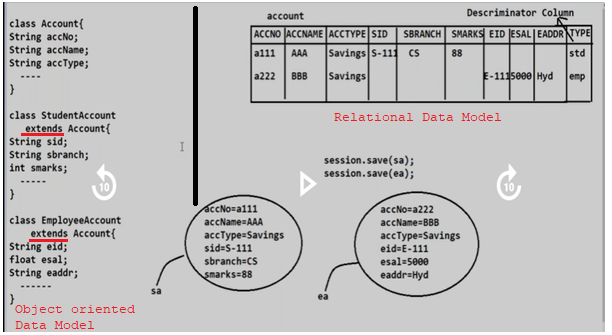
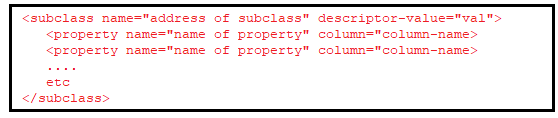
**1.Table per Hierarchy:-** we can map the whole hierarchy by single table only. Here, an extra column (also known as **discriminator column**) is created in the table to know the kind of record.



**1.1.Mapping between Inheritances classes and relation using XML tags:**

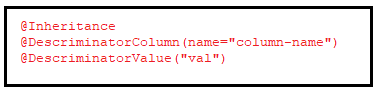
**1.<subclass>:-** It is paired tag. It should be written in <class> tag. It is child tag to <class> tag. We map only the subclass properties with columns of table.



2.<descriptor>:- It is single-tone tag. It should be written in <class> tag. It is used to map descriptor column of table.



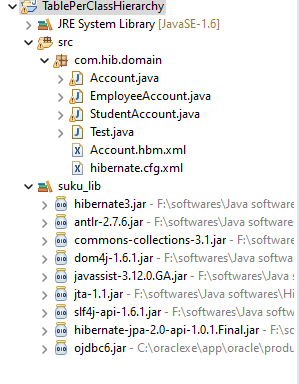
**1.21.1.Mapping between Inheritances classes and relation using annotated classes:**



The first-two annotations are written just above the parent class.

The last-annotation is written just above the parent class.

Example1:- using Mapping file.



**Domain classes:**

**1.Account.java**

**package** com.hib.domain;

**import** java.io.Serializable;

**public** **class** Account **implements** Serializable{

**private** **int** accNo;

**private** String accName;

**private** String accType;

**public** **int** getAccNo() {

**return** accNo;

}

**public** **void** setAccNo(**int** accNo) {

**this**.accNo = accNo;

}

**public** String getAccName() {

**return** accName;

}

**public** **void** setAccName(String accName) {

**this**.accName = accName;

}

**public** String getAccType() {

**return** accType;

}

**public** **void** setAccType(String accType) {

**this**.accType = accType;

}

}

**2.StudentAccount.java**

**package** com.hib.domain;

**public** **class** StudentAccount **extends** Account {

**private** **int** sid;

**private** String sname;

**private** **float** smarks;

**public** **int** getSid() {

**return** sid;

}

**public** **void** setSid(**int** sid) {

**this**.sid = sid;

}

**public** String getSname() {

**return** sname;

}

**public** **void** setSname(String sname) {

**this**.sname = sname;

}

**public** **float** getSmarks() {

**return** smarks;

}

**public** **void** setSmarks(**float** smarks) {

**this**.smarks = smarks;

}

}

**3.EmployeeAccount.java**

**package** com.hib.domain;

**public** **class** EmployeeAccount **extends** Account {

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

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

**private** String eaddr;

**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;

}

**public** String getEaddr() {

**return** eaddr;

}

**public** **void** setEaddr(String eaddr) {

**this**.eaddr = eaddr;

}

}

**MappingFile:Account.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.hib.domain.Account"* table=*"account"*>

<id name=*"accNo"* column=*'accno'*/>

<discriminator column=*"Type"* type=*"string"*/>

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

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

<subclass name=*"com.hib.domain.StudentAccount"* discriminator-value=*"stu"*>

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

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

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

</subclass>

<subclass name=*"com.hib.domain.EmployeeAccount"* discriminator-value=*"emp"*>

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

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

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

</subclass>

</class>

</hibernate-mapping>

**Hibernate configuration file: 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/hib/domain/Account.hbm.xml"*/>

</session-factory>

</hibernate-configuration>

**Clientapplication:Test.java**

**package** com.hib.domain;

**import** org.hibernate.Session;

**import** org.hibernate.SessionFactory;

**import** org.hibernate.Transaction;

**import** org.hibernate.cfg.Configuration;

**public** **class** Test {

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

Configuration cfg=**new** Configuration();

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

SessionFactory sf= cfg.buildSessionFactory();

Session s=sf.openSession();

Transaction t=s.beginTransaction();

StudentAccount sa=**new** StudentAccount();

EmployeeAccount ea=**new** EmployeeAccount();

ea.setAccNo(8);

ea.setAccName("su1");

ea.setAccType("current1");

ea.setEid(9);

ea.setEsal(23.3f);

ea.setEaddr("ong");

sa.setAccNo(7);

sa.setAccName("su");

sa.setAccType("savings");

sa.setSid(1);

sa.setSname("suku");

sa.setSmarks(777f);

s.save(ea);

