AFLOKKAT
Conseil et Formation

Programmation WEB avec PHP

Matière : Programmation WEB avec PHP

Date : 13.06.2022

Formateur : Malerba Sylvain

Auteur : Malerba Sylvain

SOMMAIRE

AFLOKKAT
Conseil et Formation

01 Inclusion de fichiers

02 Manipulation des fichiers

03 Les dates

04 Les expressions régulières

AFLOKKAT
Conseil et Formation

01 Inclusion de fichiers

01 Inclusion de fichiers

AFLOKKAT
Conseil et Formation

Il existe 4 types d'inclusion de fichiers

- **include**: inclut le code d'un fichier. Si inexistant, erreur Warning (non bloquante
- **require:** inclut le code d'un fichier. Si inexistant, erreur Fatal (bloquante)
- **include_once**: inclut le code d'un fichier une seule fois. Si inexistant, erreur Warning (non bloquante)
- **require_once**: inclut le code d'un fichier une seule fois. Si inexistant, erreur Fatal (bloquante)

01 Exercice

- A partir de l'exercice précédent, déplacez les fonctions is_logged() et la fonction d'affichage du login dans un autre fichier PHP
- Importez ce fichier dans votre exercice

01 Solution

AFLOKKAT
Conseil et Formation

```php
<?php session_start();

  require_once("exo12bis.php");
 ?>
```

```php
<?php

function is_logged() {

    return isset($_SESSION["login"]);

}

function get_login() {

    if (is_logged()) {

      return $_SESSION["login"];

    }

}
?>
```

AFLOKKAT
Conseil et Formation

02 Manipulation des fichiers

02

Manipulation des fichiers (Lecture)

AFLOKKAT
Conseil et Formation

- **Fonction file_get_contents():** retourne le contenu du fichier dans une chaîne de caractères
- **Fonction nl2br():** formate les sauts de lignes

02 Manipulation des fichiers (Exemple)

AFLOKKAT
Conseil et Formation

Peugeot Renault BMW Audi
Peugeot Renault BMW Audi
Peugeot
Renault
BMW
Audi

```php
<?php

  $voiture = file_get_contents("voiture.txt");

  $voiture2 = file_get_contents("C:/wamp64/www/php/voiture.txt");

  echo $voiture;

  echo "<br>";

  echo $voiture2;

  echo "<br>";

  echo nl2br($voiture2);
?>
```

02

Manipulation des fichiers (Fonctions)

AFLOKKAT
Conseil et Formation

- **Fonction fopen():** ouvre un fichier et retourne une ressource
- **Fonction fread() :** renvoie le contenu d'un fichier jusqu'à un nombre d'octets
- **Fonction fgets() :** lit un fichier ligne par ligne
- **Fonction feof() :** indique si la fin du fichier a été atteinte
- **Fonction fclose() :** ferme la ressource du fichier

02 Manipulation des fichiers (Exemple)

AFLOKKAT
Conseil et Formation

```php
<?php
   $voiture = fopen('voiture.txt', 'r');

   while (!feof($voiture)) {

       $ligne = fgets($voiture);

       echo 'La ligne "' . $ligne . '" contient ' . strlen($ligne) . ' caractères <br><br>';

   }

  fclose($voiture);
?>
```

La ligne "Liste des vehicules: " contient 22 caractères

La ligne "Peugeot " contient 9 caractères

La ligne "Renault " contient 9 caractères

La ligne "BMW " contient 5 caractères

La ligne "Audi" contient 4 caractères

02

Manipulation des fichiers (Ecriture)

AFLOKKAT
Conseil et Formation

- **Fonction file_put_contents():** écrit une chaîne de caractères dans un fichier
- **L'argument FILE_APPEND:** permet de ne pas écraser, et d'écrire à la suite

02 Manipulation des fichiers (Exemple)

AFLOKKAT
Conseil et Formation

```php
<?php
   $file_path = "voiture.txt";

   file_put_contents($file_path, "\n SEAT", FILE_APPEND);

   file_put_contents($file_path, "\n TESLA", FILE_APPEND);

   echo "Le contenu du fichier est:<br>";

   echo nl2br(file_get_contents($file_path));
?>
```

Le contenu du fichier est:
Liste des vehicules:
Peugeot
Renault
BMW
Audi
SEAT
TESLA

02

Manipulation des fichiers (Autres fonctions)

AFLOKKAT
Conseil et Formation

- **Fonction filesize():** retourne la taille du fichier en octets
- **Fonction is_file():** indique si un fichier existe
- **Fonction file_exists():** indique si un fichier ou un répertoire existe
- **Fonction rename() :** renomme un fichier
- **Fonction unlink() :** supprime un fichier

02

Manipulation des fichiers (Autres fonctions)

