Name: - Nandkishor Desai

Program Name: - Hibernate-App-oneToMany & Hibernate-App-ManyToOne

```
package com.app.model;

import java.util.List;

import javax.persistence.CascadeType;
import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.OneToMany;
import javax.persistence.OrderColumn;
import javax.persistence.Table;

@Entity
@Table(name = "user_table")
public class User {
	@Id
	@Column(name = "user_id")
	private int userId;
	@Column(name = "first_name")
	private String fName;
	@Column(name = "last_Name")
	private String lname;
	@OneToMany(targetEntity = PhoneNumber.class, cascade = CascadeType.ALL,
			orphanRemoval = true)
	@JoinColumn(name="unid",referencedColumnName = "user_id")
	@OrderColumn(name = "list_index")
	private List<PhoneNumber> phoneNumber;
	public int getUserId() {
		return userId;
	}
	public void setUserId(int userId) {
		this.userId = userId;
	}
	public String getfName() {
		return fName;
	}
	public void setfName(String fName) {
		this.fName = fName;
	}
	public String getLname() {
		return lname;
	}
	public void setLname(String lname) {
		this.lname = lname;
	}
	@Override
	public String toString() {
		return "User [userId=" + userId + ", fName=" + fName + ", lname=" + lname + ", phoneNumber=" + 
phoneNumber
				+ "]";
	}
	public User() {
		super();
		// TODO Auto-generated constructor stub
	}
	public User(int userId, String fName, String lname, List<PhoneNumber> phoneNumber) {
		super();
		this.userId = 001;
		this.fName = "abc";
		this.lname = "xyz";
		this.phoneNumber = phoneNumber;
	}
	public List<PhoneNumber> getPhoneNumber() {
		return phoneNumber;
	}
	public void setPhoneNumber(List<PhoneNumber> phoneNumber) {
		this.phoneNumber = phoneNumber;
	}
}
```

```
package com.app.model;


import javax.persistence.Entity;
import javax.persistence.Id;


@Entity
public class Department {
@Id
	private int deptno;
	private String deptName;
	private String deptHead;
	public int getDeptno() {
		return deptno;
	}
	public void setDeptno(int deptno) {
		this.deptno = deptno;
	}
	public String getDeptName() {
		return deptName;
	}
	public void setDeptName(String deptName) {
		this.deptName = deptName;
	}
	public String getDeptHead() {
		return deptHead;
	}
	public void setDeptHead(String deptHead) {
		this.deptHead = deptHead;
```

```
    }
    public Department() {
        super();
        // TODO Auto-generated constructor stub
    }
    public Department(int deptno, String deptName, String deptHead) {
        super();
        this.deptno = deptno;
        this.deptName = deptName;
        this.deptHead = deptHead;
    }
}------------------------------------------------------------------------------------------------
package com.app.model;


import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.Id;
import javax.persistence.Table;


@Entity
@Table(name = "phoneNumber")
public class PhoneNumber {
@Id
    private int phone;
    @Column(name = "number_type")
    private String numberType;
    public int getPhone() {
        return phone;
    }
    public void setPhone(int phone) {
        this.phone = phone;
    }
    public String getNumberType() {
        return numberType;
    }
    public void setNumberType(String numberType) {
        this.numberType = numberType;
    }
    @Override
    public String toString() {
        return "PhoneNumber [phone=" + phone + ", numberType=" + numberType + "]";
    }

}
```

```
package com.app.model;


import javax.persistence.CascadeType;
