

CIR1

Web dynamique – travaux pratiques 2

Nathalie HELAL

1 Préambule

- Créez le répertoire TP2 (dans le répertoire C:\EasyPHP-Devserver-17\eds-www) dans lequel vous mettrez l'ensemble de vos fichiers php, et autre documents (si besoin) pour ce TP.
- A la fin de ce TP, il faut déposer (sur [campus.isen](http://campus.isen.fr)) l'ensemble de vos fichiers **sous forme d'un seul dossier compressé .zip**. Attention, nommez chaque exercice comme indiqué.

2 Objectif de ce TP

Les fonctions intégrées PHP par exemple :

- Obtenir des informations sur les variables
- Fonctions de chaînes de caractères
- Travailler avec les sous-chaînes de caractères
- Les fonctions mathématiques
- Génération de nombres aléatoires
- Documentation
- Comment trouver les fonctions intégrées
- Fonctions de tableaux : affichage, ajout, modif, push, pop, shift et unshift.

Traiter et collecter les données des formulaires notamment avec les méthodes `_GET` et `_POST`.

PARTIE A

3 Fonctions de chaînes de caractères

Créez une page **fonction_chaineschar.php** permettant :

1. d'imprimer la chaîne suivante inversée: ".pu ti peeK .taerg gniod er'uoY" ;
2. d'imprimer la chaîne suivante mais avec toutes les lettres en minuscules: "BIENTÔT, THiS aura l'air NoRmAL."
3. d'imprimer la chaîne "\ nIl n'y a pas de place comme à la maison. \ N" trois fois.

4 Fonctions mathématiques

Créez une page `fonction_math.php`, où vous allez écrire deux fonctions `CalculateDistance()` et `CalculateTip()`, puis testez (appelez) chacune de ces deux fonctions avec différents jeux de données.

4.1 La fonction `CalculateDistance()` calcule la distance entre deux nombres. La fonction doit renvoyer le même résultat, quel que soit l'ordre dans lequel les arguments sont passés dans la fonction. Voici quelques exemples de fonctionnement de cette fonction:

- `calculateDistance(-1, 4)` doit retourner 5
- `calculateDistance(4, -1)` doit retourner 5
- `calculateDistance(3, 7)` doit retourner 4
- `calculateDistance(7, 3)` doit retourner 4

Une fois que vous avez fini d'écrire cette fonction, vous devez l'exécuter et la testez pour vous assurer qu'elle fonctionne correctement.

4.2 La fonction `CalculateTip()` prend comme argument un nombre représentant le coût total d'un repas. Votre fonction doit calculer un nouveau total avec un pourboire de 18% ajouté et renvoyer cette valeur arrondie à l'entier le plus proche, exemple :

- `calculateTip(100)` doit retourner 118
- `calculateTip(35)` doit retourner 41
- `calculateTip(88.77)` doit retourner 105

Une fois que vous avez fini d'écrire cette fonction, vous devez l'exécuter pour vous assurer qu'elle fonctionne correctement.

5 Génération de nombres aléatoires

Utilisez `getrandmax()` pour déterminer le plus grand entier aléatoire dans votre environnement. Maintenant que vous connaissez les limites de votre environnement, utilisez la fonction `rand()`.

Ensuite, effectuez une suite de tirages de nombres aléatoires entre 100 et 999 jusqu'à obtenir une suite composée d'un nombre pair suivi de deux nombres impairs. Affichez chaque tirage jusqu'à obtenir le tirage demandé. Créez la page `tirage.php` qui permet d'afficher cette suite de tirages : ci-dessous deux exemples illustrant ce que vous devez obtenir. Habillez cette page comme illustré dans les deux figures suivantes.



Nombres aléatoires

Suite de tirages

1 : 158, 690, 162
2 : 516, 702, 458
3 : 565, 610, 197
4 : 140, 580, 660
5 : 302, 607, 740
6 : 413, 427, 482
7 : 231, 966, 430
8 : 637, 486, 265
9 : 621, 593, 690
10 : 308, 859, 322
11 : 478, 841, 554
12 : 246, 833, 262
13 : 466, 337, 491

Résultat obtenu en 13 coups

Un autre exemple :

Nombres aléatoires

Suite de tirages

1 : 433, 786, 198
2 : 197, 608, 994
3 : 269, 447, 233
4 : 632, 171, 111

Résultat obtenu en 4 coups

6 Documentation

Ici la documentation de la fonction intégrée `str_pad()`. Soit quatre variables `$a`, `$b`, `$c` et `$d` définies comme indiqué dans le code suivant.

```
<?php
$a = 29;
$b = "You did it!";
$c = STR_PAD_BOTH;
$d = "*~*";
?>
```

Créez la page **doc_str_pad.php** en charge d'appeler **str_pad()** ayant ces quatre variables comme arguments afin qu'elle renvoie la chaîne: `*~**~**~*You did it!*~**~**~*`

Vous devez déterminer dans quel ordre passer les arguments à la fonction. Utilisez `echo` pour imprimer le résultat de l'appel de la fonction.

