

TP1 Introduction au langage à balises HTML

Exercice 1

- 1) Recopier le code ci-dessous dans un éditeur de texte (sublime text ou autre). Pensez à ajouter les balises obligatoire pour faire fonctionner votre code.

```
1 <div> <!-- Balise ouvrante -->
2 <h1>Balise de titre de niveau 1</h1>
3 <h2>Balise de titre de niveau 2</h2>
4 <h5>Saut de ligne automatique après une balise titre</h5>
5 <h6>Jusqu'à 6 niveaux de titre</h6>
6 <p>
7   Paragraphe de contenu. <strong>Texte en gras.</strong> <em>Texte en italique.</em>
8 </p>
9 <p>
10  Il y a également un saut de ligne après un paragraphe.
11  <br> <!-- Balise orpheline -->
12  La balise <strong>br</strong> permet un retour à la ligne d'un texte.
13  <br>Il n' y a pas d'espaces verticaux entre deux lignes séparées par une balise <strong>br</strong>
14 </p>
15 </div> <!-- Balise fermante -->
16
17 <!-- Balise de commentaire -->
```

- 2) Enregistrer le fichier sous exercice1.html
- 3) Puis, ouvrez le avec un navigateur web et visualiser le résultat

Exercice 2

- Créer vos fichiers HTML.
Rappel : une page en *HTML* se compose d'une structure de base :

```
<!DOCTYPE html>
<html>
<head>
  <meta charset="utf-8">
  <title>Titre du site</title>
  <!-- commentaire : description de la page-->
  <!--.....à compléter (liens (fichiers CSS,javascript), balises META ....-->
</head>
<body>
  <!-- commentaire : contenu de la page-->
  .....à compléter
</body>
</html>
```

- Utiliser **Sublime Text** ou un autre éditeur pour créer une page **exercice2.html** (attention aux extensions, seules les extensions **.html .htm .php** sont prises en compte par le serveur Web apache)
- Utiliser les snippets de *sublime Text* pour créer la structure *HTML* de base. Taper **html** puis [tabulation].
- Rajouter si ce n'est pas fait l'encodage **utf8** (ceci permet d'afficher correctement les caractères éèùà ... du langage français) : **<meta charset="utf-8">** dans la balise **<head>**

- Ajouter le titre dans les attributs de la page : `<title>TP1 programmation Web</title>`, donner le rôle de la balise `<title>`

Premier Test : Afficher votre “page” HTML dans le Navigateur

- Par la suite toutes les modifications se font dans le corps de la page Web (**body**), insérer :
 - une image servant de lien vers **exercice1.html**. (balise `img`) avec le texte alternatif “mon image devrait apparaître” de largeur 250 pixels et de hauteur 200 pixels et un titre au choix (balise ``). Remarque le symbole `/` fin de balise n’est pas obligatoire. Utiliser les attributs **width** et **height** pour définir les dimensions de l’image. Cette balise image est placée à l’intérieur d’une balise lien (` `)
 - une ligne horizontale. (balise `<hr>`)
 - un lien (interne) intitulé “lien vers exercice1” vers `exercice1.html` (balise ` texte du lien `)
 - un lien (externe) ` Cours – TD – TP `, donner le rôle de l’attribut : **title**
- rajouter une icône dans la partie **head** :
exemple : `<link rel="icon" href="path/to/favicon-16.png" sizes="16x16" type="image/png">`
Aidez vous de cette [documentation](#), voir de la [documentation de W3C](#)



[Lien vers page3](#)
[Cours - TD - TP](#)

aperçu écran de la page “page1.html”

Valider votre page sur le site <http://validator.w3.org/>

Remarque : [Outils/Développement Web/Code source de la page] ou [Ctrl u] affiche en Rouge les erreurs détectées par le “parseur” du navigateur

Remarque : Certaines erreurs peuvent être liées au fait que l’on ne peut pas mettre certaines balises dans d’autres balises. Vous pouvez vous aider des sites présentés lors de la configuration de votre bureau pour connaître les balises parents/enfants.

