

Date:14-08-2022

**COURSE ID: 20TS3103AA**

**COURSE NAME: JAVA FULL STACK DEVELOPMENT + MICROSERVICES**

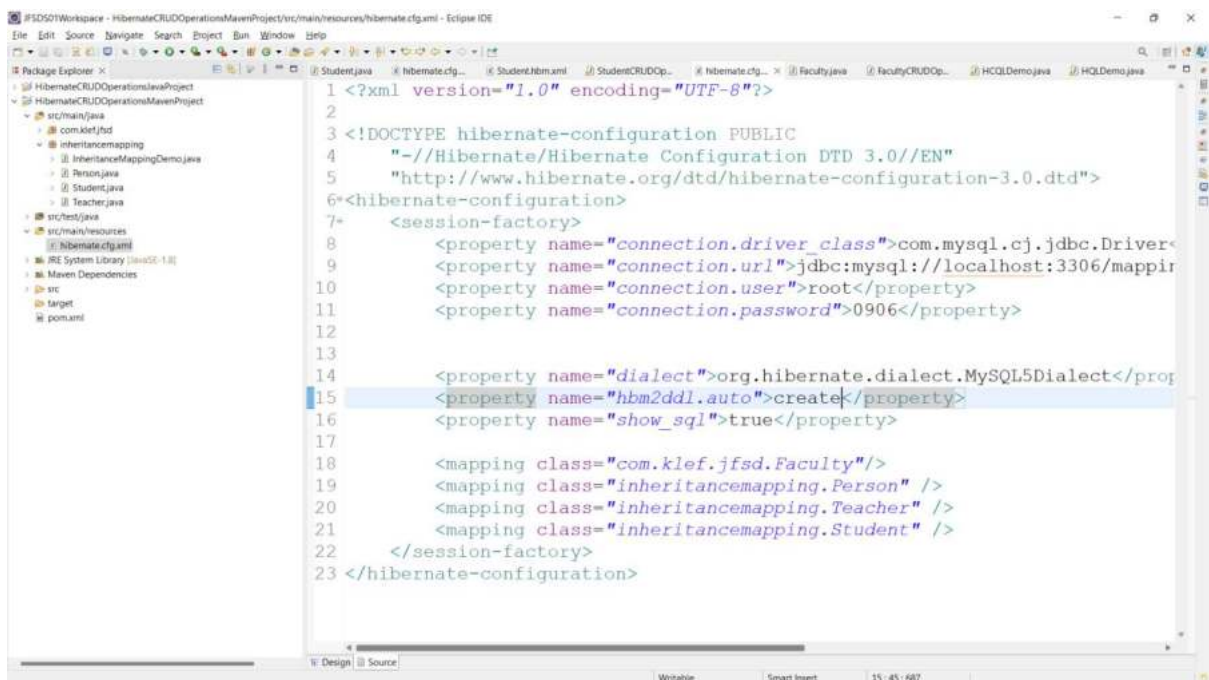
**LAB:04**

ID:2000030185

NAME: SUPRITH CH

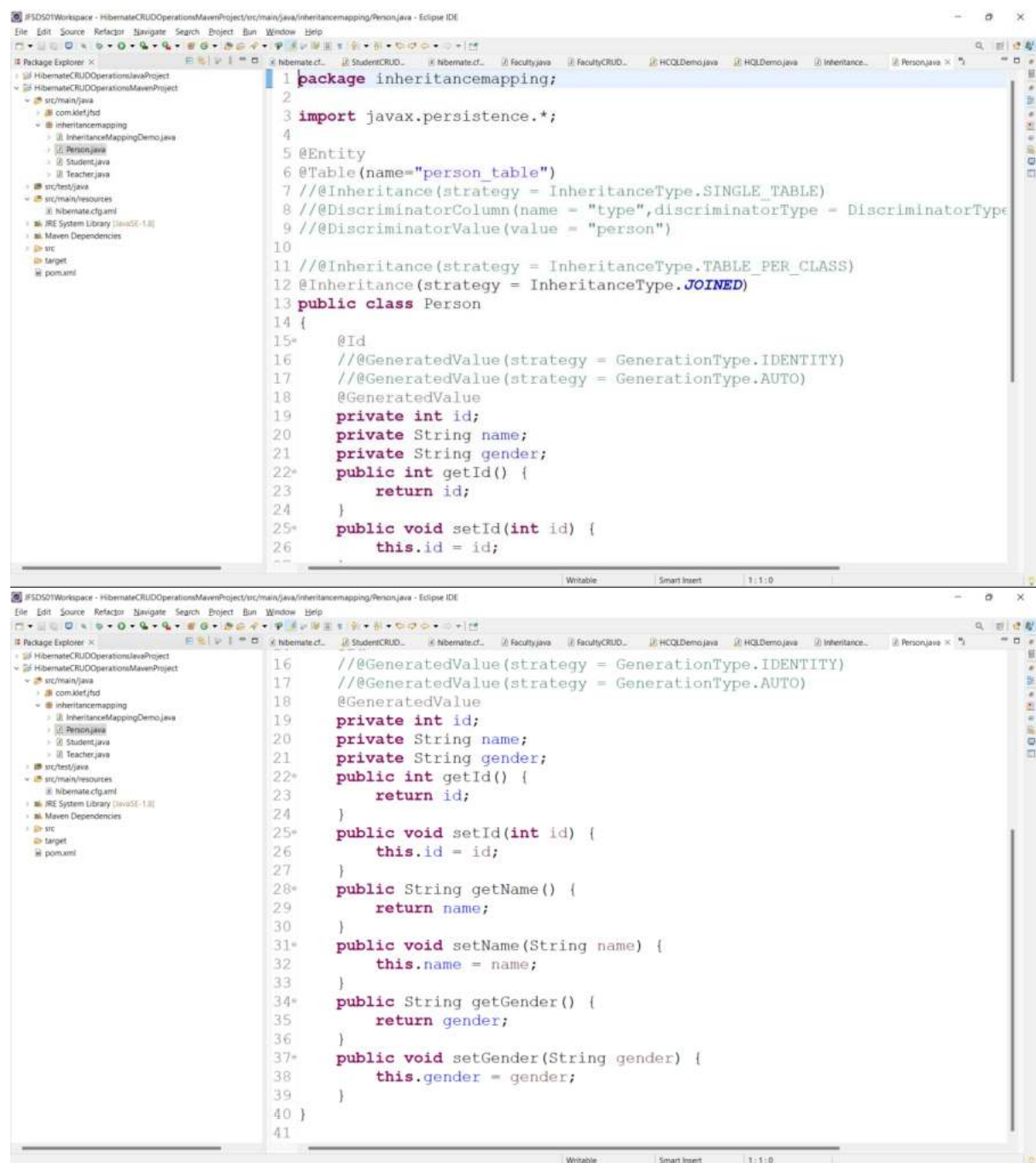
- ▼ HibernateCRUDOperationsMavenProject
  - ▼ src/main/java
    - ▼ com.klef.jfsd
      - ▼ inheritancemapping
        - ▼ InheritanceMappingDemo.java
        - ▼ Person.java
        - ▼ Student.java
        - ▼ Teacher.java
    - ▼ src/test/java
    - ▼ src/main/resources
      - ▼ hibernate.cfg.xml
    - ▼ JRE System Library [JavaSE-1.8]
    - ▼ Maven Dependencies
    - ▼ src
    - ▼ target
    - ▼ pom.xml

Hibernate.cfg.xml:



```
1 <?xml version="1.0" encoding="UTF-8"?>
2
3 <!DOCTYPE hibernate-configuration PUBLIC
4     "-//Hibernate/Hibernate Configuration DTD 3.0//EN"
5     "http://www.hibernate.org/dtd/hibernate-configuration-3.0.dtd">
6 <hibernate-configuration>
7     <session-factory>
8         <property name="connection.driver_class">com.mysql.cj.jdbc.Driver</property>
9         <property name="connection.url">jdbc:mysql://localhost:3306/mapping</property>
10        <property name="connection.user">root</property>
11        <property name="connection.password">0906</property>
12
13        <property name="dialect">org.hibernate.dialect.MySQL5Dialect</property>
14        <property name="hbm2ddl.auto">create</property>
15        <property name="show_sql">true</property>
16
17        <mapping class="com.klef.jfsd.Faculty"/>
18        <mapping class="inheritancemapping.Person"/>
19        <mapping class="inheritancemapping.Teacher"/>
20        <mapping class="inheritancemapping.Student"/>
21    </session-factory>
22 </hibernate-configuration>
23
```

