

Initiation à html et à la création d'un site web

Introduction :

Concevoir un site web consiste à définir :

- l'emplacement où ce site sera hébergé
- à qui ce site s'adresse
- le design des pages qui le constituent (organisation et présentation des informations)
- le contenu de ces pages qui sera en général écrit à l'aide du langage html

Nous nous plaçons dans le contexte du site de vos pages personnelles qui seront hébergées sous votre compte unix (accessibles par un serveur de l'enst) et qui seront lues par tout visiteur auquel vous souhaitez présenter des informations personnelles ou autres.

Le but de ce TP est de vous montrer comment concevoir les pages qui le constituent en faisant un travail tel que la mise à jour et les évolutions du site se feront à un moindre coût.

Structure globale d'une page html :

Elle est composée à partir de balises qui peuvent être définies avec des attributs précisant les conditions d'utilisation de la balise. Elles sont ouvertes puis fermées et définissent comment le navigateur interprète le texte inclus dans la balise (`<balise> texte </balise>`).

Toute page html doit être structurée par les balises suivantes :

```
<html>
  <head>
    <title>titre de la page</title>           (entete de la page)
  </head>
  <body>
    bla bla bla                               (corps de la page)
  </body>
</html>
```

Le corps de la page est composé du texte qu'on souhaite afficher, des images, des liens, ... et dans lequel on trouvera d'autres balises, avec d'autres attributs ...

Nous allons définir une page html qui contient votre cursus.

Commencez par définir une première page html qui ne contiendra que les balises décrites ci-dessus.

Vous enregistrerez cette page dans le répertoire `public_html`, sous le nom `cursus.html`.

Vérifiez qu'elle s'affiche bien.

Définissez les paramètres externes de la page (dans l'entête):

```
<meta name="description" content="page perso">
<meta name="auteur" content="moi">
<meta name="keywords" content="enst paris">
```

Les balises `meta` n'apparaissent pas à l'affichage de la page, mais sont utilisées, par exemple, par les moteurs de recherche pour savoir comment référencer votre page.

Elles peuvent aussi servir à effectuer quelques calculs que nous n'aborderons pas ici.

Définissez la structure générale de présentation de la page

Pour cela, vous définirez dans le corps de la page (**body**) un tableau comme suit :

Zone 1 : mon logo	Zone 2 : mon cursus
Zone 3 : Le plan du site	Zone 4 : 2004-2005 : classe prépa au lycée...
Zone 5 : écrivez-moi	

Les balises de définition d'un tel tableau sont les suivantes :

```
<table width="100%" border="1">
<tr>
    <td>mon logo</td><td>mon cursus</td>    (première ligne avec 2 colonnes)
</tr>
<tr>
    <td>le plan...</td><td>2004-2005 ...</td> (deuxième ligne avec 2 colonnes)
</tr>
<tr>
    <td colspan="2">écrivez-moi</td>          (troisième ligne avec 2 colonnes)
</tr>
</table>
```

De manière générale, on définit un tableau par la balise `<table></table>`

Une ligne se définit par la balise `<tr></tr>`

Une colonne (à l'intérieur d'une ligne) est définie avec `<td></td>` ou avec `<th></th>` s'il s'agit des titres d'une colonne.

L'attribut `colspan` signifie que la cellule prendra la place de deux cellules par rapport au découpage complet du tableau.

Définir le style d'écriture dans ce tableau (dans l'entête) :

On définira une feuille de style à laquelle certains éléments de la page se référeront

On fera référence à la feuille de style en incluant la ligne suivante dans l'entête de la page html.:

```
<link    title="ma feuille de style perso"  type="text/css"
        rel="stylesheet" href="styles/styleperso.css" />
```

Pour faire la feuille de style, définissez dans le répertoire `public_html`, un sous-répertoire de nom `styles` (si vous ne l'avez pas déjà fait au premier tp).

Vous définirez ensuite un fichier de nom `styleperso.css` que vous enregistrerez dans le répertoire `styles`.

Ce fichier constitue la feuille de style. Elle va contenir, dans une syntaxe précise, des définitions de présentation de texte ou d'éléments `html`.

Elle sera simple au départ. Vous pourrez l'enrichir ultérieurement en fonction de vos besoins en ajoutant d'autres styles.

