

Projet : Gestion d'un système de partage d'objets

IMPORTANT :

- Ce TP est à faire en équipe de **quatre**. Une fois votre équipe formée, venez me donner les noms de chaque membre.

Mise en situation

Votre tâche consiste à gérer un système de partage d'objets. Ceux-ci peuvent être de différentes catégories (vêtement, meuble, nourriture...) et chaque catégorie a des attributs différents (par exemple, on pourrait garder la taille et le matériau pour les vêtements et la date de péremption pour la nourriture). Les attributs et la quantité d'objets sont à votre discrétion, mais ils contribuent à la complexité du projet.

Un utilisateur de ce système peut être soit un annonceur d'objet à partager, soit une personne qui désire profiter du partage. Le partage d'objet peut être gratuit (un don) comme il peut être payant selon un prix précisé par l'annonceur (une vente). Le partage peut être définitif ou pour une période donnée.

Une personne intéressée par un type d'objet ou une marque en particulier peut demander que le système la notifie lorsqu'un objet correspondant vient d'être proposé pour un partage. Pour envoyer des notifications, le système prendra en considération des informations comme la proximité de l'objet par rapport à la personne intéressée.

Le but de ce projet est de vous faire réfléchir à comment conceptualiser une situation réelle pour la transformer en base de données. Ainsi, vous êtes encouragés à ajouter toute autre fonctionnalité qui vous semble nécessaire ou intéressante. Voici quelques exemples:

- Clavardage entre les membres
- Historique des annonces
- Historique des ventes d'un annonceur
- Évaluation des membres
- etc.

Tâche

Il s'agit, à partir de cette description et de votre propre investigation:

- de concevoir un modèle entité - association pour représenter le schéma de la base, réunissant entités et liens
- de concevoir à partir de là un modèle relationnel pour représenter le schéma de la base, et de le définir dans le langage de définition de données PostgreSQL
- de charger la base avec des données réalistes (chaque base devra comporter entre 50 et 100 occurrences)
- de concevoir une dizaine de requêtes en SQL. Le choix des requêtes est laissé libre, mais 5 d'entre elles devront faire intervenir au moins 4 relations. La complexité des requêtes est un facteur décisif.
- de réaliser une petite application qui doit englober les instructions SQL. Ici, vous disposez de deux choix :
 - Application (Python/Java/...) avec des commandes/boutons qui communiquent avec une base de données en PostgreSQL et qui affichent les résultats dans la console ou dans une interface.
 - Application web avec des pages, formulaires, boutons qui communiquent avec une base de données PostgreSQL et qui affichent le résultat dans le navigateur web.

À remettre

a) Rapport

- La représentation E-A de la base relationnelle accompagné d'explications
- Le schéma relationnel de la base
- Explications du code DDL
 - ❖ Exemple : comment avez-vous choisi vos clés primaires? Comment avez-vous géré les formats standards (téléphone,etc.)? Pourquoi avez-vous dupliqué tel attribut ou séparé telle table? Bref, expliquez les choix d'implémentation que vous avez faits.
- L'ensemble des requêtes en SQL et explication du résultat attendu
 - ❖ Exemple : 'Select * from toto' → Sélectionne toutes les colonnes et les occurrences de la table toto.
- Captures d'écran de l'application avec description de fonctionnement de l'application (mini-guide utilisateur)

b) Fichiers SQL

- La définition de la base de données en DDL (DDL.sql)
- L'ensemble des requêtes et l'application en SQL (LMD.sql)

c) Code source de votre application

Critères de correction

Les critères utilisés pour apprécier le projet sont

- la complexité et l'exactitude du schéma EA et du modèle relationnel (25)
- les explications dans le rapport (ddl et documentation de l'application) (15)
- la complexité, l'optimisation et l'exactitude des requêtes (35)
- la qualité de l'application (25)

Voici des liens pour vous aider

Java : https://www.tutorialspoint.com/postgresql/postgresql_java.htm

PostgreSQL JDBC : <https://jdbc.postgresql.org/>

Étapes pour le setup de JDBC et de la DB:

<http://www.postgresqltutorial.com/postgresql-jdbc/>

Python : https://www.tutorialspoint.com/postgresql/postgresql_python.htm (via psycopg2)

PHP : https://www.tutorialspoint.com/postgresql/postgresql_php.htm