

Internet et Outils

L1/IO2 2007- 2008

S2-IO2

PHP suite

François Armand

armand@informatique.univ-paris-diderot.fr

Plan Général Prévisionnel (1/2)

(non contractuel)

- Cours Internet et Outils:
 - [1/12] Intro, Internet, Web, XHTML (2H)
 - [2/12] XHTML(2H)
 - [3/12] CSS (2H)
 - [4/12] PHP (2H)
 - [5/12] **PHP suite (2H)**
 - [6/12] Introduction MySQL, Table simple (2H)

• [7/12] ► **Partiel (2H) 15 Mars 08 14H-16H** ◀
Amphis 2A et 8C

Plan Général Prévisionnel (2/2)

(non contractuel)

- Bases de données
 - [8/12] Tables multiples, Schémas (2H)
 - [9/12] Concepts un peu plus avancés (2H)
 - [10/12] Introduction aux réseaux(2H)
 - [11/12] Réseaux: compléments (2H)
 - [12/12] Révisions (2H)

PHP: Plan

- **Ce que vous allez découvrir**
- Rappels
- Tableaux, Constantes
- Erreurs, Objets
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- Transfert fichier vers client
- Ce qu'il faut retenir

Ce que vous allez découvrir

- Comment stocker / mettre à jour des informations reçues depuis un formulaire
- Comment transférer un fichier depuis le site du client sur le serveur
- Comment proposer à un utilisateur une liste de fichiers présents sur le serveur
- Comment transférer un fichier depuis un serveur sur la machine de l'utilisateur

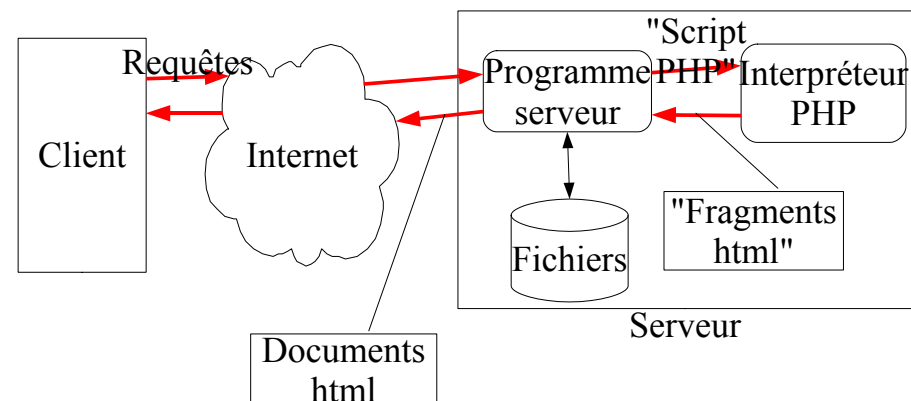
PHP: Plan

- Ce que vous allez découvrir
- **Rappels**
- Tableaux, Constantes
- Erreurs, Objets
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- Transfert fichier vers client
- Ce qu'il faut retenir

Rappels: PHP

- Langage de "programmation" proche du C
- Langage de script
 - interprète des commandes "built-in" ou pas
 - de nombreux modules d'interface avec d'autres outils dont évidemment MySQL
- S'utilise et s'exécute côté serveur pour produire "aisément" un document [x]html à renvoyer au client
- Imbriqué avec le "document html"
- Renommer le fichier ".html" en ".php"

Rappels: Architecture CGI



Rappels: Rôle du serveur

- Avec du [x]html, le serveur se "contente" de trouver le fichier contenant le document correspondant à l'URL et de le renvoyer
- Avec PHP, le serveur doit d'abord analyser les documents pour:
 - trouver toutes les occurrences de `<?php.... ?>`
 - faire interpréter les commandes php par un interpréteur php
 - Substituer les chaînes `<?php... ?>` par le résultat de l'interprétation

Rappels : exemple

```
<?php
if (strpos($_SERVER['HTTP_USER_AGENT'], 'MSIE') !==
    FALSE) { ?>
    <h3>strpos() n'a pas retourné FALSE</h3>
    <p>Vous utilisez Internet Explorer</p>
<?php } else { ?>
    <h3>strpos() a retourné FALSE</h3>
    <p>Vous n'utilisez pas Internet Explorer</p>
<?php } ?>
```

Rappels PHP

- Commentaires:
 - `/*commentaire */`
 - `// commentaire`
 - `# commentaire`
- Noms des variables :
 - Composés de lettres, chiffres, `_` ; Longueur quelconque
 - Commencent par `$`, Lettre ou `_` après le `$`
 - Majuscules et Minuscules ne sont pas identiques

