



Programmation Web Serveur

TP 1

Les exercices suivants sont les exercices du premier chapitre de cours de PHP. Pour tous les exercices, nous allons utiliser le serveur local. Pour cela, il suffit de mettre ces documents dans le répertoire public_html qui se trouve dans votre répertoire personnel. Pour accéder à ces documents à travers un navigateur, il faut alors taper l'adresse Web :

`http://localhost/~VOTRE_NUMERO_ETUDIANT/chemin/nom_du_fichier`

où chemin correspond au chemin où se trouve votre fichier à partir du répertoire public_html. Par exemple, si votre numéro d'étudiant est le 001, si vous créez un dossier TP1 dans votre répertoire public_html et un fichier exercice1.php dans TP1, pour accéder à ce fichier, vous devez spécifier l'adresse Web :

`http://localhost/~001/TP1/exercice1.php`

Attention : Les données écrites dans votre répertoire personnel sont supprimées à chaque redémarrage de l'ordinateur. Les fichiers que vous souhaitez garder doivent être copiés dans le répertoire /iutv/Mes_Montages/login

Comme l'ensemble des réponses aux exercices contiennent du code PHP, il n'est pas possible d'ouvrir le fichier directement à l'aide du navigateur. Il est donc impératif de passer par le serveur local.

Pour tous les exercices, nous utiliserons le navigateur Mozilla Firefox.

Remarque : Les exercices notés par une astérisque proviennent du site openclassroom.

Exercice 1 : Premiers pas en PHP

Voici un code php.

```
<!doctype html>
<html>
<head>
<title> <?php echo 'Premiers pas en PHP'; ?> </title>
<meta charset="utf-8"/>
</head>
<body>
<h1> Mes premiers pas en PHP </h1>
<?php $tps=2; echo '<p> Je débute depuis '; echo $tps; echo ' heures... </p>';?>
<p> Mais cela a l'air intéressant <?php echo '!' ?> </p>
<?php echo '
<h2> Vive le PHP </h2>
<p> Les pages vont pouvoir être dynamiques! </p>
'; ?>
<p> Encore quelques paragraphes </p>
echo '<p> Avant dernier paragraphe </p>';
<p> Voilà, c'est terminé! </p>
</body>
</html>
```

Répondre aux questions suivantes :

- Donner dans ce fichier les parties correspondant à du code PHP et à du code HTML.
- Si ce fichier s'appelle exo1.php, et s'il est stocké sur le site Web www.exemple.org dans le répertoire PremierTP, comment "exécuter" ce fichier? Que donne son exécution? Ceci est-il valide? Pourquoi? Corriger alors le problème.
- Remplacer les trois instructions echo de la ligne 9 par une seule.

Exercice 2 : Inclusion d'en-tête et pied de page

Le langage PHP permet d'inclure des fichiers dans d'autres. Ceci permet alors de décomposer un code HTML ou PHP en plusieurs parties logiques et d'insérer ensuite ces différentes parties dans un fichier PHP.

- Créer un fichier debut_code_html.php contenant tout le code HTML jusqu'à la balise <body> incluse.
- Créer un fichier fin_code_html.php contenant les balises </body></html>.
- Créer un fichier code.php contenant un titre et un paragraphe et incluant les fichiers debut_code_html.php et fin_code_html.php.
- Modifier le fichier debut_code_html.php pour que le titre de l'onglet (balise <title>) dépende de la valeur de la variable \$title définie dans le fichier code.php.

Exercice 3 : Initiation aux tableaux

Voici un tableau contenant les différents mois de l'année :

```
$mois = [  
    "janvier",  
    "février",  
    "mars",  
    "avril",  
    "mai",  
    "juin",  
    "juillet",  
    "août",  
    "septembre",  
    "octobre",  
    "novembre",  
    "décembre"  
];
```

- Quelles sont les clés ? Parcourir ce tableau pour afficher les mois de l'année sous la forme d'une liste ordonnée.

