

Langage PHP

TD N°1

Chaines de caractères, Tableaux, Fonctions, Formulaire

Exercice 1 :

Formatez l’affichage d’une suite de chaînes contenant des noms et prénoms en respectant les critères suivants :

- un prénom et un nom par ligne ;
- chaque mot (nom ou prénom) est affiché sur 20 caractères ;
- toutes les initiales des mots doivent se superposer verticalement ;
- le premier caractère d’un prénom doit être en majuscule ;
- tous les caractères d’un nom doivent être en majuscule ;

Exercice 2 :

- 1- Créez un tableau indicé **Tab1** d’entiers variant de 1 à 50.
- 2- Créez à partir de **Tab1** un deuxième tableau indicé **Tab2** de nombres variant de 0 à 5.
- 3- Créez un troisième tableau associatif **Tab3** dont les clés X varient de 0 à 5 et dont les valeurs sont $\cos(X)$.
- 4- Affichez les valeurs du tableau **Tab3** dans un tableau HTML composé de deux colonnes : la première colonne représente les valeurs de X, tandis que la deuxième colonne représente les valeurs obtenues pour $\cos(X)$. Le tableau HTML affiché doit occuper 60% de la largeur de la fenêtre avec une bordure d’épaisseur égale à 3.

Exercice 3 :

- 1- Créez un fichier **somme.php** où vous définissez la fonction **SOMME** qui retourne la somme de la série de terme général $V_n = a^{3n+5} / n!$. Le résultat retourné doit être arrondi à un nombre d passé en paramètre à la fonction SOMME et définissant le nombre de décimales à afficher.
- 2- Écrivez le script PHP permettant d’afficher le résultat retourné par la fonction **SOMME**.

Exercice 4 :

- 1- Écrivez une fonction **TransTab** qui transforme chacun des éléments d’un tableau de

chaines de caractères passé en paramètre de façon à ce que le premier caractère soit en majuscule et les autres caractères en minuscule quelle que soit leur casse initiale.

- 2- Écrivez le script PHP permettant d'afficher un tableau de chaines de caractères avant et après d'être transformé par la fonction **TransTab**.

Exercice 5 :

- 1- Écrire un formulaire qui permet de choisir un nombre de trois chiffres parmi une liste de choix. Le formulaire fait appel à un script php qui effectue des tirages aléatoires et compte le nombre de tirages nécessaire pour obtenir le nombre choisi initialement. Arrêter les tirages et afficher le nombre de coups réalisés.
- 2- Écrire la fonction `tri_tab_assoc($tab, $mode)` qui permet de trier un tableau associatif (\$tab) selon les clés si la valeur de l'argument \$mode est égale à "keys" et selon les valeurs si \$mode = "values".
- 3- Dans un fichier calcul.php, écrire un formulaire de calculatrice : 2 cases pour la saisie des opérandes, une liste de 5 opérations (+, -, *, /, %). Le formulaire appelle un script php qui affiche l'opération demandée et le résultat.

The diagram shows a calculator form with the following components and annotations:

- Couleur de fond : #00FFCC**: Points to the cyan background of the calculator title bar.
- Titre niveau 2 Police : Arial**: Points to the title "Calculatrice" in the title bar.
- Tableau de bordure 1**: Points to the border of the calculator body.
- Form fields**: Two input boxes for operands, a dropdown menu for operations (showing "+"), and a "Résultat:" label with an output box.
- Buttons**: "Afficher le résultat" and "Annuler" buttons.

Dans la machine à opérations, faites une vérification pour savoir si les 2 chiffres ont bien été remplis par l'utilisateur ; sinon, afficher un message d'erreur. Vérifier aussi que les deux champs contiennent des nombres.

