create table gender(id int primary key, name varchar(90));

insert into gender values(1, 'male');

insert into gender values(2, 'female');

create table education(id int primary key, name varchar(90));

insert into education values(1, 'MCA');

insert into education values(2, 'B.Tech');

create table skill(id int primary key, name varchar(90));

insert into skill values(1, 'c');

insert into skill values(2, 'c++');

create table project(id int primary key, name varchar(90));

insert into project values(1, 'citi');

insert into project values(2, 'axis');

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.lara</groupId>

<artifactId>spring-hibernate</artifactId>

<version>0.0.1-SNAPSHOT</version>

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>4.0.6.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-orm</artifactId>

<version>4.0.6.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-tx</artifactId>

<version>4.0.6.RELEASE</version>

</dependency>

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>4.3.6.Final</version>

</dependency>

<dependency>

<groupId>com.oracle</groupId>

<artifactId>ojdbc5</artifactId>

<version>11.2.0</version>

</dependency>

</dependencies>

</project>

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

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:aop="http://www.springframework.org/schema/aop"

xmlns:tx="http://www.springframework.org/schema/tx"

xsi:schemaLocation="

http://www.springframework.org/schema/beans

http://www.springframework.org/schema/beans/spring-beans.xsd

http://www.springframework.org/schema/tx

http://www.springframework.org/schema/tx/spring-tx.xsd

http://www.springframework.org/schema/aop

http://www.springframework.org/schema/aop/spring-aop.xsd">

<tx:annotation-driven/>

<bean id="dmds"

class="org.springframework.jdbc.datasource.DriverManagerDataSource">

<property name="driverClassName"

value="oracle.jdbc.driver.OracleDriver"/>

<property name="url"

value="jdbc:oracle:thin:@localhost:1521:XE"/>

<property name="username" value="system"/>

<property name="password" value="great123"/>

</bean>

<bean id="sessionFactoryBean"

class="org.springframework.orm.hibernate4.LocalSessionFactoryBean">

<property name="dataSource" ref="dmds"/>

<property name="hibernateProperties">

<props>

<prop key="hibernate.dialect">org.hibernate.dialect.OracleDialect</prop>

<prop key="hibernate.hbm2ddl.auto">update</prop>

<prop key="hibernate.show\_sql">true</prop>

</props>

</property>

<property name="mappingResources">

<list>

<value>com/lara/Gender.hbm.xml</value>

<value>com/lara/Skill.hbm.xml</value>

<value>com/lara/Project.hbm.xml</value>

<value>com/lara/Education.hbm.xml</value>

<value>com/lara/Employee.hbm.xml</value>

</list>

</property>

</bean>

<bean id="transactionManager"

class="org.springframework.orm.hibernate4.HibernateTransactionManager">

<property name="sessionFactory"

ref="sessionFactoryBean" />

</bean>

<bean id="genderDao" class="com.lara.GenderDAO">

<property name="sessionFactory"

ref="sessionFactoryBean"/>

</bean>

<bean id="educationDao" class="com.lara.EducationDAO">

<property name="sessionFactory"

ref="sessionFactoryBean"/>

</bean>

<bean id="projectDao" class="com.lara.ProjectDAO">

<property name="sessionFactory"

ref="sessionFactoryBean"/>

</bean>

<bean id="skillDao" class="com.lara.SkillDAO">

<property name="sessionFactory"

ref="sessionFactoryBean"/>

</bean>

<bean id="empDao" class="com.lara.EmployeeDAO">

<property name="sessionFactory"

ref="sessionFactoryBean"/>

</bean>

</beans>

packagecom.lara;

public class Skill

{

privateint id;

private String name;

publicintgetId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<!--

-->

<hibernate-mapping package="com.lara">

<class name="Skill" table="Skill">

<id name="id"/>

<property name="name"/>

</class>

</hibernate-mapping>

packagecom.lara;

importjava.util.List;

importorg.hibernate.Session;

importorg.hibernate.SessionFactory;

importorg.springframework.transaction.annotation.Transactional;

public class SkillDAO

{

privateSessionFactorysessionFactory;

public void setSessionFactory(SessionFactorysessionFactory)

