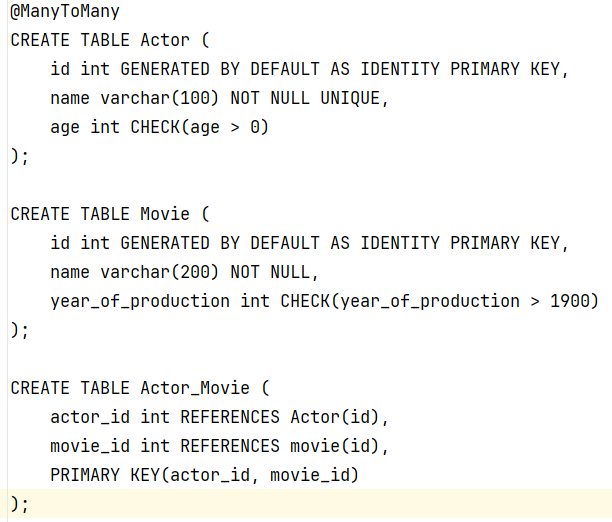
**Hibernate Отношение Many to Many**

У нас есть связь многие ко многим. У нас есть три таблицы, таблица актеров, таблица фильмов и связывающая таблица, которая связывает актеров с фильмами.

SQL команды для создания таблиц в БД:



Чтобы выстраивать связь между Many to Many есть следующие аннотации: **@ManyToMany** и **@ JoinTable**.

В данном случае у нас есть сущность актер и сущность фильм. У Актера есть список фильмов, а у фильма есть список актеров. Эти поля мы помечаем аннотацией **@ManyToMany**. Так же есть аннотация **@JoinTable**, которая даёт понять **Hibernate**`у какая у нас связывающая таблица. Аннотацицю **@JoinTable** можно использовать на любой стороне связи. Тоесть в связи **@ManyToMany** нет как таковой owining side как в предыдущих случаях потому-что ни одна из сущностей не владеет внешним ключом.

В аннотации **@JoinTable** указывается:

**name** – название связывающей таблицы.

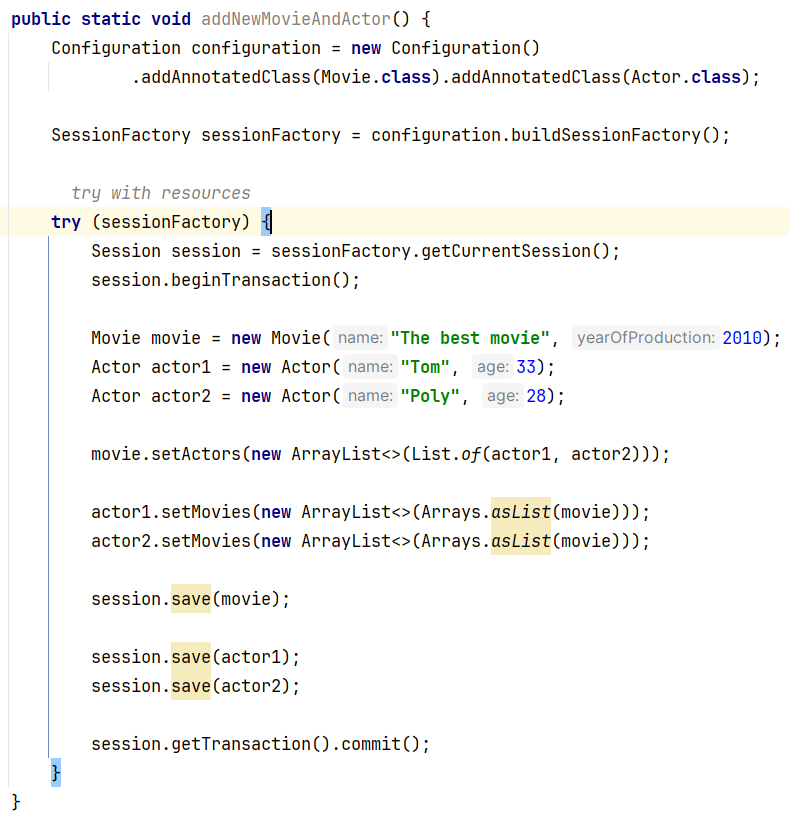
**joinColumns** – указываем ту колонку которая принадлежит текущей сущности.

**inverseJoinColumns** – указываем ту колонку которая принадлежит той сущности которая находится на другой стороне нашей связи (другую таблицу).

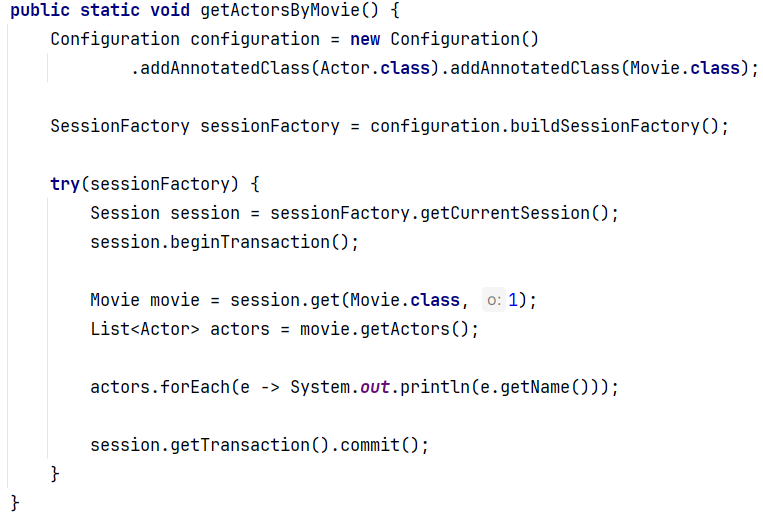
В связи **@ManyToMany** мы могли бы поместить аннотацию **@JoinTable** на другую сторону так как здесь нету понятия главной сущности.

**Примеры:**

**1.** Создание актеров и фильмами



**2.** Вывод актеров из фильма.



**3.** Добавление нового фильма для уже созданного актера.



**4.** Удаление фильма у существующего актера.