## Person.java



```
1 package inheritanceMapping;
2
3 import javax.persistence.*;
4
5 @Entity
6 @Table(name="person_table")
7 // @Inheritance(strategy = InheritanceType.SINGLE_TABLE)
8 // @DiscriminatorColumn(name = "type", discriminatorType = DiscriminatorType.STRING)
9 // @DiscriminatorValue(value = "person")
10
11 // @Inheritance(strategy = InheritanceType.TABLE_PER_CLASS)
12 @Inheritance(strategy = InheritanceType.JOINED)
13 public class Person
14 {
15     @Id
16     // @GeneratedValue(strategy = GenerationType.IDENTITY)
17     // @GeneratedValue(strategy = GenerationType.AUTO)
18     @GeneratedValue
19     private int id;
20     private String name;
21     private String gender;
22     public int getId() {
23         return id;
24     }
25     public void setId(int id) {
26         this.id = id;
27     }
28     public String getName() {
29         return name;
30     }
31     public void setName(String name) {
32         this.name = name;
33     }
34     public String getGender() {
35         return gender;
36     }
37     public void setGender(String gender) {
38         this.gender = gender;
39     }
40 }
41
```

## Student.java

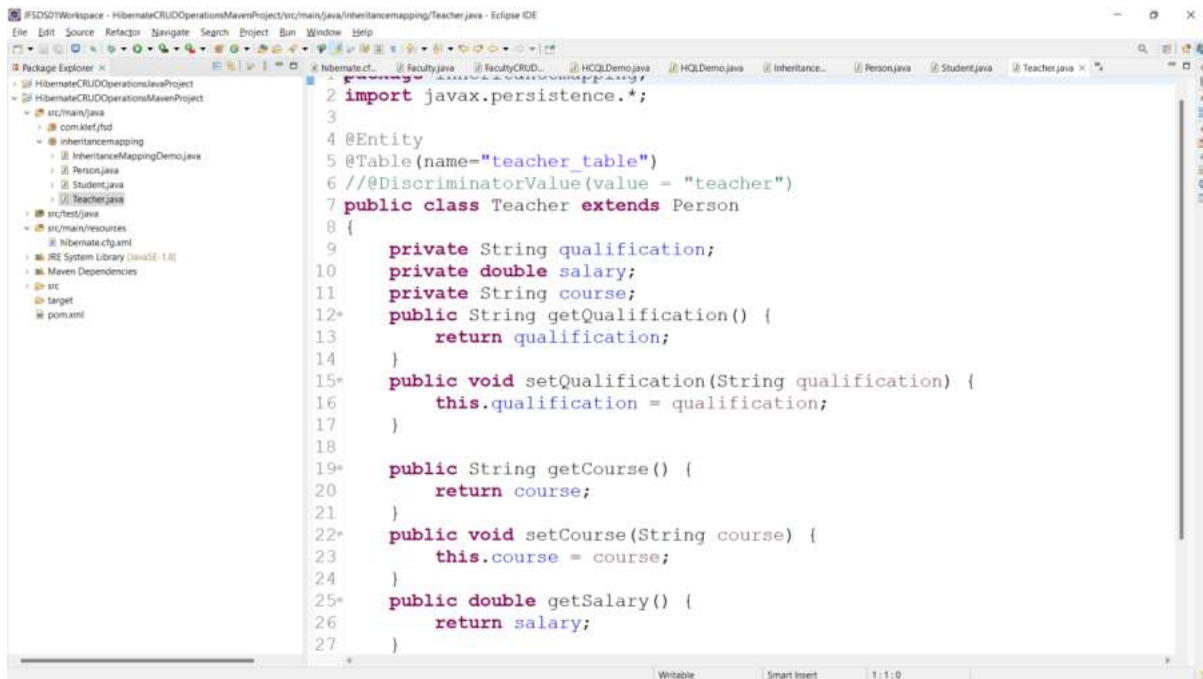
```
#SDS01Workspace - HibernateCRUDOperationsMavenProject/src/main/java/inheritancemapping/Student.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
  HibernateCRUDOperationsMavenProject
    src/main/java
      com.kiet.hjd
        inheritancemapping
          InheritanceMappingDemo.java
          Person.java
          Student.java
          Teacher.java
    src/test/java
    src/main/resources
      hibernate.cfg.xml
  JRE System Library [JavaSE-1.8]
  Maven Dependencies
  src
  target
  pom.xml

1 package inheritancemapping;
2 import javax.persistence.*;
3
4 @Entity
