ONE TO ONE MAPPING

Primary class🡪

@Entity

@Data

@Table(name = "employee\_info")

public class Employee {

@Id

@Column

private int eid;

@Column

private String ename;

// @Exclude

// @OneToOne(cascade = CascadeType.ALL, mappedBy = "employee")

// private Laptop laptop;

}

Secondary class:

@Table(name = "laptop\_info")

@Entity

@Data

**public** **class** Laptop {

@Id

@Column

**private** **int** sid;

@Column

**private** String sname;

@Exclude

@OneToOne(cascade = CascadeType.***ALL***)

@JoinColumn(name = "eid")

**private** Employee employee;

}

ONE WAY BINDING:

public class OneWayTest {

public static void main(String[] args) {

EntityManagerFactory entityManagerFactory = null;

EntityManager entityManager = null;

EntityTransaction entityTransaction = null;

Employee employee = new Employee();

employee.setEid(10);

employee.setEname("Phillip");

Laptop laptop = new Laptop();

laptop.setSid(100);

laptop.setSname("Asus");

laptop.setEmployee(employee);

try {

entityManagerFactory = Persistence.createEntityManagerFactory("mapping");

entityManager = entityManagerFactory.createEntityManager();

entityTransaction = entityManager.getTransaction();

entityTransaction.begin();

entityManager.persist(laptop);

entityTransaction.commit();

// Laptop laptop2 = entityManager.find(Laptop.class, 30);

// System.out.println(laptop2.getEmployee());

} catch (Exception e) {

e.printStackTrace();

}

}

}

TWO WAY BINDING:

FOR TWOWAY FIRST UNCOMMENT THE COMMENTED LINES IN THE EMPLOYEE.JAVA

public class TwoWay {

public static void main(String[] args) {

EntityManagerFactory entityManagerFactory = null;

EntityManager entityManager = null;

EntityTransaction entityTransaction = null;

Employee employee = new Employee();

employee.setEid(30);

employee.setEname("Sammy");

Laptop laptop = new Laptop();

laptop.setSid(300);

laptop.setSname("Lenovo");

employee.setLaptop(laptop);

laptop.setEmployee(employee);

try {

entityManagerFactory = Persistence.createEntityManagerFactory("mapping");

entityManager = entityManagerFactory.createEntityManager();

entityTransaction = entityManager.getTransaction();

entityTransaction.begin();

entityManager.persist(laptop);

// entityManager.persist(employee);

Employee employee2 = entityManager.find(Employee.class,20 );

// System.out.println(laptop.getEmployee().getEname());

System.out.println(employee2.getLaptop().getSname());

entityTransaction.commit();

} catch (Exception e) {

e.printStackTrace();

}

}

}