHP n'apparaît côté client dans la mesure où tout le code a été interprété.

VI.4. Intégration d'un script PHP dans une page HTML

Un script PHP peut comprendre à la fois du code PHP et du code HTML, non interprété. On doit donc encadrer les parties comportant le code PHP entre 2 balises `<? et ?>`. Le reste de la page n'est pas interprété.

```
<html>< head>< title>
```

```
<? $titrepage = "Mon premier script PHP";
```

```
echo $titrepage; ?>
```

```
</ title></ head>< body>
```

```
<h1><? echo $titrepage ?></ h1>  
  
<? echo " <b> Hello, World ! </ b>"; ?>  
  
</ body></ html>
```

Note: La balise `<? php` est équivalente à `<?>`. On peut également utiliser les balises `<script language=" php">` et `</ script>`. Enfin, pour la programmation ASP, les balises `<% et %>` sont également reconnues.

VI.5. Les variables

VI.5.1. Visibilité et affectation

PHP n'est pas un langage fortement structuré, il ne contient donc pas de partie déclarative clairement définie. Pour définir une variable, il suffit de l'initialiser.

Les variables sont précédées du signe \$, quelque soit leurs types. Ainsi pour déclarer une variable **var** :

```
$var= 1;
```

La variable **\$var** est alors définie et vaut 1. Elle devient immédiatement accessible et ce jusqu'à la fin du script.

VI.5.2. Type de variables

Les variables PHP sont à typage faible. C'est PHP qui décide de son type lors de l'affectation. Il existe six types de données :

- Entier (*int, integer*)
- Décimal (*real, float, double*)
- Chaîne de caractères (*string*)
- Tableau (*array*)
- Objet (*object*)
- Booléen (*boolean, uniquement PHP4*)

Il est parfois utile de forcer le type d'une variable. On utilise la fonction **settype** ou bien les opérateurs de casting (int), (string).

settype renvoie vrai si la conversion a fonctionné, faux sinon.

```
$a= 3. 1415;
```

```
$result= settype( $a, "integer" ); // => $a = 3 , $result = 3
```

VI.6. Les constantes

L'utilisateur peut définir des constantes dont la valeur est fixée une fois pour toute. Les constantes ne portent pas le symbole \$ (dollars) en début d'identificateur et ne sont pas modifiables.

define("var",valeur) : définit la constante **var** (sans \$) de valeur **valeur**

VI.7. Les opérateurs

VI.7.1. Opérateurs arithmétiques

Les opérateurs arithmétiques sont :

+ (addition), - (soustraction), * (multiplié), / (divisé), % (modulo), ++ (incrément), --(décrément).

VI.7.2. Opérateurs d'assignement

Les opérateurs d'assignement sont :

= (affectation), *= ($\$x*=\y équivalent à $\$x=\$x*\$y$), /=, +=, -=, %=

VI.7.3. Opérateurs logiques

Les opérateurs logiques sont :

and, **&&** (et), **or**, **||** (ou), **xor** (ou exclusif), **!** (non)

Les variables booléennes

Elles prennent pour valeurs **TRUE** (vrai) et **FALSE** (faux). Une valeur entière nulle est automatiquement considérée comme **FALSE**. Tout comme une chaîne de caractères vide `""`. Ou encore comme les chaînes `"0"` et `'0'` ou même l'entier `0` est considérée comme **FALSE**.

Exemple :

```
if(0) echo 1; // faux
if("") echo 2; // faux
if("0") echo 3; // faux
if("00") echo 4;
if('0') echo 5; // faux
if('00') echo 6;
if(" ") echo 7;
```

Cet exemple affiche **467**. Donc l'espace ou la chaîne `"00"` ne sont pas considérés comme **FALSE**.

VI.7.4. Opérateurs de comparaison

Les opérateurs de comparaison sont :

`==` (égalité), `<` (inférieur strict), `<=` (inférieur large), `>`, `>=`, `!=` (différence)

VI.8. L'affichage du texte

Les fonctions d'affichage :

echo() : écriture dans le navigateur

print() : écriture dans le navigateur

printf([\$format, \$arg1, \$arg2]) : écriture formatée comme en C, i.e. la chaîne de caractère est constante et contient le format d'affichage des variables passées en argument

Exemples :

```
echo "Bonjour $name";
```

```
print("Bonjour $name");
```

```
printf("Bonjour %s", $name);
```

VI.9. Exemples de fonctions Mathématiques

Il existe plusieurs fonctions mathématiques. Par exemple :

abs(\$x) : valeur absolue.

ceil(\$x) : arrondi supérieur.

floor(\$x) : arrondi inférieur.

pow(\$x,\$y) : x exposant y.

round(\$x,\$i) : arrondi de x à la ième décimale.

max(\$a, \$b, \$c ...) : retourne l'argument de valeur maximum.

pi() : retourne la valeur de Pi.

Et aussi : **cos, sin, tan, exp, log, min, pi, sqrt.**

- *Quelques constantes :*

M_PI : valeur de pi (3.14159265358979323846).

M_E : valeur de e (2.7182818284590452354).

- *Nombres aléatoires :*

rand([\$x,\$y]) ou **mt_rand([\$x,\$y])** : valeur entière aléatoire entre 0 et RAND_MAX si x et y ne sont pas définis, entre x et RAND_MAX si seul x est défini, entre x et y si ces deux paramètres sont définis.

srand() ou **mt_srand()** : initialisation du générateur aléatoire.

getrandmax() ou **mt_getrandmax()** : retourne la valeur du plus grand entier pouvant être généré.

Sachant que **mt_rand()** est meilleure que **rand()** car elle est plus rapide et plus sûre puisque l'algorithme utilisé se base sur la cryptographie.

- *Formatage d'un nombre :*

number_format (\$nbr,\$dec,[\$a,\$b]) : retourne une chaîne de caractères représentant le nombre **\$nbr** avec **\$dec** décimales après formatage. La chaîne **\$a** représente le symbole faisant office de virgule et **\$b** le séparateur de milliers.

Par défaut, le formatage est anglophone : \$a = "." et \$b = ",".

Très utile pour représenter les nombres élevés au format francophone.

Exemples :

```
number_format (1000000.3333); // affiche 1,000,000
```

```
number_format (1000000.3333,2); // affiche 1,000,000.33  
number_format (1000000.3333,2,"","."); // affiche 1.000.000,33
```

VI.10. Les chaînes de caractère

Une variable chaîne de caractères n'est pas limitée en nombre de caractères. Elle est toujours délimitée par des simples quotes ou des doubles quotes.

Exemples :

```
$nom = "Tounsi";
```

```
$prenom = 'Mohamed';
```

Les doubles quotes permettent l'évaluation des variables et caractères spéciaux contenus dans la chaîne (comme en C ou en Shell) alors que les simples ne le permettent pas.

Exemples :

```
echo "Nom: $nom"; // affiche Nom: Tounsi
```

```
echo 'Nom: $nom'; // affiche Nom: $nom
```

Quelques caractères spéciaux : `\n` (nouvelle ligne), `\r` (retour à la ligne), `\t` (tabulation horizontale), `\\` (antislash), `\$` (caractère dollars), `\"` (double quote).

Exemple :

```
echo "Hello Word !\n";
```

- *Opérateur de concaténation de chaînes :*

. (point)

- *Quelques fonctions :*

strlen(\$str) : retourne le nombre de caractères d'une chaîne.

strtolower(\$str) : conversion en minuscules.

strtoupper(\$str) : conversion en majuscules.

trim(\$str) : suppression des espaces de début et de fin de chaîne.

substr(\$str,\$i,\$j) : retourne une sous chaîne de **\$str** de taille **\$j** et débutant à la position **\$i**.

strnatcmp(\$str1,\$str2) : comparaison de 2 chaînes.

addslashes(\$str) : désécialise les caractères spéciaux (', ", \).

ord(\$char) : retourne la valeur ASCII du caractère **\$char**.

VI.11. Les expressions régulières

Les expressions régulières sont un outil puissant pour la recherche de motifs dans une chaîne de caractères.

VI.11.1. Fonctions :

eregi(\$motif, \$str) : teste l'existence du motif **\$motif** dans la chaîne **\$str**.

eregi_replace(\$motif, \$newstr, \$str) : remplace les occurrences de **\$motif** dans **\$str** par la chaîne **\$newstr**.

split(\$motif, \$str) : retourne un tableau des sous-chaînes de **\$str** délimitées par les occurrences de **\$motif**.

NB :

Les fonctions **eregi**, **eregi_replace** et **spliti** ne différencient pas les majuscules et les minuscules.

Exemple :

```
if (eregi("Sfax", $adresse))      echo "Vous habitez Sfax. ";
```

Les motifs peuvent être très complexes et contenir des caractères spéciaux.

VI.11.2. Les caractères spéciaux :

[abcdef] : intervalle de caractères, teste si l'un d'eux est présent.

[a-f] : plage de caractères : teste la présence de tous les caractères minuscules entre 'a' et 'f'

[^0-9] : exclusion des caractères de '0' à '9'.

\^ : recherche du caractère '^' que l'on déspecialise par l'antislash \

. : remplace un caractère.

? : rend facultatif le caractère qu'il précède.

+ : indique que le caractère précédent peut apparaître une ou plusieurs fois.

***** : pareil que + Mais le caractère précédent peut ne pas apparaître du tout.

{i,j} : retrouve une chaîne contenant entre au minimum **i** et au maximum **j** fois le motif qu'il précède.

{i,} : idem mais pas de limite maximum.

{i} : retrouve une séquence d'exactly **i** fois le motif qu'il précède.

^ : le motif suivant doit apparaître en début de chaîne.

\$: le motif suivant doit apparaître en fin de chaîne.

Exemples de motifs :

"[A-Z]" : recherche toutes les majuscules

"[a-zA-Z]" : recherche toutes les lettres de l'alphabet minuscules ou majuscules

"[^aeuyio]" : exclu les voyelles

"^Le " : toute chaîne commençant par le mot "Le " suivi d'un espace

"\$\.com" : toute chaîne se terminant par ".com" (déspecialise le point)

Exemples :

```
if (!ereg("^(.+)\.(.+)$", $email)) echo "Email invalide";  
$email = eregi_replace("@", "-nospam@", $email);
```

Ce dernier exemple remplace "moi@ici.com" en "moi-nospam@ici.com".

VI.12. Dates et heures

Les fonctions de dates et heures sont incontournables sur Internet et sont indispensables pour la conversion en français des dates fournies par la base de données MySQL qui les code au format anglophone (YYYY-DD-MM hh:mm:ss).

Quelques fonctions :

date("\$format") : retourne une chaîne de caractères contenant la date et/ou l'heure locale au format spécifié.

getdate() : retourne un tableau associatif contenant la date et l'heure.

checkdate(\$month, \$day, \$year) : vérifie la validité d'une date.

Exemple 1 :

```
echo date("Y-m-d H:i:s");
```

```
/* affiche la date au format MySQL : '2002-03-31 22:30:29' */
```

Exemple 2 :

```
if(checkdate(03, 31,2005))
    echo "La date est valide";
```

Exemple 3 :

```
$aujourd'hui = getdate();
$mois = $aujourd'hui['mon'];
$jour = $aujourd'hui['mday'];
$annee = $aujourd'hui['year'];
echo "$jour/$mois/$annee";
```

- Les formats pour date :

d Jour du mois sur deux chiffres [01..31]	h Heure au format 12h [01..12]
j Jour du mois sans les zéros initiaux	au format 24h sans les zéros initiaux
l Jour de la semaine textuel en version longue et en anglais	g Heure au format 12h sans les zéros initiaux
D Jour de la semaine textuel en trois lettres et en anglais	H Heure au format 24h [00..23]
w Jour de la semaine numérique [0..6] (0: dimanche)	G Heure
z Jour de l'année [0..365]	i Minutes [00..59]
m Mois de l'année sur deux chiffres [01..12]	s Secondes [00..59]
n Mois sans les zéros initiaux	a am ou pm
F Mois textuel en version longue et en	A AM ou PM
	L Booléen pour savoir si l'année est

anglais	bissextile (1) ou pas (0)
M Mois textuel en trois lettres	S Suffixe ordinal anglais d'un nombre (ex: <i>nd</i> pour 2)
Y Année sur 4 chiffres	t Nombre de jour dans le mois donné [28..31]
y Année sur 2 chiffres	U Secondes depuis une époque
	Z Décalage horaire en secondes [-43200..43200]

VI.13. Structures de contrôle

VI.13.1. Structures conditionnelles

(Même syntaxe qu'en langage C)

```
if( ... ) {
```

```
    ...
```

```
} elseif {
```

```
    ...
```

```
} else {
```

```
    ...
```

```
}
```

```
switch( ... ) {
```

```
    case ... : { ... } break
```

```
    ...
```

```
    default : { ... }  
}
```

V.13.2. Structures de boucle

(Même syntaxe qu'en langage C)

```
for( ... ; ... ; ... ) {
```

```
    ...
```

```
}
```

```
while( ... ) {
```

```
    ...
```

```
}
```

```
do {
```

```
    ...
```

```
} while( ... );
```

L'instruction **break** permet de quitter prématurément une boucle.

Exemple :

```
while($nbr = $stab[$i++]) {  
    echo $nbr. "<br />";  
    if($nbr == 0)
```

```
        break;
    }
}
```

L'instruction **continue** permet d'éluder les instructions suivantes de l'itération courante de la boucle pour passer à la suivante.

Exemple :

```
for($i=1; $i<=10; $i++) {
    if($tab[$i] == 0)
        continue;
    echo $tab[$i];
}
```

Pour stopper un script, il existe deux fonctions :

die arrête un script et affiche un message d'erreur dans le navigateur..

exit l'arrête aussi mais sans afficher de message d'erreur.

Ces fonctions stoppent tout le script, pas seulement le bloc en cours.

VI.14. Les tableaux

VI.14.1. Initialiser un tableau

Une variable tableau est de type **array**. Un tableau accepte des éléments de tout type. Les éléments d'un tableau peuvent être de types différents et sont séparés d'une virgule.

Un tableau peut être initialisé avec la syntaxe **array**.

Exemple :

```
$tab_colors = array('red', 'yellow', 'blue', 'white');
```

```
$tab = array('Ali', 2002, 20.5, $name);
```

Mais il peut aussi être initialisé au fur et à mesure.

Exemples :

```
$prenoms[ ] = "Ali";           $villes[0] = "Sfax";
```

```
$prenoms[ ] = "Salah";        $villes[1] = "Sousse";
```

```
$prenoms[ ] = "Fatma";        $villes[2] = "Tunis ";
```

L'appel d'un élément du tableau se fait à partir de son indice (dont l'origine est zéro comme en C).

Exemple :

```
echo $tab[10]; // pour afficher le 11ème élément
```

VI.14.2. Parcours d'un tableau

Exemple :

```
$tab = array('Ali', 'Salah', 'Fatma');
```

```
$i=0;
```

```
while($i <= count($tab)) { // count() retourne le nombre d'éléments
```

```
    echo $tab[$i] '\n';
```

```
$i++;  
}
```

VI.14.3. Quelques fonctions

count(\$tab), sizeof : retournent le nombre d'éléments du tableau.

in_array(\$var,\$tab) : dit si la valeur de **\$var** existe dans le tableau **\$tab**.

list(\$var1,\$var2...) : transforme une liste de variables en tableau.

range(\$i,\$j) : retourne un tableau contenant un intervalle de valeurs.

shuffle(\$tab) : mélange les éléments d'un tableau.

sort(\$tab) : trie alphanumérique les éléments du tableau.

rsort(\$tab) : trie alphanumérique inverse les éléments du tableau.

array_merge(\$tab1,\$tab2,\$tab3...) : concatène les tableaux passés en arguments.

array_rand(\$tab) : retourne un élément du tableau au hasard.

VI.14.4. Les tableaux associatifs

A/ Déclaration

Un tableau associatif est un tableau dont l'index est une chaîne de caractères au lieu d'un nombre. On parle aussi de ``hash array" ou ``hash". Il se déclare comme un tableau traditionnel, la distinction se fait lors de l'affectation.

Exemples :

```
$personne["Nom"] = " Tounsi ";  
$personne["Prénom"] = " Mohamed ";  
Ou aussi : $personne = array("Nom" => " Tounsi ", "Prénom" => " Mohamed ");
```

Remarques :

- L'élément d'un tableau peut être un autre tableau.
- Les tableaux associatifs permettent de préserver une structure de données.

B/ Parcours d'un tableau associatif :

Exemple 1 :

```
foreach($personne as $elem) {  
    echo $elem;
```

```
}
```

Ici on accède directement aux éléments du tableau sans passer par les clés.

Exemple 2 :

```
foreach($personne as $key => $elem) {  
    echo "$key : $elem";  
}
```

Ici on accède simultanément aux clés et aux éléments.

C/ Quelques fonctions s'appliquant aux tableaux associatifs :

array_count_values(\$tab) : retourne un tableau contenant les valeurs du tableau **\$tab** comme clés et leurs fréquence comme valeur (utile pour évaluer les redondances).

array_keys(\$tab) : retourne un tableau contenant les clés du tableau associatif **\$tab**.

array_values(\$tab) : retourne un tableau contenant les valeurs du tableau associatif **\$tab**.

array_search(\$val,\$tab) : retourne la clé associée à la valeur **\$val**.

reset(\$tab) : place le pointeur sur le premier élément.

current(\$tab) : retourne la valeur de l'élément courant.

next(\$tab) : place le pointeur sur l'élément suivant.

prev(\$tab) : place le pointeur sur l'élément précédant.

each(\$tab) : retourne la paire clé/valeur courante et avance le pointeur.

VI.15. Les fonctions

A l'image de tout langage structuré, en PHP, une fonction est une suite d'instructions qui peut remplir n'importe quelle tâche. Tout code PHP valide figure dans le corps (ou le code) d'une fonction.

Il n'y a pas de distinction fonctions / procédures en PHP.

Les fonctions PHP prennent de 0 à n paramètres. Ces paramètres peuvent être de type quelconque.

Remarque :

Il faut implémenter la fonction en amont de son utilisation, contrairement au langage C. Dans le cas contraire, PHP sort une erreur du type *Call to unsupported or undefined function (fonction) in (file) on line (number)*.

On ne peut pas déclarer le prototype d'une fonction comme par exemple en Pascal.

VI.15.1. Déclaration

La syntaxe de déclaration s'appuie sur le mot clé **function**. Ce mot clé est immédiatement suivi du nom de la fonction par lequel on va l'appeler depuis n'importe quel endroit du code PHP, puis des parenthèses destinées à accueillir les éventuels paramètres.

Exemple :

```
function bonjour() {  
    echo " Bonjour ";  
}  
.....  
bonjour(); // Affiche " Bonjour " à l'écran.
```

Les fonctions peuvent ou non renvoyer un résultat. On utilise l'instruction **return**. La variable retournée peut être de type quelconque. Elle est transmise par copie.

Exemple :

```
function bonjour2() {
```

```
return " Bonjour ";  
}  
  
.....  
echo bonjour2() ; // Affiche " Bonjour " à l'écran.
```

Le mode de fonctionnement est sensiblement différent, la fonction *bonjour* affiche directement le mot " Bonjour " à l'écran, alors que s'affiche le résultat de *bonjour2*.

Par défaut, les variables globales ne sont pas connues à l'intérieur du corps d'une fonction. On peut cependant y accéder à l'aide du mot- clé **global**.

Exemple :

```
$debug_mode= 1; // variable globale  
  
....  
function mafonction() {  
global $debug_mode;  
if( $debug_mode )  
echo "[ DEBUG] in fonction mafonction()";  
  
....  
}
```

Une autre solution est d'utiliser le tableau associatif **\$GLOBALS**, qui contient toutes les variables globales déclarées à un instant T :

`$GLOBALS['debug_ mode']` équivaut à `$debug_ mode`.

VI.15.2. Le passage des paramètres par valeur

Afin de passer des paramètres à la fonction, il suffit de les insérer à l'intérieur des parenthèses prévues à cet effet.

Exemple :

```
function bonjour($ prénom, $nom) {  
    $chaîne = " Bonjour $prénom $nom " ;  
  
    // On construit la phrase complète dans la variable locale $chaîne.  
  
    return $chaîne ;  
  
    // On renvoie la valeur de $chaîne comme résultat de la fonction.  
}  
.....  
echo bonjour(" Mohamed" , "TOUNSI") ;  
  
// Affiche " Bonjour Mohamed TOUNSI " à l'écran.
```

VI.15.3. Le passage des paramètres par référence

Par défaut, les paramètres sont transmis par **copie**, c'est à dire que la fonction possède une copie locale de la variable envoyée. Avec la méthode du passage des paramètres par référence, on passe à la fonction l'adresse mémoire d'une variable existante. Cela se fait en précédant de **&** le nom du paramètre. Cela permet de modifier ce paramètre dans la fonction.

Exemple :

```
function bonjour(&$ phrase, $prénom, $nom) {  
    $phrase = " Bonjour $prénom $nom " ;  
  
}  
.....  
$chaîne = " " ;  
  
bonjour($ chaîne, "Mohamed" , "TOUNSI") ;  
  
echo $chaîne ; // Affiche " Bonjour Mohamed TOUNSI " à l'écran.
```

VI.15.4. Le passage des paramètres par défaut

Les paramètres optionnels sont autorisés : il suffit de leur affecter une **valeur par défaut**.

Exemple :

```
function mafonction( $param1 = "inconnu", $param2="" ) {  
    echo "param1=$ param1 param2=$ param2\n";  
}  
...  
mafonction( "toto", "titi" ); // => "param1= toto param2= titi"  
mafonction( "toto" ); // => "param1= toto param2="  
mafonction(); // => "param1= inconnu param2="
```

VI.16. La fonction Include

Cette fonction permet d'inclure le contenu d'une page PHP dans une autre page PHP. Cela peut être utile par exemple pour insérer un menu de navigation dans toutes les pages du site.

On appelle la fonction Include à l'endroit où l'on souhaite que la page incluse apparaisse, de la façon suivante:

```
<? include("page_incluse.php"); ?>
```

Si l'on veut qu'elle apparaisse tout en haut (par exemple si toutes les pages ont le même design en haut), ce sera la première instruction. Ensuite on peut constituer le reste de la page.

Remarque:

La fonction **require** fait la même chose, mais elle arrête le script en cas d'erreur, contrairement à la fonction include.

VI.17. Des fonctions utiles

- **La fonction `isset($var)`** : Pour tester l'existence d'un élément, on utilise la fonction `isset()`

Exemple :

```
if( isset( $calories["pommes"] ) ) {  
    echo "une pomme contient ", $calories["pommes"] , " calories\n";  
}else{  
    echo "pas de calories définies pour la pomme\n";  
}
```

- **La fonction `unset($var)`** : Pour détruire une variable.

- **La fonction `empty($var)`** : La fonction `empty()` permet de contrôler si un champs est vide.

Exemple :

```
<?  
if(empty($titre))  
{  
print("<center>Le '<b>Titre</b>' est vide !</center>");  
exit();  
}  
?>
```

Si la variable `$titre` est vide alors on affiche le message : 'Le titre est vide' et on arrête l'exécution du reste du code avec la commande `exit()`.

- **gettype(\$var)** : Pour retourner le type de la variable.

- **is_long(), is_double(), is_string(), is_array(), is_object(), is_bool(), is_float(), is_numeric(), is_integer(), is_int()** : Pour tester le type de la variable.

VI.18. Les fichiers

PHP fournit plusieurs fonctions qui permettent de prendre en charge l'accès au système de fichiers du système d'exploitation du serveur.

Voici les opérations élémentaires sur les fichiers en PHP :

copy(\$ source, \$destination) : Copie d'un fichier.

\$fp= fopen(" filename", \$mode) : Ouvre un fichier et retourne un "id" de fichier.

fclose(\$ fp) : Ferme un fichier ouvert.

rename(" ancien", "nouveau") : Renomme un fichier.

fwrite(\$ fp, \$str) : Écrit la chaîne de caractères \$str.

fputs(\$ fp, \$str) : Correspond à fwrite().

readfile("filename") : Lit un fichier et retourne son contenu.

fgets(\$ fp, \$maxlength) : Lit une ligne d'un fichier.

fread(\$ fp, \$length) : Lit un nombre donné d'octets à partir d'un fichier.

L'accès à un fichier se fait toujours par un **identificateur** de fichier. Cet "id" est créé avec la fonction **fopen()** et, est requis comme paramètre par la plupart des autres fonctions de fichiers en PHP .

Exemple :

```
$path="/usr/local/apache/htdocs/donnees.txt";
```

```
$mode=" w";  
  
if ($ fp= fopen($ path, $mode) )  
{ echo "Le fichier a été ouvert"; }  
  
else  
  
echo "Fichier impossible à ouvrir";  
  
if ( close($ fp) )  
  
echo " et a été refermé";  
  
?>
```

VI.19. Traitement des formulaires en PHP

Nous allons étudier un exemple concret (voir Fig. 4.2.) en commençant par définir les objectifs de notre formulaire. Admettons que nous devons élaborer un formulaire assurant la saisie du nom, prénom et de l'adresse d'un visiteur.



The image shows a web form titled "Saisie" in a black header bar. Below the header, there are three input fields stacked vertically, labeled "Nom", "Prénom", and "Adresse email" on the left. To the right of each label is a white rectangular input box with a blue border. Below the "Adresse email" input box is a button labeled "Envoyer" with a blue border and a light blue gradient background.

