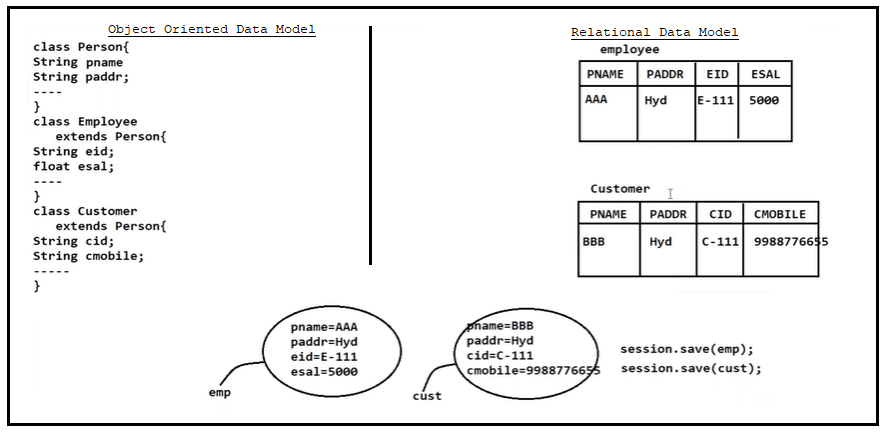
**Table per concrete Class:**

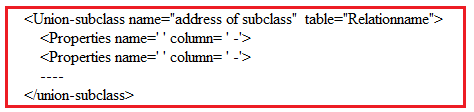
In Relation DB, There is separate relation for every sub class. This relation should have columns with respect to sub class and super class.



**1.Mapping Using XML tags:-**

a.<union-subclass>:- This is child tag to <class> tag. It provides mapping between subclass and relation.

Syntax:

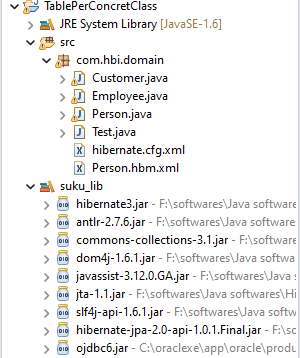


**2.Mapping Using annotations:-**

**@Inheritance(strategy=”inheritanceType.TABLE\_\_PER\_CLASS”)**

This annotation is only applied to superclass.

Example:1 Using XML tags.



**Domain classes:**

**1.Person.java**

**package** com.hbi.domain;

**import** java.io.Serializable;

**public** **class** Person **implements** Serializable {

**private** String pname;

**private** String paddr;

**public** String getPname() {

**return** pname;

}

**public** **void** setPname(String pname) {

**this**.pname = pname;

}

**public** String getPaddr() {

**return** paddr;

}

**public** **void** setPaddr(String paddr) {

**this**.paddr = paddr;

}

}

**2.Customer.java**

**package** com.hbi.domain;

**public** **class** Customer **extends** Person {

**private** **int** cid;

**private** String cmobile;

**public** **int** getCid() {

**return** cid;

}

**public** **void** setCid(**int** cid) {

**this**.cid = cid;

}

**public** String getCmobile() {

**return** cmobile;

}

**public** **void** setCmobile(String cmobile) {

**this**.cmobile = cmobile;

}

}

**3.Employee.java**

**package** com.hbi.domain;

**public** **class** Employee **extends** Person {

**private** **int** eid;

**private** **float** esal;

**public** **int** getEid() {

**return** eid;

}

**public** **void** setEid(**int** eid) {

**this**.eid = eid;

}

**public** **float** getEsal() {

**return** esal;

}

**public** **void** setEsal(**float** esal) {

**this**.esal = esal;

}

}

**HibernateMappingFile: Person.hbm.xml**

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

<!DOCTYPE hibernate-mapping PUBLIC

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

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

<hibernate-mapping>

<class name=*"com.hbi.domain.Person"*>

<id name=*"pname"* column=*"pname"*/>

<property name=*"paddr"* column=*"paddr"*/>

<union-subclass name=*"com.hbi.domain.Customer"* table=*"customer"*>

<property name=*"cid"* column=*"cid"*/>

<property name=*"cmobile"* column=*"cmobile"*/>

</union-subclass>

<union-subclass name=*"com.hbi.domain.Employee"* table=*"employee"*>

<property name=*"eid"* column=*"eid"*/>

<property name=*"esal"* column=*"esal"*/>

</union-subclass>

</class>

</hibernate-mapping>

**HibernateConfigurationFile:hibernate.cfg.xml**

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

<!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>

<property name=*"hibernate.connection.driver\_class"*>oracle.jdbc.OracleDriver</property>

<property name=*"hibernate.connection.url"*>jdbc:oracle:thin:@localhost:1521:xe</property>

<property name=*"hibernate.connection.username"*>system</property>

<property name=*"hibernate.connection.password"*>tiger</property>

<property name=*"hibernate.dialect"*>org.hibernate.dialect.OracleDialect</property>

<mapping resource=*"com/hbi/domain/Person.hbm.xml"*/>

</session-factory>

</hibernate-configuration>

**ClientApplication:Test.java**

package com.hbi.domain;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

import org.hibernate.cfg.Configuration;

import org.hibernate.classic.Session;

public class Test {

public static void main(String[] args) {

Configuration cfg=new Configuration();

cfg.configure("com/hbi/domain/hibernate.cfg.xml");