5 @Table(name="student_table")
6 // @DiscriminatorValue(value = "student")
7 public class Student extends Person
8 {
9     private String program;
10    private double cgpa;
11    private int backlogs;
12    public String getProgram() {
13        return program;
14    }
15    public void setProgram(String program) {
16        this.program = program;
17    }
18    public double getCgpa() {
19        return cgpa;
20    }
21    public void setCgpa(double cgpa) {
22        this.cgpa = cgpa;
23    }
24    public int getBacklogs() {
25        return backlogs;
26    }
27 }
```

```
#SDS01Workspace - HibernateCRUDOperationsMavenProject/src/main/java/inheritancemapping/Student.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
  HibernateCRUDOperationsMavenProject
    src/main/java
      com.kiet.hjd
        inheritancemapping
          InheritanceMappingDemo.java
          Person.java
          Student.java
          Teacher.java
    src/test/java
    src/main/resources
      hibernate.cfg.xml
  JRE System Library [JavaSE-1.8]
  Maven Dependencies
  src
  target
  pom.xml

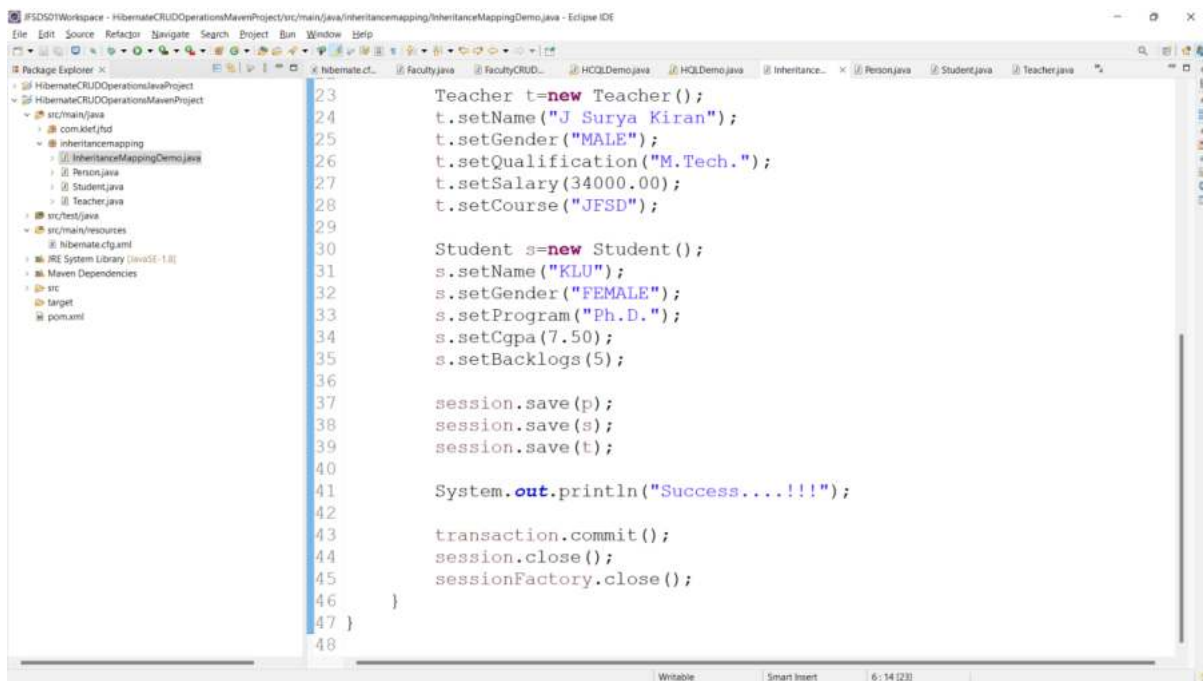
6 // @DiscriminatorValue(value = "student")
7 public class Student extends Person
8 {
9     private String program;
10    private double cgpa;
11    private int backlogs;
12    public String getProgram() {
13        return program;
14    }
15    public void setProgram(String program) {
16        this.program = program;
17    }
18    public double getCgpa() {
19        return cgpa;
20    }
21    public void setCgpa(double cgpa) {
22        this.cgpa = cgpa;
23    }
24    public int getBacklogs() {
25        return backlogs;
26    }
27    public void setBacklogs(int backlogs) {
28        this.backlogs = backlogs;
29    }
30 }
31 }
```

