

PROGRAMMATION WEB LANGAGES HTML/CSS

1
Programmation Web
Outils de communication
E. CLERMONT

ORGANISATION DE L'UE DE COMMUNICATION WEB

Apports de cours HTML5/CSS3/PHP/MySQL
9 février: lancement du projet (binômes)
26 & 27 mars : soutenance des projets
30 mars: Examen machine

COMM - Outils de communication

2

SOMMAIRE

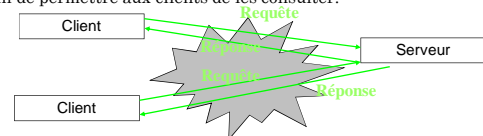
- 1- Introduction à la programmation web
- 2- Les balises HTML de base
- 3- Mise en forme avec CSS
- 4- Mise en page
- 5- Les tableaux
- 6- Les formulaires HTML
- 7- Audio et vidéo

COMM - Outils de communication

3

INTRODUCTION

- Internet est basé sur une architecture de réseau nommée **client-serveur**
- Serveurs : machines hébergeant l'information et en fournissant l'accès
- Clients : machines cherchant cette information
- Les pages composant un site donné sont stockées sur un serveur afin de permettre aux clients de les consulter.



- Différents **protocoles de communication** :
 - http : HyperText transfer protocol
 - ftp : file transfer protocol
 - smtp : send mail transfer protocol
 - etc...

COMM - Outils de communication

4

ACCÈS AUX SERVEURS WEB

- L'accès se fait grâce à une **URL (Uniform Resource Locator)**
Le plus souvent de la forme : <http://nomMachine/nomDocument> avec :
 - nomMachine** qui peut être :
 - Un nom de machine ex : www.ensc.fr
 - L'adresse IP d'une machine ex : 192.247.93.4
 - nomDocument** qui peut être :
 - Le nom du document ex : <http://www.urec.fr/index.html>
 - Le nom du document précédé par son chemin d'accès ex : <http://www.ybet.be/internet-22/liste-meta.htm>
 - Le nom d'un programme suivi des paramètres ex : <http://www.med.univ-rennes1.fr/cgi-bin/idx/rechidx.pl?rech=cardiopathie>
 - Le nom d'un document avec positionnement sur une ancre ex : <http://www.med.univ-rennes1.fr/etud/pharmacopiediatric.htm#1.3>
- Une URL peut être un document local
 - Exemple sur Mac : <file:///KIWI/Utilisateurs/Dupont/index.html>
 - ou sur PC : <file:///C:/Mes Documents/essai.html>
- Important** : Une URL peut être relative au document courant
=> Un lien vers un autre document situés dans un même répertoire peut se faire avec l'URL simple : [document.html](#)

HTML ET CSS : DEUX LANGAGES POUR CRÉER UN SITE WEB



Pour créer un site web, il faut donner des instructions à l'ordinateur. Il ne suffit pas simplement de taper le texte qu'il y aura dans son site.
=> indiquer où placer ce texte, insérer des images, créer des liens entre les pages, ...

Le rôle de HTML et CSS

- HTML (HyperText Markup Language)**
Apparu dès 1991 lors du lancement du Web.
Rôle : gérer et organiser le contenu. C'est donc en HTML que vous écrirez ce que vous souhaitez que la page affiche : du texte, des liens, des images...
 - CSS (Cascading Style Sheets)**
Rôle : gérer l'apparence de la page web (agencement, positionnement, décoration, couleur, taille du texte...).
- Ce langage est venu compléter le HTML en 1996.

Vous pouvez créer un site web uniquement en HTML, mais celui-ci ne sera pas très beau : l'information apparaîtra "brute".
=> le langage CSS vient toujours le compléter.

SOMMAIRE

- 1- Introduction à la programmation web
- 2- Les balises HTML de base
- 3- Mise en forme avec CSS
- 4- Mise en page
- 5- Les tableaux
- 6- Les formulaires HTML
- 7- Audio et vidéo

COMM - Outils de communication

7

HTML : HYPERTEXT MARKUP LANGUAGE

- HTML = ensemble de balises (ou "tags") utilisées pour définir les différents composants d'un document
 - Langage se décrivant comme un ensemble de balises ouvrantes et fermantes avec un contenu de type texte, image, animation.
 - Ces balises sont interprétées par le navigateur client (Opera, Google Chrome, Internet Explorer, Netscape, Mozilla Firefox,...).
- L'interprétation de certaines balises peut varier d'un navigateur à l'autre (ex : taille des textes).
- Les balises HTML utilisent les caractères < et > comme délimiteurs
Exemple : <title>Exemple de balise HTML</title>
 - Un fichier rédigé en HTML doit être suffixé par .html ou .HTML ou .htm ou .HTM

COMM - Outils de communication

8

CONVENTIONS

- Pas de chevauchement des balises
<h1>Eléments <u>correctement emboîtés</u></h1>
<h1>Eléments <u>incorrectement emboîtés</h1></u></h1>
- Balises écrites en minuscules (Préconisation HTML5)
<p class="p1">Balisage correct</p>
<p CLASS="p1">Balisage incorrect en HTML5</p>
- Balises non-vides associées à une balise de fin
Elément liste correct
Elément liste incorrect
- Indication de fin pour les éléments vides

<title/> ⇔ <title></title>
- Valeurs d'attributs encadrées par des guillemets

FONCTIONNEMENT DES BALISES

Les balises s'imbriquent les unes dans les autres.

1	2	3	4
<Balise_1> </Balise_1> <Balise_2> </Balise_2>	<Balise_1> <Balise_2> </Balise_2> </Balise_1>	<Balise_1> <Balise_2> </Balise_2> <Balise_3> </Balise_3> </Balise_1>	<Balise_1> <Balise_2> <Balise_3> </Balise_3> </Balise_2> </Balise_1>

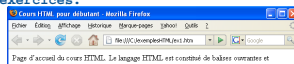
COMM - Outils de communication

STRUCTURE DE BASE D'UN DOCUMENT HTML

Le code source d'une page HTML est toujours structuré de la même manière:

Exemple : fichier ex1.htm

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8" />
    <title>Cours HTML pour débutant</title>
  </head>
  <body>
    Page d'accueil du cours HTML. Le langage HTML est constitué
    de balises ouvrantes et fermantes interprétées par le navigateur
    du client. La page 1 décrit l'essentiel du cours. La page 2 les
    exemples et la page 3 les exercices.
  </body>
</html>
```



Visualisation dans un navigateur

STRUCTURE DE BASE D'UN DOCUMENT HTML

- Doctype (contraction pour "Document Type Declaration") : indispensable pour indiquer qu'il s'agit d'une page web et la version de (X)HTML utilisée
<!doctype html>

- Html

La balise <html> : balise principale du code. Elle englobe le contenu de la page. La balise fermante </html> se trouve tout à la fin du code

Code : HTML

<html>

...

</html>

Une page web est constituée de 2 parties :

- L'en-tête <head> => informations générales sur la page: titre, encodage (pour la gestion des caractères spéciaux), ...

Cette section est généralement assez courte. Les informations que l'en-tête contient ne sont pas affichées sur la page, ce sont simplement des informations générales à destination de l'ordinateur.

- Le corps <body> => partie principale de la page. Tout ce qui est écrit dans cette section sera affiché à l'écran.

=> A l'intérieur du corps, la majeure partie du code sera écrite.

LES EN-TÊTES

- o **<html> </html>** : balises de début et de fin du document HTML
- o **<head> </head>** : informations non affichées concernant le document
- o **<title> </title>** : titre du document
→ information apparaissant dans la barre de titre du navigateur du client
Il est conseillé que le titre soit assez court (moins de 100 caractères en général).
- o **<meta ...>** : situées dans l'en-tête d'une page HTML (entre la balise <head> et </head>
→ commandes guidant les moteurs de recherche pour indexer une page
- o **<body> </body>** : corps du document
→ informations affichées par le navigateur du client web
- o **<!-- Commentaires
pouvant être répartis sur plusieurs lignes
-->** pour l'insertion de commentaires

LE CORPS DE LA PAGE

