

# TP de révisions

## Internet et Outils (IO2)

Le but de ce TP, outre de vous faire réviser les connaissances acquises cette année, est de vous faire créer un mini site web, aussi avancé que possible, pour partager des photos avec des amis. Le site devra par exemple avoir les fonctionnalités suivantes : chacun doit pouvoir se connecter avec un login et un mot de passe (ou se créer un compte), et mettre des photos en ligne, classées par album. Des mots clefs pourront être associés aux photos, et il y aura une fonction de recherche par mot clef, par album ou par date. Enfin, chacun pourra laisser des notes et des commentaires pour chaque photo.

### 1 Lancement du site

#### Exercice 1 — *Création des bases de données*

Créez une nouvelle base de données contenant les tables de la figure suivante (cf TP 13 pour le mode d'emploi). Créez comme dans le TP13 un fichier `connect.php` qui contient les éléments de connexion à la base de données. Puis créez une page `index.php` qui se connecte à la base de données en utilisant la commande `require(connect.php)`.

#### Exercice 2 — *Identification et création de login*

Lorsqu'un utilisateur se connecte, on veut qu'il entre obligatoirement son login (qui sera ensuite sauvegardé pendant toute la durée de la session, ou bien plus longtemps dans un cookie si vous préférez). Créez la page `index.php`. Lorsqu'aucun login n'est sauvegardé dans la session courante, cette page devra afficher deux formulaires, l'un qui permet de s'identifier via login et mot de passe, l'autre qui permet de se créer un nouveau login avec un champ pour définir un mot de passe et un champ de confirmation du mot de passe. Le premier formulaire doit accéder à la table `Utilisateurs` de la base de données et vérifier que le nom d'utilisateur et le mot de passe soient bien valides. Le second formulaire doit vérifier que le nouveau login ne soit pas déjà présent dans la table `Utilisateurs`, que les deux mot de passe rentrés soient bien les mêmes, et si toutes ces conditions sont remplies, doit créer le nouveau login. Dans chacun des cas, le login doit être enregistré pour la session courante (ou dans un cookie).

#### Exercice 3 — *Un peu de mise en page*

Voilà, maintenant il faut prendre quelques décisions. Comment organiser le site ? Une possibilité (mais vous pouvez adopter une autre solution) serait de créer, en plus de `index.php` deux autres pages : l'une contenant un formulaire pour uploader de nouvelles photos, et l'autre pour rechercher des photos, leur associer des notes, des commentaires ou des mots clefs. Une autre solution est de tout faire sur la même page, mais cela risque de devenir assez confus. Quoique vous décidiez, organisez la page `index.php` pour que l'on puisse clairement accéder à toutes les fonctionnalités. (C'est donc le moment de ressortir votre vieux fichier css préféré...).

### 2 Ajout de nouvelles photos

#### Exercice 4 — *Upload de photos*

Créez le formulaire qui permet à l'utilisateur connecté d'uploader une nouvelle photo sur le serveur. Ce formulaire devra permettre d'entrer les champs suivant : nom de fichier , titre

```

Terminal — mysql — 79x81
mysql> SHOW TABLES;
+-----+
| Tables_in_revisions |
+-----+
| Albums               |
| Commentaires        |
| Motsclés             |
| Notes                |
| Photos               |
| Utilisateurs        |
+-----+
6 rows in set (0.01 sec)

mysql>
mysql>
mysql> DESC Albums;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| IDalbum | int(11)   | NO   | PRI | NULL    | auto_increment |
| Nom     | varchar(30) | NO   |     |         |               |
| IDuser  | int(11)   | NO   |     |         |               |
| Date    | date      | YES  |     | NULL    |               |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.27 sec)

mysql> DESC Commentaires;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| IDphoto    | int(11)   | NO   |     |         |               |
| IDuser     | int(11)   | NO   |     |         |               |
| Commentaire | varchar(150) | NO   |     |         |               |
| Date       | date      | NO   |     |         |               |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.08 sec)

mysql> DESC Motsclés;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| IDphoto | int(11)   | NO   |     |         |               |
| Mot     | varchar(30) | NO   |     |         |               |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql> DESC Notes;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| IDphoto | int(11) | NO   | PRI |         |               |
| IDuser  | int(11) | NO   | PRI |         |               |
| Note    | int(11) | YES  |     | NULL    |               |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.07 sec)

mysql> DESC Photos;
+-----+-----+-----+-----+-----+-----+
| Field | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| IDphoto | int(11)   | NO   | PRI | NULL    | auto_increment |
| Titre   | varchar(30) | NO   |     |         |               |
| Date    | date      | NO   |     |         |               |
| Lieu    | varchar(30) | YES  |     | NULL    |               |
| IDuser  | int(11)   | NO   |     |         |               |
| IDalbum | int(11)   | YES  |     | NULL    |               |
| Source  | varchar(30) | YES  |     | NULL    |               |
+-----+-----+-----+-----+-----+-----+
7 rows in set (0.08 sec)

mysql> DESC Utilisateurs;
+-----+-----+-----+-----+-----+-----+
| Field      | Type      | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| IDuser     | int(11)   | NO   | PRI | NULL    | auto_increment |
| Pseudo     | varchar(15) | NO   |     |         |               |
| MotdePasse | varchar(30) | NO   |     |         |               |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql>