{

this.sessionFactory = sessionFactory;

}

@Transactional

public List<Skill> list()

{

Session session = sessionFactory.getCurrentSession();

return (List<Skill>) session.createCriteria(Skill.class).list();

}

}

packagecom.lara;

public class Project

{

privateint id;

private String name;

publicintgetId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<!--

-->

<hibernate-mapping package="com.lara">

<class name="Project" table="project">

<id name="id"/>

<property name="name"/>

</class>

</hibernate-mapping>

packagecom.lara;

importjava.util.List;

importorg.hibernate.Session;

importorg.hibernate.SessionFactory;

importorg.springframework.transaction.annotation.Transactional;

public class ProjectDAO

{

privateSessionFactorysessionFactory;

public void setSessionFactory(SessionFactorysessionFactory)

{

this.sessionFactory = sessionFactory;

}

@Transactional

public List<Project> list()

{

Session session = sessionFactory.getCurrentSession();

return (List<Project>) session.createCriteria(Project.class).list();

}

}

packagecom.lara;

public class Gender

{

privateint id;

private String name;

publicintgetId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<!--

-->

<hibernate-mapping package="com.lara">

<class name="Gender" table="Gender">

<id name="id"/>

<property name="name"/>

</class>

</hibernate-mapping>

packagecom.lara;

importjava.util.List;

importorg.hibernate.Session;

importorg.hibernate.SessionFactory;

importorg.springframework.transaction.annotation.Transactional;

public class GenderDAO

{

privateSessionFactorysessionFactory;

public void setSessionFactory(SessionFactorysessionFactory)

{

this.sessionFactory = sessionFactory;

}

@Transactional

public List<Gender> list()

{

Session session = sessionFactory.getCurrentSession();

return (List<Gender>) session.createCriteria(Gender.class).list();

}

}

packagecom.lara;

public class Education

{

privateint id;

private String name;

publicintgetId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

}

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<!--

-->

<hibernate-mapping package="com.lara">

<class name="Education" table="Education">

<id name="id"/>

<property name="name"/>

</class>

</hibernate-mapping>

packagecom.lara;

importjava.util.List;

importorg.hibernate.Session;

importorg.hibernate.SessionFactory;

importorg.springframework.transaction.annotation.Transactional;

public class EducationDAO

{

privateSessionFactorysessionFactory;

public void setSessionFactory(SessionFactorysessionFactory)

{

this.sessionFactory = sessionFactory;

}

@Transactional

public List<Education> list()

{

Session session = sessionFactory.getCurrentSession();

return (List<Education>) session.createCriteria(Education.class).list();

}

}

packagecom.lara;

importjava.util.Set;

public class Employee

{

privateint id;

private String firstName;

private String lastName;

privateint age;

private Gender gender;

private Education education;

private Project project;

private Set skills;

publicintgetId() {

return id;

}

public void setId(int id) {

this.id = id;

}

public String getFirstName() {

returnfirstName;

}

public void setFirstName(String firstName) {

this.firstName = firstName;

}

public String getLastName() {

returnlastName;

}

public void setLastName(String lastName) {

this.lastName = lastName;

}

publicintgetAge() {

return age;

}

public void setAge(int age) {

this.age = age;

}

public Gender getGender() {

return gender;

}

public void setGender(Gender gender) {

this.gender = gender;

}

public Education getEducation() {

return education;

}

public void setEducation(Education education) {

this.education = education;

}

public Project getProject() {

return project;

}

public void setProject(Project project) {

this.project = project;

}

public Set getSkills() {

return skills;

}

public void setSkills(Set skills) {

this.skills = skills;

}

}

<?xml version="1.0"?>

<!DOCTYPE hibernate-mapping PUBLIC

"-//Hibernate/Hibernate Mapping DTD 3.0//EN"

