

HTML L'essentiel

(Développer un site web en HTML-CSS)



Table des matières

I. Généralités	4
A. Les fondamentaux	5
B. Fonctionnement d'internet	9
C. L'environnement de votre site	19
D. Réflexion et organisation de son projet	25
II. Le langage HTML	30
A. Format de fichier	31
B. Terminologie du langage	34
C. Grammaire du langage	38
D. L'en-tête du document HTML	40
E. Le corps du document HTML	44
III. Le langage CSS	62
A. Format de fichier	62
B. Lien avec le HTML	62
C. Terminologie du langage	63
D. Grammaire des sélecteurs	63
E. Sémantique de la déclaration	63
IV. Et après : JS, HTML5 et CSS3	64
A. Notions de JavaScript	65
B. Les changements du HTML5	66
C. Les nouveautés de CSS3	66
D. Les nouveautés JavaScript	66
V. Un composeur html : Kompozer	68
VI Appoyer	60



Introduction

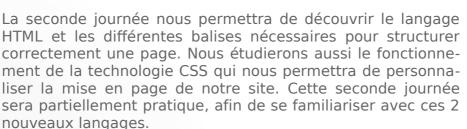
L'objectif de cette formation est de vous faire découvrir le fonctionnement d'un site internet, de la feuille blanche jusqu'à la mise en ligne de vos premières pages.

On va découvrir pour cela plusieurs concept techniques, mais aussi marketing ou commerciaux. L'idée n'étant pas de vous transformer en un expert du développement de site web, mais plutôt des connaisseurs capable de bâtir un petit site, et de comprendre les enjeux des plus grands projets.

Cette formation se concentre principalement sur l'apprentissage de 2 technologies maintenant indissociables : le HTML et le CSS. Ils forment le tandem qui gère le fonctionnement de 100% des sites. Sa connaissance vous permettra de comprendre la structure et la conception des sites internet que vous fréquentez au quotidien.

Entièrement basée sur des logiciels libres, cette formation vous permettra de découvrir des logiciels que vous serez libre d'utiliser chez vous.

En première journée, nous allons découvrir le fonctionnement d'internet, du réseau et des serveurs, ainsi que les différents concepts nécessaire pour appréhender un projet de site internet. A l'issu de cette journée, vous saurez tout ce qu'il faut savoir pour organiser son projet de site internet.



Enfin, la troisième journée sera exclusivement composée de pratique, avec la mise en place d'un Travail Pratique qui se déroulera tout au long de la journée. Il nous permettra de découvrir le fonctionnement du logiciel Kompozer, afin de réaliser notre premier site internet.





Informations sur le document :

Titre	HTML L'essentiel
Fichier	support-html-1.0.odt
Crée le	25. novembre 2012 à 09:33
Modifié le	18 décembre 2012 à 18:00
Version	1.0.70
Nbre de pages	69
Auteur	Christophe LARUE



I. GÉNÉRALITÉS

Les sites Web sont omniprésents, voire incontournables, dans notre quotidien. Ils peuvent prendre différentes formes (blog, commercial, sociaux,communication, plateforme collaborative, ...).

ils sont maintenant une composante essentielle de la communication d'une entreprise et de plus en plus souvent un canal commercial extraordinaire.

Ce chapitre vous fera comprendre ce qu'est un site Web et toutes les techniques ou technologies nécessaires pour élaborer une stratégie et un positionnement sur internet.



Généralités	4
A. Les fondamentaux	4
B. Fonctionnement d'internet	4
C L'environnement de votre site	5



A. Les fondamentaux

1. Qu'est-ce qu'un site internet ?

Un site web est un ensemble de pages électroniques, chaque page présentant une information sous forme de texte, d'images ou d'autres éléments multimédia.

Ces fichiers sont mis à disposition 24h sur 24h sur des serveurs qui en distribuent des copie aux visiteurs qui en font la demande. Ces derniers utilisent un navigateur pour consulter ces pages électroniques.

Les pages d'un site sont organisées selon une structure généralement arborescente (dossier, sous-dossiers, ...). la page principale étant appelée page d'accueil (home-page en anglais)

La navigation entre les différentes pages repose sur l'emploi d'hyperliens, permettant de rendre actif les éléments d'une page, et amener l'utilisateur vers d'autre pages, d'autres sites ou d'autres fichiers.



Le **serveur web** envoi des **pages HTML** vers le **navigateur** qui s'occupe de comprendre le code HTML et afficher le résultat. Les **hyperliens** permettent de rendre cette consultation **interactive**.



2. Les différents types de sites internet

Les sites internet sont souvent caractérisés par leur objectifs et la richesse des fonctionnalités qu'ils offrent à leurs internautes.

Tous les sites utilisent du HTML et du CSS pour fonctionner, mais ils vont :

- présenter des informations différentes
- offrir des services plus ou moins avancés
- utiliser des outils plus ou moins évolués

Bien que chaque projet internet soit différent, voici quelques « type de site » correspondant à des besoins commun à la plupart des entreprises ou organisations.

Site vitrine	Site de présentation d'une entreprise. Simple et peu étoffé, il assure une présence sur internet
Forum	Site participatif permettant de communiquer avec une communautés sur un sujet précis (marque, produits, thèmes, opinions)
e-commerce	Site permettant de consulter le catalogue produit d'une entreprise et offrir aux clients un moyen de consommation
Collaboratif	Site permettant le partage de connaissance autour de sujets spéci- fique. Permet un partage de document de travail.
Réseaux sociaux	Site web permettant de communiquer avec les clients ou les pros- pects. Utilisation d'outils sociaux pour créer ou cultiver une marque ou un produit
Site mobile	Version optimisée pour les mobiles du site classique. Permet d'optimiser la mise en page pour des écrans de plus petite taille.
Blog	Site permettant de diffuser des informations régulières et plus ou moins informelles permettant de communiquer auprès d'une com- munautés

Les sites web sont également souvent pensés différemment en fonction du type d'auditoire. On ne pensera pas de la même un site pour le grand public et un site pour le personnel. Les informations diffusées auprès de nos partenaires sont rigoureusement différentes des informations disponibles pour le grand public.

On fait ainsi la distinction entre les sites :

a. internet

Public: Grand public, information disponible pour tout le monde



Accessibilité: Permanente

Objectif: Fournir des informations et des fonctionnalités utiles pour l'ensembles des internautes. C'est un site qui permet de communiquer auprès de n'importe qu'elle type de visiteur : étudiant, prospects, administration, concurrents

b. intranet

Public : Employés, personnel de l'entreprise

Accessibilité: Heures de travail, restriction aux locaux de l'entreprise

Objectif: Fournir des informations permettant de réaliser son travail quotidien de la manière la plus optimales. Permet de diffuser rapidement des documents internes, facilite la communication, l'échange et le travail autour des projets de l'entreprise. Manipule des informations plus ou moins sensible, la sécurité est donc éssentielle.

c. Extranet

Public: Partenaires, personnel extérieur **Accessibilité**: Permanente mais restrictive

Objectif: Fournir des informations et des fonctionnalités auprès des différents partenaires de l'entreprise (fournisseurs, partenaires, co-traitants). Permet partager des documents autour des métiers de l'entreprise. Manipule des informations plutôt sensible, la sécurité est donc essentielle sur ce type de site.

H

Les sites internet, intranet et extranet sont dorénavant des composants essentiel du **système d'information de l'entreprise**.

Il est important d'avoir une bonne connaissance de ce dernier avant de mettre en place, ou de modifier, un site.





3. Le contenu d'un site internet

Le HTML est un langage de balisage souple et léger, particulièrement adapté aux contraintes techniques du web.

a. Le texte

Internet est encore pour un petit moment un média majoritairement écrit. Le rédactionnel est une composante majeur des sites web de qualité.