Fig.6.2. : Un formulaire PHP

Le script Html simplifié de cette page est comme suit :

```

1 <html>
2 <head>
3 <title>Document sans titre</title>
4 </head>
5 <body>
6 <form name="form1" method="post" action="script.php">
7     Nom           :<input type="text" name="nom"><br>
8     Prénom        :<input type="text" name="prenom"><br>
9     Adresse email :<input type="text" name="email"><br>
10    <input type="submit" name="Submit" value="Envoyer"><br>
11 </form>
12 </body>
13 </html>
    
```

Fig.6.3. : Script HTML pour écrire le formulaire précédent

Trois champs seront en saisie obligatoire : nom, prénom et adresse e-mail. Et dans une certaine mesure, nous chercherons à contrôler la validité de l'adresse e-mail saisie.

Pour finir, notre formulaire devra renvoyer un message au client afin de l'inviter à modifier sa saisie si cette dernière est invalide, ou de l'informer que sa saisie a été prise en compte dans le cas contraire.

Lorsqu'on clique sur le bouton envoyer le fichier script.php sera exécuté contenant le script suivant :

```

1 <html>
2 <head>
3 <title>Exemple de test</title>
4 </head>
5 <body>
6 <?
7 $echec="";
8 if ($nom=="") $echec=$echec."-le champ nom est obligatoire <br>";
9 if ($prenom=="") $echec=$echec."-le champ prénom est obligatoire <br>";
10 if ($email=="") $echec=$echec."-le champ email est obligatoire <br>";
11     elseif (!ereg("^(.+)?@(.+)\.(.+)$", $email))
12         $echec=$echec."-le champ email est invalide <br>";
13 if ($echec == "") {
14     $message = "Succès.<br>Votre saisie a été prise en compte.<br>Merci.";
15     print $message;}
16 else { $message="Echec.<br>Corrigez votre saisie.<br>".$echec;
17     print $message;}
18 ?>
19 </body>
20 </html>
    
```

Fig.6.4. Script PHP permettant le contrôle de la saisie des données

VI.20. Accès aux SGBD

En général, la communication entre un programme et une base de données suit le schéma suivant :



Fig.6.5. : La communication entre un programme et une base de données

En programmation PHP, il existe 2 méthodes pour mettre en place cette architecture :

1. accéder nativement à la base par l'intermédiaire de l'API de son middleware associé,
2. passer par ODBC, l'avantage d'ODBC est de proposer une API unifiée quelque soit le SGBD utilisé.

En plus d'ODBC, PHP gère en accès natifs de nombreux SGBD : Oracle, Sybase, Informix, MySQL, Adabas, Empress, FilePro, InterBase, mSQL, PostgreSQL, Solid, SQLServer, Unix Dbm.

VI.21. Connexion avec MySQL

VI.21.1. Présentation de MySQL

MySQL est un système de gestion de base de données (SGBD) libre implémenté avec le langage de requête SQL. Le serveur de base de données MySQL est très souvent utilisé avec le langage PHP

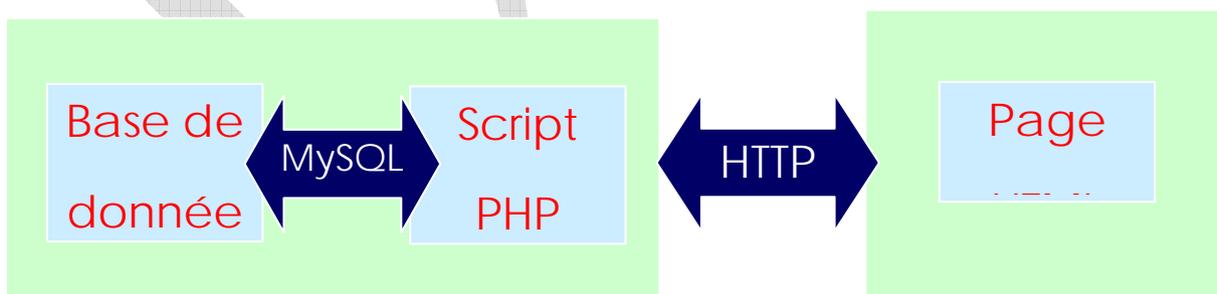


Fig.6.6. : Principe d'interaction avec MySQL

Il existe un outil libre et gratuit développé par la communauté des programmeurs libres : phpMyAdmin qui permet l'administration aisée des bases de données MySQL

avec php. Il est disponible sur : <http://sourceforge.net/projects/phpmyadmin/> et <http://www.phpmyadmin.net>.

Avec MySQL on peut créer plusieurs bases de données sur un serveur. Une base est composée de tables contenant des enregistrements.

VI.21.2. Les fonctions de connexion avec MySQL

Pour se connecter à une base de type MySQL depuis php, il faut spécifier un nom de serveur, un nom d'utilisateur, un mot de passe et un nom de base. Voici quelques fonctions de connexion :

mysql_connect(\$server,\$user,\$password) : permet de se connecter au serveur **\$server** en tant qu'utilisateur **\$user** avec le mot de passe **\$password**, retourne TRUE en cas de succès, FALSE sinon.

mysql_select_db(\$base[\$id]) : permet de choisir la base **\$base**, retourne TRUE en cas de succès, sinon FALSE.

mysql_close([\$id]) : permet de fermer la connexion.

mysql_pconnect : idem que **mysql_connect** sauf que la connexion est persistante, il n'y a donc pas besoin de rouvrir la connexion à chaque script qui travaille sur la même base. Les identifiants de connexion ne sont pas nécessaires si on ne se connecte qu'à une seule base à la fois, ils permettent seulement de lever toute ambiguïté en cas de connexions multiples.

Exemple 1 :

```
if( $id = mysql_connect("localhost","root","iset") ) {  
    if( $id_db = mysql_select_db("miniprojet") ) {  
        echo "Succès de connexion. ";  
        /* code du script ... */  
    }  
}
```

```
    } else {  
        die("Echec de connexion à la base. ");  
    }  
    mysql_close($id);  
} else {  
    die("Echec de connexion au serveur de base de données. ");  
}
```

Exemple 2 :

```
@mysql_connect("localhost","root","iset") or die("Echec de connexion au serveur. ");  
@mysql_select_db("gigabase ") or die("Echec de sélection de la base. ");
```

Le symbole @ permet d'éviter le renvoi de valeur par la fonction qu'il précède.

VI.21.3. Interrogation de la base

Pour envoyer une requête à une base de donnée, il existe la fonction : **mysql_query(\$str)** qui prend pour paramètre une chaîne de caractères qui contient la requête écrite en SQL et retourne un identificateur de résultat ou FALSE si échec.

Les requêtes les plus couramment utilisées sont : **CREATE** (création d'une table), **ALTER** (modification d'une table), **SELECT** (sélection), **INSERT** (insertion), **UPDATE** (mise à jour des données), **DELETE** (suppression des données), etc.

Exemple :

```
$result = mysql_query("SELECT address FROM users WHERE name=\"\$name\"");
```

Cet exemple recherche l'adresse de l'utilisateur portant pour nom la valeur de la chaîne **\$name**. L'identificateur de résultat **\$result** permettra à d'autres fonctions d'extraire ligne par ligne les données retournées par le serveur.

Attention ! Contrairement à Oracle (SQL*PLUS), les requêtes MySQL ne se terminent pas par un point virgule ';' et n'autorisent pas les SELECT imbriqués.

VI.21.4. Extraction des données

Une fois la requête effectuée et l'identificateur de résultat acquis, il ne reste plus qu'à extraire les données retournées par le serveur.

Sous Oracle, l'affichage des résultats d'une requête se fait ligne par ligne, sous MySQL, c'est pareil. Une boucle permettra de recueillir chacune des lignes à partir de l'identifiant de résultat.

Exemple :

```
SQL > SELECT * FROM users;
```

```
ID  NAME  VILLE
```

```
-----
```

```
1 Mohamed Sfax      1ère ligne
2 Ali     Tunis      2ème ligne
3 Salah  Sousse     3ème ligne
```

Une ligne contient (sauf cas particulier) plusieurs valeurs correspondantes aux différents attributs retournés par la requête.

mysql_fetch_row(\$result) : retourne une ligne de résultat sous la forme d'un tableau. Les éléments du tableau étant les valeurs des attributs de la ligne. Retourne FALSE s'il n'y a plus aucune ligne.

Exemple 1 :

```
$requet = "SELECT * FROM users";  
  
if($result = mysql_query($requet)) {      while($ligne = mysql_fetch_row($result)) {  
    $id = $ligne[0];  
    $name = $ligne[1];  
    $address = $ligne[2];  
    echo "$id - $name, $address <br />";  
    }  
} else {  
    echo "Erreur de requête de base de données. ";  
}
```

Ici, on accède aux valeurs de la ligne par leur indice dans le tableau.

mysql_fetch_array(\$result) : retourne un tableau associatif. Les clés étant les noms des attributs et leurs valeurs associées leurs valeurs respectives. Retourne FALSE s'il n'y a plus aucune ligne.

Exemple 2 :

```
$requet = "SELECT * FROM users";  
  
if($result = mysql_query($requet)) {      while($ligne = mysql_fetch_array($result)) {  
    $id = $ligne["id"];  
    $name = $ligne["name"];  
    $address = $ligne["address"];  
}
```

```
        echo "$id - $name, $address <br />";
    }
} else {
    echo "Erreur de requête de base de données. ";
}
```

Ici, on accède aux valeurs de la ligne par l'attribut dans le tableau associatif.

mysql_fetch_object(\$result) : retourne un objet. Les attributs de l'objet correspondent à ceux de la ligne de résultat. Et les valeurs des attributs de l'objet correspondent à ceux de la ligne de résultat. Retourne FALSE s'il n'y a plus aucune ligne.

Exemple 3 :

```
$requet = "SELECT * FROM users";
if($result = mysql_query($requet)) {    while($ligne = mysql_fetch_object($result)) {
    $id = $ligne->id;
    $name = $ligne->name;
    $address = $ligne->address;
    echo "$id - $name, $address <br />";
}
} else {
    echo "Erreur de requête de base de données. ";
}
```

Ici, on accède aux valeurs par leur attribut dans l'objet.

Quelques fonctions supplémentaires très utiles :

mysql_free_result(\$result) : efface de la mémoire du serveur les lignes de résultat de la requête identifiées par **\$requet**. Très utile pour améliorer les performances du serveur.

mysql_insert_id([\$id]) : retourne l'identifiant d'un attribut clé primaire AUTO_INCREMENT de la dernière insertion.

mysql_num_fields(\$result) : retourne le nombre d'attributs du résultats.

mysql_num_rows(\$result) : retourne le nombre de lignes du résultats.

VI.21.5. Exemple : Insertion des données à partir de formulaire

On commence par créer la table client, puis on crée le formulaire correspondant :

```
<html>
<form method="POST" action="ajout.php">
<center>
<input type="text" name="nom" size="20" value="nom" maxlength="35"> <input
type="text" name="prenom" size="20" value="prenom" maxlength="35"><br>
<input type="text" name="email" size="20" value="email" maxlength="70"> <input
type="text" name="icq" size="20" value="icq" maxlength="11"><br>
<input type="text" name="titre" size="20" value="titre du site" maxlength="70"> <input
type="text" name="url" size="20" value="url du site" maxlength="255"><br>
<input type="submit" value="Envoyer" name="envoyer">
</center>
</form>
</html>
```

Concernant le script correspondant à "ajout.php" il faut l'écrire de la façon suivante :

```
<?
/* On commence par récupérer les valeurs avec $_POST si les champs ne sont
vides avec la fonction isset*/
if(isset($_POST['nom'])) $nom=$_POST['nom'] ;

if(isset($_POST['prenom'])) $nom=$_POST['prenom'] ;

if(isset($_POST['email'])) $nom=$_POST['email'] ;
```

```
if(isset($_POST['icq'])) $nom=$_POST['icq'] ;

if(isset($_POST['titre'])) $nom=$_POST['titre'] ;

if(isset($_POST['url'])) $nom=$_POST['url'] ;

// connexion à la base
$db = mysql_connect('localhost', 'login', 'password') or die('Erreur de
connexion '.mysql_error());

// sélection de la base
mysql_select_db('nom_de_la_base',$db) or die('Erreur de selection
'.mysql_error());

// on ecrit la requete sql
$sql = "INSERT INTO client VALUES('','$nom','$prenom','$email','$icq','$tit
re','$url)";

// on insère les informations du formulaire dans la table
mysql_query($sql) or die('Erreur SQL !'.$sql.'<br>'.mysql_error());

// on affiche le résultat pour le visiteur
echo 'Vos infos on été ajoutées.';

// on ferme la connexion
mysql_close();
?>
```

VI.22. Les cookies et les sessions

VI.22.1. Les sessions

Le principe des sessions consiste à associer à toute une navigation dans un site par un internaute, un identifiant de session unique. Lors du premier accès à une page du site, la session est créée, l'identifiant de session est généré par le serveur et transmis au client. Lors des accès ultérieurs, le client transmet l'identifiant de session au serveur, est donc reconnu, et les informations gardées pour lui par le serveur peuvent être restaurées. La manière la plus simple de mémoriser l'identifiant de session est d'utiliser un cookie, généré automatiquement par PHP lors de l'ouverture de la session. Néanmoins, cela ne fonctionnera pas si l'utilisateur a désactivé les cookies.

Pour créer une session, il faut appeler la fonction PHP **session_start()**. Pour mémoriser une variable sur le serveur, on utilisera le tableau **superglobal \$_SESSION** (PHP>=4.1).

Exemple :

On écrira : `$_SESSION["prix"] = 12.3` dans un premier fichier PHP.

Si on appelle la fonction `session_start()` quand une session a déjà été créée, les variables enregistrées dans la session sont restaurées.

Dans un deuxième fichier PHP, après l'appel à la fonction `session_start()`, `$_SESSION["prix"]` aura retrouvé sa valeur.

On pourra savoir si une variable `x` est bien enregistrée dans la session en testant la valeur de `isset($_SESSION["x"])`.

Dans les versions de PHP inférieures à 4.1, la mémorisation des variables se fait par l'appel à la fonction `session_register("toto")`, où `$toto` est la variable à mémoriser.

Une alternative utilisant le tableau global `$HTTP_SESSION_VARS` consiste à affecter `$HTTP_SESSION_VARS["toto"] = $toto;`

Avec les variables automatiques (PHP < 4.2 ou PHP >= 4.2 et `register_globals` activé), `$toto` prendra automatiquement la valeur qu'il avait à la fin de l'exécution du dernier script.

Avec les variables globales, `$HTTP_SESSION_VARS["toto"]` retrouvera également sa valeur. L'utilisation de `$_SESSION` (ou `$HTTP_SESSION_VARS` avec PHP 4.0.6 et plus ancien) est recommandé pour une meilleure sécurité et un code plus facilement maintenable.

Avec `$_SESSION`, il n'y a pas besoin d'utiliser les fonctions `session_register`, `session_unregister` et `session_is_registered`. Les variables de sessions sont accessibles comme toute autre variable.

VI.22.2. Les cookies

Un cookie est un petit fichier texte, qui est stocké par un site web sur le disque dur de l'internaute. Ce stockage est réalisé par le navigateur. Cela est utilisé par le site web visité pour "reconnaître" l'internaute. Aussi, quand plusieurs jours après la visite de l'internaute, il revient sur le même site, celui-ci demandera le cookie, et s'il est encore présent sur le disque dur de l'internaute, le site web lira les informations contenues dans le cookie et le "reconnaîtra". Le verbe reconnaître est mis entre guillemets, car il s'agit d'une pseudo-authentification...

Certains internautes ont donc pour habitude de désactiver les cookies, ce qui a pour effet de rendre certains services inopérants. Les concepteurs de sites web doivent prendre en compte

ce comportement, soit en avertissant les utilisateurs qu'ils doivent activer les cookies, soit en utilisant d'autres mécanismes.

Lorsque le programmeur d'un site web souhaite mémoriser un grand nombre d'informations sur un utilisateur du site web, les cookies (qui stockent les informations dans le navigateur du client) sont inadaptées. Il vaut mieux mémoriser ces informations sur le serveur, les associer à un identifiant unique, et ne mémoriser sur le navigateur (coté client) que cet identifiant. C'est exactement ce principe qui est utilisé par les sessions, introduites par la version 4 de PHP.

Exemples :

- Enregistrer une variable avec `$_SESSION`.

```
<?
phpsession_start();
// Utilisez $HTTP_SESSION_VARS avec PHP 4.0.6 ou plus ancien
if (!isset($_SESSION['count'])) { $_SESSION['compteur'] = 0;}
else { $_SESSION['compteur']++;}
?>
```

- Retirer une variable de session avec `$_SESSION` et `register_globals` inactif.

```
<?
phpsession_start();
// Utilisez $HTTP_SESSION_VARS avec PHP 4.0.6 ou plus ancien
unset($_SESSION['compteur']);
?>
```

- Enregistrer une variable avec `register_globals` activé

```
<?
php if (!session_is_registered('compteur'))
    {session_register("compteur");
    $compteur = 0;}
else { $compteur++; }
?>
```

- Retirer une variable de session avec `$_SESSION` et `register_globals` activé

```
<?
php session_start();
// Avec PHP 4.3 et plus récent, vous pouvez simplement utiliser l'exemple précédent
session_unregister('compteur');
?>
```



Module : Multimédia et développement Web

Chapitre 7 : Introduction aux technologies XML

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Attention !

Ce produit pédagogique numérisé est la propriété exclusive de l'UVT. Il est strictement interdit de la reproduire à des fins commerciales. Seul le téléchargement ou impression pour un usage personnel (1 copie par utilisateur) est permis.

INTRODUCTION AUX TECHNOLOGIES XML



Après avoir étudié cette partie, l'étudiant sera en mesure de :

- Acquérir les fondements du langage de balisage extensible (XML).
- Connaître des technologies qui lui sont associées (XML1.0, DTD, XPath)

VI.1. Introduction

VII.1.1. Un peu d'histoire

XML (eXtensible Markup Language) n'est pas vraiment une nouveauté puisqu'il est un sous-ensemble de SGML (Standardized General Markup Language) développé par IBM en 1969, comme langage de référence en milieu professionnel pour tout ce qui concerne la gestion électronique des documents.

Comme SGML est un peu trop complexe pour être utilisé sur le Web, XML est une version simplifiée de SGML, adapté spécialement au Web.

VII.1.2. Comparaison entre HTML et XML

Le seul point commun entre le HTML et le XML est qu'ils sont issus tous les deux de la même "mère" soit le SGML avec lequel ils partagent des caractéristiques communes :

- ✓ Ils fonctionnent avec des balises.
- ✓ Ils sont indépendants de la plateforme.
- ✓ Ils sont en mode texte.

Le XML est un langage de balises comme le HTML mais il est extensible, évolutif. En XML, les balises ne sont pas prédéfinies. C'est le développeur qui doit définir ses propres balises. Le XML et le HTML sont deux langages distincts. Voici un tableau comparatif de ces deux langages :

Le XML	Le HTML
Il décrit, structure, stocke, transporte et échange des données.	Il affiche des données par l'intermédiaire d'un navigateur.
C'est un générateur de langages [métalanguage].	C'est un langage statique de publication sur le Web.
Outre les PCs, le XML est adapté à d'autres outils comme les mobiles, les pockets, etc.	Le HTML est surtout conçu pour les ordinateurs de type PC.

Pour le XML, le W3C est reparti d'une feuille blanche et a mis en place un nouveau langage très structuré.	Le Html avec la version 4.0 est arrivé à bout de course et est devenu un langage hybride et en final peu structuré..
Le XML est un langage strict dont l'écriture doit être rigoureuse.	Le Html, à cause des navigateurs récents, est devenu très permissif.

VII.1.3. Les limites et les objectifs

Le XML est un métalangage soit un langage pour écrire d'autres langages. Il est à la base de toute une série de nouveaux langages qui sont ou qui seront utilisés dans la conception des pages Internet comme :

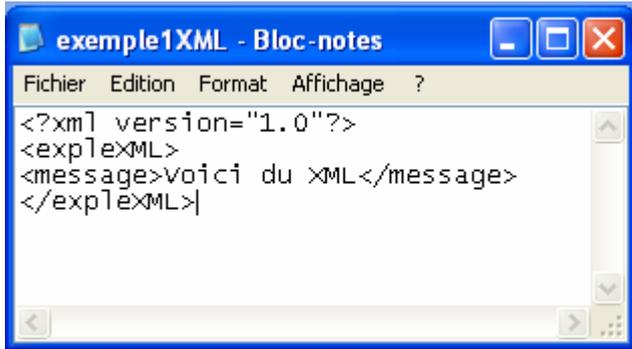
- ✓ Le XHTML (successeur du Html) Mais il est par ailleurs aussi un des "enfants" engendrés par le XML. En deux mots, pour faire un peu le ménage dans les dérivés du Html au fil des différentes versions, le W3C a conçu le XHTML qui n'est en fait qu'une reformulation du HTML 4.0 selon la syntaxe et les règles du XML.
- ✓ Le WML pour le Wap des téléphones mobiles,
- ✓ Le MathML pour les mathématiques,
- ✓ Etc.

VII.2. Le langage XML

VII.2.1. Structure de base d'un document XML

Le Html a été conçu pour afficher de l'information alors que le XML a été créé pour structurer de l'information.

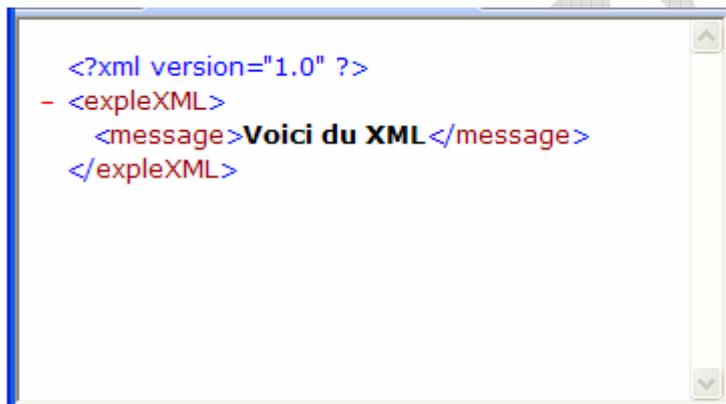
Faisons un premier document XML, soit les codes XML suivants:



```

Fichier Edition Format Affichage ?
<?xml version="1.0"?>
<expleXML>
<message>voici du XML</message>
</expleXML>
    
```

- Ouvrir un éditeur de page web.
- Ecrire ce script, puis enregistrer ce document avec l'extension .xml
- Ouvrir ce fichier sous Internet Explorer, ce qui donne :



```

<?xml version="1.0" ?>
- <expleXML>
  <message>Voici du XML</message>
</expleXML>
    
```

On remarque qu'il y a un petit signe - affiché devant la balise <expleXML> (une balise englobante). Il suffit de cliquer sur le signe pour masquer celle-ci. Et bien entendu de cliquer sur le signe + pour la faire réapparaître.

VII.2.2. La syntaxe du XML

Le XML impose des règles de syntaxe très spécifiques par rapport au Html. En outre, on retrouvera ces mêmes règles de syntaxe dans tous les langages dérivés du XML comme le XHTML ou le WML par exemple.

- ✓ **Le XML est un langage de balises [Markup Langage]** : Mais au contraire du Html où les balises sont définies, on doit inventer nos balises. Il y a quelques règles pour la composition des noms :

- Les noms peuvent contenir des lettres, des chiffres ou d'autres caractères.
- Les noms ne peuvent pas débuter par un nombre ou un signe de ponctuation.
- Les noms ne peuvent pas commencer par les lettres xml (ou XML ou Xml...).
- Les noms ne peuvent pas contenir des espaces.
- La longueur des noms est libre mais on conseille de rester raisonnable.
- On évitera certains signes qui pourraient selon les logiciels, prêter à confusion comme "-", ";", ".", "<", ">", etc.
- Les caractères accentués comme é, à, ê, ï, ù sont à priori permis mais pourraient être mal interprétés par certains programmes (déconseillé).

Par exemple :

```
xml:lan extrait_de titre poeme79
```

- ✓ **<?xml version="1.0" encoding="ISO-8859-1"?>** : Généralement tout fichier XML commence par cette instruction : la déclaration **<?xml version="1.0"?>** indique au navigateur que ce qui suit est un document XML selon sa version 1.0. **Encoding="ISO-8859-1"** représente le jeu de caractère pour les francophones et qui accepte les caractères accentués. Mais il existe d'autres jeux de caractères comme UTF-8 ou UTF-16 plutôt destinés aux anglo-saxons car ils n'acceptent pas les accents.
- ✓ **Les balises sont sensibles aux majuscules et minuscules** : Ainsi, la balise <Message> est différente de la balise <message>. La balise d'ouverture et la balise de fermeture doivent donc être identiques. Ainsi par exemple ; <Message> ... </message> est incorrect et <message> ... </message> est correct. Une tendance se dégage pour n'écrire que des balises qu'en minuscules, limitant ainsi les erreurs possibles.
- ✓ **Toute balise ouverte doit être fermée** :

Par exemple :

En Html, ce qui suit est affiché correctement :

```
<p>
```

```
<ul>
```

```
<li>Point 1
```

```
<li>Point 2
```

Le XML est beaucoup plus strict. On devrait avoir :

```
<p>
<ul>
<li>Point 1</li>
<li>Point 2</li>
</ul>
</p>
```

- ✓ **Les balises doivent être correctement imbriquées** : Le XML est très préoccupé par la structure des données, des balises mal imbriquées sont des fautes graves de sens.

Ainsi l'écriture suivante est incorrecte car les balises ne sont pas bien imbriquées :

```
<parent><enfant>Loïc</parent></enfant>
```

L'écriture correcte avec une bonne imbrication des éléments est :

```
<parent><enfant>Marine</enfant></parent>
```

- ✓ **Tout document XML doit comporter une racine** : En fait, la première paire de balises d'un document XML sera considéré comme la balise de racine [root]. Tous les autres éléments seront imbriqués entre ces balises de racine.