"http://www.hibernate.org/dtd/hibernate-mapping-3.0.dtd">

<!--

-->

<hibernate-mapping package="com.lara">

<class name="Employee" table="Employee">

<id name="id"/>

<property name="firstName"/>

<property name="lastName"/>

<property name="age"/>

<many-to-one name="gender"

class="Gender"

column="gender\_id"/>

<many-to-one name="education"

class="Education"

column="education\_id"/>

<many-to-one name="project"

class="Project"

column="project\_id"/>

<set name="skills" table="e\_s">

<key column="e\_id"/>

<many-to-many column="s\_id" class="Skill"/>

</set>

</class>

</hibernate-mapping>

packagecom.lara;

importjava.util.List;

importorg.hibernate.Session;

importorg.hibernate.SessionFactory;

importorg.springframework.transaction.annotation.Transactional;

public class EmployeeDAO

{

privateSessionFactorysessionFactory;

public void setSessionFactory(SessionFactorysessionFactory)

{

this.sessionFactory = sessionFactory;

}

@Transactional

publicint save(Employee emp)

{

Session session = sessionFactory.openSession();

return (Integer)session.save(emp);

}

}

packagecom.lara;

importjava.util.HashSet;

importjava.util.List;

importjava.util.Scanner;

importjava.util.Set;

import org.springframework.context.support.ClassPathXmlApplicationContext;

public class Manager

{

public static void main(String[] args)

{

ClassPathXmlApplicationContext context =

newClassPathXmlApplicationContext("beans.xml");

GenderDAOgenderDao =

(GenderDAO) context.getBean("genderDao");

EducationDAOeducationDao =

(EducationDAO) context.getBean("educationDao");

ProjectDAOprojectDao =

(ProjectDAO) context.getBean("projectDao");

SkillDAOskillDao =

(SkillDAO) context.getBean("skillDao");

EmployeeDAOemployeeDao =

(EmployeeDAO) context.getBean("empDao");

Employee emp = new Employee();

Scanner sc = new Scanner(System.in);

System.out.println("Enter Employee id");

emp.setId(sc.nextInt());

System.out.println("Enter Employee First Name");

emp.setFirstName(sc.next());

System.out.println("Enter Employee Last Name");

emp.setLastName(sc.next());

System.out.println("Enter Employee age");

emp.setAge(sc.nextInt());

List<Gender> genders = genderDao.list();

Gender gender = new Gender();

System.out.println("Available genders");

for(Gender g1 : genders)

{

System.out.print("enter ");

System.out.print(g1.getId() + " for ");

System.out.println(g1.getName());

}

gender.setId(sc.nextInt());

emp.setGender(gender);

List<Education> educations = educationDao.list();

Education education = new Education();

System.out.println("Available Educations");

for(Education e1 : educations)

{

System.out.print("enter ");

System.out.print(e1.getId() + " for ");

System.out.println(e1.getName());

}

education.setId(sc.nextInt());

emp.setEducation(education);

List<Project> projects = projectDao.list();

Project project = new Project();

System.out.println("Available Projects");

for(Project p1 : projects)

{

System.out.print("enter ");

System.out.print(p1.getId() + " for ");

System.out.println(p1.getName());

}

project.setId(sc.nextInt());

emp.setProject(project);

List<Skill> skills = skillDao.list();

System.out.println("Available Skills");

for(Skill s1 : skills)

{

System.out.print(s1.getId() + " for ");

System.out.println(s1.getName());

}

System.out.println("enter skills with comma(,) as a separator");

String skillsStr = sc.next();

String[] skillsStrArray = skillsStr.split(",");

Set set = new HashSet();

Skill skill = null;

for(int i = 0; i <skillsStrArray.length; i++)

{

skill = new Skill();

skill.setId(Integer.parseInt(skillsStrArray[i]));

set.add(skill);

}

emp.setSkills(set);

int id = employeeDao.save(emp);

System.out.println(id + " saved");

}

}