La rédaction doit être claire et concise. La lecture écran est moins confortable que le papier, en tenir comte et faire usage de l'hyper-navigation pour éclaircir le texte.

b. Les images

Les images utilisées sont gif, jpg ou png (remplaçant du gif).

c. Les vidéos & les animations multimédia (Flash)

Les videos sont arrivés tardivement sur internet, et les sites exploitent encore peut l'ensemble de ses posibilités.

Flash est un format d'animation vectoriel, donc léger; Sa lecture nécessite un plugin spécifique et son utilisation tend à disparaitre au profit du HTML5 et du CSS3 plus ouverts.

d. Exemples de HTML

Le site du w3schools vous présente plusieurs exemple de code HTML. Dans chaque partie, et pour chaque exemples, une partie « essayez vous-même » vous permettra de modifier et tester le code HTML directement sur le site, et le récupérer par copier-coller.

Pour essayer cela, rendez-vous sur http://w3schools.com/html

Pour observer les pièges à éviter lors de la construction de vos pages, vous pouvez consulter le site worstoftheweb.



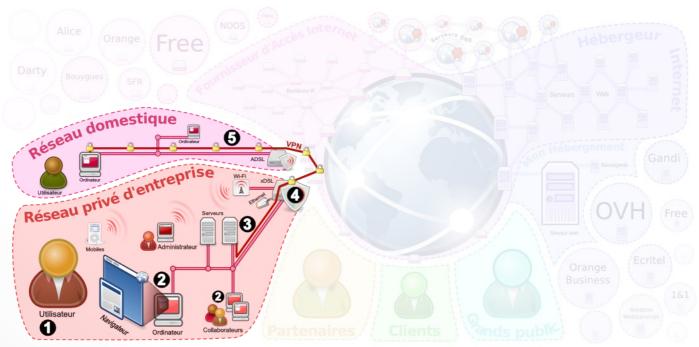
Pour trouver et évaluer des exemples de site bien construit, ou avec un esthétisme particulier, vous pouvez consulter le site worstoftheweb.



B. Fonctionnement d'internet

1. Réseau & FAI

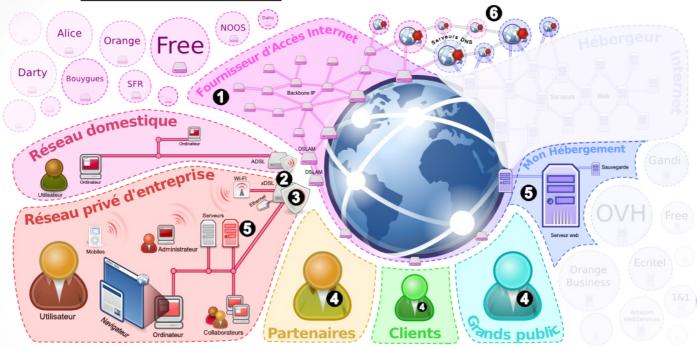
a. Réseau interne et externe



- Au départ de tout, il y a l'**utilisateur** qui utilise un ordinateur pour effectuer ses tâches (document, mail, dessin, communication, jeux,...)
- 2. Cet **ordinateur** est connecté à un réseau local pour être en relation avec les autres ordinateurs de l'entreprise ou du domicile.
- Un serveur local orchestre le fonctionnement du réseau. On parle alors de «services réseaux ».comme par exemple:
 - l'attribution d'adresse IP (DHCP)
 - la surveillance réseau (monitoring) et Restrictions d'accès
 - le partage de fichiers (NFS, Samba)
- 4. Le réseau local n'est pas accessible de l'extérieur et chaque entreprise sécurise son réseau local, protégé d'internet grâce à un pare-feu
- L'accès VPN permet d'accéder, de manière sécurisée, au réseau local lorsqu'on est physiquement distant. Une fois connecté, l'utilisateur est considéré physiquement présent dans les locaux.



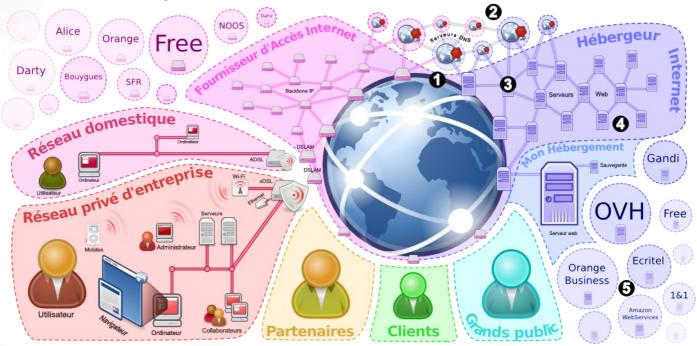
b. Le fournisseur d'accès



- Pour avoir accès au réseau internet, une entreprise (ou un particulier) doit souscrire un abonnement auprès d'un fournisseur d'accès Internet (FAI) tel que Orange, free, Alice, Darty,...
- 2. Après installation du boitier ADSL (ou autre technologie selon), le serveur réseau local est capable de dialoguer avec le réseau internet.
- Le pare-feu de chaque entreprise prend en charge l'aiguillage et le contrôle de toute information entrante ou sortante. Il est en général configuré pour interdire l'accès depuis l'extérieur.
- 4. Toutes les entreprises connectés à internet peuvent alors échanger des informations via ce réseau global.
- **5.** Le responsable informatique peut maintenant choisir ou installer et héberger **les services** informatiques de son entreprise :
 - Sur internet: ce qui doit être constamment disponible et partagé.
 - Sur le réseau local: ce qui est critique et/ou ce qui n'est accessible qu'à une population restreinte.
- Le principal service apporté par notre FAI, est l'accès à ses serveurs DNS. DNS est un système d'annuaire mondial permettant de traduire un nom de domaine en une adresse IP.



c. L'hébergeur web et les fournisseur de services

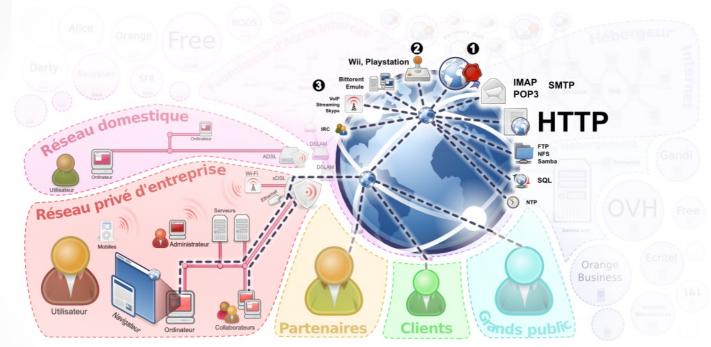


De l'autre côté, nous avons les hébergeurs. Leur rôle consiste à :

- Établir et superviser les inter-connexions avec les autres réseaux
- 2. Maintenir et rendre disponible leurs serveurs DNS
- 3. Gérer les flux réseau
- 4. Fournir des machines fiables et disponibles pour stocker les données
- 5. Proposer de plus en plus de **services à haute valeur ajoutée** via notamment le Cloud-Computing



d. Le réseau internet



L'accès à internet permet d'accéder à de multiples services.

Pour Communiquer sur internet, notre ordinateur a besoin de 3 éléments.

- Savoir à qui s'adresser (adresse IP, DNS)
- 2. Savoir dans quelle langue dialoguer (Protocole de communication)
- 3. Savoir de quoi parle t-on (mail, document, fichiers, données,...)

