SHIJU THOMAS M.Y
DEPT. OF COMPUTER SCIENCE
RCSS

Employee Management System using Annotation

Project: Hn_wa1

Table Structure

```
create table EMPLOYEE (
   id INT NOT NULL auto_increment,
   first_name VARCHAR(20) default NULL,
   last_name  VARCHAR(20) default NULL,
   salary     INT  default NULL,
   PRIMARY KEY (id)
);
```

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="hibernate.dialect">org.hibernate.dialect.MySQLDialect</property>

  <property name="hibernate.connection.driver_class">com.mysql.jdbc.Driver</property>

  <property
name="hibernate.connection.url">jdbc:mysql://localhost:3306/empms?zeroDateTimeBehavior=con
vertToNull</property>
```

```
  <property name="hibernate.connection.username">root</property>
 </session-factory>
</hibernate-configuration>
```

Employee.java

```
package hn_wa1;
import javax.persistence.*;
@Entity
@Table(name = "EMPLOYEE")
public class Employee {
  @Id @GeneratedValue
  @Column(name = "id")
  private int id;
  @Column(name = "first_name")
  private String firstName;
  @Column(name = "last_name")
  private String lastName;
  @Column(name = "salary")
  private int salary;
  public Employee() {}
  public int getId() {
    return id;
  }
  public void setId( int id ) {
    this.id = id;
  }
  public String getFirstName() {
    return firstName;
  }
```

```
  public void setFirstName( String first_name ) {
    this.firstName = first_name;
  }
  public String getLastName() {
    return lastName;
  }
  public void setLastName( String last_name ) {
    this.lastName = last_name;
  }
   public int getSalary() {
    return salary;
  }
  public void setSalary( int salary ) {
    this.salary = salary;
  }
}
```

Hn_wa1.java

```
package hn_wa1;
import java.util.List;
import java.util.Date;
import java.util.Iterator;

import org.hibernate.HibernateException;
import org.hibernate.Session;
import org.hibernate.Transaction;
import org.hibernate.cfg.AnnotationConfiguration;
import org.hibernate.SessionFactory;
```

```
import org.hibernate.cfg.Configuration;

/**
 *
 * @author Administrator
 */
public class Hn_wa1 {

   private static SessionFactory factory;
  public static void main(String[] args) {

     try {
       factory = new
AnnotationConfiguration().configure().addAnnotatedClass(Employee.class).buildSessionFactory();
     } catch (Throwable ex) {
       System.err.println("Failed to create sessionFactory object." + ex);
       throw new ExceptionInInitializerError(ex);
     }

     Hn_wa1 ME = new Hn_wa1();



     Integer empID1 = ME.addEmployee("abcd1", "abcd2", 1000);
     Integer empID2 = ME.addEmployee("asdf1", "asdf2", 5000);
     Integer empID3 = ME.addEmployee("bbc1", "bbc2", 10000);



     ME.listEmployees();
```

```
  ME.updateEmployee(empID1, 5000);


  //ME.deleteEmployee(empID2);


  ME.listEmployees();
}


public Integer addEmployee(String fname, String lname, int salary){
  Session session = factory.openSession();
  Transaction tx = null;
  Integer employeeID = null;

  try {
    tx = session.beginTransaction();
    Employee employee = new Employee();
    employee.setFirstName(fname);
    employee.setLastName(lname);
    employee.setSalary(salary);
    employeeID = (Integer) session.save(employee);
    tx.commit();
  } catch (HibernateException e) {
    if (tx!=null) tx.rollback();
    e.printStackTrace();
  } finally {
    session.close();
```

```
  }
  return employeeID;
}


public void listEmployees( ){
  Session session = factory.openSession();
  Transaction tx = null;

  try {
    tx = session.beginTransaction();
    List employees = session.createQuery("FROM Employee").list();
    for (Iterator iterator = employees.iterator(); iterator.hasNext();){
      Employee employee = (Employee) iterator.next();
      System.out.print("First Name: " + employee.getFirstName());
      System.out.print("  Last Name: " + employee.getLastName());
      System.out.println("  Salary: " + employee.getSalary());
    }
    tx.commit();
  } catch (HibernateException e) {
    if (tx!=null) tx.rollback();
    e.printStackTrace();
  } finally {
    session.close();
  }
}


public void updateEmployee(Integer EmployeeID, int salary ){
```

```
  Session session = factory.openSession();
  Transaction tx = null;

  try {
    tx = session.beginTransaction();
    Employee employee = (Employee)session.get(Employee.class, EmployeeID);
    employee.setSalary( salary );
                session.update(employee);
    tx.commit();
  } catch (HibernateException e) {
    if (tx!=null) tx.rollback();
    e.printStackTrace();
  } finally {
    session.close();
  }
}


public void deleteEmployee(Integer EmployeeID){
  Session session = factory.openSession();
  Transaction tx = null;

  try {
    tx = session.beginTransaction();
    Employee employee = (Employee)session.get(Employee.class, EmployeeID);
    session.delete(employee);
    tx.commit();
  } catch (HibernateException e) {
    if (tx!=null) tx.rollback();
```

```
        e.printStackTrace();
      } finally {
        session.close();
      }
    }
}
```