Aide : consultez la documentation sur cette fonction.

Remarque : L'un des arguments (et sa variable correspondante) est un type PHP appelé: les constantes prédéfinies. C'est l'argument de la variable `$c`. Ici une liste de constantes prédéfinies pour les chaînes de caractères par le langage PHP. Les constantes prédéfinies sont similaires aux variables: ce sont des noms qui contiennent des valeurs. Contrairement aux variables, les constantes prédéfinies sont définies par le langage, pas par nous, et elles sont constantes, et donc ne peuvent pas changer de valeur.

7 Trouver des fonctions intégrées/built-in

Pour en savoir plus sur les fonctions intégrées (et les autres fonctionnalités du langage), c'est utile de vous familiariser avec la documentation PHP. Cependant, la liste est considérable. Pour cela, voici des listes organisées par sujet: [liste de fonctions de chaîne de caractères en PHP](#) et d'une [liste de fonctions mathématiques](#).

Vous pouvez également utiliser Google pour accéder à la documentation officielle de PHP (php.net). Par exemple, lorsque je cherche sur google «valeur absolue PHP», le premier résultat sera un lien vers la fonction `abs()` de la documentation PHP.

Dans les deux exercices qui suivent (7.1, 7.2 et 7.3), vous devez créer un script **find_builtin.php** où vous aller définir des fonctions qui appellent des fonctions intégrées. Cependant, cette fois, vous devez rechercher et trouver la fonction intégrée qu'il faut utiliser dans le corps de la fonction.

7.1 Créez une fonction **convertToShout()** qui prend comme argument une chaîne de caractères et retourne :

1. cette chaîne mais convertie en lettres majuscules ; et
2. ajoute à la fin de la chaîne en majuscules un point d'exclamation.

Vous devez utiliser une fonction intégrée spécifique pour effectuer cette conversion. Une fois que vous avez fini d'écrire votre fonction, vous devez l'exécuter - en utilisant `echo` pour voir sa valeur retournée (pour vous assurer qu'elle fonctionne correctement).

Voici quelques exemples:

- `convertToShout("je ne sais pas")` devrait retourner "JE NE SAIS PAS!"
- `convertToShout("oh, ok, c'est bien")` devrait retourner "OH, OK, C'EST BIEN!"

Remarquez le point d'exclamation à la fin.

7.2 Créez une fonction **CalculateCircleArea()** qui prend comme argument (paramètre) le diamètre d'un cercle et retourne sa surface. Vous devez utiliser une fonction intégrée pour obtenir la précision exacte du Pi que nous recherchons. Une fois que vous avez fini d'écrire votre fonction, vous devez l'exécuter et la tester.

Voici quelques exemples:

- CalculateCircleArea (25) devrait renvoyer 490.87385212341
- CalculateCircleArea (30) devrait renvoyer 706.8583470577
- CalculateCircleArea (872) devrait renvoyer 597204.19707681
- CalculateCircleArea (6) devrait renvoyer 28.274333882308
- CalculateCircleArea (29) devrait renvoyer 660.51985541725

7.3 Créez une fonction qui s'appelle **verificationPassword()**. Elle prendra un argument de type string. Elle devra retourner un boolean qui vaut true si le password respecte les règles suivantes :

1. faire au moins 8 caractères ;
2. avoir au moins 1 chiffre ;
3. avoir au moins une majuscule et une minuscule.

Vous devez utiliser des fonctions intégrées pour implémenter ces conditions.

Notez qu'on reprend cet exercice du TP1 (exercice supplémentaire). Toutefois, on va simplifier le code de la fonction cette fois, en utilisant des fonctions intégrées.

8 Fonctions pour gérer et manipuler les tableaux

Rappel : les tableaux sont des collections ordonnées de données. Ce sont des structures de données fondamentales en informatique.