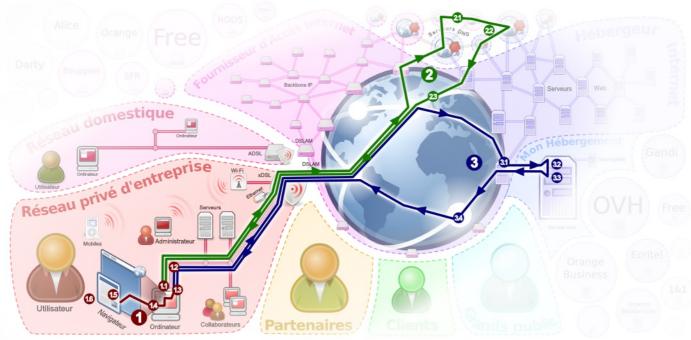
Les logiciels installés sur le poste client (client mail, navigateur web, explorateur de fichier, ...) sont là pour comprendre ces échanges.

e. Les protocoles de communication

Jeux	Wii, Playstation, GooglePlay
Données	SQL, JSON, SOAP
Média	VoIP, Streaming video
Peer to peer	E-Mule, Bittorent, Me.ga
Fichiers	File Transfert Protocol, NFS, Samba
Messagerie	IMAP, POP3, Simple Message Transfert Protocol
Temps	Network Time Protocol
Pages internet	HypeText Transfert Protocol, Proxy
Nom de domaine	Domain Name Server



2. Le cas de la consultation d'une page web



- Pour afficher un site internet, le navigateur internet s'occupe de:
 - 1. Obtenir l'adresse du serveur (DNS)
 - 2. Demander la page internet au serveur web du site (protocole http)
 - 3. Récupérer toute la réponse (protocole http)
 - 4. Lire et comprendre la réponse (language html)
 - 5. Afficher la page à l'écran
 - 6. Attendre une nouvelle demande (clic, lien, envoi de formulaire, ...)
- **2.** Le serveur DNS prend en charge :
 - La recherche d'adresse IP
 - 2. La conservation d'un cache
 - 3. La livraison de la réponse le plus rapidement possible
- 3. Le serveur Web, quand à lui, s'occupe de:
 - 1. Attendre les demandes des utilisateurs
 - 2. Comprendre la demande (en language http)
 - 3. Trouver ou générer un fichier HTML
 - 4. Renvoyer la réponse (le contenu du fichier)



a. La résolution de nom et le serveur DNS

En tout premier lieu, le navigateur envoie une demande de résolution de nom auprès du serveur DNS de notre FAI. Le serveur va alors :

- 1. Écouter les demandes de résolution
- 2. Interroger ses voisins pour trouver l'adresse IP correspondante
- 3. Mémoriser la réponse pour la prochaine demande
- 4. Renvoyer la réponse au navigateur

b. La communication HTTP

Une fois obtenue l'adresse IP correspondant au serveur web du site, le navigateur peut lui adresser sa demande (pour obtenir la page internet).

Il va formaliser cette demande en utilisant le protocole HTTP. Ce dernier va lui permettre :

- 1. De préciser le document recherché
- 2. De transporter les autorisations nécessaires si besoin
- 3. De fournir une trace de la visite (navigateur, cookie, ip source)
- 4. De transporter les documents (html, jpeg, pdf, formulaire)

Exemple de requête HTTP vers un serveur web :



GET /index.html HTTP/1.1

Host: www.plb.fr

User-Agent: Mozilla/5.0 (X11; Linux x86_64) Gecko/201001 Firefox/16.0

Accept: text/html,application/xml;q=0.9,*/*;q=0.8 **Accept-Language**: fr,fr-fr;q=0.8,en-us;q=0.5,en;q=0.3

Accept-Encoding: gzip, deflate

Referer: http://www.google.fr/url?sa=tt

Cookie: utmcsr=google Cache-Control: max-age=0

Réponse du serveur :



HTTP/1.1 200 **OK**

Date: Wed, 12 Dec 2012 22:43:47 GMT

Server: Apache/2.2.17 (Ubuntu)

Content-Encoding: gzip **Content-Length**: 5106

Keep-Alive: timeout=15, max=100 **Content-Type**: text/html; charset=UTF-8





c. L'Unified Ressource Locator (URL)

Vous avez peut être déjà entendu parler du terme URL. Si ça n'est pas le cas, vous l'avez déjà vu sans vous en rendre compte.

L'URL est une adresse unique permettant d'accéder et de consulter n'importe quel document sur internet. Voici une exemple d'URL complète :



http://www.plb.fr/blog/2012/article.html#chapitre1?lang=fr

Si l'on décompose cette adresse, on obtient :

http://	Le protocole d'échange, en l'occurrence HTTP
www.	Le nom du sous-domaine. En général, www pour le site internet
plb.fr	Le nom de domaine
Blog/2012 /	Nom des dossiers et sous dossiers
article.html	Nom du fichier recherché (html ou pdf, doc, jpg, avi, zip,)
#chapitre1	Un eplacement dans une page HTML (ancre interne)
?	Indication de paramètre
lang=	Paramètre nommé lang
fr	Valeur du paramètre lang





3. Le serveur web



a. Nom de domaine et accès au serveur

Accepter les requêtes

b. L'enregistrement des logs

Mémoriser les informations de visite

c. La gestion des droits d'accès

Analyser la demande, Vérifier les autorisations d'accès

d. Lecture des documents

Trouver ou générer le document demandé

e. Génération des documents

Trouver ou générer le document demandé

f. La livraison du contenu

Lire le contenu de ce fichier

Renvoyer le résultat lu, au fur et à mesure

Attendre la prochaine demande



4. Le navigateur



a. Le document source

- Une fois le document HTML retourné par le serveur, le navigateur peut commencer sa lecture. Un premier passage lui permet de repérer les fichiers complémentaires (images, animations, vidéo, son, styles CSS, Javascript, etc...). Le navigateur lance la demande de ces fichiers auprès du serveur.
- 2. Tous les fichiers sont réceptionnés et mis en cache par le navigateur, ce qui lui évitera d'avoir à les télécharger une seconde fois.

b. L'interprétation et le DOM

- 1. Le document est analysé pour vérifier qu'il est « bien formé ». Toutes les erreurs de balise et de non respect de la syntaxe XML seront corrigées.
- 2. Le document est analysé pour vérifier le bon respect du langage HTML. Le doc-type HTML permet de spécifier la version HTML de notre document. Le navigateur s'en sert pour connaître la version de notre HTML, et les règles qui s'appliquent. En cas d'erreur, il tente de corriger nos erreurs.
- 3. Le document est chargé dans l'espace mémoire du navigateur sous forme d'arborescence, c'est le Document Obect Model (DOM).
- 4. Un interpréteur CSS est associé au DOM. Toutes les règles CSS y sont chargées et associées avec les éléments correspondants.



5. Un interpréteur Javascript est également associé à notre DOM, et l'ensemble des scripts y sont chargés.

c. Le rendu

- 1. Avant d'afficher définitivement le résultat graphique issu du DOM, le navigateur va faire appel, si nécessaire, aux lecteurs externes (flash, pdf, ...).
- 2. Enfin, le résultat graphique apparait à l'écran.

d. L'interactivité

Pendant que l'internaute consulte la page, le navigateur écoute les mouvements de souris ou de clavier et fait réagir le DOM en fonction afin de produire des effets ou lancer des actions, comme par exemple :

- · survol de menu
- images changeantes
- info-bulles
- complément d'informations
- Affichage d'informations dynamiques



C. L'environnement de votre site

1. Le nom de domaine

a. Qu'est-ce que c'est?

Un nom de domaine est composé d'un nom et d'une extension. L'organisme IANA (Internet Assigned Numbers Authority) est en charge d'attribuer et de superviser l'ensemble des noms de domaines.



