

Chapitre :

Pages Internet



I. Observation de pages

Nous utiliserons le logiciel firefox pour observer et comparer quelques pages Internet.

Exercice 1

Sur chacune des pages suivantes :

- <http://www.ac-nancy-metz.fr/>
- [http://fr.wikipedia.org/wiki/Wikipédia:Accueil_principal](http://fr.wikipedia.org/wiki/Wikip%C3%A9dia:Accueil_principal)
(plus simplement : <http://fr.wikipedia.org/>)
- <http://www.google.fr/>
- <http://www.les-mathematiques.net/>
- <http://patrick.thevenon.free.fr/>

utiliser les outils suivants (disponibles en cliquant droit dans la page) :

1. Code source de la page
2. Examiner l'élément

afin de :

- distinguer ce qui est commun à toutes les pages
- voir les éléments qui peuvent varier

On pourra s'aider des couleurs différentes du code source, utiliser le bouton « Examiner » en se déplaçant dans la page et cliquer droit sur les éléments s'affichant en bas de l'outil d'examen.

II. Le HTML

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

L'**Hypertext Markup Language**, abrégé en **HTML**, est un langage utilisé pour écrire des pages Internet. Il utilise des balises (Markups) et permet de faire de l'hypertexte (le fait de pouvoir lier des mots d'un texte à une nouvelle page), d'où son nom.

Le contenu d'une page HTML peut être très varié :

- Du simple texte
- Des images (fixes ou animées)
- De la vidéo
- Des formulaires (permettant d'écrire un mot, de cocher des cases, etc.)
- Et surtout des liens vers d'autres pages

Le type de contenu qui peut être intégré dans une page a évolué avec le langage HTML, qui en est actuellement à la version 5 (la première à permettre entre autres d'inclure directement des vidéos, sans passer par flash). Il existe également d'autres langages, comme le XHTML (le X étant pour *Extensible*) qui lui est très similaire.

2. Structure d'une page HTML

Le code source d'une page HTML a en principe au minimum une structure telle que la suivante :

```

<!DOCTYPE html PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html>
  <head>
    <title>
      Exemple de HTML
    </title>
  </head>
  <body>
    Ceci est une phrase avec un <a href="cible.html">hyperlien</a>.
    <p>
      Ceci est un paragraphe où il n'y a pas d'hyperlien.
    </p>
  </body>
</html>

```

La toute première ligne indique au navigateur la version du langage utilisée (ici HTML 2.0).

Ensuite commence le code proprement dit de la page. Tous les mots entre < et > sont les balises. Pour chaque balise qui ouvre une structure, il y a une balise qui la referme. Cette dernière porte le même nom précédé d'un slash :

<head> ... </head>, <body> ... </body>, <a> ... , etc.

On remarque que pour certaines balises comme <a>, il est possible d'ajouter des éléments (). Certaines balises ne nécessitent pas d'être fermées par une autre balise ; on écrit alors un slash à la fin de la balise, comme par exemple <link ... /> ou que l'on verra plus bas.

On observe généralement une indentation du texte qui permet une lecture claire de la structure.

En particulier, tout le code est entre <html> et </html>, et on observe deux parties principales à l'intérieur : l'entête (<head>) et le corps (<body>).

a. Entête

L'entête, délimitée par les balises <head> et </head>, contient les informations générales sur la page. C'est ici que l'on donne le titre de la page avec <title>. On peut ajouter d'autres éléments comme :

```

<meta http-equiv="Content-type" content="text/html; charset=utf-8"/>
<link rel="icon" href="Docs/icône.gif"/>
<link rel="stylesheet" type="text/css" href="style.css"/>
<script type="text/javascript" src="java.scripts" ></script>

```

<meta> permet d'indiquer plusieurs choses comme l'auteur, le type de codage des caractères, etc.

<link> permet de donner des adresses de fichiers qui peuvent être l'icône de la page, contenir le style de la page (que l'on verra plus tard), etc.

<script> permet de donner l'adresse d'un fichier (ou directement le texte), contenant un code (ou script) de nature différente du html qui peut être utilisé dans la page, en particulier du code javascript.

Il en existe d'autres.

b. Corps

Le corps, délimité par les balises <body> et </body>, est la partie qui décrit le contenu de la page. C'est donc en principe la partie la plus importante dans le code.

Ici les balises indiquent la structure de la page.

```

<div id="page">
  <div id="gauche">
    <div id="photo">
      <a href="Docs/Preuve.png" title="Agrandir">
        </a>
      </div>
    <div id="menu">
      <h2 class="c1">Enseignement</h2>
      <ul>
        <li><a class="c1" href="Enseignement/2011">2011-2012</a></li>
        <li><a class="c1" href="Enseignement/2012">2012-2013</a></li>
      </ul>
      <h2 class="c3">Bonus</h2>
      <ul>
        <li><a class="c3" href="Bonus/Liens">Liens</a></li>
      </ul>
    </div>
  </div>
  <div id="droite">
    <h2>Accueil</h2>
    <h3>Présentation</h3>
    <ul>
      <li>Établissement actuel :
        <dl>
          <dd>Lycée Alfred Kastler</dd>
          <dd>1 rue de Münnerstadt</dd>
          <dd>55 700 Stenay</dd>
        </dl>
      </li>
    </ul>
  </div>
</div>

```

Il existe beaucoup de balises. Principalement :

<h1>, <h2>, ... permettent de mettre des titres hiérarchisés (<h1> étant le plus grand, <h6> le plus petit).

