
TD 03

Normalisation Langage de Définition de Données (LDD)

Partie I : Normalisation

Reprenons le cas de la **Gestion de séjours dans une chaîne d'hôtels** (TD2)

Question 1 : Normalisation du MLD

- (a) Le MLD que vous avez réalisé est-il en 1^{ère} forme normale ? Corriger les problèmes.
- (b) Expliquer pourquoi le modèle traduit mal la phrase **Chaque hôtel gère ses propres tarifs de consommation.**
- (c) Corriger les problèmes pour qu'il soit en 2FN.
- (d) Expliquer pourquoi le modèle n'est pas en 3FN.
- (e) Corriger le modèle pour qu'il soit en 3FN.
- (f) Proposer un MLD correspondant à votre modèle normalisé.

Partie II : LDD

Exercice 1 : Prise en main de MySQL

1. Lancer MySQL en ligne de commande.
 - (a) Sur Linux :
 - (i) `sudo service mysql start`
 - (ii) `mysql -u root -p`
 - (b) Sur Windows :
 - (i) `C :\wamp64bin\mysql\mysql15.7.14\bin >mysql.exe -u root`
2. Créer une nouvelle base de données :
`CREATE DATABASE pokemons ;`
3. Utiliser une base de données :
`USE pokemons ;`
4. Exécuter un script SQL. Vous pouvez taper directement vos ordres dans MySQL à la suite du prompt en validant chaque instruction par un ‘;’. Il est cependant préférable de les éditer dans un fichier séparé et les charger ensuite dans MySQL avec la commande :
`source chemin/pokemon_tables.sql ;`

Exercice 2 : Pokémons, du MLD vers la BDD

On utilise le MLD suivant :

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

Equipe(id, nom, couleur)

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

Emplacement(id, latitude, longitude)

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

1. Écrire dans un fichier **pokemon_tables.sql** le script de création des tables à partir du MLD précédent. Exécuter votre script au fur et à mesure pour tester.

2. Vous devez avoir retrouvé les tables créées avec le script. Afficher la description et le contenu de chaque table avec desc et select.

3. Insérer quelques données dans la table Equipe :

```
insert into Equipe values (1, 'Bravoure', 'Rouge');
```

```
insert into Equipe values (2, 'Sagesse', 'Bleue');
```

4. Vérifier maintenant que la table Equipe contient bien des données.

5. Vérifier la contrainte de clé primaire en ajoutant une équipe avec un identifiant déjà pris

```
insert into Equipe values (2, 'Intuition', 'Jaune');
```

6. Si vous constatez des problèmes avec la structure de vos tables, modifier votre script et recharger le. Pensez à supprimer vos tables au début du script pour tout reprendre à zéro.

7. Compléter votre script de création de table en vous assurant d'avoir les contraintes suivantes :

- Clés primaires

- Clés étrangères

- Ne pas permettre la suppression d'une équipe tant qu'un joueur la réfère
- Supprimer tous les joueurs d'une équipe quand cette dernière est supprimée

- Mettre comme date par défaut la date du système

- Mettre 0 comme le niveau par défaut

- Rendre obligatoire les informations personnage, niveau, latitude, longitude Un personnage est unique

- **Point de combat ≥ 0**

- Latitude est entre -90 et 90, longitude est entre -180 et 180