Par exemple :

```
<parents>
  <enfants>
    <petits_enfants> ... </petits_enfants>
  </enfants>
</parents>
```

- ✓ **Les valeurs des attributs doivent toujours être mises entre des guillemets** : Le XML peut avoir (comme le Html) des attributs avec des valeurs. En XML, les valeurs des attributs doivent obligatoirement être entre des guillemets, au contraire du Html où leur absence n'a plus beaucoup d'importance.

Ainsi, l'écriture suivante est incorrecte car il manque les guillemets.

```
<date_anniversaire=071185>
```

La bonne écriture est :

```
<date_anniversaire="071185">
```

- ✓ **La section CDATA** : Lorsqu'un texte contient des caractères qui jouent un rôle de délimiteur dans la syntaxe XML, il est nécessaire de pouvoir annihiler ce rôle. Ceci peut être fait en insérant le texte contenant ces délimiteurs dans une section CDATA sous la forme suivante :

```
<![CDATA[texte contenant des délimiteurs]]>
```

Le texte inséré peut contenir n'importe quels caractères excepté la chaîne]]. Une section CDATA ne peut donc pas en contenir une autre.

Par exemple : « L'expression `<ALT>2794</ALT>` est un élément XML. » peut être représentée par l'élément suivant :

```
<phrase>L'expression <![CDATA[<ALT>2794</ALT>]]> est un élément XML.</phrase>
```

Remarque :

Le XML est un langage strict. Votre document doit impérativement respecter la syntaxe du XML. On dira alors que le document est "bien formé". Seuls les documents "bien formés" seront affichés correctement. A la moindre erreur de syntaxe, le document ne sera pas ou ne sera que partiellement affiché.

VII.2.3. Le DTD ou Document Type Declaration

Le DTD ou Document Type Declaration ou encore Document Type Definition est l'ensemble des règles et des propriétés que doit suivre le document XML. Ces règles définissent généralement le nom et le contenu de chaque balise et le contexte dans lequel elles doivent exister. Cette formalisation des éléments est particulièrement utile lorsqu'on utilise de façon récurrente des balises dans un document XML.

On distingue deux types de DTD : le DTD interne et le DTD externe.

VII.2.3.1. Le DTD interne

Dans le DTD interne, On peut inclure nos propre DTD au code source du fichier XML. Le DTD interne suit la syntaxe suivante :

```
<!DOCTYPE élément-racine [
    déclaration des éléments
]>
```

Voici un exemple :

Instructions	Significations
<pre><?xml standalone="yes"?> version="1.0" <!DOCTYPE parent [<!ELEMENT parent (garcon,fille)> <!ELEMENT garcon (#PCDATA)> <!ELEMENT fille (#PCDATA)>]> <parent> <garcon>Ahmad</garcon> <fille>Fatma</fille> </parent></pre>	<ul style="list-style-type: none"> - Puisqu'on définit un DTD interne, ce fichier est indépendant (standalone) - Début du DTD interne avec parent comme élément de racine - Cet élément racine soit parent contiendra les sous- éléments garçon et fille - #PCDATA indique à XML que l'élément garçon contient des données exprimées en chiffres ou en lettres - Idem pour l'élément fille - Fin du DTD - Racine du document XML - Fin du document XML

VII.2.3.2. Le DTD externe

Le DTD externe suivra la syntaxe suivante :

```
<!DOCTYPE élément-racine SYSTEM "nom_du_fichier.dtd">
```

Le même fichier que ci-dessus serait alors :

```
<?xml version="1.0" standalone="no"?>
  <!DOCTYPE parent SYSTEM "parent.dtd">
  <parent>
    <garcon>Ahmad</garcon>

    <fille>Fatma</fille>
  </parent>
```

Le fichier de DTD externe (ici dans le même répertoire) "parent.dtd" contiendrait :

```
<!ELEMENT parent (garcon,fille)>
<!ELEMENT garcon (#PCDATA)>
<!ELEMENT fille (#PCDATA)>
```

Mais il est aussi possible de faire référence à un DTD externe situé sur un autre site comme pour par exemple le XHTML :

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//FR"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

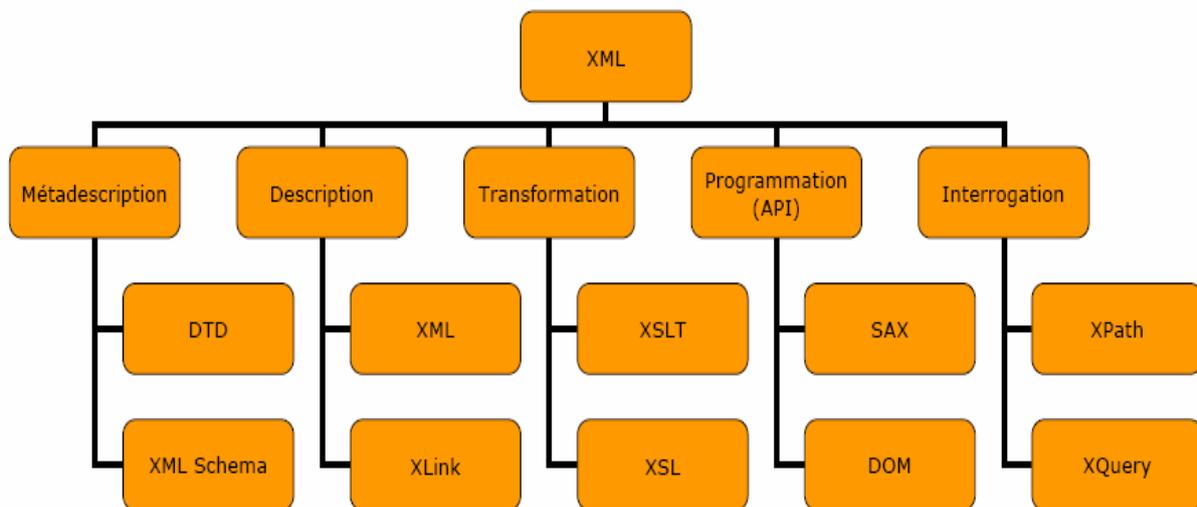
Remarque :

Un document "bien formé" est un document qui respecte les règles générales de syntaxe du XML alors qu'il est "valide" s'il respecte les règles spécifiques de son DTD

VII.3. La famille de technologie XML

VII.3.1. Introduction

Voici un extrait de la famille de technologies XML :



Le XML est une famille de technologies ou encore un métalangage soit un langage pour écrire d'autres langages.

Cette famille est composée par des langages dérivés du langage XML. Ces langages reprennent donc toutes les règles de syntaxe du XML (détaillée dans la partie relative au XML).

En bref :

- ✓ les balises sensibles à la casse, s'écrivent en minuscules.
- ✓ toutes les balises ouvertes doivent être impérativement fermées.
- ✓ les balises vides auront aussi un signe de fermeture soit <balise/>.
- ✓ les balises doivent être correctement imbriquées.
- ✓ les valeurs des attributs doivent toujours être mises entre des guillemets.
- ✓ le document XSL devra être "bien formé".

Les principaux langages de manipulation de documents XML :XPath, XSLT, XQuery... opèrent sur un modèle de données commun : le modèle XDM (XML Data Model). Dans ce modèle, un document XML est représenté sous forme d'un arbre de document.

Ses nouveaux langages sont répartis en 5 domaines : Description, Métadescription, Interrogation, Transformation et Programmation.

VII.3.2. Les langages de description : XML 1.0, XLink, XPointer

VII.3.2.1. XML 1.0

Il définit ce que sont les « balises » et les « attributs », mais autour de cette description, un nombre de plus en plus important de modules facultatifs ont été définis, fournissant des ensembles de balises et d'attributs ou des lignes directrices pour des tâches particulières.

VII.3.2.2. XLink et XPointer

La description de la structure hypertextuelle d'un document XML est faite au moyen :

- ✓ du **langage XLink** qui permet de décrire un lien en termes d'éléments XML,
- ✓ du **langage XPointer** qui permet de localiser les fragments de documents qui constituent les extrémités d'un lien.

VII.3.3. Les langages de Métadescription : XML Schema

Un schéma écrit en **XML Schema** définit une classe de documents. XML Schema offre pour cela un système très évolué, à la différence des DTD dont le seul type atomique est le type chaîne de caractères.

L'espace de noms associé à XML Schema est : <http://www.w3.org/2001/XMLSchema> dont le préfixe est xsd. Un schéma XML est une suite de définitions de types et de déclarations d'attributs.

VII.3.4. Les langages d'Interrogation : XPath, XQuery

VII.3.4.1. XPath

Le langage XPath opère sur l'arbre d'un document. Il permet de sélectionner l'ensemble des nœuds que l'on peut atteindre en suivant, à partir d'un nœud donné de l'arbre d'un document, tous les chemins conformes à un modèle appelé chemin de localisation.

Le langage XPath est un langage fonctionnel et donc un langage d'expressions.

VII.3.4.2. XQuery

XQuery est un langage fonctionnel d'interrogation de données XML en cours de recommandation par le W3C. C'est un langage déclaratif « à la SQL » :

- ✓ SQL manipule des tables (ensemble de n-uplets de valeurs atomiques),
- ✓ XQuery manipule des séquences d'items (valeurs atomiques et nœuds d'arbres XML).

XQuery est un langage puissant mais complexe à mettre en œuvre :

- ✓ à cause de la complexité des structures manipulées : les arbres XML,
- ✓ à cause de la semi-structuration de ces données : complexité du typage notamment.

VII.3.5. Les langages de Transformation :XSL, XSLT

XSLT permet de produire un nouveau document XML par transformation d'un document XML existant. Une transformation s'exprime sous la forme d'une feuille de style qui est un document XML composé d'un ensemble de règles de transformation.

Une règle comporte deux parties :

- ✓ un modèle de chemin exprimé en termes du langage XPath,
- ✓ une transformation sous la forme d'une suite d'instructions.

L'espace de noms associé au langage XSLT est : <http://www.w3.org/1999/XSL/Transform> dont le préfixe usuel est xsl.

Le langage XSL est plus qu'une série de balises pour afficher du XML. Il comporte aussi des possibilités plus qu'utiles quand on est confronté à des données. Il permet :

- ✓ de trier les données XML en ordre croissant ou décroissant.
- ✓ de filtrer des éléments XML en fonction de certains critères.
- ✓ de choisir des éléments.
- ✓ de retenir des éléments par des tests conditionnels.

VII.3.6. Programmation API : SAX, DOM

VII.3.6.1. Définition de l'API

L'API (Application Programming Interface) est une interface de programmation d'applications, contenant un ensemble de fonctions courantes de bas niveau, bien documentées, permettant de programmer des applications de «Haut Niveau». On obtient ainsi des bibliothèques de routines, stockées par exemple dans des DLL.

VII.3.6.2. SAX et DOM

Simple API for XML ou SAX est une API générale pour la lecture d'un flux XML. Il existe des implémentations de cette API dans la majorité des langages (en tout cas l'implémentation existe pour C++, C#, Java, Pascal, perl, PHP, ...) car XML s'est largement imposé aujourd'hui dans le monde du logiciel pour l'échange d'informations. Il existe deux grandes API pour relire les flux XML : DOM et SAX.

SAX est événementielle, ce qui signifie qu'il existerait des événements dans un fichier XML alors que DOM transforme l'arborescence XML en arborescence du langage cible. Et bien, si l'on considère le flux XML entrant, il devient plus facile de le découvrir par des événements : une balise ouvrante est un événement, une balise fermante un autre événement...

UVA

TP1 :
L'INTERNET

UVT

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Partie I : L'INTERNET

I.1. La navigation WEB



A/ Structure d'un navigateur

Lorsque vous lancez votre navigateur Web pour la première fois (la plupart du temps en cliquant sur l'icône placée sur votre bureau), une page Web prédéfinie, votre page d'accueil apparaît

- *La barre d'outils*



La rangée de boutons en haut de votre navigateur Web, également appelée barre d'outils, vous aide dans votre exploration du Web, vous permettant même de conserver la trace de vos déplacements.

- **Précédent** vous renvoie à la page précédemment visitée.
- **Suivant** vous permet de revenir à la page que vous venez de quitter.
- **Accueil/Démarrage** vous transfère vers la page d'accueil de votre choix. Si vous n'en avez sélectionné aucune, ce bouton vous renvoie à la page d'accueil par défaut, généralement le site Web de Microsoft ou Netscape.
- **Recharger** ou **Actualiser** recharge la page Web, comme son nom l'indique. Quand utiliser ce bouton ? Parfois tous les éléments d'une page Web ne se chargent pas complètement du premier coup, suite à une interruption du transfert des fichiers. De même, lorsque vous

téléchargez une page Web, les données sont mises en mémoire cache, ce qui signifie qu'elles sont temporairement sauvegardées sur votre ordinateur. Lorsque vous souhaitez rouvrir la page, cette dernière est récupérée par votre navigateur Web dans la mémoire cache au lieu d'être redemandée au serveur.

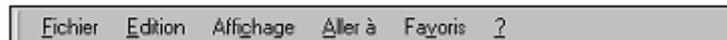
- **Imprimer** permet l'impression sur papier du document chargé dans votre navigateur.
- Le bouton **Arrêter** interrompt le chargement de la page en cours par le navigateur.
- Le bouton **Rechercher** effectue une connexion aux pages des sites Web Microsoft et Netscape, proposant un ensemble de répertoires Internet et d'outils de recherche.
- **Signets** ou **Favoris** présentent une liste dans laquelle vous pouvez enregistrer les adresses des sites Web que vous souhaitez visiter à nouveau. Une fois que vous avez ajouté un URL à votre liste, il vous suffit de cliquer dessus pour retourner à la page Web auquel il correspond, ce qui vous évite de le retaper.

- **La barre d'adresses**

Un champ nommé « **Emplacement** », « **Atteindre** » ou « **Adresse** » figure juste sous la barre d'outils. C'est à cet endroit que vous tapez l'adresse du site Web que vous souhaitez visiter. Une fois votre saisie effectuée, appuyez sur la touche Entrée ou Retour pour accéder au site.

Si vous cliquez sur le petit triangle à droite en regard du champ Adresse, une liste déroulante s'affiche contenant les adresses des sites que vous avez récemment visités. Pour retourner sur un site, il vous suffit de cliquer sur son adresse.

- **La barre de menus**



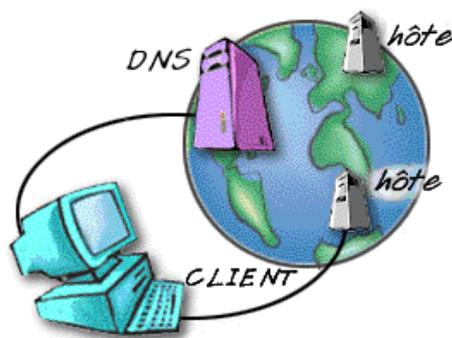
Située tout en haut de la fenêtre du navigateur, la barre de menus présente toutes les fonctionnalités liées à une page Web, comme la possibilité de l'**enregistrer** sur votre disque dur (Fichier + Enregistrer).

- **La barre d'état**



Au bas de votre navigateur Web se trouve un encadré que l'on appelle la barre d'état. Elle vous montre l'état d'avancement de vos actions, comme l'adresse du site auquel vous vous connectez, l'état de la connexion avec l'ordinateur hôte et la taille des fichiers à télécharger.

B/ Le nom du domaine



Pour entrer l'URL d'un site, il suffit de le taper dans la **barre d'adresses** de votre navigateur Web, située juste en dessous de la barre d'outils.



La plupart des URL que vous utiliserez commence par **http**, abréviation correspondant à Hypertext Transfer Protocol. Http est le protocole utilisé pour le transfert des fichiers **HTML** sur le Web. Voici quelques autres informations importantes sur les URL :

1. en règle générale, un URL ne contient aucun espace ;
2. il comporte toujours des barres obliques (//) ;
3. si vous saisissez un URL de façon incorrecte, le navigateur ne pourra pas localiser le site ou la ressource que vous recherchez.
4. pour connaître l'URL se rapportant à un lien, il vous suffit de déplacer le curseur de la souris sur le lien. Le pointeur se transforme en main et l'URL

s'affiche dans la *barre d'état* du navigateur, généralement située en bas de l'écran.

C/ La mémorisation des URL fréquemment utilisés

La plupart des navigateurs gardent en mémoire les URL que vous souhaitez réutiliser, en les ajoutant à un menu spécifique (Netscape Navigator, : **Signets**, Microsoft Explorer : **Favoris**). Une fois que vous avez ajouté un URL à votre liste, il vous suffit de cliquer dessus pour retourner à la page Web auquel il correspond.

Lorsque vous souhaitez enregistrer l'adresse d'un site, sélectionnez l'option de menu Signets ou le menu Favoris et cliquez sur **Ajouter**. Si vous cliquez à nouveau sur l'icône, le titre de la page Web enregistrée s'affiche en fin de liste des sites préférés. Pour visiter un site favori, il vous suffit de cliquer sur son nom.

Pour enregistrer la page actuelle parmi vos sites préférés, cliquez une fois dessus à l'aide du bouton droit de la souris, puis sélectionnez dans le menu contextuel l'option Ajouter signet ou Ajouter aux favoris.

Pour trier ces adresses, si vous utilisez Internet Explorer :

- Cliquez sur le bouton **Favoris** de la barre d'outils pour ouvrir la fenêtre Favoris.
- Cliquez sur le bouton **Organiser**. La fenêtre **Organiser les Favoris** s'ouvre.
- Cliquez sur le bouton **Créer un dossier**.
- Donnez un nom au dossier créé. L'organisation des signets peut être triée par sujets (Alimentation, Voyages, Actualités, Sports, etc.).
- Cliquez une fois sur chacun de vos favoris.

Enfoncez le bouton gauche de votre souris afin de pouvoir les déplacer un à un vers le dossier correspondant.

Si vous utilisez Navigator :

- Cliquez sur l'icône **Signets** pour afficher tous vos signets.
- Sélectionnez l'option **Modifier les signets**.
- Dans le menu de la boîte **Signets** qui s'affiche, sélectionnez l'option **Nouveau dossier**.
- Dans le champ **Nom**, entrez le nom du dossier, puis cliquez sur **OK**.
- Retournez ensuite à votre liste de signets, cliquez sur l'un d'entre eux de façon à le mettre en surbrillance, et déplacez-le vers le dossier auquel il se rapporte.

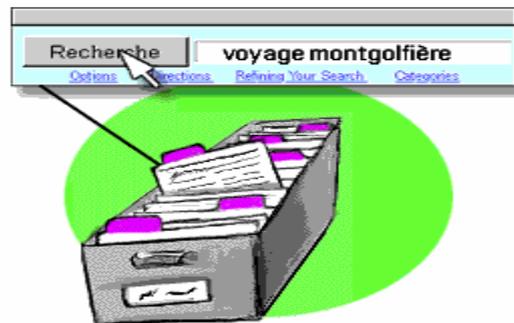
Pour modifier les noms des signets avec Explorer :

- Dans le menu **Favoris**, sélectionnez l'option **Organiser les Favoris**.
- Dans la boîte de dialogue qui s'affiche, sélectionnez les favoris que vous souhaitez modifier.
- Cliquez sur le bouton **Renommer** puis entrer un nouveau nom.

Avec Navigator :

- Sélectionnez le signet à modifier en cliquant une fois dessus.
- Dans le menu **Edition** de la boîte **Signets**, sélectionnez l'option **Propriétés des signets**.
- Entrez le nouveau nom et cliquez sur **OK**.

D/ La recherche des informations



Le mathématicien anglais, George Boole, a développé une algèbre sur la logique, devenu le fondement des recherches dans les bases de données informatiques. Cette logique utilise des mots appelés **opérateurs** pour déterminer si une instruction est vraie ou fausse. Les opérateurs les plus courants sont **AND**, **OR** et **NOT**. Ces trois petits mots peuvent s'avérer d'une grande utilité lors des recherches en ligne. En voici quelques illustrations :

Termes recherchés...	Résultats...
guide AND touristique	Des documents contenant les deux mots.
guide OR touristique	Le nombre le plus important d'occurrences ; des documents contenant l'un des deux mots.
guide NOT touristique	Des documents sur les guides, mais pas sur les guides touristiques, ce qui permet de restreindre efficacement le champ.

Remarque :

Certains moteurs comportent une variante des opérateurs AND et NOT. Un symbole + devant un nom (sans espace intermédiaire) signifie que les documents trouvés doivent contenir ce mot. Un symbole - placé devant un mot-clé, exclut des résultats les documents comportant ce mot.

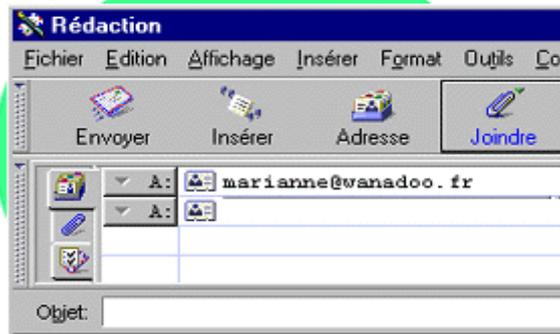
Notez par ailleurs que tous les mots devant être placés dans le document doivent être précédés d'un signe +, même le premier.

Soit l'exemple suivant : +Internet garantit la présence s'Internet dans les documents trouvés.

I.2. Le courrier électronique



A/ La structure d'un courrier électronique



Les messages électroniques sont semblables aux lettres et se composent de **deux parties principales**.

L'**en tête** contient le nom et l'adresse du destinataire et aussi de toutes les personnes placées en copie, ainsi que l'objet du message. Certains programmes de messagerie affichent également votre nom et votre adresse ainsi que la date d'envoi du message. Le **corps** contient le message.

L'adresse que vous indiquez doit être correcte tout comme lorsque vous envoyez des lettres. Si l'adresse indiquée est erronée ou comporte une faute de frappe, le message vous est retourné, via la routine de retour à l'expéditeur pour cause d'adresse inconnue.

Lorsque vous recevez un courrier électronique, l'en tête vous indique sa provenance, sa date et son mode d'envoi. On peut comparer cela à un cachet de poste électronique.

B/ Les adresses électroniques

Les adresses électroniques sont généralement constituées de **deux parties principales** :

professeur@isetsf.tn

- En premier vient le **nom d'utilisateur** (professeur) qui fait référence à la boîte aux lettres du destinataire.
- Ensuite apparaît le **caractère arrobas** (@).
- Puis vient le **nom d'hôte** (isetsf), encore appelé le **nom de domaine**. Ce dernier se rapporte au **serveur de messagerie**, l'ordinateur sur lequel le destinataire dispose d'une boîte aux lettres électronique. Il s'agit bien souvent du nom d'une institution ou d'une société. Il est suivi d'un point (.).
- Puis de deux ou trois lettres (.tdans le cas présent) indiquant le nom de domaine.

C/ La création d'un compte

Pour établir un nouveau compte de messagerie électronique, vous avez besoin d'indiquer quelques informations à votre sujet, et de choisir un nom et un mot de passe pour votre compte. Votre nom ou identifiant de compte est intégré à votre adresse électronique.

Les noms de compte peuvent utiliser des lettres et des chiffres tels que « professeur2000 », mais ils ne peuvent contenir d'espaces et leur longueur est limitée selon les services.

Des centaines de sites Web offrent des messageries électroniques gratuites sur le Web, tels que :



D/ La création d'un message

Une fois le compte est créé, vous avez la possibilité de créer et de recevoir des messages.

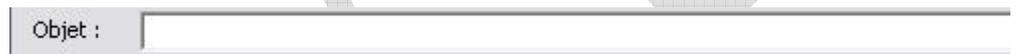
Pour pouvoir créer un message il faut :

- Cliquez sur l'icône Créer un message ou Nouveau Message

- Tapez l'adresse électronique de votre correspondant



- Tapez le sujet de votre message



- Tapez le texte de votre message



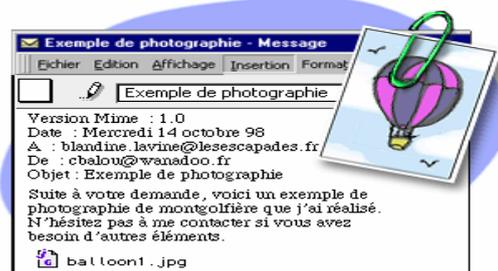
- Cliquez sur l'icône Envoyer



E/ La consultation des messages

Pour consulter vos messages, vous avez besoin d'indiquer le nom et le mot de passe pour votre compte. Puis cliquer sur **Lire** les messages. Vous aurez la liste des messages, qu'il suffit d'y cliquer deux fois ou de cliquer sur **Ouvrir** pour pouvoir les lire.

F/ L'insertion de fichiers en pièces jointes



L'insertion de fichiers en pièces jointes vous permet d'envoyer des documents formatés, des photographies, des fichiers son et vidéo en pièces jointes à vos courriers électroniques.

Il suffit de cliquer sur un trombone, icône apparaissant dans la barre d'outils du programme, ou sur **Fichier joint**. Puis vous sélectionnez le fichier à joindre en le localisant sur le disque dur ou l'unité concernée. Une fois la sélection effectuée, soit le fichier apparaît sous forme d'icône dans le corps du texte, soit son nom s'affiche dans l'en-tête du message.

La plupart des programmes vous permettent de joindre plusieurs fichiers.

I.3. Le téléchargement



L'un des meilleurs endroits pour trouver des programmes téléchargeables est le site Shareware.com du CNET.

Ci-dessous, un guide des formats de fichier courants sur Internet.