Exercice 6 :

Donner un aperçu du résultat qui sera affiché par le navigateur suite à l'exécution du code PHP suivant :

```
< ? php
    $chaine1= "Paris" ;
    $chaine2="London" ;
    function message ($ch1,$ch2)
    {
```

```
$chaine1 = "Je suis actuellement à $ch1";
$chaine2 = "Rome";
global $variable1;
global $variable2;
$ch2 = "$variable1 je serai $ch2 pour $variable2";
$variable2 = "nnnnnnnnn !";
return array ($chaine1,$ch2);
}
$variable1 = "La semaine prochaine, ";
$variable2 = "présenter un papier dans une conférence internationale";
$montab = message ($chaine1,"à Barcelone");
echo $montab[0]. "<br/>";
echo $chaine1. "<br/>";
echo $montab[1]. "<br/>";
$variable1 = "Dans deux semaines, ";
$variable2 = "les vacances ";
$montab = message ($chaine2,"aux Maldives");
echo $montab[0]. "<br/>";
echo $chaine2. "<br/>";
echo $montab[1]. "<br/>";
echo $variable2;
?>
```

Exercice 7 :

On se propose de réaliser un calculateur de matrice en ligne permettant de calculer pour une matrice carrée d'ordre 2 donnée :

- Son inverse ;
- Son déterminant ;
- Sa trace ;
- Sa transposée ;

La figure 2 présente un aperçu du calculateur en question.

- 1- Ecrire le code HTML permettant de reproduire fidèlement la page web illustrée par la figure 2.a .

N.B : N'oublier pas de préciser clairement les valeurs affectées, respectivement, aux attributs « **METHOD** » et « **ACTION** » de la balise « **FORM** ».

- 2- Ecrire une fonction Javascript nommée **validation ()** permettant de vérifier le remplissage correct du champ texte relatif à la matrice à traiter dans le formulaire avant sa soumission. En cas d'erreur, la fonction doit afficher sous forme de boîte de dialogue un message indiquant à l'utilisateur le type d'erreur.

N.B : il est à noter que lors de la saisie de la matrice à traiter, l'utilisateur doit respecter le format suivant :

Pour une matrice $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$, on doit saisir dans le champ texte correspondant du formulaire : **a ; b ; c ; d ;**

- 3- Ecrire une fonction PHP nommée **conversion(\$chaine)** permettant de convertir la chaîne de caractères représentant la matrice entrée par l'utilisateur (passée en paramètre) en une matrice (i.e tableau multidimensionnel) et retourner le résultat trouvé.
- 4- Ecrire une fonction PHP nommée **inverse_matrice(\$matrice)** permettant de calculer et afficher l'inverse d'une matrice carrée d'ordre 2 passée en paramètre.

N.B : Soit $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ une matrice carrée d'ordre 2. **A** est dite inversible si **ad - bc ≠ 0** et dans ce cas $A^{-1} = \frac{1}{ad-bc} \begin{pmatrix} d & -b \\ -c & a \end{pmatrix}$

- 5- Ecrire une fonction PHP nommée **determinant_matrice(\$matrice)** permettant de calculer et afficher le déterminant d'une matrice carrée d'ordre 2 passée en paramètre.

N.B : Soit $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ une matrice carrée d'ordre 2. Le déterminant de A noté det(A) est égal à **ad - bc**.

- 6- Ecrire une fonction PHP nommée **trace_matrice(\$matrice)** permettant de calculer et afficher la trace d'une matrice carrée d'ordre 2 passée en paramètre.

N.B : Soit $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ une matrice carrée d'ordre 2. La trace de A est égale à **a+d**.

- 7- Ecrire une fonction PHP nommée **transposee_matrice(\$matrice)** permettant de calculer et afficher la transposée d'une matrice carrée d'ordre 2 passée en paramètre.

N.B : Soit $A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}$ une matrice carrée d'ordre 2. La transposée de A notée A^t est égale à $A^t = \begin{pmatrix} a & c \\ b & d \end{pmatrix}$.

- 8- Ecrire le code PHP qui permet de récupérer la matrice entrée par l'utilisateur, faire le calcul nécessaire selon le choix de l'utilisateur suite au click d'un des quatre boutons (Inverse, Déterminant, Trace ou Transposée), et afficher le résultat trouvé (comme illustré par la figure 2.b).

Calculateur de matrice

Matrice :

Figure 2 (a) : Le formulaire dans son état initial

Calculateur de matrice

Matrice :

La trace de la matrice entrée est : 5

Figure 2 (b) : Le résultat affiché suite au click du bouton Trace

Figure 2