

Différentes politiques de chargement

Vous allez expérimenter dans ce TP différentes politiques de chargement des entités. Nous nous baserons sur l'exemple du cours. Il est sous forme d'archive sur moodle. Deux programmes sont disponibles :

- `cinema.ApplicationCreation` : il est à exécuter une fois. Il crée les tables et les séquences et il insère un certain nombre de données dans les tables.
- `cinema.ApplicationSelections` : c'est le programme que vous aurez à modifier pour réaliser les différentes expériences.

Le fait de travailler avec JAVA SE et non en JAVA EE impose quelques restrictions quant aux stratégies de chargement des objets : la politique de chargement pour les relations **ManyToOne** et **OneToOne** est forcément **eager** même si on demande explicitement **lazy**. Pour s'en affranchir, nous ferons les changements suivants :

- nous ajoutons cette propriété dans le fichier `persistence.xml` :

```
<property name="eclipselink.weaving" value="true"/>
```

- pour l'exécution nous utiliserons le `javaengine` d'ECLIPSELINK avec la commande suivante :

```
java -javaagent:${JLIB}/eclipselink.jar -cp .:$ECL_PATH cinema.ApplicationSelection
```

Ceci permettra à ECLIPSELINK de modifier dynamiquement le bytecode des classes afin d'utiliser par exemple la politique **lazy** pour les relations **ManyToOne**.

Pour commencer, exécutez avec `dbeaver` le script `nettoyage.sql` que vous trouverez dans le répertoire `cinema`. Puis exécutez le programme `cinema.ApplicationCreation` qui créera les tables et les peuplera de données. Vous pourrez ensuite positionner dans le fichier `persistence.xml` la propriété `eclipselink.ddl-generation` à `none`. Dans les questions suivantes, vous aurez au moins à modifier la méthode `main` de la classe `cinema.ApplicationSelection`.

Question 1

En utilisant une requête JPQL, récupérez la liste des films puis affichez leur titre. Examinez les requêtes produites par l'exécution de votre programme. Vu que nous n'utilisons que les titres des films, toutes les requêtes sont-elles utiles ?

Question 2

Modifiez le code en conséquence afin de n'avoir qu'une seule requête générée par notre programme `cinema.ApplicationSelection` qui ne sera pas, lui, modifié.

Question 3

Modifiez le programme `cinema.ApplicationSelection` en affichant pour chaque film en plus du titre son réalisateur. Examinez de nouveau les requêtes générées par ECLIPSELINK.

Question 4

En modifiant le code, faites en sorte que nous n'ayons plus qu'une requête à l'exécution de notre programme.

Question 5

Modifiez le programme `cinema.ApplicationSelection` en affichant pour chaque film en plus du titre et son réalisateur tous ses acteurs. Examinez de nouveau les requêtes générées par ECLIPSELINK.

Question 6

En modifiant le code, faites en sorte que nous n'ayons plus qu'une requête à l'exécution de notre programme.