SessionFactory sf=cfg.buildSessionFactory();

Session s=sf.openSession();

Transaction t=s.beginTransaction();

Employee e=new Employee();

Customer c=new Customer();

e.setEid(1);

e.setEsal(7000);

e.setPaddr("nlr");

e.setPname("suku");

c.setCid(8);

c.setCmobile("7989");

c.setPname("sv");

c.setPaddr("chi");

s.save(e);

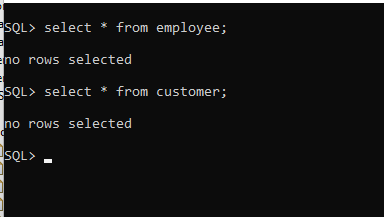
s.save(c);

t.commit();

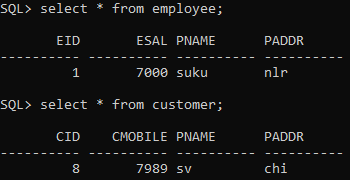
}

}

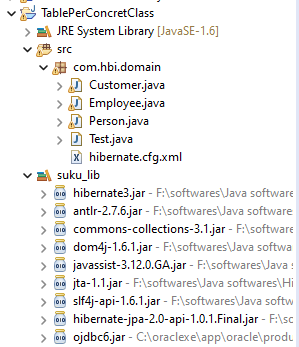
**Output:-**

****

**After program execution completed**

****

**Example2: Using annotations.**

****

**Domain /annotated classes:**

**1.person.java**

**package** com.hbi.domain;

**import** java.io.Serializable;

**import** javax.persistence.Column;

**import** javax.persistence.Entity;

**import** javax.persistence.Id;

**import** javax.persistence.Inheritance;

**import** javax.persistence.InheritanceType;

**import** javax.persistence.Table;

@Entity

@Inheritance(strategy=InheritanceType.***TABLE\_PER\_CLASS***)

**public** **class** Person **implements** Serializable {

@Id

@Column(name="pname")

**private** String pname;

@Column(name="paddr")

**private** String paddr;

**public** String getPname() {

**return** pname;

}

**public** **void** setPname(String pname) {

**this**.pname = pname;

}

**public** String getPaddr() {

**return** paddr;

}

**public** **void** setPaddr(String paddr) {

**this**.paddr = paddr;

}

}

**2.Employee.java**

**package** com.hbi.domain;

**import** javax.persistence.Column;

**import** javax.persistence.Entity;

**import** javax.persistence.Table;

@Entity

@Table(name="employee")

**public** **class** Employee **extends** Person {

@Column(name="eid")

**private** **int** eid;

@Column(name="esal")

**private** **float** esal;

**public** **int** getEid() {

**return** eid;

}

**public** **void** setEid(**int** eid) {

**this**.eid = eid;

}

**public** **float** getEsal() {

**return** esal;

}

**public** **void** setEsal(**float** esal) {

**this**.esal = esal;

}

}

**3.Customer.java**

**package** com.hbi.domain;

**import** javax.persistence.Column;

**import** javax.persistence.Entity;

**import** javax.persistence.Table;

@Entity

@Table(name="customer")

**public** **class** Customer **extends** Person {

@Column(name="cid")

**private** **int** cid;

@Column(name="cmobile")

**private** String cmobile;

**public** **int** getCid() {

**return** cid;

}

**public** **void** setCid(**int** cid) {

**this**.cid = cid;

}

**public** String getCmobile() {

**return** cmobile;

}

**public** **void** setCmobile(String cmobile) {

**this**.cmobile = cmobile;

}

}

**HibernateConfigurationFile: hibernate.cfg.xml**

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

<!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>

<property name=*"hibernate.connection.driver\_class"*>oracle.jdbc.OracleDriver</property>

<property name=*"hibernate.connection.url"*>jdbc:oracle:thin:@localhost:1521:xe</property>

<property name=*"hibernate.connection.username"*>system</property>

<property name=*"hibernate.connection.password"*>tiger</property>

<property name=*"hibernate.dialect"*>org.hibernate.dialect.OracleDialect</property>

**<mapping class=*"com.hbi.domain.Person"*/>**

**<mapping class=*"com.hbi.domain.Employee"*/>**

**<mapping class=*"com.hbi.domain.Customer"*/>**

</session-factory>

</hibernate-configuration>

**ClientApplication:Test.java**

package com.hbi.domain;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

import org.hibernate.cfg.Configuration;

import org.hibernate.classic.Session;

public class Test {

public static void main(String[] args) {

Configuration cfg=new Configuration();

cfg.configure("com/hbi/domain/hibernate.cfg.xml");

SessionFactory sf=cfg.buildSessionFactory();

Session s=sf.openSession();

Transaction t=s.beginTransaction();

Employee e=new Employee();

Customer c=new Customer();

e.setEid(1);

e.setEsal(7000);

e.setPaddr("nlr");

e.setPname("suku");

c.setCid(8);

c.setCmobile("7989");

c.setPname("sv");

c.setPaddr("chi");

s.save(e);

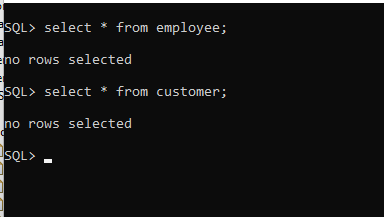
s.save(c);

t.commit();

}

}

Output:-



After program execution completed