- Fichiers en texte seul

.html/.htm

Langage utilisé pour la création de documents Web. Ce type de fichier nécessite un navigateur Web, tel que Netscape Navigator ou Internet Explorer pour l'affichage.

.txt

Fichier en texte. Ces fichiers peuvent être visualisés à l'aide d'un traitement de texte tel que Microsoft Word ou un simple éditeur de texte tel que Simple Text ou BBEdit pour Mac. Sur PC, vous pouvez utiliser le bloc-notes fourni avec le système d'exploitation Windows ou le programme shareware Programmer's File Editor (pfexxx.zip).

- Documents formatés

.doc

Format PC courant pour les fichiers texte formatés. Cependant, il peut arriver que vous rencontriez des fichiers portant cette extension qui ne

correspondent pas à des fichiers texte. Généralement, il s'agit de documents créés à l'aide de Microsoft Word ou WordPerfect pour Windows.

.pdf

Portable Document Format est un format propriétaire développé par Adobe Systems, Inc. qui permet à des documents formatés (y compris des brochures ou d'autres documents comportant des graphiques) d'être transférés sur le Net et d'avoir le même aspect sur toutes les machines. Adobe Acrobat Reader est nécessaire pour afficher ce type de fichier.

- Fichiers compressés et encodés

.arj

Format assez courant pour les machines MS-DOS.

.exe

Programme DOS ou Windows ou fichier exécutable qui se décompresse lui-même.

.gz/gzip

Programme de compression le plus couramment utilisé.

.zip

Norme de compression courante pour DOS et Windows qui utilise l'utilitaire DOS, PKZIP. Ces fichiers peuvent être décompressés sous PC à l'aide de **WinZIP**.

- Fichiers graphiques

.gif

Format de fichier graphique le plus courant sur Internet. Gif signifie Graphics Interchange Format.

.jpg/jpeg/jfif

Norme de compression courante, utilisée pour les photographies et les images statiques.

- Fichiers sons**.mp3**

Format de fichier le plus répandu sur le Web pour la diffusion de musique ayant la qualité du CD. Un fichier d'un Mo équivaut environ à une minute de musique. Ce type de fichier nécessite un lecteur MP3, lequel est disponible à la fois pour Macintosh et pour Windows.

.wav

Format de fichier son natif pour Windows. Il existe un bon programme : Win Play! (wplnyxx.zip) qui permet de lire ce type de fichiers ainsi que d'autres formats courants.

- Fichiers vidéo**.avi**

Format de fichier vidéo standard pour Windows. Pour lire ces fichiers, utilisez un lecteur AVI Video pour Windows (aviprox.exe).

.mov/.movie

Format courant pour les films QuickTime

.mpg/mpeg

Format standard pour les films sur Internet, qui utilise le schéma de compression MPEG.

I.4. Internet Relay Chat (IRC)



Pour réaliser une discussion, vous avez tout d'abord besoin d'un logiciel de **lecture de news**. *Netscape Navigator* et *Internet Explorer* intègrent tous deux des programmes de ce type :

- En premier lieu, cherchez une chat room qui vous intéresse.
- Une fois sur le site, il faudra sans doute vous enregistrer. Vous êtes libre d'utiliser votre nom, mais la plupart des gens s'inventent un pseudonyme. Sachez que l'absence de nom lors de la connexion entraîne l'attribution par défaut du statut d'utilisateur **Anonymous** (anonyme).
- Cliquez sur le bouton d'entrée, et suivez les instructions pour vous intégrer à une conversation.
- Une fois entré, vous arriverez sans doute au beau milieu d'une conversation. Prenez le temps de comprendre le fil directeur des échanges.
- Pour participer à la discussion, tapez simplement votre message dans le champ vierge en bas de votre écran, et cliquez sur le bouton **Talk** ou appuyez sur la touche Entrée ou Retour de votre clavier. Vos messages sont alors postés vers la chat room et chacun des participants peut vous répondre.

I.5. Les Newsgroups



Pour accéder aux news, il faut disposer de :

- **Lecteur de news** : Un outil pour lire ou poster les articles. On distingue :

- **WinVn.**
- **Free Agent.**
- Ou alors tout simplement utiliser le navigateur WEB comme Netscape Navigator et Internet Explorer.

- **Un serveur de news** : Pour récupérer les articles, le lecteur de news doit avoir le nom du serveur. Il s'agit d'une base de données contenant la totalité des articles. Exemples :

- **News.voila.fr**
- **Deja.com**

Pour publier des messages à partir d'un lecteur de news :

- Vous devez tout d'abord entrer quelques informations personnelles, telles que votre nom, celui de votre société et votre adresse électronique. Ces éléments doivent systématiquement être fournis.

- Une fois votre lecteur de news configuré, vous répondrez peut-être à un précédent message, ou vous lancerez vous-même un débat.

- Votre message, encore appelé article, contient des informations d'en-tête suivies du texte de l'article. Chaque article est destiné à un ou plusieurs groupes figurant sur la ligne des groupes de discussion de l'en-tête.

- Vous pouvez envoyer votre message à plusieurs groupes de discussion à la fois.

- Un **article posté en réponse** possède la même ligne objet que l'article d'origine. Seule la mention « Re: » est ajoutée. Ceci permet aux autres utilisateurs du groupe de

retrouver plus facilement le fil conducteur de la discussion. Généralement, un message posté en réponse contient le texte initial entre guillemets.

UVA

TP 2.1 :
LE MULTIMEDIA

UVT

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Partie II : LE MULTIMEDIA

II.1. Macromedia Flash 8



II.2.1. Présentation de l'espace de travail

Avec Flash, Macromedia a réuni plusieurs idées et technologies puissantes en un programme unique permettant aux utilisateurs de télécharger des présentations multimédias sur le Web. C'est que Flash offre l'utilisation d'images vectorielles comme mode graphique par défaut. Aussi, l'approche de développement adoptée par Flash permet de réaliser des présentations multimédias complexes tout en maintenant les fichiers à une taille raisonnable. De plus Flash permet de réaliser une version unique d'un objet, qu'on pourra ensuite utiliser à nouveau ailleurs sans avoir à recréer l'objet.

Un dernier facteur qui permet à Flash de créer des fichiers multimédias à chargement rapide sur le Web est la lecture en continu (streaming) qui consiste à télécharger la première page alors que le reste se télécharge en arrière plan.

II.2.2. Les exercices d'application

Exercice 1 : Disparition d'effet de couleur d'un symbole en utilisant l'interpolation de mouvement

- Ouvrez un nouveau document. En utilisant l'outil cercle, dessiner un cercle de couleur bleu (par exemple).
- Double cliquez sur ce cercle pour le sélectionner en totalité (fond + contour).
- Choisissez «Insertion», «Nouveau symbole», «graphique».
- Sélectionnez l'image clé 10 et choisissez «Insertion», «scénario» puis «image clé».
- Sélectionnez l'instance du symbole sur la scène. Dans la fenêtre propriété de l'instance, dans l'option couleur choisir l'effet alpha et fixer son pourcentage à zéro.

- Sélectionnez la première image clé. Dans propriétés, choisissez «Interpolation de mouvement» et enfin choisissez «contrôle», «tester l'animation» pour voir la disparition d'effet de couleur.

Exercice 2 : Les effets scénarios

Essayer d'appliquer et tester les effets scénarios sur une image importée :

1. Créez une nouvelle animation
2. Importez une image dans la scène : **fichier→importer→importer dans la scène**
3. Sélectionnez l'image importée à laquelle vous souhaitez ajouter l'effet de scénario. Choisissez **Insertion > Effets de scénario**. Choisissez ensuite **Assistants, Effets ou transition/Transformation** dans le sous-menu, puis sélectionnez un effet dans la liste.

Choisissez par exemple l'effet **Transformation**

Exercice 3 : Interpolation de mouvement

- Sélectionnez Outil Ovale et faites un cercle dans la première image clé.
- Sélectionnez ce cercle. Allez à Insertion puis Convertissez en symbole.
- Allez à la quinzième image clé. Insertion. Image-clé. Déplacez le cercle à l'endroit que vous souhaitez. Dans le menu insertion, choisissez « Interpolation de mouvement».
- Allez à la trentième image clé. Insertion. Image-clé. Déplacez à nouveau le cercle. Dans le menu insertion, choisissez « Interpolation de mouvement» puis Contrôle. Testez l'animation.

Maintenant, tout en gardant ce même exemple, nous allons voir que l'on peut modifier certaines choses. Nous allons nous positionner à la quinzième image clé et nous allons sélectionner le cercle et le redimensionner cliquez avec le bouton droit sur le cercle, Redimensionnez-le. Même chose en allant sur la trentième image clé. Testez votre animation.

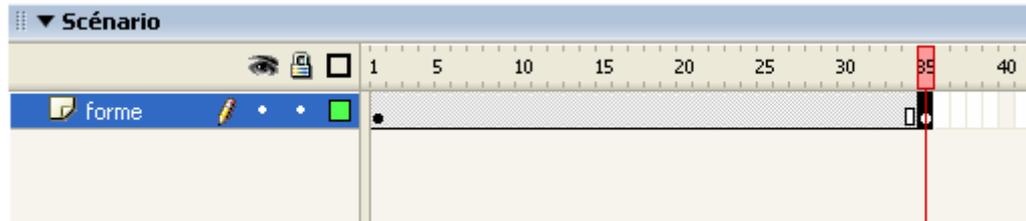
Exercice 4 : Création d'un guide de mouvement

- Dessinez votre objet. Sélectionnez-le avec l'Outil Flèche. Insertion. Convertissez-le en symbole (Graphique). OK. Aller à l'image 40. Insertion. Image-clé.
- Sélectionnez la colonne des calques. Insertion. Guide de mouvement.
- Allez à l'image 1 (du Guide Calque). Sélectionnez l'Outil Crayon. Dessinez la future trajectoire.
- Sélectionnez l'outil flèche. Prenez le carré et l'amenez au début de la trajectoire. Allez à l'image 40. Prenez le carré et l'amenez à la fin de la trajectoire.
- Faites un clic droit sur le calque 1. Créez l'interpolation de mouvement.
- Contrôle. Tester l'animation.

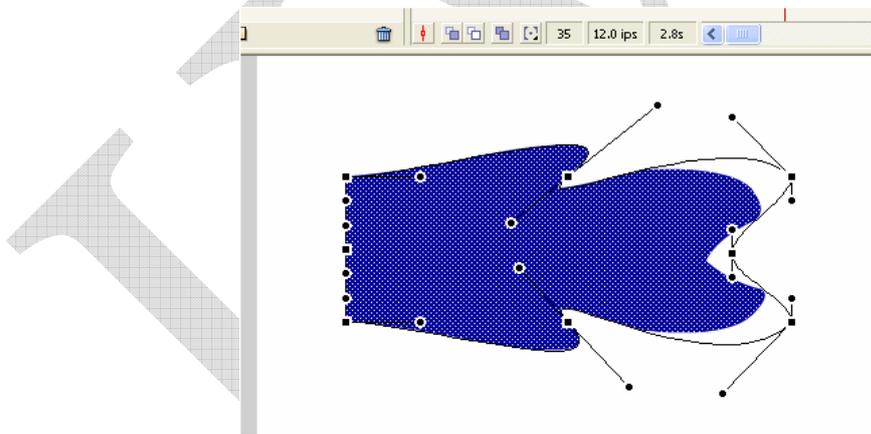
Exercice5 : Morphing

1) Animation par interpolation de forme

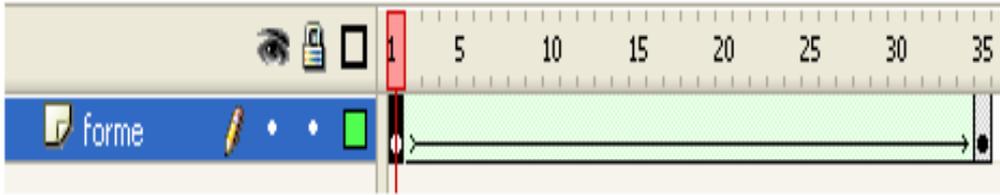
- Créez un nouveau document,
- Dans la palette d'outil ; sélectionnez l'outil rectangle, choisissez une couleur de remplissage.
- Sur la scène, créez un rectangle.
- Prenez l'outil flèche et sélectionnez le contour de l'objet puis supprimez-le.
- Sélectionnez une tranche, à l'image 35 par exemple et faites (Insertion / Image-clé) ou (F6).



- Sélectionnez le rectangle dans l'image 35, dans la palette d'outil, sélectionnez l'outil transformation libre  se trouvant juste à côté de l'outil rectangle, activez l'option enveloppe  qui se trouve en bas de la barre d'outils, et faites les déformations voulues sur l'image finale



- Choisissez une nouvelle couleur pour la forme finale dans l'image 35.
- Revenez et sélectionnez l'image 1 ; dans la palette propriété : pour « Interpolation » choisissez « Forme » puis pour Accélération 10 par exemple.
- Dans l'échelle de temps entre l'image 1 et l'image 35 apparaît une flèche sur fond vert pâle.



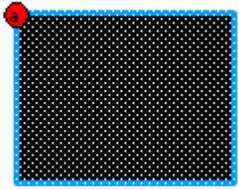
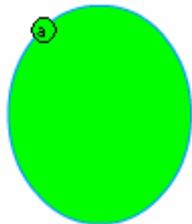
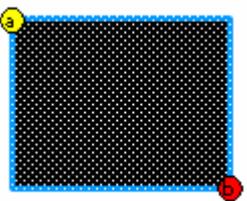
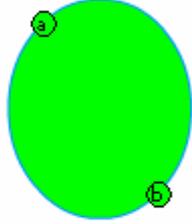
- Tester l'animation

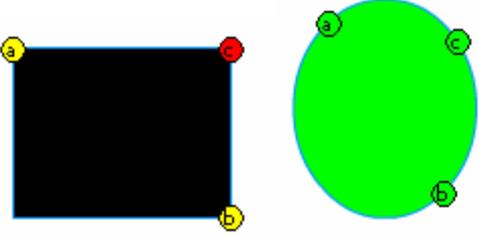
2) Les bases du morphing sous flash.

Pour cet exercice, nous avons repris l'interpolation de formes que nous avons vu précédemment sur laquelle nous allons ajouter des repères de formes vous permettant ainsi de contraindre le trajet d'un point précis du début à la fin de l'animation.

- Créez une animation simple par interpolation de forme du type : rectangle se transformant en rond.

- Allez sur l'image 1, et faites (Modification / Forme/ Ajouter les repères de formes) ou (Ctrl + Maj + H).

<p>1) Déplacez le point (a) comme ci-dessous.</p> 	<p>2) Allez maintenant sur l'image 35, positionnez le point comme ci-dessous.</p> 
<p>3) Revenir sur l'image 1, puis de nouveau ajoutez un repère (Ctrl + Maj + H) et positionnez le point (b) comme ci-dessous.</p> 	<p>4) De nouveau, retournez sur l'image 35 et placez le point (b) comme ci-dessous.</p> 

<p>5) Puis recommencez l'opération d'ajout de repères de formes (Ctrl + Maj + H) ; et placez le point (c) comme suit.</p> 	<p>6) Libre à vous d'ajouter autant de point que vous le souhaitez et aux endroits désirés. Faites des essais pour voir comment réagit l'interpolation. Voilà, pour voir le résultat, faites (Ctrl + Entrée).</p>
---	---

Exercice 6 : Utilisation des masques

Un masque sert à voir un objet à travers la forme d'un autre objet.

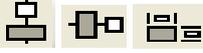
Créer un nouveau fichier, ensuite nous allons préparer deux symboles : le masque et l'objet masqué.

1. Cliquez sur le premier calque actif sur la scène et dans le menu fichier choisissez importez dans la scène. Nous allons sélectionner une image à partir du disque qui va jouer le rôle d'un objet masqué.

2. Une fois l'image importée dans la scène cliquez simultanément sur Ctr+k pour affichez le panneaux « aligner ». On va essayer d'aligner notre image dans le centre de la scène et ajuster sa taille exactement comme la taille de la scène :



a. Activez la touche vers la scène  du panneau aligner.

b. Cliquez sur ces trois boutons : 

c. Ainsi nous avons notre image parfaitement au centre de la scène et de même taille que cette dernière.

3. On va maintenant créer le calque masque. Cliquez sur « ajouter un nouveau calque »  et nommez ce nouveau calque « masque » et nommez le premier calque « image ».

4. Sélectionnez le calque image et dessinez un cercle au centre parfaitement arrondi (astuce quand vous dessinez le cercle maintenez la touche Shift appuyez pour avoir un cercle parfaitement arrondi) qui va jouer le rôle d'un masque.



5. Sélectionnez le contour du cercle et supprimez-le car le contour n'a pas d'effet dans le calque masque

6. Assurez-vous que le cercle soit au centre de notre scène. On peut utiliser le panneau « aligner » et en activant le bouton vers la scène . Vous centrez le cercle au milieu de la scène .

7. Cliquez avec le bouton droit sur le calque masque et choisissez masque.

Le calque masque se transforme d'apparence. De même le calque image pour indiquer que l'un est masque et l'autre est un calque masqué. Aussi les deux calques sont verrouillés automatiquement



Le verrouillage des deux calques permet de voir l'effet du masque en mode création.

8. Vous pouvez tester votre animation en appuyant sur (ctrl+entrée).

Nous allons maintenant procéder à animer notre cercle masque.

Déplacer ou transformer le masque

La première chose à faire est donc de transformer notre masque en symbole. La procédure est toute simple :

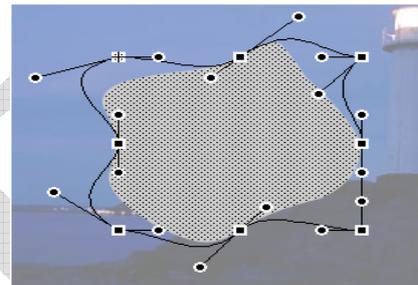
1. Sélectionnez le calque masque et désactivez le verrouillage de ce calque.
2. Sélectionnez le cercle et le convertir en un symbole de type Clip (touche F8) portant le nom de cercle.

- Double cliquez sur le symbole cercle pour le placer dans le contenu du clip cercle.



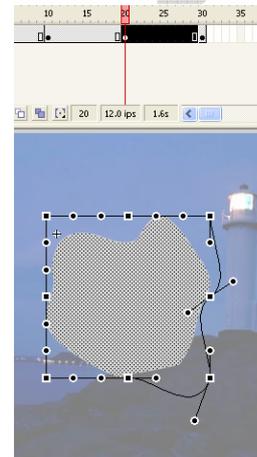
- On va procéder à créer une interpolation de forme du calque cercle, sélectionnez l'image 10 et insérez une image clé (F6).

- Sélectionnez le cercle a ce niveau, et déformez-le avec l'outil de transformation libre ( puis ) pour avoir presque cette forme :

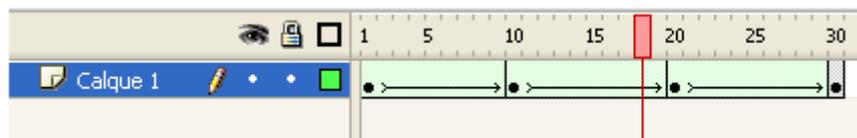


- Insérez maintenant une image clé dans le même calque au niveau de l'image 20, et une autre image clé au niveau de l'image 30

- Sélectionnez l'image clé 20 et déformez de nouveau le cercle avec l'outil de transformation libre ( puis ) pour avoir presque cette forme



- Maintenant vous sélectionnez l'image clef 1 et dans la palette de propriété vous sélectionnez « interpolation de forme ».
- Répétez l'étape 9 de même pour les images 10 et 20 pour avoir toutes les images clé. Puis interpolez-les en forme.

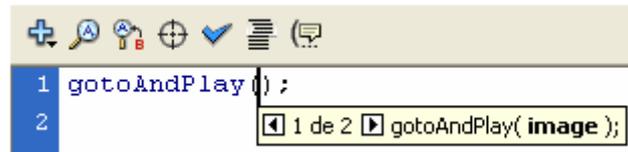


- Maintenant nous allons procéder à ajouter une action script pour que le clip cercle quand il boucle il revient directement à l'image 10 et recommence la lecture. Pour cela sélectionnez l'image 30 de notre animation et ouvrez le

panneau Action script : fenêtre/panneaux de développement/action ou directement ave F9.

Dans le panneau « action script » cliquez sur l'icône  puis sélectionnez successivement : **Fonction globales/contrôle du scénario/gotoAndPlay**.

La fonction gotoAndPlay est tapée maintenant dans le panneau action script et elle attend de vous de taper un paramètre qui correspond au numéro d'image où on doit retourner :

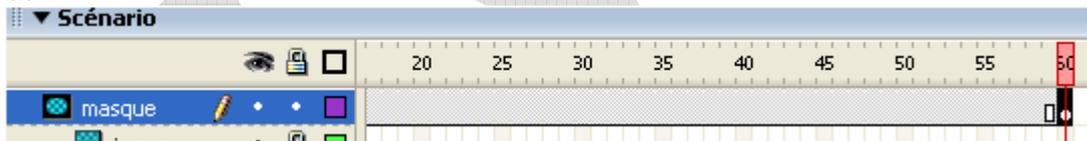


Alors vous tapez la valeur 10 ; qui veut dire l'image 10.

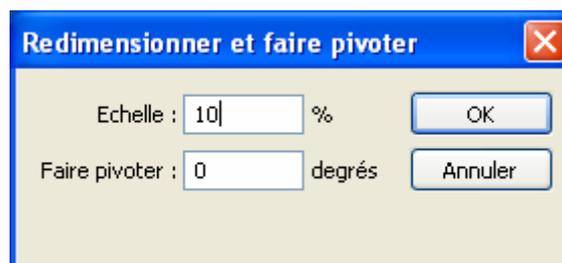
On aura donc cette instruction dans le panneau action script : **gotoAndPlay(10);**

11. Vous pouvez maintenant tester votre animation.
12. Nous allons maintenant animer le clip cercle dans la séquence principale de notre scène. Pour cela, il faut revenir au scénario principal : vous pouvez soit cliquer sur  ou sur l'icône .

13. Dans le calque masque vous insérez une image clef au nouveau de l'image 60 :

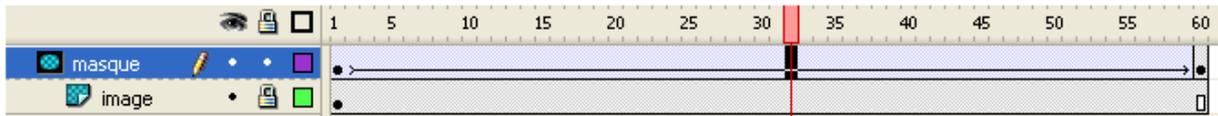


14. Dans l'image 60 du calque on va redimensionner notre clip cercle pour qu'il touche les bords de notre scène. On peut donc utiliser le panneau aligner (ctrl +k) puis vous cliquez .
15. Vous revenez maintenant à l'image 1 du calque masque, vous sélectionnez le symbole cercle et vous cliquez simultanément sur (ctrl+alt+s) ou bien **menu Modification+Transformer+ Redimensionner et faire pivoter** pour afficher le panneau « redimensionner et faire pivoter » et vous tapez devant échelle 10 (pour réduire la taille du cercle de 10%) puis tapez Entrée.



16. puis vous cliquez au milieu de ce calque masque avec le bouton droit de la souris et créez une interpolation de mouvement.
17. maintenant nous allons allonger notre calque image vers l'image 60 il suffit de sélectionner l'image 60 de ce calque puis F5 (insérer une image).

Remarque : on a pu de même interpoler le calque image c'est un choix possible.

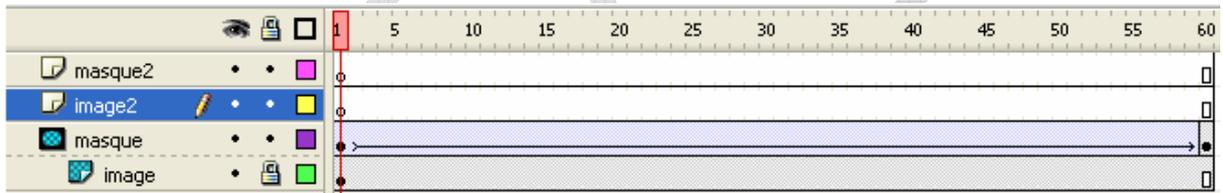


18. tester votre animation.

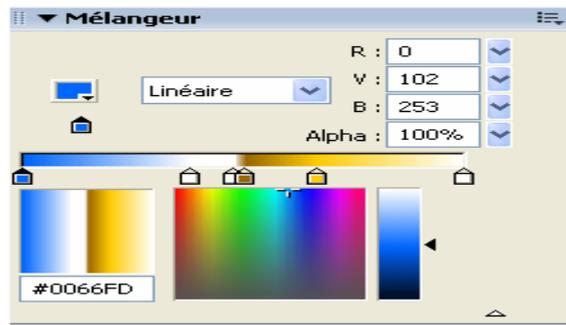
Utiliser du texte, déplacer l'image sous le masque

Il est maintenant grand temps de signer votre composition, et utiliser un texte pour faire un masque est aussi simple que de faire un cercle :

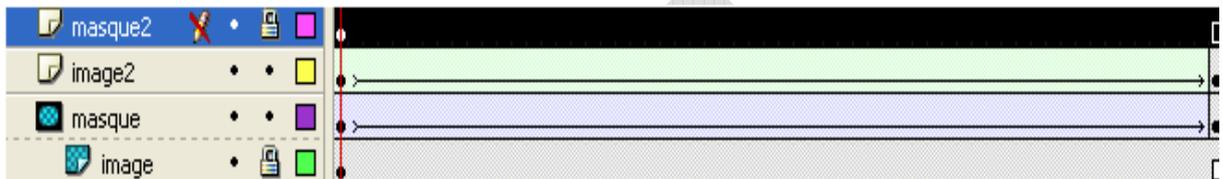
1. Encore une fois, ajoutez 2 nouveaux calques au dessus des 2 précédents, renommez-les « image2 » et « masque2 ».



