

Exercice n° 1 : Conception de bases de données

Une chaîne de restaurants sur tout le territoire national, veut gérer automatiquement la gestion de ses établissements.

Cette chaîne possède des restaurants dans toutes les grandes villes du pays. Chaque restaurant est caractérisé par un numéro unique (NUMREST), et une adresse (ADREREST). Il possède un ensemble de tables. Une table est caractérisée par un numéro (NUMTAB), sa capacité (NBPlace) (4, ou 6 couverts par exemple), la ZONE où elle est située (Fumeur ou non fumeur). Deux tables dans des restaurants différents peuvent avoir le même numéro.

Un restaurant propose à ses clients un service de réservation de table. Une table, au cours d'une même soirée, peut être utilisée plusieurs fois par plusieurs services à des périodes différentes (ainsi, on peut distinguer le premier service de 20h à 21h, le deuxième service de 21h à 22h etc.). Un service est identifié par un numéro unique (NUMSERVICE) et désigne l'ensemble des repas pris par des clients à une table durant la même période (PERIODE). Un service donne lieu à une addition et une seule.

On voudrait retrouver tous les produits consommés par les clients au cours d'un service ainsi que leurs quantités afin de pouvoir effectuer l'addition.

Le client qui est enregistré dans la base de données est celui qui réserve et qui paye l'addition. Il est caractérisé par son numéro unique (NUMCLI), son nom (NOMCLI), son adresse (ADRECLI) et son numéro de compte (NUMCOMPTE).

Les clients choisissent des plats et consommations sur une carte de produits (on appellera indifféremment produit un plat, une boisson un dessert etc.). Un produit est caractérisé par son numéro unique (NUMPROD), son nom (NOMPROD), son prix (PRIXP).

Lorsqu'un client fait une demande de réservation par téléphone, on génère une réservation caractérisée par un numéro unique (NUMRES), la date de réservation (DATERES), le restaurant, la table et la période réservées.

Question 1 : En appliquant l'algorithme de synthèse pas à pas, construire le schéma relationnel en 3FN de cette étude de cas.

On veut maintenant enrichir le modèle obtenu en tenant compte d'une description plus précise de la carte des produits. La carte est constituée de plusieurs catégories de produits : Plats, Desserts, Boissons et Formules :

- Un plat est décrit par son nom, son prix et un libellé qui le décrit.
- Un dessert est décrit par son nom, son prix et sa catégorie (fruit, crème, glace etc.).
- Une boisson est décrite par son nom, son prix et sa nature (boisson gazeuse, non gazeuse etc.)
- Une formule est une composition d'un plat, d'un dessert et d'une boisson. Chaque formule a un prix fixe.

Question 2 : Proposez une amélioration du schéma obtenu dans Question 1 en rajoutant des dépendances fonctionnelles ainsi que les nouvelles relations qui en résultent.

Vous pouvez ajouter des attributs, si nécessaire.

Le schéma final doit être en 3ème forme normale.

Exercice n°2 : Langages relationnels

Considérons la base de données prenant en charge une gestion des prêts consentis auprès des clients de la Banque Bladi (**BB**):

- **Client**(code, nom, prénom, adresse, daten, lieu, tel, email, datei, typc) : Un client, auprès de BB, doit être connu par les informations telles nom, prénom, adresse, date et lieu de naissance(daten, lieu), téléphone (tel), email ainsi sa date inscription (datei) et un type client (typc). Il a un code unique (code).
- **Compte**(numéro, code, montant) : un compte bancaire est identifié par numéro, concerne un client (code) et lui correspond un solde (montant).
- **Prêt**(codp, code, montp, typp, datp, deb, fin) : Le prêt (codé par codp) est consenti à un client (code) avec un montant (montp) à une date (datp). Ce prêt a un type (typp) et a une période de remboursement entre date debut (deb) et date fin (fin).
- **Garantie**(codp, codb, datb, mont_est) : le prêt (codp) est garanti par des biens (codb) à des dates fixées (datb) avec des montants estimés (mont_est).
- **Echéancier**(codp, datpr, montpr, datpa, montpa): L'échéancier de remboursement suit un calendrier fixe. A une date donnée (datpr), on doit rembourser le prêt (identifié par codp) en versant un montant (montpr). On notifie le montant du client remboursé (montpa) avec la date correspondante (datpa).
- **Type_prêt**(typp, minp, maxp, durp): les types de prêt offerts par la banque BB (typp) ont un montant compris entre un montant minimum (min) et un montant maximum (max) avec une durée de remboursement (durp).
- **Bien**(codb, desb, defb, codc, ancb, surb, haub, dabb, montb) : Chaque bien identifié par un code (codb), appartenant à un client (de code codc), est décrit par les informations telles la désignation (desb), un descriptif (defb), l'ancienneté (anch), la surface (surb), la hauteur (haub), la date d'évaluation du bien (dabb) ainsi que le montant correspondant (montb).

Les attributs code et codc sont du même domaine

a) Exprimer les requêtes suivantes en langage Algébrique :

1. Quels sont les clients (code, nom et prénom) qui sont de même type que le client de code C1
2. Quels sont les clients (code) dont tous les biens sont sous garantie

b) Exprimer les requêtes suivantes en langage Prédicatif :

1. Quels sont les clients (code et nom) qui ont effectué tous les types de prêt du client de code C1
2. Quels sont les clients (code et nom) qui n'ont effectué aucun prêt

c) Exprimer les requêtes suivantes en SQL :

1. On voudrait vérifier que le montant d'un prêt ne dépasse pas le montant total estimés des biens sous garantie pour ce prêt. Ecrire la requête qui donne les prêts (codp) dont le montant total estimé des biens sous garantie est inférieur au montant du prêt.
2. Créer une vue et l'utiliser pour donner les clients (code et nom) dont le montant total de leurs comptes bancaires est le plus élevé.

Bon courage