**EclipseLink – persistence.xml**

**Hibernate - hibernate.cfg.xml**

<?xml version='1.0' encoding='UTF-8'?>

<!DOCTYPE hibernate-configuration PUBLIC

          "-//Hibernate/Hibernate Configuration DTD 3.0//EN"

          "http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">

<hibernate-configuration>

    <session-factory>

        <property name="hbm2ddl.auto">update</property>

        <property name="dialect">org.hibernate.dialect.Oracle9Dialect</property>

        <property name="connection.url">jdbc:oracle:thin:@localhost:1521:xe</property>

        <property name="connection.username">system</property>

        <property name="connection.password">oracle</property>

        <property name="connection.driver\_class">oracle.jdbc.driver.OracleDriver</property>

    <mapping resource="employee.hbm.xml"/>

    </session-factory>

</hibernate-configuration>

**EclipseLink**

**(**[**https://wiki.eclipse.org/EclipseLink/Examples/JPA/JSF\_Tutorial**](https://wiki.eclipse.org/EclipseLink/Examples/JPA/JSF_Tutorial)**)**

**Source Code: (**[**http://www.oracle.com/technetwork/topics/order-jsf-jpa-example-128127.zip**](http://www.oracle.com/technetwork/topics/order-jsf-jpa-example-128127.zip)**)**

The entity manager is the primary JPA interface you use to perform basic persistence operations on your entities (create, read, update, and delete).

public void requestCancelOrder(long orderId){

//Create an EntityManager from the Factory stored in the JPAResourceBean

EntityManager em = jpaResourceBean.getEMF().createEntityManager();

try{

//changes will be made so begin a transaction

em.getTransaction().begin();

//find the order that will be deleted. This step ensures the order

//will be managed as the specification requires the object be

//managed before remove can be called.

Order order = em.find(Order.class, orderId);

//set the order to be delet4ed

em.remove(order);

//commit the transaction, this will cause the the delete SQL to be

//sent to the database.

em.getTransaction().commit();

}finally{

em.close();

}

}

**Hibernate:**

**package** com.javatpoint.mypackage;

**import** org.hibernate.Session;

**import** org.hibernate.SessionFactory;

**import** org.hibernate.Transaction;

**import** org.hibernate.cfg.Configuration;

**public** **class** StoreData {

**public** **static** **void** main(String[] args) {

    //creating configuration object

    Configuration cfg=**new** Configuration();

    cfg.configure("hibernate.cfg.xml");//populates the data of the configuration file

    //creating seession factory object

    SessionFactory factory=cfg.buildSessionFactory();

    //creating session object

    Session session=factory.openSession();

    //creating transaction object

    Transaction t=session.beginTransaction();

    Employee e1=**new** Employee();

    e1.setId(115);

    e1.setFirstName("sonoo");

    e1.setLastName("jaiswal");

    session.persist(e1);//persisting the object

    t.commit();//transaction is commited

    session.close();

    System.out.println("successfully saved");

}

}

**Annotation References:** [**http://www.eclipse.org/eclipselink/documentation/2.6/jpa/extensions/annotations\_ref.htm**](http://www.eclipse.org/eclipselink/documentation/2.6/jpa/extensions/annotations_ref.htm)

**Complete EclipseLink Reference:** [**http://www.eclipse.org/eclipselink/documentation/2.6/**](http://www.eclipse.org/eclipselink/documentation/2.6/)