

Conception de Sites Web Interactifs : TP 9

Login des membres - Variables de session

1 Environnement de travail

Démarrer sous WINDOWS et mettez vous dans des conditions de travail habituel (Notepad++, EasyPhp 1.8). Aujourd'hui, on travaillera dans le dossier `tp9`.

2 Inclure le fichier `connect_db.php`

Comme d'habitude, tous les fichiers qui utilisent votre base de données, utiliseront le fichier `connect_db.php` en l'incluant avec

```
require_once("connect_db.php");
```

3 Aujourd'hui : Variables de session

Dans cet exercice, vous allez apprendre à utiliser les variables de session. Le tableau des variables de session (`$_SESSION`) permet d'enregistrer des informations qui pourront être utilisées tout au long d'une session d'un utilisateur donné, et en particulier sur plusieurs pages web d'un site web, sans passer à chaque fois par des paramètres dans l'URL ou dans les formulaires.

Pour cela, **en tout début de chaque page PHP**, mettez le code suivant :

```
<?php session_start(); ?>
```

Bien sûr, vous pouvez placer cette instruction aussi dans le fichier `connect_db.php`, étant donné que ce fichier est inclus en tout début de chaque page PHP.

Dans cet exercice, vous allez créer des sondages à choix multiples pour les internautes. Pour cela, on aura besoin de deux tables, une pour les questions, et une autre pour les réponses à la question, afin d'avoir un nombre de choix variable par question.

Exercice 9.1 *Création d'une table de la base de données*

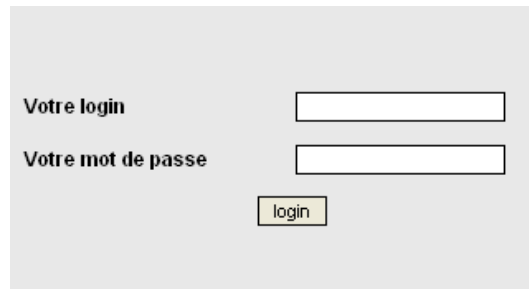
Cette exercice concerne uniquement les étudiants qui n'ont pas encore créé de table `membres` dans le TP6.

1. Créer une table dans votre base de données `base_cswd` qui s'appelle `membres` avec les champs suivants : un champ INT qui s'appelle `id` avec les attributs clé primaire ainsi que "auto-increment", un champ TEXT qui s'appelle `pseudo`, et un champ TEXT qui s'appelle `motdepasse`, un champ TEXT qui s'appelle `prenom`, un champ TEXT qui s'appelle `nom`, un champ TEXT qui s'appelle `email`
2. Créer quelques enregistrements dans cette table.

Exercice 9.2 *Création des tables*

1. Créer une table qui s'appelle `sondages` dans votre base de données `base_cswd` avec les champs suivant : un champs INT qui s'appelle `id` avec les attributs clé primaire ainsi que "auto-increment", et un champ TEXT qui s'appelle `question`.
2. Créer une deuxième table qui s'appelle `reponses` dans votre base de données `base_cswd` avec les champs suivant : un champs INT qui s'appelle `id` avec les attributs clé primaire ainsi que "auto-increment", un champ TEXT qui s'appelle `texte`, un champ INT qui s'appelle `votes`, et un champ INT qui s'appelle `id_sondage` (qui est en fait la clé étrangère pour identifier de quelle sondage il s'agit).

Exercice 9.3 *Connexion*



The image shows a login form on a light gray background. It contains two text input fields. The first field is labeled "Votre login" and the second is labeled "Votre mot de passe". Below the second field is a button labeled "login".

1. Créer une page PHP `connexion.php` qui permet à un membre de se connecter au site (un formulaire avec un champ texte et un champ mot de passe ainsi qu'un bouton valider) et qui renvoie à la page `verifier_connexion.php`.
2. Créer la page PHP `verifier_connexion.php` qui vérifie si le membre donné existe dans la base de donnée. Vous pouvez vous en servir de la requête suivante, qui utilise la clause "WHERE":

```
$sql = "SELECT * FROM membres WHERE pseudo = '...';";
```

3. Si le membre n'existe pas, afficher un message, ainsi qu'un lien hypertexte qui renvoie vers la page `connexion.php`.
4. Si le membre existe, vérifier si le mot de passe correspond bien à celui stocké dans la base de données. S'il ne correspond pas, afficher un message, ainsi qu'un lien hypertexte qui renvoie vers la page `connexion.php`.
5. Si le membre existe et le mot de passe correspond, afficher un message de bienvenu. De plus, enregistrer affecter les deux variables de session `$_SESSION['pseudo']` ainsi que `$_SESSION['membre_id']` avec le pseudo et son identifiant, respectivement. *Remarque* : Si le pseudo est stocké dans une variable intitulé `$pseudo`, c'est uniquement l'affectation `$_SESSION['pseudo'] = $pseudo`.

Exercice 9.4 *Informations uniquement pour les membres*

Créer une nouvelle page, par exemple `secret.php`, qui affichent des informations qui doivent

être lu uniquement par les membres du site, ou le message "Accès interdit!" pour les non-membres. *Remarque* : il suffit de tester si la variable de session `$_SESSION['membre_id']` est supérieur à 0)

Exercice 9.5 *Deconnexion*

Créer une nouvelle page `deconnexion.php` qui permet au membre de se déconnecter. *Remarque* : il suffit d'affecter les variables de sessions avec 0 !

Exercice 9.6 *Page infos du membre*

Ecrire une nouvelle page `membre_infos.php` que vous pouvez appeler `membre_infos.php?id=3` par exemple, et qui permet d'afficher les informations relatif au membre `id`. Par contre, afficher uniquement les informations confidentiel (comme par exemple le mot de passe ...) pour le membre connecté ! *Remarque* : il suffit de tester si la variable de session `$_SESSION['membre_id']` est égal à `id`)

Vérifier tous vos fichiers XHTML et CSS avec les validateurs (pour les feuilles avec du code PHP, attention : passer par "afficher source" dans votre navigateur").

<http://validator.w3.org/>

<http://jigsaw.w3.org/css-validator/>