Rappels: variables

- En PHP une variable peut-être:
 - définie avec une valeur, `$var="toto";`
 - définie et vide, `$var=""`;
 - non définie
- Types des variables:
 - Entier: `$note=1728; // Bravo c'est excellent!`
 - Nombre à virgule flottante: `$temp=37.2; // Le matin`
 - Chaîne de caractères:
 - `$str ="1780" //chaîne pas entier!`
 - Booléen: `$condition=TRUE; $condition=FALSE;`

Rappels: chaînes

- Comprises entre
 - Guillemets: "abcd \" ef ' g \$var "
 - => substitution de \$var par sa valeur
 - Apostrophes: 'abcd \" ef " g \$var '
 - => pas de substitution de \$var
- Concaténées par .
 - "abc"."abc" => "abcabc"

Rappels: Contrôle d'exécution

- **if** (cond) {bloc;} **else**{ bloc;}
- **for** (initialisation; condition; fin d'itération) {bloc;}
- **while** (condition) { bloc; }
- **do** { bloc; } **while** (condition);
- **switch** (\$var) { **case** Valeur1: bloc; **break**;
 case Valeur2: bloc; **break**;
 default: bloc;
 }

Rappels: Fonction

- Définition de fonction

```
function verif_champ($valeur, $champ, $liste)
{
    if (empty($valeur)) {
        if (!empty($liste)){
            $liste = $liste.", ";
        }
        $liste = $liste.$champ;
    }
    return $liste;
}
```

- Appel (utilisation) de la fonction:

```
$liste = verif_champ($email, "E-mail", $ChampsManquants);
$liste = verif_champ($pays, "Pays", $ChampsManquants);
```

Rappels: Global

```
function verif_champ($valeur, $champ)
{
    global $ChampsManquants;
    if (empty($valeur)) {
        if (!empty($ChampsManquants)){
            $ChampsManquants = $ChampsManquants.",
";
        }
        $ChampsManquants =
$ChampsManquants.$champ;
    }
}
```

Vérifier le contenu d'une variable

- `isset($var)` // Vrai si définie, même vide
- `empty($var)` // Vrai si vide, 0, ou pas définie
- `is_int($var)` // Vrai si type entier
- `is_float($var)`
- `is_numeric($str)` // Vrai si chaîne numérique: 0..9
- `is_string($str)`
- `ereg("motif", $chaine)`

Chaînes

- `strtolower($str)`
 - Renvoie la chaîne \$str après avoir converti toutes les majuscules en minuscules
 - `strtolower("AH! Je ris Me VoIR Si BELLE");`
 - => ah! je ris me voir si belle
- `strtoupper($str)`
 - Renvoie la chaîne \$str après avoir converti toutes les minuscules en majuscules
 - `strtoupper("AH! Je ris Me VoIR Si BELLE");`
 - => AH! JE RIS ME VOIR SI BELLE

Date

- `date(format)`
 - `spec[- / .]spec...`
 - Les spécificateurs sont séparés par – ou / ou .
 - Spécificateurs: (liste non exhaustive)
 - M abréviation mois en anglais
 - F mois en toute lettre en anglais
 - d jour dans le mois
 - l jour de la semaine en lettres en anglais
 - H heures entre 0 et 24
 - h heures entre 0 et 12
 - i minutes
 - s secondes...

PHP: Plan

- Ce que vous allez découvrir
- Rappels
- **Tableaux**, Constantes
- Erreurs, Objets
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- Transfert fichier vers client
- Ce qu'il faut retenir

Tableaux : Impression

- `print_r($Livre);`
Array ([0] => Les Misérables
[1] => Harry Potter
[2] => Le seigneur des anneaux)
- `var_dump($Livre);`
array(3) { [0]=> string(14) "Les Misérables"
[1]=> string(12) "Harry Potter"
[2]=> string(23) "Le seigneur des anneaux" }

François Armand

21

Tableaux: Modifier

- `$Livre[1]="";`
- `print_r($Livre);`
Array ([0] => Les Misérables
[1] =>
[2] => Le seigneur des anneaux)
- `unset($Livre[1]); print_r($Livre);`
Array ([0] => Les Misérables
[2] => Le seigneur des anneaux)

François Armand

22

Tableaux: Trier

- `sort($Livre); var_dump($Livre);`
array(3) {
[1]=> string(12) "Harry Potter"
[2]=> string(23) "Le seigneur des anneaux"
[0]=> string(14) "Les Misérables" }
- `asort, ksort, rsort, arsort, krsort`

