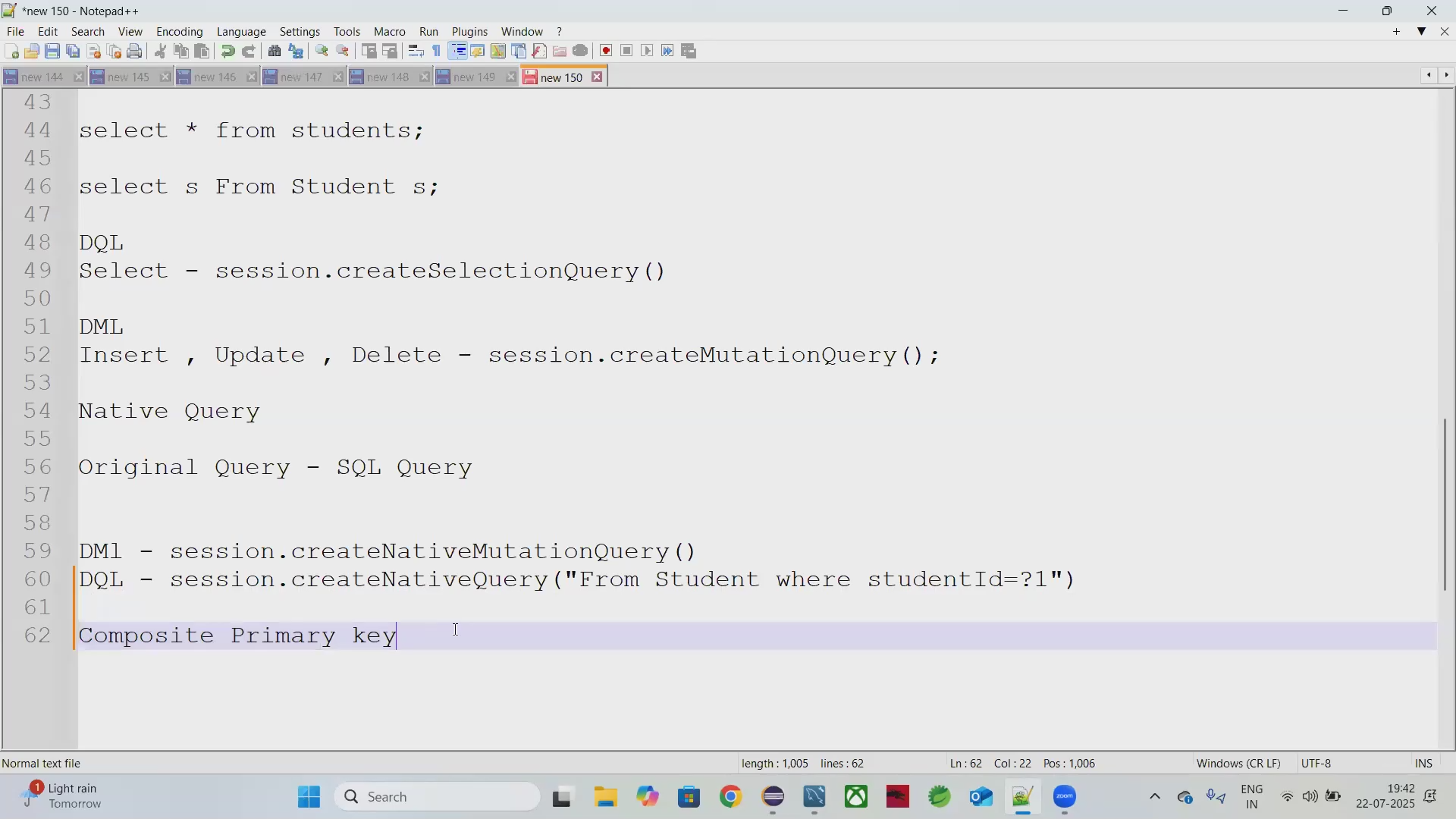
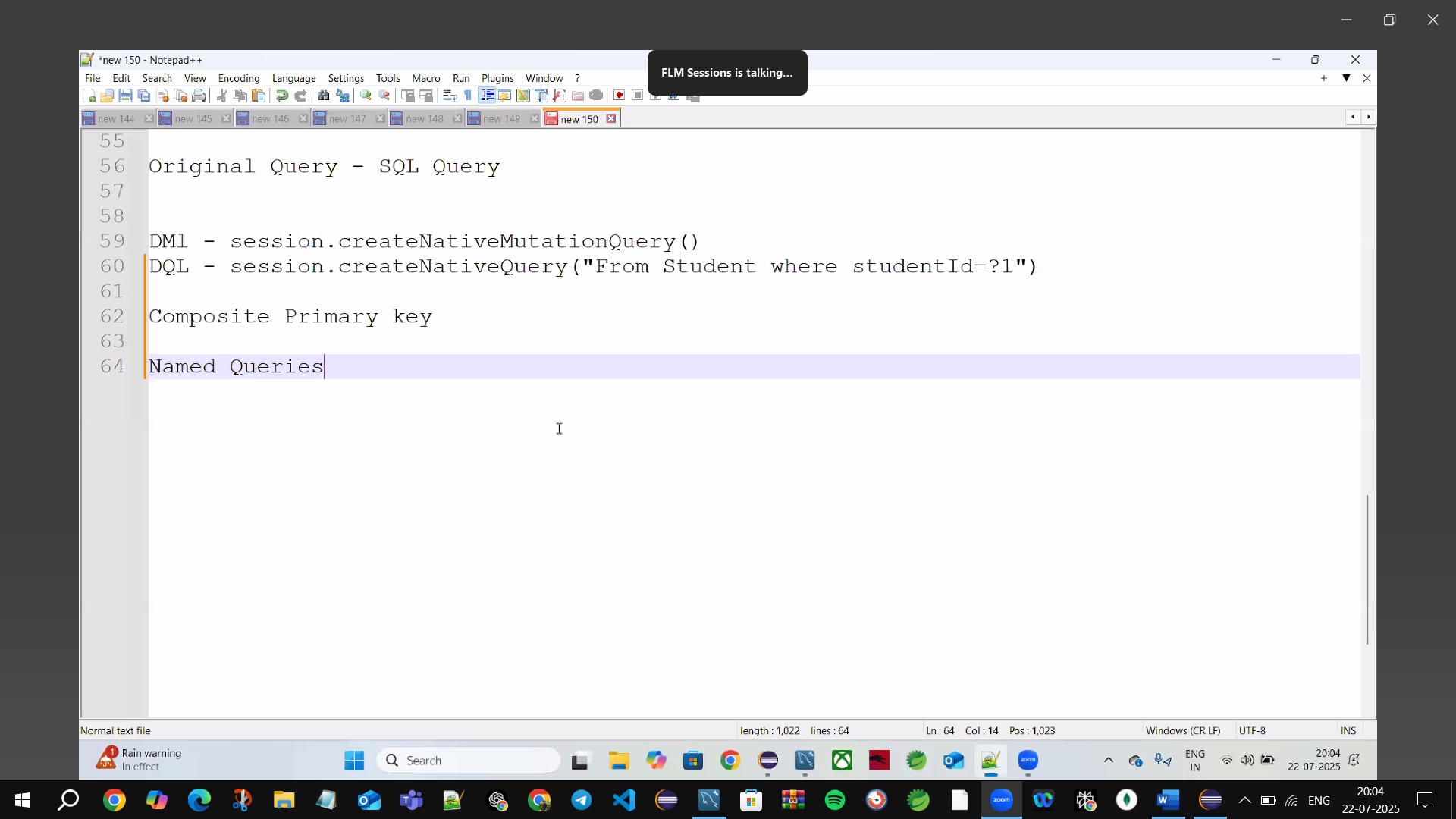
**4.Embeddable\_NamedQueries\_One2One**



