

Bachashankar

Name: Madhura Itke

Program Name: - HQL-App

```
package com.app.bean;
```

```
import javax.persistence.Entity;
```

```
import javax.persistence.Id;
```

```
import javax.persistence.Table;
```

```
@Entity
```

```
@Table(name = "employee_hql")
```

```
public class Employee {
```

```
    @Id
```

```
    private int empld;
```

```
    private String empName;
```

```
    private String empAddress;
```

```
    private double empSal;
```

```
    public int getEmpld() {
```

```
        return empld;
```

```
    }
```

```
    public void setEmpld(int empld) {
```

```
        this.empld = empld;
```

```
    }
```

```
    public String getEmpName() {
```

```
        return empName;
```

```
    }
```

```
    public void setEmpName(String empName) {
```

```
        this.empName = empName;
```

```
    }
```

```
    public String getEmpAddress() {
```

```
        return empAddress;
```

```
    }
```

```
    public void setEmpAddress(String empAddress) {
```

```
this.empAddress = empAddress;
```

```

    }

    public double getEmpSal() {
        return empSal;
    }

    public void setEmpSal(double empSal) {
        this.empSal = empSal;
    }

    public Employee(int empId, String empName, String empAddress, double empSal) {
        super();
        this.empId = empId;
        this.empName = empName;
        this.empAddress = empAddress;
        this.empSal = empSal;
    }

    public Employee() {
        super();
        // TODO Auto-generated constructor stub
    }

```

```

}

```

```

package com.app.factory;

```

```

import com.app.dao.EmployeeDao;

```

```

import com.app.dao.impl.EmployeeDaoImpl;

```

```

public class EmployeeFactory {
    public static EmployeeDao getEmployeeDao() {
        return new EmployeeDaoImpl();
    }
}

```

```

}

```

```

package com.app.dao;

```

```

import java.util.List;

```

```
import com.app.bean.Employee;
```

```
public interface EmployeeDao {  
    int updateData(Employee emp);  
    int insertData(Employee emp);  
    int deleteData(int id);  
    List<Employee> listEmployee();  
    List<Employee> getEmployee(int id);  
}
```

```
package com.app.dao.impl;
```

```
import java.util.List;
```

```
import org.hibernate.Session;
```

```
import org.hibernate.Transaction;
```

```
import org.hibernate.query.Query;
```

```
import com.app.bean.Employee;
```

```
import com.app.dao.EmployeeDao;
```

```
import com.app.utility.EmployeeUtil;
```

```
public class EmployeeDaoImpl implements EmployeeDao{
```

```
    public int updateData(Employee emp) {
```

```
        // TODO Auto-generated method stub
```

```
        Session session=EmployeeUtil.getSession();
```

```
        Transaction tx=null;
```

```
        try {
```

```
            tx=session.beginTransaction();
```

```
            Query<Employee>query=session.createQuery("update Employee set  
empAddress='"+emp.getEmpAddress()+"' where empId="+emp.getEmpId());
```

```
            session.update(emp);
```

```
        tx.commit();
        EmployeeUtil.closeSession();
        return 1;
    }
}
```

```
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
        tx.rollback();
        return 0;
    }
}
```

```
}
```

```
public int insertData(Employee emp) {
    // TODO Auto-generated method stub
```

```
    Session session=EmployeeUtil.getSession();
```

```
    Transaction tx=null;
```

```
    try {
```

```
        tx=session.beginTransaction();
```

```
        session.persist(emp);
```

```
        tx.commit();
```

```
        EmployeeUtil.closeSession();
```

```
        return 1;
```

```
    } catch (Exception e) {
```

```
        // TODO: handle exception
```

```
        e.printStackTrace();
```

```
        tx.rollback();
```

```
        return 0;
```

```
    }
```

```
}
```

```
public int deleteData(int id) {
```

```

        // TODO Auto-generated method stub
        Session session=EmployeeUtil.getSession();
        Transaction tx=null;
        try {
            tx=session.beginTransaction();

String hql="delete from Employee where empId =" +id;

            Query<Employee>query=session.createQuery(hql);

            int row=query.executeUpdate();
            tx.commit();
            EmployeeUtil.closeSession();
            return row;
        } catch (Exception e) {
            // TODO: handle exception
            e.printStackTrace();
            tx.rollback();
            return 0;
        }
    }
}