Exercice 3

- Créer une page de nom **page2.html** qui contient :
 - un lien avec le texte suivant : « retour à la page 1 d'accueil » et une image pour retourner sur la page 1
 - une ligne horizontale
 - un lien dans une image sur <http://www.google.fr> en utilisant l'attribut *target* pour ouvrir un nouvel onglet dans le navigateur (attribut **target="_blank"**). Choisir une image sur internet et mettre comme dimension 100px X 100 px avec les attributs **WIDTH** et **HEIGHT**.

Remarque importante sur les liens : les liens utilisent des URLs, voici une page avec quelques exemples d'URLs : <http://defeo.lu/aws/lessons/url> , une erreur classique est d'oublier le **http://** devant une URL absolue.

[retour](#)



aperçu écran de la page “page2.html”

Valider votre page sur le site <http://validator.w3.org/>

Exercice 4

Créer une page de nom **page3.html** qui contient (de haut en bas) :

- le texte noir suivant en utilisant les méta caractères pour les caractères : û é €:
“le produit coûte éventuellement 200 €”
- IUT (en utilisant la balise **<abbr>** pour afficher « Institut Universitaire de Technologie » si la souris est dessus)
- $e^{-1/x} * i_5$ (voir aperçu écran en utilisant les balises **SUB** et **SUP**)
- un texte gras puis gras italique puis gras italique gros (en utilisant les balises **<i>**)

Aidez vous de cet exemple :

le produit coûte éventuellement 200 €

I.U.T.

$e^{-1/x} * i_5$

un texte gras puis gras italique puis gras italique gros(strong)

aperçu écran du début de la page “page3.html”

REMARQUE : la lettre b de la balise **** signifie *bold* (gras) alors que **** signifie important (de même **** signifie emphase : mise en valeur). [Exemple de site web](#) qui parle de l'intérêt sémantique de ces balises.

Compléter la page HTML avec :

- Une ligne horizontale.
- 2 listes à puces imbriquées dans une autre liste numérotée. Laisser les formes des puces par défaut, en HTML les mises en forme se font avec du CSS) :

```

21 <ol>
22   <li> ligne 1
23     <ul>
24       <li> ligne 1.1</li>
25       <li> ligne 1.2</li>
26     </ul>
27   </li>
28
29   <li> ligne 2
30     <ul>
31       <li> ligne 2.1</li>
32       <li> ligne 2.2 </li>
33     </ul>
34   </li>
35 </ol>
36
37 <hr />
38 <a href="tpl_page1.html">retour vers page1</a><br />
39 <br />

```

-
1. ligne 1
 - ligne 1.1
 - ligne 1.2
 2. ligne 2
 - ligne 2.1
 - ligne 2.2
-

[retour vers page1](#)

REMARQUE : En langage HTML pour imbriquer une liste dans une autre liste, celle ci doit être imbriquée dans une balise ``. La signification de `ol` ou `ul` ne définit pas forcément une liste à puces ou numérotée mais juste une liste d'éléments ordonnés ou non ordonnés (*ul: unordered list, ol: ordered list*).

- Une ligne horizontale.
- Un lien interne sur la page 1 (exercice1.html)

Exercice 5

- Créer une page (fichier) de nom **exercice5.html** normalisée : avec une balise `<doctype>` `<html>` `<head>` `<body>` et les balises `<meta>`. Rajouter les balises pour mettre un titre, afficher les caractères spéciaux (èàè...) en utf8, afficher une icône (Faire un copier/coller du tp précédent).
- Rajouter dans la balise `<body>` un tableau de la forme suivante :

cellule1	cellule2	cellule 4
cellule3		
cellule5	cellule 6	

aperçu écran du tableau


```

<table border="4">
<tr><td>cellule1</td><td>cellule2</td><td>cellule 4</td></tr>
<tr><td>cellule3</td><td></td><td></td></tr>
<tr><td>cellule5</td><td>cellule 6</td><td></td></tr>
</table>

