

SEANCE 1

Université Mohamed Premier Année universitaire 2017/2018

Faculté des Sciences Filière SMI S6 TP 1 Module « TC WEB Avancée » (PHP)

I. Environnement Apache/PHP/MySQL avec Wamp-Server

- "DocumentRoot" : c'est l'emplacement du site sur le PC (voir ci-dessus) par défaut on a : c:/wamp/www, Vérifiez si Wamp est installé sur votre machine.

-Ecrire un script PHP qui affiche bonjour je suis un programmeur PHP, enregistrez sous C:/wamp/www/tpsmi6-php-2016/index.php ouvrez ce fichier directement avec votre navigateur web. Vos remarques ? OUI Php ne peut pas être interprété par le navigateur.

-Maintenant lancer Wamp ou Easyphp essayez de le découvrir, tapez dans votre navigateur localhost ou 127.0.0.1 dans le navigateur web. PHP est interprété par le serveur PHP

-Ecrire un script Php avec la fonction phpinfo. Essayez de voir la configuration de votre PHP et comparez-la avec le fichier php.ini.

Correction

```
<?php phpinfo(); ?>
```

Exercice1 :(affichage avec html et caractères)

-Ecrire une page HTML en PHP script qui va afficher à l'écran un formulaire, les champs du formulaire sont : Nom ; Prénom ; Date de naissance ; Sexe (checkbox) ; Ville (list selection) ; E-mail ; et Commentaire (text-area).

Correction

Ici l'étudiant doit créer un formulaire en HTML puis l'afficher avec la fonction ECHO de PHP.

-Ecrire un script qui va afficher à l'écran les symboles ♥ (9829 hexa), et ♠ (9812 jusqu'à 9820 hexa) augmenter la taille et changer la couleur de ces symboles avec balises HTML/CSS.

Correction

```
<?php
$monCoeur = "<big>tout les symbole coeur et echecs ; &#9829; &#9812; &#9813 ;
&#9814; &#9815; &#9816; &#9817; &#9818; &#9819; &#9820 </big><br/>";
print $monCoeur ;
$monCoeur = "J'ai un gros Coeur ! : <big><big><big>&#9829;";
print $monCoeur ; //on imprime le contenu de la variable
?>
```

- Déterminer le numéro de version de PHP, le nom du système d'exploitation de votre serveur ainsi que la langue du navigateur du poste client. Ajoutez à ce script pour déterminer l'IP, le DNS, la version du navigateur, l'heure sur le serveur, etc...

```
<?php
```

```

echo "Version de PHP : ",PHP_VERSION, "<br />";
echo "Système d'exploitation du serveur : ",PHP_OS, "<br />";
echo "Langue du navigateur client : ",$_SERVER["HTTP_ACCEPT_LANGUAGE"], "<br />";
?>

```

Correction

Ici l'étudiant a des fonctions qu'il ajoute simplement au script.

Exercice2 :(déclaration variables)

-Ecrire un script PHP qui va déclarer les variables suivantes x, y, z, var1, var2, var3, afficher le résultat avec les 3 fonctions d'affichages de PHP (**echo**, **printf**, **print**) les valeurs de tous ces variables à la fin du script.

x=Oracle ; y= MySQL; z=x; x=PHP 5; y=x; var1 =10 ; var2 assignation par référence à var1, et var3 par valeur de var1.

Correction

```

<?php
$x="PostgreSQL";
$y="MySQL"
;
$z=&$x;
$x="PHP
5";
$y=&$x;
Var1=
?>
L'etudiant va
afficher avec
echo " $x
"<br />";etc...

```

Exercice 3 variable typage)

-Donner la **valeur** de chacune des variables pendant et à la fin du script suivant et vérifier l'évolution du type de ces variables avec la fonction `gettype`:

```
<?php
$x="PHP5";
$a[]=&$x;
$y=" 5 eme version de PHP";
$z=$y*10;
$x=$y;
$y*=$z;
$a[0]="MySQL";
?>
```

Correction

```
<?php
$x="PHP5";
echo "\$x vaut : $x et est de type ", gettype($x), "<br />";
$a[]=&$x;
echo "\$a[0] vaut : $a[0] et est de type ", gettype($a), "<br />";
$y=" 5 eme version de PHP";
echo "\$y vaut : $y et est de type ", gettype($y), "<br />";
```

```
$z=$y*10;
echo "\$z vaut : $z et est de type ", gettype($z), "<br />";
$x=$y;
echo "\$x vaut : $x et est de type ", gettype($x), "<br />";
$y*=$z;
echo "\$y vaut : $y et est de type ", gettype($y), "<br />";
$a[0]="MySQL";
echo "\$a[0] vaut : {$a[0]} et est de type ", gettype($a), "<br />";
?>
```

-Réécrivez le script avec var_dump (\$a);quelles sont vos remarques. Voir l'exemple en bas.

```
<?php $a = 3.14 ; echo '<p>var_dump de $a : ' ; var_dump ($a); ?>
```

Correction

Ici l'étudiant doit remplacer gettype par var_dump (\$a) simplement;

Exercice 4 (formatage chaîne caractères)

-Ecrire un script PHP qui transforme une chaîne écrite dans des casses différentes afin que chaque mot ait une initiale en majuscule les autres en minuscule. Utilisez les fonctions telles que **ucwords** et **strtolower**, la chaîne transformée est la suivante :

« TransFormeZ unE ChaîNE éCRITe dans des cASses diFFéReNTes afin qUe chAQue MOT ait une inITiale en MAJUSCULE"; »

Le résultat doit être comme le suivant :

Transformez Une Chaîne Écrite Dans Des Casses Différentes Afin Que Chaque Mot Ait Une Initiale En Majuscule.

Correction

```
<?php
$ch="TransFormeZ unE ChaîNE éCRITe dans des cASses diFFéReNTes afin
qUe chAQue MOT ait une inITiale en MAJUSCULE";
10
echo ucwords(strtolower($ch));
?>
```

Exercice 5 : expression conditionnelle

Rédiger une expression conditionnelle pour tester si un nombre est à la fois un multiple de 7 et de 5. Testez avec les variables x=35, 7350, 1425.

Correction

```
<?php
$x=7350;
if($x%7==0 AND $x%5==0)
{
echo "$x est multiple de 7 et de 5 <br />";
}
```

```

else
{
echo "$x n'est pas multiple de 7 et de 5 <br />";
}
?>

```

Exercice 6 :

-Créer un tableau dont les indices varient de 11 à 36 et dont les valeurs sont des lettres de A à Z.

Lire ensuite ce tableau avec une boucle **for** voir le code PHP. Pour afficher les indices et les valeurs (la fonction chr(n) retourne le caractère dont le code ASCII vaut n).

```

<?php
for($i=11;$i<=36;$i++)
{
$tab[$i]=chr(54+$i);
}
//Lecture avec for
for($i=11;$i<=36;$i++)
{
echo "Elément d'indice $i : $tab[$i] <br />";
}
?>

```

-Réécrivez ce script avec une boucle foreach .

Correction

```

<?php
for($i=11;$i<=36;$i++)
{
$tab[$i]=chr(54+$i);
}
//Lecture avec for
for($i=11;$i<=36;$i++)
{
echo "Elément d'indice $i : $tab[$i] <br />";
}
echo "<hr />";//Ligne séparatrice
//Lecture avec foreach
foreach($tab as $cle=>$valeur)
{
echo "Elément d'indice $cle : $valeur <br />";
}
?>

```