TP 1 : Produire du HTML et paramétrer un script

PHP est un language de programmation dynamique interprété, à différencier de HTML, un langage de description de pages. Le PHP est souvent utilisé par l'intermédaire d'un serveur WEB

Objectifs du TP:

Ce TP dure 2h

- Comprendre les bases de fonctionnement du language PHP. Développer des scripts simples en PHP.
- Savoir produire du HTML et mélanger HTML avec PHP. Savoir passer des paramètres à un script PHP.
- 1. Démarrer en PHP

1.1 Interpréter le PHP : "Hello world"

pour fabriquer du HTML. Un programme PHP s'exécute donc grâce à un interpréteur. Un interpréteur se trouve dans le fichier /usr/bin/php. Important: dans un fichier PHP, tout ce qui est en dehors de la balise <?php ... ?> est recopié dans le flux de sortie sans modification.

Travail à réaliser :

Se placer dans votre répertoire de travail pour ce TP.

print("Hello world !\n");

Pour tester la syntaxe de ce programme utiliser la commande : php -1 hello.php

3. Exécuter ce programme avec la commande :

php hello.php

4. Découvrez ce que fait l'option -s, et comment en afficher le résultat.

php -s hello.php

Travail optionnel:

www-etu-info.iut2.upmf-grenoble.fr Ce serveur a également été configuré pour automatiquement se dérouter vers le répertoire :

~/public_html d'un utilisateur ayant le login : VotreNomDeLogin, lorsque l'URL demandée débute par :

WEB.

Attention: le serveur Apache doit accéder physiquement aux fichiers de votre répertoire Web pour les afficher. Il faut donc changer les droit d'accès aux pages contenues dans votre répertoire public html. Par défaut, ces fichiers ne sont accessibles qu'à vous-même. Actuellement, la configuration que nous préconisons est d'ajouter uniquement les droits en lecture other à vos fichiers,

plus le droit à other de traverser les répertoires (Voir la commande chmod option o+rX). Pour placer ces droits facilement, nous avons préparé un script, de nom :

qu'il faut exécuter à chaque fois que vous déposez de nouveaux fichiers sur votre répertoire WEB. Une page WEB stockée dans votre répertoire public_html/MonChemin/Mapage.php est ainsi

Notez bien que le répertoire public html ne fait pas partie de l'URL. Conseil: stocker cette adresse dans les bookmarks de votre navigateur.

2.1 PHP dans HTML

// Lieux ou se trouve le TP

<head> <meta charset="utf-8">

<html lang="fr" dir="ltr"> <title>Ma home page</title> </head> <body> <h1>Ceci est ma Home page</h1> > Bonjour vous êtes à l'<?= \$location ?>. Nous sommes un jour <?php if (\$day % 2 == 0) { ?> pair <?php } else { ?> impair <?php } ?> </body> </html> Il commence par du PHP qui initialise 4 variables : o \$location : une chaine de caractère qui décrit l'endroit où vous êtes. \$name : votre nom de login. • \$day : le numéro du jour. Notez la fonction de conversion d'une chaîne vers un entier. Ce type de conversion est implicite en PHP mais il vaut mieux la rendre explicite. \$date: la date d'aujourd'hui dans une chaîne au format Jour/Mois/Année.

?>

impair

2.2 Produire du HTML Le langage PHP est principalement utilisé pour introduire du texte calculé par un programme, au milieu de chaînes de caractère constantes. Dans un fichier PHP, tout ce qui est en dehors de la balise <?php ... ?> est produit en sortie sans modification. Dans cet exercice on vous demande de créer du HTML avec un programme PHP pour produire une table de multiplication. Travail à réaliser :

Complétez le fichier home.php pour qu'il utilise toutes les variables et affiche la date d'aujourd'hui.

