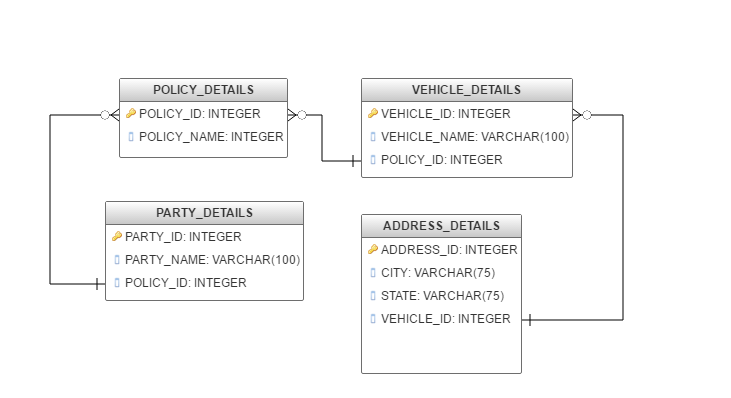
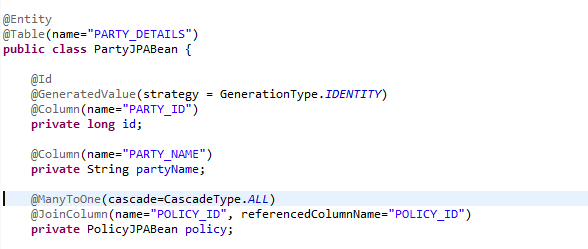
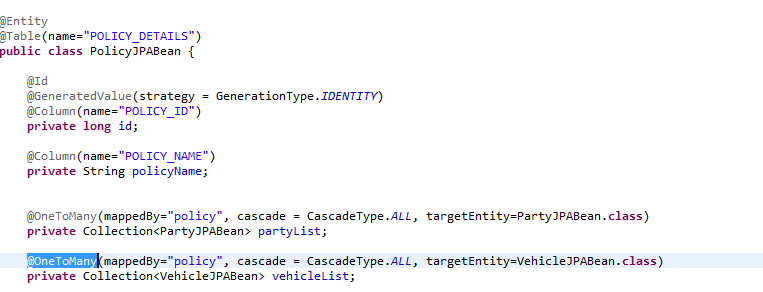
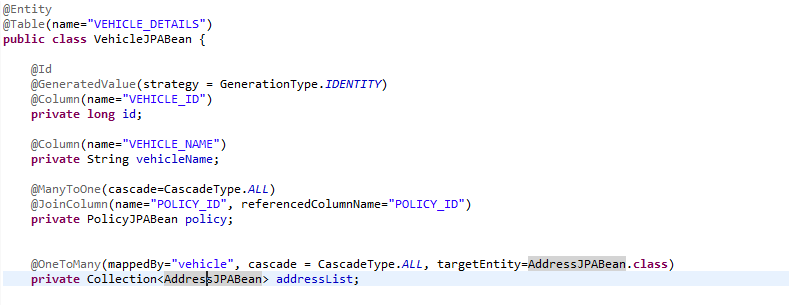
**Database Structure**

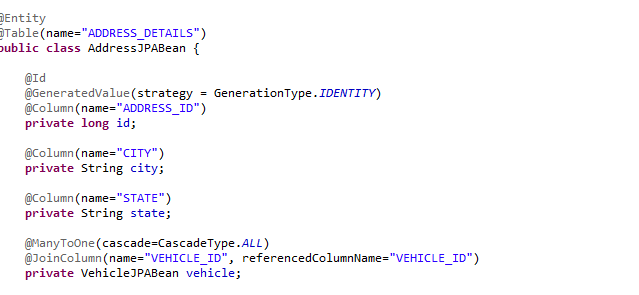


**Entity Structure**



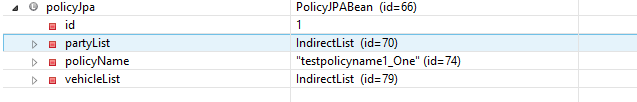






**Difference between FetchTypes**

* Modified PolicyJPABean and VehicleJPABean
* Added FetchType.LAZY
* Only One query triggered : SELECT POLICY\_ID, POLICY\_NAME FROM POLICY\_DETAILS WHERE (POLICY\_NAME = ?)
* Only Policy details loaded in memory