import javax.persistence.Entity;
import javax.persistence.FetchType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;


@Entity
public class EmpDetails {
@Id
    private int eno;
    private String ename;
    private long salary;
    @ManyToOne(targetEntity = Department.class, cascade=CascadeType.ALL,fetch =FetchType.EAGER )
    @JoinColumn(name = "deptno",referencedColumnName = "deptno")
    private Department department;
    public int getEno() {
        return eno;
    }
    public void setEno(int eno) {
        this.eno = eno;
    }
```

```
	public String getEname() {
		return ename;
	}
	public void setEname(String ename) {
		this.ename = ename;
	}
	public long getSalary() {
		return salary;
	}
	public void setSalary(long salary) {
		this.salary = salary;
	}
	public Department getDepartment() {
		return department;
	}
	public void setDepartment(Department department) {
		this.department = department;
	}
	public EmpDetails(int eno, String ename, long salary, Department department) {
		super();
		this.eno = eno;
		this.ename = ename;
		this.salary = salary;
		this.department = department;
	}
	public EmpDetails() {
		// TODO Auto-generated constructor stub
	}
	public void setDepartment() {
		// TODO Auto-generated method stub

	}

}
```

```
package com.app.model;

import javax.persistence.CascadeType;
import javax.persistence.Entity;
import javax.persistence.FetchType;
import javax.persistence.Id;
import javax.persistence.JoinColumn;
import javax.persistence.ManyToOne;

@Entity
public class EmpDetails {
@Id
        private int eno;
        private String ename;
        private long salary;
        @ManyToOne(targetEntity = Department.class, cascade=CascadeType.ALL,fetch =FetchType.EAGER )
        @JoinColumn(name = "deptno",referencedColumnName = "deptno")
        private Department department;
        public int getEno() {
                return eno;
        }
        public void setEno(int eno) {
                this.eno = eno;
        }
        public String getEname() {
                return ename;
        }
        public void setEname(String ename) {
                this.ename = ename;
        }
        public long getSalary() {
                return salary;
        }
        public void setSalary(long salary) {
```

```
		this.salary = salary;
	}
	public Department getDepartment() {
		return department;
	}
	public void setDepartment(Department department) {
		this.department = department;
	}
	public EmpDetails(int eno, String ename, long salary, Department department) {
		super();
		this.eno = eno;
		this.ename = ename;
		this.salary = salary;
		this.department = department;
	}
	public EmpDetails() {
		// TODO Auto-generated constructor stub
	}
	public void setDepartment() {
		// TODO Auto-generated method stub

	}

}
```

```
package com.app.factory;


import com.app.dao.ManyToOneDao;
import com.app.dao.OneToManyDao;
import com.app.dao.impl.ManyToOneDaoImpl;
import com.app.dao.impl.OneToManyDaoImpl;
	//OneToManyDao one= new OneToManyDaoImpl();
public class OneToManyFactory {
	public static OneToManyDao getInstance() {
		return new OneToManyDaoImpl();
	}

	public static ManyToOneDao getManyInstance() {

		return new ManyToOneDaoImpl();

	}

}
```

```
package com.app.dao;

public interface ManyToOneDao {
	void addEmployeeWithDept();


}
```

```
package com.app.dao;

public interface OneToManyDao {
	void insertData(); //public Abstract void insertData;
	void listofData();
}
```