François Armand

23

Tableaux

- Créer avec "index":
 - `$Films[0]="Shrek";`
 - `$Films[1]="Shrek 2";`
 - Pas d'obligation de commencer à 0
- Créer avec "clefs":
 - `$capitale['FR']="Paris";`
 - `$capitale['UK']="Londres";`
 - `$capitale['DE']="Berlin";`

François Armand

24

Tableaux

- Créer sans index explicite:
 - `$Livre[]="Les Misérables";`
 - `$Livre[]="Harry Potter";`
 - `$Livre[]="Le seigneur des anneaux";`
- `echo "Livre[0]";`
Les Misérables
- `$Livre= array ("Les Misérables", "Harry Potter", "Le seigneur des anneaux");`

Tableaux

- `$rue= array(4 => "Rue de Lapp", "Rue de la Paix");`
- `echo $rue[5]`
 - Rue de la Paix
- `$annees= range(2001, 2038);`
- On peut aussi créer des tableaux multi-dimensionnels
 - `$Notes['Jules'] ['Exo 1']; $Notes['Jules'] ['Exo 2'];`
 - `$Notes['Jim'] ['Exo 1']; $Notes['Jim'] ['Exo 2'];`

Tableau: Parcourir

```
foreach($Livre as $index => $Titre)
{
    echo "Livre Numéro ",$index," : $Titre<br/>";
};
```

Livre Numéro 0 : Les Misérables

Livre Numéro 1 : Harry Potter

Livre Numéro 2 : Le seigneur des anneaux

Tableau: Parcourir

- Manuellement:
 - `current($tableau);`
 - `next($tableau); previous($tableau)`
 - `end($tableau); reset($tableau)`
- Exemple:


```
reset($Livre);
echo "current($Livre)"; //=> Les Misérables
next($Livre);
echo "current($Livre)"; // => Harry Potter
```

Tableau

- Taille:
 - \$size= `count($Livre);`
 - \$size= `sizeof($Livre);`
- On peut aussi:
 - Convertir un tableau en chaîne et réciproquement
 - \$Tableau=`explode("x", $str)` // x est un séparateur
 - \$str=`implode("x",$Tableau);`

Tableau

- Comparer:
 - \$diff=`array_diff($Tableau1, $Tableau2);`
- Sommer:
 - \$tab=`array(3,3,3);`
 - `array_somme($tab) //=> 9`
- Supprimer les éléments redondants:
 - \$tab=`array(Moi, Toi, Moi, Lui, Moi, Nous, Moi);`
 - \$tab_simple=`array_unique($tab)`
 - // => \$tab_simple=`array(Moi, Toi, Lui, Nous);`

Tableaux prédéfinis

- **\$ _REQUEST** (et /ou **\$ _POST \$ _GET**)
 - Contient les variables renvoyées par POST ou GET depuis un formulaire HTML
 - `<input type="text" size="20" name="prenom" />`
 - paramètre pour PHP: `$ _REQUEST['prenom']`
- **\$ _SERVER**:
 - Informations relatives au serveur
 - Ex : `$ _SERVER["HTTP_USER_AGENT"]`
 - Ex: `$ _SERVER["SERVER_NAME"]`

PHP: Plan

- Ce que vous allez découvrir
- Rappels
- Tableaux, **Constantes**
- Erreurs, Objets
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- Transfert fichier vers client
- Ce qu'il faut retenir

Constantes

- `define("non_cste","valeur_cste");`
 - `define("US_TO_EUR",1.22);`
 - `define("TITRE","COURS L1/IO2");`
- Ne doivent pas correspondre aux mots-clés de PHP

```
echo "Le cours: ",TITRE," a lieu le Vendredi<br />";
echo "5 Dollars valent: ", 5*US_TO_EUR,"Euros";
    Le cours: COURS L1/IO2 a lieu le Vendredi
    5 Dollars valent: 6.1 Euros!
```

François Armand

33

Constantes, Variables: exemple

```
<?php define("TITRE", "Cours L1/IO2");
define("US_TO_EUR", 1.22);
$Mes_dollars=5; $Tes_dollars="";
?>
```

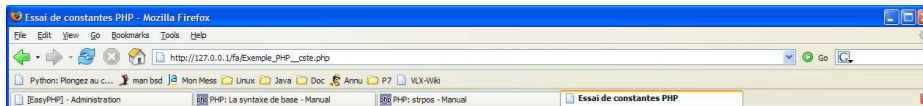
<p> Utilisons les constantes definies ci-dessus</p>

```
<?php
echo "Ce ", TITRE, " a lieu chaque semaine.<br />";
echo "Mes ", $Mes_dollars, " dollars valent ";
echo $Mes_dollars * US_TO_EUR, " euros.<br />";
echo "Tes ", $Tes_dollars, " dollars valent ";
echo $Tes_dollars * US_TO_EUR, " euros.<br />";
unset($Mes_dollars);
echo "Il me reste ", $Mes_dollars, " dollars.";
```

François Armand

34

Constantes, Variables: résultat



Un essai d'utilisation de constantes

Utilisons les constantes définies ci-dessus

```
Ce Cours L1/IO2 a lieu chaque semaine.
Mes 5 dollars valent 6.1 euros.
Tes dollars valent 0 euros.
Il me reste
```

Notice: Undefined variable: Mes_dollars in c:\documents and settings\francois\desktop\francois\p7\paris7\11_06_07\myeasyphp\exemple_php__cste.php on line 19 dollars.

