# Création des classes métier

## Produits et catégories

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

```
S'il hérite de KObject
public class KCategorie extends KObject
```

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

```
hasMany(KLigne.class); un produit correspond à une ligne, il y a un produit par
ligne
belongsTo(KCategorie.class); les produits correspondent à une catégorie, il y a des
produits dans une catégorie
```

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

```
keyFields="id"; , la clé primaire est déclaré en keyFields
```

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

| | **Java** | **SQL** |
|---|---|---|
| **Entier** | int | int |
| **Chaîne de caractère** | string | varchar |
| **Flottant** | float | float |

5.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 le constructeur)







### Programme de test

1.À quoi correspond la méthode kstart() ?

code Java

```
public static void main(String[] args){
    try{
```

```
        Ko.kstart();

        KCategorie aCategorie=new KCategorie("Presse");
        aCategorie.add(Ko.kdatabase());

        KProduit aProduit= new KProduit("Programmez!", 10, aCategorie);
        aProduit.add(Ko.kdatabase());

        Ko.kstop();
}
```

```
La méthode kstart() correspond au démarrage de l'application java qui permet
d'ajouter un produit dans la base de données et d'ajouter une catégorie
```

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

```
Cela a étais traduit par le fait que l'id de la table “catégorie” s'est mis en clé
étrangère dans la table “produits”.
```

3.Quelles requêtes SQL ont été créées par KObject pour réaliser la persistance ?

```
Insertion de la valeur dans la catégorie
INSERT INTO Categorie(libelle) VALUES('Presse')
```

```
Insertion de la valeur dans la table produit
INSERT INTO produit(idCategorie,prix,nom) VALUES('-1','3.0','Programmez!')
```

4.Que se passe t-il si l'insertion de la catégorie échoue ?

```
Si l'insertion de la catégorie échoue et que l'on devait inserer un produit alors
le produit ne sera pas inserer
```

# Chargement d'un objet

1.Précisez ce que charge exactement KObject lors du chargement d'un Objet

```
Lors du chargement d'un Objet, KObject va charger tous les objets en relation.
Par exemple en chargeant une catégorie, tous les produits de cette catégorie seront
chargés.
SQL : KDataBase.sendQuery → SELECT * FROM categorie WHERE categorie.id='13'
KOBJECT : KCategorie.loadFromDb → {id=13}
KOBJECT : KConstraintHasMany.load → class metier.KProduit
SQL : KDataBase.sendQuery → SELECT * FROM produit WHERE idCategorie='13'
```

2.Précisez comment sont chargées les instances liées à un objet chargé pour les liens belongsTo et hasMany

```
Les instances liées  à un object pour le lien __belongsTo__ est chargées dans un
premier temps en récupérant par exemple le produit à l'aide d'un select puis il va
récupérer la catégorie qui correspond à ce produit.
```

```
Les instances liées  à un object pour le lien __hasMany__ est chargées dans un premier temps la catégorie va être ainsi chargée puis ensuite les produits de cette catégories seront chargés
```

3.En quoi consiste le chargement paresseux de KObject ?

```
Le chargement paresseux de KObject consiste a charge par défaut tous les objets en rapport avec un autre.
Par exemple, quand il affiche tous les produits de cette catégorie, il les charges tous.
```

## Chargement de listes d'objets

## Projection

1.Interprétez et expliquez le résultat obtenu

## Sélection

1.Combien de requêtes SQL sont exécutées par KObject ?
2.Comment l'interprétez vous ?
3.Combien de requêtes SQL sont maintenant exécutées par KObject ?
4.Comment l'interprétez vous ?
5.Interprétez les requêtes SQL exécutées par KObject

## Sélection avec distinct et projection

1.Expliquer ce que fait le programme

# Gestion des commandes

1.Justifiez l'appel des méthodes permettant de mettre en oeuvre la contrainte d'intégrité multiple

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

Permanent link:
**http://slamwiki2.kobject.net/slam4/orm/etudiants/pierrick?rev=1354638200**

Last update: **2019/08/31 14:39**