



Module PHP

Cours 5

SOMMAIRE

- Exercice variables de sessions
- Exercices conditions et boucle
- Différences include et require
- La redirection
- Les variables superglobales
- L'upload de fichier
- Les fonctions SQL
- TP 4

EXERCICE VARIABLE DE SESSIONS

- Expliquer à quoi servent les variables de sessions
- Comment les utilisent-on?

EXERCICES

- Demander à un utilisateur de saisir une heure, créer une condition qui affiche « Bonne matinée » si c'est le matin, « Bonne après-midi » si c'est l'après-midi ou « Bonne nuit » si c'est la nuit.
- Afficher la table de multiplication du chiffre saisi par un utilisateur.
- Ecrire les chiffres de 0 à 100 à l'aide d'une boucle.

DIFFÉRENCES ENTRE INCLUDE ET REQUIRE

La fonction `include()` permet d'inclure un fichier dans un programme. Elle s'utilise comme suit: `include 'header.php';`

La fonction `require()` quant à elle, permet d'indiquer qu'un fichier est requis, c'est-à-dire que s'il n'est pas trouvé cela génère une erreur. `require 'header.php';`

Il est possible d'indiquer qu'un fichier soit inclus qu'une seule fois à l'aide la fonction: `include_once 'header.php';`

De même, il est possible d'indiquer qu'un fichier est requis qu'une seule fois: `require_once 'header.php';`

LE REDIRECTION

La redirection en PHP peut être effectuée à l'aide de la fonction `header()`. Pour la configuration, il suffit de créer un fichier `index.php` dans le répertoire à partir duquel vous souhaitez faire la redirection en y spécifiant le contenu suivant:

```
<? Php
    header (« Location: http://www.redirection.com/");
?>
```

Où «`http://www.redirection.com/`» est l'URL où vous souhaitez rediriger les utilisateurs.

Il est possible de la même manière de rediriger vers un fichier, comme ceci:

```
<? Php
    header ("Location: mon_dossier/mon_fichier.php");
?>
```

LES SUPERGLOBALES

Elles sont écrites en majuscules et commencent toutes par un underscore , Citez celles que vous connaissez.

Les superglobales sont toutes des array.

Ces variables sont automatiquement créées par PHP à chaque fois qu'une page est chargée. Ces variables existent donc sur toutes les pages et sont accessibles partout : au milieu de votre code, au début, dans les fonctions etc...

LES SUPERGLOBALES

Nous avons déjà vu `$_POST`, `$_GET`, `$_SESSIONS`

À présent, nous allons voir `$_SERVER` et `$_COOKIE`:

- `$_SERVER` permet de récupérer des valeurs retournées par le serveur. Pour les afficher, il faut indiquer ce qu'on demande entre crochets puisque que c'est un array.

Exemple:

```
<?php
    echo $_SERVER['SERVER_NAME']; affiche le nom relatif du serveur
    echo $_SERVER['HTTP_REFERER']; affiche le nom de la page précédente.
?>
```

Voici une liste: <https://www.php.net/reserved.variables.server>

LES SUPERGLOBALES

- `$_COOKIE`: Un cookie est un fichier texte que l'on stocke sur le poste du visiteur, il permet de stocker des données tel que le pseudo par exemple.

La fonction PHP qui permet d'écrire un cookie est `setcookie()`.

On utilise généralement les 3 premiers paramètres en respectant l'ordre suivant:

- Le nom du cookie (ex : "Nom")
- La valeur du cookie (ex : "Paul")
- La date d'expiration du cookie, sous forme de timestamp (ex : 1090521508)

Le 3ème paramètre définit la durée de vie du cookie, après cette date il sera supprimé

Exemple:

```
<?php
```

```
    $timestamp = time() + 365*24*3600; // Le cookie a une durée de vie de 1an  
    setcookie('Nom', 'Dupond', $timestamp); // On écrit un cookie  
    setcookie('Prenom', 'Patrick', $timestamp); // Un autre cookie...
```

```
?>
```

Notez que ce code sera noté avant le début du code HTML.

Maintenant pour afficher le contenu du cookie, on utilise `$_COOKIE['Nom_cookie']`

```
echo "Bonjour " .$_COOKIE['Nom'] . " " .$_COOKIE['Prenom']; // Affichera les cookies de l'exemple
```

L'UPLOAD DE FICHER

Prenons pour exemple l'upload d'une image, nous aurons besoin de 3 éléments:

- Un champ de type upload,
- Des instructions php,
- Un dossier où seront stockées les images

L'UPLOAD D'IMAGE:

Champ de type upload

```
<form action="upload.php" method="post" enctype="multipart/form-data">  
  
  <label for="upload">Fichier:</label>  
  
  <input type="file" name="photo" id="upload">  
  
  <input type="submit" name="submit" value="Upload">  
  
</form>
```

L'UPLOAD D'IMAGE:

Instructions php

1) Vérifier si le fichier a été uploadé sans erreur.

```
if(isset($_FILES["photo"]) && $_FILES["photo"]["error"] == 0){
```

2) On utilise la variable superglobale `$_FILES` pour récupérer les infos de l'image:

```
$filename = $_FILES["photo"]["name"];
```

```
$filetype = $_FILES["photo"]["type"];
```

```
$filesize = $_FILES["photo"]["size"];
```

3) On upload le fichier dans le dossier à l'aide de la fonction

```
move_uploaded_file(from, to)
```

```
move_uploaded_file($_FILES["photo"]["tmp_name"], "images/" . $_FILES["photo"]["name"]);  
}
```

L'UPLOAD D'IMAGE: Dossier de stockage

Il sert à recueillir les images uploader.

Il faut s'assurer que son chemin est bien indiqué dans le code php pour permettre le stockage.

Aller plus loin:

<https://www.damienflandrin.fr/blog/post/tutoriel-comment-uploader-un-fichier-en-php>

LES FONCTIONS SQL

Les fonctions SQL permettent d'effectuer des requêtes plus élaborées, par exemple adaptant les résultats pour qu'une chaîne soit affichée en majuscule ou bien pour enregistrer une chaîne avec la date actuelle.

Voici quelques exemples:

SUM() calculer la somme totale d'une colonne de valeurs numériques.

MAX() obtenir le résultat maximum

MIN() obtenir le résultat minimum

COUNT() compter le nombre de lignes dans un résultat

ROUND() arrondir la valeur

UPPER() afficher une chaîne en majuscule

LOWER() afficher une chaîne en minuscule

NOW() date et heure actuelle

RAND() retourner un nombre aléatoire

CONCAT() concaténer des chaînes de caractères

CURRENT_DATE() date actuelle

TP 4

Une fois l'utilisateur connecté, il devrait pouvoir laisser un commentaire pour donner son avis sur le site.

Le commentaire devra s'afficher à la suite des paragraphes.

On doit savoir qui a déposé le commentaire, le username suffira.

TP 5

- Un utilisateur non connecté doit pouvoir lire les commentaires mais ne peut pas en rédiger.
- S'il souhaite déposer un commentaire, il faut lui proposer de se connecter ou de créer un compte.