- variable vide =>chaîne vide ou entier nul
- `$Mes_dollars` est non définie après `unset`

François Armand

35

Constantes prédéfinies

- `__FILE__` : nom du fichier "en cours"
- `__LINE__` : ligne courante dans le fichier

```
echo "<br /> Je suis à ",__FILE__," : ",__LINE__"
```

Je suis à ex5.php : 30

Exemple, ([php](#))

François Armand

36

PHP: Plan

- Ce que vous allez découvrir
- Rappels
- Tableaux, Constantes
- Erreurs, Objets
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- Transfert fichier vers client
- Ce qu'il faut retenir

François Armand

37

Modifier le niveau d'erreurs indiquées

- En début de script PHP:
 - `error_reporting(OPTIONS);`
- Options:
 - 0: Pas de message d'erreur affiché (à utiliser seulement quand le script fonctionne et est publié sur un site, pour ne pas déranger l'utilisateur)
 - `E_ALL`: Toutes les erreurs
 - Rien: renvoie le niveau courant

François Armand

38

Exemple erreurs et commentaires

```
<?php
```

```
echo "Niveau Erreur Courant:", error_reporting();
error_reporting(E_ALL); //Je veux tous les details!
echo "<br />Niveau Erreur Courant:",error_reporting();
echo "Nous sommes le: " . Date("d / m / y"); # La date
/* commentaire sur
plusieurs lignes */
```

Il manque une virgule!

```
?>
```

- Niveaux d'erreurs:
 - `E_ERROR` | `E_WARNING` | `E_PARSE`....

François Armand

39

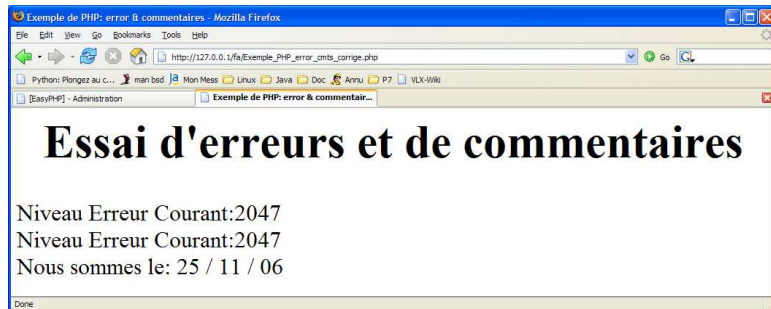
Erreurs reportées



François Armand

40

Après correction (php)



François Armand

41

PHP: Plan

- Ce que vous allez découvrir
- Rappels
- Tableaux, Constantes
- Erreurs, **Objets**
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- Transfert fichier vers client
- Ce qu'il faut retenir

François Armand

42

Classe

- PHP offre une notion d'objet (PHP 4 et suivants)
- Nous en ferons une utilisation extrêmement limitée pour nous simplifier quelques problèmes, nous ne ferons pas de "programmation objet".
- Nos besoins:
 - Regrouper des variables: tous les champs saisis dans un formulaire.
 - On pourra ainsi plus facilement les stocker dans un (seul) tableau et envisager de stocker ce tableau dans un fichier. Les informations pourront ainsi survivre entre deux sessions, 2 démarrages système

François Armand

43

Classe: Exemple

```

class Utilisateur {
    var $Nom, $Prenom,
        $Adresse, $Email,
        $Tel, $Pays,
        $Pseudo,
        $Password, // Pas besoin du double
        $Age;

} ;

```

-

François Armand

44

Tableau d'objets

- On peut ensuite manipuler ces objets comme des variables, et donc les ranger dans un tableau
- Clé utilisée: Le pseudo saisi.
- **\$TousLesUsers**[\$user->Pseudo] = \$user;
 - Equivalent à
 - **\$TousLesUsers**['LeProf'] = \$user;
 - Pour \$user->Nom='LeProf'

François Armand

45

PHP: Plan

- Ce que vous allez découvrir
- Rappels
- Tableaux, Constantes
- Erreurs, Objets
- **Exemple fonction**
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- Transfert fichier vers client
- Ce qu'il faut retenir

François Armand

46

Formulaire

Exemple simple de formulaire et de script php - Mozilla Firefox

Un formulaire

Les champs écrits en rouge doivent être obligatoirement renseignés.