- L'emplacement d'un élément dans un tableau est appelé son index.
- Les éléments d'un tableau ordonné sont classés par ordre numérique croissant en commençant par l'indice zéro.
- Pour construire des tableaux ordonnés on peut utiliser la fonction PHP intégrée: array() ou une syntaxe de tableau courte, par ex. [1,2,3].
- Nous pouvons imprimer des tableaux en utilisant la fonction intégrée print_r() ou en les convertissant en chaînes de caractères en utilisant la fonction implode() dans echo.
- On peut utiliser les crochets ([]) pour accéder aux éléments d'un tableau par leur index.
- On peut ajouter des éléments à la fin d'un tableau en ajoutant des crochets ([]) au nom de la variable représentant un tableau et en leur attribuant une valeur avec l'opérateur d'affectation (=).
- On peut changer les éléments d'un tableau, en utilisant leur indice et l'opérateur d'affectation.
- La fonction array_pop() supprime le dernier élément d'un tableau.
- La fonction array_push() ajoute des éléments à la fin d'un tableau.
- La fonction array_shift() supprime le premier élément d'un tableau.
- La fonction array_unshift () ajoute des éléments au début du tableau.

Créez le script **fct_tableau.php** où vous allez écrire votre code pour les exercices de cette section 8 de la partie A du TP.

8.1 Étant donné les deux tableaux

```
$message = ["Oh hello", " You're doing great", " Keep up the good work!\n"];
et
```

```
$favorite_nums = [7, 201, 33, 88, 91];
```

Utilisez echo et la fonction implode() pour imprimer le tableau \$message en utilisant "!" comme le \$glu. Utilisez ensuite la fonction print_r() pour imprimer le tableau \$favorite_nums.

8.2 Etant donné le tableau \$stack

```
$stack= ["wild success", "failure", "struggle"];
```

Utilisez la fonction `array_push()` pour ajouter deux éléments au tableau: "blocker" et "impediment". Utilisez ensuite la fonction `array_pop()` pour supprimer tous les éléments du tableau sauf un.

8.3 Utilisons des tableaux (et notre imagination) pour enregistrer les records en athlétisme sur les épreuves de saut de 100 mètres dans les jeux olympiques.

En 1991, Carl Lewis a battu le record de 9,90 secondes. Son record a été dépassé en 1994 par Leroy Burrell. Ensuite en 1996, Donovan Bailey a battu ce dernier, puis Maurice Greene en 1999, et finalement Tim Montgomery en 2002.

Créez un tableau `$record_holders`

```
$record_holders = [ ] ;
```

1. Utilisez la fonction `array_unshift()` pour ajouter "Carl Lewis", "Leroy Burrell", "Donovan Bailey", "Maurice Greene" et "Tim Montgomery" au tableau `$record_holders`. Il faut ajouter les noms tel que les noms mentionnés au début ("Carl Lewis") aient toujours des emplacements d'indices plus élevés que les derniers, par exemple "Tim Montgomery" se trouve à l'indice 0 et "Carl Lewis" à l'indice 4 de `$record_holders`.

Remarque : C'est utile d'imprimer régulièrement votre tableau.

2. Le dossier de Tim Montgomery a été invalidé. Utilisez la fonction `array_shift()` pour le supprimer du tableau `$record_holders`.
3. Ensuite, des nouveaux records ont été remportés par "Asafa Powell" en 2005, puis par "Justin Gatlin" en 2006. Utilisez la fonction `array_unshift()` pour les ajouter au tableau `$record_holders`.
4. Le record de Justin Gatlin a également été invalidé. Utilisez la fonction `array_shift()` pour le supprimer du tableau.

8.4 Reprenez l'exercice 5 du TP1 (section 3.9.5) pour créer des nouvelles versions des fonctions `premierElementTableau()` et `dernierElementTableau()`. Pour rappel, la fonction **`premierElementTableau()`** prend un argument de type array et retourne le premier élément de cet argument(tableau). Si le tableau est vide, la fonction retourne *null*. La fonction **`dernierElementTableau()`** prend un argument de type array et retourne le dernier élément de cet argument. Si ce tableau est vide, la fonction retourne *null*. Toutefois, ces fonctions utilisent cette fois des fonctions intégrées PHP sur les tableaux.