```
<html>
<head>
  <meta charset="utf-8" />
  <title>Cours HTML pour débutants – Paragraphes</title>
</head>
<body>
  <p>
    Page d'accueil du cours HTML <br/>
    Le langage HTML est constitué de balises ouvrantes
    et fermantes interprétées par le navigateur du client.
  </p>
  <p>
    La page 2 comporte les exemples.
    La page 3 contient les exercices.
  </p>
</body>
</html>
```

- o **<p> </p>** : pour organiser son texte en paragraphes.
- o **
** : pour aller à la ligne

LES DÉLIMITEURS

- o Dans un document HTML, les espaces et retours à la ligne ne sont pas interprétés par les clients web
=> il faut les définir explicitement
- o Les délimiteurs permettent d'aérer et de formater le document
 - Saut de paragraphe : **<p>**
 - Saut à la ligne : **
**
 - Insertion d'une ligne horizontale de séparation : **<hr/>**
 - Division de bloc : **<div>**
Permet également de cadrer un bloc de texte (≠ <P> par l'absence de saut de ligne en début et fin de bloc)

15

COMM - Outils de communication

TITRES DE PARAGRAPHES

- o Représentés par des fontes de caractères de différentes tailles, de la plus grande à la plus petite
 - <h1>Une très grande taille</h1>**
 - <h2>Une grande taille</h2>**
 - <h3>Une taille moyenne</h3>**
 - <h4>Une petite taille</h4>**
 - <h5>Une très petite taille</h5>**
 - <h6>Une taille vraiment minuscule</h6>**
- Attention à ne pas confondre pas avec la balise **<title>** :
La balise **<title>** affiche le titre de la page dans la barre de titre du navigateur.
Les titres **<h>** servent à créer des titres qui seront affichés à l'intérieur de la page web.

16

COMM - Outils de communication

LA MISE EN VALEUR

Pour faire ressortir certaines parties de texte, HTML propose différents moyens:

- o **Mettre un peu en valeur avec la balise **
Utiliser la balise **** a pour conséquence de mettre le texte en italique. C'est le navigateur qui choisit comment afficher les mots. On lui dit que les mots sont assez importants et pour faire ressortir cette information, il change l'apparence du texte en utilisant l'italique.
- o **Mettre bien en valeur avec la balise **
Le texte s'affiche en gras. Le navigateur a choisi d'afficher en gras les mots importants pour les faire plus ressortir.
On pourra décider plus tard en CSS d'afficher les mots "importants" d'une autre façon que le gras si on le souhaite.
- o **Marquer le texte avec la balise <mark> </mark>**
pour faire ressortir visuellement une portion de texte. Le texte n'est pas forcément considéré comme important mais on veut qu'il se distingue bien du reste du texte.
Par défaut, **<mark>** a pour effet de surligner le texte.
On pourra modifier l'affichage en CSS (décider de surligner dans une autre couleur, d'encadrer le texte, etc.).

Page d'accueil du cours HTML

Le langage HTML est constitué de balises ouvrantes et fermantes interprétées par le navigateur du client.

Des cours pour les débutants

La page 1 comporte des aspects de cours La page 2 comporte les exemples La page 3 contient les exercices

Vous pourrez ainsi apprendre à créer un site web sans rien y connaître

Une communauté active

Si vous avez un problème ou un sujet que vous ne comprenez pas, rejoignez-nous sur le forum!

18

LES LISTES

- Éléments très utilisés
- Différentes sortes :

- Listes à puces non ordonnées
 - texte 1
 - texte 2

```
<ul>
  <li> texte 1 </li>
  <li> texte 2 </li>
</ul>
```

- Listes numérotées
 1. texte 1
 2. texte 2

```
<ol>
  <li> texte 1 </li>
  <li> texte 2 </li>
</ol>
```

Avec les lettres de l'alphabet : `<OL type="a">`

- Listes de définition

Terme 1 à définir
Définition du terme 1
Terme 2 à définir
Définition du terme 2
suite du terme 2 *COMM - Outils*

```
<dl>
  <dt> Terme 1 à définir</dt>
  <dd> Définition du terme 1 </dd>
  <dt> Terme 2 à définir </dt>
  <dd> Définition du terme 2 </dd>
  <dd> suite du terme 2 </dd>
</dl>
```

EXEMPLE

Possibilité d'imbriquer les listes :

```
<h3>UEs d'informatique – ENSC 1re année</h3>
<ol> <li>Semestre 1 :
  <ul>
    <li>IPROG : Intro à la programmation</li>
    <li>COMM: Outils de communications - Base de données</li>
  </ul>
</li>
<li>Semestre 2 :
  <ul>
    <li>PAV : Programmation avancée</li>
    <li>COMM: Outils de communications - Programmation
    web</li>
  </ul>
</li>
</ol>
```

UEs d'informatique – ENSC 1ère Année

1. Semestre 1 :
 - IPROG : Intro à la programmation
 - COMM: Outils de communications - Base de données
2. Semestre 2 :
 - PAV : Programmation avancée
 - COMM: Outils de communications - Programmation web

INSÉRER UNE IMAGE

Balise `` - C'est une balise de type orpheline (comme `
`).

La balise doit être accompagnée de **2 attributs obligatoires** :

- **src** : il permet d'indiquer où se trouve l'image que l'on veut insérer.
 - Soit un chemin en absolu (ex. `http://www.site.com/fleur.png`),
 - Soit un chemin en relatif (ce qu'on fait le plus souvent).

- **alt** : "texte alternatif". On doit toujours indiquer un texte alternatif à l'image, c'est-à-dire un court texte qui décrit ce que contient l'image.

Ce texte sera affiché à la place de l'image si celle-ci ne peut pas être téléchargée, ou sur les navigateurs de personnes handicapées (non-voyants) qui ne peuvent malheureusement pas "voir" l'image.

Cela aide aussi les robots des moteurs de recherche pour les recherches d'images.

Les images doivent se trouver obligatoirement à l'intérieur d'un **paragraphe** (`<p>` `</p>`).

```
<p>
  Une photo de montagne:<br />
  
</p>
```

INSÉRER UNE IMAGE

Formats d'images à placer :

- photo : utilisez un JPEG.
- graphique avec peu de couleurs (moins de 256) : utilisez un PNG8 bits, ou éventuellement un GIF.
- graphique avec beaucoup de couleurs : PNG24 bits.
- image animée : GIF animé.

Erreurs à éviter

- Bannir les autres formats

Les autres formats non cités ici, comme le format BITMAP (*.bmp) sont à bannir car bien souvent ils ne sont pas compressés, donc trop gros. Ils ne sont pas du tout adaptés au web. On peut en mettre sur son site, mais le chargement sera vraiment extrêmement long.

- bien choisir le nom de l'image (noms en minuscules, sans espaces ni accents. Par exemple : `mon_image.png`)

COMM - Outils de communication

22

INSÉRER UNE IMAGE

- Ajouter une **infobulle**

L'attribut fait pour afficher une bulle d'aide est le même que pour les liens : il s'agit de **title**.

Cet attribut est facultatif (contrairement à alt).

```
<p>
  Une photo de montagne:<br />
  
</p>
```

COMM - Outils de communication

23

HYPERLIENS

- Possibilité de définir un lien vers un autre document en utilisant la balise :

```
<a href="url"> texte du lien </a>
```

- Dans le navigateur, *texte du lien* apparaîtra (en général) souligné et de couleur bleue

→ informe l'utilisateur que cette portion de texte est un **hyperlien** qui va appeler l'URL correspondante

- L'hyperlien peut être de plusieurs types :

- **Un autre document HTML**

D'un autre site (chemin absolu)

```
<a href="http://www.urec.fr/index.html">Serveur Web de l'UREC</a>
```

Vers une autre page de son propre site (chemin relatif)

```
<a href="page2.html"> Ma page 2 </a>
```

Si la page courante et `page2.html` sont situées dans le même répertoire

```
<a href="contenu/autredossier/page2.html">
```

```
<a href="..page2.html">
```

- **Un lien vers une image**

```
<a href="retina.jpg">Image de la rétine</a>
```

- **Un lien sur fichier son, ou vidéo à charger, un fichier compressé...**

```
<a href="anim.mov">Cliquez ici pour visionner l'animation</a>
```

- **Un lien vers un programme de composition de mail**


```
<a href="mailto:adresse"> texte </a>
```

Cette directive `mailto` permet de spécifier le champ « sujet » du mail :

```
<a href="mailto:adresse?subject=Réunion du 29/01/13">texte</a>
```

EXEMPLE

Voici un exemple de document HTML



Voici quelques liens possibles :

- [Carte des serveurs Web](#)
- [Description de la grippe](#)

```
<html>
<head>
<title>Exemple de document HTML</title>
</head>
<body>
<h1>Voici un exemple de document HTML</h1>

Voici quelques liens possibles :
<ul>
<li><a href="http://Web.urec.fr/france/France.html">
Carte des serveurs Web</a> </li>
<li><a href="http://www.med.univ-rennes1.fr/cgi-
bin/adm/reponse.pl?prg=1&txt=grippe">
Description de la grippe</a> </li>
</ul>
<hr/>
</body>
</html>
```

25

COMM - Outils de communication

LES ANCRES

- o Liens vers une partie spécifique d'un document local

=>attribut id pour donner un nom à l'ancre.

<h2 id="mon_ancre">Titre</h2>

- o Ensuite, faire un lien, mais cette fois l'attribut href contiendra un dièse (#) suivi du nom de l'ancre.

Aller vers l'ancre

- o Ces ancres pourront être appelées (utilisation du # suivi de *nom_ancre*) dans le même document, ce qui permet de créer des références croisées
- Sommaire
- Retour en haut de page

26

COMM - Outils de communication

EXEMPLE

```
<html>
<head>
<title>TITRE</title>
</head>
<body>
Sommaire :
<ul>
<li><a href="#parag1">Paragraphe I</a> </li>
<li><a href="#parag2">Paragraphe II</a> </li>
</ul>
...
<h1 id="parag1">Paragraphe I</h1>
bla, bla ...
...
<h2 id="parag2">Paragraphe II</h2>
bla, bla ...
</body>
</html>
```

Sommaire :

- [Paragraphe I](#)
- [Paragraphe II](#)

...

Paragraphe I

bla, bla ...

Paragraphe II

bla, bla ...

27

COMM - Outils de communication

EXEMPLE 2

- o Ces ancres d'un document peuvent être appelées depuis un autre document.

