

Devoir #1: Partie I — Fondements

Date de remise: Vendredi 7 Oct à 6h00 (dans la boîte aux devoirs).

Valeur: 5% de la note finale.

Suggestion: Consulter les exemples du manuel.

Ce devoir porte sur les designs conceptuel et logique ainsi que sur les requêtes SQL. Certains des exercices viennent (légèrement modifiés) du manuel.

1. – **40%**. Dans cet exercice vous devez faire un design d'une base de données pour une banque dont le nom est *Main Savings Inc.*. Notez que tous les aspects d'un design réaliste ne sont pas pris en compte ici. Vous allez seulement prendre en compte un certain nombre d'aspects qui illustrent quelques uns des concepts importants du design des diagrammes ER vus en classe. La base de données de *Main Savings Inc.* a les exigences suivantes:

- Main Savings Inc. est organisée en plusieurs branches. Chaque branche a un nom unique et est localisée dans une cité bien définie. Bien plus, chaque branche a un actif.
- Les clients de Main Savings Inc. ont un identificateur unique, un prénom et un nom de famille, une adresse et une cité de résidence. Chaque client doit avoir un employé de la banque qui lui sert de banquier. Ce dernier supervise à la fois les emprunts effectués par le client et la possession des comptes par ce dernier. Ce rôle de supervision a un type (par exemple "banquier personnel" ou "responsable des prêts").
- Les employés de Main Savings Inc. sont identifiables par un numéro d'identification. De plus, la banque stocke le prénom ainsi que le nom de famille de l'employé, son numéro de téléphone et sa date de début d'emploi. Main Savings Inc. garde aussi les traces de quel employé travaille pour quel manager.
- Main Savings Inc. offre des prêts à ses clients. Un prêt est initié dans une branche bien spécifique et peut être octroyé à un ou plusieurs clients. Chaque prêt a un numéro d'identification unique ainsi qu'un montant. Main Savings Inc. garde les traces des remboursements comme suit: un numéro de remboursement identifie chaque remboursement de chaque prêt; chaque prêt doit avoir un tel numéro de remboursement ainsi qu'une date et le montant du remboursement.
- Main Savings Inc. offre des comptes d'épargne ainsi des comptes courants à ses clients. Chaque client peut avoir plus qu'un compte de chaque type et chaque compte peut avoir plus d'un propriétaire. Chaque compte est identifié par un numéro de compte et a une balance ainsi que la dernière date à laquelle chacun des propriétaires y a accédé. Finalement, chaque compte d'épargne a un taux d'intérêt et chaque compte courant a une indication de la hauteur du découvert permis.

Sur base des informations ci-dessus, faites ce qui suit:

- (a) Dessinez un diagramme ER pour la base de données de Main Savings Inc.. Ce faisant, indiquez les divers attributs de chaque ensemble d'entités et de relations. Spécifiez les contraintes de clé et participation pour chaque ensemble de relations. Spécifiez aussi toute contrainte de recouvrement nécessaire.

- (b) Traduisez votre diagramme ER en un schéma relationnel montrant les instructions SQL nécessaires pour créer les relations du modèle relationnel. N'utiliser que les clés et les contraintes de domaines *null*. Toute contrainte du diagram ER que vous ne pouvez exprimer en SQL doit être expliquée.

2. **(Exercice 5.2 modifié) – 30%**: Considérez les relations suivantes dont la signification est claire (Les clés primaires sont soulignées):

Suppliers(*sid* : **int**, *sname* : **string**, *address* : **string**, *rating* : **real**)

Parts(*pid* : **int**, *pname* : **string**, *color* : **string**)

Catalog(*sid* : **int**, *pid* : **int**, *cost* : **real**)

Sur base de ce schéma, formulez les requêtes suivantes **à la fois** en algèbre relationnelle et en calcul relationnel des domaines:

- (a) Trouvez les noms des pièces pour lesquelles il y a quelques fournisseurs.
- (b) Trouvez les noms des fournisseurs qui fournissent toutes les pièces rouges.
- (c) Trouvez les ids des pièces fournies par au moins deux fournisseurs différents.
- (d) Trouvez les ids des fournisseurs qui fournissent une pièce verte et une pièce rouge.
- (e) Trouvez les ids des fournisseurs qui fournissent seulement des pièces rouges.

Exprimez les contraintes d'intégrité suivantes en SQL (comme un CHECK, une contrainte de domaine, de clé primaire ou étrangère, ou comme une assertion) ou comme un trigger.

- (a) Les fournisseurs doivent avoir un niveau (= "rating") minimum de 5.
- (b) Chaque pièce a un minimum de 2 et un maximum de 5 fournisseurs.
- (c) Chaque fournisseur doit fournir la pièce dont le id est P111.
- (d) Chaque fois que le niveau d'un fournisseur est augmenté, les coûts des pièces fournies par ce fournisseur subissent aussi une augmentation de 10%. (Suggestion: Utilisez un trigger.)

Finalement, exprimez les requêtes suivantes en SQL:

- (a) Pour chaque pièce, trouvez le nom du fournisseur qui charge le plus pour cette pièce.
- (b) Trouvez les ids des fournisseurs qui fournissent seulement des pièces rouges.
- (c) Pour chaque fournisseur qui fournit seulement des pièces vertes, donnez le nom de ce fournisseur ainsi que le nombre total des pièces que le fournisseur fournit.
- (d) Trouvez les ids des fournisseurs qui fournissent une pièce verte et une pièce rouge.
- (e) Trouvez les noms des fournisseurs qui fournissent toutes les pièces rouges.

3. **(Exercices 2.8 et 3.18) – 30%**: Voir manuel pages 55 et 97.