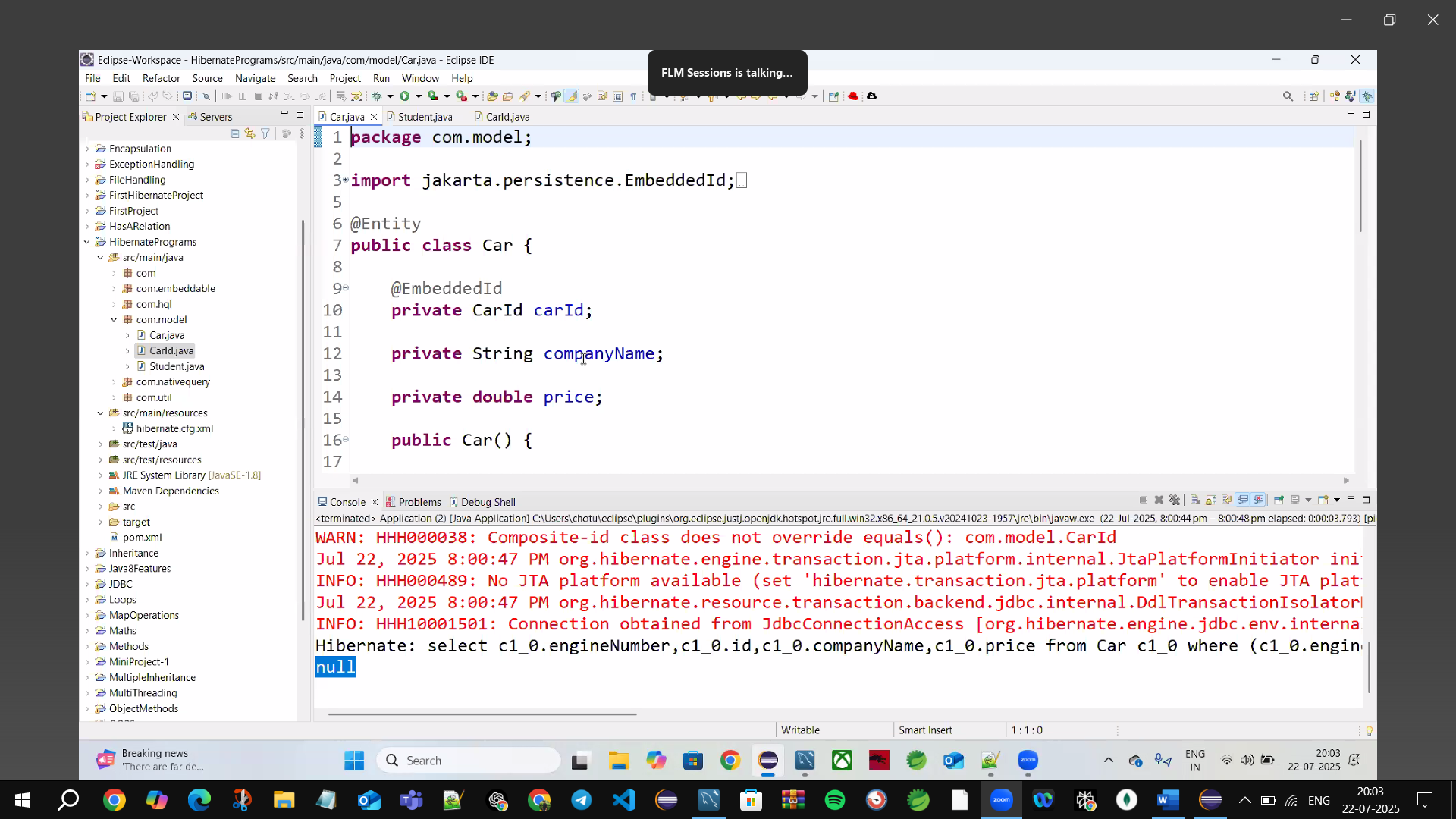


* **Composite Primary key🡪**when 2 or more columns act as a primary key (P.K)

**Composite Primary key🡪**put @Embeddable (CarId.java)

**🡪@Embeddable means:** this class field can be embedded to form a Primary Key

**Actual @entity**🡪@EmbeddedId (Car.java)

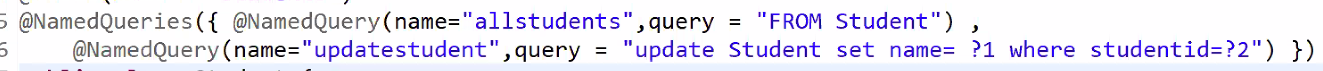


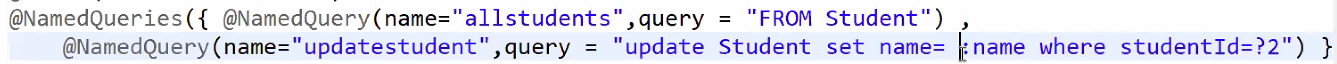
Named Queries: - repeated Queries

🡪execution will be faster

🡪pre-compiled Query



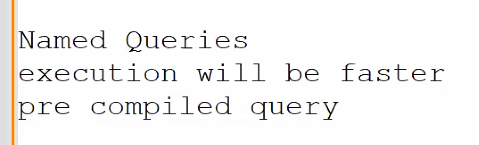




Name:name(column name)

studentId=?2(column name)