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



import java.util.ArrayList;

import java.util.List;



import org.hibernate.Session;

import org.hibernate.Transaction;



import com.app.dao.ManyToOneDao;

import com.app.model.Department;

import com.app.model.EmpDetails;

import com.app.util.UtilityClass;



public class ManyToOneDaoImpl implements ManyToOneDao {



	public void addEmployeeWithDept() {

		// TODO Auto-generated method stub

		Session session=UtilityClass.getSession();



		Department dept1= new Department(1,"HR","Wakle");

		Department dept2=new Department(2,"Production","Shinde");
```

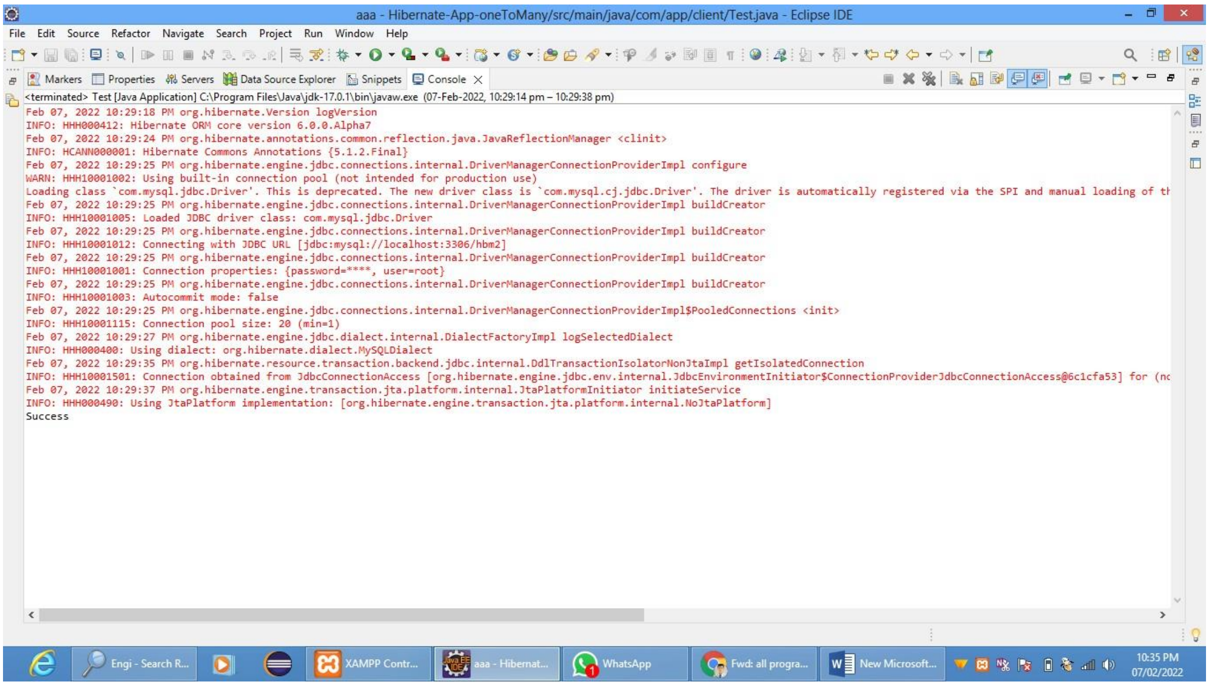
```
        List<Department> dept=new ArrayList<Department>();
        dept.add(dept1);
        dept.add(dept2);



        //EmpDetails emp1=new EmpDetails(5001,"atul",50098,dept);
        //EmpDetails emp2=new EmpDetails(5002, "Anant", 6542, dept2);

        EmpDetails em=new EmpDetails();
        em.setDepartment(dept2);
        em.setSalary(2020);
        em.setEname("Pallavi");
        em.setEno(5003);



        Transaction tx=session.beginTransaction();
        session.update(em);
        //session.saveOrUpdate(emp1);
        tx.commit();
        UtilityClass.closeSession();
    }