```

```

public List<Employee> listEmployee() {
    // TODO Auto-generated method stub
    Session session=EmployeeUtil.getSession();
    Transaction tx=null;
    String hql="From Employee";
    Query<Employee>query=session.createQuery(hql);
    List<Employee> list=query.list();
    EmployeeUtil.closeSession();
    return list;
}

```

```

public List<Employee> getEmployee(int id) {
    // TODO Auto-generated method stub

```

```

        // TODO Auto-generated method stub
        Session session=EmployeeUtil.getSession();
        Transaction tx=null;
        String hql="From Employee Where empld =" +id;
        Query<Employee>query=session.createQuery(hql);
        //query.setParameter(1, id);
        List<Employee> list=query.list();
        EmployeeUtil.closeSession();
        return list;
    }

}

package com.app.utility;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.cfg.Configuration;

public class EmployeeUtil {
    private static SessionFactory factory;
    static {
        try {
            factory = new
Configuration().configure("com/app/config/employee.cfg.xml").buildSessionFactory();

        } catch (Exception e) {
            e.printStackTrace();
        }
    }
    static ThreadLocal<Session> local=new ThreadLocal();
    static Session session=null;
    public static Session getSession() {
        try {

```

```

        if(local.get()==null) {
            session=factory.openSession();
            local.set(session);
            return session;
        }else {
            return local.get();
        }
    } catch (Exception e) {
        // TODO: handle exception
        return null;
    }
}

public static void closeSession() {
    try {
        session.close();
    } catch (Exception e) {
        // TODO: handle exception
        e.printStackTrace();
    }
}
}
}

```

```
package com.app.test;
```

```
import java.util.List;
```

```
import java.util.Scanner;
```

```
import com.app.bean.Employee;
```

```
import com.app.dao.EmployeeDao;
```

```
import com.app.factory.EmployeeFactory;
```

```
public class Client {
```

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

```
        // TODO Auto-generated method stub
    }
}

```



```

EmployeeDao empDao=EmployeeFactory.getEmployeeDao();

Scanner sn =new Scanner(System.in);

int choice;

String conti;

do {

    System.out.println("!-----HQL Operation----- !");

    System.out.println("1. insert data");

    System.out.println("2. update data");

    System.out.println("3. delete data");

    System.out.println("4. Get All data");

    System.out.println("5. get Single data");

    System.out.println("!-----End -----!");

    System.out.println("Enter you choice:");

    choice=sn.nextInt();


    switch (choice) {

    case 1:

        System.out.println("Enter your id:");

        int id=sn.nextInt();

        System.out.println("Enter your name:");

        String name=sn.next();

        System.out.println("Enter your Address:");

        String address=sn.next();

        System.out.println("Enter your Salary:");

        double sal=sn.nextDouble();

        Employee emp=new Employee(id, name, address, sal);

        int i=empDao.insertData(emp);

        if(i==1)

        {

            System.out.println("Data inserted successfully.");

        }else {

            System.out.println("Data Not Inserted something went wrong..!");

        }

        break;

```

case 2:

```
System.out.println("Enter your id:");
int id2=sn.nextInt();
System.out.println("Enter your name:");
String name1=sn.next();
System.out.println("Enter your Address:");
String address1=sn.next();
System.out.println("Enter your Salary:");
double sal1=sn.nextDouble();
Employee emp5=new Employee(id2, name1, address1, sal1);
int i2=empDao.updateData(emp5);
if(i2==1)
{
    System.out.println("Data update successfully.");
}else {
    System.out.println("Data Not Inserted something went wrong..!");
}
break;
```

case 3:

```
System.out.println("Enter your id:");
int id1=sn.nextInt();

int row=empDao.deleteData(id1);
if(row==1)
{
    System.out.println("Data deleted successfully.");
}else {
    System.out.println("Data Not Inserted something went wrong..!");
}
break;
```