```
<html>
<head>
<title>Deuxième document</title>
</head>
<body>
bla, bla...
<a href="mon_fichier.html#parag2">
Vous accédez ici au paragraphe II du
document mon_fichier
</a>
</body>
</html>
```

28

COMM - Outils de communication

CAS PRATIQUES D'UTILISATION DES LIENS

- o Un lien qui ouvre une nouvelle fenêtre

=> rajouter **target="_blank"** à la balise **<a>** :

<p>Bonjour. Souhaitez-vous visiter le
** Site de l'ENSC ?</p>**

Selon la configuration du navigateur, la page s'affichera dans une nouvelle fenêtre ou un nouvel onglet. Vous ne pouvez pas choisir entre l'ouverture d'une nouvelle fenêtre ou d'un nouvel onglet.

- o Un lien pour télécharger un fichier

Même principe que pour un lien vers une page web, mais en indiquant cette fois le nom du fichier à télécharger.

<p>
Télécharger le fichier
</p>

Le navigateur, voyant qu'il ne s'agit pas d'une page web à afficher, va lancer la procédure de téléchargement lorsqu'on cliquera sur le lien.

29

COMM - Outils de communication

SOMMAIRE

- 1- Introduction à la programmation web
- 2- Les balises HTML de base
- 3- Mise en forme avec CSS
- 4- Mise en page
- 5- Les tableaux
- 6- Les formulaires HTML
- 7- Audio et vidéo

30

COMM - Outils de communication

CSS

CSS (Cascading Style Sheets) est un langage qui vient compléter le HTML (1996)

Rôle : **Gérer la mise en forme du site** : couleur, taille, police des textes, bordures, fond,...

Aujourd'hui, CSS3.

Exemple : site csszengarden
<http://www.csszengarden.com/tr/francais/>

Où écrire le CSS?

- Dans un fichier dédié .css (méthode la plus recommandée)
- Dans l'en-tête head du fichier HTML
- Directement dans les balises du fichier HTML via un attribut style (À éviter)

COMM - Outils de communication

31

DÉFINITION DU CSS DANS UN FICHIER DÉDIÉ .CSS

Page.html

Code : HTML -
<!DOCTYPE html>
<html>
 <head>
 <meta charset="utf-8" />
 <link rel="stylesheet" href="style.css" />
 <title>Premiers tests du CSS</title>
 </head>
 <body>
 <h1>Mon super site</h1>
 <p>Bonjour et bienvenue sur mon site !</p>
 <p>Pour le moment, mon site est un peu vide. Patientez encore un peu !</p>
 </body>
</html>

Cette ligne indique que ce fichier HTML est associé à un fichier appelé style.css chargé de sa mise en forme.

style.css

p { color : blue; }

COMM - Outils de communication

32

DÉFINITION DU CSS DANS L'EN-TÊTE HEAD

=> directement dans une balise <style> à l'intérieur de l'en-tête <head>.

Exemple:

Code : HTML -
<!DOCTYPE html>
<html>
 <head>
 <meta charset="utf-8" />
 <style> p { color: blue; } </style>
 <title>Premiers tests du CSS</title>
 </head>
 <body>
 <h1>Mon super site</h1>
 <p>Bonjour et bienvenue sur mon site !</p>
 <p>Pour le moment, le site est un peu vide. Patientez encore un peu </p>
 </body>
</html>

DÉFINITION DU CSS DIRECTEMENT DANS LES BALISES

o Directement dans les balises (non recommandé)

méthode à manipuler avec précaution : vous pouvez ajouter un attribut style à n'importe quelle balise.

Code : HTML
<!DOCTYPE html>
<html>
 <head>
 <meta charset="utf-8" />
 <title>Premiers tests du CSS</title>
 </head>
 <body>
 <h1>Mon super site</h1>
 <p style="color: blue;">Bonjour et bienvenue sur mon site !</p>
 <p>Pour le moment, le site est un peu vide. Patientez encore un peu </p>
 </body>
</html>

COMM - Outils de communication

34

CSS : APPLIQUER UN MÊME STYLE À PLUSIEURS BALISES

```
h1 { color: blue; }  
em { color: blue; }
```

Avec cette feuille de style,

titres <h1> et textes importants doivent s'afficher en bleu.

Il existe un moyen en CSS d'aller plus vite si les 2 balises doivent avoir la même présentation.

=> combiner la déclaration en séparant les noms des balises par une virgule :

```
h1, em { color: blue; }
```

COMM - Outils de communication

35

CSS: APPLIQUER UN STYLE – ATTRIBUTS CLASS ET ID

o Appliquer un style : class et id

Avec la méthode précédente, TOUS les paragraphes seront par exemple écrits en bleu.

- o Comment faire pour que seulement certains paragraphes soient écrits d'une manière différente ?
 - On pourrait placer notre code CSS dans un attribut style sur la balise que l'on vise, mais ce n'est pas recommandé (option d'un fichier CSS externe préférable).

Pour résoudre le problème, on peut utiliser des attributs spéciaux qui fonctionnent sur toutes les balises :

- L'attribut class
- L'attribut id

COMM - Outils de communication

36

CSS: APPLIQUER UN STYLE – ATTRIBUTS CLASS ET ID

Attribut class

Attribut class : attribut que l'on peut insérer avec n'importe quelle balise : titre, paragraphe, image, ...

```
<h1 class=""></h1>
```

```
<p class=""></p>
```

```
<img class="" />
```

```
maPage.html
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8" />
    <link rel="stylesheet" href="style.css" />
    <title>Premiers tests du CSS</title>
  </head>
  <body>
    <h1>Mon super site</h1>
    <p class="introduction">Bonjour et bienvenue sur mon site !</p>
    <p>Pour le moment, mon site est un peu <em>vide</em>.
    Patientez encore un peu !</p>
  </body>
</html>
```

```
.introduction { color: blue; }
```

37

CSS: APPLIQUER UN STYLE – ATTRIBUTS CLASS ET ID

Attribut id

Il fonctionne de la même manière que class MAIS il ne peut être utilisé *QU'UNE fois* dans le code.

Quel intérêt ? Il y en a assez peu, cela vous sera utile si vous faites du Javascript plus tard pour reconnaître certaines balises.

En pratique, id seulement pour des éléments uniques sur la page, comme par exemple le logo :

Code HTML ``

En utilisant des id, dans le CSS il faut faire précéder le nom de l'id par un dièse (#) :

Code CSS `#logo { /* Indiquer les propriétés CSS ici */ }`

Remarque : 2 balises peuvent avoir le même nom avec l'attribut class. Un nom d'id doit en revanche être unique dans la page HTML.

CSS: APPLIQUER UN STYLE : BALISES UNIVERSELLES

On peut avoir besoin d'appliquer une class (ou un id) à certains textes qui ne sont pas à l'origine entourés par des balises.

class est un attribut => applicable uniquement sur une balise.

Exemple, pour modifier uniquement "bienvenue" dans le paragraphe suivant :

Code : HTML

```
<p>Bonjour et bienvenue sur mon site !</p>
```

Pas de balise autour de « bienvenue » => utilisation des balises universelles.

39

COMM - Outils de communication

CSS: APPLIQUER UN STYLE : BALISES UNIVERSELLES

=> 2 balises dites *universelles* :

o `` `` : balise de type *inline*.

Balise que l'on place au sein d'un paragraphe de texte, pour sélectionner certains mots uniquement.

`` et `` sont de la même famille.

=> s'utilise donc au milieu d'un paragraphe.

Code : HTML

```
<p>Bonjour et <span class="salutations">bienvenue</span> sur mon site !</p>
```

Code : CSS

```
.salutations { color: blue; }
```

o `<div>` `</div>` : balise de type *block* qui entoure un bloc de texte.

Les balises de la même famille sont `<p>`, `<h1>`, etc. Ces balises ont quelque chose en commun : elles créent un nouveau "bloc" dans la page, et provoquent donc obligatoirement un retour à la ligne. `<div>` est une balise fréquemment utilisée dans la construction d'un design, comme nous le verrons plus tard.

COMM - Outils de communication

40

CSS: APPLIQUER UN STYLE : SÉLECTEURS AVANCÉS

* : sélecteur universel

Code : CSS `* { }`

Sélectionne toutes les balises sans exception.

* : sélecteur universel.

B1 B2 : une balise contenue dans une autre

Code : CSS `h3 em { }`

Sélectionne toutes les balises `` situées à l'intérieur d'une balise `<h3>`.

Attention : Il n'y a pas de virgule entre les deux noms de balise.

Exemple de code HTML correspondant :

```
<h3>Titre avec <em>texte important</em></h3>
```

B1 + B2 : une balise qui en suit une autre

Code : CSS `h3 + p { }`

Sélectionne la première balise `<p>` située après un titre `<h3>`.

Exemple : `<h3>Titre</h3> <p>Paragraphe</p>`

41

CSS: APPLIQUER UN STYLE : SÉLECTEURS AVANCÉS

A[attribut] : une balise qui possède un attribut

Code : CSS `a[title] { ... }`

Sélectionne tous les liens `<a>` qui possèdent un attribut title.

Exemple Code : HTML : ``

A[attribut="Valeur"] : une balise, un attribut et une valeur exacte

Code : CSS `a[title="Cliquez ici"] { }`

Même principe que dans le cas précédent, mais l'attribut doit en plus avoir exactement pour valeur "Cliquez ici".

Exemple : Code : HTML ``

A[attribut*="Valeur"] : une balise, un attribut et une valeur

Code : CSS `a[title*="ici"] { }`

Idem, l'attribut doit cette fois contenir dans sa valeur le mot "ici" (peu importe sa position).

Exemple Code : HTML ``

42

COMM

CSS: MISE EN FORME DU TEXTE - TAILLE

La taille

Pour modifier la taille du texte => utilisation de la propriété font-size.

Plusieurs techniques proposées :

- Indiquer une taille absolue : en pixels, en centimètres ou millimètres. Cette méthode est très précise mais ne l'utiliser que si c'est absolument nécessaire, car on risque parfois d'indiquer une taille trop petite pour certains lecteurs.

Code : CSS

```
p {
  font-size: 14px; /* Paragraphes de 14 pixels */
}
h1 {
  font-size: 40px; /* Titres de 40 pixels */
}
```

- Indiquer une taille relative : en pourcentage, "em" ou "ex", cette technique a l'avantage d'être plus souple. Elle s'adapte plus facilement aux préférences de taille des visiteurs.

COMM - Outils de communication

43

CSS: MISE EN FORME DU TEXTE - TAILLE

Une valeur relative

Méthode recommandée : le texte s'adapte alors plus facilement aux préférences de tous les visiteurs.

Il y a plusieurs moyens d'indiquer une valeur relative.

- o Vous pouvez par exemple écrire la taille avec les mots suivants :

xx-small : minuscule
x-small : très petit
small : petit
medium : moyen
large : grand
x-large : très grand
xx-large : gigantesque.

Code : CSS

```
p {
  font-size: small;
}
h1 {
  font-size: large;
}
```

Défaut : « seulement » 7 tailles disponibles (car il n'y a que 7 mots).

COMM - Outils de communication

44

CSS: MISE EN FORME DU TEXTE - TAILLE

- o Autre technique : indiquer la taille en « em », une unité spécifique au CSS.

1em : texte de taille normale.

Pour grossir le texte, inscrire une valeur supérieure à 1.

Par exemple : 1.2em.

Pour réduire le texte, inscrivez une valeur inférieure à 1.

Par exemple, comme 0.7em.

Code CSS:

```
p {
  font-size: 0.8em;
}
h1 {
  font-size: 1.3em;
}
```

D'autres unités disponibles :

- "ex" qui fonctionne sur le même principe que « em » mais qui est plus petit de base)
- le pourcentage (80%, 130%...).

