

## TP de Bases de Données

On considère une base de données relationnelles pour un grand magasin. Cette base est composée des relations suivantes :

**PROD1**(NP1, DESCR1, PRIX1)  
**PROD2**(NP2, DESCR2, PRIX2)  
**E1**(NE, NOM, ADR, RAYON)  
**E2**(NE, DEB, FIN, SALAIRE)  
**MAG**(RAYON, ETAGE, DIR)  
**VENTES**(NP, RAYON)

La signification de ces relations est la suivante :

PROD1(n, d, p) : le produit de numéro n (unique) a pour description d et pour prix p  
 PROD2(n, d, p) : le produit de numéro n (unique) a pour description d et pour prix p  
 E1(e,n,a,r) : l'employé de numéro e (unique) a pour nom n, pour adresse a et travaille au rayon r  
 E2(e,d,f,s) : de la date d à la date f, l'employé de numéro e a eu s pour salaire (si f = nil, il s'agit de son salaire actuel)  
 MAG(r,t,d) : le rayon de nom r est situé à l'étage t et a pour directeur l'employé de numéro d  
 VENTES(n,r) : le produit de numéro n est vendu au rayon de nom r

**1<sup>ère</sup> partie** : vérification de contraintes :

On souhaite par ailleurs que les conditions suivantes soient vérifiées :

- (C1)  $PROD1[NP1] \cap PROD2[NP2] = \emptyset$   
 (C2)  $E1[NE] = E2[NE]$   
 (C3)  $E1[NE] \supseteq MAG[DIR]$   
 (C4)  $VENTES[RAYON] = MAG[RAYON]$   
 (C5)  $(PROD[NP1] \cup PROD2[NP2]) \supseteq VENTES[NP]$   
 (C6)  $MAG(r, et, d) \Rightarrow \exists n, \exists a$  tels que  $E1(d, n, a, r)$

Pour chacune de ces contraintes ainsi que pour les clés (attributs uniques), écrire la (ou les) requêtes SQL permettant de vérifier si elles sont respectées.

Donner les n-uplets ne respectant pas chacune de ces contraintes

Les éliminer de la base en utilisant la commande SQL :

Delete from R where ...

Exemple : pour éliminer le produit 12 de la relation PROD1

Delete from PROD1 where NP1 = 12

**2<sup>ème</sup> partie** : Donner en SQL une expression pour les questions suivantes :

- Q1. Numéros des produits vendus par tous les rayons situés au premier étage  
 Q2. Numéros des produits vendus par les rayons se trouvant au même étage que le rayon 3  
 Q3. Numéros, descriptions et prix des produits vendus par les rayons se trouvant au même étage que le rayon 3  
 Q4. Produits qui ne sont pas en vente  
 Q5. Produits vendus à tous les rayons  
 Q6. Rayons vendant tous les produits de la relation PROD1  
 Q7. Rayons n'ayant qu'un employé  
 Q8. Donner la formulation SQL pour calculer les n-uplets de la relation DESSOUS définie par  
 DESSOUS(r1, r2) le rayon r1 est situé au dessous du rayon r2  
 Q9. Créer la relation DESSOUS de la façon suivante :  
 CREATE TABLE DESSOUS(r1, r2, primary key(r1,r2))

As {votre requête de la question Q8}

Vérifier que la relation a bien été créée :  
describe DESSOUS

Vérifier qu'elle contient les bonnes informations

En cas de problème, vous pouvez la détruire :  
Drop table DESSOUS  
Et recommencer sa construction

Pour la relation E2, on considère que l'on a maintenant les deux contraintes suivantes  
(C9)  $\forall e$ , cardinalité((E2 :NE=e))  $\leq 2$   
(C10)  $E2(e, d, f, s)$  ET  $f \neq Nil \Rightarrow \exists s' E2(e, f, nil, s')$

Exprimer en SQL les questions suivantes (les contraintes C1 à C10 sont vérifiées) :

- Q11. Pour chaque employé augmenté, son augmentation depuis son arrivée dans l'entreprise
- Q12. Salaires actuels des employés qui dirigent des rayons où l'on vend des produits de prix  $> 20$
- Q13. Employés gagnant actuellement plus que leur directeur
- Q14. Pour chaque rayon, les numéros, descriptions et prix des produits
- Q15. Moyenne des salaires par rayon

Q16. Créer la relation  
 $PROD(NP, DESCR, PRIX) := PROD1 \cup PROD2$   
à la place de PROD1 et PROD2

Q17. Compte tenu de cette nouvelle relation, comment trouver les directeurs de rayons où l'on vend les produits les plus chers ?