```

code source html pour créer le tableau

- Rajouter l'attribut `border="4"` dans la balise `<table>` pour afficher les bordures du tableau (ce style sera placé par la suite dans un fichier de style CSS)
- Rajouter des liens dans le tableau :
 - dans cellule 4 : un lien pour aller sur page2.html
 - dans cellule 6 : une image qui permet de faire un lien pour aller sur la page <https://www.google.com/>

cellule1	cellule2	
cellule3		
cellule5	 cellule 6	
		cellule4

aperçu écran du tableau

- Utiliser **colspan** et **rowspan** pour fusionner les cellules, utiliser [la documentation suivante](#)

Valider votre page sur le site <http://validator.w3.org/>

Remarque : [Outils/Developpement Web/Code source de la page] ou [Ctrl u] affiche en Rouge les erreurs détectées par le parseur du navigateur

Exercice 6

Créer une page de nom **exercice5.html** normalisée qui contient (de haut en bas) :

- Un formulaire avec la balise `<form>`.
- Ajouter dans cette balise l'attribut «action» ayant pour valeur "page1.html" : `action="page1.html"`, la propriété «action» est obligatoire dans cette balise.
- Ajouter également comme propriété `method="get"` (2 valeurs possibles **GET** ou **POST**, le rôle très important en PHP !)

REMARQUE IMPORTANTE : Lorsque vous soumettez le formulaire : le serveur web va charger comme nouvelle page le contenu de l'attribut «**action**» : la page "exercice1.html". Le serveur passe en paramètres tous les champs «**input**». Chaque paramètre possède comme nom l'attribut «**name**» du champ «**input**» et comme valeur le contenu de champ ou sa valeur «**value**» (attribut non obligatoire). Avec la méthode "GET", on peut voir ces paramètres dans l'URL. Avec la méthode "POST", ils sont cachés.

- rajouter la balise `<fieldset>`
- Ce formulaire contient :
- un champ de saisie texte avec comme label "nom :" et nommé "nom" (cf. propriété name).
 - un champ de saisie texte avec comme label "pseudo (login) :" et nommé "pseudo" (cf. propriété name).

nom :	<input type="text" value="votre nom ?"/>
pseudo (login) :	<input type="text" value="entrer votre login"/>
<input type="button" value="valider"/>	

aperçu écran du formulaire

```
<form method="get" action="page2.html" >
<fieldset>
  <label>nom :</label>
  <input type="text" name="nom" value="votre nom ?" />
  <br />
  <label>pseudo (login) :</label>
  <input type="text" name="pseudo" value="" placeholder="entrer votre login" />
  <br />
  <input type="submit" name="valider" value="valider" />
</fieldset>
</form>
```

code source html pour créer le formulaire

Remarque : un formulaire se compose généralement de la balise `<form>`, avec à l'intérieur la balise `<fieldset>`. Chaque champ se compose ensuite de `<label>` pour le texte et `<input>` pour le champ de saisie.

L'attribut `placeholder` est une indication pour l'utilisateur. L'attribut `value` est la valeur initiale du contrôle, c'est cet attribut qui est fonctionnellement très important.

Rajouter ensuite :

- Un champ "caché" (cf. doc HTML) nommé "cache" ayant pour valeur un chiffre quelconque.
- 2 champs de saisie de « password » nommés "passwd1" et "passwd2".
- un champ de saisie pour entrer un nom de fichier nommé "fichier". un bouton de soumission nommé "valider".

Après avoir rempli le formulaire et appuyé sur le bouton, vous devriez voir apparaître dans l'URL les valeurs entrées dans les champs.

Validateur HTML / XHTML / ... TP2 programmation W

nom :

pseudo (login) :

mot de passe :

mot de passe (recommencer):

fichier (photo):

aperçu écran du formulaire final

Conclusion : visualiser le contenu de l'URL, son contenu sera très utile en PHP !

Valider votre page sur le site <http://validator.w3.org/>

Pour les plus rapides :

Consulter la documentation sur le champ `input`. Tester d'autres types de champ d'entrée : "date", "number", "range" ... pour ces nouveaux champs "input"