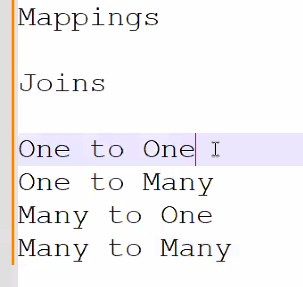


**Singleton Design Pattern:** 1 class has only 1 object



DML🡪No need Class names

🡪DQL🡪need Class names



Mappings same as Spring Boot also

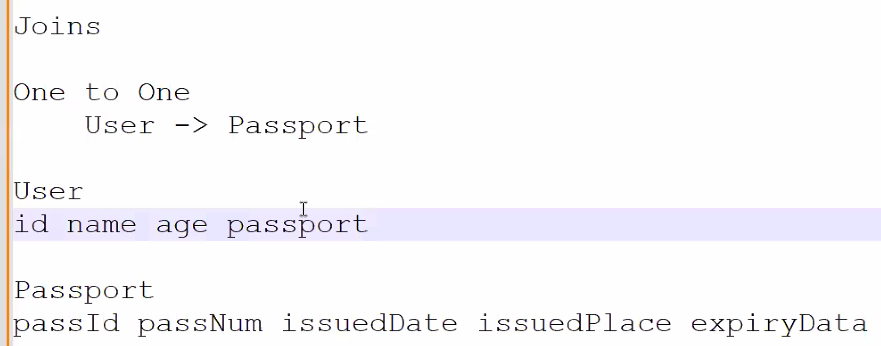
**Mappings 🡪4 types**

**1.one to one🡪** one user🡪 one passport

**2.One to Many**

**3.Many to one**

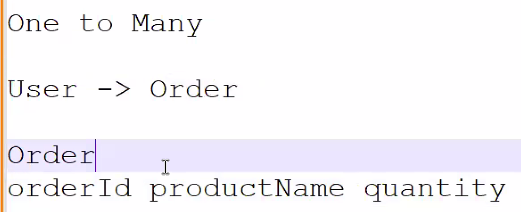
**4.Many to Many**

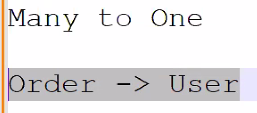


🡪Has a relationship

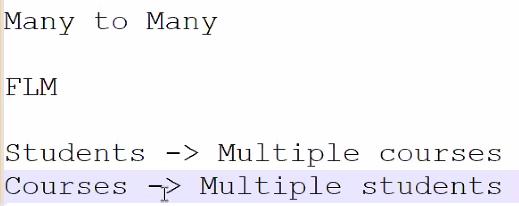
1user 🡪have only 1 passport

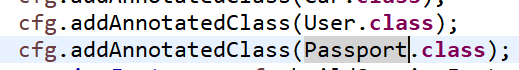
🡪One table Relation with only one table 1🡪1



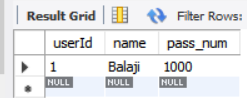
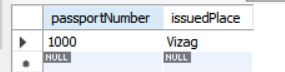


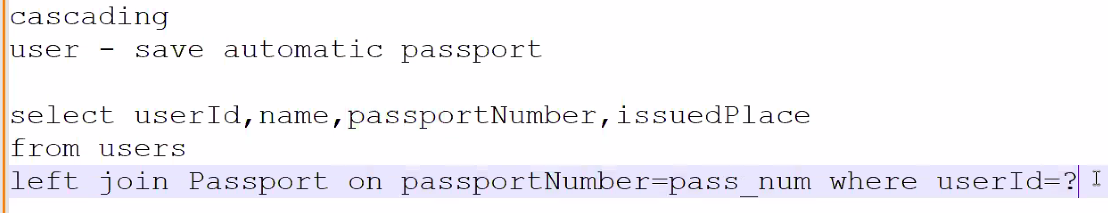
Many orders form 🡪1 user

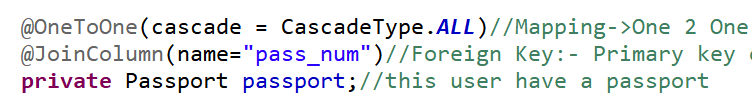




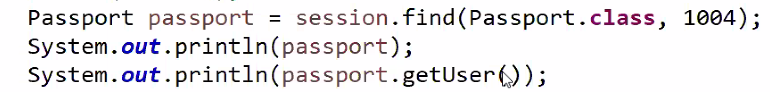
Add class in🡪 HibernateUtil

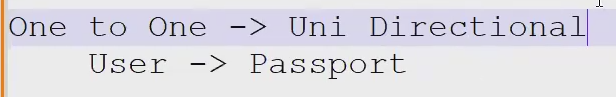


🡪Manually Persist the Passport class or use Cascading

🡪We can join userid in Passport class(makes Composite primary key)

