**Project name:StudentPro**

**Student.java**

package com.org;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name="student\_details")

public class Student {

@Id

@GeneratedValue

@Column(name="std\_id",unique=true)

private int rollno;

@Column(name="std\_name")

private String name;

@Column(name="std\_year")

private int year;

@Column(name="std\_semester")

private int semester;

@Column(name="std\_dpt")

private String dept;

static int stdcount;

static{

stdcount=0;

}

public Student(){

super();

}

public Student(int rollno, String name, int year, int semester, String dept) {

super();

this.rollno = rollno;

this.name = name;

this.year = year;

this.semester = semester;

this.dept = dept;

Student.stdcount++;

}

public int getRollno() {

return rollno;

}

public void setRollno(int rollno) {

this.rollno = rollno;

}

public String getName() {

return name;

}

public void setName(String name) {

this.name = name;

}

public int getYear() {

return year;

}

public void setYear(int year) {

this.year = year;

}

public int getSemester() {

return semester;

}

public void setSemester(int semester) {

this.semester = semester;

}

public String getDept() {

return dept;

}

public void setDept(String dept) {

this.dept = dept;

}

}

**StudentSolution.java**

package com.org;

import java.io.BufferedReader;

import java.io.IOException;

import java.io.InputStreamReader;

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.cfg.Configuration;

public class StudentSolution {

public static void main(String[] args) throws IOException {

SessionFactory sf = new Configuration().configure().buildSessionFactory();

Session session = sf.openSession();

BufferedReader bf = new BufferedReader(new InputStreamReader(System.in));

int rollno,no=0;

String name,n1;

int year,y1;

int semester,s1;

String dept,d1;

System.out.println("Inserting the values in database");

while(no<1){

session.beginTransaction();

System.out.println("Enter Student rollno");

rollno=Integer.valueOf(bf.readLine());

System.out.println("Enter Student name");

name=bf.readLine();

System.out.println("Enter Student year");

year=Integer.valueOf(bf.readLine());

System.out.println("Enter Student semester");

semester=Integer.valueOf(bf.readLine());

System.out.println("Enter Student Department");

dept=bf.readLine();

Student st=new Student(rollno, name, year, semester, dept);

session.save(st);

session.getTransaction().commit();

no++;

}

System.out.println("Total count of the Student is "+Student.stdcount);

System.out.println("Updating the values in database");

Student std=session.get(Student.class,1);

session.beginTransaction();

System.out.println("Enter Student name");

n1=bf.readLine();

System.out.println("Enter Student year");

y1=Integer.valueOf(bf.readLine());

System.out.println("Enter Student semester");

s1=Integer.valueOf(bf.readLine());

System.out.println("Enter Student Department");

d1=bf.readLine();

std.setName(n1);

std.setYear(y1);

std.setSemester(s1);

std.setDept(d1);

session.update(std);

session.getTransaction().commit();

System.out.println("The Values is updated");

System.out.println("Deleting the values in database");

session.beginTransaction();

Student stu=session.get(Student.class,2);

session.delete(stu);

session.getTransaction().commit();

System.out.println("Deleted");

session.close();

}

}

Hibernate.cfg.xml

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

<!-- ~ Hibernate, Relational Persistence for Idiomatic Java ~ ~ License:

GNU Lesser General Public License (LGPL), version 2.1 or later. ~ See the

lgpl.txt file in the root directory or <http://www.gnu.org/licenses/lgpl-2.1.html>. -->

<!DOCTYPE hibernate-configuration PUBLIC

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

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

<hibernate-configuration>

<session-factory>

<!-- Database connection settings -->

<property name="connection.driver\_class">com.mysql.jdbc.Driver</property>

<property name="connection.url">jdbc:mysql://localhost:3306/sample</property>

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

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

<!-- JDBC connection pool (use the built-in) -->

<property name="connection.pool\_size">10</property>

<!-- SQL dialect -->

<property name="dialect">org.hibernate.dialect.MySQL5Dialect</property>

<!-- Disable the second-level cache -->

<property name="cache.provider\_class">org.hibernate.cache.internal.NoCacheProvider</property>

<!-- Echo all executed SQL to stdout -->

<property name="show\_sql">true</property>

<!-- Drop and re-create the database schema on startup -->

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

<!-- Names the annotated entity class -->

<!-- <mapping class="com.org.Employee" /> -->

<mapping class="com.org.Student" />

</session-factory>

</hibernate-configuration>