Exemple:

startx.fr est le nom de domaine de la société STARTX, pour la France

b. Les extensions

L'extension peut être nationale (.fr, .be, .tv,...) ou générique (.com, .org, .info,...)



Pour consulter la liste des extensions existantes, et les organismes en charge de leur supervision, vous pouvez vous rendre sur : http://www.iana.org/domains/root/db/

i) Les domaines génériques

Ils sont confié en gérance à certaines entreprises, la plupart du temps américaines (Verisign). Ces entreprises sont supervisées par les principaux contributeurs de l'IANA. C'est cet organisme qui est chargé de gérer et superviser la gestion de ces extensions de domaine. Dans certains cas, il faut fournir un justificatif. (association non lucrative pour .org)

ii) Les domaines nationaux

Chaque extension nationale est gérée par l'administration du pays concerné. Ils fixent les règles d'attributions et les prix. Les extensions génériques sont géré par des entreprises américaines supervisée par les principaux contributeurs de l'IANA.

c. Les registrars

Avant d'acheter un nom de domaine, vérifier si il est libre sur les bases Whois. Ce mécanisme permet d'interroger les bases DNS, afin d'obtenir le détail du propriétaire d'un nom de domaine.



Un bon registrar vous proposera d'héberger votre nom sur son serveur DNS en assurant une redirection vers une adresse IP ou nom, ou de vous laisser le choix de l'hébergement.

De nos jours, la plupart des registrars sont hébergeurs et les hébergeurs proposent des options d'achat de nom de domaines lors de l'achat d'un serveur web.

<u>d. La propriété</u>

Tant que vous payez et que vous renouvelez le règlement de votre nom de domaine, vous êtes le seul a pouvoir effectuer des opération sur les serveur DNS, et donc assurer la gestion de votre nom de domaine.

Le fonctionnement est identique au droit de propriété anglo-saxons ou la terre appartient au roi, et les sujets ont des concessions de différentes durée.

Vous pouvez acheter votre nom de domaine pour 1an, 2ans, 5ans, 10ans ou 99ans. Vous devez le renouveler à chaque échéances et risquez de le perdre dès que le terme de votre contrat est échu.

e. La relation avec le serveur web

Lorsque vous êtes propriétaire d'un nom de domaine, vous avez accès à l'administration de votre zone DNS. Cela vous permets de créer autant de sous-domaine que vous le souhaitez. A chaque sous-domaine, vous allez associer une adresse IP correspondant au serveur capable de traiter la demande.

En installant les bon logiciels sur les serveurs correspondant aux adresses IP, vous pourrez associer ces sous-domaines à différents services.



Exemple:

imap.startx.fr smtp.startx.fr www.startx.fr intranet.startx.fr pour les services de livraison d'e-mail pour les services d'envoi d'e-mail Service web pour le site internt public un autre service HTTP pour le site interne

2. L'hébergement

a. Hébergement mutualisé

L'hébergement mutualisé consiste en un serveur sur lequel sont hébergés plusieurs sites. On mutualise les ressources (disque dur, RAM, processeur) mais aussi le prix.





Un prix inférieur Un serveur déjà configuré et standardisé



Risque qu'un « colocataire » surconsomme les ressources du serveur On ne peu pas personnaliser la configuration.

b. Hébergement dédié

L'hébergement dédié nous fournit un serveur qui, comme son nom l'indique, nous est dédié. L'ensemble de la machine vous est entièrement dédié, pendant toute la durée du contrat définie avec votre hébergeur.



Disponibilité de 100% des ressources de la machine Posibilité de configurer tous les services et logiciels souhaités



Un prix élevé

Nécessite souvent d'avoir des compétences en administration de serveur

c. Hébergement privé

On parle d'hébergement privé lorsque vous achetez votre propre serveur, et que vous l'installez dans un réseau entièrement dédié à votre entreprise. C'est le cas de votre réseau local (vos locaux), mais aussi lorsque vous achetez un sous-réseau auprès de votre hébergeur.

L'ensemble de la machine vous est entièrement dédié, mais en plus vous pouvez gérer d'une manière beaucoup plus fine les accès et la sécurité de ce serveur.



Disponibilité de 100% des ressources de la machine Gestion avancé de la sécurité des accès aux informations



Un prix élevé Compétences en administration de serveur indispensable

On peut également installer un logiciel de serveur web sur notre propre ordinateur, ce qui nous permet de simuler le futur environnement de notre serveur web. On développe sur notre ordinateur, test en condition réelle, et ou peut ensuite publier le contenu sur le serveur web de production.

Pour créer son serveur web, il suffit d'installer un service web (IIS, Apache).



Attention, lorsque vous hébergez votre serveur web dans votre réseau local (serveur ou PC), vous êtes limité par la qualité de service de votre FAI, qui limite bien souvent le trafic dans ce sens.

d. Qualité de service

Elle se ressent surtout au niveau du support technique. Les réseaux et les serveur sont souvent identiques. Ils sont maintenant majoritairement virtualisé, ce qui rend encore plus flexible l'attribution des ressources. Il est quasi impossible de savoir où sont situés les serveur physiquement!

e. Les services connexes

Le manager et ses fonctionalités

L'accès mobile du manager

f. Ordres de prix

Pour de bonnes raisons (techniques) ou parfois de mauvaises (renomée, publicité etc).

Comparer les données techniques, tester éventuellement le serveur si période d'essai possible.

3. Les CMS

- a. Avantages et inconvénients
- b. Installation
- c. Administration
- d. Evolution



4. La gestion de votre site

<u>a. Mise à jour</u>

On peut utiliser un logiciel tel que Filezilla.

Transfert intégré dans les environnements de développement.

Le cas des CMS

b. Les fichiers de notre site

les règles de nomages des fichiers sur internet

c. Contenu & responsabilités

Elle est entièrement la vôtre : attention, dans le cas de sites participatifs, vous êtes complice des propos tenu par les utilisateurs de votre site. Ceci concerne les problèmes de copyright, mais aussi tout ce qui porte atteinte à la dignité humaine (contenus pédophiles, incitations à la haine raciale, propos diffamatoire, négationniste, révisionnistes, etc... Note : Les blagues belges sont tolérées.)

d. Statistiques et suivi des internautes

Les hébergeurs proposent des statistiques concernant le nombre de visiteurs, les pages les plus visitées, les mots-clefs, Des systèmes plus fins (Analytics, redsheriff, weborama) peuvent être utilisés (nécessitent une installation).

e. Le référencement

Le référencement doit se faire par inscription sur le moteur de recherche (google, altavista, etc) ou l'annuaire (yahoo). Dans le cas d'un annuaire, il faut s'inscrire dans la bonne rubrique.

Un référencement gratuit peut prendre jusqu'à 6 mois



- d'attente.
- Si on paie c'est immédiat.
- Il est possible d'acheter des mots clés (enchères).

f. La sauvegarde et la restauration

- Créer une archive de son site Web.

Dans le cas de web dynamique, il est prudent de récupérer les données régulièrement par FTP (et SQL si BDD).

- Savoir la restaurer.

Il importe de tester la restauration des la phase initiale, afin de s'assurer que tout fonctionne.





D. Réflexion et organisation de son projet

Il faut commencer par définir ce que l'on veut présenter, et la façon dont on veut le faire (graphisme, navigation), en gardant à l'esprit la cible des utilisateurs. Un site artistique n'est pas conçu comme un site administratif.