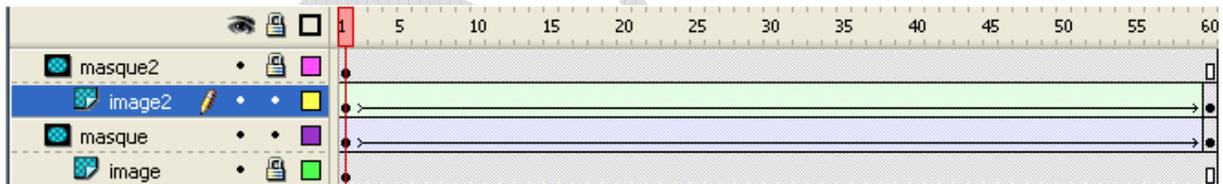
2. Cette fois nous commençons par remplir le calque et « masque2 », cliquez dessus, sélectionnez l'outil texte et écrivez votre nom dans une zone blanche du dessin.
3. Sélectionnez le calque « image2 », (c'est à dire le calque qui va être vu à travers le masque qui a, cette fois, la forme d'un texte) et dessinez un rectangle sans contour dont le fond est un mélange de couleurs. La taille de ce rectangle a presque la largeur de la scène et qui a comme épaisseur celle de notre texte.
4. Insérer une image clé dans le calque masque au niveau 50, sélectionnez l'image 50. Puis modifiez l'ordre de couleurs à l'aide du panneau mélangeur pour avoir une nouvelle disposition de couleurs dégradées.



- sélectionnez l'image 1 du calque image et dans la palette de propriété sélectionnez interpolation de forme.



- Il ne reste maintenant que la couverture du calque « masque 2 » en « masque » et ceci en cliquant avec le bouton droit sur ce calque et en choisissant « calque masque ».



- Testez votre animation.
- Vous remarquez que le masque n'a pas donné son effet, en fait le masque ne peut interpréter que les formes, donc il faut convertir notre texte en une forme. Vous revenez à la scène principale, sélectionnez votre texte puis (ctrl+B) 2 fois pour le séparer.
- testez l'animation.

Exercice 7 : Création d'un bouton avec lien

- Ouvrez un nouveau document, à l'aide de l'outil rectangle, dessinez un rectangle.
- Insertion + convertir en symbole (bouton).
- Double cliquez sur ce symbole pour voir ses différentes fonctions.

4. Faites un clic droit sur la première image-clé « Haut », puis copiez « copier les images », faites un clic-droit dans la deuxième case « Dessus », puis enfin, collez « coller les images ». Vous avez maintenant 2 images identiques.
5. Répétez la même chose pour les fonctions « Abaissé » et « Cliquable ». Allez à la première fonction « Haut », à l'aide de l'outil texte écrire « Haut » sur le bouton.
6. Allez à la deuxième fonction « Dessus », à l'aide de l'outil texte écrire « Dessus » sur le bouton.
7. Allez à la troisième fonction « Abaissé », à l'aide de l'outil texte écrire « Abaissé » sur le bouton.
8. Contrôlez et tester l'animation.

Affectation d'un lien à ce bouton

- Sélectionnez l'objet bouton puis sélectionnez fenêtre, Actions.
Un panneau s'ouvre, Il permet de piloter les modules de programmation Action Script de Flash MX.
- Dans la fenêtre des Actions, allez à actions, contrôle d'animation, on, press pour associer un gestionnaire d'événement à ce bouton.

```
on(Press) {  
    }  
}
```
- Positionnez le curseur à l'intérieur des accolades.
- Allez à Actions Navigateur réseau Double cliquez maintenant sur GetURL.

Vous remarquez un certain nombre d'options en bas du panneau.

- Dans le champ "URL", entrez l'URL à laquelle vous voulez lier votre bouton.
- Dans le champ "fenêtre", laissez-le tel quel si vous voulez que la page appelée se charge dans la fenêtre où se trouve votre bouton. Sinon, sélectionnez "_blank" pour charger la page dans une nouvelle fenêtre.
- Enfin, le champ "variables", laissez-le vide.
- Choisissez Contrôle et Testez l'animation.

Exercice 8 : Réalisation d'un bouton poussoir (un bouton prédéfini et modifiable)

1. Ouvrez un nouveau document et faites ensuite apparaître la fenêtre composants.
2. Cliquez sur l'élément radio PushButton et posez-le sur la scène.
3. Insertion, convertir en symbole, comportement clip.
4. Faites ensuite apparaître la fenêtre des propriétés. Remplissez le champ « Label » et validez.

5. Associez ensuite un gestionnaire d'événements à ce bouton en informant le champ « Click Handler » avec le nom de ce gestionnaire : "OnClickEvent" par exemple. A chaque clic sur le bouton, ce gestionnaire d'événements sera exécuté.

6. Sélectionnez à présent l'image clé 1 et faites apparaître la fenêtre « Actions ».

Passez en mode d'édition « expert », (en haut à droite de la fenêtre). Indiquez ensuite :

```
onClickEvent (mouseDown) {
    trace("bien je valide !");
}
```

7. contrôle + test movie

Exercice 9 : Modifier l'icône de la souris.

1. Importez une image (Fichier + Importer) de votre disque dur.

Remarque : voyez la différence entre Importer et Importer dans la bibliothèque.

2. Faites apparaître la bibliothèque (Ctrl + L). Faites glisser l'image qui se trouve dans votre bibliothèque vers votre scène.

3. Sélectionnez l'image et convertissez-la en clip (Insertion + Convertir en symbole et nommez le zip).

4. Supprimez l'image de votre scène.

5. Faites glisser le clip qui se trouve dans votre bibliothèque vers votre scène.

6. Sélectionnez votre image.

7. Placez-vous au frame1 de votre calque et Insérez les actions suivantes :

```
Mouse.hide();
startDrag(zip,true);
```

L'action "Mouse.hide();" permet de cacher l'icône de la souris. Pour l'action "startDrag" mettez zip dans le champ "Cible" (cela signifie que c'est le clip "zip" qui devra suivre les mouvements de la souris).

Exercice 10 : Contrôle de clip à l'aide de Action script

Il est possible de contrôler l'animation d'un clip à l'aide d'Action Script. Dans l'exemple suivant nous arrêtons le clip quand l'utilisateur clique sur le clip, et nous le faisons continuer lorsqu'une touche est pressée.

- Créez une scène avec un clip. Affichez la boîte de dialogue d'action sur les objets pour le clip en question. Pour être sûr de bien avoir sélectionné le bon clip, vous pouvez y arriver en cliquant sur la flèche bleue en bas à droite du panneau "propriété".

- Dans la boîte de dialogue, cliquez sur "action" puis "onClickEvent" qui permet de récupérer les évènements liés aux clips. Vous remarquez que du code accompagné de certaines clauses

apparaît. Dans notre cas vous pouvez choisir lequel écrire. En cliquant sur le code, vous remarquerez que vous pouvez changer les évènements

- Il faut maintenant décider ce qu'il faut faire lorsque l'évènement sera récupéré. Dans notre cas nous voulons stopper le clip si on clique sur entrée, sélectionnez alors "stop".

- Lorsque vous cliquez sur une action, le code est ajouté juste après la ligne bleue foncé qui indique la ligne sélectionnée.

```
onClipEvent (keyDown) {
```

```
Stop() ;
```

```
}
```

- Nous allons maintenant faire rejouer le clip lorsque l'on presse une touche. L'évènement à capturer est alors "touche enfoncée" (différent de touche relâchée), et l'action est "play".

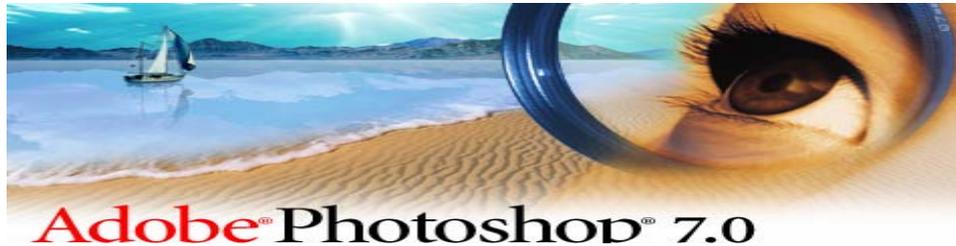
TP 2.2 :
LE MULTIMEDIA

UVT

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

II.3. Adobe Photoshop 7

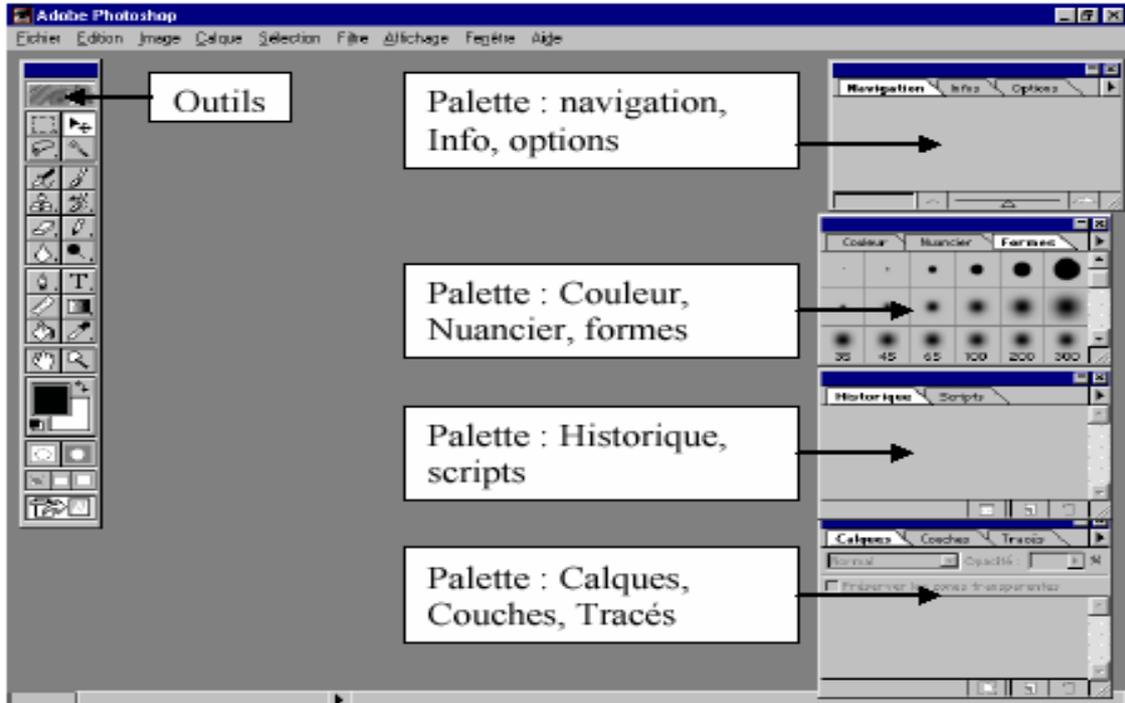


II.3.1. Présentation de Photoshop

Photoshop offre une panoplie complète d'outils de correction, de peinture, de dessin et Web qui permettent d'effectuer tous les travaux de retouche d'image avec efficacité et c'est grâce aux fonctionnalités de la palette Historique et des effets de calque modifiables.

II.3.2. L'interface Photoshop

L'interface photoshop se compose d'un menu classique dans lequel on retrouve les fonctions mais aussi des palettes flottantes dont l'utilisation est primordiale et facilite la prise en main du logiciel. Ils sont activables par le menu « fenêtre ».



A/ Les outils

Ils sont divisés en plusieurs catégories qui sont actifs suivant le contexte d'utilisation.

<p>✓ Outils de sélection</p> <p>Outils de sélection; Outil de déplacement; Outils lasso; Baguette magique; Plumes;</p> 	<p>✓ Outils à dessin</p> <p>Aérographe; Pinceau; Tampons; Forme historique; Gomme à effacer; Crayon; Outils de netteté; Outils de densité;</p> 	<p>✓ Les autres outils</p> <p>Outils de texte; Outil de mesure; Outils de dégradé; Pot de peinture; Pipettes; Main; Loupe.</p> 
---	--	---

B/ Les palettes

Elles permettent de travailler rapidement sur les propriétés des images sans passer par la barre de menu.

- **Navigation** : Permet de faire le zoom sur l'image et de positionner la zone visible.
- **Info** : Donne la position du curseur, la couleur de l'image à la position du curseur et , lors d'une sélection, donne la hauteur et la largeur de la sélection.
- **Option** : Permet de définir les options des outils (après sélection d'un outil).
- **Couleur** : Permet la création de nouvelles couleurs.
- **Nuancier** : Permet de choisir parmi une sélection de couleurs prédéfinies.
- **Formes** : Permet de choisir la forme de trait du crayon, pinceau et aérographe. Il est aussi possible de créer ses propres formes.
- **Historique** : Conserve l'historique des actions posées. Il permet de supprimer une action, de créer un nouveau document et de créer des copies.
- **Scripts** : Permet de crée des commandes répétitives.
- **Calque** : Permet d'ajouter, supprimer et de naviguer sur les calques. Permet aussi de définir les options d'affichage des calques.
- **Couche** : Permet de créer et supprimer une couche.
- **Tracé** : Permet de créer, manipuler et supprimer les tracés.

II.3.3. Les exercices d'application

Exercice 1 : Scanner une image et changer sa taille et/ou son extension

- 1) Scannez une image.
- 2) Choisissez Image + taille de l'image.
Changez la taille de l'image.
- 3) Puis Fichier + enregistrer pour le web.

Exercice 2 : Bordure de photo - Déchirure

- 1) Ouvrez une image dans un calque au dessus du fond. Augmentez la taille de travail de l'image (Image + Taille de la zone de travail) de 120 %.
- 2) Activez le calque image et choisissez Calque + Ajouter un masque de fusion + Tout faire apparaître.
- 3) Puis faites ctrl-clic sur l'image pour charger une sélection de l'image. Puis diminuez la sélection avec Sélection + Modifier + Contracter.

Intervertissez la sélection et remplissez par la couleur du fond puis désélectionnez.

- 4) Maintenant on travaille sur le masque de fusion. Choisissez un filtre + contour +

effets pointilliste.

5) Ensuite vous pouvez appliquer des effets d'ombres.

Exercice 3 : Texture

1) Ouvrez un document, choisissez une couleur de premier plan et remplissez le calque fond.

2) Choisissez un filtre de bruit Filtre + Bruit + Ajout de bruit pour créer des variations de couleur.

3) Appliquez une série de filtre jusqu 'à obtenir l'effet recherché. Regardez les effets du filtre : Filtre + Nuages par différences.

Exercice 4 : Motif

1) Ouvrez un document de petite taille (200*200) et dessinez votre motif.

2) Appliquez un filtre de déplacement. Filtre + Autres + Déplacement avec 80 comme valeur de déplacement. Ceci permet de dessiner à l'emplacement des bords de l'image.

3) Complétez votre dessin, définissez-le comme motif et appliquez-le sur une image pour le tester.

4) Utilisez des filtres différents pour modifier le motif.

Exercice 5 : Texture Effets de texte

1) Dans un document avec un fond noir. Tapez votre texte en noir.

2) Sélectionnez le texte (Ctrl-Click dans le calque du texte) et créez un nouveau calque en dessous du calque Texte. Augmentez la sélection (Sélection + Modifier + Agrandir) puis arrondissez la sélection (Sélection + Modifier + Adoucir). Remplissez la sélection de blanc.

3) Créez un nouveau calque en dessous du calque blanc, augmentez et adoucissez encore la sélection puis remplissez-la de jaune. Refaites encore une fois cette étape avec une plus grande augmentation et avec la couleur verte.

4) Désélectionnez et appliquez un filtre de flou gaussien de plus en plus grand sur les calques blanc, jaune et vert.

Exercice 6 : Effet de Gravure

1) Choisissez une texture d'arrière plan. Créez un texte blanc dans un nouveau calque.

2) Dupliquez le calque de fond, placez l'opacité sur 80 et placez-le en première position. Associez-le avec le calque précédent.

3) Créez un nouveau calque pour les ombres. Remplissez-le en noir. Sélectionnez le texte (Ctrl clique sur le calque texte). Et remplissez la sélection de blanc. Appliquez le filtre de translation (Filtre + Divers + Translation) avec 4 et 4. Choisissez Filtre + Atténuation + Flou Gaussien. Ces filtres créent le détail et la profondeur.

4) Associez le calque au calque précédent (Ctrl-G). Réglez l'opacité sur 80 et son mode sur produit.

5) Créez un nouveau calque pour les lumières. Faites un Ctrl-Clic sur le texte pour avoir la sélection. Décalez la sélection (Sélection + transformer) d'un pixel vers la droite et d'un pixel vers la bas à l'aide du clavier. Combinez cette sélection avec une nouvelle sélection du texte en faisant Ctrl-Alt-Clic sur le calque texte.

6) Remplissez cette sélection de blanc et réglez le mode du calque sur superposition.

TP 2.3 :
LE MULTIMEDIA

UVT

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

II.4. Cool Edit Pro

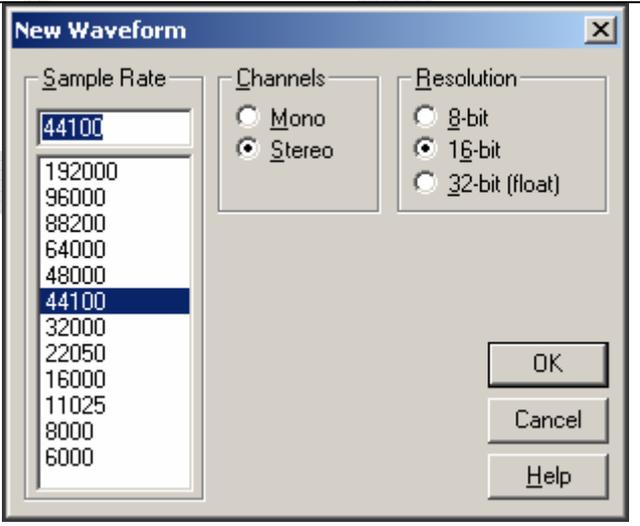
II.4.1. Présentation de Cool Edit Pro

Cool Edit est un logiciel de musique électronique, qui permet de modifier les sons, modulations, distorsion, amplification des sons, simulation d'échos en 3D et modification de temps.

II.4.2. Les exercices d'application

Exercice 1 : Capture d'un son (bruit, parole, etc.) quelconque

- 1) Configurez un microphone.
- 2) Lancez le logiciel.
- 3) Cliquez sur l'icône Record  pour capturer un son quelconque.

<ol style="list-style-type: none"> 4) Choisissez la qualité de la numérisation : <ul style="list-style-type: none"> - <i>La fréquence d'échantillonnage</i> : (Sample Rate) généralement égale à 44,1 KHz pour une qualité CD. - <i>Le nombre de canaux</i> (Channel) : 1 canal (son mono) ou bien 2 canaux (son stéréophonique.) - <i>Le nombre de bits par échantillon</i> : (Resolution) généralement égale à 16 bits pour une qualité CD 	
---	--

- 5) Cliquez sur OK
- 6) Capturez une parole quelconque puis cliquez sur Record.
- 7) File + Save as et choisissez par exemple l'extension .wav.

Exercice 2 : Compresser et changer l'extension d'un fichier sonore

- 1) Ouvrez le fichier que vous avez numérisé dans l'exercice 1.
- 2) Sauvegardez ce fichier en tant que fichier mp3.
- 3) Testez la différence de taille.

TP 3 :

DEVELOPPEMENT WEB

UVT

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Partie III : Développement Web

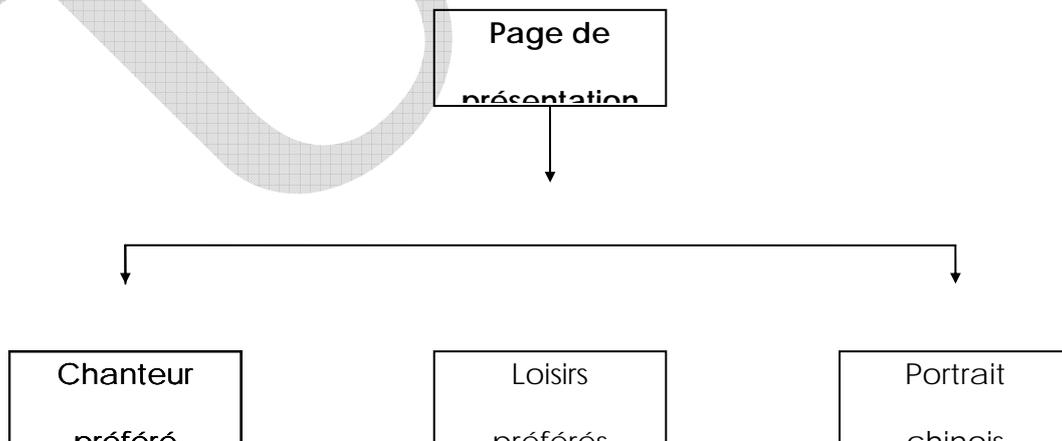
La réalisation d'un site WEB avec DreamWeaver 8



Macromedia DreamWeaver est un logiciel professionnel destiné à la conception, au développement et à la gestion de sites Web. Ses fonctions d'édition visuelle permettent de créer rapidement des pages sans rédiger une ligne de code.

A/ Plan du site

Toujours ce site comprendra une page de présentation de l'auteur et 3 "sous-pages" consacrées à son chanteur préféré, ses loisirs préférés et un portrait chinois.

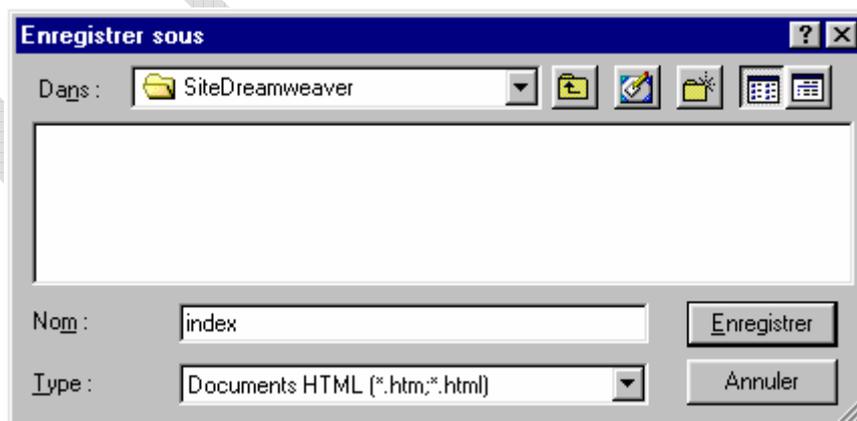


B/ Mise en place des pages du site

- Nous mettrons tous les éléments du site (fichiers Html, images, etc) dans le même répertoire. Créez un nouveau répertoire ou dossier que l'on appellera ici "SiteDreamweaver".



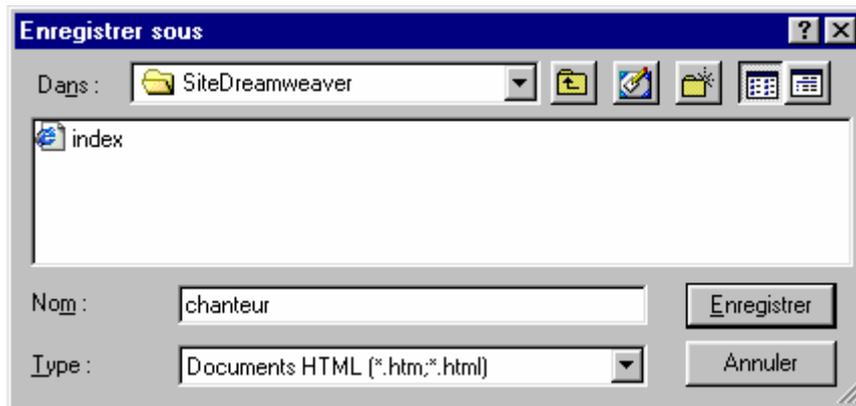
- Ouvrez le programme Dreamweaver.
Démarrer → Programmes → Macromedia Dreamweaver → Dreamweaver
- Fermons toutes ces petites fenêtres qui encombrant l'écran pour ne garder que la fenêtre principale.
- On va tout de suite enregistrer la page ouverte par défaut : **Fichier → Enregistrer sous...**
Comme indiqué sur la capture d'écran, désignez le chemin du dossier pour enregistrer le fichier **Dans** "SiteDreamweaver" et donnez comme **Nom** "index". Ce fichier (la page de présentation) sera la page d'accueil [ou homepage] de votre site. Par convention, on appelle toujours ce fichier "index.htm" ou "index.html". C'est le fichier ouvert par défaut par les navigateurs Internet.



Nous avons créé ainsi le fichier (vide) de notre page d'accueil.

- On reprend la même procédure pour les autres fichiers. Commençons par la page consacrée au chanteur préféré. Un conseil, utilisez toujours des minuscules pour vos noms de fichiers.

Faites **Fichier** → **Nouveau**, puis **Fichier** → **Enregistrer sous...** et toujours **Dans** "SiteDreamweaver" donnez comme nom au fichier le **Nom** de "chanteur".



- Faites de même pour la page consacrée aux loisirs préférés : **Fichier** → **Nouveau**, puis **Fichier** → **Enregistrer sous ...** toujours **Dans** "SiteDreamweaver" avec comme nom de fichier le **Nom** de "loisirs".