6 10 12 14 16 18 20 12 15 18 21 24 27 30 6 9

<?php

else {

if (\$day % 2 == 0) {

print("impair");

print("pair");

12 16 20 24 28 32 36 40 10 15 20 25 30 35 40 45 50

18 27 36 45 54 63 72 81 90 **10** | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 | 90 | 100

| 2.3 Passer des paramètres au serveur : un calculateur HTTP |
|--|
| Depuis un navigateur WEB, on peut passer des paramètres à un programme PHP par 2 méthodes différentes : |
| Méthode GET: les paramètres sont transmis par le client directement dans l'URL (commande "GET" du protocole réseau HTTP). Les paramètres apparaissent alors dans la barre d'adresse
du navigateur. Par exemple: |
| http://MACHINE/CHEMIN/PROG.php?var1=val1&var2=val2 |
| Le serveur récupère ces paramètres et les transfère au programme PROG.php. La partie de l'URL après le caractère ? s'appelle la <u>query string.</u> • Méthode POST : dans cette méthode le client (navigateur) transmet les paramètres directement au serveur dans le corps du message à la suite de l'en-tête HTTP Les paramètres n'apparaissent alors plus dans la barre d'adresse du navigateur. |
| PHP récupère automatiquement les données de la query string encodées dans l'URL. Il place alors toutes les associations nom - valeur de cette query string dans un tableau spécial de nom \$_GET. Ce tableau permet d'accéder directement à une valeur en donnant en indice son nom. Par exemple, si l'URL est la suivante : |
| http://monSite.com/monCode.php?nom=marcel&age=25 |

if (isset(\$ GET['nom'])) {

\$age = \$_GET['age']; // idem avec l'age Attention: la query string n'est pas forcément correcte. Il faut donc tester l'existence de l'indice pour \$_GET avec la fonction isset() avant de l'utiliser, sous peine d'un arrêt du programme, car c'est une erreur en PHP d'accéder à des cases inexistantes d'un tableau. Le bon usage consiste donc à faire un test avant de récupérer la valeur dans une variable, et en l'absence de valeur, produire une erreur, ou placer une valeur par défaut. Exemple :

Si le test est vrai, retourne la valeur de l'expression qui suit le ?, sinon celle qui suit le :;

Alors le tableau \$_GET contient les deux cases suivantes :

On peut utiliser l'opérateur fonctionnel conditionnel ternaire ? : pour raccourcir l'écriture : \$nom = isset(\$_GET['nom']) ? \$_GET['nom'] : "inconnu";

Finalement, on peut encore simpfier cette écriture en utilisant "l'opérateur de coalescence de null" noté ?? qui retourne la valeur de droite si celle de gauche est nulle, sinon celle de gauche.

1. Réalisez le programme calcul.php qui prend trois paramètres dans l'URL : a, b et op et qui réalise le calcul de a op b puis affiche le résultat de cette opération. Par exemple, si l'URL est :

4. Rajoutez des tests d'erreur à votre programme pour qu'il ne fonctionne que si les trois paramètres a, b et op sont donnés. Sinon, une erreur doit être affichée. Pour savoir si une variable ou

alors l'affichage doit être : Calcul

\$nom = \$_GET['nom'] ?? "inconnu";

Travail à faire :

une case d'un tableau existe utilisez la fonction : isset(). Par exemple avec l'URL : tp2/calcul.php?a=10&b=40 votre programme doit afficher :

Utilisez l'instruction switch pour aiguiller le code vers le bon calcul.

3.1 Des abréviations Les tableaux associatifs sont très utilisés dans les langages de scripts car ils permettent de stocker de petites quantités de données de manière simple car indexées par une chaine de caractères. Le langage PHP n'a que des tableaux associatifs. On crée un tableau vide de la manière suivante :

<?= abbrAll() ?>

Travail à faire

\$tab = array();

Calcul

> Le langage <?= abbr('PHP') ?> produit généralement du <?= abbr('HTML') ?>.

à partir de données stockées dans un tableau associatif.

Voici toutes les abréviations connues :

\$personne["adresse"] = "9 rue des écoles";

On crée un tableau initialisé avec la syntaxe suivante :

\$personne = array ('nom' => 'marcel', 'age' => 27);

On ajoute ensuite une entrée au tableau (index plus valeur) de la manière suivante :

Voici toutes les abréviations connues : HTMLHyperText Markeup Language

Dans cet exercice, nous désirons produire automatiquement le détail d'abréviations dans un texte HTML. Nous désirons par exemple que le texte :

// tableau HTML de toutes les abrévations connues function abbrAll() : string { ... }

maintenant reproduire abbr.html.