1. La conception

- . Analyse des besoins
- · S'assurer qu'il existe réellement un besoin
- · Définir précisément ce besoin
- · Définir la finalité du produit (objectif général)
- · Définir le public-cible (âge, catégorie socio-professionnelle,...)
- · Tenter de situer les contraintes
- · S'assurer de la faisabilité
- 2. Etablir le cahier des charges

Il s'agit d'un contrat écrit liant commanditaires et producteurs en vue d'éviter toute

surprise ou malentendu. Le cahier des charges définit principalement :

- · Les objectifs spécifiques du produit
- · Le contenu
- · Les stratégies pédagogiques (si il y a lieu)
- · Les contraintes (délais, budget, matériel)
- · Les modalités de validation (comment évaluer)
- · La nature et les modalités d'intervention du commanditaire
- · Les conditions de maintenance du produit

2. l'arborescence du site



Concevoir le scénario

- · Mise en forme sur papier du dialogue machine-utilisateur (structure)
- · Mise en scène du contenu (mise en page)

L'arborescence doit être simple , évitez l'originalité si vous voulez que l'information soit trouvée.

Spécifier la matière

récolter ou créer l'entièreté du contenu iconographique et rédactionnel, sous différentes formes. En général, ce travail est du ressort du commanditaire.

3. L'ergonomie

4. Les standards en vigueur

Le développement web est standardisé par les normes édictées par le w3c.

Ces normes évoluent continuellement, mais les navigateurs ne suivent pas forcément au même rythme (parfois +, parfois -).

En cas de doute, consulter www.w3c.org;

5. Intégration

Intégration des éléments média selon le scénario défini, dans un logiciel adapté à la

cible.

6. Gérer la validation

Évaluation du produit :

· Éclatée dans le temps



- · A chaque étape, car l'évaluation d'une fonction influe sur la fonction suivante.
- 7. Réguler l'action

Évaluation de la dynamique de production : nécessité d'une fonction qui encadre les

autres pour la cohérence de l'action et l'optimisation des ressources (matérielles et

humaines):

- · Gérer le bon déroulement du projet
- · Organiser et contrôler le travail de l'équipe

.

Suivre toutes les phases du projet (afin d'anticiper et de prévenir tout retard ou dérive par rapport au budget et/ou au planning)

Ce rôle est en général confié au chef de projet.

a. Organisation de notre espace de travail

- i) Les dossiers
- ii) Les logiciels
- iii) Les ressources originales

6. Le cas des images et des vidéos

a. Création des images

Les éléments images peuvent s'obtenir depuis un appareil

photo numérique, un scanner ou par création directe.

Les logiciels nécessaires sont photoshop, irfan view, etc.

Remarque : en général, une suite logicielle est fournie avec le scanneur ou l'appareil numérique.

b. Redimensionnement des images

Les images doivent être en 72 ppp

et au format écran final.

Eventuellement, il faudra aussi RECADRER (= retailler) l'image.



En effet, si vous avez une image très large (cf. bandeau pub 68px x 550px), un redimensionnement proportionnel (pour éviter les déformations), vous obligerait à diviser la largeur par 3 (550 / 172), donc la hauteur devra aussi être divisée par 3. Ce qui donnera une hauteur de 20 px environ.... Donc un tout petit logo!

c. Compression des images

Les images doivent être au format Jpg, png ou gif Gif gère la transparence, Jpeg non.

Par contre jpeg est plus léger.

Donc, si pas besoin de transparence (ce qui est le cas ici), utiliser le format jpg.

(NB. On parle ici du format de SORTIE (pour l'image finale). Il est possible de partir d'une image fournie en GIF et la transformer en Jpg.)

d. TP

Redimensionnement d'image en JPG et détourage en GIF. A reprendre sur le support de cours WNI

e. Le cas des vidéos

principe de récupération depuis un caméscope:

Si c'est un numérique (DV), la récupération est directe. Sinon il faut une carte d'acquisition.

Pour l'édition, on utilise Première ou des outils système (window movie maker).

Le format final doit être wmv, mov (quicktime) ou Flash. Un plugin sera éventuellement nécessaire pour la restitution.

Réaliser des animations Flash : présentation des outils.
 Le plus standard est Flash (Adobe, ex-Macromedia) mais il existe des variantes gratuites.

Le principe d'animation utilise une ligne de temps et plusieurs pistes d'images.



7. La gestion des couleurs

Les couleurs sont exprimées par une combinaison des composantes lumineuses Rouge, Verte et Bleue (RVB).

Pour chacune de ces composantes, on attribue une valeur de luminosité allant de 0

(éteint) à 255 (luminosité maximale).

Exemple:

Noir = 0.0.0

Rouge = 255,0,0

Vert = 0,255,0

Gris foncé = 100,100,100

Gris clair = 200,200,200

En html, ces valeurs doivent être utilisées en hexadécimal.

Exemple:

255 = FF

200 = C8

100 = 64

Vous pouvez utiliser la calculatrice windows, en mode scientifique.



II. LE LANGAGE HTML

Même sans rentrer dans les détails, il est utile de comprendre le principe du HTML si l'on veut maîtriser les logiciels de création web.



١.	Le langage HTML	. 12
	A. Format de fichier	. 12
	B. Terminologie du langage	.12
	C. Grammaire du langage	. 13
	D. Sémantique du langage	13



A. Format de fichier

Pour créer un fichier HTML, un simple fichier texte suffit. Vous créez un fichier texte (format txt uniquement) avec notepad. Vous changez son extension en html et vous avez un fichier HTML. Pas besoin de logiciels couteux ou complexe, un simple éditeur suffit.

En double-cliquant dessus, c'est le navigateur qui s'ouvrira et vous permettra de consulter le résultat de votre page.



Exemple:

catalogue.html contact.php

nom du fichier présentant un catalogue produit nom du fichier présentant vos coordonnées

Le nom de la première page doit toujours être nommée index (.html ou .htm).

Vous pouvez choisir les noms de tout vos autres fichiers comme vous le souhaitez. Il faut cependant respecter les règles de nommages des fichiers publier sur internet.

1. Le Doctype et les DTD

a. Rôle et intérêt d'une DTD

Le "Doctype" défini le **type de document utilisé**. Notre document se termine en .html, il a des balises, mais comme nous l'avons déjà évoqué, il y a plusieurs version de HTML.

C'est le doctype qui permettra de **préciser dans quel version du HTML** nous avons écrit notre page HTML.

La Déclaration de Type de Document (DTD) sert à indiquer à quelles règles d'écriture obéit le code de notre page HTML (ou XHTML).

Comme beaucoup de document utilisent les même règles, elle sont sont enregistrer dans des documents au formalisme assez particulier, et hébergées sur le site du W3C. les DTD sont destinées à être lues avant tout par des logiciels (parseurs, navigateur), et non par des êtres humains.





Vous pouvez voir sur le site du W3C les DTD officielles utilisées par nos navigateurs, comme par exemple :

http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd

b. Comment mettre une DTD dans ma page HTML

Dans le code HTML, notre déclaration doit être la première chose, **avant la balise <html>**. Elle est matérialisée par la syntaxe suivante ; <! DOCTYPE xxx>. C'est une balise particulière, appelée « instruction » qui va permettre au navigateur d'**adapter son comportement**, mais qu'il ne traitera pas comme les autres balises.



Mettez toujours une déclaration de type dans vos vos documents HTML. Cela limite les erreurs de fonctionnement du HTML.

c. XHTML ou HTML

Le HTML est un langage reposant sur le formalisme du SGML, tandis que le XHTML respecte le formalisme du XML (lui même hérité du SGML;)

Le SGML est un langage plus couteux (complexe à traiter) que ne l'est le XML, version light et flexible du SGML, optimisée pour les processeurs DOM ou SAX.