On considère maintenant le tableau suivant donnant pour chaque mois de l'année, le nombre de jours qu'il contient (On suppose que l'année n'est pas bissextile.) :

```
$mois = [
    "janvier"    => 31,
    "février"    => 28,
    "mars"       => 31,
    "avril"      => 30,
    "mai"        => 31,
    "juin"       => 30,
    "juillet"    => 31,
    "août"       => 31,
    "septembre"  => 30,
    "octobre"    => 31,
    "novembre"   => 30,
    "décembre"   => 31
];
```

- Quelles sont les clés du tableau ? Les valeurs ? Parcourir ce tableau pour afficher les mois de l'année sous la forme d'une liste ordonnée (comme dans la question précédente).
- Modifier le code pour afficher dans la liste après le nom de chaque mois le nombre de jours qu'il contient.

Exercice 4 : Tableau à deux dimensions

On définit le tableau suivant :

```
$personnes = [
    'mdupond' => ['Prénom' => 'Martin', 'Nom' => 'Dupond', 'Age' => 25, 'Ville' => 'Paris'           ],
    'jm'      => ['Prénom' => 'Jean'   , 'Nom' => 'Martin', 'Age' => 20, 'Ville' => 'Villetaneuse'],
    'toto'    => ['Prénom' => 'Tom'    , 'Nom' => 'Tonge'  , 'Age' => 18, 'Ville' => 'Epinay'           ],
    'arn'     => ['Prénom' => 'Arnaud', 'Nom' => 'Dupond', 'Age' => 33, 'Ville' => 'Paris'           ],
    'email'   => ['Prénom' => 'Emilie', 'Nom' => 'Ailta'  , 'Age' => 46, 'Ville' => 'Villetaneuse'],
    'dask'    => ['Prénom' => 'Damien', 'Nom' => 'Askier', 'Age' => 7  , 'Ville' => 'Villetaneuse']
];
```

- Quelles sont les clés du tableau `$personnes` et leur type ? De quel type sont les valeurs de ce tableau ? Quelle est la valeur associée à 'toto' ?
- Comment accéder à la valeur 33 dans le tableau ? À la valeur 'Epinay' ? Au tableau contenant les valeurs 'Damien', 'Askier', 7, 'Villetaneuse' ?
- Écrire une fonction prenant en paramètre un tableau (avec une structure similaire à `$personnes`) et l'affichant sous forme d'une table HTML. Le code HTML obtenu avec le tableau `$personnes` doit être :

```
<table>
  <tr>
    <td>Martin</td>
    <td>Dupond</td>
    <td>25</td>
    <td>Paris</td>
  </tr>
  <tr>
    <td>Jean</td>
    <td>Martin</td>
    <td>20</td>
    <td>Villetaneuse</td>
  </tr>
  <tr>
    <td>Tom</td>
    <td>Tonge</td>
    <td>18</td>
    <td>Epinay</td>
  </tr>
  <tr>
    <td>Arnaud</td>
    <td>Dupond</td>
    <td>33</td>
    <td>Paris</td>
  </tr>
  <tr>
    <td>Emilie</td>
    <td>Ailta</td>
    <td>46</td>
    <td>Villetaneuse</td>
  </tr>
  <tr>
    <td>Damien</td>
    <td>Askier</td>
    <td>7</td>
    <td>Villetaneuse</td>
  </tr>
</table>
```

- Modifier la fonction précédente pour ajouter une ligne d'en-têtes. Les en-têtes seront les clés du tableau contenu dans la première case.

Dans le cas de `$personnes` , les en-têtes seront Prénom, Nom, Age, Ville.

Indice : Utiliser `array_keys($personnes)[0]` pour récupérer la première clé du tableau `$personnes` .