AFLOKKAT
Conseil et Formation

- **Fonction mkdir():** crée un répertoire
- **Fonction rmdir() :** supprime un répertoire vide
- **Fonction opendir() :** ouvre un répertoire et renvoie une ressource
- **Fonction readdir() :** lit le contenu d'un répertoire fichier par fichier
- **Fonction closedir() :** ferme la ressource d'un répertoire

ATTENTION à la notion de droit

02

Rappel sur les droits (mode symbolique)

AFLOKKAT
Conseil et Formation

Lettres pour les droits d‘accès	Signification
r	Droit de lecture
w	Droit d‘écriture
x	Droit d‘exécution

02

Rappel sur les droits (mode symbolique)

AFLOKKAT
Conseil et Formation

Lettres pour les catégories d‘utilisateurs	Signification
u	Catégorie d‘utilisateur « user », propriétaire
g	Catégorie d‘utilisateur « group », groupe
o	Catégorie d‘utilisateur « others », autres utilisateurs
a	« all » : la commande concerne toutes les catégories d’utilisateurs.

02

Rappel sur les droits (mode symbolique)

AFLOKKAT
Conseil et Formation

Opérateurs	Signification
+	Attribution de droits
-	Suppression de droits

02 Questions

AFLOKKAT
Conseil et Formation

Que font les commandes suivantes ?

- chmod ugo+rw exemple.txt
- chmod a+rw exemple.txt
- chmod g-wx exemple.txt

02

Rappel sur les droits (mode octal)

AFLOKKAT
Conseil et Formation

Il s'agit d'une séquence sur 3 chiffres:

Position des chiffres selon la catégorie d'utilisateur	Signification
1	Catégorie d'utilisateur « user », propriétaire
2	Catégorie d'utilisateur « group », groupe
3	Catégorie d'utilisateur « others », autres utilisateurs

02

Rappel sur les droits (mode octal)

AFLOKKAT
Conseil et Formation

Lettres pour les catégories d'utilisateurs	Signification
4	Lecture
2	Ecriture
1	Exécution
0	Aucun droit

02 Questions

AFLOKKAT
Conseil et Formation

Que font les commandes suivantes ?

- chmod 640 exemple.txt
- chmod 777 exemple.txt
- chmod 000 exemple.txt

AFLOKKAT
Conseil et Formation

03 Les dates

03

Les dates (Fonctions)

AFLOKKAT
Conseil et Formation

- **Fonction time():** retourne un timestamp (nombre de secondes depuis 01/01/1970) ex : 1604153923
- **Fonction date() :** retourne une date sous forme de chaîne de caractères selon un format https://www.php.net/manual/fr/datetime.format.php
- **Fonction strtotime() :** convertit une chaîne de caractères (en anglais) en timestamp
- **Fonction date_default_timezone_set() :** définit un fuseau horaire

03

Les dates (Définir timezone)

AFLOKKAT
Conseil et Formation

- **Définir une timezone dans le fichier php.ini**

```
[Date]
; Defines the default timezone used by the date functions
; http://php.net/date.timezone
date.timezone ="Europe/Paris"
```

03

Exercice

AFLOKKAT
Conseil et Formation

- **A partir de l'exercice précédent, créez un fichier de log**

  - Date de l'action

  - Nom de l'utilisateur

  - Action réalisée (Connexion / Déconnexion)

03 Solution

AFLOKKAT
Conseil et Formation

```php
if (isset($_COOKIE["user"])) {

   $tab = explode("|", $_COOKIE["user"]);

   if (password_verify($password, $tab[1])) {

     $_SESSION["login"] = $tab[0];

     $log_path = "log.txt";

     $content = date("d-m-Y H:i:s")." l'utilisateur : ".$_SESSION["login"]." s'est connecté";

     file_put_contents($log_path, "\n".$content, FILE_APPEND);

   }
}
```

```
15-06-2022 11:33:01 l'utilisateur : a@a.fr s'est connecté
15-06-2022 11:33:01 l'utilisateur : a@a.fr s'est déconnecté
```

```php
if (isset($_GET["action"]) && $_GET["action"] == "logout") {

  $log_path = "log.txt";

  $content = date("d-m-Y H:i:s")." l'utilisateur : ".$_SESSION["login"]." s'est déconnecté ";

  file_put_contents($log_path, "\n".$content, FILE_APPEND);

  session_destroy();
  setcookie("user", "", 0);
  unset($_SESSION);
  unset($_COOKIE);
```

AFLOKKAT
Conseil et Formation

04 Expressions régulières

04

Expressions régulières

AFLOKKAT
Conseil et Formation

- **Lien utile : https://regexr.com/**
- **Une expression régulière (aussi abrégé en « regex ») est une séquence de caractères qu'on va définir et qui va nous servir de schéma de recherche**
- **Utilisation d'un "masque" ou "pattern" à rechercher**
- **Utilisée pour filtrer et vérifier la validité des données (email, mot de passe,nom etc..)**

