

TP8 : Modèle de données : cas BIBLIO

Procédures stockées, fonctions et déclencheurs sous MySQL

Activités du Référentiel :

A4.1.3 Conception ou adaptation d'une base de données
A4.1.7 Développement, utilisation ou adaptation de composants logiciels

Objectifs techniques:

- ✓ Développer au sein d'un SGBD
- ✓ Créer des procédures ou fonction stockées et les utiliser
- ✓ Créer des déclencheurs répondant à des contraintes d'analyse

Cours associé : FT4.1 Procédures stockées et fonctions
FT4.2 Déclencheurs

Contexte :

La base de données de travail est la base de données BIBLIO :

TYPE_LIVRE(IdType, LibelleT)
LIVRE (CodeLivre, Titre, Genre, DateEdition, IdType#)
EXEMPLAIRE (CodeLivre#, Numero, DateAchat)
AUTEUR (CodeAuteur, NomAuteur, Prenomauteur)
ECRIRE (CodeLivre#, CodeAuteur#)
ADHERENT (CodeAdherent, NomAdherent, PrenomAdherent, DateNaissance)
ADHERENT_JUNIOR(CodeAdherentJunior#, tousTypes)
ADHERENT_ADULTE(CodeAdherentAdulte#, centreInteret , email)
EMPRUNTER (CodeLivre#, Numero#, DateEmprunt, CodeAdherent#, DateRetour)
DROIT_JUNIOR (CodeAdherentJunior#, IdType#)

Les codes sont de type entier, les libellés et noms sont de type chaînes de caractères et les dates sont au format smalldatetime.

Type_Livre correspond à 'BD Adulte', 'BD Enfant', 'Roman', 'Documentaire', 'Magasine', 'Aventure'....

Les **adhérents de type Junior** n'ont pas forcément l'autorisation d'emprunter tous les types de livres. Le champ 'tousTypes' est à 1 si le junior a l'autorisation parentale d'emprunter tous types de livres sinon il est à 0 et dans ce cas les types autorisés pour chaque junior sont donc précisés dans DROIT_JUNIOR.

Les **adhérents de type Adulte** ont le droit d'emprunter tous les types de livres. On connaît de plus, éventuellement, leur centre d'intérêt principal afin de leur envoyer par mail régulièrement des propositions de livres pouvant les intéresser.

PARTIE 1 : Création de la Base de Données sous MySQL

Bien comprendre le schéma relationnel présenté ci-dessus : repérer les entités, d'éventuels héritages, entités faibles, repérer les associations et d'éventuelles contraintes d'unicité. Refaire un modèle conceptuel si cela vous aide.

Créer la BD présentée ci-dessus.

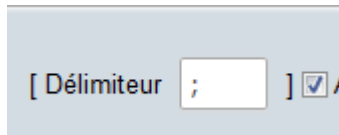
Y insérer quelques tuples créant ainsi votre jeu d'essais significatifs.

PARTIE 2 : Procédures et fonctions stockées

1. Créer des vues qui vous semblent utiles pour la suite du développement (simplification des requêtes) comme toutes les informations des adhérents junior ou adulte, toutes informations concernant la liste des droits pour les adhérents juniors...
2. Procédures et fonctions à créer : **N'oubliez pas qu'une procédure, une fonction ou une vue créée peut être appelée dans une autre procédure/fonction.**

Aides SQL :

- **Penser à modifier le délimiteur** ; à chaque création par | en bas de la zone de saisie SQL.



- Pour mettre des champs en minuscules : **LOWER(nomChamp)**
1. Créer une procédure qui donne le titre de tous les livres du type « roman ».
 2. Créer une procédure qui donne les caractéristiques d'un (ou des) auteur(s) dont le titre du livre est passé en paramètre.
 3. Ecrire une fonction qui retourne si un adhérent junior (dont le code est passé en paramètre) est autorisé à emprunter un type de livres (dont l'identifiant est passé en paramètre).
 4. Ecrire une procédure qui permet d'enregistrer un emprunt pour un adulte donné, toutes les informations nécessaires seront passées en paramètre.
 5. Créer une procédure qui donne les informations sur les adhérents qui ont un emprunt en cours après une date donnée en paramètre.
 6. Créer une fonction qui compte le nombre de livres.
 7. Créer une procédure qui compte le nombre de livres empruntés pour un adhérent dont le numéro sera passé en paramètre, ce nombre sera donné en paramètre de sortie.
 8. Créer une fonction qui affiche pour un adhérent dont le code est passé en paramètre, le nombre de livres non rendus.

PARTIE 3 : Déclencheurs

Pour chaque déclencheur créé, vous noterez les instructions qui vous permettent de le tester.

Ne pas oublier qu'un déclencheur peut appeler une procédure ou une fonction stockée.

1. Créer un déclencheur qui, lors de la suppression d'un adhérent, supprime tous les emprunts qui lui sont liés.
2. Créer un déclencheur qui, lors d'une insertion d'un adhérent vérifie la contrainte de partition XT entre ADHERENT_ADULTE et ADHERENT_JUNIOR.
3. Créer un déclencheur qui, lors de l'insertion d'un emprunt pour un junior, vérifie que le type du livre emprunté appartient bien à la liste des types autorisés pour ce junior.
4. Créer un déclencheur qui, à la mise à jour de la date de retour d'un emprunt, vérifie bien que cette date est cohérente par rapport à la date d'emprunt.
5. Créer un déclencheur qui, à l'insertion d'un emprunt, quel qu'il soit, vérifie la contrainte limitative de 7 livres maximum par adhérent par emprunt.

Créer une application de type Windows Form liée à cette base de données.

Vous gérerez la BD en mode connectée ou déconnectée (comme bon vous semble).

Créer une interface simple permettant juste le **listing de tous les emprunts (en cours et passés) et la gestion d'un nouvel emprunt ainsi que les retours.**

On doit trouver les fonctionnalités permettant de :

- Lister tous les emprunts
- Sélectionner parmi ces emprunts ceux en cours
- Enregistrer un nouvel emprunt pour tout type d'adhérent
- Gérer les retours (mise à jour de la date de retour)
- Si possible, lister tous les emprunts en retard (ou les gérer par un code couleur dans le listing de tous les emprunts).

On considère que les adhérents sont déjà enregistrés (issus d'une autre application de gestion des adhérents : à faire si vous avez le temps). De même, tous les livres proposés à l'emprunt existent déjà dans la base de données.

Vous testerez votre application en proposant un **plan de tests** (liste de tous les tests effectués – actions passées – commentaires), notamment pour mettre en avant la validité de vos déclencheurs.