```
p { font-size: 90%; }
```

CSS: MISE EN FORME DU TEXTE – POLICE

La police

Pour qu'une police s'affiche correctement, il faut les internautes l'aient. Si un internaute n'a pas la même police, son navigateur prendra une police par défaut (une police standard) qui n'aura peut-être rien à voir avec ce à quoi vous vous attendiez.

Depuis CSS 3, possible de faire télécharger automatiquement une police au navigateur.

Modifier la police utilisée

=> propriété CSS : font-family.

Code : CSS

```
balise {
  font-family: police;
}
```

=> Pour éviter qu'il n'y ait de problème si l'internaute n'a pas la même police que vous, on précise en général plusieurs noms de police, séparés par des virgules :

Code : CSS

```
balise {
  font-family: police1, police2, police3, police4;
}
```

COMM

CSS: MISE EN FORME DU TEXTE – POLICE

Le navigateur essaiera d'abord de mettre la police1. S'il ne l'a pas, il essaiera de mettre la police2, et ainsi de suite.

En général, on indique en tout dernier « serif » (police standard qui ne se sera utilisée que si aucune autre police n'a été trouvée).

Code : CSS

```
p {
  font-family: Impact, "Arial Black", Arial, Verdana, serif;
}
```

Si la police comporte des espaces, il faut ajouter des guillemets

Liste de polices qui fonctionnent bien sur la plupart des navigateurs :

Arial
Arial Black
Comic Sans MS
Courier New
Georgia
Impact
Times New Roman
Trebuchet MS
Verdana

COMM - Outils de communication

47

CSS: MISE EN FORME DU TEXTE – POLICE

Comment utiliser une police spécifique pour un site web?

Avec CSS 3, il existe un moyen d'utiliser n'importe quelle police sur son site. Cela fonctionne bien avec la plupart des navigateurs.

Mais attention, il y a des défauts :

-le navigateur de vos visiteurs devra télécharger automatiquement le fichier de la police, qui peut peser 1 Mo voire plus.

La plupart des polices sont soumises au droit d'auteur, il n'est donc pas légal de les utiliser sur son site. Certains sites comme fontquirrel.com ou dafont.com en proposent un certain nombre libres de droit à télécharger.

Il existe plusieurs formats de fichier de police, et ils ne fonctionnent pas sur tous les navigateurs.

COMM - Outils de communication

48

CSS: MISE EN FORME DU TEXTE – POLICE

Différents formats de fichiers de police :

.ttf : TrueType Font. Fonctionne sur IE9 et tous les autres navigateurs.

.eot : Embedded OpenType. Fonctionne sur Internet Explorer uniquement, toutes versions. Ce format est propriétaire de Microsoft.

.otf : OpenType Font. Ne fonctionne pas sur Internet Explorer

.svg : SVG Font. Le seul format reconnu sur iPhone et iPad pour le moment.

.woff : Web Open Font Format. Nouveau format conçu pour le Web qui fonctionne sur IE9 et tous les autres navigateurs.

En CSS, pour définir une nouvelle police :

```
Code : CSS
@font-face {
  font-family: 'MaPolicePreferee';
  src: url('MaPolicePreferee.eot');
}
```

Le fichier de police (ici MaPolicePreferee.eot) doit être stocké dans le même dossier que le fichier CSS (ou dans un sous-dossier si utilisation d'un chemin relatif).

CSS: MISE EN VALEUR DU TEXTE

Italique, gras, souligné...

- Mettre en italique

=> font-style, qui peut prendre 3 valeurs :

italic : le texte sera mis en italique.

oblique : le texte sera aussi mis en italique (en penchant les lettres).

normal : le texte sera normal (par défaut).

```
Code : CSS
h2
{
  font-style: italic;
}
```

- Mettre en gras

=> Propriété CSS : font-weight qui prend les valeurs :

bold : le texte sera en gras.

normal : le texte sera écrit normalement (par défaut).

```
Code : CSS
h1
{
  font-weight: bold;
}
```

COMM - Outils de communication

50

CSS: MISE EN FORME DU TEXTE

- Souligner

=> propriété CSS : text-decoration.

Elle permet entre autres de souligner le texte, mais pas seulement.

Valeurs qu'elle peut prendre :

underline : souligné.

line-through : barré.

overline : ligne au-dessus.

blink : clignotant. Ne marche pas sur tous les navigateurs (Internet Explorer et Google Chrome notamment).

none : normal (par défaut).

```
Code : CSS
h1
{
  text-decoration: blink;
}
.souligne
{
  text-decoration: underline;
}
.barre
{
  text-decoration: line-through;
}
.ligne_dessus
{
  text-decoration: overline;
}
```

COMM - Outils de communication

CSS: MISE EN FORME DU TEXTE - ALIGNEMENT

- L'alignement

Le langage CSS nous permet de faire tous les alignements que l'on connaît : à gauche, centré, à droite et justifié.

=> propriété text-align qui prend les valeurs :

left : le texte sera aligné à gauche (c'est le réglage par défaut).

center : le texte sera centré.

right : le texte sera aligné à droite.

justify : le texte sera "justifié".

```
Code : CSS
h1
{
  text-align: center;
}
p
{
  text-align: justify;
}
.signature
{
  text-align: right;
}
```

Impossible de modifier l'alignement du texte d'une balise inline (comme , <a>, , ...).

L'alignement ne fonctionne que sur des balises de type block (<p>, <div>, <h1>, <h2>, ...)

Logique : on ne peut pas modifier l'alignement de quelques mots au milieu d'un paragraphe. C'est en général le paragraphe entier qu'il faut aligner

CSS: MISE EN FORME DU TEXTE - COULEUR

Modifier la couleur du texte => propriété color

Méthode 1 : Indiquer le nom de la couleur (en anglais)

Le seul défaut de cette méthode : il n'existe que 16 couleurs "standard".

Pour passer tous les titres en vert, on peut donc écrire :

Code : CSS

```
h1
{
  color: green;
}
```

white
silver
grey
black
red
maroon
lime
green
yellow
olive
blue
navy
fuchsia
purple
aqua
teal

COMM - Outils de communication

53

CSS: MISE EN FORME DU TEXTE - COULEUR

Méthode 2 : La notation hexadécimale

Un nom de couleur en hexadécimal doit toujours commencer par écrire un dièse (#), suivi de 6 lettres ou chiffres allant de 0 à 9 et de A à F.

Ces lettres ou chiffres fonctionnent deux par deux. Les 2 premiers indiquent une quantité de rouge, les 2 suivants une quantité de vert, et les 2 derniers une quantité de bleu. En mélangeant ces quantités, on peut obtenir la couleur désirée.

#000000 : noir

#FFFFFF : blanc.

Exemple pour appliquer la couleur blanche en hexadécimal sur les paragraphes :

```
Code : CSS
p
{
  color: #FFFFFF;
}
```

COMM - Outils de communication

54

CSS: MISE EN FORME DU TEXTE - COULEUR

Méthode 3 : La méthode RGB (Red-Green-Blue)

Comme pour la notation hexadécimale, on doit définir une quantité de rouge, de vert et de bleu pour choisir une couleur. Ces quantités sont toujours comprises entre 0 et 255.

=> rgb(Rouge, Vert, Bleu)

```
Code : CSS
p
{
    color : rgb(240,96,204);
}
```

COMM - Outils de communication

55

CSS: COULEUR DE FOND

Couleur de fond

Pour indiquer une couleur de fond

=> propriété CSS : background-color.

Elle s'utilise de la même manière que la propriété color :

- taper le nom d'une couleur,
- l'écrire en notation hexadécimale
- utiliser la méthode RGB.

Pour indiquer la couleur de fond de la page web, il faut travailler sur la balise <body>.

