

Etude de cas : Système de restauration

GOODFOOD est une société de restauration disposant de plusieurs restaurants dans de nombreuses villes et aéroports d'Europe. GOODFOOD veut informatiser son système d'information pour améliorer sa prestation de services auprès des clients. Les caractéristiques de ce système d'information se présentent comme suit :

Un restaurant possède une ou plusieurs salles dans lesquelles sont disposées plusieurs tables. Les clients du restaurant s'installent généralement à une table, et passent une commande. La commande d'une table comprend les plats commandés et la quantité associée à chaque plat. Pour pouvoir gérer efficacement le service, on suppose qu'à une table donnée correspond une commande. Pour organiser le service, le patron du restaurant a proposé qu'une table ne soit servie que par un seul serveur. Afin de placer le maximum de clients, on connaît le nombre maximal de places par table. Un serveur qui travaille dans le restaurant est affecté à une ou plusieurs tables pour toute une journée. Il existe plusieurs catégories de serveurs : serveur, maître d'hôtel, chef de salle, ... Chaque serveur a un nom et un prénom. Les commandes des clients du restaurant portent sur un ou plusieurs plats. Afin de pouvoir rapidement envoyer la commande à la cuisine, et pouvoir également sortir rapidement l'addition, on conservera la date et l'heure de la commande, la date et l'heure de l'encaissement, le mode de paiement, le montant total, et le nombre de personnes. Un plat d'une commande est décrit par un libellé, un type (entrée, viande, poisson, dessert, ...) et un prix unitaire.

Travail demandé

- Modéliser les données de GOODFOOD par un schéma de relations. Pour ce faire, vous déterminerez les dépendances fonctionnelles, vous calculerez la couverture minimale, puis vous concevrez les relations en 3 FN.
- Créer les tables ainsi que le jeu de données. Pour cela, vous écrirez les requêtes SQL pour la création des tables (incluant les contraintes d'intégrité) et l'insertion des enregistrements.
- Ecrire une application en Java permettant de manipuler la base de données de GOODFOOD via l'API JDBC. L'application doit implémenter les fonctionnalités suivantes :
 1. Détermination de la liste des plats (numéro et nom du plat) servis à une période donnée (date début, date fin).
 2. Affichage de la liste des plats (numéro et nom du plat) qui n'ont jamais été commandés pendant une période donnée (date début, date fin).
 3. Etablissement de la liste serveurs (nom et date) qui ont servi une table donnée à une période donnée (date début, date fin).
 4. Affichage (dans un ordre décroissant) du chiffre d'affaire et le nombre de commandes réalisés par chaque serveur (nom, chiffre d'affaire, nombre de commandes) en une période donnée (date début, date fin).

5. Affichage de la liste des serveurs (nom) n'ayant pas réalisé de chiffre d'affaire durant une période donnée (date début, date fin).
6. Calcul du montant total d'une commande donnée (numéro de commande) et la mise à jour de la table COMMANDE.
7. Création et destruction de scripts SQL pour des triggers. A ce titre, vous écrierez les deux triggers suivants :
 - a. Enregistrement dans une table **auditer** toutes les commandes servies par un maître d'hôtel et dont le montant par personne est inférieur à 15 euros. La table **auditer** contient les informations suivantes : numéro de la commande, numéro de la table, date de commande, nombre de personnes, montant de la commande.
 - b. Vérification de la contrainte suivante : la quantité d'un plat commandé ne doit pas dépasser le nombre de personnes indiqué dans la commande.

Travail à rendre

Le projet est à réaliser en binôme. Il faut nous rendre les fichiers suivants :

1. Le script SQL pour créer les tables ainsi que les données
2. Le(s) programmes(s) java bien commenté(s) avec des exemples d'utilisation pour les fonctionnalités décrites ci-dessus.

Mettre tous les fichiers dans un répertoire portant vos noms, et compresser ce répertoire au format ZIP dont le nom de fichier doit suivre le format suivant :

Groupe_NomBinôme1_NomBinôme2.zip

Enfin, envoyer le fichier compressé à votre enseignant.