Nom Votre nom Prénom

Adresse

E-mail Téléphone

PAYS FR UK US BE

Pseudo:

Mot de passe Confirmez le

Votre age

0-3 ans
3-12 ans
12-18 ans
18-29 ans
29-55 ans

François Armand

47

Vérification du formulaire

- Définition des variables
defini ("ENVOI", "Envoi!");
\$user = new Utilisateur();
- Pour chacune des valeurs du formulaire:
if (isset(\$_GET['nom'])) {
 \$user->Nom=\$_GET['nom']; // accès à un champ
}

François Armand

48

Fonction vérification des champs

```
<?php
function verif_champ($champ, $oblig)
{
    global $ChampsManquants;
    if (isset($_GET[$champ])){
        $valeur = $_GET[$champ];
        if (!empty($valeur)) {
            return $valeur;
        }
    }
    // A suivre
}
```

Fonction vérification des champs

```
if ($oblig) {
    if (!empty($ChampsManquants)){
        $ChampsManquants=$ChampsManquants.", ";
    }
    $ChampsManquants = $ChampsManquants.$champ;
}
return "";
}
```

Script de traitement du formulaire

```
$ChampsManquants="";
class Utilisateur {
    var $Nom, $Prenom,
        $Adresse, $Email,
        $Tel, $Pays,
        $Pseudo, $Password,
        $Age;
};
define ("OBLIG", TRUE);
define ("NON_OBLIG", FALSE);
```

Script de traitement du formulaire

```
$repasswd=""; $user = new Utilisateur();
$user->nom = verif_champ('nom', OBLIG);
$user->prenom= verif_champ('prenom', OBLIG);
$user->Adresse=verif_champ('adresse',
    NON_OBLIG);
$user->Email = verif_champ('email', OBLIG);
$user->Pays = verif_champ('pays', OBLIG);
$user->Pseudo=verif_champ('pseudo', OBLIG);
$user->Password=verif_champ('passwd', OBLIG);
$repasswd = verif_champ('repasswd', OBLIG);
$user->Age = verif_champ('age', NON_OBLIG);
$user->Tel = verif_champ('phone', NON_OBLIG);
```

Fonction: argument par référence

```
function verif_champ($champ, &$liste)
{
    // Début identique version précédente
    if (is_string($liste)) {
        if (!empty($liste)){
            $liste = $liste.", ";
        }
        $liste= $liste.$champ;
    }
    return " ";
}
```

François Armand

53

Fonction: argument par référence

```
$user->nom = verif_champ('nom',
                        $ChampManquants);
```

```
$nonObl = NON_OBLIG; // Boolean
```

```
$user->Adresse = verif_champ('adresse',
                             $nonObl);
```

- On ne peut pas passer une constante par référence, seulement une variable

François Armand

54

PHP: Plan

- Ce que vous allez découvrir
- Rappels
- Tableaux, Constantes
- Erreurs, Objets
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- Transfert fichier vers client
- Ce qu'il faut retenir

François Armand

55

Script de traitement du formulaire

```
if (!empty($ChampsManquants)) {
    exit("Vous avez oublier de saisir les
    champs suivants: ".$ChampsManquants.'!');
}
if (strcmp($user->Password, $repasswd)) {
    exit("mots de passe pas identiques!");
}
if (strlen($user->Password) <6) {
    exit("mot de passe pas assez long!");
}
$user->Password=crypt($user->Password, "AZ");
```

François Armand

56

Accès aux fichiers

- But:
 - Stocker les informations reçues par formulaire dans un fichier
 - Pouvoir les afficher, les modifier, vérifier...
- Par souci de simplicité:
 - On va ranger tout un tableau dans un fichier
 - Ré-initialiser un tableau à partir d'un fichier