- Vérifier que l'affichage obtenu est correct lorsque la fonction prend en paramètre le tableau suivant :

```
$scores = [
    [ 'Joueur' => 'Camille' , 'score' => 156 ],
    [ 'Joueur' => 'Guillaume', 'score' => 254 ],
    [ 'Joueur' => 'Julien'   , 'score' => 192 ],
    [ 'Joueur' => 'Nabila'   , 'score' => 317 ],
    [ 'Joueur' => 'Lorianne' , 'score' => 235 ],
    [ 'Joueur' => 'Tom'      , 'score' => 83  ],
    [ 'Joueur' => 'Michael'  , 'score' => 325 ],
    [ 'Joueur' => 'Eddy'     , 'score' => 299 ]
];
```

Exercice 5 : Tableau à deux dimensions

On considère les deux tableaux suivants :

```
$tabMagazines = [
    'le monde'           => [ 'frequence' => 'quotidien', 'type' => 'actualité', 'prix' => 220 ],
    'le point'           => [ 'frequence' => 'hebdo'    , 'type' => 'actualité', 'prix' => 80  ],
    'causette'           => [ 'frequence' => 'mensuel'   , 'type' => 'féministe', 'prix' => 180 ],
    'politis'            => [ 'frequence' => 'hebdo'    , 'type' => 'opinion'  , 'prix' => 100 ],
    'le monde diplomatique' => [ 'frequence' => 'mensuel' , 'type' => 'analyse'  , 'prix' => 60  ]
];

$tabMagazinesAbonne = [ 'le monde', 'le monde diplomatique' ];
```

1. Afficher sur une ligne le nom de tous les magazines triés par ordre alphabétique et séparés par des virgules, sans faire de boucle. Vous utiliserez des fonctions déjà existantes telles que [implode](#), [sort](#) et [array_keys](#) que vous trouverez dans le [manuel php](#).
2. Afficher les magazines exactement de la façon suivante en supposant qu'il peut y avoir beaucoup de magazines et beaucoup de propriétés associées :
 - le monde (quotidien, actualité, 220)

- le point (hebdo, actualité, 80)
- causette (mensuel, féministe, 180)
- politis (hebdo, opinion, 100)
- le monde diplomatique (mensuel, analyse, 60)

3. En utilisant le tableau tabMagazinesAbonne contenant le nom des magazines d'un abonné, calculer le prix total de son abonnement.

Exercice 6 : Objets

1. Créer une classe date contenant 3 attributs privés:

- `$d` pour coder le jour,
- `$m` pour coder le mois,
- `$y` pour coder l'année.

2. Créer le constructeur prenant en paramètre une chaîne de caractères sous la forme DD/MM/YYYY et initialisant les attributs en fonction des valeurs dans la chaîne. Si le format n'est pas bon (il n'y a pas 3 entiers séparés par des slashes), mettre comme date le 01/01/2000.

3. Créer la méthode display affichant la date sous la forme DD/MM/YYYY.

4. Créer la méthode isLeap retournant true si l'année est bissextile, et false sinon.

5. Créer un attribut constant JOURS_MOIS qui est un tableau de 12 cases, chaque case contenant le nombre de jours dans le mois (on mettra 29 jours pour le mois de février). Faire commencer les indices du tableau JOURS_MOIS à 1.

6. Définir la fonction isValid retournant true si la date est valide, et false sinon. Modifier le constructeur afin que celui-ci initialise la date à 01/01/2000 si la chaîne de caractères passée en paramètre ne correspond pas à une date valide.

Exercice 7 : Paramètres dans l'url

1. Appeler une page PHP en passant dans l'url un paramètre de nom pseudo. Le script devra vérifier si la valeur du paramètre correspond à un pseudonyme (c'est-à-dire une clé) du tableau `$personnes` défini dans l'exercice 4. Si c'est le cas, le script doit afficher le pseudo et les informations associées contenues dans le tableau `$personnes`. Il doit sinon afficher Désolé, votre pseudonyme n'apparaît pas dans la liste.
2. Créer un formulaire permettant à l'utilisateur de saisir le pseudonyme à rechercher afin de faciliter la saisie pour l'utilisateur. Mettre ensuite directement le formulaire dans le script PHP créé précédemment afin de pouvoir effectuer facilement plusieurs recherches. Faire en sorte que le champ de saisie du pseudonyme contienne la dernière valeur saisie.
3. Créer un deuxième formulaire demandant un pseudonyme, un prénom, un nom, un âge et une ville, et ajoutant dans le tableau `$personnes` une nouvelle personne dont les valeurs sont celles données par le formulaire. Ajouter plusieurs personnes et expliquer alors le problème. (Afficher le tableau en entier pour mieux voir le problème.) Comment remédier à ceci (réponse sans code car les connaissances nécessaires à la réponse n'ont pas encore été vues en cours) ?