## Teacher.java



The screenshot shows the Eclipse IDE with the 'Teacher.java' file open. The Package Explorer on the left shows the project structure: 'hibernateCRUDOperationsMavenProject' with sub-packages 'com.kiet.fhd' and 'inheritancemapping'. The 'inheritancemapping' package contains 'Person.java', 'Student.java', and 'Teacher.java'. The main editor displays the code for 'Teacher.java', which is a JPA entity extending 'Person'. It has attributes 'qualification' (String), 'salary' (double), and 'course' (String), each with corresponding getter and setter methods. The entity is mapped to the 'teacher\_table' with a discriminator value of 'teacher'.

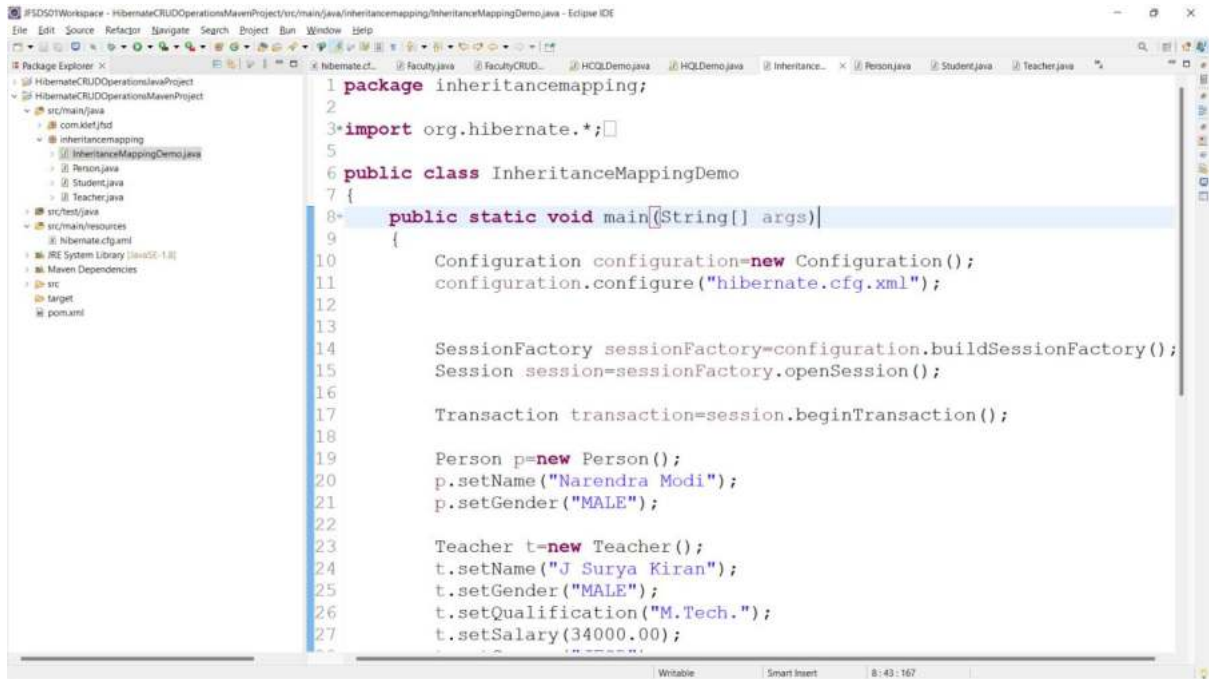
```
1 import javax.persistence.*;
2
3
4 @Entity
5 @Table(name="teacher_table")
6 //@DiscriminatorValue(value = "teacher")
7 public class Teacher extends Person
8 {
9     private String qualification;
10    private double salary;
11    private String course;
12    public String getQualification() {
13        return qualification;
14    }
15    public void setQualification(String qualification) {
16        this.qualification = qualification;
17    }
18
19    public String getCourse() {
20        return course;
21    }
22    public void setCourse(String course) {
23        this.course = course;
24    }
25    public double getSalary() {
26        return salary;
27    }
28 }
```



