

Informations Complémentaires r307 PHP PDO PGSQL

Alias Apache

- Créer un nouveau répertoire dédié à ce projet r307 et y placer un fichier index.php contenant simplement l'instruction suivante :
`<?="Je suis bien en PHP sur mon project r307Project";`
- Créer un nouvel alias r307phppdo qui pointe vers le chemin du répertoire ci-dessus, pour que votre projet soit accessible depuis <http://localhost/r307phppdo/>
⇒ C:\Users\guerr\owncloud\www\katana_win\r307-phppdo

Concernant l'initialisation de votre base PostgreSQL

- Nom de la base de données :
 - r307project
- User :
 - r307user
- Password :
 - r307pass (ou autre mais souvenez-vous en !)

⇒ On commence par créer la base de données r307project avec le user postgres (admin par défaut)

```
$ createdb -h localhost -U postgres r307project
```

⇒ Ensuite on crée l'utilisateur r307user qui n'a aucun droit par défaut

```
$ createuser -U postgres --interactive r307user
```

Le nouveau rôle est-il super-utilisateur ? (o/n) n

Le nouveau rôle est-il autorisé à créer des bases de données ? (o/n) n

Le nouveau rôle est-il autorisé à créer de nouveaux rôles ? (o/n) n

Mot de passe : <ici c'est le mot de passe de l'utilisateur postgresSQL pour créer le nouvel user>

⇒ On se connecte ensuite à postgresSQL en tant qu'admin pour donner les droits à r307user sur la base r307project

```
$ psql -U postgres
```

```
postgres=# GRANT ALL PRIVILEGES ON DATABASE r307project TO r307user;
```

```
postgres=# ALTER DATABASE r307project OWNER TO r307user;
```

```
postgres=# ALTER ROLE r307user WITH PASSWORD 'r307pass';
```

```
postgres=# quit
```

⇒ Enfin, on peut se connecter avec l'utilisateur r307user sur la base de données r307project

```
$ psql -U r307user r307project
```

Sujet TP - Mise en pratique

Exercice 0 - Initialisation du projet

- Récupérer le TD3 de r301 et le copier dans le répertoire de r307
- Reprendre les instructions de création de la table mp_users et créer un nouvel utilisateur
- Ajuster le .env et le lib/mypdo.php pour connecter non plus le serveur MySQL mais le serveur PostgreSQL
- Vérifier que vous arrivez bien à vous connecter avec l'utilisateur que vous avez renseigné dans la table mp_users

Exercice 1a - Premiers ajustements sur Etudiant

- Ajuster la classe Etudiant afin qu'elle corresponde à la structure de la table etudiant présente dans cette version sous PostgreSQL. (la liaison avec le user n'existe plus et les attributs sont différents).
- Ajuster les contrôleurs et vues de list et card (qui font de l'affichage) également
- Ajuster les contrôleurs et vues de add et edit (avec formulaire) également
- Ajuster enfin le delete (toujours en laissant la confirmation bien évidemment)

Exercice 1b - Déclinez sur les autres entités

- Faites de même pour les autres entités du projet (Enseignants, Matières, Modules, Epreuves)

Exercice 2 - Tables de classements

L'objectif est d'avoir facilement accès aux différents classements des étudiants ainsi que leurs moyennes, que ce soit de manière globale mais également dans les différentes matières.

2a. Classements des étudiants à partir d'une fiche spécifique

Ces informations seront accessibles depuis différentes interfaces :

- Sur epreuves → card.php, vous ajouterez sous la fiche le classement des étudiants concernés
- Sur matieres → card.php, vous ajouterez sous la fiche le classement des étudiants concernés toutes les epreuves confondues
- Sur modules → card.php, vous ajouterez sous la fiche le classement des étudiants concernés toutes les matieres confondues
- Sur etudiants → card.php, vous ajouterez sous la fiche le classement le cet étudiant dans chaque module concerné

2b. Classements à partir d'un élément général

Ces informations seront accessibles depuis différentes interfaces :

- Sur matieres → index.php, vous ajouterez les différents classements des étudiants :
 - Classement général regroupant toutes les épreuves de cette matière
 - Classement pour chaque épreuve appartenant à cette matière
- Sur modules → index.php, vous ajouterez les différents classements des étudiants :
 - Classement général regroupant toutes les matières de ce module
 - Classement pour chaque matière appartenant à ce module
- Sur etudiants → index.php, vous ajouterez les différents classements des étudiants :
 - Classement général de tous les étudiants par année, regroupant tous les modules de cette année

Exercice 3 - PL/PGSQL

Afin de déplacer le code au sein de la base de données et non plus dans votre code applicatif PHP, vous implémenterez les procédures suivantes :

1. Concernant l'insertion de nouvelles données : ajout_enseignant, ajout_etudiant, ajout_module, ajout_matiere, ajout_note
2. Concernant leurs mises à jours : modif_enseignant, modif_etudiant, modif_module, modif_matiere, modif_note

A noter que les dates de création et de mise à jour ne devront pas être passées en paramètres mais bien directement définies au sein de la procédure.

Enfin, vous ajusterez votre code applicatif PHP afin qu'il prenne en compte ces routines à la place des requêtes qui étaient précédemment exécutées.

Afin de pouvoir évaluer vos réponses de l'exercice 1, vous commenterez les lignes précédemment écrites pour qu'elles restent accessibles.

Exercice 4 - Optimisation des classements

Vous avez certainement constaté que les classements sont réalisés à partir de valeurs calculées (soit au sein de vos requêtes, soit au sein de votre code PHP), mais qu'ils sont donc relancés à chaque visite. c'est donc l'occasion de mettre en application des routines en PL/PGSQL pour éviter cette surconsommation de ressources.

L'objectif est donc de mettre en place les tables relatives aux différents classements, et de réaliser les routines (fonctions / procédures / triggers / ...) qui permettront de mettre à jour automatiquement les classements lorsqu'une nouvelle note est ajoutée.

Afin de pouvoir évaluer vos réponses à cet exercice, vous noterez le code de ces éléments dans un fichier ex4.sql présent dans votre projet.

Et bien évidemment, vous mettre également à jours vos différentes pages de classements afin qu'ils fassent maintenant appel à ces nouvelles tables.