```

de la photo, mot(s)-clef(s) (facultatif). Dans ce formulaire, on devra aussi pouvoir choisir (bouton radio) si on veut mettre la photo dans un album quelconque déjà existant (un menu déroulant devra proposer tous les noms d'album existant), dans un album déjà existant *créé par l'utilisateur courant* (un menu déroulant devra donc proposer tous les noms d'album créés par l'utilisateur) ou dans un nouvel album dont on devra entrer le nom.

La validation de ce formulaire devra vérifier que toutes les informations nécessaires ont été entrées, et faire les modifications suivantes dans la base de données.

1. Ajouter la photo à la table `Photos` en remplissant tous les champs .
2. Si un mot clef a été rentré, mettre à jour la table `Motscles`.
3. Si un nouvel album a été créé, mettre à jour la table `Albums`.

#### **Exercice 5 — Affichage du résultat**

Vous pouvez faire afficher un message d'erreur si quelque chose s'est mal passé. Idéalement, le message d'erreur explique ce qui n'a pas marché : " absence de titre de la photo" ou encore "nom de fichier déjà existant dans la base de données", etc etc. Dans le cas contraire, vous pouvez aussi faire afficher un message disant que la transaction s'est bien passée, et signalant à l'utilisateur qu'il peut uploader une nouvelle photo.

### **3 Fonctions de recherche et d'affichage**

#### **Exercice 6 — Rechercher des photos**

Créez le formulaire et le code php qui permettent de créer la requête pour chercher des image par mot clef, par propriétaire, par album et par date (du style date de la photo entre telle date et telle date). Le but est de créer un genre de "recherche avancée" où l'utilisateur ne rentre que les champs qui l'intéressent, et la requête doit renvoyer toutes les photos qui correspondent à TOUS les critères. Donc par exemple si l'utilisateur laisse tous les champs vide, la requête renvoie toute la base de donnée `Photos`.

#### **Exercice 7 — Affichage des titres des photos**

Créez une fonction qui affiche tous les titres et les miniatures des photos du résultat d'une requête. L'utilisateur doit pouvoir accéder à la page de l'exercice suivant grâce à un lien (sur le titre et sur la vignette). La valeur que l'on va envoyer à la page suivante devra être le numéro d'identification de la photo choisie.

#### **Exercice 8 — Affichage d'une photo en particulier**

La page `affichage.php` doit afficher la photo sélectionnée ainsi que toutes les informations disponibles pour cette photo (nom de l'auteur, date, titre, note globale, mots clefs associés, album, créateur de l'album, etc...). N'oubliez pas un lien pour permettre de revenir à la page précédente.

### **4 Notation des images**

#### **Exercice 9 — Ajout de mots clef, de commentaires et d'une note**

Sur la page `affichage.php` créez un petit formulaire qui permette de laisser des mots clefs associés à cette photo, un commentaire et une note. A la validation du formulaire ces informations devront être rentrées dans la base de données. Attention, chaque personne n'est autorisée à noter chaque photo qu'une seule fois...

#### **Exercice 10 — Première de couverture**

Maintenant que nous avons un système de notation, je vous propose de finir en beauté en affichant sur `index.php` la photo ayant le score maximum (ou peut être le trio gagnant!).