Création des classes métier

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

La table assurant la persistance d'un objet est déclarée s'il hérite de KObject –> public class KCategorie extends KObject.

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

"hasMany(KLigne.class)" un produit correspondant 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.

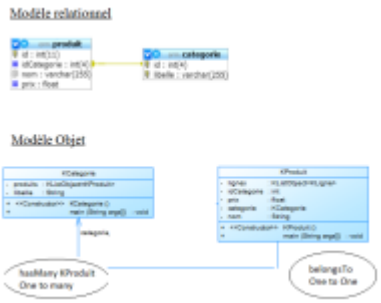
3)Comment est déclarée la clé primaire de la table ?

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.

int	int	entier
string	varchar	chaine

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) {
        Ko.useCache=true;
        KCache.loadAllCache();
        try {
            Ko.kstart();
            KScriptTimer.start();
            KScriptTimer.start("KProduit");
```

2)Comment ont été traduits les liens objet entre le membre categorie et produits entre ces classes dans les tables de la base ? Les liens objets entre "catégorie" et "produits" entre ces classes dans les tables de la base de données ont été traduit par le simple fait que l'id de la table "catégorie" s'est mis en clé étrangère de la table "produits". 3)Quelles requêtes SQL ont été créées par KObject pour réaliser la persistance ? Afin de réaliser la persistance, KObject a utilisé les requêtes SQL d'insertion suivantes : *(visibles dans la console lors du test)* INSERT INTO categorie(libelle) VALUES('Presse') *INSERT INTO produit(idCategorie,prix,nom)*

VALUES('19','3.0','Programmez!') 4)Que se passe t-il si l'insertion de la catégorie échoue ? En cas d'échec de l'insertion de la catégorie, il est impossible par la suite d'insérer des produits car un produits appartient à une catégorie et ces deux objets sont liés par l'id de la catégorie. **Chargement d'un objet**

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

code Java

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

“hasMany(KLigne.class)” un produit correspondant 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.

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

code Java

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

“hasMany(KLigne.class)” un produit correspondant 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)En quoi consiste le chargement paresseux de KObject ?

code Java

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

keyFields=“id”; , la clé primaire est déclaré en keyFields
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 ?

***Remplacer le lien belongsTo sur la classe Produit par :**

belongsTo(KCategorie.class).setLazy(true);

1)Combien de requêtes SQL sont maintenant exécutées par KObject ?

2)Comment l'interprétez vous ?

***A partir du programme :**

1)Interprétez les requêtes SQL exécutées par KObject

Sélection avec distinct et projection

1)Expliquer ce que fait le programme

Création de commandes

1)Analysez puis commentez chaque ligne (dans le code) de ce programme 2)Vérifier que l'exécution a effectué les ajouts dans la base de données

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

Permanent link:
http://slamwiki2.kobject.net/slam4/orm/etudiants/david?rev=1354637970

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