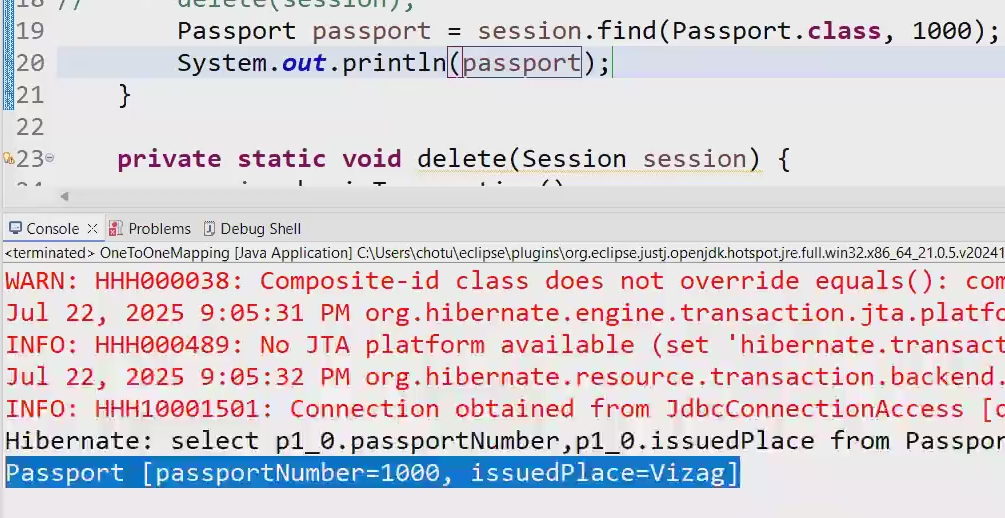
 

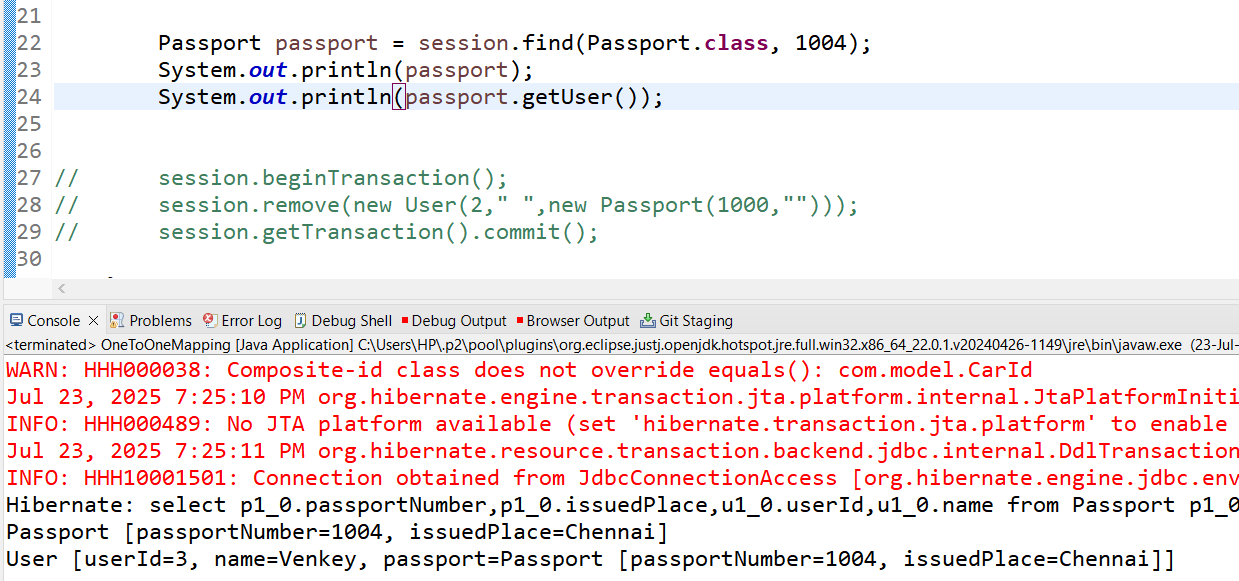
🡪Stack overflow (bcz user🡪passport🡪passport 🡪user both sides printing)



From user able to bring passport details

Not getting user details from passport





Getting passport & user🡪Bi Directional