```
Code : CSS
body /* On travaille sur la balise body, donc sur TOUTE la page */
{
    background-color: black; /* Le fond de la page sera noir */
    color: white; /* Le texte de la page sera blanc */
}
```

56

CSS ET HÉRITAGE

En CSS, si on applique un style à une balise,

toutes les balises qui se trouvent à l'intérieur prendront le même style.

La balise <body> contient entre autres les balises de paragraphe <p> et de titre <h1>. Si on applique une couleur de fond noire et une couleur de texte blanche à la balise <body>, tous les titres et paragraphes auront eux aussi une couleur de fond noire et un texte de couleur blanche

=> phénomène d'héritage : les balises qui se trouvent à l'intérieur d'une autre balise "héritent" de ses propriétés.

Les propriétés CSS sont héritées en cascade : si vous donnez un style à un élément, tous les sous-éléments auront le même style.

(d'où le nom "CSS", pour "Cascading Style Sheets",
càd "Feuilles de style en cascade")

COMM - Outils de communication

57

CSS ET HÉRITAGE

Le CSS et l'héritage

Si par la suite vous voulez vos titres en rouge, ce style aura la priorité et vos titres seront donc en rouge.

En revanche, si vous n'indiquez rien de particulier (comme précédemment), les titres hériteront de la couleur blanche.

Ce principe est vrai pour l'ensemble des propriétés CSS:

Toutes les propriétés CSS seront héritées : vous pouvez par exemple demander une mise en gras dans la balise <body>, et tous vos titres et paragraphes seront en gras.

COMM - Outils de communication

58

CSS: COULEUR DE FOND

Exemple d'héritage avec la balise <mark>

Possibilité de changer la couleur de fond de n'importe quel élément : titres, paragraphes, certains mots... Dans ce cas, ils apparaîtront surlignés (comme si on avait passé un surligneur).

```
Code : CSS
body
{
    background-color : black;
    color: white;
}
mark
/* La couleur de fond est prioritaire à celle de toute la page */
background-color: red;
}
```

COMM - Outils de communication

59

CSS: IMAGE DE FOND

L'image de fond ne s'applique pas forcément à la page entière.

On peut aussi associer une image de fond aux titres, paragraphes, ...

Appliquer une image de fond

=> Propriété : background-image.

Valeur : url("nom_de_l_image.png").

Par exemple :

```
body
{
    background-image: url("neige.png");
}
```

L'image peut être au format PNG, JPEG ou GIF.

L'adresse indiquant où se trouve l'image de fond peut être écrite en absolu (http://...) ou en relatif (fond.png).

Attention : Pour une adresse en relatif dans le fichier CSS, l'adresse de l'image doit être indiquée par rapport au fichier .css et non pas par rapport au fichier .html.

CSS: IMAGE DE FOND

Options disponibles pour l'image de fond

On peut compléter la propriété **background-image** par plusieurs autres propriétés qui permettent de changer le comportement de l'image de fond.

- background-attachment : fixer le fond
on voit alors le texte "glisser" par-dessus le fond.

2 valeurs sont disponibles :

fixed : l'image de fond reste fixe.
scroll : l'image de fond défile avec le texte (par défaut).

```
Code : CSS
body
{
    background-image: url("neige.png");
    background-attachment: fixed; /* Le fond restera fixe */
}
```

CSS: IMAGE DE FOND

Options disponibles pour l'image de fond

- background-repeat : répétition du fond

Par défaut, l'image de fond est répétée en mosaïque.

On peut changer ce comportement avec la propriété **background-repeat** :

no-repeat : fond non répété=>image unique sur la page.

repeat-x : fond répété uniquement sur la première ligne, horizontalement.

repeat-y : fond répété uniquement sur la première colonne, verticalement.

repeat : le fond répété en mosaïque (par défaut).

Exemple d'utilisation :

```
Code : CSS
body
{
    background-image: url("soleil.png");
    background-repeat: no-repeat;
}
```

CSS: TRANSPARENCE

Le CSS nous permet de jouer très facilement avec les niveaux de transparence des éléments

- La propriété **opacity** permet d'indiquer le niveau d'opacité (c'est l'inverse de la transparence).

Valeur 1: l'élément est totalement opaque : comportement par défaut.

Valeur 0: l'élément sera totalement transparent.

Il faut donc choisir une valeur comprise entre 0 et 1.

Par exemple avec 0.6, l'élément sera opaque à 60%... et on verra donc à travers.

```
Code : CSS
p
{
    opacity: 0.6;
}
```

Si vous appliquez la propriété **opacity** à un élément de la page, tout le contenu de cet élément sera rendu transparent (même les images, les autres blocs à l'intérieur, etc.).

Si vous voulez juste rendre la couleur de fond transparente, utilisez plutôt la notation **RGBA** ci-après.

CSS: TRANSPARENCE

- La notation **RGBA**: Ils'agit en fait de la notation **RGB** vue précédemment, mais avec un quatrième paramètre : le niveau de transparence (appelé "canal alpha").

valeur 1, le fond est complètement opaque.

Valeur inférieure à 1, il est transparent.

```
Code : CSS
p
{
    background-color: rgba(255, 0, 0, 0.5); /* Fond rouge à moitié transparent */
}
```

=> même effet qu'avec **opacity** .

CSS: BORDURES ET OMBRES

Bordures standard

=> super-propriété **border**

Pour border on peut utiliser jusqu'à 3 valeurs pour modifier l'apparence de la bordure :

- La largeur : valeur en pixels (exemple: 2px).
- La couleur : soit un nom de couleur, soit une valeur hexadécimale, soit une valeur rgb.
- Le type de bordure : simple trait, ou des pointillés, ou tirets,...

Valeurs disponibles :

none : pas de bordure (par défaut)
solid : un trait simple.
dotted : pointillés.
dashed : tirets.
double : bordure double.
groove : en relief.
ridge : autre effet relief.
inset : effet 3D enfoncé.
outset : effet 3D surélevé.

CSS: BORDURES ET OMBRES

Ainsi, pour une bordure bleue en tirets de 2 pixels autour des titres:

```
Code : CSS
h1
{
    border: 2px blue dashed;
}
```

Pour des bordures différentes en fonction du côté (haut, bas, gauche ou droite), vous pouvez le faire en utilisant ces 4 propriétés :

border-top : bordure en haut.

border-bottom : bordure en bas.

border-left : bordure à gauche.

border-right : bordure à droite.

Il existe aussi des équivalents pour paramétrer chaque détail de la bordure si besoin: **border-top-width** pour modifier l'épaisseur de la bordure du haut, **border-top-color** pour la couleur du haut, etc.

Ce sont aussi des super-propriétés, elles fonctionnent comme **border** mais ne s'appliquent donc qu'à un seul côté.

CSS: BORDURES ET OMBRES

Pour ajouter une bordure uniquement à gauche et à droite des paragraphes, on écrira donc :

```
Code : CSS
p
{
  border-left: 2px solid black;
  border-right: 2px solid black;
}
```

On peut modifier les bordures de n'importe quel type d'élément sur sa page. Ici sur les paragraphes, mais on peut aussi modifier la bordure de ses images, des textes importants comme , etc.

CSS: BORDURES ET OMBRES

- Bordures arrondies
- ⇒ propriété border-radius
- ⇒ Il suffit d'indiquer la taille ("l'importance") de l'arrondi en pixels :

```
Code : CSS
p
{
  border-radius: 10px;
}
```

CSS: OMBRES

2 types d'ombres :

- Les ombres des boîtes

- Les ombres du texte

- **box-shadow** : les ombres des boîtes

La propriété box-shadow s'applique à tout le bloc et prend 4 valeurs dans cet ordre :

1. Le décalage horizontal de l'ombre
2. Le décalage vertical de l'ombre
3. L'adoucissement du dégradé
4. La couleur de l'ombre

Exemple, pour une ombre noire de 6 pixels, sans adoucissement:

- **text-shadow** : l'ombre du texte

Avec text-shadow, ajout d'une ombre directement sur les lettres de votre texte .

Les valeurs fonctionnent exactement de la même façon que box-shadow : décalage, adoucissement et couleur.

```
Code : CSS
p
{
  box-shadow: 6px 6px 0px black;
}
```

```
Code : CSS
p
{
  text-shadow: 2px 2px 4px black;
}
```

CSS: CRÉATION D'APPARENCES DYNAMIQUES

- modifier l'apparence des éléments de façon dynamique, c'est-à-dire que des éléments peuvent changer de forme une fois que la page a été chargée.
- => les pseudo-formats.

- Changement d'apparence :
 - Au survol
 - Lors du clic
 - Lors du focus (élément sélectionné)
 - Lorsqu'un lien a été visité

CSS: CRÉATION D'APPARENCES DYNAMIQUES

- Au survol

⇒ hover : information que l'on rajoute après le nom de la balise (ou de la classe) dans le CSS:

```
Code : CSS
a:hover
{
  }
```

Apparence que doivent avoir les liens lorsqu'on pointe dessus

Même si on l'utilise souvent sur les liens, possibilité de modifier l'apparence de n'importe quel élément. Par exemple, modifier l'apparence des paragraphes lorsqu'on pointe dessus.

```
Code : CSS
a /* Liens par défaut (non survolés) */
{
  text-decoration: none;
  color: red;
  font-style: italic;
}
a:hover /* Apparence au survol des liens */
{
  text-decoration: underline;
  color: green;
}
```

CSS: CRÉATION D'APPARENCES DYNAMIQUES

- Au clic et lors de la sélection
- **:active** : au moment du clic

Le pseudo-format :active permet d'appliquer un style particulier au moment du clic. En pratique, il n'est utilisé que sur les liens.

Le lien gardera cette apparence très peu de temps.

On peut par exemple changer la couleur de fond du lien lorsqu'on clique dessus :

```
Code : CSS
a:active /* Quand le visiteur clique sur le lien */
{
  background-color: #FFCC66;
}
```

- **:focus** : lorsque l'élément est sélectionné

Le pseudo-format :focus applique un style lorsque l'élément est sélectionné.

Une fois cliqué, le lien reste "sélectionné" (il y a une petite bordure en pointillés autour).

