create table person(pid number primary key, pname varchar(40), age number)

create table orders(oid number primary key, onum number, oitem varchar(40), pid number references person(pid))

create sequence hibOdrSeq start with 1000 increment by 1

create sequence hibPerSeq start with 10000 increment by 1

insert into person values(hibPerSeq.nextval, 'Admin', 45)

insert into person values(hibPerSeq.nextval, 'Trainer', 35)

insert into person values(hibPerSeq.nextval, 'Lead', 38)

insert into person values(hibPerSeq.nextval, 'Manager', 38)

insert into orders values(hibOdrSeq.nextval, 3256, 'laddu', 10000)

insert into orders values(hibOdrSeq.nextval, 3257, 'Pizza', 10001)

insert into orders values(hibOdrSeq.nextval, 3255, 'Donut', 10002)

insert into orders values(hibOdrSeq.nextval, 3259, 'Pizza', 10003)

insert into orders values(hibOdrSeq.nextval, 3259, 'Donut', 10003)

select \* from person

select \* from orders

<modelVersion>4.0.0</modelVersion> <groupId>com.wipro</groupId> <artifactId>EmpDeptMavenHib</artifactId> <version>0.0.1-SNAPSHOT</version> <dependencies> <dependency> <groupId>org.hibernate</groupId> <artifactId>hibernate-core</artifactId> <version>5.4.9.Final</version> </dependency> <dependency> <groupId>com.oracle.database.jdbc</groupId> <artifactId>ojdbc6</artifactId> <version>11.2.0.4</version> </dependency> </dependencies></project>

QUERY

create table dept( deptno int primary key, dname varchar(20), loc varchar(30))Empcreate table Emp( eid int primary key, name varchar(20), sal int, deptno int references dept(deptno))

Department.java

from Athma M to everyone: 12:27 PM

package beanpkg;import java.util.Set;public class Department { private int dno; private String dname; private String loc; private Set employees; public Department(){ super(); } public Department(int dno, String dname, String loc) { super(); this.dno = dno; this.dname = dname; this.loc = loc; } public int getDno() { return dno; } public void setDno(int dno) { this.dno = dno; } public String getDname() { return dname; } public void setDname(String dname) { this.dname = dname; } public String getLoc() { return loc; } public void setLoc(String loc) { this.loc = loc; } public Set getEmployees() { return employees; } public void setEmployees(Set employees) { this.employees = employees; } @Override public String toString() { return "Department [dno=" + dno + ", dname=" + dname + ", loc=" + loc + "]"; } }

Employee.java

from Athma M to everyone: 12:28 PM

package beanpkg;public class Employee { private int eid; private String ename; private int sal; private Department dept; public Employee(){ super(); } public Employee(int eid, String ename, int sal, Department dept) { super(); this.eid = eid; this.ename = ename; this.sal = sal; this.dept = dept; } public int getEid() { return eid; } public void setEid(int eid) { this.eid = eid; } public String getEname() { return ename; } public void setEname(String ename) { this.ename = ename; } public int getSal() { return sal; } public void setSal(int sal) { this.sal = sal; } public Department getDept() { return dept; } public void setDept(Department dept) { this.dept = dept; } @Override public String toString() { return "Employee [eid=" + eid + ", ename=" + ename + ", sal=" + sal + ", dept=" + dept + "]"; } }

department.hbm.xml

from Athma M to everyone: 12:28 PM

<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN" "http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd"><hibernate-mapping><class name="beanpkg.Department" table="dept"> <id name="dno" column="deptno" type="int"> <generator class="assigned"></generator> </id> <property name="dname" column="dname" type="string"></property> <property name="loc" column="loc" type="string"></property> <set name="employees" table="emp" inverse="false" lazy="true"> <key> <column name="deptno" /> </key> <one-to-many class="beanpkg.Employee"/> </set></class></hibernate-mapping>

employee.hbm.xml

from Athma M to everyone: 12:29 PM

<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE hibernate-mapping PUBLIC "-//Hibernate/Hibernate Mapping DTD 3.0//EN" "http://hibernate.sourceforge.net/hibernate-mapping-3.0.dtd"><hibernate-mapping> <class name="beanpkg.Employee" table="emp"> <id name="eid" type="int"> <column name="eid" /> <generator class="assigned" /> </id> <property name="ename" type="string"> <column name="name" /> </property> <property name="sal" type="int"> <column name="sal" /> </property> <many-to-one name="dept" class="beanpkg.Department" fetch="join"> <column name="deptno" /> </many-to-one> </class></hibernate-mapping>

hibernate.cfg.xml

<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE hibernate-configuration SYSTEM "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd"><hibernate-configuration> <session-factory> <property name="hibernate.dialect">org.hibernate.dialect.Oracle10gDialect</property> <property name="hibernate.connection.driver\_class">oracle.jdbc.OracleDriver</property> <!-- scott is the database name --> <property name="hibernate.connection.url">jdbc:oracle:thin:@localhost:1521:xe</property> <property name="hibernate.connection.username">scott</property> <property name="hibernate.connection.password">tiger</property> <!-- <property name="hibernate.show\_sql">true</property> --> <!-- List of XML mapping files --> <mapping resource="department.hbm.xml" /> <mapping resource="employee.hbm.xml" /> <!-- mapping class="com.wipro.bean.Department" / --> </session-factory></hibernate-configuration>

EmpDeptMainClass.java

from Athma M to everyone: 12:31 PM

package mainpkg;import java.util.List;import org.hibernate.Query;import org.hibernate.Session;import org.hibernate.SessionFactory;import org.hibernate.Transaction;import org.hibernate.cfg.Configuration;public class EmpDeptMainClass { public static void main(String[] args) { Configuration cfg=new Configuration(); cfg.configure("hibernate.cfg.xml"); SessionFactory factory=cfg.buildSessionFactory(); Session session=factory.openSession(); Transaction tn=session.beginTransaction(); String hql = "FROM Employee where dept.dname like :dept\_name"; Query query = session.createQuery(hql); query.setParameter("dept\_name", "TT"); List listResult = query.list(); listResult.forEach(System.out::println); tn.commit(); session.close(); factory.close(); }}