

TD3 – Normalisation + LDD

Objectifs:

- Appliquer des règles génériques de validité de données (Normalisation)
- Utiliser MySQL pour la première fois
- Créer le squelette d'une base de données (LDD)

Partie 1. Normalisation

Exo 1

Soit la relation suivante avec les données concernées :

Appartenance (#idStagiaire, annee, anneeFormation, specialite)

L'attribut *idStagiaire* est une clé étrangère qui fait référence à l'identifiant du stagiaire dans une autre relation. L'attribut *annee* indique l'année du déroulement du stage comme 2022 ou 2023. L'attribut *anneeFormation* représente l'année de formation du stagiaire : 1^e_Annee, 2^e_Annee ou 3^e_Annee. L'attribut *specialite* est utilisé pour représenter toutes les différentes spécialités : Anatomie, dermatologie, ophtalmologie, etc., sachant que chaque année de formation a une liste de spécialités précises et chaque spécialité est proposée dans une seule année de formation. Exemple : un étudiant en 2^e année a le choix entre chirurgie, dermatologie et ophtalmologie.

Question : En quelle forme normale est cette relation ? Pourquoi ? Si elle n'est pas en 3^{ème} forme normale (3FN), expliquez les éventuels problèmes et proposez une décomposition qui permet de la rendre en 3FN.

Exo 2

La pédagogie par projets de Cy Tech permet aux élèves ingénieurs de réaliser différents types de projet pendant leur cursus. Voici les données concernées :

- Nom du projet : GL1, GL2, TIPE, PFE, ...
- Volume horaire du projet : le nom du projet permet d'identifier son volume horaire. Par exemple, le projet GL1 dure 22 heures, le projet PFE correspond à 175 heures.

- Matière : on connaît les matières associées à un projet. Par exemple le projet GL2 concerne les matières Analyse orientée objet et Programmation Java.
- Groupe : ING1, ING2, ...
- Nombre d'élèves : on connaît le nombre d'élèves de chaque groupe.
- Professeur : pour chaque projet, on associe à chaque groupe un professeur responsable.

Soit le MLD suivant :

Projet (nomP, volume, matieres)

ResponsableProjet (#nomP, groupe, nombreEleves, professeur)

Question1 : Listez toutes les dépendances fonctionnelles de ce MLD.

Question 2 : En quelle forme normale sont les relations de ce schéma ? Pourquoi ? Si elles ne sont pas en 3ème forme normale (3FN), proposez une décomposition qui permet de les rendre en 3FN.

Partie 2. LDD

Exo 1 – Prise en main de MySQL

Objectif : Création d'un utilisateur qui n'est pas root

1. Ouvrir un terminal
2. Vérifier que le service mysql est lancé

```
sudo service mysql status
```

S'il n'est pas lancé

```
sudo service mysql start
```

3. Lancer mysql en root

```
sudo mysql -u root -p
```

Le mot de passe ROOT par défaut est : cytech0001

4. Créer un nouvel utilisateur

```
CREATE USER 'user'@'localhost' IDENTIFIED BY 'password';
```

Ici, la commande crée un user qui a pour login user dont le mot de passe est password

A vous de choisir vos identifiants !

ATTENTION :

- les côtes -> ' sont obligatoires
- le point virgule -> ; est obligatoire

- Si vous avez l'erreur suivante à la création du USER

```
mysql> create user 'mki'@'localhost' identified by 'bdd23';
ERROR 1819 (HY000): Your password does not satisfy the current policy requirements
mysql> 
```

Solution mettre la politique des mdp à ZERO

```
SET GLOBAL validate_password.policy = 0;
```

RE essayer de créer un nouvel utilisateur

```
CREATE USER 'user'@'localhost' IDENTIFIED BY 'password';
```

Donner les privilèges au nouvel utilisateur

```
GRANT ALL PRIVILEGES ON *.* TO 'user'@'localhost';
```

Ici, la commande donne tous les droits d'accès au user d'identifiant user

5. Quitter MySQL et tester la connexion avec le nouvel utilisateur

```
mysql -u user -p
```

Tips !

- NE PAS OUBLIER LE USER QUE L'ON VIENT DE CRÉER
- Trouver un identifiant et mdp que l'on n'oublie pas. Vous serez amené à l'écrire en clair dans des fichiers de configuration, notamment en programmation web

Quelques commandes utiles !

- create database < nom de la bd >;
- show databases;
- use < nom de la bd >;
- show tables;
- describe < nom de la table >;
- create table < nom de la table >;
- source file.sql ; // pour exécuter le fichier sql

Exo 2 – Pokémon, du MLD vers la BDD

On utilise le MLD suivant :

Joueur (idJoueur, pseudonyme, sexe, niveau, #idEquipe)

Equipe (idEquipe, nom, couleur)

Pokemon (idPokemon, nom, espece, pointCombat, #idJoueur)

Emplacement (idEmplacement, latitude, longitude)

Apparition (#idPokemon, #idEmplacement, date, duree)

1. Écrire dans un fichier **pokemon.sql** le script permettant de créer la base de données << Pokemon >>. Exécuter le script.
2. Quels mots clés dois-je rajouter pour supprimer la base de données si elle existe déjà ?
3. Indiquer dans le script que vous allez utiliser cette base de données
4. Ajouter dans le fichier, le script permettant la création des tables à partir du MLD précédent. (Attention : ne pas oublier les clés primaires et étrangères)
5. Ajouter les instructions au script permettant de modifier le nom de la colonne date dans la table Apparition par horaire
6. Mettre comme date par défaut la date du système pour la table Apparition
7. Mettre 0 comme le niveau par défaut
8. Rendre obligatoire les informations niveau, latitude, longitude
9. Le pseudonyme est unique. Modifier la table Joueur.
10. Ajouter la contrainte : Point de combat ≥ 0
11. Ajouter la contrainte : Latitude est entre -90 et 90, longitude est entre -180 et 180
12. Supprimer la contrainte sur le pseudonyme.