```
a:focus /* Quand le visiteur sélectionne le lien */
{
  background-color: #FFCC66;
}
```

CSS: CRÉATION D'APPARENCES DYNAMIQUES

- Lorsque le lien a déjà été visité
possible d'appliquer un style à un lien vers une page qui a déjà été vue. Par défaut, le navigateur colore le lien en un violet

- Possibilité de changer cette apparence avec `:visited`

```
Code : CSS
a:visited /* Quand le visiteur a déjà vu la page concernée */
{
  color: #AAA; /* Appliquer une couleur grise */
}
```

- Si vous ne souhaitez pas que les liens déjà visités soient colorés d'une façon différente, il vous faudra leur appliquer la même couleur qu'aux liens normaux.

SOMMAIRE

- 1- Introduction à la programmation web
- 2- Les balises HTML de base
- 3- Mise en forme avec CSS
- 4- Mise en page
- 5- Les tableaux
- 6- Les formulaires HTML
- 7- Audio et vidéo

COMM - Outils de communication

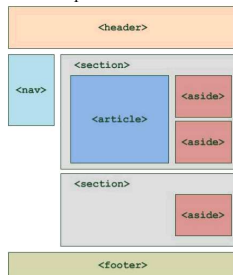
74

MISE EN PAGE

Structurer sa page

En général, une page web est constituée d'un en-tête (tout en haut), de menus de navigation (en haut ou sur les côtés), de différentes sections au centre... et d'un pied de page (tout en bas).

=> nouvelles balises HTML dédiées à la structuration du site, introduites par HTML5.



Exemple d'organisation de la page.

HTML: BALISES STRUCTURANTES

Les balises structurantes de HTML5

- **<header> : l'en-tête**

La plupart des sites web possède en général un en-tête. On y trouve le plus souvent un logo, une bannière, le slogan du site...

Vous devrez placer ces informations à l'intérieur de la balise

```
<header> :
Code : HTML
<header>
<!-- Placez ici le contenu de l'en-tête de votre page -->
</header>
```

L'en-tête peut contenir tout ce que vous voulez : images, liens, textes...

Il peut y avoir plusieurs en-têtes dans une page. Si celle-ci est découpée en plusieurs sections, chaque section peut en effet avoir son propre <header>.

HTML : BALISES STRUCTURANTES

- **<footer> : le pied de page**

A l'inverse de l'en-tête, le pied de page se trouve en général tout en bas du document. On y trouve des informations comme des liens de contact, le nom de l'auteur, les mentions légales, etc.

```
Code : HTML
<footer>
<!-- Placez ici le contenu du pied de page -->
</footer>
```

- **<nav> : principaux liens de navigation**

La balise <nav> doit regrouper tous les principaux liens de navigation du site.

Vous y placerez par exemple le menu principal du site.

Généralement, le menu est réalisé sous forme de liste à puces à l'intérieur de la balise <nav> :

```
Code : HTML
<nav>
<ul>
<li><a href="index.html">Accueil</a></li>
<li><a href="forum.html">Forum</a></li>
<li><a href="contact.html">Contact</a></li>
</ul>
</nav>
```

HTML : BALISES STRUCTURANTES

- **<section> : une section de page**

=> regroupement des contenus en fonction de leur thématique. Elle englobe généralement une portion du contenu au centre de la page.

```
Code : HTML
<section>
<h1>Ma section de page</h1>
<p>Bla bla bla</p>
</section>
```

- **<aside> : informations complémentaires**

La balise <aside> est conçue pour contenir des informations complémentaires au document que l'on visualise. Ces informations sont généralement placées sur le côté (bien que ça ne soit pas une obligation).

```
Code : HTML
<aside>
<!-- Placez ici des informations complémentaires -->
</aside>
```

Il peut y avoir plusieurs blocs <aside> dans la page

HTML : BALISES STRUCTURANTES

o <article> : un article indépendant

=> sert à englober une portion généralement autonome de la page. C'est une portion de la page qui pourrait (par exemple) être reprise ailleurs sur un autre site. C'est le cas par exemple des actualités (articles de journaux ou de blogs).

```
Code : HTML
<article>
<h1>Mon article</h1>
<p>Bla bla bla bla</p>
</article>
```

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>Site de l'ENSC</title>
</head>
<body>
  <header>
    <h1>ENSC</h1>
    <h2>Ecole Nationale Supérieure de Cognitique</h2>
  </header>
  <nav>
    <ul>
      <li><a href="#">Accueil</a></li>
      <li><a href="#">Blog</a></li>
      <li><a href="#">CV</a></li>
    </ul>
  </nav>
  <section>
    <h3>A propos de l'auteur</h3>
    <p>Bla bla bla bla.</p>
  </section>
  <aside>
    <h3>Mon bel article</h3>
    <p>Bla bla bla bla (texte de l'article)</p>
  </aside>
  <article>
    <h1>Mon bel article</h1>
    <p>Bla bla bla bla (texte de l'article)</p>
  </article>
</body>
</html>
```



CSS: MODÈLE DE BOÎTE

Une page web peut être vue comme une succession et un empilement de boîtes, qu'on appelle "blocs".

La plupart des éléments vus précédemment sont des blocs : <header>, <article>, <nav>... Mais aussi les paragraphes <p>, les titres <h1>...

=> manipuler ces blocs comme de véritables boîtes : leur donner des dimensions, les agencer en jouant sur leurs marges, gérer leur contenu... pour éviter que le texte ne dépasse ces blocs

o Les balises de type block et inline

En HTML, la plupart des balises peuvent se ranger dans 2 catégories :

Les balises inline : par exemple des liens <a>.

une balise de type "inline" se trouve obligatoirement à l'intérieur d'une balise "block". Une balise inline ne crée pas de retour à la ligne, le texte qui se trouve à l'intérieur s'écrit donc à la suite du texte précédent, sur la même ligne

Les balises block : par exemple des paragraphes <p></p>.

une balise de type "block" sur une page web crée automatiquement un retour à la ligne avant et après.

CSS: MODÈLE DE BOÎTE

o Les dimensions

o travailler uniquement sur des balises de type "block".

o Un bloc a des dimensions précises : largeur/hauteur.

=> 2 propriétés CSS :

- **width** : largeur du bloc en pixels (px) ou en pourcentage (%).
- **height** : hauteur du bloc en pixels (px), soit en pourcentage (%).

Par défaut, un bloc prend 100% de la largeur disponible.

p { width: 50%; /* paragraphes aient une largeur de 50% */ }

Les pourcentages seront utiles pour créer un design qui s'adapte automatiquement à la résolution du visiteur.

Pour créer des blocs ayant une dimension précise en pixels :

p { width: 250px; }

CSS: MODÈLE DE BOÎTE

o Minimum et maximum

On peut demander à ce qu'un bloc ait des dimensions minimales et maximales.

- **min-width** : largeur minimale
- **min-height** : hauteur minimale
- **max-width** : largeur maximale
- **max-height** : hauteur maximale

Code : CSS

p { width: 50%;

min-width: 400px; /* les paragraphes occupent 50% de la largeur et exigent qu'il y ait au moins 400 pixels de large dans tous les cas */ }

o

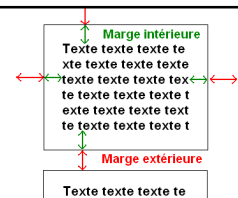
CSS: MODÈLE DE BOÎTE

o Les marges

Tous les blocs possèdent des marges.

Il existe 2 types de marges :

- Les marges intérieures
- Les marges extérieures



o En CSS, on peut modifier la taille des marges avec les 2 propriétés :

- **padding** : indique la taille de la marge intérieure en général en pixels (px).
- **margin** : indique la taille de la marge extérieure le plus souvent des pixels.

p { width: 350px; border: 1px solid black; text-align: justify; padding: 12px; /* Marge intérieure 12px */ margin: 50px; /* Marge extérieure de 50px */ }

CSS: MODÈLE DE BOITE

Les marges

Pour définir une marge en haut, en bas, à gauche et à droite, il va falloir utiliser des propriétés plus précises...
(même principe que pour la propriété **border**)

liste des propriétés pour **margin** et **padding** :

- **margin-top** : marge extérieure en haut.
- **margin-bottom** : marge extérieure en bas.
- **margin-left** : marge extérieure à gauche.
- **margin-right** : marge extérieure à droite
- **padding-top** : marge intérieure en haut.
- **padding-bottom** : marge intérieure en bas.
- **padding-left** : marge intérieure à gauche.
- **padding-right** : marge intérieure à droite.

o

CSS: MODÈLE DE BOITE

Centrer des blocs

Pour centrer, il faut respecter les règles suivantes :

- Donner une largeur au bloc (avec la propriété **width**)
- Indiquer que vous voulez des marges extérieures automatiques, à l'aide de : **margin: auto;**

```
p
{ width: 350px; /* On a indiqué une largeur (obligatoire) */
  margin: auto; /* On demande à ce que le bloc soit centré avec "auto" */
  border: 1px solid black;
  text-align: justify;
  padding: 12px;
  margin-bottom: 20px; }
```

Seul le centrage horizontal est permis.

CSS: POSITIONNEMENT

Le positionnement flottant

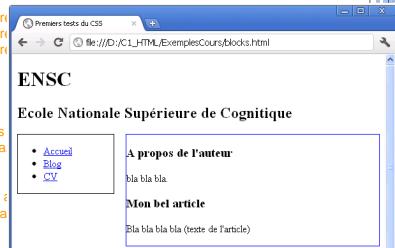
- La technique présentée ici l'est à titre d'information. Elle est utilisée par la majorité des sites à l'heure actuelle mais comporte un certain nombre de défauts.
- Une meilleure technique sera présentée un peu plus loin, le positionnement inline-block

Il se trouve que cette propriété est aujourd'hui utilisée par la majorité des sites web pour... faire la mise en page ! En effet, si on veut placer son menu à gauche et le contenu de sa page à droite, c'est a priori un bon moyen.