Et enfin pour la page consacrée au portrait chinois :

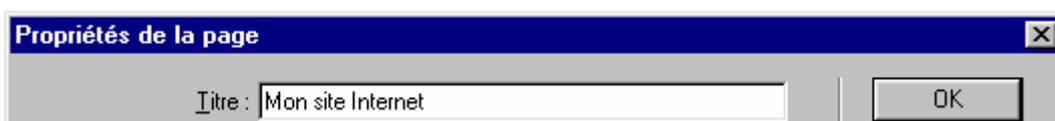
Fichier → **Nouveau**, puis **Fichier** → **Enregistrer sous ...** toujours **Dans** "SiteDreamweaver" avec comme nom de fichier le **Nom** de "chinois".

C/ La page de présentation (index.htm)

- Revenons à la page de présentation. Pour ce faire, on peut (notamment) passer par **Fichier** → **Ouvrir** → dossier "SiteDreamweaver" → index.htm



- Prenons la bonne habitude de donner un titre à la page. Nous verrons plus tard que ce titre s'affichera tout en haut dans la fenêtre du navigateur. **Modifier** → **Propriétés de la page** et donnons comme **Titre** "Mon site Internet". **OK**.

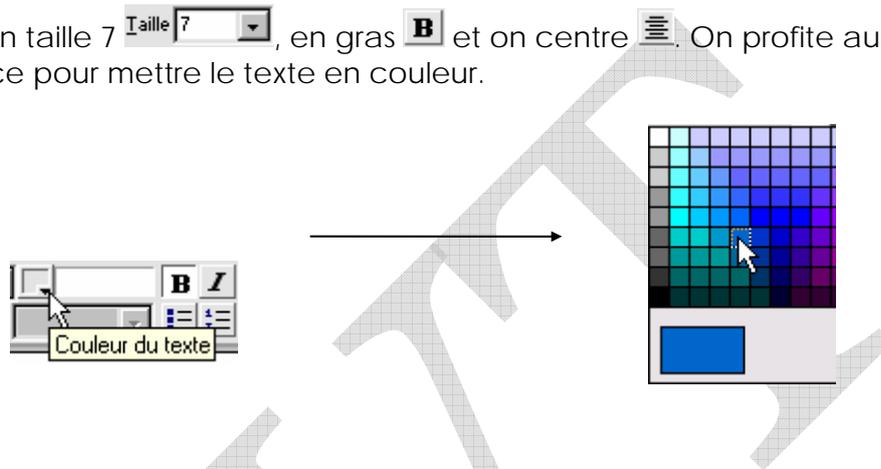


- Tapez maintenant dans la fenêtre principale : "Mon site Internet" que l'on met en surbrillance.
- On appelle la barre des propriétés. **Fenêtre → Propriétés** (ou **Modifier → Propriétés de la sélection** ou Ctrl + F3).

Mon site Internet



On met en taille 7 , en gras **B** et on centre . On profite aussi de la surbrillance pour mettre le texte en couleur.



- On ferme la fenêtre des propriétés et on passe à la ligne (nouveau paragraphe = Enter).
- On entre ses coordonnées soit :

Mohamed Tounsi (nouvelle ligne = Maj + Enter)

69, Rue les fleurs (Maj + Enter)

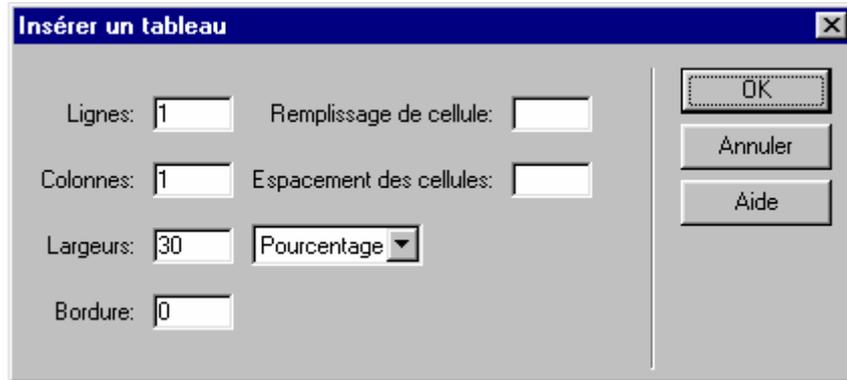
3000 Sfax (Maj + Enter)

Tunisie

- On peut modifier le format de ce texte en appelant la barre de propriétés (si elle n'est déjà pas présente). Mettre le texte en Surbrillance. Puis **Fenêtre → Propriétés...**
On prend par exemple, une taille 5 et on désactive le bouton "Gras". Après avoir refermé la barre de propriétés, on passe à un nouveau paragraphe [enter].
- On va maintenant créer un tableau (invisible) d'une ligne et d'une colonne. Les tableaux sont des éléments très importants pour le design d'une page Web et leur

usage est donc très fréquent.

Insertion → **Tableau** et reproduisez les données suivantes.



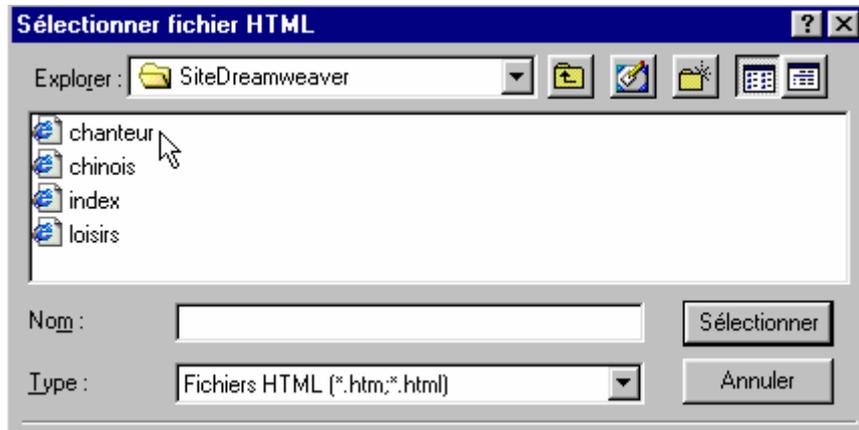
- On clique dans la zone du tableau et on appelle la barre de propriétés (**Fenêtre** → **Propriétés**).

Dans celle-ci, on demande une liste à puces .

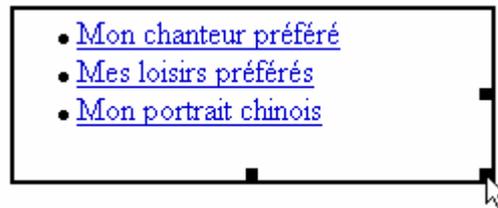
On encode alors :

- Mon chanteur préféré
- Mes loisirs préférés
- Mon portrait chinois

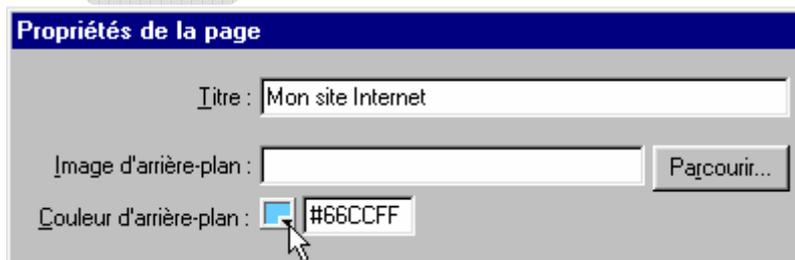
- Nous allons maintenant effectuer les liens. Nous allons relier "Mon chanteur préféré" à la page "chanteur.htm". Mettez "Chanteur préféré" en surbrillance et cliquez l'icône de dossier pour déterminer la cible.



- On procède de même pour "Mes loisirs préférés" et pour "Mon portrait Chinois".
- On peut modifier directement la taille et la position du tableau en cliquant sur sa bordure.



- On peut aussi ajouter un fond de couleur à la page. **Modifier** → **Propriétés de la page**.



Veillez que votre couleur de fond soit toujours compatible avec votre couleur de texte. Votre texte et votre présentation doivent toujours rester lisibles.

- Enregistrez votre travail. **Fichier → Enregistrer** ou **Fichier → Enregistrer tout**

D/ La page "Mon chanteur préféré" (chanteur.htm)

- Passez par **Fichier → Ouvrir** pour ouvrir la page chanteur.htm.
- Nous n'avons pas encore donné de titre à la page, ce que nous faisons de suite.

Modifier → Propriétés de la page et donnons comme **Titre** "Mon chanteur préféré".

- Pour faciliter la vie du lecteur, on inclut des "outils de navigation" pour lui permettre de se mouvoir dans votre site sans passer par les boutons "Précédent" et "Suivant" de son navigateur Internet. Dans le cadre de ce travail, nous utiliserons, dans un premier plan, des mots pour faire les liens mais il est plus sympathique à l'avenir de prévoir des petites images.

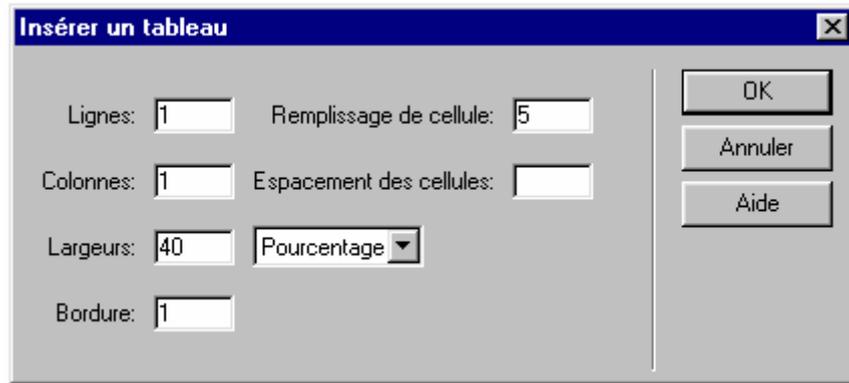
Tapez "Page d'accueil". Surbrillance. Appelez la barre de propriétés (si elle n'est déjà pas présente) et alignez à droite .

- Profitons de la surbrillance pour effectuer également le lien vers la page de présentation.
Activation de l'icône de lien , puis sélection dans le dossier "SiteDreamweaver" du fichier index.htm.

- Créons un nouveau paragraphe (enter) et demandons l'alignement au centre .
On encode le titre (à ne pas confondre avec le titre de la page Web, celui qui apparaît dans le haut de la fenêtre du navigateur) soit "Mon chanteur préféré".

On peut lui donner la taille 6, le mettre en gras et lui donner une couleur en utilisant la barre de propriétés.

- Nous allons insérer un tableau. Menu **Insertion → Tableau**.
Entrez toutes les spécifications comme ci-dessous.



La case "Remplissage de cellule" entourera le texte des bords du tableau de 5 pixels d'espace libre.

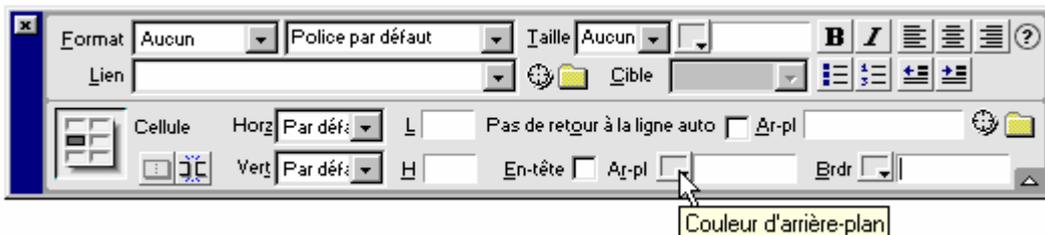
- Si votre tableau n'était pas bien centré, demandez l'alignement centré dans la barre de propriété du tableau. Si celle-ci n'était pas présente, elle peut s'obtenir en cliquant sur le bord inférieur du tableau.



- Et si nous donnions un fond de couleur au tableau ? Lorsque votre curseur est dans le tableau, cliquez sur la petite flèche au bord inférieur droit de la barre de propriété.



La barre de propriété vous propose des options supplémentaires. On peut y définir la couleur d'arrière-plan du tableau.



Vous encodez alors dans la cellule du tableau vos préférences concernant votre chanteur préféré et les commentaires qu'il vous inspire.

- N'oubliez pas que l'on peut modifier directement la largeur du tableau pour l'adapter à votre texte.
- On enregistre son travail. **Fichier → Enregistrer** ou **Fichier → Enregistrer tout**.

E/ Premier test dans le navigateur

- Vous pouvez déjà visualiser votre site sous un navigateur. Dreamweaver vous offre la possibilité de le faire directement sans avoir à ouvrir votre navigateur. Soit **Fichier → Aperçu dans le navigateur → iexplore**. Sinon passez au préalable par "Editer la liste des navigateurs".
- Vous devez avoir ce résultat :

<p style="text-align: center;">Mon site Internet</p> <p style="text-align: center;">Mohamed Tounsi 69, Rue les fleurs 3000 Sfax Tunisie</p> <ul style="list-style-type: none"> • Mon chanteur préféré • Mes loisirs préférés • Mon portrait chinois 	<p style="text-align: right;">Page d'accueil</p> <p style="text-align: center;">Mon Chanteur préféré</p> <div style="border: 1px solid black; padding: 10px; margin: 10px auto; width: 80%;"> <p style="text-align: center;">Jacques Brel</p> <p style="text-align: center;"><i>"la pluie est traversière"</i></p> </div>
---	---

Vérifiez en cliquant sur "chanteur préféré" de la page de présentation, si le lien vers "chanteur" est bien actif. De même, dans la page chanteur, en cliquant sur "page d'accueil", vous devez revenir à la page index.

F/ La page "Mes loisirs préférés" (loisirs.htm)

- Passez par **Fichier → Ouvrir** pour ouvrir la page loisirs.htm.
- Donnons un titre à la page. **Modifier → Propriétés de la page** et donnons comme **Titre** "Mes loisirs préférés".
- On n'oublie pas de mettre les outils de navigation. Tapez "Page d'accueil". Surbrillance. Appelez la barre de propriétés (si elle n'est déjà pas présente) et alignez à droite . Activation de l'icône , sélection dans le dossier "SiteDreamweaver" du fichier index.htm.
- Créons un nouveau paragraphe (enter) et demandons l'alignement au centre . On encode le titre soit "Mes loisirs préférés". On peut lui donner la taille 6, le mettre en gras et lui donner une couleur pour garder une certaine identité graphique par rapport aux autres pages.
- Notre but sera de réaliser la présentation suivante.

Le foot

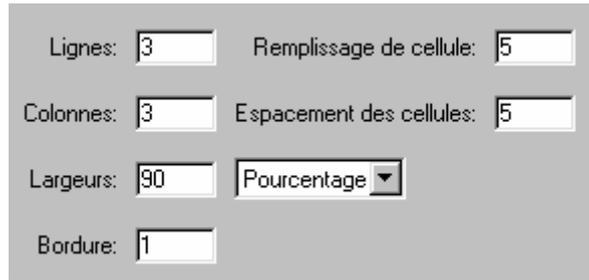
Les animaux

La lecture

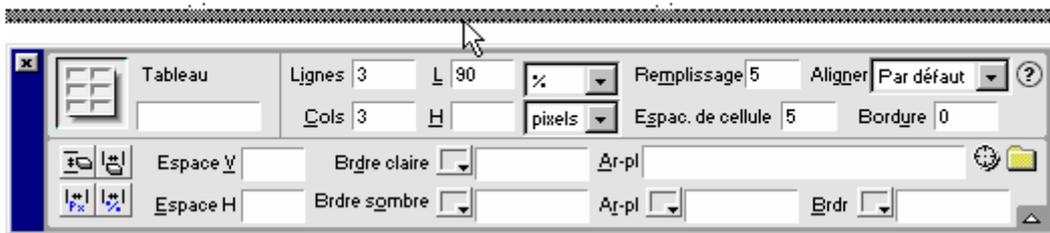
Nous allons insérer un tableau. Passons par un autre moyen pour afficher la barre de propriétés. Par **Fenêtres → Objets**, on affiche la barre des objets et on clique sur l'icône qui représente un tableau.



- On demande un tableau de 3 lignes et 3 colonnes. Indiquez aussi les autres spécifications comme dans la capture d'écran.



- En cliquant sur un bord du tableau, une barre de propriétés spécifique au tableau apparaît. Si votre tableau n'est pas centré, profitez maintenant pour demander l'alignement centré.



- Cliquez sur la première cellule en haut à gauche. Entrez votre texte que vous souhaitez. Vous aurez remarqué que la barre de propriétés a changé pour une barre spécifique aux cellules.



On donne au texte les attributs de taille, d'alignement et de présentation dans la première partie de la barre de la façon qui vous est déjà habituelle. Dans la seconde partie de la barre, on peut donner une couleur d'arrière-plan à la cellule et déterminer la couleur du bord de la cellule.

- On procède de même pour la cellule centrale et la cellule inférieure droite.

- Et nous terminerons par l'insertion d'une ligne horizontale.

Insertion → Barre horizontale ou Fenêtres → Outils et



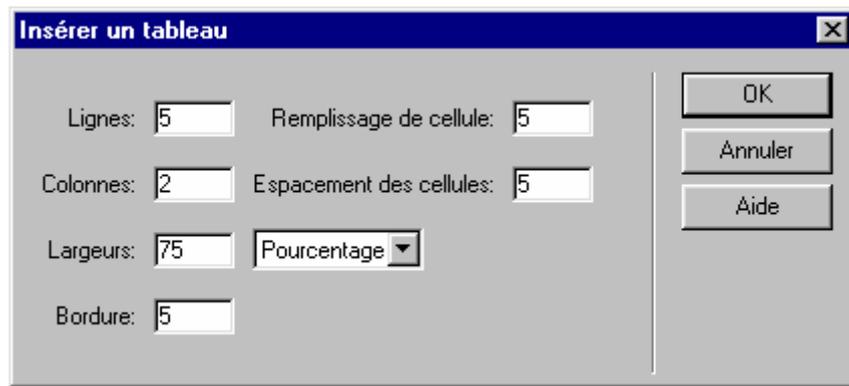
- Ce qui vous donne comme résultat la ligne suivante.



- Enregistrez votre travail. **Fichier → Enregistrer** ou **Fichier → Enregistrer tout**

G/ La page "Mon portrait chinois" (chinois.htm)

- Passez par **Fichier → Ouvrir** pour ouvrir la page chinois.htm.
- Donnons un titre à la page. **Modifier → Propriétés de la page** et donnons comme **Titre** "Mon portrait chinois".
- On n'oublie pas de mettre les outils de navigation. Tapez "Page d'accueil". Surbrillance. Appelez la barre de propriétés (si elle n'est déjà pas présente) et alignez à droite . Activation de l'icône , sélection dans le dossier "SiteDreamweaver" du fichier index.htm.
- Créons un nouveau paragraphe (enter) et demandons l'alignement au centre . On encode le titre soit "Mon portrait chinois". On peut lui donner la taille 6, le mettre en gras et lui donner une couleur pour garder une certaine identité graphique par rapport aux autres pages.
- On insère un tableau de 5 lignes et 2 colonnes selon la procédure maintenant habituelle. Entrez les autres spécifications suivantes.



- Vous obtenez un tableau qui prend la forme ci-dessous que vous complétez..

Couleur préférée	Le bleu
Acteur préféré	Yves Montand
Actrice préférée	Pamela Anderson
Chiffre préféré	Le 7
Pays préféré	L'Espagne

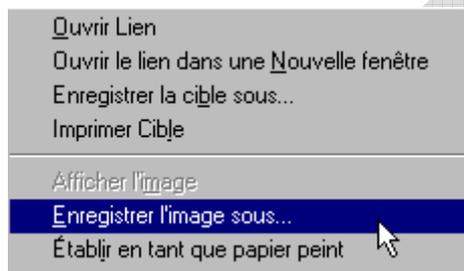
- Enregistrez tout votre travail (oubli classique). **Fichier** → **Enregistrer tout**.
- Votre exercice sur la création d'un site est ainsi terminé.
- Ouvrir Microsoft Explorer. Pour ce faire cliquer son icône. Pour rappel, puisqu'il s'agit d'un fichier local, on passera par **Fichier** → **Ouvrir** → **Parcourir** → **chemin** d'accès au répertoire "SiteDreamweaver" → **fichier** index.htm → **Ouvrir**.

H/ Les images

Les images sur Internet doivent être au format gif ou jpeg.

Pour convertir le format bmp des traitements de texte, il vous faudra passer par un programme graphique pour faire les conversions. (exemple Paint Shop Pro 5, téléchargeable sur Internet). Les versions récentes comprennent même un petit programme "Animation Shop" pour faire des images gif animées du meilleur effet.

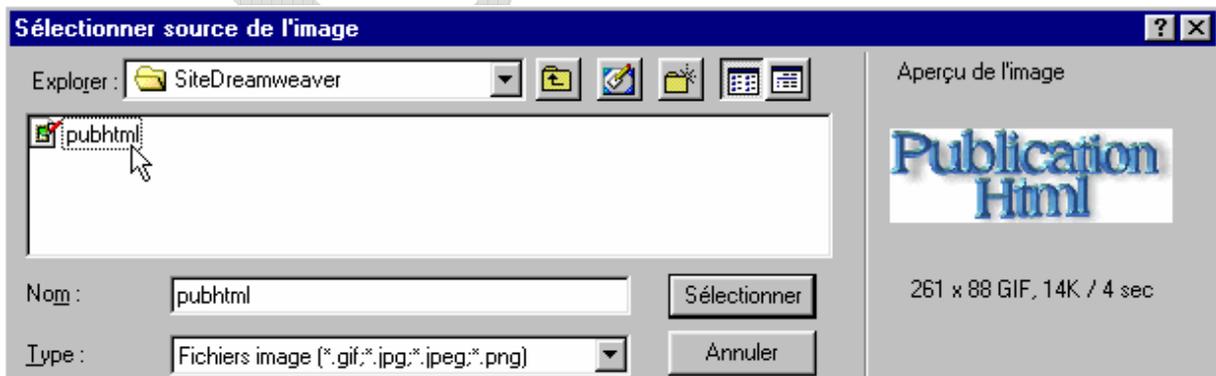
Pour commencer, vous pouvez toujours récupérer des images au cours de vos surfs sur le Web. Il suffit de cliquer sur l'image, **bouton droit** de la souris → **Enregistrer l'image sous...**



Une fois que vous avez votre image au format correct, enregistrez-la dans le même répertoire que vos fichiers Html (ici "Site Dreamweaver").

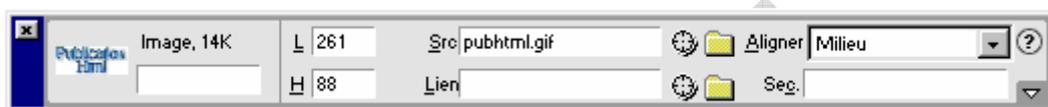
Pour inclure l'image dans votre page Html, il suffit de faire **Insertion** → **Image** ou cliquer sur l'icône  de la barre d'outils.

Sélectionnez l'image.



Dreamweaver vous propose différentes informations par ailleurs fort utiles : un aperçu de l'image, ses dimensions (ici 261 x 88), son format (ici GIF) et une estimation du temps de chargement (ici 4 sec.).

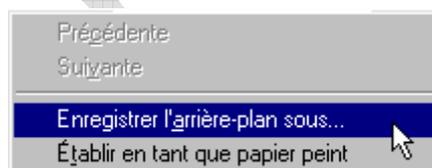
Dans la barre de propriétés de l'image, remarquons l'option "Aligner" pour l'alignement de l'image et l'option "Lien" pour associer un lien éventuel à votre image.



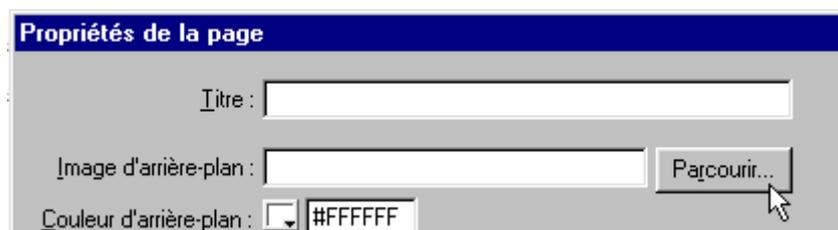
- Les images d'arrière-plan

Avant de quitter les images, le langage Html vous offre la possibilité de mettre de petites images en arrière-plan [background] qui viendront tapisser votre fond de page en mosaïque.

Ces images peuvent aussi être récupérées sur le Web. Il suffit de cliquer sur le fond de la page, **bouton droit** de la souris → **Enregistrer l'arrière-plan sous...**



Pour l'inclure dans votre page, **Modifier** → **Propriétés de la page**



- Les polices de caractères

Dans l'état actuel de la technologie, le navigateur n'affichera réellement que les polices de caractères qui existent sur l'ordinateur du visiteur de votre site. Il serait illusoire d'utiliser des polices de caractères très sophistiquées que votre "client" ne possède peut-être pas sur sa machine.