<div> permet de créer une sorte de boîte qui pourra permettre de structurer l'apparence.

 et permettent de faire des listes. Il existe différentes sortes de listes et donc de balises.

Et d'autres encore (tableaux, formulaires, vidéos...)!

On remarque que l'on peut donner des noms (id) aux <div>, utiliser des classes (class), nous verrons l'utilité de cela plus tard.

Il faut tout de même citer deux autres balises parmi les plus communes. Le code permettant de faire un lien vers une autre page est le suivant :

```
<a href="adresse de la page">Texte sur lequel on clique</a>
```

On ajoute une image ainsi :

```

```

Et l'on peut tout à fait rendre une image « cliquable » en insérant une image plutôt qu'en écrivant du texte entre les balises <a> et .

3. Construction d'une page

Le code d'une page Internet, écrit en HTML, s'écrit à l'aide d'un simple éditeur de texte (plus simple qu'un logiciel de traitement de texte). Le fichier créé a en général pour extension .html (ou .htm).

Il est toujours possible d'enregistrer une page Internet sur son ordinateur à partir du navigateur (avec « Enregistrer sous... »). On peut alors en général voir le contenu (presque complet) de la page enregistrée, même sans être connecté. On peut s'aider de la documentation trouvée sur de nombreux sites pour créer une page Internet et connaître entre autres toutes les balises. Il y a beaucoup de sites en anglais, en particulier les sites officiels des langages (HTML, XHTML), comme par exemple <http://www.w3.org/TR/html4/> pour le HTML4. Le W3C (World Wide Web Consortium) est une communauté de personnes qui travaillent ensemble au développement de standards du Web, c'est lui qui définit les standards des langages pour les pages Internet.

On trouve évidemment des pages en français, comme <http://www.aliasdmc.fr/courshtml/>

On peut toujours faire des recherches sur Internet et trouver d'autres sites qui traitent des points précis ou expliquent autrement certaines choses.

Exercice 2

Créer un fichier HTML avec une extension .html qui pourra être ouverte par le navigateur.

Pour que la page soit accessible sur Internet, il faut qu'elle soit disponible sur un réseau, c'est à dire que le fichier .html soit copié sur un serveur qui communique avec l'extérieur.

Nous allons utiliser un compte créé spécialement pour cela. La procédure qui permet de copier notre fichier sur le serveur utilise une extension de firefox : fireftp.

On se connecte au serveur avec le nom d'utilisateur et le mot de passe, et on copie simplement les fichiers nécessaires à l'affichage de la page Internet créée (par exemple il ne faut pas oublier les images s'il y en a besoin).

Dans le cas où l'adresse de notre site est "http://www.mapage.com" et que l'on place un fichier "test.html" directement à la racine du serveur, on accède à la page à l'adresse "http://www.mapage.com/test.html".

Il existe un cas particulier de nom de page : "index.html". Dans ce cas, il est inutile de donner son nom complet ; le navigateur sait que "http://www.mapage.com/" correspond à "http://www.mapage.com/index.html".

Autrement dit, une page dont l'adresse ne termine pas par .html, mais simplement par / (slash), est le plus souvent une page index.html (ou index.php mais nous en reparlerons plus tard).

Exercice 3

Copier les fichiers .html créés sur le serveur, et aller sur la page Internet.

III. Apparence (CSS)

1. Généralités

Si l'on reste sur le type de code HTML indiqué jusqu'ici dans ce cours, on obtient des pages très simples et peu élégantes. Il est possible d'avoir la main sur la présentation de chaque élément affiché sur la page, autrement dit de donner un style à la page. Et le style fait beaucoup !

Exemple Aller sur une page Internet et choisir « Affichage > Style de la page > Aucun style ».

Le style se définit grâce au CSS (*Cascading Style Sheets* ou « Feuilles de style en cascade »). C'est aussi un standard défini par le W3C qui date des années 1990, et qui a été pris en charge par les navigateurs vers les années 2000. Il existe là aussi différentes versions, la dernière en construction étant la 4.

Le but de la spécialité ISN n'étant pas de traiter ce problème, nous n'entrerons pas dans le détail. Nous nous contenterons éventuellement de prendre des fichiers existant et de les adapter, sans chercher à compliquer la page.

2. Écriture et test de divers fichiers css

Presque toutes les pages web actuelles utilisent un fichier css qu'il est possible de récupérer. Pour cela il suffit de voir le code source de la page, dans l'entête :

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

Il suffit d'entrer l'adresse du fichier (en ajoutant en général l'adresse de base du site), et l'on peut lire le contenu du fichier, donc le copier.

Globalement on peut retenir qu'il est possible de définir des attributs (taille, couleur, etc) à tout élément (titre, texte simple).

