

Base de données

| Contraintes d'intégrité | bla |
|-------------------------|---|
| Catégorie → Produit 1,n | Catégorie(id, libelle) |
| Produit → Catégorie 1,1 | Produit(id, #idCategorie, #Ligne) |
| Produit → Ligne 1,n | Ligne(numero, idCommande, idProduit, #quantite) |
| Ligne → Produit 1,1 | Commande(id) |
| Ligne → Commande 1,n | |
| Commande → Ligne 1,1 | |

Réponse aux questions

1. Comment est déclarée la table assurant la persistance d'un objet ?\

Grâce à "@Entity"

2. Comment est déclaré le mapping entre un membre de la classe et un champ de la table relationnelle ?\

Grâce à "@Table(name="className")"

3. Comment est déclarée la clé primaire de la table ?\

Grâce à "@id"

4. Quelles sont les possibilités de déclaration des clés primaires ?\

"@GeneratedValue(strategy=GenerationType.IDENTITY)" est l'auto-incrément de la clé primaire.

5. Réaliser un tableau montrant la correspondance de type (entier, chaîne, etc.) entre les propriétés d'une classe et les champs d'une table.\

Tableau comparatif des valeurs entre JAVA et SQL

| | Java | SQL |
|---------------------|--------|---------|
| Entier | int | int |
| Chaîne de caractère | String | varchar |
| Flottant | Float | Float |

Schéma d'explication de la bidirectionnalité de deux classes

6. Montrez à l'aide d'un schéma (par ex. deux classes liées au dessus de deux tables liées) comment se paramètre le lien bidirectionnel entre deux classes (en spécifiant les éléments à fournir dans les annotations)



Programme de Test

Analysez le code du programme et répondez aux questions en vous aidant au besoin de la documentation :

1. À quoi correspond la méthode `persist()` ?\

La méthode `persist()` permet d'ajouter un objet à la base de données.

2. À quoi correspond la méthode `commit()` ?\

Elle correspond à la fin de l'accès à la base de données

3. Comment ont été traduits les liens objet entre le membre `categorie` et produits entre ces classes dans les tables de la base ?\

Un paramètre "aCategorie" a été passé au constructeur de `Produit`.

4. Quelles requêtes SQL ont été créées par Hibernate pour réaliser la persistance ?\

Hibernate a créé une requête `INSERT INTO Categorie (libelle) VALUES (?)`
et une requête `INSERT INTO Produit (idCategorie, nom, prix, id) VALUES (?, ?, ?, ?)`

5. Pourquoi comportent t-elles des points d'interrogation ?\

Les points d'interrogation sont remplacés par les paramètres de l'objet instancié, ici `Produit` & `Categorie`. Ils permettent aussi la réutilisation de de ses requêtes avec les même paramètres.

Chargement d'un objet

Programme de chargement d'un produit & d'une catégorie

A partir de ses 2 programmes et de leur exécution :

1. Précisez ce que charge exactement Hibernate lors du chargement d'un Objet Au chargement d'un objet Hibernate charge les objets nécessaires aux tests mais aussi les objets qui y sont associés. Si l'on charge un produit, sa catégorie sera aussi chargée.

2. Précisez comment sont chargés les instances liées à un objet chargé pour les liens onToMany et manyToOne

[|h code java](#)

```
Hibernate:
  select
    produits0_.idCategorie as idCatego4_1_,
    produits0_.id as id1_,
    produits0_.id as id1_0_,
    produits0_.idCategorie as idCatego4_1_0_,
    produits0_.nom as nom1_0_,
    produits0_.prix as prix1_0_
  from
    Produit produits0_
  where
    produits0_.idCategorie=?
```

3. En quoi consiste le chargement paresseux d'Hibernate et la qualification lazy (rechercher dans l'aide)

From:

<http://slamwiki2.kobject.net/> - Broken SlamWiki 2.0

Permanent link:

<http://slamwiki2.kobject.net/slam4/orm/etudiants/guillaume?rev=1354637903>

Last update: 2019/08/31 14:39