Le choix du XHTML par rapport au HTML impose plus de rigueur. En contrepartie, votre document sera « parsable » à moindre cout (légèrement) et pourra être facilement manipulé par des outils de traitement par lots, ou des programmes basés sur des parseurs (googleBot, CMS, ERP)

Lorsque l'on veut écrire une page compatible XML, donc XHTML, il nous faut respecter les règles suivantes :

- Balises et propriétés en minuscules
- Toute balise ouvrante doit être fermée
- Les balises doivent être correctement imbriquées
- Les valeurs doivent obligatoirement être entre apostrophes ou guillemets
- Chaque propriété doit avoir une valeur (ex : checked="checked")

d. Les types de DTD

HTML L'essentiel

HTML 5	Cette DTD est identique au XHTML 1.0 Strict, mais avec de nouvelles balises pour la gestion du contenu et de sa structure
XHTML 1.0 Strict	Cette DTD est identique au HTML 4,01 Strict, mais avec un balisage qui doit être écrit comme XML bien formé.
HTML 4.01 Strict	Cette DTD contient tous les éléments et attributs HTML qui ne sont pas obsolète (comme la police) ou sur les framesets.
HTML 4.01 Transitional	Cette DTD contient tous les éléments et attributs HTML, Y COM- PRIS les éléments obsolètes (comme la police)

e. Comment choisir son type de document

Pour une meilleure performance, les navigateurs ont implémenté ces règles dans leurs moteurs respectifs. En revanche, la DTD est indispensable aux validateurs (X)HTML et aux parseurs validant.

Un document même valide et dénué de DTD pourrai poser un problème de rendu dans certain navigateurs et le rendu CSS pourra différer selon la présence ou non d'une DTD.

Les 3 DTD les plus communes :

HTML Compatible

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

XHTML

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">

HTML5

<!DOCTYPE html>



B. Terminologie du langage

1. Balise

Le « M » de HTML signifie « Markup », une « balise » en Français. C'est le concept principal de ce langage, qui définit sa structure et son formalisme. Elle permet de délimiter une informations.

L'idée derrière ces balises, est de délimiter les informations qui seront mises en page dans le document. Comme si on avait des parenthèses mais avec des couleurs et des noms différents.



Les balises HTML utilisent les caractères < et > comme délimiteurs. Le symbole < est immédiatement suivi du nom de la balise.



Le **nom de balise** peut être écrit en minuscule ou en majuscule. L'utilisation des **minuscules** est cependant plus répandue et facilite la compatibilité xHTML de votre document

Chaque balise ouverte doit être fermée. Pour fermer une balise on utilise la même syntaxe que pour la balise d'ouverture, au-quelle on ajoute un symbole / après le symbole < .





Une balise peut toujours contenir du texte, ou d'autres balises. Dans certain cas, elle ne peuvent rien contenir. On parle alors de **balise unitaire**, balise vide ou encore balise ouvrante-fermante. Il est recommandé, dans ce cas précis, de rajouter un / juste avant le symbole >. Cela permet d'ouvrir et de fermer la balise sur une seule ligne.

Exemple :

représente une balise unitaire (retour à la ligne)

Lorsque l'on place plusieurs balises à l'intérieur les unes des autres, il faut toujours mettre les balises de fermetures dans l'ordre inverse de l'ouverture. Dans l'exemple précédent, nous avons d'abord fermé la balise italique, avant celle du paragraphe (la balise bleue, avant la noir)

2. Attribut

une balise peut avoir un ou plusieurs paramètres , appelé des « attributs », représenté sous forme attribut="valeur".

Les attributs doivent respecter les règles suivantes :

- Le nom d'attribut est toujours en minuscule
- Ces attributs se placent uniquement dans la balise d'ouverture
- Ils sont séparé de la balise, et entre eux, par un espace
- Le nom d'attribut est toujours immédiatement suivi par le symbole égal
- La valeur s'inscrit toujours entre les 2 guillemets



3. Élément

Un élément est constitué d'une balise ouvrante, de sa balise fermante, et de tout son contenu, texte, autres balises, etc...



Exemple:

<body>

Mon autre texte contenus (texte et balises)

</body>

L'élément body contient la balise et tout Mon
texte son contenu : les 2 paragraphes et leurs





4. Commentaire

Pour mettre des éléments en commentaire, nous devons les encadrer par les symbole <!-- et -->. Tout ce qui est contenu entre ses 2 code ne sera pas interpreter par le navigateur. Il n'en tiendra pas compte pour l'affichage.



Exemple:

<!-- ceci est un commentaire, ce code ne sera pas utilisé -->





C. Grammaire du langage

1. Imbrication de balise et indentation

Les indentations ne sont pas obligatoires, mais rendent le document source plus lisible pour un être humain. C'est utile pour les web-masters et les développeurs, mais inutile pour les parsers et les navigateurs.

Lorsque le navigateur commence l'analyse de notre document HTML, il supprime toutes les tabluations, tous les retours à la ligne et rassemble tout les espaces multiples par un seul espace.

```
Exemple:
Code non indenté :
                            Code indenté:
<html>
                             <html>
<head>
                               <head>
<title>Page non indentée
                                  <title>Page indentée</title>
</title>
                               </head>
</head>
                               <body>
<body>
                                  <h1><img src="l.jpg"/></h1>
<h1><img src="l.jpg"/>
                               </body>
                            </html>
</h1>
</body>
</html>
```

2. Structure de base d'un document HTML

Lorsque l'on souhaite créer une page HTML complète, nous avons besoin de respecter une certaine structure, rigide, qui nous est imposée par le langage.

A 2 niveaux, cette structure nous permettra de faire la différence entre les données périphériques au document (meta-informations HTML) et les données utile à l'affichage des informations présentées.

Cette structure doit respecter:

- Une balise racine <html>
- Deux balises de structure
 - Une balise d'en-tête du document <head>
 - Une balise de corps du document <body>



3. Racine du document < html>

a. La balise <html>

La balise html indique au navigateur qu'il s'agit d'un document HTML. Elle représente la racine d'un document HTML et elle contient tous les autres éléments HTML (à l'exception de la balise <! DOCTYPE>).



En HTML5, l'attribut manifest à été ajouté pour permettre aux développeurs de gérer plus finement le cache des application web.

b. Les attributs de la balise <html>

manifest	url	L'adresse d'un document décrivant le cache des fichiers utilisés par le site ou l'application internet
xmlns	url	Corespond a l'espace de nom XML du langage HTML. Le plus souvent http://www.w3.org/1999/xhtml

c. Le contenu de la balise <html>

Les 2 seuls balises autorisés sont <head> et <body>. Un seul et unique élément de chaque doit être présent. La balise <head> doit précéder la balise <body>.

Elles vont nous permettre d'isoler :

- le contexte de la page dans la balise <head>
- le contenu affichable dans la partie <body>



D. L'en-tête du document HTML

1. L'élément < head>

indique au navigateur des données périphériques au document (meta-informations HTML)

a. La balise <head>

Elle doit toujours être placée directement dans la balise https://example.com/html et toujours avant la balise
 https://example.com/html et toujours avant la balise

 due nous verrons par la suite.



b. Les attributs de la balise <head>

Cette balise ne dispose d'aucun attribut.

c. Le contenu de la balise <head>

Sont autorisées les balises suivantes : <title>, <style>, <base>, <link>, <meta>, <script>, <noscript>. Elle vont nous permettre de configurer le contexte de notre page web, par exemple pour :

- optimiser le référencement avec la balise <meta> et <title>
- améliorer l'affichage graphique via CSS avec les balises <link> et <style>
- ajouter du dynamisme avec les balises <script> et <noscript>

2. L'élément < title>

Représente le titre du document.

a. La balise <title>

Elle doit toujours être placée directement dans la balise <head>.