3.2 Des fuseaux horaires

L'affichage doit ressembler à cela :

Important: Ces deux fonctions ne doivent pas faire d'affichage : pas d' echo , ni de print

function abbr(string \$abbr): string {...}

 Complétez maintenant la partie HTML en y insérant les appels aux fonctions dans des balises racourcies <?= XXX... >

Nous vous proposons de réaliser une page HTML générée par PHP qui affiche l'heure actuelle dans différentes villes du monde. Vous allez devoir parcourir un tableau de villes. Comme les tableaux PHP (array) sont indexés par des chaînes de caractères, il faut une instruction spéciale qui parcourt un tableau : c'est l'instruction foreach. La fonction date default timezone set permet de choisir un fuseau horaire. La fonction date("H:i d/m/Y\n") produit alors une chaîne de caractères indiquant les heures, minutes et la date, en fonction du fuseau horaire choisi. (voir la documentation PHP pour la signification des paramètres).

America/Anchorage 14:31 26/04/2020 America/Los_Angeles 15:31 26/04/2020

'Africa/Kigali', 'Asia/Singapore', 'Australia/Sydney', 'Pacific/Auckland');

Africa/Kigali Asia/Singapore Australia/Sydney

America/Guadeloupe 18:31 26/04/2020 Europe/Paris 00:31 27/04/2020 00:31 27/04/2020 06:31 27/04/2020 08:31 27/04/2020 Pacific/Auckland 10:31 27/04/2020 • Télécharger les fichiers incomplets de ce projet : data/heure monde.zip, puis compléter le fichier PHP pour faire afficher l'heure dans la liste de fuseaux horaires donnée dans le tableau Modifiez la production du header de l'en-tête HTTP produit automatiquement par PHP pour y ajouter la ligne :

L'heure dans le monde

(array) de la variable \$villes.

Travail à faire :

Cela indique que le client doit rafraichir la page au bout de 60 secondes. Cela permet ainsi à la page de donner l'heure à la minute près. Utilisez la fonction header() pour modifier la production du header HTTP. Attention : header fonctionne uniquement si aucun octet n'a encore été produit directement (par PHP, par ex: print), ou indirectement (extérieur des balises PHP) car PHP fabrique le header de la réponse HTTP dès la sortie du premier octet.

Sujet de TP Programmation WEB (PHP).

2. Créer un fichier de nom hello.php contenant les balises de script <?php et ?>, puis l'unique instruction :

1. Modifiez la première ligne de ce fichier pour pouvoir lancer automatiquement le bon interpréteur. Utilisez les caractères "magiques" '#' et '!' appelés shebang, suivi du chemin de l'interpréteur. Changez les droits d'exécution, et tester l'exécution de ce programme avec la même syntaxe que l'exécution d'un script shell ou de n'importe quel programme exécutable. 2. L'interpréteur du serveur WEB Le serveur Apache possède un interpréteur du langage PHP qui est déclenché lorsque le navigateur client demande un fichier se terminant par .php. Ce comportement peut être modifié en changeant le fichier de configuration du serveur Apache. Nous utilisons un serveur Apache qui a été installé sur la machine :

http://www-etu-info.iut2.upmf-grenoble.fr/~VotreNomDeLogin/ Pensez à ajouter cette URL dans vos signets (i.e. bookmarks). À partir de cet exercice, vous devez travailler dans votre répertoire Web public html, et activer vos pages php au travers du serveur

setup-public-html accessible à travers l'URL: http://www-etu-info.iut2.upmf-grenoble.fr/~VotreNomDeLogin/MonChemin/Mapage.php

PHP a été conçu pour s'insérer dans des pages HTML. Pour insérer dans le HTML, une donnée contenue dans une variable PHP, il faut la sortir à travers le "flux standard de sortie" par une

fonction comme print, printf ou echo. On peut également utiliser la balise raccourcie <?= ...exp... ?> qui évalue l'expression et remplace cette balise par le résultat de l'expression. Par exemple: Le carré de 123 est égal à <?= 123*123 ?>