* Modified to FetchType.EAGER
* All queries triggered while fetching policy details
* Queries triggered:

SELECT POLICY\_ID, POLICY\_NAME FROM POLICY\_DETAILS WHERE (POLICY\_NAME = ?)

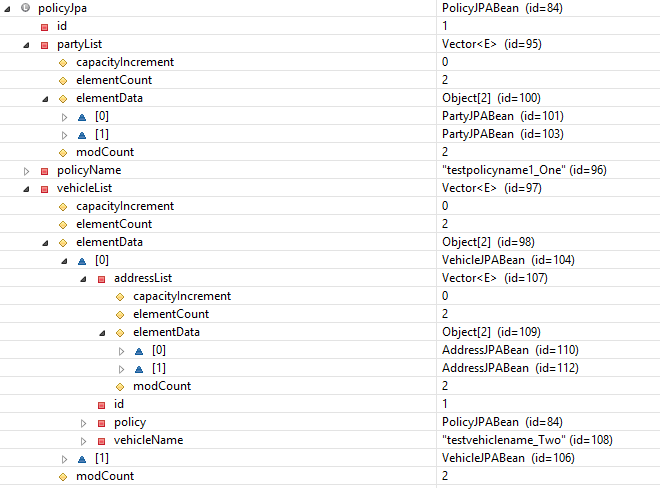
SELECT PARTY\_ID, PARTY\_NAME, POLICY\_ID FROM PARTY\_DETAILS WHERE (POLICY\_ID = ?)

SELECT VEHICLE\_ID, VEHICLE\_NAME, POLICY\_ID FROM VEHICLE\_DETAILS WHERE (POLICY\_ID = ?

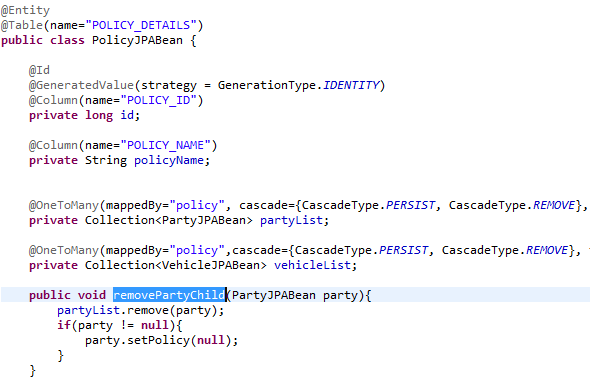
SELECT ADDRESS\_ID, CITY, STATE, VEHICLE\_ID FROM ADDRESS\_DETAILS WHERE (VEHICLE\_ID =?

SELECT ADDRESS\_ID, CITY, STATE, VEHICLE\_ID FROM ADDRESS\_DETAILS WHERE (VEHICLE\_ID =?

* All details loaded in memory



**Remove** **Operations**

* For Removing the relationship between the entities.
* 
* Will remove the relation ship between the parent and child entities
* Below queries are triggered

SELECT POLICY\_ID, POLICY\_NAME FROM POLICY\_DETAILS WHERE (POLICY\_NAME = ?)

SELECT PARTY\_ID, PARTY\_NAME, POLICY\_ID FROM PARTY\_DETAILS WHERE (POLICY\_ID = ?)

UPDATE PARTY\_DETAILS SET POLICY\_ID = ? WHERE (PARTY\_ID = ?)

* With orphanRemoval=true, the child record will be deleted