Les navigateurs utilisent ce titre dans l'onglet, dans le titre du programme, et dans le marque page. Les moteurs de recherche s'en servent comme titre de résultat (en bleu dans google).





Exemple:

<title> Mon titre de document </title>

b. Les attributs de la balise <title>

Cette balise ne dispose d'aucun attribut.

c. Le contenu de la balise <title>

Un texte court, sans balise HTML et qui représente le titre de votre document. Par titre, on n'entends le titre « interne » au document, pas le titre qui apparaitra dans notre page avec une taille de police imposante.

Faire attention aux choix de titre, car c'est un éléments clefs pour le confort de navigation de vos internautes, et pour la qualité de votre référencement.

3. L'élément < link>

Elle définit la relation entre un document et une ressource externe.

a. La balise <link>

La balise link> est principalement utilisée pour relier les feuilles de style à notre document HTML. Elle sert également pour l'affichage du petit icône affiché dans l'onglet de notre page.

La balise <link> peut être présent plusieurs fois dans le document, mais uniquement dans la balise <head>.



Exemple:

<link rel="stylesheet" type="text/css" href="theme.css"/>

b. Les attributs de la balise <link>

href	url	Spécifie l'emplacement du fichier lié a cette page HTML
media	text	Permet de préciser sur quels appareil ce lien doit être exploité. Avec un syntaxe simple vous pouvez adapter votre site sur de nombreux supports (braille, tv, mobile)
rel	text	Spécifie le type de relation entre le document HTML et le document lié. Par exemple icon, archives, bookmark, stylesheet
sizes	text	LxH. Largeur x hauteur de l'icône lié. Cet attribut n'est utile que pour les lien avec rel="icon"

href	url	Spécifie l'emplacement du fichier lié a cette page HTML
type		Type mime pour le document lié. Obligatoire pour certain type de relation comme rel="stylesheet" ou l'on rajoute toujours type="text/css"

c. Le contenu de la balise <link>

Cette balise est un élément vide. Il ne contient que des attributs.

4. <style>

Elle permet de déclarer des règles de mise en page.

a. La balise <style>

La balise <style> est quasi exclusivement utilisée pour intégrer des règles CSS directement dans une page HTML en particulier.

La balise <style> peut être présente plusieurs fois dans le document, généralement dans la partie <head>. On peut le placer dans le contenu de notre page, a condition d'utiliser l'attribut scoped.



b. Les attributs de la balise <style>

media	text	Permet de préciser sur quels appareil ce lien doit être exploité. Avec un syntaxe simple vous pouvez adapter votre site sur de nombreux supports (braille, tv, mobile)
type	mime	Type mime pour le document lié. Obligatoire pour certain type de relation comme rel="stylesheet" ou l'on rajoute toujours type="text/css"
Scoped	scoped	Dans la partie <body> permet de définir des styles applicable uniquement à son parent immédiat, et tout ses descendants.</body>

c. Le contenu de la balise <style>

Tout le contenu de cette balise est du code CSS que nous étudierons au chapitre suivant.



5. L'élément <script>





E. Le corps du document HTML

1. L'élément <BODY>

Paramètres du Body

- Background : image de fond (url)
- BGColor : couleur de fond (RVB hexa code red, pink,..)
- Text : couleur de texte par défaut
- Link : couleur de liens par défaut (Alink actif Vlink visité)

a. Les éléments de structure du texte , <h1> à <h6>

Dans le texte affiché, les espaces multiples et les retours à la

ligne ne sont pas interprétés par le navigateur.

Des balises permettent de formater le document.

Paragraphe

paramètres align (left, center, right)

La balise h représente un titre de contenu. Elle utilise des polices de caractères de différentes tailles, de la plus grande à la plus petite :



<h1>une très grande taille</h1>

<h6>une taille vraiment minuscule</h6>

b. Les éléments , et <cite>

mettre du texte en gras (bold)

texte gras

mettre du texte en italique (italic)

<i>texte en italique</i>

souligner du texte (underline)

<u>texte souligné</u>

Le résultat dépend du Navigateur!

Pour insister sur un texte (en général gras)

j'insiste

Pour mettre du texte en valeur (en général italique)

texte en valeur

Pour insérer une citation (en général italique)

<cite>|'union fait |a force</cite>



c. Les éléments de séparation <hr>> et

Ligne horizontale de séparation

<hr>

paramètres width (50 ou 50%) et align (left, center, right)

Saut de ligne

 en xhtml)

d. Les éléments de listes , et et i>

Types de Listes

Liste non ordonnée

<!/ul>

Liste ordonnée

<0|></0|>

Elément de la liste

<|i>

Nos BDs

Astérix

Tintin



- Yoko
- < 0 |>
- Les BDs
- Astérix
- Le Tour de Gaule
- Chez les Belges

- Les DVDs
- Les Gadgets
- </01>

e. Les éléments de listes de définitions <dl>, <dt> et <dd>

- < dl>
- <dt>Astérix</dt>
- <dd>un héros gaulois</dd>
- <dt>Tintin</dt>
- < dd >



un héros belge
br>à vocation

internationale.

</dd>

<dt>Yoko</dt>

< dd >

une héroïne japonaise
à

vocation interplanétaire.

</dd>

</dl>

f. L'élément lien <a>

Le principe de l'Hyper Texte (HTML) se fonde sur les hyperliens

(liens vers d'autres documents ou parties du document).

Ces documents peuvent être des documents HTML, des

images, du son, des films...

Le Navigateur présente ces liens sous forme de mots soulignés

ou d'image encadrée (=ancre ou lien).

Il existe quatre types principaux de liens :

1. vers un document complet distant :



```
<a href="url"> ancre </a>
exemple:
<a href="http://www.auppd.net/real.htm">info</a>
2. vers un document complet local :
<a href=fichier_local"> ancre </a>
exemple : pour accéder à une page du même site
<a href="contact.htm"> cliquer ici </a>
3.vers une partie d'un document local
<a name="etiquette" />
pour définir un point de branchement
(ancre)
<a href=contact.htm#etiquette"> ancre </a>
pour faire le lien.
exemple : pour accéder au paragraphe 3 du
document
<a name="[3]" /> par. 3 : les soucis
<a href="exemple.htm#[3]">note [3] </a>
```



```
vers une partie du document courant :
```


pour définir un point de branchement

 ancre

pour faire le lien.

Les liens peuvent aussi servir à :

adresser un message électronique :

 ancre ou

 ancre

mail

Appeler une instruction javascript :

alerte

alerte 2

g. L'élément image

HTML permet d'insérer des images dans du texte.

Ces images apparaissent dans le corps du texte, mais aussi

comme ancre, ou même comme document référencé.

Les clients WWW acceptent les formats d'images suivants :

HTML L'essentiel

GIF pour les images 256 couleurs (dont transparence)

JPEG pour les images couleurs

PNG comme remplaçant du GIF

Les images dans le texte

La balise permet d'insérer une image dans le texte.

syntaxe:

```
<IMG SRC="image" ALIGN="top" ALT="titre">
```

src = adresse d'une image

```
align = "MIDDLE" "TOP" "BOTTOM" "LEFT" ou "RIGHT"
```

alt = "Un titre" (référencement et accessibilité).

```
<img align="left" src="couv/bd00001.jpg">Astérix
```


br>le fameux héros gaulois

et sa célèbre potion magique.

<br clear="all">

Tintin

br>le fameux héros belge

et son petit chien.



```
<br/>
<img align="center" src="couv/bd00003.jpg">Yoko<br/>
<br>la fameuse héroïne venue de loin.<br/>
<br/>
```

h. L'élément <form> et les formulaires

Un formulaire est une fiche (ensemble de champs de saisie)

que l'utilisateur peut remplir, ces informations ainsi saisies sont
traitées par le serveur à l'aide d'un programme développé à
cet effet.