PARTIE B

9 Gestion d'un formulaire

Créez une page **formulaire.html** contenant un formulaire en charge de demander à l'utilisateur son nom (champ de type texte), son prénom (champ de type texte), son genre (bouton radio homme/femme), sa couleur préférée (menu sélection entre rouge, vert, bleu, jaune), et ses sports pratiqués (boîtes à cocher). La page **formulaire.html** peut être constituée du code suivant par exemple :

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title> Formulaire </title>
</head>
<body>
<form method="post" action="formulaire_action.php">
  <fieldset>
    <legend>Informations sur vous</legend>
    <label for="nom">Nom</label> :
    <input type="text" name="nom" id="nom" />
    <label for="prenom">Prénom</label> :
    <input type="text" name="prenom" id="prenom" />
    <hr />
    <input type="radio" name="genre" id="genre" value="homme"/>
    <label for="homme">Homme</label>
    <input type="radio" name="genre" id="genre" value="femme"/>
    <label for="femme">Femme</label>
    <hr />
    <label>Couleur préférée</label> :
    <select name="couleur">
      <option value="rouge">Rouge</option>
      <option value="vert">Vert</option>
      <option value="bleu">Bleu</option>
      <option value="jaune">Jaune</option>
    </select>
    <hr />
    <label>Sports pratiqués</label> :
    <input type="checkbox" name="sports[]" value="football" />Football
    <input type="checkbox" name="sports[]" value="rugby" />Rugby
    <input type="checkbox" name="sports[]" value="golf" />Golf
    <input type="checkbox" name="sports[]" value="jogging" />Jogging
    <input type="checkbox" name="sports[]" value="autre" />Autre
    <hr />
    <label for="soumission">Soumission</label> :
    <input type="submit" name="soumission" id="soumission" />
  </fieldset>
</form>
</body>
</html>
```

Créez la page action **formulaire_action.php** de ce formulaire en charge d’afficher une synthèse des données saisies. Par exemple :

Bonjour madame ou monsieur en fonction du genre saisi suivi du nom et prénom. Votre couleur préférée est ..., vous ne pratiquez aucun sport ou les sports suivants ... en fonction des cases cochés.

Aide : Trouvez le code manquant à la place des ... pour compléter la page formulaire_action.php donnée ci-dessous.

```
<!DOCTYPE html>
<html lang="fr">
<head>
  <meta charset="utf-8">
  <title> Formulaire Action </title>
</head>
<body>

<?php
    $nom=$_POST['nom'];
    ...
    $genre=$_POST['genre'];
    ...
    ...

    echo "<p> Bonjour ";
    if($genre=="homme")
        echo "monsieur ";
    else if ($genre=="femme")
        echo "madame ";
    echo "$nom ... Votre couleur préférée est le ...";

    if(count($sports)==0)
        echo "Vous ne pratiquez aucun sport.</p>";
    else {
        echo "Vous pratiquez";
        foreach ($sports as ... => ... )
            echo " le ... ";
        echo "</p>";
    }
?>
</body>
</html>
```

- Une fois que votre formulaire.html et formulaire_action.php fonctionnent. Essayez d’envoyer des champs vides à la page formulaire_action. Cela entrainera une erreur, exemple : « **Notice:** Undefined index: ... »

- Corrigez cette erreur en modifiant la script formulaire_action.php, afin de vérifier chaque donnée collectée du formulaire est non nulle avant de l'afficher. Vous pouvez utiliser la fonction PHP isset() (voir documentation sur php.net).

10 Convertisseur Euros/Dollars

Créer une page convertisseur.html contenant un formulaire composé de :

- une zone d'options permettant de choisir le sens de conversion que l'on désire, dollars vers euros ou l'inverse (boutons radios) ;
- un champ texte permettant de saisir la valeur que l'on souhaite convertir.

Convertisseur Euros/Dollars

Euros vers Dollars : ☐

Dollars vers Euros : ☐

Valeur :

Réalisez la page **convertisseur_action.php** (l'URL action de ce formulaire), contenant un script PHP qui va convertir la valeur saisie dans le sens précisé.

Ci-dessous, un exemple d'affichage d'une conversion dans le sens Euros vers Dollars

← → ↻ 🏠 ⓘ | 127.0.0.1/TP2-corr/convertisseur_action.php

Rechercher dans la page

Convertisseur Euros/Dollars

2 euros valent 2.2 dollars

Ci-dessous, un exemple d'affichage d'une conversion dans le sens Dollars vers Euros

← → ↻ 🏠 ⓘ | 127.0.0.1/TP2-corr/convertisseur_action.php

Rechercher dans la page

Convertisseur Euros/Dollars

3 dollars valent 2.7272727272727 euros