Attention : cette propriété n'a pas été conçue pour faire la mise en page à la base => elle présente quelques petits défauts

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8" />
  <title>Site de l'ENSC</title>
</head>
<body>
  <header>
    <h1>ENSC</h1>
    <h2>Ecole Nationale Supérieure de Cognitique</h2>
  </header>
  <nav>
    <ul>
      <li><a href="#">Accueil</a>
      <li><a href="#">Blog</a>
      <li><a href="#">CV</a>
    </ul>
  </nav>
  <section>
    <aside>
      <h3>A propos</h3>
      <p>bla bla bla</p>
    </aside>
    <article>
      <h3>Mon bel article</h3>
      <p>Bla bla bla</p>
    </article>
  </section>
</body>
</html>
```

```
nav { float: left; width: 150px; border: 1px solid black; }
section { margin-left: 170px; border: 1px solid blue; }
```



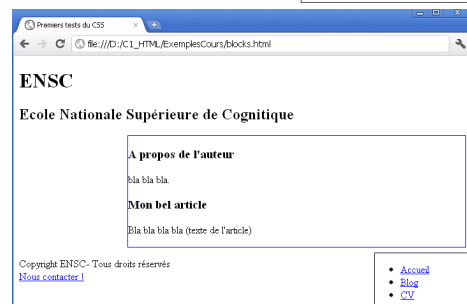
CSS: POSITIONNEMENT

Les positionnements absolu, fixe et relatif

- Le positionnement absolu** : il nous permet de placer un élément n'importe où sur la page (en haut à gauche, en bas à droite, tout au centre, etc..)
- Le positionnement fixe** : identique au positionnement absolu, mais cette fois l'élément reste toujours visible, même si on descend plus bas dans la page.
- Le positionnement relatif** : permet de décaler l'élément par rapport à sa position normale.
- Utilisation de la propriété CSS **position** à laquelle on donne une de ces valeurs :
 - **absolute** : positionnement absolu.
 - **fixed** : positionnement fixe.
 - **relative** : positionnement relatif.

CSS: POSITIONNEMENT

```
nav
{ border: 1px solid black;
  position: absolute;
  right: 0px;
  bottom: 0px; }
```



CSS: POSITIONNEMENT

Le positionnement inline-block

Transformation des éléments en inline-block avec la propriété **display**.

Les éléments de type inline-block :

- se positionnent les uns à côté des autres
- possèdent des dimensions précises

=> transformation en inline-block des éléments que l'on veut placer côte à côte : le menu de navigation et la section du centre de la page.

=> Utilisation d'une nouvelle propriété, normalement réservée aux tableaux : **vertical-align**. Cette propriété permet de modifier l'alignement vertical des éléments.

Quelques-unes des valeurs possibles pour cette propriété :

- **baseline** : aligne la base de l'élément avec celle de l'élément parent (par défaut)
- **top** : aligne en haut
- **middle** : aligne au milieu
- **bottom** : aligne en bas
- (valeur en px ou %) : aligne à une certaine distance de la ligne de base (baseline)

CSS: POSITIONNEMENT

Code : CSS -

```
nav { display: inline-block; width: 150px; border: 1px solid black; vertical-align: top; }
section { display: inline-block; border: 1px solid blue; vertical-align: top; }
```

ENSC

Ecole Nationale Supérieure de Cognitique

<ul style="list-style-type: none">• Accueil• Blog• CV	A propos de l'auteur bla bla bla Mon bel article Bla bla bla bla (texte de l'article) <small>Copyright ENSC - Tous droits réservés Nous contacter</small>
---	---

SOMMAIRE

- 1- Introduction à la programmation web
- 2- Les balises HTML de base
- 3- Mise en forme avec CSS
- 4- Mise en page
- 5- Les tableaux
- 6- Les formulaires HTML
- 7- Audio et vidéo

93

COMM - Outils de communication

LES TABLEAUX

<table> }

<caption> : titre du tableau (affiché en haut du tableau) à insérer après la balise **<table>**

<tr> : (Table Row) indique une nouvelle ligne dans le tableau

<th> : (Table Header) indique une nouvelle colonne de titre
Colspan="n" : permet de définir une ligne identique sur n colonnes

Rowspan="m" : permet de définir une colonne identique sur m lignes

<td> : (Table Data) indique une nouvelle cellule

94

COMM - Outils de communication

LES TABLEAUX

Exemple

```
Code HTML
...
<table>
<tr>
<td>Carmen</td>
<td>33 ans</td>
<td>Espagne</td>
</tr>
<tr>
<td>Michelle</td>
<td>26 ans</td>
<td>Etats-Unis</td>
</tr>
</table>
....
```

Rajouter des bordures :

```
Code : CSS
td /* Toutes les cellules des tableaux... */
{
border: 1px solid black; /* ... auront une bordure de 1px */
}
```

95

COMM - Outils de communication

LES TABLEAUX

propriété CSS spécifique aux tableaux: **border-collapse** pour coller les bordures entre elles.

Cette propriété peut prendre 2 valeurs :

collapse : les bordures seront collées entre elles.
separate : les bordures seront dissociées (valeur par défaut)

```
Code : CSS
table
{
border-collapse: collapse; /* Bordures du tableau collées */
}
td
{
border: 1px solid black;
}
```

Rajouter des bordures :

96

COMM - Outils de communication

```

<table border="1">
  <tr><th rowspan="2">STATUT</th>
    <th colspan="2">Identification</th>
    <tr>
      <th>NOM</th>
      <th>TEL</th>
    </tr>
  </tr>
  <tr>
    <th rowspan="3">Enseignant</th>
    <td>Thiessard Frantz</td>
    <td>4516</td>
  </tr>
  <tr>
    <td>Dibois Julie</td>
    <td>4855</td>
  </tr>
  <tr>
    <td>Case Matthieu</td>
    <td>4312</td>
  </tr>
  <tr>
    <th rowspan="2">IATOS</th>
    <td>Dupond Jean</td>
    <td>2132</td>
  </tr>
  <tr>
    <td>Durand Marcel</td>
    <td>1033</td>
  </tr>
</table>

```

```

table
{
border-collapse: collapse;
}
td, th /*bordure sur les td ET les th */
{
border: 1px solid black;
}

```

STATUT	Identification	
	NOM	TEL
Enseignant	Thiessard Frantz	4516
	Dibois Julie	4855
	Case Matthieu	4312
IATOS	Dupond Jean	2132
	Durand Marcel	1033

Fusion de cellules

EXERCICE APPLICATIF

- Ecrire le code HTML qui permet de créer le tableau suivant:

Exemple de tableau

<7>		
<6>	<3>	<4>
<2>	<1>	
<5>		

SOMMAIRE

- 1- Introduction à la programmation web
- 2- Les balises HTML de base
- 3- Mise en forme avec CSS
- 4- Mise en page
- 5- Les tableaux
- 6- Les formulaires HTML**
- 7- Audio et vidéo

5- LES FORMULAIRES HTML

- Circulation d'informations le plus souvent dans le sens serveur vers le client
mais nécessité d'envoyer de l'information depuis le client vers le serveur.
=> Utilisation des **formulaires**
- Formulaire** : Document HTML qui accepte des balises supplémentaires permettant de saisir de l'information et de la renvoyer au serveur.
- A chaque formulaire sera **associé un programme** sur un serveur qui sera chargé de collecter les données : programme CGI, ASP, PHP, ...
- Interactivité des pages Web

5- LES FORMULAIRES HTML -SYNTAXE

- Déclaration


```
<form action="url"> ... </form>
```

 - `url` est soit <mailto:adresse>, soit un script cgi ou php, ...
- Option `method` avec pour valeurs :
 - `GET` : les paramètres sont alors rajoutés à l'URL
=> déconseillé car taille des paramètres limitée
 - `POST` : envoi un message à part entière au serveur web
=> conseillé

Exemple :

```
<form method="POST" action="mon_premier_prog.php">
```

5- COMMANDES DES FORMULAIRES : ZONES DE SAISIE

- `<input ...>` pour insérer une **zone de saisie**

Attributs :

- `name` : attribue un nom logique au champ (nom du paramètre qui sera transmis au serveur)
- `value` : sert à spécifier la valeur par défaut d'un champ de saisie (valeur qui sera récupérée par le serveur)
- `size` : taille de la boîte de saisie
- `maxlength` : pour limiter la taille du texte entré

17

5- LES DIFFÉRENTES ZONES DE SAISIE

Attribut TYPE :

- text (par défaut)

Nom :

```
<form method="post" action="traitement.php">
<p>
<label for="nom">Votre nom</label> :
<input type="text" name="nom" id="nom" />
</p>
</form>
```

- hidden : champ n'apparaissant pas
- password : champ de saisie pour mots de passe Mot de passe :
- textarea : champ de saisie multiligne
- email : champ de saisie pour une adresse email
- url : champ de saisie pour une URL
- tel : champ de saisie pour un numéro de téléphone
- number : champ de saisie d'un nombre. Le champ s'affichera en général avec des flèches pour changer la valeur

Attributs :

- min : valeur minimale autorisée.
- max : valeur maximale autorisée.
- step : c'est le "pas" de déplacement. (1 en 1, 2 en 2,...)
- range : champ de sélection d'un nombre avec un curseur
- color : champ de saisie de couleur

COMM - Outils de communication

5- LES DIFFÉRENTES ZONES DE SAISIE

- Gestion des dates :
 - pour la date (29/01/2013 par exemple)
 - time : pour l'heure (13:37 par exemple)
 - week : pour la semaine
 - month : pour le mois
 - datetime : pour la date et l'heure (avec gestion du décalage horaire)
 - datetime-local pour la date et l'heure (sans gestion du décalage horaire)
- Search : champ de recherche

COMM - Outils de communication

104

5- LES DIFFÉRENTES ZONES DE SAISIE

- checkbox : bouton à choix multiple
 - Autre attribut checked : indique que la boîte à cocher est sur « on »

Exemple :

Antécédents :