Un formulaire commence et finit par

<FORM>

</FORM>

avec pour FORM:

ACTION = "programme exécutable" (= destination)

METHOD = GET ou POST (mode d'envoi)

Il existe quatre types de champs :

1.Bouton



<INPUT TYPE NAME VALUE CHECKED>

avec:

TYPE =

- RADIO pour les boutons multiples
- CHECKBOX pour les cases à cocher
- SUBMIT pour envoyer le formulaire une fois rempli
- RESET pour effacer le contenu du formulaire

NAME = "un nom" qui identifie le bouton

VALUE = "valeur" donnée au bouton lorsqu'il est sélectionné

CHECKED définit :

- un bouton par défaut (RADIO) ou
- une case cochée avant saisie (CHECKBOX)

.Texte libre sur une ligne

<INPUT TYPE NAME SIZE>

avec:

TYPE =

- TEXT pour saisir du texte
- PASSWORD les caractères sont représentes par des '*'



• HIDDEN le champ est caché

NAME = "un nom" qui identifie le champ texte

3.Texte libre sur plusieurs lignes

<TEXTAREA NAME ROWS COLS>

</TEXTAREA>

avec:

NAME = "un nom" qui identifie le champ texte

ROWS = nombre de lignes visibles

COLS = nombre de colonnes visibles

.Menu

<SELECT NAME>

</SELECT>

avec:

NAME = "un nom" qui identifie le menu

<OPTION VALUE SELECTED> </OPTION>

représente chaque choix du menu

SELECTED qui signale le choix par défaut



```
<form action="traite.php" method=post>
<input type="radio" name="sex" value="f"
checked>
femme
<input type="radio" name= "sex" value="h">
homme
Nom:
<input type="text" name="nom" size="30">
Prénom:
<input type="text" name="pnom" size="30">
Mail:
<input type="text" name="mail" size=30>
votre opinion
<select name="opinion1">
<option value="1">très bien</option>
<option value="2" selected>bien
<option value="3">passable</option>
</select>
```



vos commentaires

<textarea name="com1" rows="3" cols="40">

</textarea>

<input type="submit" value="envoyer">

<input type="reset" value="annuler">

</form>

i. Les éléments de tableau , <thead>, , et

déclarer le début et la fin d'un tableau :

Table peut avoir les attributs suivants :

border

épaisseur du contour(0 = rien)

cellpadding espace entre l'objet et le contour d'une cellule

cellspacing espace entre les cellules

width largeur du tableau (px ou %)

height hauteur du tableau

bgcolor couleur de fond.



Ligne de tableau (table row)

Une cellule dans une ligne :

tintin

ligne n°1

Astérix

Le Grand Fossé

Cellule 1 de la

ligne n°2

ligne2

Tintin

L'Ile Noire

Cellule 2 de

la ligne2



Yoko

ligne n°3

j. La balise Iframe

Une iframe permet de présenter du contenu externe.

<html>

<body>

<h1>présentation iframe</h1>

<iframe

src="http://www.cmbp.info"

width="150" height="100"

>

texte alternatif.

</iframe>

</body>

</html>



k. Les éléments de bloc neutre <div> et

Ce bloc (calque) est paramétrable en position, couleur etc.

<div> </div>

I. Les entités

Les caractères spéciaux - H.T.M.L.

Un caractère spécial est défini par un code6 entre & et ;

exemple : € affichera € .

Ceci permet d'assurer l'affichage correct du caractère

indépendamment du réglage linguistique du navigateur.

Les caractères < , > et & étant interprétés par HTML, les

séquences suivantes permettent leur affichage.

<<

> >

&..... &

"....."

permet de forcer un "blanc"

Les caractères accentués



é é É É è è ê ê à à ï ç Ç © ou © ©

2. Les attributs globaux

Ces attributs sont utilisable sur tout les éléments de la section
body> de notre document. Ils correspondent a des caractéristiques communes a tous les éléments visuels de notre page.

title	texte	Information complémentaire à un élément (infobulle)
tabindex	nombre	Ordre de tabulation des éléments

HTML L'essentiel

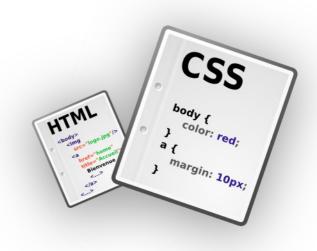
title	texte	Information complémentaire à un élément (infobulle)
style	Règle css	Règle CSS applicable sur l'élément
lang	Code lang	La langue du texte contenu dans cet élément
id	texte	Un identifiant permettant de repérer cet élément unique
hidden	true false	Indique l'etat d'affichage de l'éléments (cacher du contenu)
class	texte	Un ou plusieurs nom de classe (voir partie CSS)

- a. L'attribut title=""pour les info-bulles
- <u>b. L'attribut style=" " de lien avec le CSS</u>
- c. L'attribut class=" " de lien avec le CSS
- <u>d. L'attribut id=" " de lien avec le CSS</u>

3. Liens CSS - HTML



III. LE LANGAGE CSS



Ш	. Le langage CSS	. 16
	A. Format de fichier	. 17
	B. Lien avec le HTML	. 17
	C. Terminologie du langage	. 17
	D. Grammaire des sélecteurs	. 17
	E. Sémantique de la déclaration	.17

A. Format de fichier

B. Lien avec le HTML



C. Terminologie du langage

- 1. Sélecteur
- 2. Déclaration

D. Grammaire des sélecteurs

- 1. Sélecteur d'élément
- 2. Sélecteur de classe
- 3. Sélecteur d'identifiant
- 4. Combinaison de sélecteur
- 5. Condition

E. Sémantique de la déclaration

- 1. Texte et police de caractère
- 2. Couleur & fond
- 3. Espacement, bordures & marges
- 4. Positionnement & taille
- 5. Visibilité



IV. ET APRÈS : JS, HTML5 ET CSS3



V	[/] . Et après : JS, HTML5 et CSS3	18
	A. Notions de JavaScript	19
	B. Les changements du HTML5	20
	C. Les nouveautés de CSS3	20
	D. Les nouveautés lavaScript	20



A. Notions de JavaScript

- 1. Format de fichier
- 2. Lien avec le HTML
- 3. La syntaxe du Javascript
 - a. Instructions & commentaire
 - b. Variables
 - c. Fonctions & Objets
- 4. Les bases du langage
 - a. Opérateurs
 - b. Contrôles & itérations
- 5. Les objets du navigateur
 - <u>a. L'objet Window</u>
 - b. L'objet document & le DOM
 - c. Les événements
- 6. Les frameworks Javascripts
 - a. Avantages & inconvénients
 - b. Principe de fonctionnement



c. Les principaux frameworks

B. Les changements du HTML5

- 1. Pourquoi faire évoluer le HTML
- 2. Balises sémantiques
- 3. Balise video
- 4. Balises de formulaires
- 5. Attribut et interactivités

C. Les nouveautés de CSS3

- 1. Effets de texte et polices web
- 2. Couleurs & images
- 3. Formes et transformations
- 4. Animations, transitions et effets

D. Les nouveautés JavaScript

- 1. Le stockage local
- 2. Les websockets et ajax



3. Geolocalisation





V. UN COMPOSEUR HTML : KOMPOZER



V. Un composeur html : Kompozer......22

Sqdqsqsd

qsdqsd smdkmql sdkqmsl dklsqmdk sqdkqsml dlmqsdk qsdk qldkq mdk-qlm d



VI. ANNEXES

Index lexical

Index des illustrations

Index des tables

Bibliographie