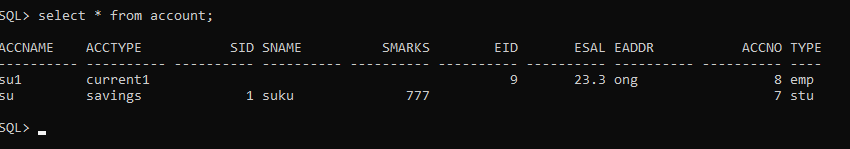
s.save(sa);

t.commit();

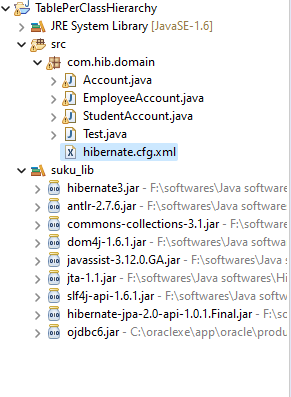
}

}

**Output:**

****

**Example:2 Using Annotated Class.**

****

**Domain /Annotated Classes:-**

1. **Account.java**
2. **package** com.hib.domain;
3. **import** java.io.Serializable;
4. **import** javax.persistence.Column;
5. **import** javax.persistence.DiscriminatorColumn;
6. **import** javax.persistence.Entity;
7. **import** javax.persistence.Id;
8. **import** javax.persistence.Inheritance;
9. **import** javax.persistence.Table;
10. @Inheritance
11. @DiscriminatorColumn(name="Type")
12. @Entity
13. @Table(name="account")
14. **public** **class** Account **implements** Serializable{
15. @Id
16. @Column(name="accNo")
17. **private** **int** accNo;
18. @Column(name="accName")
19. **private** String accName;
20. @Column(name="accType")
21. **private** String accType;
22. **public** **int** getAccNo() {
23. **return** accNo;
24. }
25. **public** **void** setAccNo(**int** accNo) {
26. **this**.accNo = accNo;
27. }
28. **public** String getAccName() {
29. **return** accName;
30. }
31. **public** **void** setAccName(String accName) {
32. **this**.accName = accName;
33. }
34. **public** String getAccType() {
35. **return** accType;
36. }
37. **public** **void** setAccType(String accType) {
38. **this**.accType = accType;
39. }
41. }

**StudentAccount.java**

**package** com.hib.domain;

**import** javax.persistence.Column;

**import** javax.persistence.DiscriminatorValue;

**import** javax.persistence.Entity;

@Entity

@DiscriminatorValue("stu")

**public** **class** StudentAccount **extends** Account {

@Column(name="sid")

**private** **int** sid;

@Column(name="sname")

**private** String sname;

@Column(name="smarks")

**private** **float** smarks;

**public** **int** getSid() {

**return** sid;

}

**public** **void** setSid(**int** sid) {

**this**.sid = sid;

}

**public** String getSname() {

**return** sname;

}

**public** **void** setSname(String sname) {

**this**.sname = sname;

}

**public** **float** getSmarks() {

**return** smarks;

}

**public** **void** setSmarks(**float** smarks) {

**this**.smarks = smarks;

}

}

**EmployeeAccount.java**

**package** com.hib.domain;

**import** javax.persistence.Column;

**import** javax.persistence.DiscriminatorValue;

**import** javax.persistence.Entity;

@Entity

@DiscriminatorValue("emp")

**public** **class** EmployeeAccount **extends** Account {

@Column(name="eid")

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

@Column(name="esal")

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

@Column(name="eaddr")

**private** String eaddr;

**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;

}

**public** String getEaddr() {

**return** eaddr;

}

**public** **void** setEaddr(String eaddr) {

**this**.eaddr = eaddr;

}

}

**3.Hibernate Configuration file: 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.hib.domain.StudentAccount"*/>**

**<mapping class=*"com.hib.domain.EmployeeAccount"*/>**

</session-factory>

</hibernate-configuration>

**4.ClientApplication:Test.java**

package com.hib.domain;

import org.hibernate.Session;

import org.hibernate.SessionFactory;

import org.hibernate.Transaction;

import org.hibernate.cfg.Configuration;

public class Test {

public static void main(String[] args) {

Configuration cfg=new Configuration();

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

SessionFactory sf= cfg.buildSessionFactory();

Session s=sf.openSession();

Transaction t=s.beginTransaction();

StudentAccount sa=new StudentAccount();

EmployeeAccount ea=new EmployeeAccount();

ea.setAccNo(8);

ea.setAccName("su1");

ea.setAccType("current1");

ea.setEid(9);

ea.setEsal(23.3f);

ea.setEaddr("ong");

sa.setAccNo(7);

sa.setAccName("su");

sa.setAccType("savings");

sa.setSid(1);

sa.setSname("suku");

sa.setSmarks(777f);

s.save(ea);

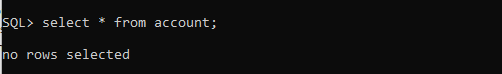
s.save(sa);

t.commit();

}

}

Output:-



After Execution

