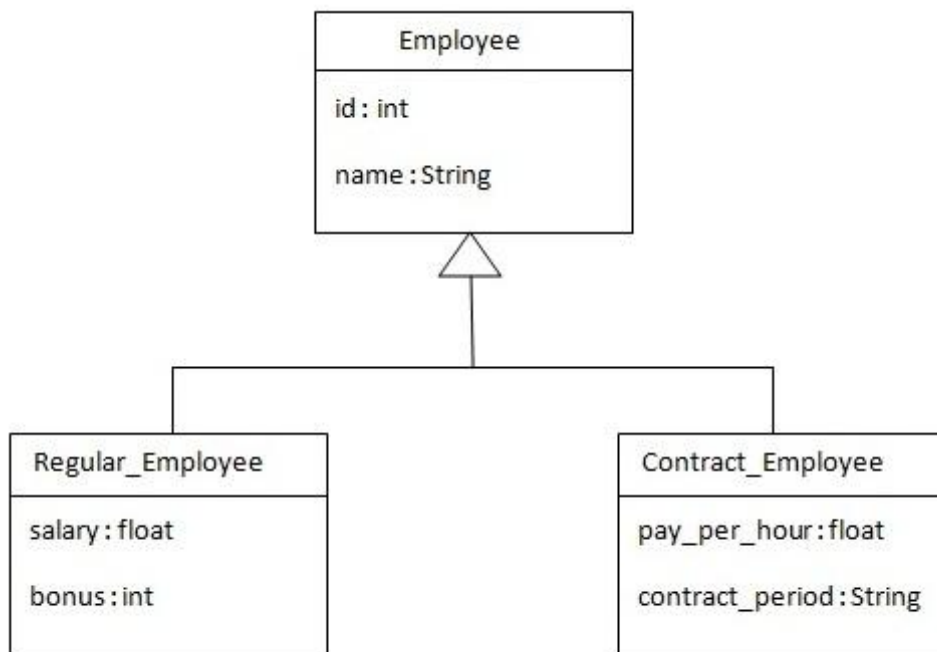


Table Per Subclass using Annotation

As we have specified earlier, in case of table per subclass strategy, tables are created as per persistent classes but they are related using primary and foreign key. So there will not be duplicate columns in the relation.

We need to specify **@Inheritance(strategy=InheritanceType.JOINED)** in the parent class and **@PrimaryKeyJoinColumn** annotation in the subclasses.

Let's see the hierarchy of classes that we are going to map.



The table structure for each table will be as follows:

Table structure for Employee class

Column Name	Data Type	Nullable	Default	Primary Key
ID	NUMBER(10,0)	No	-	1
NAME	VARCHAR2(255)	Yes	-	-
				1 - 2

Table structure for Regular_Employee class

Column Name	Data Type	Nullable	Default	Primary Key
EID	NUMBER(10,0)	No	-	1
SALARY	FLOAT	Yes	-	-
BONUS	NUMBER(10,0)	Yes	-	-
				1 - 3

Table structure for Contract_Employee class

Column Name	Data Type	Nullable	Default	Primary Key
EID	NUMBER(10,0)	No	-	1
PAY_PER_HOUR	FLOAT	Yes	-	-
CONTRACT_DURATION	VARCHAR2(255)	Yes	-	-
				1 - 3

Example of Table per subclass class using Annotation

In this example we are creating the three classes and provide mapping of these classes in the employee.hbm.xml file.

1) Create the Persistent classes

You need to create the persistent classes representing the inheritance. Let's create the three classes for the above hierarchy:

File: Employee.java

```
package com.javatpoint.mypackage;

import javax.persistence.*;

@Entity
@Table(name = "employee103")
@Inheritance(strategy=InheritanceType.JOINED)

public class Employee {

    @Id
    @GeneratedValue(strategy=GenerationType.AUTO)

    @Column(name = "id")
```

```
private int id;

@Column(name = "name")
private String name;

//setters and getters
}
```

File: Regular_Employee.java

```
package com.javatpoint.mypackage;

import javax.persistence.*;

@Entity
@Table(name="regularemployee103")
@PrimaryKeyJoinColumn(name="ID")
public class Regular_Employee extends Employee{

    @Column(name="salary")
    private float salary;

    @Column(name="bonus")
    private int bonus;

    //setters and getters
}
```

File: Contract_Employee.java

```
package com.javatpoint.mypackage;

import javax.persistence.*;

@Entity
@Table(name="contractemployee103")
@PrimaryKeyJoinColumn(name="ID")
public class Contract_Employee extends Employee{
```

```
@Column(name="pay_per_hour")
private float pay_per_hour;

@Column(name="contract_duration")
private String contract_duration;

//setters and getters
}
```

2) create configuration file

Open the hibernate.cfg.xml file, and add an entry of mapping resource like this:

```
<mapping class="com.javatpoint.mypackage.Employee"/>
<mapping class="com.javatpoint.mypackage.Contract_Employee"/>
<mapping class="com.javatpoint.mypackage.Regular_Employee"/>
```

Now the configuration file will look like this:

File: 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">

<!-- Generated by MyEclipse Hibernate Tools.           -->
<hibernate-configuration>

    <session-factory>
        <property name="hbm2ddl.auto">update</property>
        <property name="dialect">org.hibernate.dialect.Oracle9Dialect</property>
        <property name="connection.url">jdbc:oracle:thin:@localhost:1521:xe</property>
        <property name="connection.username">system</property>
        <property name="connection.password">oracle</property>
        <property name="connection.driver_class">oracle.jdbc.driver.OracleDriver</property>
```

```
<mapping class="com.javatpoint.mypackage.Employee"/>
<mapping class="com.javatpoint.mypackage.Contract_Employee"/>
<mapping class="com.javatpoint.mypackage.Regular_Employee"/>
</session-factory>

</hibernate-configuration>
```

The hbm2ddl.auto property is defined for creating automatic table in the database.

3) Create the class that stores the persistent object

In this class, we are simply storing the employee objects in the database.

File: StoreData.java

```
package com.javatpoint.mypackage;
import org.hibernate.*;
import org.hibernate.cfg.*;

public class StoreData {
    public static void main(String[] args) {
        AnnotationConfiguration cfg=new AnnotationConfiguration();
        Session session=cfg.configure("hibernate.cfg.xml").buildSessionFactory().openSession();

        Transaction t=session.beginTransaction();

        Employee e1=new Employee();
        e1.setName("sonoo");

        Regular_Employee e2=new Regular_Employee();
        e2.setName("Vivek Kumar");
        e2.setSalary(50000);
        e2.setBonus(5);

        Contract_Employee e3=new Contract_Employee();
        e3.setName("Arjun Kumar");
        e3.setPay_per_hour(1000);
        e3.setContract_duration("15 hours");
```

```
session.persist(e1);  
session.persist(e2);  
session.persist(e3);  
  
t.commit();  
session.close();  
System.out.println("success");  
}  
}
```

download this example developed using Myeclipse IDE

[← prev](#)[next →](#)

Share  11