04

Expressions régulières (Fonctions)

AFLOKKAT
Conseil et Formation

- **Fonctions de comparaison**

  - preg_match(): compare une regex à une chaîne de caractères et renvoie 1 si la chaîne correspond au masque

  - preg_match_all(): compare une regex à une chaîne de caractères et renvoie sous forme de tableau les résultats trouvés

Fonctions de remplacement

- preg_replace(): recherche une regex et remplace dans une chaîne

- preg_replace_callback(): recherche une regex et remplace dans une chaîne en utilisant une fonction de rappel

04

Expressions régulières (Classes de caractères)

AFLOKKAT
Conseil et Formation

- **Permet de fournir différents choix de correspondance défini []**

Métacaractère	Description
\	Caractère de protection
^	Si placé au tout début d'une classe, permet de nier la classe c'est-à-dire de chercher tout caractère qui n'appartient pas à la classe.
-	Entre deux caractères, permet d'indiquer un intervalle de caractères.

04

Expressions régulières (Classes abrégées)

AFLOKKAT
Conseil et Formation

- **Permet de fournir différents choix de correspondance défini []**

Classe abrégée	Description
\w	Représente tout caractère de « mot ». Équivalent à [a-zA-Z0-9_]
\W	Représente tout caractère qui n'est pas un caractère de « mot ». Équivalent à [^a-zA-Z0-9_]
\d	Représente un chiffre. Équivalent à [0-9]
\D	Représente tout caractère qui n'est pas un chiffre. Équivalent à [^0-9]
\s	Représente un caractère blanc (espace, retour chariot ou retour à la ligne)
\S	Représente tout caractère qui n'est pas un caractère blanc
\h	Représente un espace horizontal
\H	Représente tout caractère qui n'est pas un espace horizontal
\v	Représente un espace vertical
\V	Représente tout caractère qui n'est pas un espace vertical

04 Expressions régulières (Exemple)

```php
$masque = '/[^n]di/';
$chaine = "lundi, mardi, mercredi, jeudi, vendredi, dimanche";
preg_match_all($masque, $chaine, $out);
echo "<pre>";
print_r($out);
echo "</pre>";
$masque = '/[a-r]di/';
preg_match_all($masque, $chaine, $out);
echo "<pre>";
print_r($out);
echo "</pre>";
$masque = '/[a\-r]di/';
preg_match_all($masque, $chaine, $out);
echo "<pre>";
print_r($out);
echo "</pre>"
```

```
Array
(
    [0] => Array
        (
            [0] => rdi
            [1] => edi
            [2] => udi
            [3] => edi
            [4] =>  di
        )

)
Array
(
    [0] => Array
        (
            [0] => ndi
            [1] => rdi
            [2] => edi
            [3] => edi
        )

)
Array
(
    [0] => Array
        (
            [0] => rdi
        )

)
```

04

Expressions régulières (Ancres)

AFLOKKAT
Conseil et Formation

- **Le point « . » : n'importe quel caractère (sauf le saut de ligne)**
- **Le pipe | : sépare les alternatives**

```php
$masque1 = '/./';   $chaine = 'Salut';
preg_match_all($masque1, $chaine, $tb1);
echo '<pre>';  print_r($tb1); echo '</pre>';

$masque2 = '/es|gu|ci/';
$chaine2 = 'Ceci est un cours sur les expressions régulières';
preg_match_all($masque2, $chaine2, $tb2);
echo '<pre>';  print_r($tb2); echo '</pre>';
```

```
Array
(
    [0] => Array
        (
            [0] => S
            [1] => a
            [2] => l
            [3] => u
            [4] => t
        )

)

Array
(
    [0] => Array
        (
            [0] => ci
            [1] => es
            [2] => es
            [3] => es
            [4] => gu
            [5] => es
        )

)
```

04

Expressions régulières (Quantificateurs)

AFLOKKAT
Conseil et Formation

Quantificateur	Description
a{X}	On veut une séquence de X « a »
a{X,Y}	On veut une séquence de X à Y fois « a »
a{X,}	On veut une séquence d'au moins X fois « a » sans limite supérieure
a?	On veut 0 ou 1 « a ». Équivalent à a{0,1}
a+	On veut au moins un « a ». Équivalent à a{1,}
a*	On veut 0, 1 ou plusieurs « a ». Équivalent à a{0,}

04 Expressions régulières (Exemple)

AFLOKKAT
Conseil et Formation

```
$masque1 = '/^[A-Z].{6,},/';  // commence par une majuscule, contient au moins 6 caractères suivi d'une virgule
$masque2 = '/\d{2,2}[a-z\s]+!$/'; // contient 2 chiffres suivi de lettres minuscules ou d'espaces, et finit par un point d'exclamation
```