Sauvegarder dans un fichier

```
$users[$user->Pseudo]=$user;
```

```
// Ouvrir le fichier
```

```
if (!($Fic=fopen("FichUsers", "w+"))){
    exit("Impossible ouvrir fichier des
        utilisateurs<br/>Désolé<br/>");
}
```

```
// Ecrire le tableau dans le fichier
```

```
if (!(fwrite($Fic, serialize($users)))){
    exit("Erreur ecriture<br/>Oops<br/>");
}
```

fopen

- Ouvre un fichier ou une page via une URL
 - `$fic = fopen("Utilisateurs", mode)`
 - `$fic = fopen("http://www.pps.jussieu.fr/~armand", mode);`
- Mode:
 - "r": lecture seule, à partir du début du fichier
 - "r+": lecture/écriture, à partir du début du fichier
 - "w": écriture, fichier effacé, si fichier inexistant, fichier créé si possible,
 - "w+": idem 'w' mais lecture/écriture
 - "a": écriture à partir de la fin. Créé si inexistant
 - "a+": idem 'a' mais lecture/écriture

fwrite

- `$res= fwrite($fic, $chaîne [, $longueur]);`
 - Écrit la chaîne 'chaîne' dans le fichier `$fic` (précédemment valorisé par un appel réussi à `fopen`)
 - Tous les octets jusqu'à la fin de la chaîne sont écrits, sauf si `$longueur` est spécifié et a une valeur inférieure au nombre d'octets de la chaîne.
 - Renvoie le nombre d'octets écrits ou `FALSE` en cas d'erreur

Serialize / Unserialize

- `$str = serialize($var)`
 - Retourne une représentation '*linéarisée*' sous forme de chaîne de la valeur de la variable `$var`
- `$var = unserialize($str)`
 - Convertit une chaîne représentant une variable '*linéarisée*' en variable PHP.
 - Retourne FALSE en cas d'erreur
- Int, float, boolean, string, array, object

Lire un fichier dans un tableau

```
// Fichier => tableau de chaînes
$s = file("FichUsers");
// Tableau de chaînes => une chaîne
$s = implode("", $s);
// Une chaîne => tableau d'objets
$users = unserialize($s);
// Affecter l'élément approprié
$users[$user->Pseudo] = $user;
```

file

array `file` ('chemin')

- Renvoie le contenu du fichier de nom 'chemin' sous forme de tableau
- Chaque ligne du fichier (terminée par un caractère de retour à la ligne) est affectée à un élément du tableau

Implode / explode

`$tab = explode("delimiteur", $chaîne);`

- Renvoie un tableau de sous-chaîne, les sous-chaînes sont extraites de chaîne et délimitées par le caractère délimiteur

`$str = implode("separateur", $tableau);`

- Renvoie une chaîne contenant la représentation sous forme de chaîne de tous les éléments du tableau. Dans la chaîne finale, chaque élément est séparé du suivant par le séparateur.

Script de traitement du formulaire

```
if (!file_exists("Utilisateurs")) {
    // Le fichier n'existe pas, on le créé
    $f = fopen("Utilisateurs", "w+");
    fclose($f);
}
$s=file("Utilisateurs");
print_r($s);
$s= implode(" ", $s);
print_r($s);
$users=unserialize($s);
var_dump($users);
```

François Armand

65

Résultat de file

s après lecture du fichier:

Array ([0] =>

```
a:1:{s:6:"LeProf";O:11:"utilisateur":9:{s:3:"Nom";s:6:"armand";s:6:"Prenom";s:8:"François";s:7:"Adresse";s:28:"6 Place Jussieu
```

[1] => 75006

```
Paris";s:3:"Tel";s:0:"";s:5:"Email";s:29:"armand@ufr-info-p7.jussieu.fr";s:4:"Pays";s:2:"fr";s:6:"Pseudo";s:6:"LeProf";s:8:"Password";s:13:"AZdAzFNRbZ2ac";s:3:"Age";s:0:"";}}
```

- Le saut de ligne contenu dans le champ adresse fait que la fonction **file** renvoie un tableau avec deux éléments

François Armand

66

Résultat de « implode »

s après implode:

```
a:1:{s:6:"LeProf";O:11:"utilisateur":9:{s:3:"Nom";s:6:"armand";s:6:"Prenom";s:8:"François";s:7:"Adresse";s:28:"6 Place Jussieu 75006 Paris";s:3:"Tel";s:0:"";s:5:"Email";s:29:"armand@ufr-info-p7.jussieu.fr";s:4:"Pays";s:2:"fr";s:6:"Pseudo";s:6:"LeProf";s:8:"Password";s:13:"AZdAzFNRbZ2ac";s:3:"Age";s:0:"";}}
```

- On n'a maintenant qu'une seule chaîne

François Armand

67

Résultat de unserialize

users après unserialize:

array(1) { ["LeProf"]=> object(utilisateur)(9) {

```
["Nom"]=> string(6) "armand"
```

```
["Prenom"]=> string(8) "François"
```

```
["Adresse"]=> string(28) "6 Place Jussieu 75006 Paris"
```

```
["Tel"]=> string(0) ""
```

```
["Email"]=> string(29) "armand@ufr-info-p7.jussieu.fr"
```

```
["Pays"]=> string(2) "fr"
```

```
["Pseudo"]=> string(6) "LeProf"
```

```
["Password"]=> string(13) "AZdAzFNRbZ2ac"
```

```
["Age"]=> string(0) "" }
```

