

janvier 2016

PHP

**Exercice 1 :** Vous transfèrerez le fichier `exempleLivre.txt` ainsi que le répertoire `couvertures` et son contenu dans votre dossier de travail sur webtp .

Le but final de l'exercice sera de générer en PHP un document HTML à partir des données figurant dans le fichier `exempleLivre.txt`. Nous procéderons par étapes, en structurant le code PHP.

Le fichier `exempleLivre.txt` contient la description d'un ouvrage. Examinez son contenu. Chaque livre y est décrit par une suite de lignes, chacune représentant un type d'information particulier. Le format de chaque ligne est en deux parties séparées par le caractère `:`

- en premier on trouve le type d'information fournie
- et en deuxième on trouve l'information elle-même.

Les espaces éventuels situés au début et à la fin de chaque partie ne sont pas à prendre en compte. On supposera (pour l'instant) que la première ligne contient toujours la désignation de la couverture.

Question 1.1 :

Construire une fonction PHP `propriete($ligne)` qui reçoit en argument une **chaîne** supposée contenir une ligne de description .

Le résultat de la fonction est un **tableau associatif PHP** contenant un seul élément : La clé sera la partie gauche de la ligne de propriété, sans les espaces initiaux et terminaux, et sans parenthèses. La valeurs associée sera le partie gauche de la ligne de description sans les espaces initiaux et terminaux.

Par exemple `propriete("auteur(s) : Marini - Desberg")` sera la table

clé	valeur
"auteurs"	"Marini - Desberg"

Le modèle de document à produire vous est donné dans le fichier `exempleLivre.html`.

Examinez la partie «description» de l'ouvrage. Vous remarquerez que

- chaque information élémentaire est placée dans une balise `div`, sauf l'année qui est placée dans une balise `time`
- le nom utilisé pour la classe est obtenu à partir du type d'information indiqué dans le fichier de données, en supprimant toutefois les parenthèses éventuelles (`auteur(s)` → `auteurs`).

Chaque auteur est, de plus, placé dans un élément `span` (à l'intérieur du `div class="auteurs"`)

Question 1.2 : Construire une fonction PHP `proprieteToHTML()` qui reçoit en argument un **tableau associatif PHP** analogue à celui fabriqué à la question précédente. On supposera qu'il ne s'agit pas la propriété de type "couverture".

Le résultat de la fonction est une **chaîne de caractères** qui contient la représentation de cette ligne en HTML, comme indiqué ci-dessus.

Par exemple le résultat de l'expression

```
proprieteToHTML(propriete("série : Le Scorpion")) ;
```

sera la chaîne `'<div class="série">Le Scorpion</div>'`

NB : vous pouvez connaître la clé d'un tableau PHP d'un seul élément grâce à la fonction `key()`

Question 1.3 : Construire une fonction PHP `couvertureToHTML(...)` dont le premier argument est l'URL d'une image de couverture et le deuxième argument une chaîne à insérer dans l'attribut `alt`. Ce deuxième argument aura une valeur par défaut (la chaîne `"couverture"`).

Le résultat de la fonction sera une chaîne représentant l'élément `div` représentant la couverture.

Par exemple `couvertureToHTML("scorpion.jpg")` sera

```
'<div class="couverture"></div>'
```

Question 1.4 : Avant d'aborder la suite de l'exercice, vous allez créer un fichier `fonctionsLivre.php` dans le quel vous allez copier les 2 fonctions que vous venez de créer ainsi que la fonction `enSpan()` (feuille précédente). Ce fichier contiendra **UNIQUEMENT** les définitions de fonctions.

Question 1.5 : Vous allez maintenant écrire la page `livreUnique.php` générant automatiquement la représentation HTML à partir des informations fournies dans le fichier `exempleLivre.txt`.

On supposera que la première ligne de description concerne l'information «couverture». Les autres informations figureront dans le fichier HTML dans l'ordre où elles apparaissent dans le fichier texte.

Vous utiliserez bien évidemment les fonctions que vous venez d'écrire. Il vous faudra faire figurer dans votre fichier la commande `require('fonctionsLivre.php');`

Question 1.6 : Transférez le fichier `livres.txt` dans votre dossier de travail sur webtp. Ce fichier contient la description de plusieurs livres. Les descriptions de 2 livres successifs sont séparées par une ligne vide.

Écrire un nouveau script PHP nommé `bibliotheque.php` qui engendre une page HTML avec l'ensemble des livres décrits dans le fichier.

Question 1.7 : (à faire en dehors des séances) Associer aux pages créées lors des deux dernières questions un style permettant d'obtenir une meilleure présentation. Voir à ce sujet l'exercice du l'UE TW1 consacré à cette question.

#### Question 1.8 : Représentation d'un livre dans une structure de données

Jusque là la génération de code HTML pour représenter un livre se faisait «à la volée», au fur et à mesure de la lecture des données textuelles.

Nous allons maintenant définir une représentation interne (dans des structures PHP) d'un livre.

Une première version consistera à représenter un livre par une table associative PHP qui possèdera une clé pour chaque propriété : la clé sera le nom de la propriété et la valeur associée sera l'information correspondante.

1. Définissez une fonction `prochainLivre($file)` L'argument est un fichier ouvert. La première ligne disponible dans ce fichier est supposée être la première ligne de description d'un ouvrage. La description de l'ouvrage se termine soit par la fin du fichier, soit par une ligne vide. Le résultat de la fonction sera une table associative représentant le livre décrit. En fin d'exécution de la fonction on aura progressé dans la lecture du fichier : toutes les lignes de la description auront été consommée, ainsi que la ligne vide qui suit (le cas échéant).

2. Définissez une fonction `livreToHTML($livre)` dont l'argument est une table associative représentant un livre et dont le résultat est une chaîne contenant le code HTML présentant ce livre.

La structure HTML produite sera similaire à celle des questions précédentes, avec quelques particularités :

- L'attribut `alt` de l'image contiendra le titre de l'ouvrage précédé de «Couverture de :»
- les autres propriétés seront présentées dans l'ordre : auteurs, titre, année, série, catégorie.

3. En utilisant ces deux fonctions, écrivez un script `bibliotheque2.php`