

GIS 2^{ème} année - Devoir Surveillé S.G.B.D.

© Polytech'Lille - Olivier Caron

Durée 2 heures - Documents autorisés

1 Intégrité des données (12 points)

Soit la base de donnée relationnelle suivante :

```
create table facture (
  num_facture integer primary key,
  date_facture date not null,
  prix_total float default 0.0) ;

create table item_facture (
  num_item integer not null,
  designation text not null,
  ref_facture integer references facture on delete cascade,
  prix_item float not null,
  primary key(num_item, ref_facture)) ;
```

Une facture se décompose en un certain nombre d'items. L'information `price_total` de la table `facture` est une donnée calculée : elle correspond à la somme des prix des items correspondants à la facture.

Question : Ecrire un trigger qui garantit la cohérence de la base pour cette donnée calculée lors d'une quelconque modification des valeurs de la table `item_facture` (remarque : on considère qu'on ne modifie jamais la valeur d'une clé). Spécifiez toutes les commandes SQL nécessaires à la mise en place de ce trigger

Soient les commandes SQL suivantes :

```
1 : insert into facture values (1, now(), 0.0) ;
2 : insert into facture values (2, now(), 0.0) ;
3 : insert into item_facture values (1,'bidon1', 1, 10.0) ;
4 : insert into item_facture values (2,'bidon2', 1, 12.0) ;
5 : insert into item_facture values (1,'bidon2', 3, 12.0) ;
6 : insert into item_facture values (1,'bidon2', 2, 12.0) ;
7 : insert into item_facture values (3,'bidon3', 1, 20.0) ;
8 : insert into item_facture values (1,'bidon4',2,30.0) ;
9 : update item_facture set price_item=10 where num_item=1 ;
10: delete from facture where num_facture=2 ;
```

Question : Décrire en une ou deux lignes maximum les commandes SQL qui ne seront pas exécutées.

Question : Quel est l'état de la base après l'exécution de ces commandes ? (on considère que le trigger précédent est activé)

2 Composition Florale (8 points)

Une grande boutique de fleurs désire informatiser la création de bouquets (vive le progrès ?). Elle dispose d'une base de données qui regroupe les informations suivantes :

- Une fleur est identifiée par un numéro, décrite par un nom, un prix, une couleur et une date limite de vente et appartient à un type de fleur précis.
- Un type de fleur est identifié par un numéro et est caractérisé par un libellé (ex : 'rose', 'tulipe', etc). Un type de fleur peut se marier avec un ou plusieurs autres types de fleur.
- Un bouquet est identifié par un numéro et contient un ensemble de fleurs. Le prix du bouquet est la somme du prix des fleurs

Le schéma conceptuel correspondant à ce système d'information est conforme à celui décrit dans la figure 1.

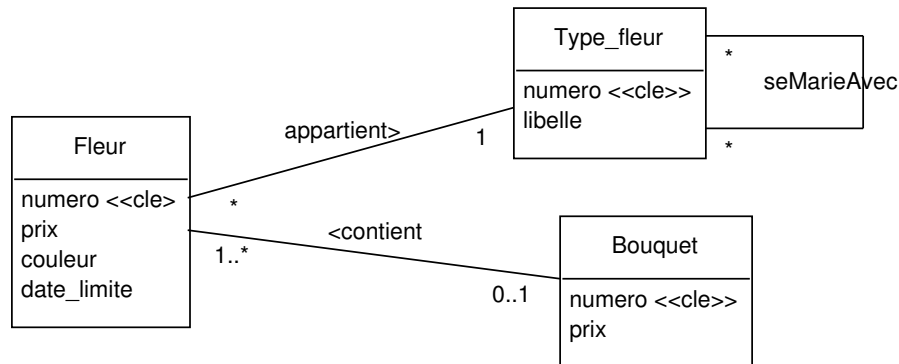


FIG. 1 – le schéma conceptuel

Ci-dessous, le schéma relationnel conforme au schéma conceptuel (les attributs préfixés par `ref_` correspondent aux clés étrangères).

```

table fleur ( numero, nom, prix, couleur, date_limite, ref_type)
table type_fleur (numero, libelle)
table mariage (ref_type1, ref_type2) ;
table bouquet (numero, prix) ;
table element_bouquet (ref_bouquet, ref_fleur) ;
  
```

Question : Ecrire la méthode Java `verifieBouquet(int codeBouquet)` qui vérifie si le bouquet est correctement constitué : c'est à dire si chaque fleur du bouquet peut se marier avec les autres. Si c'est le cas, cette méthode positionne le champ `prix` de la table `bouquet`, sinon affiche un message d'erreur.

Remarque : la connection à la base de données est déjà réalisée et cette méthode dispose des variables d'instances initialisées `db` de type `java.sql.Connection` et `stmt` de type `java.sql.Statement`.