

TP n° 10

Enregistrer des utilisateurs à un forum

Dans ce TP, nous allons voir comment gérer des utilisateurs. Nous allons nous intéresser au premier enregistrement d'un nouvel utilisateur et à la page qui lui permettra de se logger sur le site. Pour cela, nous allons utiliser la table `users` que l'on a créé avec le fichier `create_table_users.sql`. Nous vous conseillons de faire plusieurs fichiers avec des fonctions.

Exercice 1 Commencez par regarder à quoi ressemble cette table.

Exercice 2 Faites une page `nouveau.php` qui

- soit présentera un formulaire pour s'inscrire sur le site en demandant le nom et le prénom de l'utilisateur, le pseudo et le mot de passe qu'il veut utiliser. (Le pseudo servira de login ensuite.) Ce formulaire enverra les données par POST **à lui-même**.
- soit, si l'utilisateur a déjà fait sa demande testera si tout est correct (paramètres POST tous remplis) et insèrera le nouvel utilisateur dans la base. L'insertion se fait avec `mysqli_query` et retourne `false` en cas d'erreur. Si l'inscription est réussie, on affichera "inscription réussie", sinon, on affichera un message d'erreur et on réaffichera le formulaire.

Important : Le mot de passe devra être chiffré avant d'être enregistré. Sur les machines du script les méthodes `password_hash` et `password_verify` ne fonctionnant pas, on utilisera à la place la fonction `md5` pour chiffrer le mot de passe et `===` pour tester l'égalité entre les deux chaînes chiffrées. Par ailleurs, on utilisera `mysqli_real_escape_string` partout où c'est nécessaire.

Vérifiez en regardant directement dans la base que le mot de passe est bien chiffré.

Exercice 3 Faites une page `accueil.php` qui

- soit présentera un formulaire pour se logger sur le site en demandant le pseudo et le mot de passe. Ce formulaire enverra les données par POST à lui-même. On affichera aussi un lien en bas du formulaire vers `nouveau.php` pour qu'un nouvel utilisateur puisse s'inscrire.
- soit, si l'utilisateur a déjà fait son login testera si tout est correct (champs tous remplis) et vérifiera que les données correspondantes sont dans la base. Si le login est réussi, on affichera "login réussi", sinon, on affichera un message d'erreur et on réaffichera le formulaire.

Exercice 4 — Un peu plus d'infos Complétez la base de donnée et vos pages `.php` pour stocker la ville de résidence de vos utilisateurs.

On va vouloir maintenant écrire plusieurs pages web permettant de consulter la fiche d'un utilisateur, ajouter ou supprimer un utilisateur et modifier les infos d'un utilisateur. A chaque fois vous écrirez une page `.php` différente. Écrivez aussi une page `Accueil.html` qui aura quatre liens vers ces différentes pages.

Exercice 5 — Informations sur les utilisateurs Écrivez un fichier `php` qui propose un formulaire permettant de choisir le pseudonyme (nickname) d'un utilisateur, puis lorsqu'un pseudonyme a été choisi, affiche les informations concernant cet utilisateur. Par exemple, lorsque "marco" aura été choisi, la page web devra ressembler à ce qui est présenté en Figure 1.

Attention à bien utiliser des jointures ! Vous devez récupérer les données d'un utilisateurs en effectuant le moins de requêtes SQL possibles.

Exercice 6 — Interface Administrateur Nous allons maintenant ajouter des comptes administrateurs pour gérer les informations des autres comptes.

[Retourner à l'Accueil](#)

Choisissez un utilisateur marco Visualiser cet utilisateur

Nom: Dupont
Prénom: Marc
Surnom: marco
Ville : Zillisheim
Departement : Haut-Rhin
Region : Alsace

FIGURE 1 – Aperçu de la page affichant les informations des utilisateurs.

1. Ajouter un attribut “admin” à votre base de donnée Users. Ajouter un administrateur dans votre base pour pouvoir tester vos pages par la suite.
2. Ajoutez une page uniquement accessible aux administrateurs (faites les vérifications grâce aux sessions) permettant d’ajouter un utilisateur, votre page devra ressembler à la Figure 2.

[Retourner à l'Accueil](#)

Nom:
 Prénom:
 Surnom:
 Mot de passe :
 Ville : Aast
Insérer l'utilisateur

FIGURE 2 – Aperçu de la page permettant d’ajouter des utilisateurs.

3. Ajoutez à la page de l’exercice précédent un bouton pour supprimer l’utilisateur sélectionné de la base.
4. Faites en sorte que le bouton ajouté à la question précédente ne soit présent que si l’utilisateur courant est un administrateur.
5. Pour aller plus loin : un administrateur peut modifier toutes les données d’un utilisateur. Assurez-vous qu’un simple utilisateur ne peut pas modifier les données d’un autre utilisateur mais puisse modifier les siennes ! Voir Figure 3 pour un aperçu de la page demandée.

[Retourner à l'Accueil](#)

Choisissez un utilisateur big_head Acceder à cet utilisateur

Nom:
 Prénom:
 Surnom:
 Ville : Zillisheim
Modifier l'utilisateur

FIGURE 3 – Aperçu de la page permettant de modifier des utilisateurs.

6. Dans la page qui permet de choisir un utilisateur pour en afficher les données, remplacer la liste déroulante par un champ de texte. La page doit bien sûr afficher proprement un

message d'erreur si le pseudonyme saisi n'existe pas dans la base. **Attention aux injections SQL!** (cf. Figure 4)



FIGURE 4 – Source : XKCD.

Exercice 7 — Des formulaires plus propres Améliorez la qualité des formulaires permettant de sélectionner les villes pour un utilisateur. Regrouper les villes par ordre alphabétique suivant les départements comme présenté en Figure 5. Vous trouverez la référence sur 'optgroup' ici : <https://developer.mozilla.org/fr/docs/Web/HTML/Element/Optgroup>.

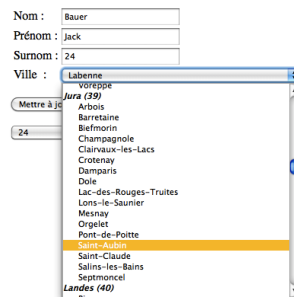


FIGURE 5 – Des 'select' utilisant des 'optgroup'.

Exercice 8 — Des requêtes plus compliquées

1. Écrivez une nouvelle page qui vous permet de choisir une ville et qui affiche la liste de tous les utilisateurs vivant dans cette ville, classée par ordre alphabétique des pseudonymes.
2. Même chose pour la liste des utilisateurs vivant dans une région commençant par la lettre 'A' si on est administrateur, par la lettre 'P' sinon.