Voici un premier style à inclure (ne pas oublier le point du début):

```
.pied_page
{
font-family : Arial, Helvetica, sans-serif;
font-style: italic;
text-align: center;
font-size : 9pt;
color: #000000;
padding-top: 6px;
padding-bottom: 6px;
}
```

L'utilisation d'un style se fait ensuite simplement en ajoutant l'attribut `class` à une balise.

Par exemple :

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

Tout ce qui se trouvera dans la balise `<p></p>` (qui définit un paragraphe) suivra ensuite la description qui est faite dans la feuille de style.

- Rques :**
1. Quand on définit un nouveau style, l'identifiant du style doit être précédé d'un point dans la page de style. Seul le nom du style apparaît quand on l'utilise (ici `pied_page` et `.pied_page`).
 2. Vous pouvez réutiliser la feuille de style du premier tp (`styles.css`) et la modifier en y ajoutant le style `.pied_page` . Vous remarquerez dans cette dernière feuille que, dans la mesure où les styles qui y sont définis correspondent à des balises `html` reconnues, le nom du style n'est pas précédé d'un point.

Remplir plus précisément les cases du tableau en définissant le rôle de chaque cellule

Dans un premier temps, vous préciserez la page de votre cursus, ainsi que des renvois vers les autres pages que vous prévoyez de faire figurer dans votre site.

Vous définirez :

- **zone 1** : votre logo
- **zone 2** : le titre de la page de cursus
- **zone 3** : le plan général du site avec les renvois vers les autres pages (hobbies, sites)
- **zone 5** : un bas de page avec un lien pour vous écrire
- **zone 4** : le contenu de la page à proprement parler : votre cursus

Les contenus des cases devront utiliser les styles de la feuille de style

Pour les différentes zones on utilisera

- **zone 1** : votre logo : pour l'instant, il contiendra uniquement votre photo (ou une autre) à l'aide d'une balise d'image :

```

```

- rqes :** - Toutes les images de votre site devront se placer dans un répertoire `images` prévu à cet effet.
- L'attribut `title` sert à afficher un commentaire en même temps que l'image
 - L'attribut `alt` sert aux mal voyants qui ne peuvent afficher les images et liront ce texte
 - Si vous n'avez pas de photo, vous pouvez utiliser l'image utilisée au premier TP:
- ```

```

On peut mettre un lien derrière l'image :

```


```

- **zone 2** : titre : une balise de titre

```
<h2 class="titre-principal"> cursus de . . . </h2>
```

**Rques :** Les balises `<h1></h1>` définissent des titres html de niveau 1  
Ceci suppose que vous avez ajouté le style `titre-principal` dans votre feuille de style

- **zone 3** : plan : une liste de liens

```

 <li class="item-menu"> mon cursus
 <li class="item-menu"> mes hobbies
 <li class="item-menu"> mes sites
préférés

```

**Rques :** Les balises `<ul></ul>` définissent une liste, `<li></li>` un élément de liste  
Ceci suppose que vous avez ajouté le style `item-menu` dans votre feuille de style

- **zone 4** : contenu de la page à proprement parler : votre cursus. Utilisez différentes balises et les styles de la feuille. Utilisez par exemple une liste (balises `<ul>` et `<li>` comme pour le menu). Vous pouvez aussi utiliser les balises `<p></p>` de changement de paragraphe, `<br />` de saut de ligne, `<h2></h2>` de titre ...
- **zone 5** : bas de page : un lien (renvoi vers un index), un envoi de message (`mailto` l'adresse du propriétaire)

```
<p class="pied_page"> copyright @ Moi mécéme
 Eécrivez-moi</p>
```

Vous avez ainsi défini la page `cursus.html`.

Vérifier qu'elle s'affiche bien comme vous le souhaitez, faites la mise au point.

**Remarque :** vous remarquerez que les caractères accentués doivent être libellés avec une syntaxe particulière (et simple) qui permet au navigateur de les afficher correctement.

## Définir un patron (gabarit) de page à partir de celle que vous venez de composer

Nous décidons maintenant que les zones 1, 3 et 5 de `cursus.html` ont des contenus qui seront toujours les mêmes pour toutes les pages de votre site. Pour utiliser cette spécificité, de votre site, nous allons définir le patron de vos pages. Ce sera un fichier modèle à partir duquel toutes les pages du site seront faites.

Pour définir le patron de vos pages, nous allons donc nous appuyer sur le fichier `cursus.html` que nous venons d'écrire:

Définir un fichier de nom `modele.php` qui contiendra au début la même chose que votre fichier `cursus.html` (simple recopie de `cursus.html` dans `modele.php` :

```
cp -p cursus.html modele.php
```

Vérifiez que cette page s'affiche bien.

