*Please note that we haven't explained full-fledged annotation based Hibernate project here. Rather we have focused on explaining how unidirectional many to many association is mapped in Hibernate.*

*To know more about annotation based basic Hibernate project please thoroughly go through project ' Hibernate5StandaloneWithFullJavaConfig' if you are not familiar with it.*

**What this project dose ?**

This project explains how unidirectional many to many association is mapped in Hibernate by using annotations.

**A short introduction**

In many-to-many association, source entity has a field that stores collection of target entities. The @*ManyToMany* JPA annotation is used to link the source entity with the target entity.

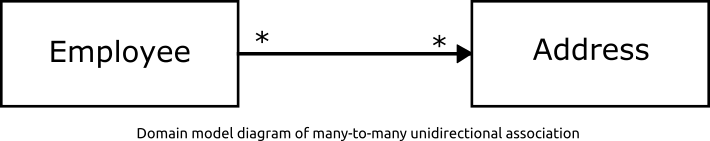
A many-to-many association always uses an intermediate join table to store the association that joins two entities. The join table is defined using the @JoinTable JPA annotation.

The many-to-many association can be either unidirectional or bidirectional.

 In unidirectional association, only source entity has a relationship field that refers to the target entities. We can navigate this type of association from one side.

**Unidirectional many-to-many association example**

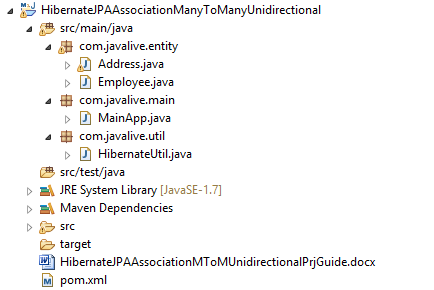
Consider the following domain model and relational model diagrams of many-to-many unidirectional association.





According to the models diagrams, an employee can have any number of addresses and an address can belong to any number of employees.

**Project Structure**

****

**Jar dependencies**

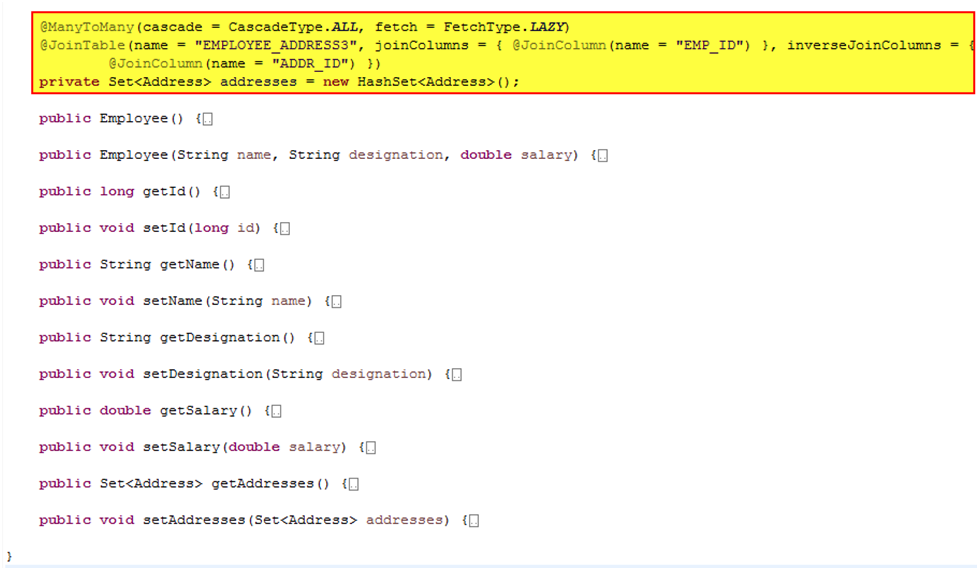
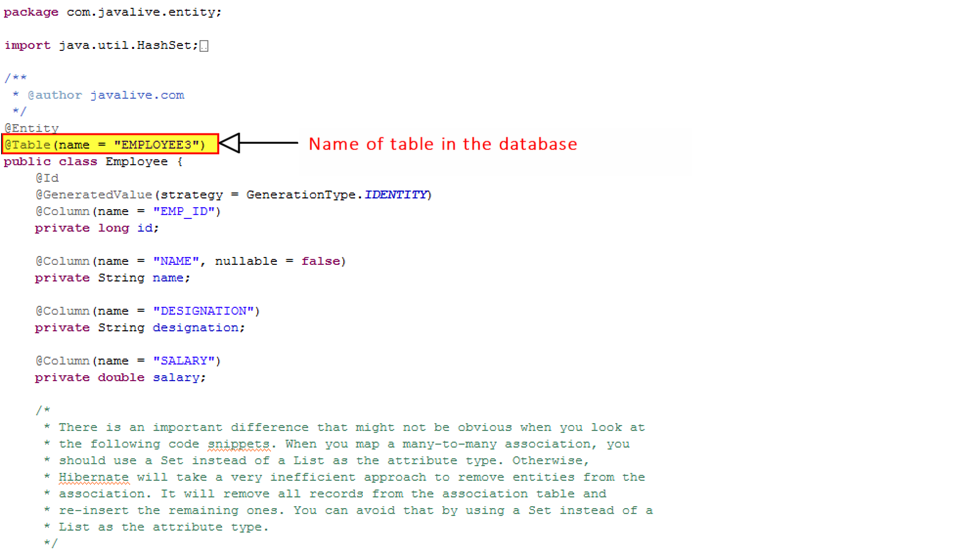
Add the following jar dependencies for Hibernate and MySQL driver in pom.xml file.