```
<input type="checkbox" name="hta" id="hta" />
<label for="hta">Hypertension</label><br/>
<input type="checkbox" name="cardio" id="cardio" />
<label for="cardio">Maladies
cardiovasculaires</label><br/>
<input type="checkbox" name="cancer" id="cancer" />
<label for="cancer">Cancer</label><br/>
```

Antécédents :

☐ Hypertension
☐ Maladies cardiovasculaires
☐ Cancer

COMM - Outils de communication

5-LES DIFFÉRENTES ZONES DE SAISIE

- Attribut TYPE (suite)
 - RADIO : bouton à choix unique (checkbox exclusif)
 - Attribut CHECKED

Sexe : Sexe : ☐ Masculin ☒ Féminin

```
<input type="radio" name="sexe" id="masculin" />
<label>Masculin for="masculin" </label> <br/>
<input type="radio" name="sexe" id="feminin" checked />
<label for="feminin">Féminin </label> <br/>
```

- Boutons de validation ou d'annulation

- submit : bouton d'envoi lançant la transmission des données suivant l'attribut Action spécifié dans la balise FORM

```
<input type="submit" name="btn_envoi" id="btn_envoi"
value="Envoyer" />
```

Envoyer

- reset : bouton de réinitialisation des champs nulaire

```
<input type="reset" value="Effacer" />
```

Effacer

- image : équivalent du bouton "submit", présenté cette fois sous forme d'image. Rajouter l'attribut src pour indiquer l'URL de l'image.

- button : bouton générique, qui n'aura (par défaut) aucun effet. En général, ce bouton est géré en Javascript pour exécuter des actions sur la page.

5-COMMANDES DES FORMULAIRES : SÉLECTION

2. <select ...> pour une liste de sélection

```
<select name="nom-d'un-menu">
<option> Option 1</option>
<option>...</option>
</select>
```

• Exemple

```
<label for="paiement"> Paiement par :</label><br/>
<select name="paiement" id="paiement">
<option value="VP">Visa Premier</option>
<option value="VB">Visa Business</option>
<option>Mastercard</option>
<option value="AE">American Express</option>
</select>
```

Paiement par :

Visa Premier
Visa Business
Mastercard

5-LISTES DE SÉLECTION (SUITE)

- Possibilité de structurer la liste

```
<select>
<optgroup label="NomLabel1">
<option>option1</option>
<option>option2</option>
</optgroup>
<optgroup label="NomLabel2">
<option>option3</option>
<option>option4</option>
</optgroup>
</select>
```

COMM - Outils de communication

108

5-LISTES DE SÉLECTION (SUITE)

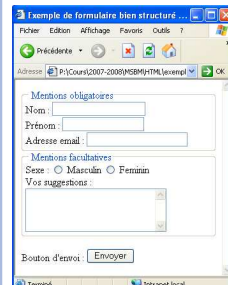
```
<form method="post" action="traitement.php">
<p>
<label for="pays">Dans quel pays habitez-vous ?</label><br />
<select name="pays" id="pays">
<optgroup label="Europe">
<option value="france">France</option>
<option value="espagne">Espagne</option>
<option value="italie">Italie</option>
<option value="royaume-uni">Royaume-Uni</option>
</optgroup>
<optgroup label="Amérique">
<option value="canada">Canada</option>
<option value="etats-unis">Etats-Unis</option>
</optgroup>
<optgroup label="Asie">
<option value="chine">Chine</option>
<option value="japon">Japon</option>
</optgroup>
</select>
</p>
</form>
```

COMM - Outils de communication



5-ORGANISATION DU FORMULAIRE

- **FIELDSET** : Faciliter les regroupements graphiques et la lisibilité



```
<form action="">
<fieldset><legend>Mentions obligatoires</legend>
<label for="nom">Nom :</label> <input type="text"
name="nom" id="nom" size="20" /><br/>
<label for="prenom">Prénom :</label> <input
type="text" name="prenom" id="prenom" size="17"
/><br/>
<label for="mail"> Adresse email : </label><input
type="email" name="mail" id="mail" size="22"/>
<br/></fieldset>
<fieldset><legend>Mentions
facultatives</legend>
Sexe :
<input type="radio" name="sexe" id="masculin" />
<label>Masculin for="masculin" </label> <br/>
<input type="radio" name="sexe" id="feminin" />
<label for="feminin">Féminin </label> <br/>
Vos suggestions :<br/> <textarea cols="25"
rows="4"></textarea></fieldset>
<p> Bouton d'envoi : <input type="submit"
value="Envoyer"/> </p>
</form>
```

Légende
du regroupement

5-COMMANDES DES FORMULAIRES : ZONE DE TEXTE

- Rendre un champ obligatoire

=>attribut required.

```
<input type="text" name="prenom" id="prenom" required />
```

Le navigateur indiquera alors au visiteur qu'il doit remplir le champ si celui-ci est vide au moment de l'envoi.

On dispose de pseudo-formats en CSS pour styliser les éléments requis (required) et invalides (invalid).

Exemple :

Code : CSS

```
:required
{
background-color: red;
}
```

Ajout d'un fond rouge aux éléments obligatoires

Sur le même principe, le pseudo-format focus est disponible pour changer l'apparence d'un champ lorsque le curseur se trouve à l'intérieur.

111

COMM - Outils de communication

SOMMAIRE

- 1- Introduction à la programmation web
- 2- Les balises HTML de base
- 3- Mise en forme avec CSS
- 4- Mise en page
- 5- Les tableaux
- 6- Les formulaires HTML
- 7- Audio et vidéo

112

COMM - Outils de communication

VIDÉO ET AUDIO

- Insertion d'un élément audio

=> balise <audio>

```
<audio src="hype_home.mp3" controls></audio>
```

On peut compléter la balise des attributs suivants :

controls : pour ajouter les boutons "Lecture", "Pause" et la barre de défilement.

width : pour modifier la largeur de l'outil de lecture audio.

loop : la musique sera jouée en boucle.

autoplay : la musique sera jouée dès le chargement de la page.

preload : indique si la musique peut être préchargée dès le chargement de la page ou non.

Elle peut prendre les valeurs:

auto (par défaut) : le navigateur décide s'il doit précharger toute la musique, uniquement les métadonnées ou rien du tout.

metadata : charge uniquement les métadonnées (durée, etc.).

none : pas de préchargement. Utile si vous ne voulez pas gaspiller de bande passante sur votre site.

113

COMM - Outils de communication

VIDÉO ET AUDIO

- Insertion d'une vidéo

=> balise <video>

=> attributs (la plupart sont les mêmes que pour la balise audio) :

poster : image à afficher à la place de la vidéo lorsque celle-ci n'est pas lancée. Par défaut, le navigateur prendra la première image de la vidéo

controls : pour ajouter les boutons "Lecture", "Pause" et la barre de défilement.

width : pour modifier la largeur de la vidéo.

height : pour modifier la hauteur de la vidéo.

loop : la vidéo sera jouée en boucle.

autoplay : la vidéo sera jouée dès le chargement de la page.

preload : indique si la vidéo peut être préchargée dès le chargement de la page ou non. Elle peut prendre les valeurs :

auto (par défaut) : le navigateur décide s'il doit précharger toute la vidéo, uniquement les métadonnées ou rien

metadata : charge uniquement les métadonnées (durée, dimensions, etc.).

none : pas de préchargement. Utile pour ne pas gaspiller de bande passante sur votre site.

```
<video src="sintel.webm" controls poster="sintel.jpg" width="600"></video>
```

114

COMM - Outils de communication

ERGONOMIE D'UN SITE WEB

1. Définir l'organisation des pages, des dossiers

2. Page d'accueil :

- C'est la 1^{ère} page de votre site que l'utilisateur verra

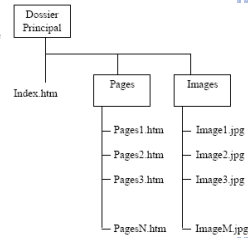
- Le nom à donner à votre page d'accueil sera imposé par votre fournisseur d'accès

Le plus souvent, ce sera « index.htm » ou « index.html »

- C'est la seule page dont le fournisseur d'accès a besoin de connaître le nom.

Les autres pages ainsi que les images sont organisées selon votre souhait

COMM - Outils de communication



115

MISE EN PAGE D'UN SITE

Les éléments importants :

1. La page d'accueil

très importante : elle décidera l'utilisateur à visiter les pages du site.
→ Doit respecter les principes de **clarté**, **précision** et **simplicité**.

2. Le menu

=> intuitif. Vous pouvez utiliser des icônes, des couleurs, du texte et si possible mélanger les trois.

3. Les couleurs

4. Liens entre les pages

La navigation intuitive permet à l'internaute de s'appropriier plus facilement le site Web. Il faut permettre de passer d'une page à une autre, revenir sur la page d'accueil, sur une page en particulier ou à la page précédente sans utiliser les boutons « précédent » et « suivant » du navigateur

→ Les liens doivent être comme des index, permettant à l'utilisateur d'un site Web de trouver très vite les informations cherchées

5. Les contenus (textes/images/animations)

La plupart du temps, il faut envisager le site Web comme allant à l'opposé d'un livre et donc l'imaginer comme un support où l'on peut faire passer de l'information sous toutes ses formes

→ Parfois, il sera bien plus efficace d'insérer une vidéo ou une simple image pour faire passer une information plutôt qu'un texte explicatif

COMM - Outils de communication

116