Produit la sortie : Le carré de 123 est égal à 15129 Travail à réaliser

 Récupérez le fichier home.php dans l'archive : data/home.zip Examinez le contenu du fichier : <?php \$location = "IUT2 de Grenoble"; // Numéro du jour, la fonction intval convertit la chaine en entier \$day = intval(date("d")); // Reste de la date \$date = date("d/m/Y"); <!DOCTYPE html>

Notez la structure de la condition qui a été découpée pour sortir soit 'pair' soit 'impair'. On aurait pu également écrire un seul bloc de PHP :

<?php if (\$day % 2 == 0) : ?> pair <?php else : ?> <?php endif; ?>

Il existe également une syntaxe alternatives des structures qui peut rendre des imbrications de code PHP avec du HTML plus clair :

• Créer un programme tableMult.php qui produit une page HTML correcte contenant une table HTML des tables de multiplications entre 1 et 10. Utilisez la possibilité de placer des styles dans la page avec la balise <style> dans la balise <head>. Vous devez obtenir le résultat suivant : Tables de multiplication

12 18 24 30 36 42 48 54 60 6 6 14 21 28 35 42 49 56 63 70 16 24 32 40 48 56 64 72 80

\$nom = \$_GET['nom']; } else { \$nom = "inconnu";

.../calcul.php?a=10&b=40&op=-10 - 40 = -30

\$nom = \$_GET['nom']; // On récupère la valeur du nom de la query string dans la variable \$nom

Operator op missing Bravo! Vous avez réalisé un programme PHP qui produit une page HTML différente en fonction de paramètres ajoutés dans l'URL. 3. Structures de données : les tableaux (array)

2. Constatez que l'opérateur + ne fonctionne pas. C'est parce que, dans la syntaxe des URL, le symbole + a une signification particulière : laquelle ?

3. Regardez sur le WEB (ex: sur http://www.w3schools.com/tags/ref_urlencode.asp) comment encoder le caractère + et faire marcher votre programme.

Soit transformé automatiquement en : Le langage <abbr title="Hypertext PreProcessor">PHP</abbr>

produit généralement du <abbr title="HyperText Markeup Language">HTML</abbr>.

• Téléchargez le fichier html : abbr.html. Examinez le fonctionnement de la balise abbr. À quoi sert-elle ? • Créez le fichier abbr.php qui va être capable de recréer la page HTML contenue dans abbr.html mais par un programme en PHP, c'est-à-dire en puisant les informations sur les abréviations dans un tableau (array). Ce tableau associatif global doit être indexé par des abrévations et contenir leur forme développée. • Structurez votre fichier PHP de la manière suivante : une première partie en PHP qui va contenir des définitions : le tableau global et deux fonctions (voir ci-dessous), et ensuite une deuxième partie qui reproduit la structure fixe du HTML. Cette partie en PHP 'pur' doit se trouver en début du fichier, car elle va contenir des définitions (i.e. tableau et fonctions) qui seront ensuite utilisées dans la partie HTML qui suit (on parlera plus tard de "vue en HTML"). Utilisez maintenant le tableau associatif dans les fonctions suivantes :

// Produit en sortie (i.e. return) une chaîne avec balise abbr et le détail

// de l'abréviation si elle existe sinon, retourne la valeur \$abbr

// Produit en sortie (i.e. return) une chaine contenant un

, ou de printf Attention: Vous êtes obligé d'indiquer, dans chaque fonction, que le tableau est global pour que les deux fonctions puissent y avoir accès.

. Cette notation signifie que le résultat de l'évaluation de l'expression "XXX..." remplace la balise, sans explicitement utiliser ni echo, ni print, ni printf, etc. Le fichier abbr.php doit alors

NB: en programmation, l'usage de variables globales reste déconseillé. Une meilleure solution serait alors de créer une classe Abbr avec deux méthodes.

Nous vous proposons d'afficher l'heure dans les villes suivantes (fuseau horaire): \$villes = array('America/Anchorage', 'America/Los_Angeles', 'America/Guadeloupe', 'Europe/Paris',

refresh: 60