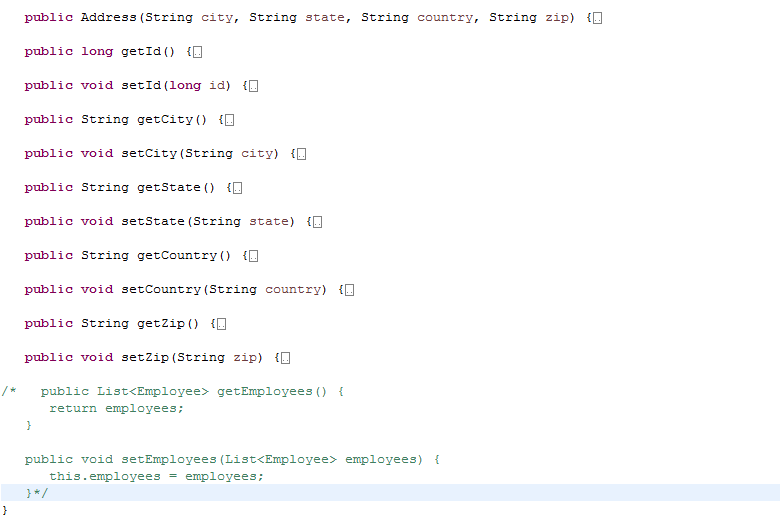
**Entity class**

Create two @Entity classes - Employee and Address, to map with EMPLOYEE and ADDRESS tables respectively.

**Employee.java**

****

**Address.java**



The @JoinTable annotation is used to create the EMPLOYEE\_ADDRESS join table. This table defines an EMP\_ID foreign key to the source entity’s table primary key and an ADDR\_ID foreign key to the target entity’s table primary key.

**Hibernate utility class**

Create a helper class HibernateUtil to bootstrap hibernate.

Map the Employee and Address entities using the #MetadataSources.addAnnotatedClass() method.

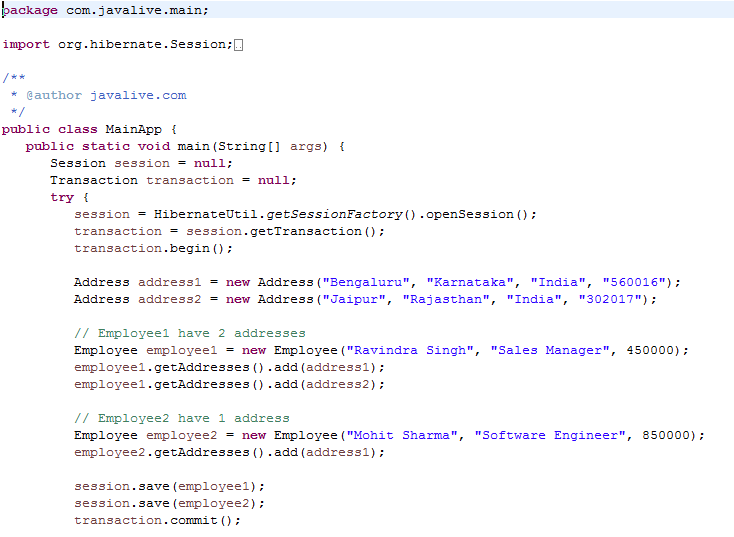
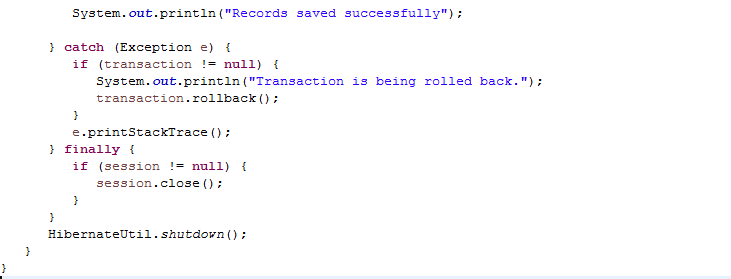
**HibernateUtil.java**



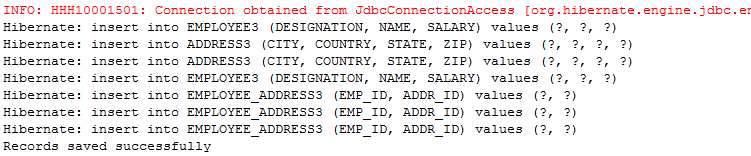
**Main class**

Create the MainApp class to run the application.

**MainApp.java**

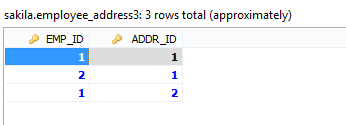
**Output**



On executing the MainApp class, you will see the following records in EMPLOYEE3, ADDRESS3 and EMPLOYEE\_ADDRESS3 tables.



****

****