**CS544**

**Enterprise Architecture**

## Exam 1 April 2020

Name\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Student ID \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**NOTE: This material is private and confidential. It is the property of MUM and is not to be disseminated.**

1. ANSWER GOES HERE
2. TRUE

**This is a major advantage of an ORM.** A RDB entity-entity relationship uses Foreign keys. **The**

**ORM “automatically” maps these relationships, reducing boiler plate code.**

1. FALSE

**JPA is an industry standard but does not replace Hibernate. In fact, Hibernate implements JPA**

**As JPA is “only” an API.**

1. ANSWER GOES HERE

**edu.mum.dao.** UserDao

public User findBySoldItemInitialPrice(String email,String itemName, BigDecimal initialPrice);

**edu.mum.dao.impl.** UserDaoImpl

**public** User findBySoldItemInitialPrice(String email, String boughtItem, BigDecimal initialPrice) {

Query query=entityManager.createQuery("select u from User u, Item i where i.seller.email=:email"

+ "and i.name = :boughtItem and i.initialPrice > :initialPrice”);

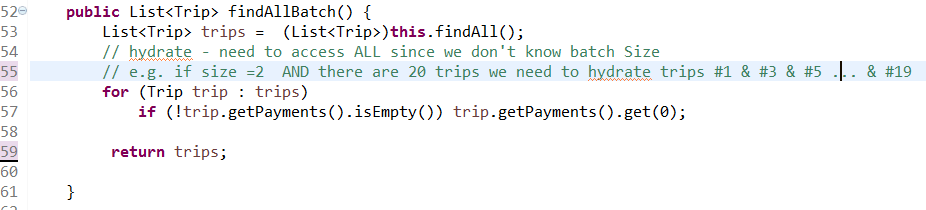
**return** (User) query.setParameter("boughtItem", boughtItem).

setParameter("email", email). setParameter("initialPrice", initialPrice).getSingleResult();

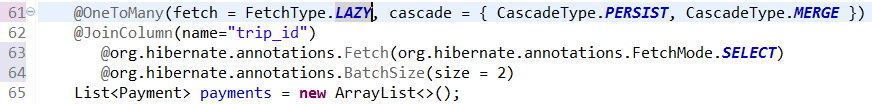
}

1. ANSWER GOES HERE

**In TripServiceImpl.Java**

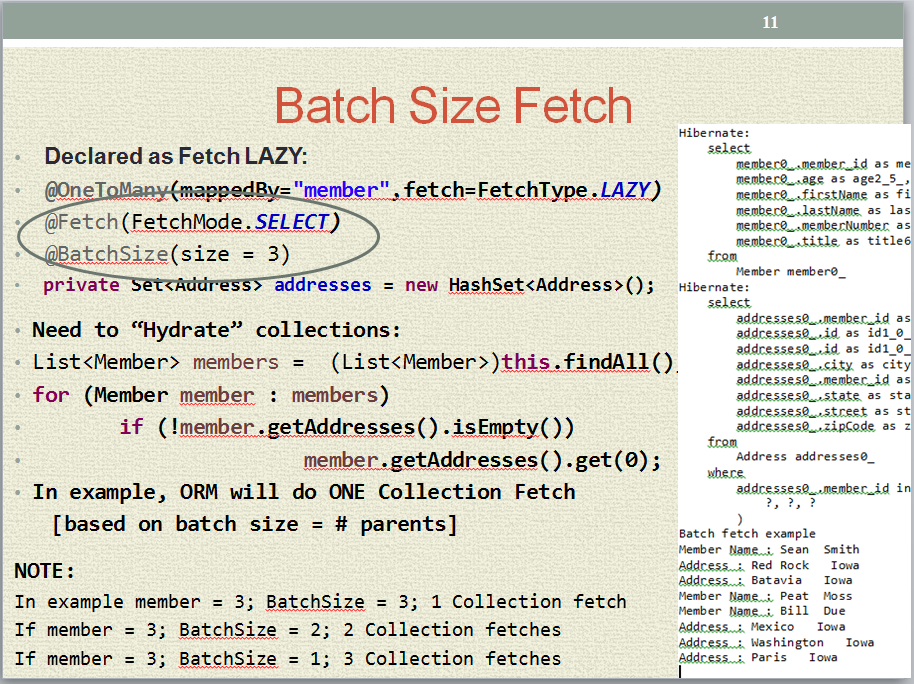
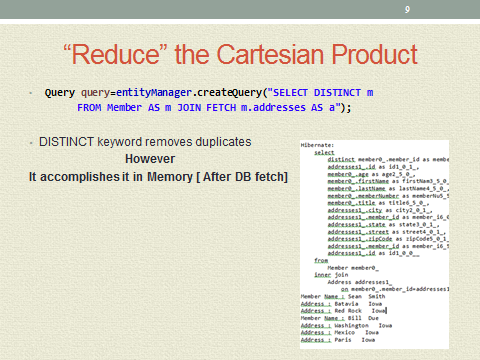
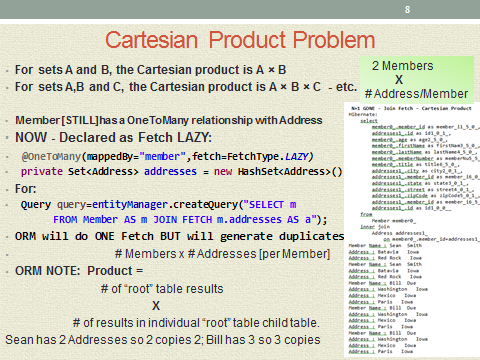
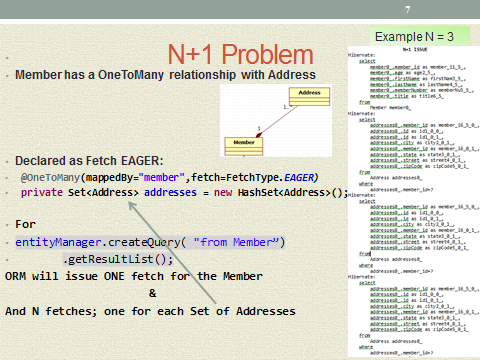
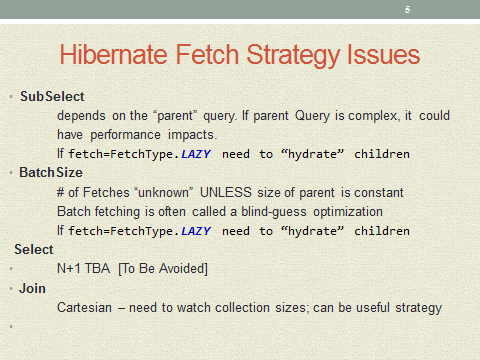


**In Trip.Java**



EXPLAIN DETAILS that it Solves N+1 AND Cartesian Product BUT WEAKNESS IS # of collection fetches is “unknown”…

IMPROVES upon fetchMode.SELECT…



1. ANSWER GOES HERE