François Armand

68

Script de traitement du formulaire

```
// 1) Si pseudo pas connu => nouvel utilisateur
// 2) Si pseudo connu      => mise à jour
$users[$user->Pseudo]=$user;
// On réécrit le tableau dans le fichier.
// On remplace le contenu précédent.
if (!( $f =fopen("Utilisateurs", "w+"))) {
    exit ("Désolé");
}
if (!( fwrite($f, serialize($users)))){
    exit("Lésodé");
}
```

François Armand

69

Script de traitement du formulaire

- On poursuit par l'affichage d'un formulaire de téléchargement de fichier du client vers le serveur (**upload**)

```
if (!readfile("upload_file.html")) {
    exit("Pas de chance");
};
```

- readfile** ("fichier");
 - Lit le fichier et l'envoie vers la sortie standard
 - Renvoie le nombre d'octets lus ou FALSE

François Armand

70

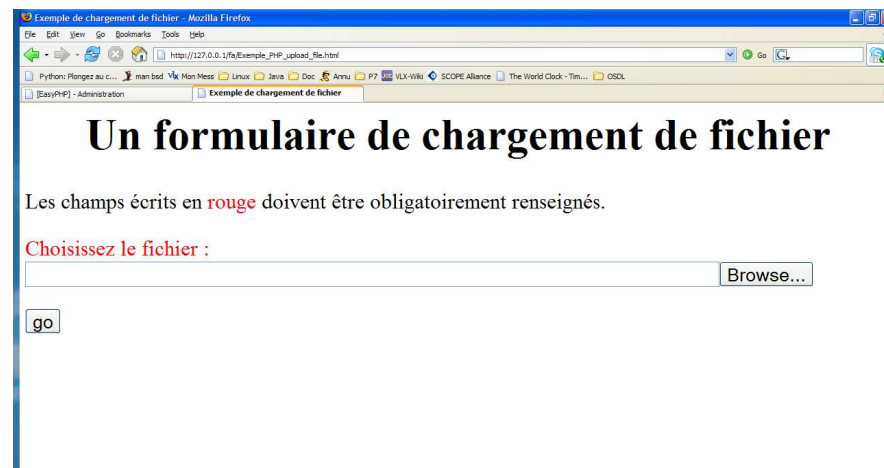
PHP: Plan

- Ce que vous allez découvrir
- Rappels
- Tableaux, Constantes
- Erreurs, Objets
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur**
- Transfert fichier vers client
- Ce qu'il faut retenir

François Armand

71

Charger des fichiers clients sur le serveur



François Armand

72

Chargement: XHTML

```
<form enctype="multipart/form-data" action="tst_upload.php"
method="post">
```

Choisissez le fichier :


```
<input type="hidden" name="MAX_FILE_SIZE"
value="30000" />
```

```
<input type="file" size="80" name="fichier" />
```

```
<input type="submit" value="go" />
```

```
</form>
```

Chargement: script PHP

```
$fichier = $_FILES['fichier']['name']; // 'fichier': champ XHTML
$codeErreur=$_FILES['fichier']['error'];
// $_FILES: tableau à 2 dimensions des fichiers chargés
if ($codeErreur == UPLOAD_ERR_OK) {
    echo "Nom côté client: ", $fichier;
    echo "Nom côté serveur :", $_FILES['fichier']['tmp_name'];
    echo "Taille :", $_FILES['fichier']['size'];
    echo "Type :", $_FILES['fichier']['type'];
    /* A suivre */
}
```

Chargement: script PHP

```
if ($codeErreur == UPLOAD_ERR_OK) {
    /* Suite */
    if ( $_FILES['fichier']['size'] > $_POST["MAX_FILE_SIZE"]) {
        echo "Trop gros<br />"; exit("File Too big");
    }
    if (!file_exists("UPLOAD_DIR")) {
        mkdir("UPLOAD_DIR", 0644);
    }
    copy($_FILES['fichier']['tmp_name'], "UPLOAD_DIR/".$fichier);
}
```

Chargement: script PHP, erreurs

```
if ($codeErreur == UPLOAD_ERR_OK) { /*...*/ } else {
    switch($codeErreur) {
        case UPLOAD_ERR_NO_FILE:
            echo "Tete de linotte! Et le fichier? Où est-il?<br/>"; break;
        case UPLOAD_ERR_INI_SIZE:
            echo "Fichier trop gros pour le serveur<br/>" break;
        case UPLOAD_ERR_FORM_SIZE:
            echo "Fichier Sumo pour le formulaire<br/>"; break;
        case UPLOAD_ERR_PARTIAL:
            echo "Fichier transféré partiellement<br/>"; break;
        default:
            echo "Loi de Murphy<br/>";
    }
}
```