Exercice 8 : Contenu HTML protégé par mot de passe*

Le but de cet exercice est de créer du contenu protégé par un mot de passe.

1. On suppose dans un premier temps qu'il n'y a qu'une seule page Web dont le contenu est protégé. Pour cela, choisir un mot de passe (par exemple : kangourou). Créer un formulaire permettant de saisir un mot de passe. Le contenu protégé doit alors s'afficher uniquement si le mot de passe est correct. Dans le cas contraire, le formulaire doit de nouveau s'afficher.
2. On suppose maintenant qu'il y a plusieurs pages (site Web) dont le contenu doit être protégé par un mot de passe. Télécharger [l'archive](#). Les 3 fichiers de cette archive doivent être protégés par un mot de passe. De plus, l'utilisateur doit se connecter une seule fois. Dès qu'il est connecté, il peut avoir accès à toutes les pages.

Indication : Il est nécessaire d'utiliser les sessions. Créer le fichier securite.php et l'inclure en début de chaque fichier à protéger. Le fichier securite.php doit activer les sessions. Il doit ensuite tester si la variable `$_SESSION` contient une clé connecte. Si ce n'est pas le cas, le formulaire doit être affiché puis le script terminé. Dans le cas contraire, la page doit s'afficher normalement. Le fichier securite.php doit également tester si le formulaire a été soumis avec le bon mot de passe pour mettre à jour la variable `$_SESSION`.

Notes :

- Pour terminer un script, il est possible d'utiliser la fonction [exit](#).

- La variable `$_SERVER["PHP_SELF"]` contient le nom du fichier appelé dans l'url. Il peut être utilisé comme valeur du paramètre action de la balise form pour que le même fichier soit appelé lors de la soumission du formulaire.

3. Donner la possibilité à l'utilisateur de se déconnecter sur n'importe quelle page. Pour cela, créer un fichier deconnexion.php. Ce script affichera un lien hypertexte sur le fichier appelé avec le paramètre action de valeur deconnexion. Inclure ce fichier dans page1.php, page2.php et page3.php. Dans le fichier securite.php, faire en sorte que lors d'un clic sur ce lien hypertexte, l'utilisateur soit déconnecté (il doit saisir à nouveau son mot de passe).

Exercice 9 : Calculatrice*

Le but de cet exercice est de créer une calculatrice simple.

1. Créer un formulaire permettant à l'utilisateur de saisir une opération simple. Pour cela, l'utilisateur devra saisir deux nombres (dans deux balises input de type text) et choisir une opération, à l'aide d'un menu déroulant select, parmi l'addition, la soustraction, la multiplication ou la division. Lors de la soumission du formulaire, le résultat de l'opération doit être affiché.
2. Afin de faciliter la saisie par l'utilisateur, créer un formulaire contenant une seule balise input de type text où l'utilisateur saisit directement l'opération (par exemple : 17.5 * 13.46). Lors de la soumission du formulaire, vérifier, à l'aide des expressions régulières, que le texte saisi par l'utilisateur correspond à une opération simple ; dans ce cas, afficher le résultat. Vous utiliserez le 3ème paramètre de preg_match.

Exercice 10 : Questionnaire à choix multiples

1. Le but de cet exercice est de faire une première page Web contenant des énigmes ou questions avec plusieurs réponses possibles (une seule d'entre elles étant correcte). L'utilisateur doit alors répondre aux différentes questions et le site affiche alors le score obtenu par l'utilisateur.

- 2. Ajouter sur la page d'accueil un formulaire pour que l'utilisateur puisse saisir son nom. Cette donnée sera utilisée par la suite pour personnaliser l'affichage du site.
- 3. Stocker, à l'aide d'un cookie le meilleur score obtenu par l'utilisateur. Dans la page du score, afficher ce meilleur score en plus du score obtenu à l'instant par l'utilisateur.