On peut définir des styles différents selon l'identité (id) ou la classe (class) d'un élément (voir l'exemple de corps de page html donné plus haut). Sur une page il ne peut y avoir qu'un élément ayant une identité donnée, alors que les classes peuvent être partagées par plusieurs éléments.

Ce qui concerne les objets ayant l'identité page est introduit dans le fichier css par un #page :

```
#page{
margin:1% 1%;
color:#000;
background-color:#ffffbe;
border:3px solid #000;
height:700px;
}
```

Ce qui concerne les objets ayant la classe c1 est introduit par un .c1 :

```
#menu .c1{
color:#655f4b;
background-color:transparent;
}

#menu a.c1{
color:#b00606;
display:block;
font-weight:bold;
margin-left:-3em;
padding-left:4px;
text-decoration:none;
}
```

Un élément ayant cette classe c1, utilisé dans le bloc ayant pour identité menu aura une certaine couleur (#655f4b). Le cas particulier d'être un lien (<a>) modifie la couleur (#b00606), mais le fond (background-color) restera le même (c'est le phénomène de cascade : la définition la plus générale reste la même pour les cas particuliers tant qu'elle n'est pas modifiée).

Des éléments d'explications (en français) se trouvent ici : <http://www.zonecss.fr/courscss/>

Pour les couleurs (par exemple) : http://www.createafreewebsite.net/color_chart.html

Exercice 4 Utiliser des fichiers css trouvés sur Internet (par exemple dans des tutoriels), n'en garder que le strict nécessaire à la page créée et modifier pour observer les différences. Ne pas oublier d'ajouter dans l'entête du fichier html l'adresse du fichier css.

IV. Pour aller plus loin

1. Certification

Les langages HTML, XHTML, CSS étant définis précisément, ils doivent respecter une syntaxe précise. Les navigateurs ne sont pas toujours très difficiles et affichent des pages même mal écrites.

Il existe des outils sur Internet pour vérifier que le code que l'on a créé est parfaitement bien formé.

pour le html et le xhtml : <http://validator.w3.org/>

Pour le css : <http://jigsaw.w3.org/css-validator/>

Cela donne une garantie que des erreurs d'affichage ne se feront pas à cause d'un mauvais code.

Exercice 5 Tester des pages web, vérifier que l'on en trouve qui sont mal écrites (ou en tout cas ne respectent pas le standard), même parmi les plus célèbres (google, wikipedia).

Testez votre page!

Malheureusement, tous les navigateurs ne réagissent pas de la même manière à un code donné. Il existe des outils permettant de savoir comment l'ensemble des navigateurs réagissent sans avoir à les installer.

Par exemple : <https://browserling.com/>

Exercice 6 Faire afficher quelques pages dans ce site en choisissant différents navigateurs.

2. Pages dynamiques

En dehors de la vidéo et des applications en flash qui donnent une impression de dynamisme à la page, il y a des outils très puissants qui permettent d'améliorer la qualité de navigation sur Internet.

En fait, la plupart des pages sont ce que l'on appelle « dynamiques » : elles s'adaptent en fonction de paramètres (nom ou pays de l'utilisateur, mots clés introduits dans des formulaires, etc.) Ce qui fait que les pages ne sont pas écrites directement en html, qui ne permet pas ce dynamisme : un fichier html est figé et affiche toujours la même chose.

Enregistrer la page google qui s'affiche sur votre ordinateur ne vous donnera pas le code d'un moteur de recherche personnel!

a. PHP

Un langage fréquemment utilisé pour faire des pages dynamiques est le PHP (PHP : Hypertext Preprocessor, c'est un acronyme récursif).

C'est en fait un langage (algorithmique) qui est interprété par le serveur de la page web. Cela signifie que le serveur exécute l'algorithme, la sortie de l'algorithme étant un code html qui est envoyé au navigateur de la personne qui regarde le site.

Le code du fichier php n'est jamais visible par l'utilisateur. Seul l'administrateur du site peut le voir et le modifier.

Sans php, il serait par exemple très laborieux (mais pas infaisable?) de faire un site de vente par Internet. Il faudrait créer une page par produit vendu (si raisonnablement on ne souhaite en afficher qu'un), alors qu'il suffit d'une seule page php bien faite pour parcourir l'ensemble des produits (on ajoute un moyen de chercher et d'afficher un produit donné). C'est un gain de temps et de mémoire. La gestion du nom du client est raisonnablement impossible en html seul.

PHP n'est pas utilisé seul, il est en général associé aussi à des bases de données qui contiennent les listes de produits et de clients avec leurs mots de passe, etc.

b. Javascript

D'autres éléments permettent une dynamique à la page affichée. Il s'agit par exemple de javascript. Cela permet entre autres de modifier l'apparence ou même des éléments de la page en fonction d'événements (clic, heure). Un des effets d'un script javascript peut être par exemple de charger une autre page Internet au bout d'un certain temps.

Afficher les résultats d'une recherche avant même d'avoir fini de taper tous les mots clés est une autre possibilité.

Le monde des pages Internet s'est considérablement enrichi depuis ses débuts!