Vous définirez ensuite 3 fichiers de la manière suivante :

- `logo.html` : un fichier contenant ce que vous avez mis en **zone 1**
- `pied-de-page.html` : un fichier contenant ce que vous avez mis dans la **zone 5**
- `menu.html` : un fichier contenant la liste de choix de la **zone 3**. Son contenu sera changé ultérieurement, nous vous expliquerons comment.

Bien sûr, veillez à donner les bons droits (644) à tous ces fichiers.

**Modifier la page `modele.php` pour qu'elle se contente d'inclure les fichiers qu'on vient de définir.**

Une inclusion se fait par exemple avec une ligne du type :

```
<? include("fichier.html"); ?>
```

qui contente d'inclure le contenu du fichier `fichier.html` là où elle est appelée.

**Rques :** Cette ligne contient un court script **php**.

**Php** est un langage de scripts (programmes) assimilable à un langage de développement (comme **C** par exemple) dont nous n'utilisons ici qu'une fonction d'inclusion de fichier.

D'où le fichier `modele.php`:

```
<table width="100%" border="1">
<tr>
 <td width="20%">
 <?php include("logo.html"); ?>
 </td>
 <td>
 <!-- emplacement du titre de la page -->
 Titre de la page
 <!-- fin du titre de la page -->
 </td>
</tr>
<tr>
 <td>
 <?php include("menu.html"); ?>
 </td>
 <td>
 <!-- emplacement du corps de la page -->
 Corps de la page
 <!-- fin du corps de la page -->
 </td>
</tr>
<tr>
 <td colspan="2">
 <?php include("pied-de-page.html"); ?>
 </td>
</tr>
```

Vous pouvez vérifier que cette page s'affiche correctement.

Vous venez de définir le patron (on dit aussi le gabarit) de votre site.

### Utiliser enfin ce modèle pour re-cr  er la page de votre cursus

Recopier ce fichier `modele.php` dans un nouveau fichier `cursus.php`.

Pour retrouver votre page initiale de cursus, il suffit d'aller modifier son titre et son contenu avec ceux que vous aviez d  finis pr  c  demment.

Vous modifierez   galement le fichier `menu.html` pour qu'il ne contienne plus qu'un seul renvoi vers le fichier `cursus.php` cette fois-ci, car c'est la seule page existant actuellement sur ce site.

**Remarque :** Les pages de votre site qui utilisent le patron ont toujours l'extension `.php`. C'est une n  cessit   pour que la fonction d'inclusion soit comprise et ex  cut  e par le serveur.

**Vous pouvez v  rifier que la modification des styles ou de tout fichier inclus (`logo.html` ou  `pied-de-page.html`) se r  percute sur la page `cursus.php`**

**Vous pouvez maintenant enrichir votre site avec d'autres pages**

D  finissez la page d'accueil de votre site. Le processus est le suivant :

- copier le fichier `modele.php` dans le fichier `index.php`
- modifier la zone de logo (`logo.html`) pour que le lien derri  re l'image pointe vers `index.php` (**zone 1**)
- modifier la zone de titre de `index.php` (**zone 2**) (ma page perso)
- modifier le contenu de `index.php` en y pla  ant des balises d'images (**zone 4**). Par exemple, le texte pourrait   tre : "Voici mon site personnel..."

L'acc  s internet    votre site se fera avec l'adresse :

<http://perso.enst.fr/~login/index.php>

Pour la page de vos hobbies, le processus est le suivant:

- copier le fichier `modele.php` dans le fichier `hobbies.php`
- modifier la zone de titre de `hobbies.php` (**zone 2**)
- modifier le contenu de `hobbies.php` en y pla  ant vos centres d'int  r  t (**zone 4**) (vous pouvez reprendre vos centres d'int  r  t du premier TP).
- modifier le fichier `menu.php` pour qu'il contienne un lien vers `hobbies.php`

Vous pouvez v  rifier que `hobbies.php` et `cursus.php` fonctionnent bien

Vous pouvez recommencer ce travail avec `sites.php` (vous pouvez reprendre les sites pr  f  r  s du premier TP)

Vous pouvez v  rifier    chaque fois que toute modification sur les fichiers inclus (le menu surtout) est effective dans le site tout entier.