Fonctions PHP, fichiers

bool **file_exists**("Chemin")

- TRUE si fichier "Chemin" existe, FALSE sinon

bool **is_file** ("filename")

- TRUE si "filename" existe et est un fichier (pas un répertoire)

bool **is_dir** ("dirname")

- TRUE si "dirname" existe et est un répertoire (pas un fichier)

Fonctions PHP, fichiers

bool **mkdir**("Chemin", mode[, recursif])

- Crée le répertoire "Chemin" avec un les droits d'accès définis par mode (en octal, ex: 0644), créé les répertoires intermédiaires nécessaires si recursif vaut TRUE

bool **copy**("source", "destination")

- Recopie le fichier "source" dans "destination", TRUE si OK

Fonctions PHP, fichiers

bool **move_uploaded_file** ("source", "destination")

- Déplace le fichier téléchargé "source". Si TRUE est renvoyé, opération réussie, "source" n'est plus un chemin de fichier valide, "destination" est le nouveau chemin du fichier

bool **unlink** ("chemin")

- Supprime le chemin d'accès au fichier

bool **rmdir** ("repertoire")

- Supprime le chemin d'accès au repertoire (vide)



Vérification du chargement

Nom coté client: Never Trust A Penguin.mpg
 Nom coté serveur :C:/Program Files/EasyPHP1-8/tmp/php19.tmp
 Taille :399123
 Type :video/mpg

Fichiers mis à votre disposition:

[04 VM def1.jpg](#)
[Never Trust A Penguin.mpg](#)
[Tulipe.gif](#)
[Tulipe.jpg](#)

PHP: Plan

- Ce que vous allez découvrir
- Rappels
- Tableaux, Constantes
- Erreurs, Objets
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- **Transfert fichier vers client**
- Ce qu'il faut retenir

François Armand

81

PHP générant le XHTML pour download

```
echo "Fichiers à votre disposition:<br/>";
if ($dir = opendir("UPLOAD_DIR")) {
    while (($f = readdir($dir)) !== FALSE) {
        if (($f != '.') && ($f != '..')) {
            echo "<a href=\"downfile.php?fic=$f\">";
            echo "$f</a><br />";
        } // if
    } // while
} // if
```

François Armand

82

PHP pour download

```
// Vérifier paramètre bien positionné
chdir("UPLOAD_DIR");
if (!is_file($fichier)) {
    exit("$fichier inexistant!<br />");
}
header("Content-type: application/force-download");
header("Content-disposition:filename=\"\$fichier\"");
readfile($fichier);
```

François Armand

83

Modifier entête HTTP

- **header("chaine");**
 - Permet de modifier les entêtes des messages HTTP envoyés par le serveur au navigateur
- "Content-type: application force-download"
 - Demande au navigateur de considérer le contenu du message comme un fichier à sauvegarder, non pas à afficher.
- "Content-disposition: filename=toto"
 - Propose un nom par défaut au navigateur
 - Attention aux espaces dans "toto"!

François Armand

84

Manipulation de chaînes

- **htmlentities**("chaîne");
 - Renvoie une chaîne où les caractères spéciaux sont remplacés par des entités.
- **htmlspecialchars**("chaîne");
 - Renvoie une chaîne où `<` `>` `&` `'` `"` sont remplacés par des entités
- **addslashes**("chaîne");
 - Renvoie une chaîne où les `'` et `"` sont préfixés par `\`

Manipulation de chaînes

- **stripslashes**("chaîne");
 - Renvoie une chaîne où les `\` sont supprimés devant les `'` et les `"`
- **urlencode**("chaîne");
 - Encode une chaîne en URL
 - Caractères non alphanumériques => **%xx**
 - Espaces remplacés par **+**

PHP: Plan

- Ce que vous allez découvrir
- Rappels
- Tableaux, Constantes
- Erreurs, Objets
- Exemple fonction
- Sauvegarde dans un fichier
- Transfert fichier vers serveur
- Transfert fichier vers client
- **Ce qu'il faut retenir**

Ce qu'il faut retenir

- La notion d'objet, tableaux d'objets
- Sérialisation, désérialisation
- Manipuler les fichiers
 - Créer, sauvegarder, lire
- Comment recevoir un fichier uploadé `$_FILES[]`
- Comment transférer un fichier vers le client
 - Générer un URL approprié
 - Modifier les entêtes HTTP