PhP - Sessions et fichiers

©EISTI

22 mars 2010

PhP et les formulaires

Récupération de données

- Nous avons manipulé au cours précédent 2 variables
  - GET : `$_GET`
  - POST : `$_POST`
- elles correspondent aux variables de passage de paramètres
- elles existent mais ne sont pas forcément initialisées

Session php

Problématique

- Le passage d'une variable se fait, par exemple d'un formulaire vers le serveur
- la page résultat traite ces variables
- problème : comment faire garder ces variables pour les utiliser de page en page sur le site tout au long de la connexion de l'utilisateur
- exemple : login, panier d'achat

Session php

Présentation

- Les sessions en PhP permettent de sauvegarder, côté serveur, des variables envoyées par le client.
- Les variables de session sont stockées dans une variable dite de session, valide sur le serveur PhP tant que l'utilisateur ne quitte pas le site
- La durée de validité d'une session est 30 min (paramétrable dans le fichier de configuration `php.ini`)
- les variables sont contenues dans le tableau associatif `$_SESSION ['nom']`

Les variables de session

Création

- Créer une variable de session :
```
<?php
session_start() ;
$_SESSION ['nom']='valeur' ;
? >
```
- Utiliser une variable de session :
```
<?php
session_start() ;
$variable=$_SESSION ['nom']='toto' ;
echo $variable ;
? >
```
- l'instruction `session_start()` doit figurer en tête de page, précédant tout instruction PhP ou HTML, dans chaque page du site où l'on utilise ces variables
- elle permet de générer un identifiant, stocké dans un cookie, puis de récupérer le tableau de variables si l'id a déjà été généré lors de la navigation

Les variables de session

Suppression

- Effacer une variable de session :

```
<?php
session_start() ;
unset($_SESSION['nom']) ;
? >
```

- Effacer toutes les variables de session :

```
<?php
session_start() ;
$_SESSION = array() ;//supprime les variables en créant un nouveau tableau
session_destroy() ;
? >
```

Les variables de session

Sauvegarde

- Stocker les variables de session dans un fichier :
  Si vous souhaitez réutiliser les variables de session ultérieurement (les sauvegarder dans un cookie, un fichier sans parrser par une BDD), vous pouvez utiliser la fonction *session_encode()* :
  <?php
  session_start() ;
  $var=session_encode() ; // retourne une chaîne contenant les variables de la session courante encodées et prêtes au stockage
  ? >
- Récupérer les variables de session :
  Il s'agit ici de récupérer les variables de session stockées à l'aide de la fonction *session_decode()* :
  <?php
  session_start() ;
  session_decode($var) ;
  ? >

Les variables de session

Idenfication

- Créer un identifiant de session :
  Si vous souhaitez connaître l'identifiant de session courante, utilisez la fonction *session_id()* : <?php
  session_start() ;
  $id=session_id() ; // retourne une chaîne contenant l'identifiant de la session courante
  ? >
- Fixer l'identifiant de session :
  Il s'agit ici de récupérer les variables de session associées à un id particulier ; ici, session_start() figure après avoir fixé l'id <?php
  session_id($id) ;
  session_start() ;
  ? >

Les autres variables

Environnement

- Variables d'environnement du serveur
- Tableau associatif `$_ENV`
  - $_ENV['OS'] : système d'exploitation
  - $_ENV['TMP'] : chemin du répertoire temporaire

Les autres variables

Serveur

- Variables contenant des informations comme les en-têtes, dossiers et chemins du script en cours
- Tableau associatif `$_SERVER`
  - $_SERVER['PHP_SELF'] : chemin du script courant
  - $_SERVER['SERVER_NAME'] : nom du serveur
  - $_SERVER['REMOTE_ADDR'] : IP du client
  - $_SERVER['DOCUMENT_ROOT'] : racine sous laquelle le script courant est exécuté
  - $_SERVER['QUERY_STRING'] : chaîne de requête qui est utilisée pour accéder à la page

Les autres variables

Présentation

- définies dans le fichier `php.ini`
- modifier avec *ini_set(variable,valeur)* ;
- récupérer la valeur initiale avec *ini_restore()* ;
- liste complète des variables $_ENV et $_SERVER et server donnée avec la fonction *phpinfo()*

Les fichiers

Présentation

- PhP premet de manipuler facilement des fichiers
- création, ouverture, fermeture, suppression
- lecture/écriture
- tests

Les fichiers

Présentation

- fonctions *fopen, fclose, touch, unlink*
- Exemples : <?php
$fic = "fichier.txt" ; touch($fic) ; //création
$monfichier = fopen($fic, "r+") ; //ouverture du flux
... traitements ...
fclose($monfichier) ; //fermeture
unlink($fic) ; //suppression
? >

Les fichiers

Droits

- Droits en ouverture d'un flux
  - r lecture seule (curseur en début de fichier)
  - r+ lecture/écriture
  - w ecriture seule + création s'il n'existe pas (curseur en début de fichier)
  - w+ lecture/écriture + création s'il n'existe pas
  - x crée et ouvre en lecture seule, échoue s'il existe déjà (curseur en début de fichier)
  - x+ crée et ouvre en lecture/écriture, échoue s'il existe déjà
  - a ecriture seule + création s'il n'existe pas (curseur en fin de fichier)
  - a+ lecture/écriture + création s'il n'existe pas + ajout en fin de fichier

Les fichiers

Manipulation

- *fgetc, fgets* : lecture par caractère, par ligne
  ex : $car = fgetc($monfichier) ; //lit un caractère du fichier
  ex : $ligne = fgets($monfichier) ; //lit une ligne de mon fichier
- *fputs* : écriture
  ex : fputs($monfichier, "Texte") ;
- *fseek* : se déplacer
  ex : fseek($monfichier, 0) ; //se positionne au début
- *ftell* : donne positionne du pointeur dans le fichier
  ex : $pos = ftell($monfichier) ;
- *file* : mise à plat du fichier dans un tableau
  ex : $tab = file($monfichier) ;
  for($i = 0 ; $i < count($tableau) ; $i++){
  echo $tab
  $i
  ;
  }

Les fichiers

Tests

- *file_exists(chaine)* : teste si le fichier nommé *chaîne* existe
- *feof* : teste la fin de fichier
  ex : while (!feof($monfichier))
- *is_dir, is_link is_file* teste si c'est un fichier ou un répertoire
- *filetype(chaine)* : retourne le type du fichier *chaine*

Les fichiers

Autres fonctions utiles

- *chown(chaine, user)* : change le propriétaire du fichier *chaine*
  `chown("toto.txt","toto") ;`
- *chmod(chaine, mode)* : change les droits du fichier *chaine*
  `chmod("toto.txt",755) ;`
- *copy(fic1,fic2)* : copie le contenu de *fic1* dans *fic2*
- *is_dir, is_link is_file* teste si c'est un fichier ou un répertoire
- *filesize(chaine)* : donne la taille du fichier *chaine*