}
```

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


import java.util.ArrayList;
import java.util.List;


import org.hibernate.Transaction;
import org.hibernate.query.Query;
import org.hibernate.Session;


import com.app.dao.OneToManyDao;
import com.app.model.PhoneNumber;
import com.app.model.User;
import com.app.util.UtilityClass;

public class OneToManyDaoImpl implements OneToManyDao {

	public void insertData() {
		// TODO Auto-generated method stub
		Session session=UtilityClass.getSession();
		Transaction tx=session.beginTransaction();

		PhoneNumber phoneNumber=new PhoneNumber();
		phoneNumber.setNumberType("home");
		phoneNumber.setPhone(97671343);

		PhoneNumber phoneNumber1=new PhoneNumber();
		phoneNumber1.setNumberType("office");
		phoneNumber1.setPhone(876543);

		List<PhoneNumber> list= new ArrayList<PhoneNumber> ();
		list.add(phoneNumber1);
		list.add(phoneNumber);

		User user=new User();
		user.setfName("Atul");
		user.setLname("Wakle");
		user.setUserId(101);
		user.setPhoneNumber(list);
		session.save(user);
		tx.commit();
		UtilityClass.closeSession();
	}

	public void listofData() {
        // TODO Auto-generated method stub
        Session session=UtilityClass.getSession();
        Query<User>query=session.createQuery("from User");
        List<User>list=query.list();
        for(User user:list) {

    System.out.println(user.getUserId()+"\t"+user.getfName()+"\t"+user.getLname()+"\t"+user.getPhoneNumber());
        }
        UtilityClass.closeSession();
    }


}
```

```
package com.app.util;


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




public class UtilityClass {
    private static SessionFactory factory;
    static {
        try {
            factory=new Configuration().configure("Hibernet-cfg.xml").buildSessionFactory();


        /*  Configuration configuration=new Configuration();
            configuration.configure("Hibernate-cfg.xml");
            factory=configuration.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.client;


import com.app.dao.ManyToOneDao;
import com.app.dao.OneToManyDao;
import com.app.factory.OneToManyFactory;


public class Test {
        public static void main(String[] args) {
                // TODO Auto-generated method stub
                //OneToManyDao dao=OneToManyFactory.getInstance();
                //dao.insertData();
                //dao.listofData();

                ManyToOneDao dao= OneToManyFactory.getManyInstance();
                dao.addEmployeeWithDept();
                System.out.println("Success");
        }

}
```

Output



```
<terminated> Test [Java Application] C:\Program Files\Java\jdk-17.0.1\bin\javaw.exe (07-Feb-2022, 10:29:14 pm – 10:29:38 pm)
Feb 07, 2022 10:29:18 PM org.hibernate.Version logVersion
INFO: HHH000412: Hibernate ORM core version 6.0.0.Alpha7
Feb 07, 2022 10:29:24 PM org.hibernate.annotations.common.reflection.java.JavaReflectionManager <clinit>
INFO: HCANN000001: Hibernate Commons Annotations {5.1.2.Final}
Feb 07, 2022 10:29:25 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl configure
WARN: HHH10001002: Using built-in connection pool (not intended for production use)
Loading class `com.mysql.jdbc.Driver'. This is deprecated. The new driver class is `com.mysql.cj.jdbc.Driver'. The driver is automatically registered via the SPI and manual loading of th
Feb 07, 2022 10:29:25 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH10001005: Loaded JDBC driver class: com.mysql.jdbc.Driver
Feb 07, 2022 10:29:25 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH10001012: Connecting with JDBC URL [jdbc:mysql://localhost:3306/hbm2]
Feb 07, 2022 10:29:25 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH10001001: Connection properties: {password=****, user=root}
Feb 07, 2022 10:29:25 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl buildCreator
INFO: HHH10001003: Autocommit mode: false
Feb 07, 2022 10:29:25 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl$PooledConnections <init>
INFO: HHH10001115: Connection pool size: 20 (min=1)
Feb 07, 2022 10:29:27 PM org.hibernate.engine.jdbc.dialect.internal.DialectFactoryImpl logSelectedDialect
INFO: HHH000400: Using dialect: org.hibernate.dialect.MySQLDialect
Feb 07, 2022 10:29:35 PM org.hibernate.resource.transaction.backend.jdbc.internal.DdlTransactionIsolatorNonJtaImpl getIsolatedConnection
INFO: HHH10001501: Connection obtained from JdbcConnectionAccess [org.hibernate.engine.jdbc.env.internal.JdbcEnvironmentInitiator$ConnectionProviderJdbcConnectionAccess@6c1cfa53] for (nc
Feb 07, 2022 10:29:37 PM org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformInitiator initiateService
INFO: HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
Success
```