# TP-TD6 - Comparaison de frameworks de mapping relationnel/objet

## Création des classes métier

## Produits et catégories

1. La table assurant la persistance d'un objet est déclaré en héritant de la classe Kobject :

```
public class KProduit extends KObject
```

2. Le mapping entre un membre de la classe et un champ de la table relationnelle est déclaré dans le constructeur en tant que membre :

```
hasMany(KLigne.class);
belongsTo(KCategorie.class);
```

3. La clé primaire de la table est déclaré avec un keyFields.

```
keyFields="id";
```

4.

Propriété d'une classe	Champs d'une table	Туре
int	int	entier
string	varchar	chaine

5. Le lien bidirectionnel entre deux classes se paramètre par :

## Programme de test

- 1. La méthode kstart() correspond au démarrage de l'application java qui permettra l'ajout dans la base de donnée.
- 2. Les liens objet entre le membre **catégorie** et **produits** entre ces classes dans les tables de la base ont été traduit par le fait que l'id de la table **catégorie** s'est mis en clé étrangère de la table **produits**.
- 3. Les requêtes SQL qui ont été créées par KObject pour réaliser la persistance sont :
  - INSERT INTO Categorie(libelle) VALUES('Presse')
  - INSERT INTO produit(idCategorie,prix,nom) VALUES('1','3.0','Programmez!')
- 4. Si l'insertion de la catégorie échoue alors l'ajout d'un produit ne pourra s'effectuer car celui-ci doit appartenir à la catégorie que l'on cherche à ajouter.

# Chargement d'un objet

# Last update: 2019/08/31 14:21

#### Programme de chargement d'une catégorie

- 1. Ce que charge exactement KObject lors du chargement d'un Objet sont l'Object lui même ainsi que tous les objets en relation. Par exemple : En chargent l'objet catégorie, tout les produits lié seront eux même chargés.
- 2. Les instances liées à un objet chargé pour les liens belongsTo et hasMany sont chargées grâce à des requêtes SQL. Dans le cas d'un hasMany, KObject sélectionne l'objet "catégorie" dans la base de donnée puis tout les objets produit lié à cette catégorie. Pour la situation d'un belongsTo, kObject sélectionne l'objet dans la base de donnée par exemple un produit, puis il sélectionnera la catégorie correspondant.
- 3. Le chargement paresseux de KObject consiste à

# Chargement de listes d'objets

### **Projection**

1. Résultat obtenu :

#### **Sélection**

- 1. "NB" requêtes SQL sont exécutées par KObject.
- 2. On peut l'interpréter par

Remplacer le lien belongsTo sur la classe Produit par : belongsTo(KCategorie.class).setLazy(true);

- 1. "NB" requêtes SQL sont maintenant exécutées par KObject.
- 2. On peut l'interpréter par

Exécutez à nouveau le programme.

1. On peut interpréter les requêtes SQL exécutées par KObject par

#### Sélection avec distinct et projection

1. Ce programme

## Gestion des commandes

1. L'appel des méthodes permettant de mettre en œuvre la contrainte d'intégrité multiple est justifié par

#### Création de commandes

- 1. Code commenté de CreateCommandeGood.java
- 2. L'exécution a effectué les ajouts dans la base de données :

## **Test Web**

From:

http://slamwiki2.kobject.net/ - SlamWiki 2.1

Permanent link:

http://slamwiki2.kobject.net/slam4/orm/etudiants/clement

Last update: 2019/08/31 14:21

