

Nom : \_\_\_\_\_

Prénom : \_\_\_\_\_

Groupe: \_\_\_\_\_

Examen de Bases de données relationnelles  
15 Décembre 2008 – Durée : 2 heures

Note :

Barème:

1		6
2		4
3		4
4		6

## 1 Agence Immobilière

On considère une base de données relationnelle d'une agence immobilière. Cette base est constituée des tables suivantes (les attributs de la clé primaire de chaque table ont été soulignés):

- *Maison* (id, prix\_demande, adresse, code\_postal, chambres, sdb, surface, idVendeur)

Cette table décrit les maisons à vendre ou ayant été vendues par l'agence :

- prix\_demande est le prix initialement demandé par le vendeur (idVendeur)
- adresse, code\_postal localisent la maison
- le nombre de chambres est chambres, le nombre de salles de bains sdb, et la surface surface.

- *Agent*(id, nom, telephone, email)

Cette table décrit les agents immobiliers de l'agence.

- *Vendeur*(id, nom, telephone, email, idAgent)

Cette table décrit les vendeurs présents, ou passés d'un bien via cette l'agence.

- *Acheteur*(id, nom, telephone, email, idAgent)

Cette table décrit les acheteurs clients de l'agence

- *Vente*(idMaison, idAcheteur, dateVente, prix\_vente)

La table Vente renseigne l'identité du bien vendu, celle de l'acheteur, la date de la vente et le prix de vente qui peut différer du prix initialement demandé.

1. Exprimez les requêtes suivantes en **algèbre relationnelle** :

- Déterminer les identifiants, adresses, prix\_demande, prix\_vente pour toutes les maisons ayant été vendues pour une somme inférieure au prix initialement demandé

.....  
.....  
.....  
.....  
.....  
.....  
.....

- Déterminer les noms de tous les acheteurs potentiels n'ayant encore rien acheté

.....  
 .....  
 .....  
 .....

## 2. Exprimez les requêtes suivantes en **SQL**

- Pour tous les codes postaux pour lesquels au moins trois ventes ont été effectuées, afficher code\_postal et prix\_vente moyen

.....  
 .....  
 .....  
 .....  
 .....  
 .....

- Trouver les adresses et les prix\_demande de toutes les maisons ayant au moins 3 chambres et deux salles de bains n'ayant pas été vendues. Chaque couple (adresse, prix\_demande) doit être affiché une fois seulement.

.....  
 .....  
 .....  
 .....  
 .....  
 .....

## 2 Donation

Pour stocker les informations concernant des donations faites par des organisations à des candidats à une élection, la table suivante a été créée:

```
CREATE TABLE donations
(nomReceveur CHAR(20) NOT NULL,
 organisationDonnatrice CHAR(20) NOT NULL,
 typeOrganisation CHAR(20),
 montant REAL,
 PRIMARY KEY (nomReceveur, organisationDonnatrice)
)
```

Pour chacune des 4 paires de requêtes SQL suivantes, dire si les requêtes sont équivalents, c'est à dire si elle renvoient le même résultat quelle que soit l'instance de la table donations. Si c'est le cas, un simple oui suffira, mais si ce n'est pas le cas, donner un exemple d'instance de la table donations pour laquelle les résultats diffèrent.

1.   a SELECT DISTINCT nomReceveur FROM donations A  
       WHERE NOT EXISTS  
         (SELECT B.organisationDonnatrice FROM donations B  
               WHERE nomReceveur = 'Campbell'  
               AND A.organisationDonnatrice <> B.organisationDonnatrice)  
   b SELECT DISTINCT nomReceveur FROM donations A  
       WHERE NOT EXISTS  
         ( (SELECT organisationDonnatrice FROM donations  
               WHERE nomReceveur = 'Campbell' )  
       EXCEPT  
         (SELECT organisationDonnatrice FROM donations B  
               WHERE B.nomReceveur = A.nomReceveur))  
   .....  
   .....  
   .....  
   .....  
   .....  
   .....  
   .....
  
