

TP 3 MySQL : procédures et triggers

Objectifs

- Comprendre et manipuler les procédures stockées.
 - Automatiser certains traitements avec des déclencheurs.
 - Appliquer ces concepts dans un système de gestion de bibliothèque.
-
- Télécharger le script http://elmouelhia.free.fr/courses/mysql/script_biblio.sql
 - Démarrer la console MYSQL
 - Exécuter la commande suivante dans la console : `source chemindefichier/script_biblio.sql`
 - Vérifier que le script a été exécuté sans erreurs.
-

Considérons le modèle logique suivant pour une base de données de gestion de bibliothèque :

- auteurs(id, nom, pays)
- livres(id, titre, nb_exemplaires, #id_auteur)
- abonnes(id, nom, prenom, fonction)
- emprunts(#id_livre, #id_abonne, date_emprunt, date_retour)

Utiliser MySQL Stored Procedure Language pour répondre aux questions suivantes

- **Procédures stockées**
 - Q1.** Créer une procédure `lister_livres_auteur(nom_auteur VARCHAR(100))` qui affiche tous les livres écrits par l'auteur dont le nom est donné.
 - Q2.** Créer une procédure, `ajouter_auteur(nom VARCHAR(100), pays VARCHAR(100))`, qui ajoute un auteur dans la table `auteurs`.
 - Q3.** Créer une procédure `ajouter_livre(titre VARCHAR(200), nb_exemplaires INT, nom_auteur VARCHAR(100))` qui ajoute un livre. Si l'auteur n'existe pas, il doit être ajouté automatiquement.
 - Q4.** Créer une procédure `supprimer_abonne(id_abonne INT)` qui supprime un abonné uniquement s'il n'a aucun emprunt en cours.
- **Fonctions**
 - Q5.** Créer une fonction `nb_livres_par_auteur(id_auteur INT)` qui retourne le nombre de livres écrits par un auteur.
 - Q6.** Créer une fonction `nb_emprunts_abonne(id_abonne INT)` qui retourne le nombre total d'emprunts effectués par un abonné.
- **Curseurs**
 - Q7.** Créer une procédure `lister_retards()` qui affiche les noms des abonnés ayant des emprunts non encore retournés depuis plus de 30 jours (utilisation d'un curseur requise)
 - Q8.** Créer une procédure `statistiques_auteurs()` qui affiche pour chaque auteur le nombre de livres écrits et le nombre total d'exemplaires.

- **Triggers**

- Q9. Créer un trigger `verif_nb_exemplaires` qui empêche l'insertion d'un livre avec moins d'un exemplaire.
- Q10. Créer un trigger `mise_a_jour_emprunt` qui, après insertion d'un emprunt, décrémente le nombre d'exemplaires disponibles du livre concerné.
- Q11. Créer un trigger `retour_livre` qui, après mise à jour d'un emprunt (retour effectué), augmente le nombre d'exemplaires disponibles.

- **Bonus**

- Q12. Créer une fonction `emprunt_possible(id_abonne INT)` qui retourne `TRUE` si l'abonné a moins de 5 emprunts en cours, `FALSE` sinon.
- Q13. Créer une procédure `historique_emprunts_abonne(id_abonne INT)` qui affiche l'historique des emprunts (titre, date d'emprunt et de retour).