case 4:

```

List<Employee> list=empDao.listEmployee();
if(list!=null)
{
    for(Employee e:list) {
        System.out.println(e.getEmpId()+"\t"+e.getEmpName()+"\t"
            +e.getEmpAddress()+"\t"+e.getEmpSal());
    }
}
else {
    System.out.println("something went wrong..!");
}
break;
case 5:
    System.out.println("Enter your id:");
    int empId=sn.nextInt();

    List<Employee> emp1=empDao.getEmployee(empId);
    if(emp1!=null)
    {
        for(Employee e:emp1) {
            System.out.println(e.getEmpId()+"\t"+e.getEmpName()+"\t"
                +e.getEmpAddress()+"\t"+e.getEmpSal());
        }
    }
    else {
        System.out.println("Data Not Inserted something went wrong..!");
    }
    break;
default:
    break;
}

System.out.println("do you want to continue...!");
conti=sn.next();

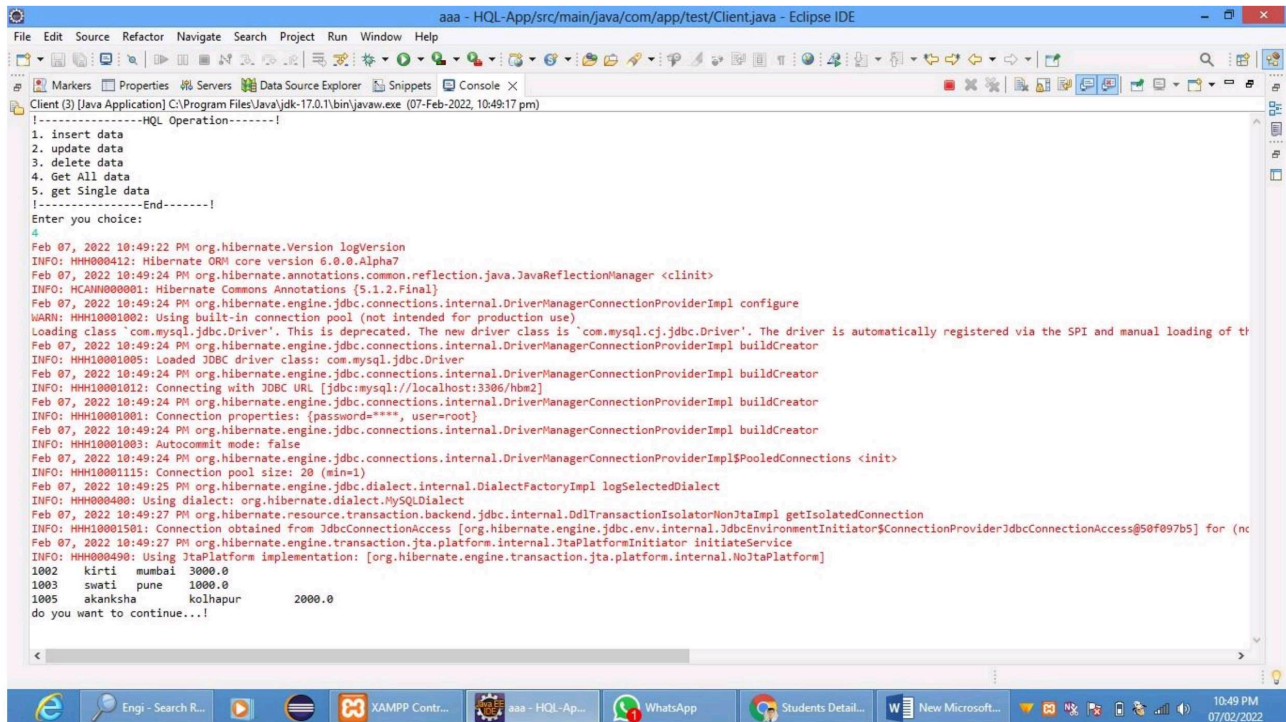
```

```
}while(conti.equalsIgnoreCase("y"));
```

```
}
```

```
}
```

Output



The screenshot shows the Eclipse IDE interface with the console window open. The console displays the output of a Java application named 'Client (3) [Java Application]'. The output includes a menu of options (1. insert data, 2. update data, 3. delete data, 4. Get All data, 5. get Single data) and a prompt 'Enter you choice:'. The user has entered '4'. The application then displays a table of data:

ID	Name	City	Age
1002	kirti	mumbai	3000.0
1003	swati	pune	1000.0
1005	akanksha	kolhapur	2000.0

The application also displays a prompt 'do you want to continue...!'. The console also shows various Hibernate and JDBC logs, including connection pool configuration, driver loading, and connection establishment details.