

mars 2013

PHP

Exercice 1 : Sur papier

On considère le code PHP suivant :

```
<?php

$i=0;
print( $i++ + "3 cochons" . " chiens\n");
print("$i" . "3 cochons" . " chiens\n");
print( $i + "4.1 cochons" + " chiens" . (3/2) . "\n");

?>
```

Question 1.1 : Indiquez le texte affiché lors de l'exécution de ce code.

Exercice 2 : Sur papier

```
<?php
/* quelques exemples de chaînes de caractères */
$a = 'PHP';
$b = 'MySQL';
echo "$a et $b " ;
echo "<br/>\n";
echo '$a et $b';
echo "<br/>\n";
echo "La cérémonie d'ouverture a lieu le . date('d')";
echo "<br/>\n";
echo "La cérémonie d'ouverture a lieu le " . date('d');
echo "<br/>\n";
echo 'La cérémonie d\'ouverture a lieu le ' . date("d");
echo "<br/>\n";
echo "Sa devise est \"Liberté Egalité Fraternité\" ";
echo "<br/>\n";
echo $a . " et " . $b ;
echo "<br/>\n";
echo $a , " et " , $b ;
echo "<br/>\n";
```

Question 2.1 : Indiquez le texte affiché lors de l'exécution de ce code. Attention, il peut y avoir des erreurs de syntaxe.

```
<?php
/* tableau */
```

```
$tab[0] = 2004 ;
$tab[1] = 31.14E7 ;
$tab[2] = "PHP4";
$tab[35] = 'MySQL' ;
$tab[3] = 24 ;
$tab[] = TRUE ;
?>
```

Question 2.2 : Que contient le tableau après l'exécution de ce code ?

Question 2.3 : Écrivez le code PHP qui permet d'obtenir le nombre d'éléments de `$tab` ainsi que ses valeurs.

Exercice 3 :

Voici un exemple de code HTML « statique » qui permet d'afficher sous forme d'une table une liste de livres. On désire rendre dynamique la génération de ce tableau. C'est à dire que la liste des livres sera mémorisée dans des structures PHP et qu'il faudra écrire des fonctions PHP permettant d'obtenir, à partir de ces données, un document HTML semblable à celui qui figure ci-contre.

Nous allons examiner plusieurs solutions possibles quant à la représentation des données en PHP

Notez que cet exercice se prolongera sur la prochaine feuille de TD.

```
<html>
<head>
  <title>liste de livres</title>
</head>
<body>
  <h1>Mes livres:</h1>
  <table>
    <tr>
      <td class="auteur">Mark Lutz</td>
      <td class="titre">Python (en français)</td>
    </tr>
    <tr>
      <td class="auteur">Gérard Swinne</td>
      <td class="titre">Apprendre à
        programmer avec Python</td>
    </tr>
  </table>
</body>
</html>
```

Question 3.1 : Dans un premier temps, les données figurent dans 2 tableaux distincts. L'un contient les auteurs et l'autre les titres. Les 2 tableaux sont supposés avoir la même longueur

```
$auteurs = array('Mark Lutz', 'Gérard Swinnen');
$titres = array('Python (en français)', 'Apprendre à programmer avec Python');
```

Écrivez le code PHP d'une fonction `tabLivres1` dont le résultat est une chaîne contenant la table HTML (et uniquement elle) obtenue à partir des données de deux tableaux qui sont ses arguments (tableau des auteurs, tableau des titres).

Notez bien que la fonction en elle-même ne doit rien « afficher » sur la sortie standard : la table HTML est renvoyée sous la forme d'une chaîne. C'est au programme appelant de faire en sorte que ce code HTML soit envoyé sur la sortie standard.

Insérez cette fonction dans une page HTML et faites en sorte que le contenu de la table s'affiche.

Question 3.2 : La structure précédente n'est pas satisfaisante, notamment parceque les données d'un même livre sont disséminées dans deux entités différentes. Nous allons maintenant représenter un livre par un tableau associatif contenant 2 clés : `"auteur"` et `"titre"`. L'ensemble des livres sera constitué par un unique tableau « classique » indexé par des indices numériques successifs, comme ceci :

```
array(
  array('auteur'=>'Mark Lutz','titre'=>'Python (en français)'),
  array('auteur'=>'Gérard Swinnen','titre'=>'Apprendre à programmer avec Python')
)
```

Écrivez une fonction `tabLivres2` ayant le même rôle que la fonction précédente (cette nouvelle fonction aura pour seul argument la liste de livres). Testez-la.

Question 3.3 : Un livre peut avoir plusieurs auteurs, mais les structures précédentes n'ont pas prévu ce cas. Proposez une évolution de la structure de données permettant de prendre ceci en compte.

Écrivez et testez la fonction `tabLivres3` adaptée à cette nouvelle structure.