- Le code source Html

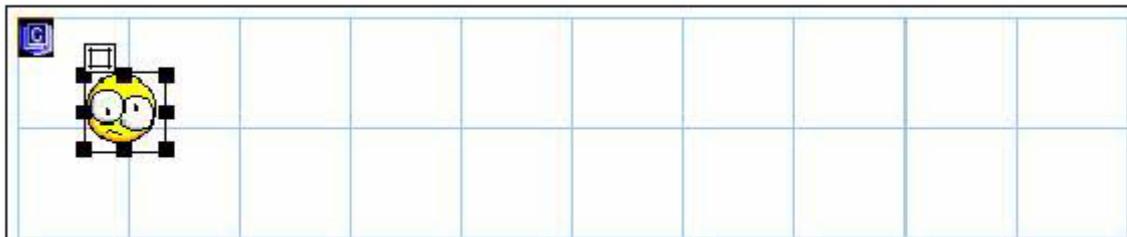
Vous pouvez à tout moment voir (et modifier) le code source Html de votre page que vous éditez sous Dreamweaver en cliquant sur  au bas de la fenêtre principale ou sur l'icône suivante de la barre du "Lanceur" (**Fenêtres → Lanceur**).



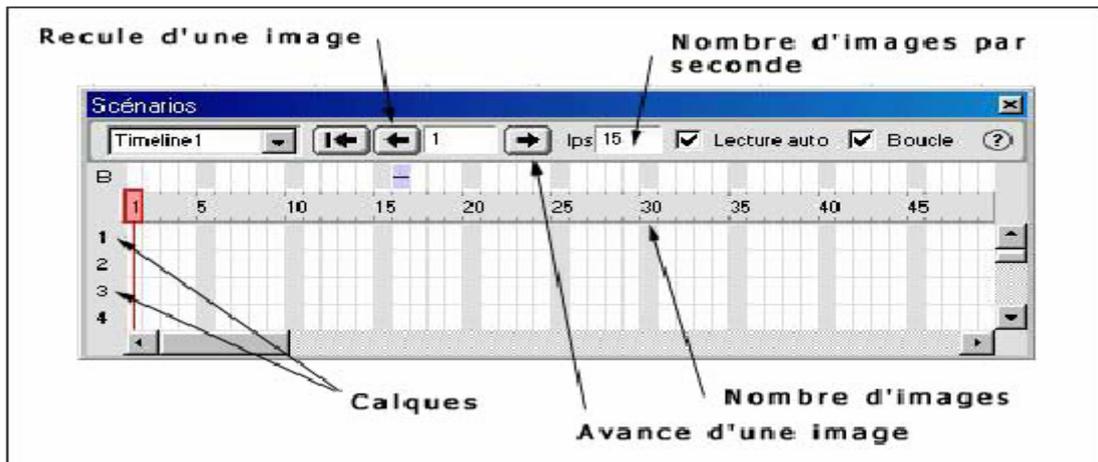
Il serait très pédagogique de consulter ce code Html après chaque opération (centrage, mise en gras, alignement droite, création de tableau).

I/ Les calques

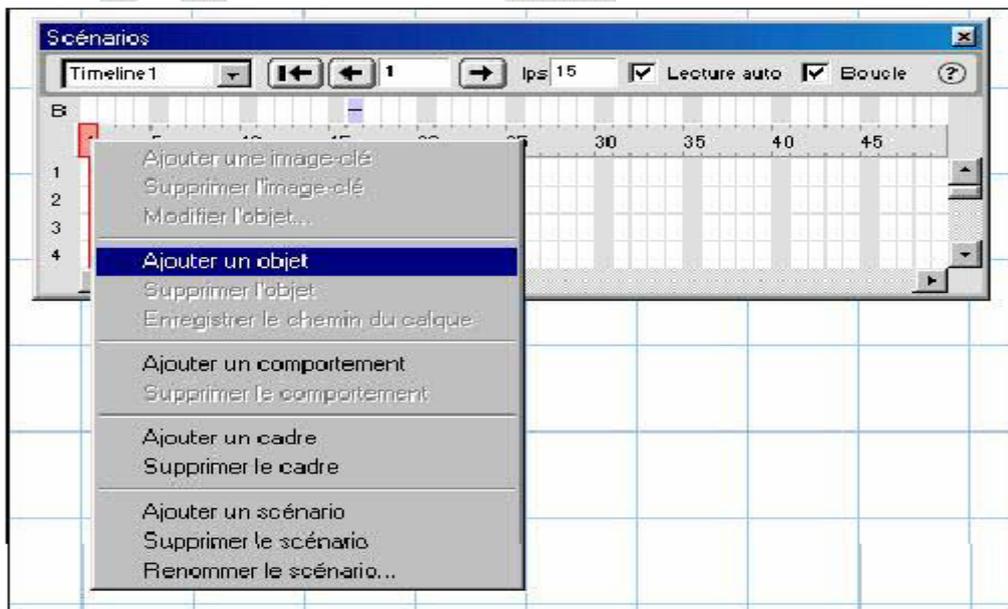
- **Etape 1** : Commencez par créer un calque avec l'icône  et placez-y votre image.



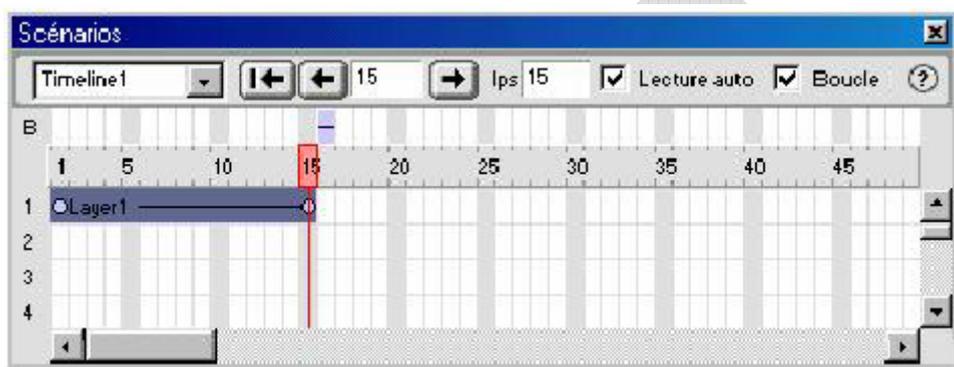
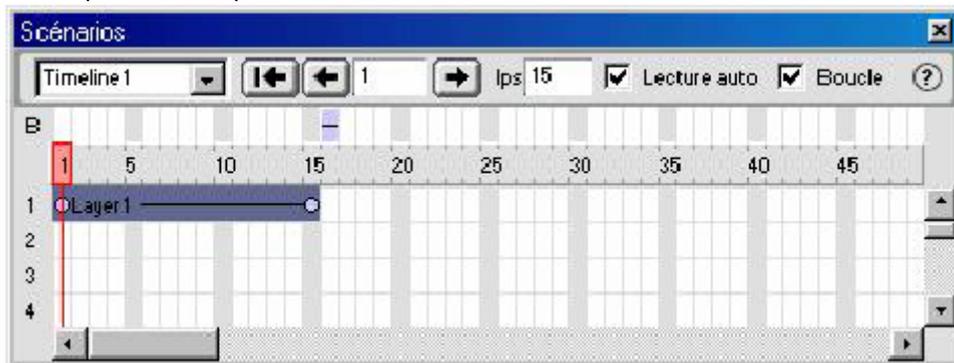
- **Etape 2 : animation du calque** : Cliquez sur l'icône scénarios du lanceur (Fenêtre + Lanceur).



- **Etape 3 :** Sur votre scénario placez -vous sur l'image 1 et cliquez sur le bouton droit de la souris. Dans le menu contextuel choisissez « Ajoutez l'objet ».



- **Etape 4 :** Placez votre indicateur de position (rectangle rouge) à l'image 15 qui correspond à la position d'arrivée de l'animation.



- **Etape 5 :** Déplacez votre calque (une fois sélectionné) à la position désirée. Vous obtiendrez un tracé gris représentant votre animation. A la fin appuyez sur F12 pour visualiser le résultat.



TP4 :

HTML

HTML

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

PARTIE IV : HTML

Exercice 1

Voici le code HTML complet d'une ébauche de document :

```
<HTML>
  <HEAD>
    <TITLE>Ma première page HTML</TITLE>
  </HEAD>

  <BODY>
    <H1> <CENTER> Ceci est une page écrite en langage HTML </CENTER></H1>
    <HR width=75%>
    <H2>Voici l'en-tête du 1er paragraphe</H2>
      Ici est placé le texte du <I>premier </I> <B>paragraphe</B>.
    <P>
    <H2>Voici l'en-tête du <second paragraphe</H2>
      Et là , son texte ...
  </BODY>
</HTML>
```

Exercice 2

Ecrire le code suivant dans un fichier appelé Tp1doc-A.htm

```
<!--Commentaire : ceci est le document : Tp1Doc-A.htm -->
<HTML>
  <HEAD>
    <TITLE>Document A</TITLE>
  </HEAD>

  <BODY>
    <H1>Ceci est le document A</H1>

    Vous en saurez plus dans le
    <A HREF="Tp1Doc-B.htm">Document B</A>.

  </BODY>
</HTML>
```

Ecrire le code suivant dans un fichier appelé Tp1doc-B.htm

```
<!--Commentaire : ceci est le document : Tp1Doc-B.htm-->

<HTML>
  <HEAD>
    <TITLE>Document B</TITLE>
```

</HEAD>

<BODY>

<H1>Ceci est le document B</H1>

Je vous propose d'en apprendre un peu plus sur l'Education Nationale dans l'Académie de :

Créteil...

<HR>

Voici la liste des saisons :

Printemps

Eté

Automne

et la liste des mois d'automne

Septembre

Octobre

Novembre

Décembre

Hiver

<HR>

Pour revenir au Document A.

</BODY>

</HTML>

Exercice 3

Ecrire un code HTML permettant de représenter le tableau suivant :

Janvier	Février	Mars
Après d'agréables vacances de nouveau au travail ...	Dans 2 semaines les vacances !	Déjà le conseil de classe
Il neige déjà	Il fait très froid depuis 10 jours	Encore de la pluie...

Exercice 4

Il s'agit de construire le formulaire suivant, destiné à recueillir des renseignements personnels.



Formulaire d'inscription

Veillez écrire :

• votre nom :	<input type="text"/>
• votre prénom :	<input type="text"/>
• votre date de naissance :	<input type="text"/>
• votre mot de passe :	<input type="text"/>

Quel est votre niveau d'étude ?

- Primaire
 Secondaire
 Baccalauréat
 Technicien Supérieur

Cochez la case convenable :

- Externe
 Demi-pensionnaire

Quelle est votre LV1 ?

Cocher toutes vos options :

- Option informatique
 LV2
 LV3
 Sciences expérimentales
 Technique
 Autre, à préciser -->

Valider

Effacer



UVT

TP5 :

JAVASCRIPT

UVT

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Partie V : JAVASCRIPT

Exercice 1 :

1- Ecrivez une fonction **OuvrirFenetre()**, déclenchée par l'événement *onClick* sur un bouton. Dans cette fonction vous utiliserez la fonction **open()** pour ouvrir une nouvelle fenêtre, nommée **fen1**.

2- Créez un nouveau bouton qui active la méthode **close()** adressée à une variable fenêtre permettant de la fermer. Il est préférable avant de tester si elle a été ouverte, avec une condition du genre `if (fen != null)`.

Exercice 2 :

Ecrivez un script qui permet d'afficher l'élément choisi à partir d'une liste.

Exercice 3 :

Ecrivez un script permettant de calculer l'âge à partir d'une date de naissance saisie à partir d'un champ texte.

Exercice 4 :

Ecrivez un script qui permet d'entrer un mot dans le champ. Puis d'appuyez sur le bouton "Lancer la recherche".

Si une page correspond à la demande, elle sera affichée; sinon une page d'information apparaît.

Essayez avec les mots "musique" ou "zique" ou "music".

TP6 :

PHP

UVT

Nom des enseignants

- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

Partie VI : PHP

Exercice 1 : affichage simple

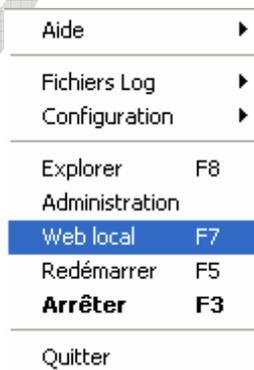
1/ Démarrez Easy PHP.

2/ Tapez ce code dans votre éditeur de texte :

```
<html>
<body>
<?
echo "Bienvenue sur mon site !";
?>
</body>
</html>
```

3/ Enregistrez la page sous c: \ program files\ easy php \ www \ exercices php\ bienvenue.php

4/ Dans la barre d'adresse de votre navigateur, tapez **http://127.0.0.1/bienvenue.php** ou bouton droit sur l'icône de Easy PHP figurant dans la barre des tâches  puis choisissez web local.



Exercice 2 : affichage de variables

1/ Tapez ce code dans votre éditeur de texte

```
<?
$var1="Mohamed";
$var2="Salah";
echo $var1 . $var2;
// Affiche "Mohamed Salah"
?>
```

2/ Enregistrez la page et visualisez le résultat.

Exercice 3 : Les conditions et les boucles

1/ Tapez ce code dans votre éditeur de texte

```
<?
$a=4;
$b=7;
if ($a>$b)
```

```
{  
echo "a est plus grand que b";  
}  
else  
{  
echo "a est plus petit que b";  
}  
?>
```

2/ Enregistrez la page et visualisez le résultat.

3/ Modifiez le code de telle sorte qu'à chaque fois que a est inférieure à b, a sera incrémentée et le test sera revérifié.

Exercice 4 : Les boucles

1/ Soit le code ci-dessus. Réécrivez-le en utilisant la boucle for.

```
<?  
$distance=1;  
while($distance<=5)  
{  
echo 'Vous avez parcouru '.$distance.' km.<br>';  
$distance++;  
}  
?>
```

2/ Mettez 4 valeurs différentes de distances dans un tableau puis affichez-les.

Exercice 5 : Les fonctions

1/ Tapez ce code dans votre éditeur de texte

```
<?
function bonjour ($nb_fois)
{
for($i=1; $i<=nb_fois; $i++)
{
echo "Bonjour !<br>";
}
}
bonjour(5);
?>
```

2/ Enregistrez la page et visualisez le résultat.

Exercice 6 : Les fonctions prédéfinis

1/ Tapez ce code dans votre éditeur de texte

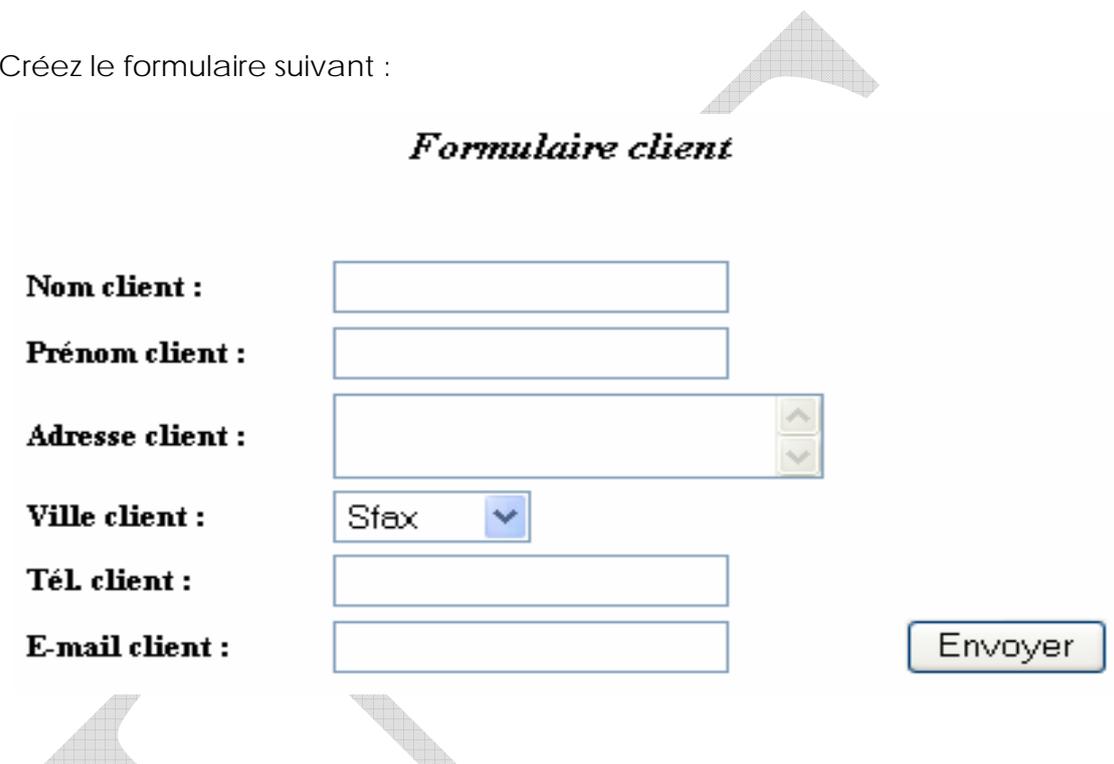
```
<?
$jour=date("d");
$mois=date("m");
$annee=date("Y");
$heure=date("H");
$minute=date("i");
echo 'Bonjour! Nous sommes le '.$jour.'/'.$mois.'/'.$annee.' et il est '.$heure.
' h '.$minute;
?>
```

2/ Enregistrez la page et visualisez le résultat.

3/ Créez une nouvelle page qui affiche la date du jour et qui appelle la page précédente à travers la méthode include(" nomfichier.php ")

Exercice 7 : Les formulaires

1/ Créez le formulaire suivant :



The screenshot shows a web form titled "Formulaire client" with the following fields and a button:

- Nom client :** A text input field.
- Prénom client :** A text input field.
- Adresse client :** A text input field with a vertical scrollbar on the right side.
- Ville client :** A dropdown menu with "Sfax" selected.
- Tél. client :** A text input field.
- E-mail client :** A text input field.
- Envoyer**: A button to submit the form.

2/ En cliquant sur le bouton Envoyer vous préparez le résultat suivant :

Inscription effectuée avec succès !

*Vous êtes M. : xxxxxxxxxxxxxx habitant à l'adresse : xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx à :
xxxxxxx*

Votre téléphone est : xxxxxxxxxxxx et votre E-mail est : xxxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Exercice 8 : Connexion à la base de données

- 1/ Créez la table client avec les champs correspondants (voir exercice 7).
- 2/ Enregistrez tous les clients inscrits dans la bases de données.
- 3/ Créez une page permettant d'afficher tous les clients inscrits.

Exercice 9 : Mise à jour de la base de données

- 1/ Créez la base données commerciale suivante :

Client (codcl, nomcl, prencl, adrcl, telcl, emailcl)

Produit (codpdt, despdt, qtepdt, prixpdt)

Commande (numcde, datecde, #codcl)

Ligne_cde (#numcde, #codpdt, qtécdees)

- 2/ Créez 3 pages pour chacune des tables Client, Produit et (Commande + ligne commande) permettant leurs saisies, leurs mises à jour et leurs suppressions.
- 3/ Affichez un état des commandes relatives à un client donné.

Exercice 10 : Création d'une session

Ecrire deux pages nommées page1.php et page2.php, incluant des liens de navigation de l'une vers l'autre. La page 1 créer une nouvelle session, et la page 2 restaure les variables de session. Vérifiez qu'une variable de session définie dans la page 1 est bien restaurée dans la page 2.

Exercice 11 : Destruction d'une session

Rajoutez dans la page 1 un nouveau lien vers la page 2, incluant une demande de destruction de session. Rajouter dans la page 2 un test vérifiant si la demande de destruction est présente. Dans le cas positif, il faut détruire la session. Vérifiez alors que le premier chargement de la page 2, (lors du clic sur ce lien) restaure bien la variable de session (la session n'est pas encore détruite) alors que lors d'un rechargement de la page 2, (par le bouton "recharger" du navigateur) la variable de la session n'est plus restaurée (la session est détruite).

Mini projet : Développement d'un site web dynamique

La librairie fictive *darilkiteb* a décidé récemment de rejoindre les rangs des grands librairies francophones en ligne, comme amazon, fnac, eyrolles, etc. Ses rayons sont très divers : Informatique, Sciences et techniques, etc.

Les objectifs fondamentaux du futur site *www.darilkiteb.com* sont :

- De présenter aux internautes des informations en 3 langues (**Arabes, Français et Anglais**) sur cette librairie
- De permettre **aux internautes** de rechercher les **ouvrages** par **thèmes, auteur** et **mot-clé**.
- De choisir des livres et de pouvoir les **commander**.

A/ Phase d'analyse

Pour cela, le premier responsable de cette librairie vous charge avant de développer le site *www.darilkiteb.com* de lui :

- 1/ Présenter et de lui expliquer l'architecture Web de ce site
- 2/ Proposer une structure générale pour ce site en précisant les liens qui existe entre la page d'accueil et les autres pages Web

B/ Phase de conception

D'après la phase d'analyse, vous avez dégagé la base de données BD_darilkiteb formée par les tables suivantes :

Auteur(cod_a,noma,prea)

Thème(cod_t, nomt)

Livre(cod_l, titrel,prixl,disponibilite,isbn,...,codt)

Auteurlivre(cod_a,cod_l)

internaute(email_c, nomc, prec, datenaissance, adr, tel,motdepasse)

Commande(num_c,datc,emailc)

Ligne_commande(num_c,cod_l,qcl)

Sachant que la solution choisie pour développer le site *www.darilkiteb.com* est : *MySQL-HTML-PHP-Apache* : Installez le logiciel Easyphp puis expliquer le rôle et la signification de chaque mot *MySQL-HTML-PHP-Apache*

C/ Phase de réalisation

- 1/ Implantez la base de données BD_darilkiteb sous MySQL.
- 2/ Préparez les interfaces suivantes :

- a/ Page d'accueil
- b/ Recherche d'ouvrages selon un /ou plusieurs critères
- c/ Inscription internautes
- d/ Commander des livres

En précisant, pour chaque interface, les objets multimédias utilisés, leurs extensions et les liens hypertextes.

3/ Développez le traitement qui permet l'inscription à distance d'un internaute.

Ce traitement permet :

- ✓ De tester la connexion au serveur MySQL.
- ✓ De tester la connexion avec la base de données.
- ✓ De tester la validé des champs saisis : nom, prénom et adresse non nulles et date de naissance et adresse électronique valides.
- ✓ De vérifier que la taille du mot de passe doit être comprise entre 5 et 10.
- ✓ De tester l'existence de cet internaute dans la base de données.
- ✓ D'insérer un enregistrement dans la table internaute.

4/ Prévoir d'autres traitements.