2.   a (SELECT DISTINCT nomReceveur FROM donations  
       WHERE montant >= 500)  
       UNION  
       (SELECT DISTINCT nomReceveur FROM donations  
       WHERE montant < 500)  
   b SELECT DISTINCT nomReceveur FROM donations  
   .....  
   .....  
   .....  
   .....  
   .....  
   .....  
   .....
  
3.   a SELECT DISTINCT A.nomReceveur FROM donations A, donations B  
       WHERE A.montant >= 1000  
           AND A.nomReceveur = B.nomReceveur  
           AND A.organisationDonnatrice <> B.organisationDonnatrice  
   b SELECT DISTINCT nomReceveur FROM donations  
       WHERE montant >= 1000  
       GROUP BY nomReceveur  
       HAVING COUNT(organisationDonnatrice) >=2  
   .....  
   .....  
   .....  
   .....

.....  
 .....  
 .....

4.   a SELECT DISTINCT nomReceveur FROM donations  
       WHERE nomReceveur NOT IN  
       (SELECT nomReceveur FROM donations  
       WHERE typeOrganisation = 'tabac')
- b SELECT DISTINCT nomReceveur FROM donations A  
       WHERE EXISTS  
       (SELECT \* FROM donations B  
       WHERE B.nomReceveur = A.nomReceveur  
       AND typeOrganisation <> 'tabac')

.....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....

### 3 Formes normales

On donne la relation :  $R(A, B, C, D, E, F)$  et l'ensemble de dépendances fonctionnelles :  
 $DF1 = (A \rightarrow B; DE \rightarrow F; B \rightarrow C)$

1. Déterminer l'ensemble des clés de R (en justifiant)

.....  
 .....  
 .....  
 .....  
 .....

2. Si R n'est pas en 3NF, la décomposer en 3NF

.....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....

3. Même questions avec l'ensemble de dépendances :

$DF2 = (AB \rightarrow C; CD \rightarrow A; C \rightarrow EB; D \rightarrow F)$

.....

.....

.....

.....

.....

.....

.....

.....

.....

# 4 Division

R est une relation définie sur {X, Y,Z, T}, S est une relation définie sur {X, Y }. Les instances respectives de R et S sont :

X	Y	Z	T	
x1	y1	z1	t1	
x1	y1	z1	t2	
x1	y2	z1	t1	
x2	y2	z1	t1	
x2	y1	z1	t1	
x1	y1	z2	t1	
x1	y1	z2	t2	
x2	y2	z2	t1	

X	Y
x1	y1
x1	y2
x2	y2

1. Ecrivez en SQL la requête  $\Pi_{X,Y,Z}(R) \div S$

.....

.....

.....

.....

.....

.....

.....

.....

.....

2. Avec les instances ci-dessus quelles sont les valeurs renvoyées par les requêtes :

(a)  $\Pi_{X,Y,Z}(R) \div S$

.....

.....

(b)  $\Pi_{X,Y,Z}(R) \div \Pi_X(S)$

.....

.....

3. Que renvoient les requêtes suivantes lorsqu'elles sont exécutées sur l'instance ci dessus.

(a) CREATE VIEW V1 AS SELECT X,Y,Z FROM R;  
SELECT COUNT(\*) , Z FROM V1 GROUP BY Z;

.....  
.....

(b) CREATE VIEW V2 AS SELECT DISTINCT X,Y,Z FROM R;  
SELECT COUNT(\*) , Z FROM V2 GROUP BY Z;

.....  
.....

(c) CREATE VIEW V3 AS SELECT DISTINCT X,Y FROM S;  
SELECT Z FROM V1 GROUP BY Z  
HAVING COUNT(\*) = ( SELECT COUNT(\*) FROM V3 );

.....  
.....  
.....

(d) SELECT Y,Z FROM V2 GROUP BY Y,Z  
HAVING COUNT(\*) = (SELECT COUNT(DISTINCT X) FROM S);

.....  
.....

(e) la requête 3.c est supposée traduite en SQL la division 2.a, la requête3.d est supposée traduire la division 2.b. L'une de ces traductions est correcte, laquelle? Quelle est la raison qui fait échouer l'autre.

.....  
.....  
.....  
.....