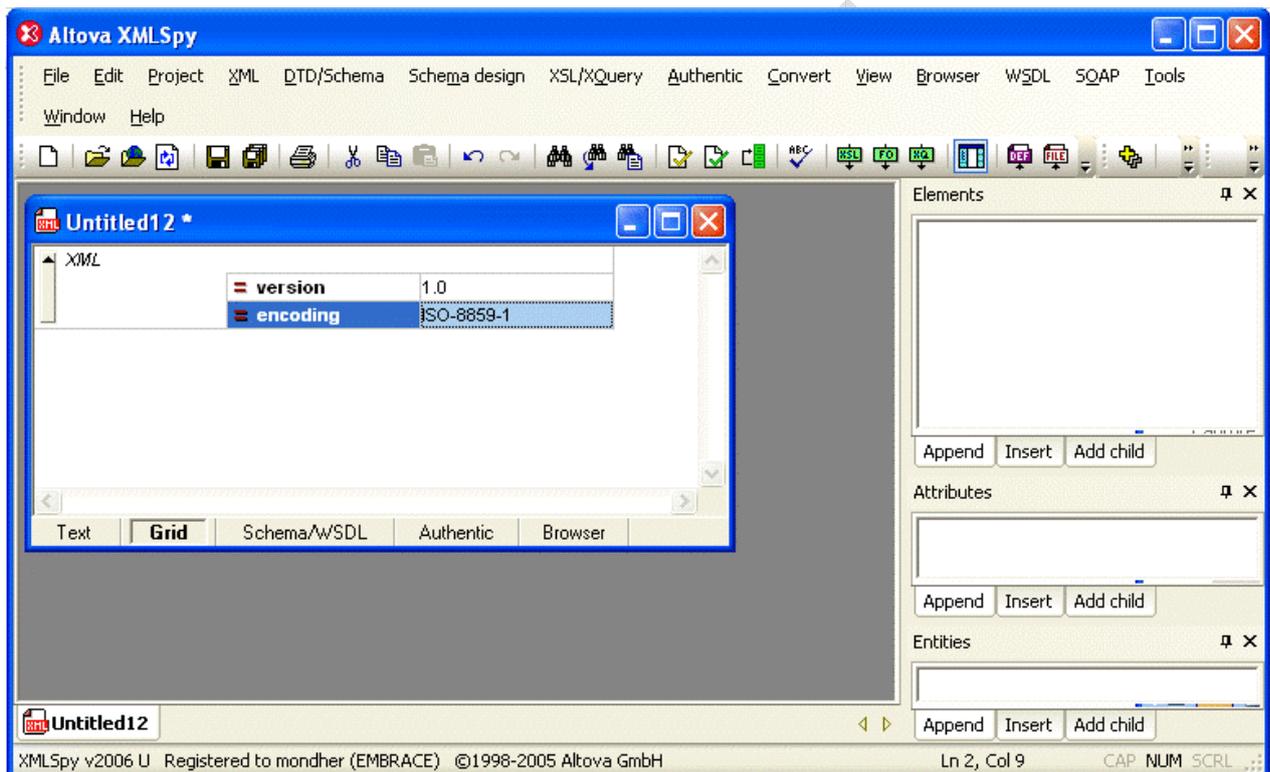
TP7 :
XML



Nom des enseignants
- Mme Sandra Hammami Tlili
- M. Lotfi Tlili
- M. Mondher Hadiji

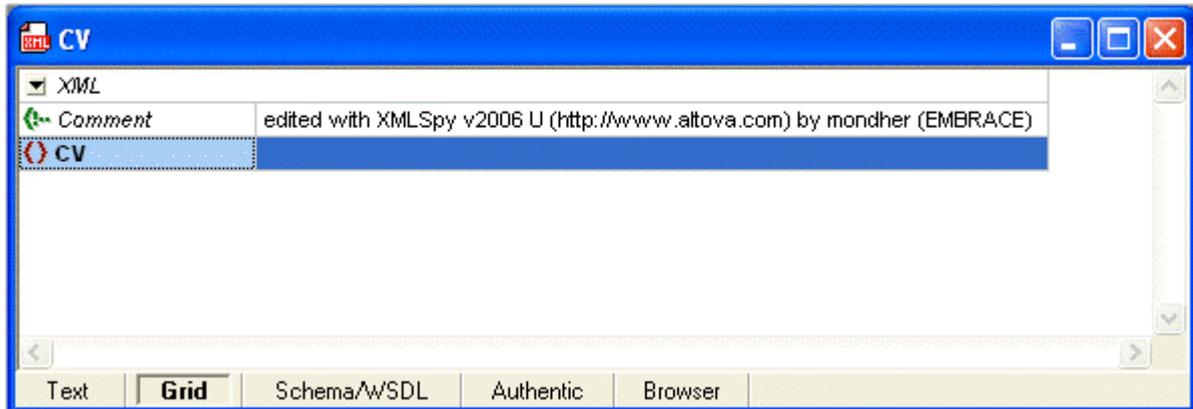
PARTIE VII : XML

Altova XMLSpy est un exemple d'éditeur XML. Voici son interface

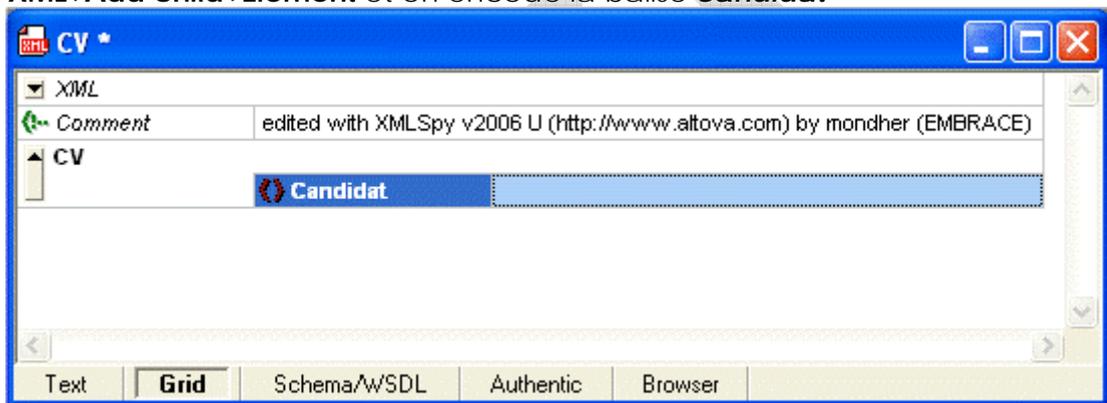


Essayons de présenter un curriculum vitae en utilisant ce logiciel :

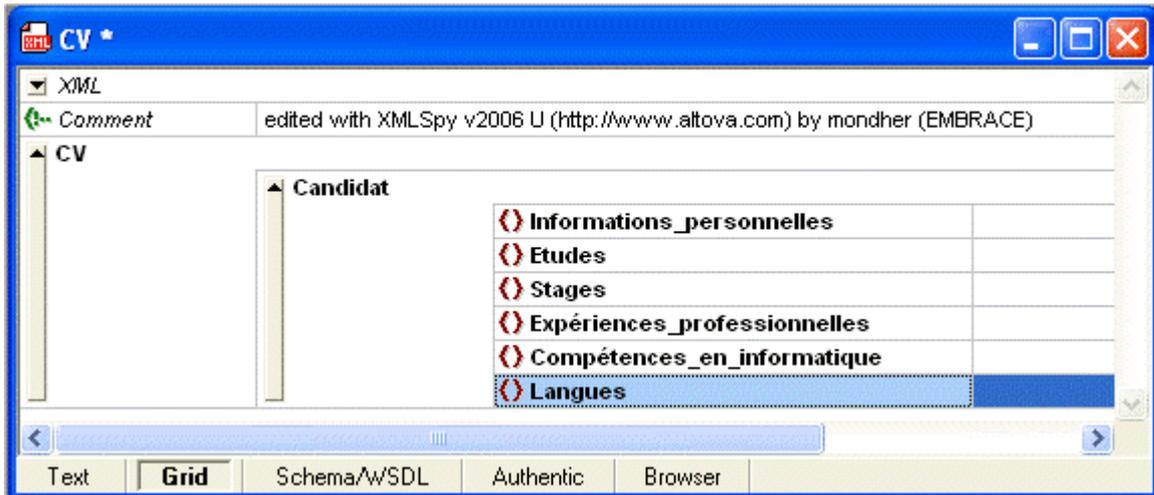
- 1) Commençons par l'élément racine. On prend le menu **XML+Insert+Element** et on encode la balise racine. Puis, on sauvegarde le fichier sous le nom de CV.xml



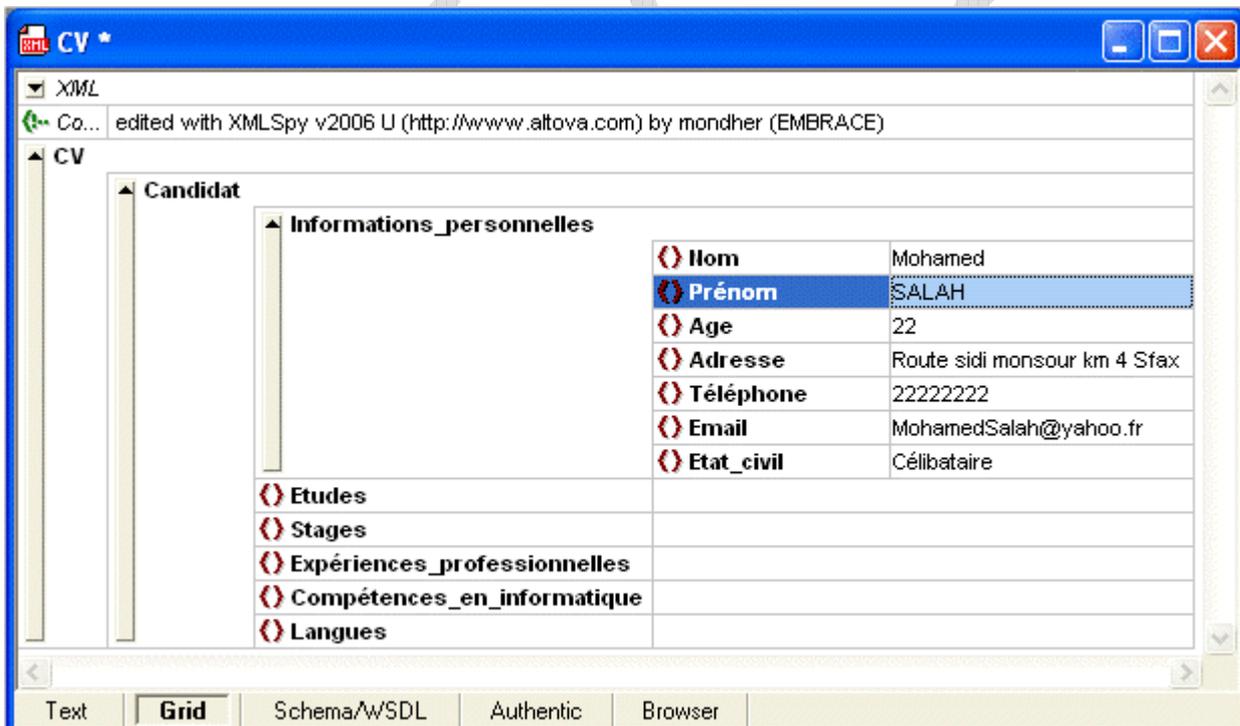
- 2) Encodons l'élément *candidat*. Après avoir cliqué dans la zone CV menu **XML+Add child+Element** et on encode la balise *candidat*



- 3) On passe aux sous-éléments. Après avoir cliqué dans la zone *candidat*, on demande 4 éléments : soit menu **XML+Add child+Element** que l'on répète 4 fois et on remplit les balises

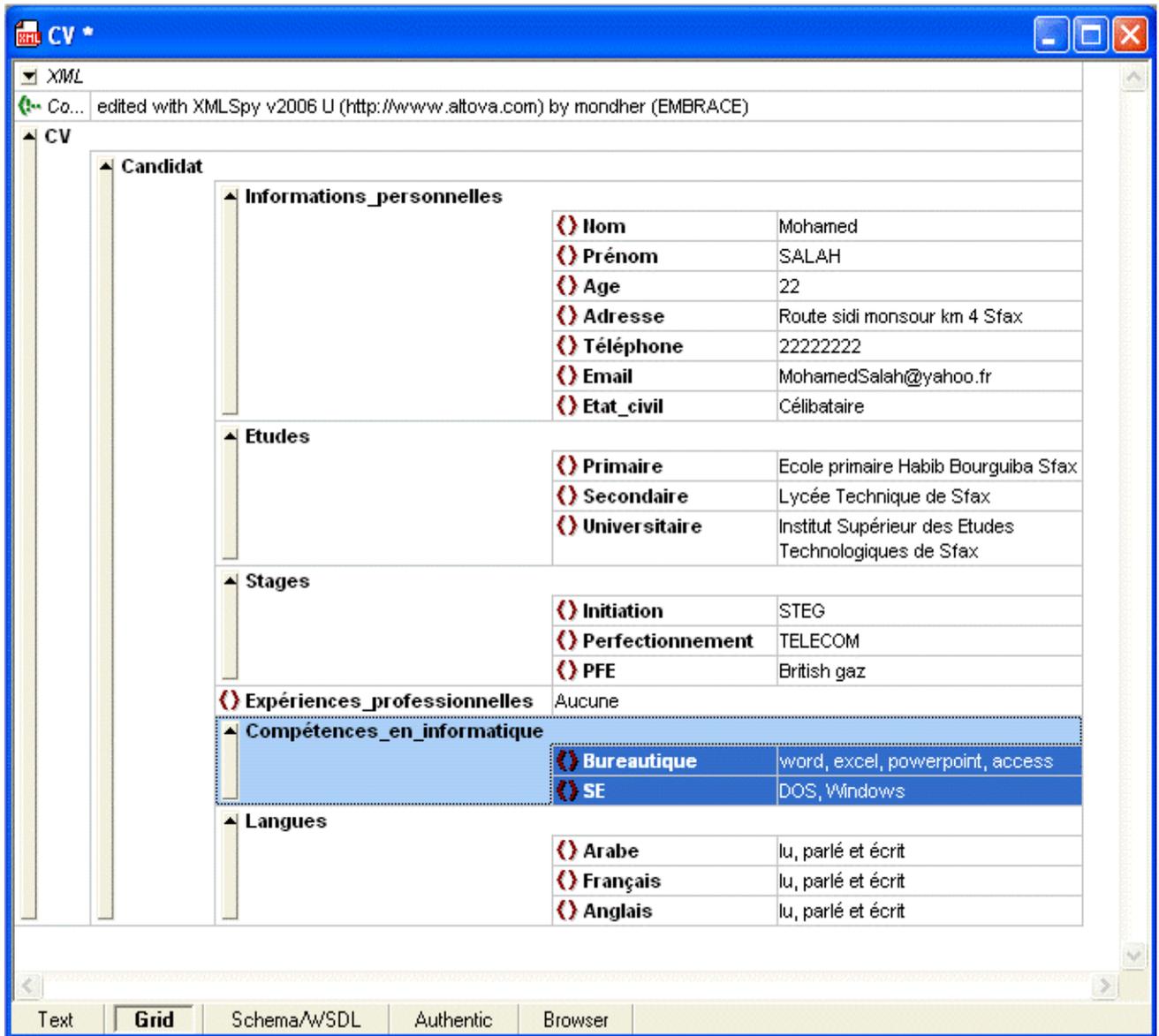


- 4) Après avoir cliqué dans la zone *Informations_personnelles*, on demande 4 éléments soit **menu XML+Add child +Element** que l'on répète 4 fois et on remplit les balises et les données correspondantes



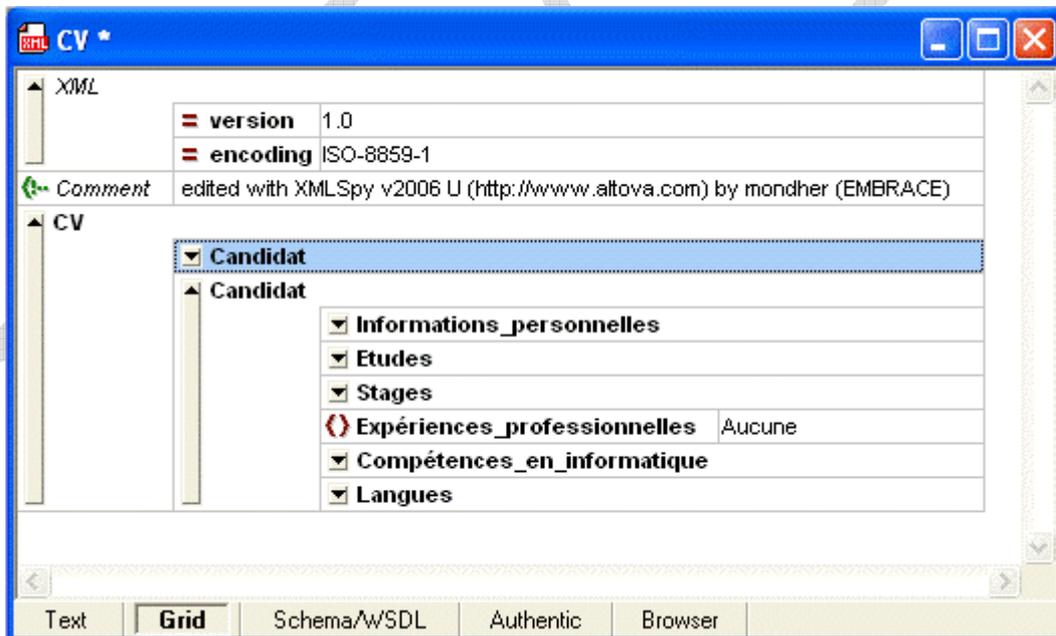
- 5) On répète les mêmes étapes pour les zones :
- Etudes*
 - Stages*
 - Expériences_professionnelles*

- d. *Compétences_en_informatique*
- e. *Langues*

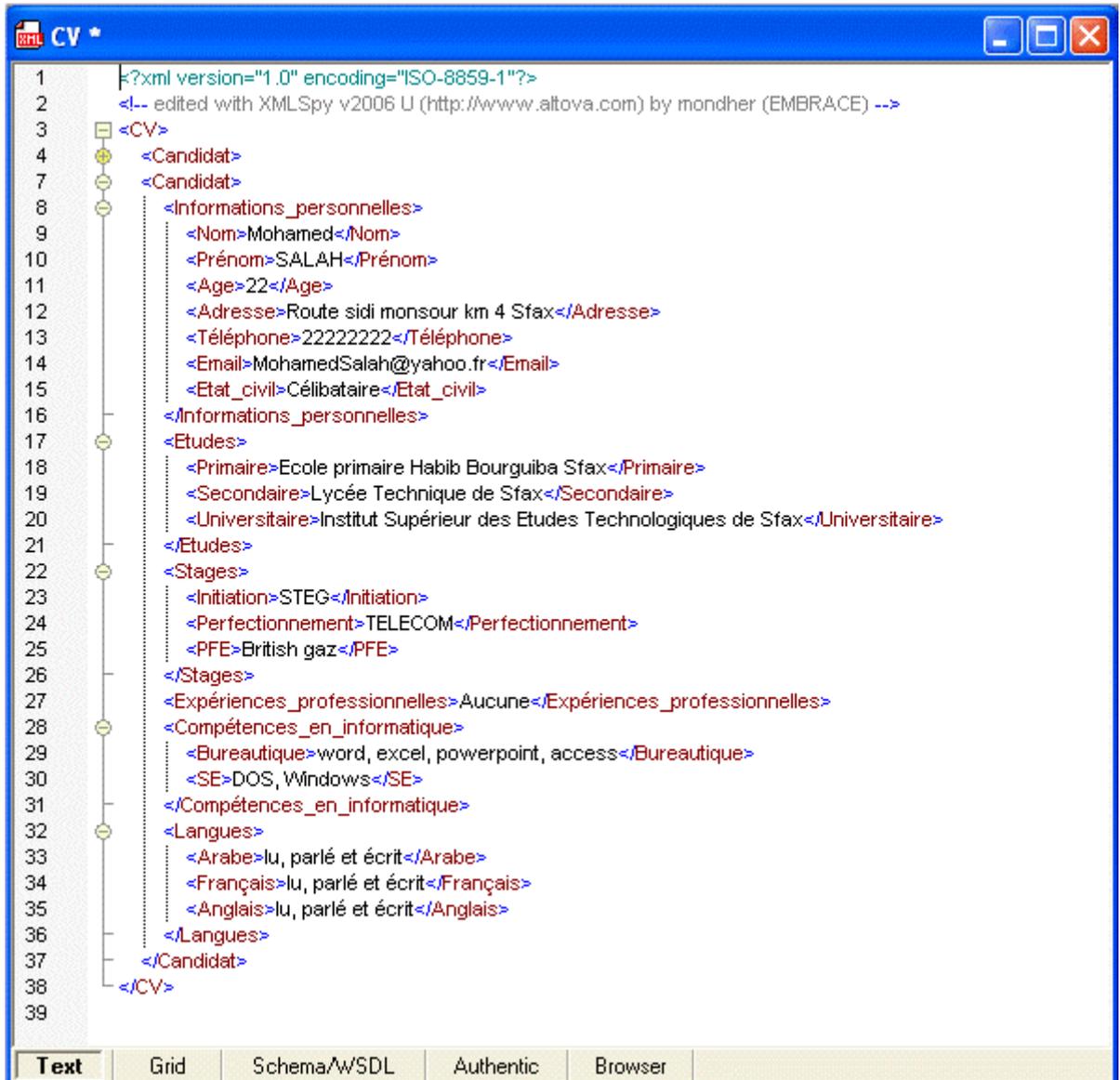




7) On refait la même chose pour les autres données *candidat* :



8) Si on clique sur l'onglet **Text**, on trouve son script XML correspondant:

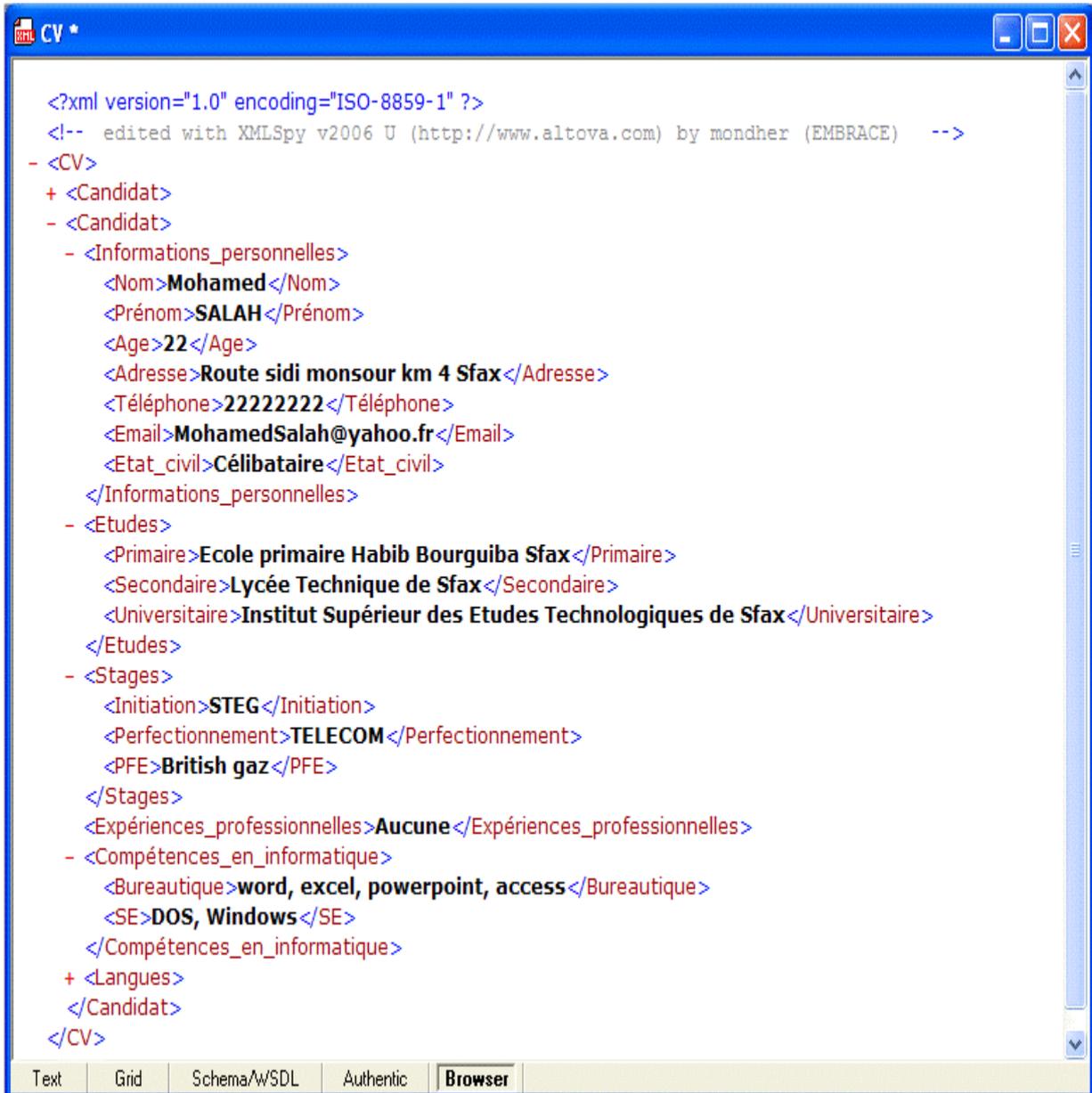


```

1  <?xml version="1.0" encoding="ISO-8859-1"?>
2  <!-- edited with XMLSpy v2006 U (http://www.altova.com) by mondher (EMBRACE) -->
3  <CV>
4  <Candidat>
7  <Candidat>
8  <Informations_personnelles>
9  <Nom>Mohamed</Nom>
10 <Prénom>SALAH</Prénom>
11 <Age>22</Age>
12 <Adresse>Route sidi monsour km 4 Sfax</Adresse>
13 <Téléphone>22222222</Téléphone>
14 <Email>MohamedSalah@yahoo.fr</Email>
15 <Etat_civil>Célibataire</Etat_civil>
16 </Informations_personnelles>
17 <Etudes>
18 <Primaire>Ecole primaire Habib Bourguiba Sfax</Primaire>
19 <Secondaire>Lycée Technique de Sfax</Secondaire>
20 <Universitaire>Institut Supérieur des Etudes Technologiques de Sfax</Universitaire>
21 </Etudes>
22 <Stages>
23 <Initiation>STEG</Initiation>
24 <Perfectionnement>TELECOM</Perfectionnement>
25 <PFE>British gaz</PFE>
26 </Stages>
27 <Expériences_professionnelles>Aucune</Expériences_professionnelles>
28 <Compétences_en_informatique>
29 <Bureautique>word, excel, powerpoint, access</Bureautique>
30 <SE>DOS, Windows</SE>
31 </Compétences_en_informatique>
32 <Langues>
33 <Arabe>lu, parlé et écrit</Arabe>
34 <Français>lu, parlé et écrit</Français>
35 <Anglais>lu, parlé et écrit</Anglais>
36 </Langues>
37 </Candidat>
38 </CV>
39

```

9) Si on clique sur l'onglet **Browser**, on trouve



```

<?xml version="1.0" encoding="ISO-8859-1" ?>
<!-- edited with XMLSpy v2006 U (http://www.altova.com) by mondher (EMBRACE) -->
- <CV>
+ <Candidat>
- <Candidat>
  - <Informations_personnelles>
    <Nom>Mohamed</Nom>
    <Prénom>SALAH</Prénom>
    <Age>22</Age>
    <Adresse>Route sidi monsour km 4 Sfax</Adresse>
    <Téléphone>22222222</Téléphone>
    <Email>MohamedSalah@yahoo.fr</Email>
    <Etat_civil>Célibataire</Etat_civil>
  </Informations_personnelles>
  - <Etudes>
    <Primaire>Ecole primaire Habib Bourguiba Sfax</Primaire>
    <Secondaire>Lycée Technique de Sfax</Secondaire>
    <Universitaire>Institut Supérieur des Etudes Technologiques de Sfax</Universitaire>
  </Etudes>
  - <Stages>
    <Initiation>STEG</Initiation>
    <Perfectionnement>TELECOM</Perfectionnement>
    <PFE>British gaz</PFE>
  </Stages>
  <Expériences_professionnelles>Aucune</Expériences_professionnelles>
  - <Compétences_en_informatique>
    <Bureautique>word, excel, powerpoint, access</Bureautique>
    <SE>DOS, Windows</SE>
  </Compétences_en_informatique>
  + <Langues>
</Candidat>
</CV>
    
```

10) Maintenant, vous êtes habitués à l'environnement de Altova XMLSpy, essayez d'améliorer ce fichier :

- a. En détaillant les balises : par exemples Stages, Expériences_professionnelles, ...
- b. En ajoutant d'autres candidats