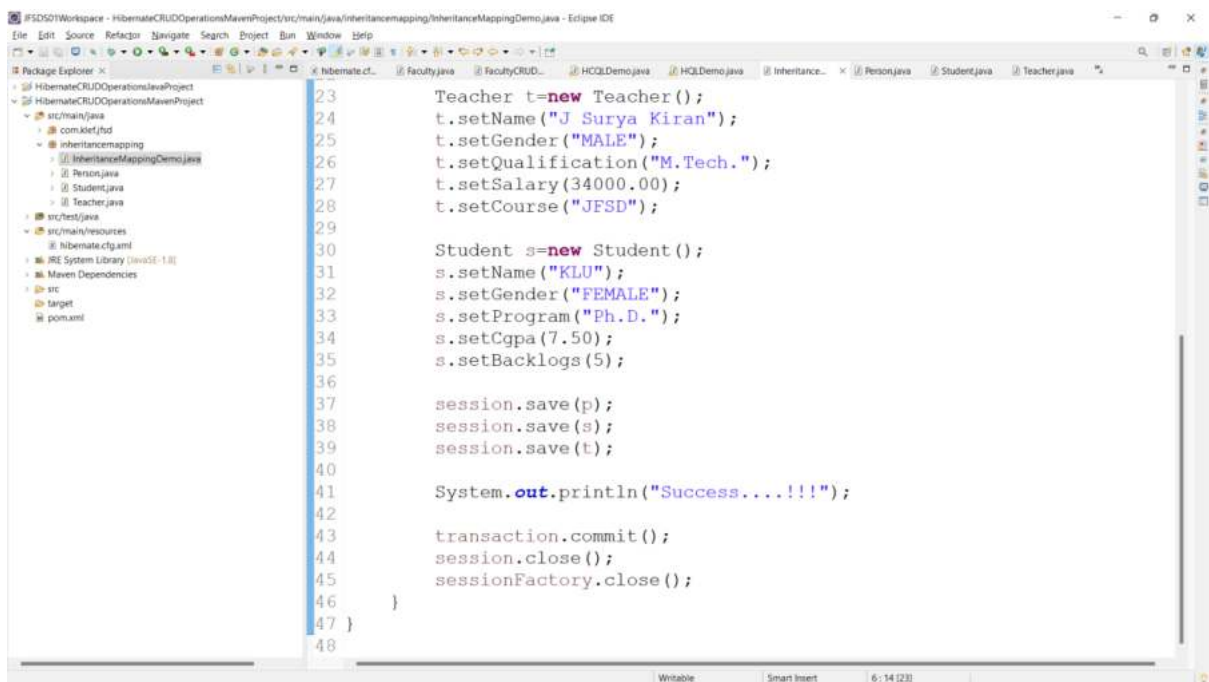
The screenshot shows the Eclipse IDE with the 'InheritanceMappingDemo.java' file open. The Package Explorer on the left shows the project structure: 'hibernateCRUDOperationsMavenProject' with sub-packages 'com.kiet.fhd' and 'inheritancemapping'. The 'inheritancemapping' package contains 'Person.java', 'Student.java', and 'Teacher.java'. The main editor displays the code for 'InheritanceMappingDemo.java', which creates instances of 'Teacher' and 'Student', sets their attributes, saves them to the database, and prints a success message.

```
23 Teacher t=new Teacher();
24 t.setName("J Surya Kiran");
25 t.setGender("MALE");
26 t.setQualification("M.Tech.");
27 t.setSalary(34000.00);
28 t.setCourse("JFSD");
29
30 Student s=new Student();
31 s.setName("KLU");
32 s.setGender("FEMALE");
33 s.setProgram("Ph.D.");
34 s.setCgpa(7.50);
35 s.setBacklogs(5);
36
37 session.save(p);
38 session.save(s);
39 session.save(t);
40
41 System.out.println("Success....!!!");
42
43 transaction.commit();
44 session.close();
45 sessionFactory.close();
46 }
47 }
48 }
```

## InheritanceMappingDemo.java



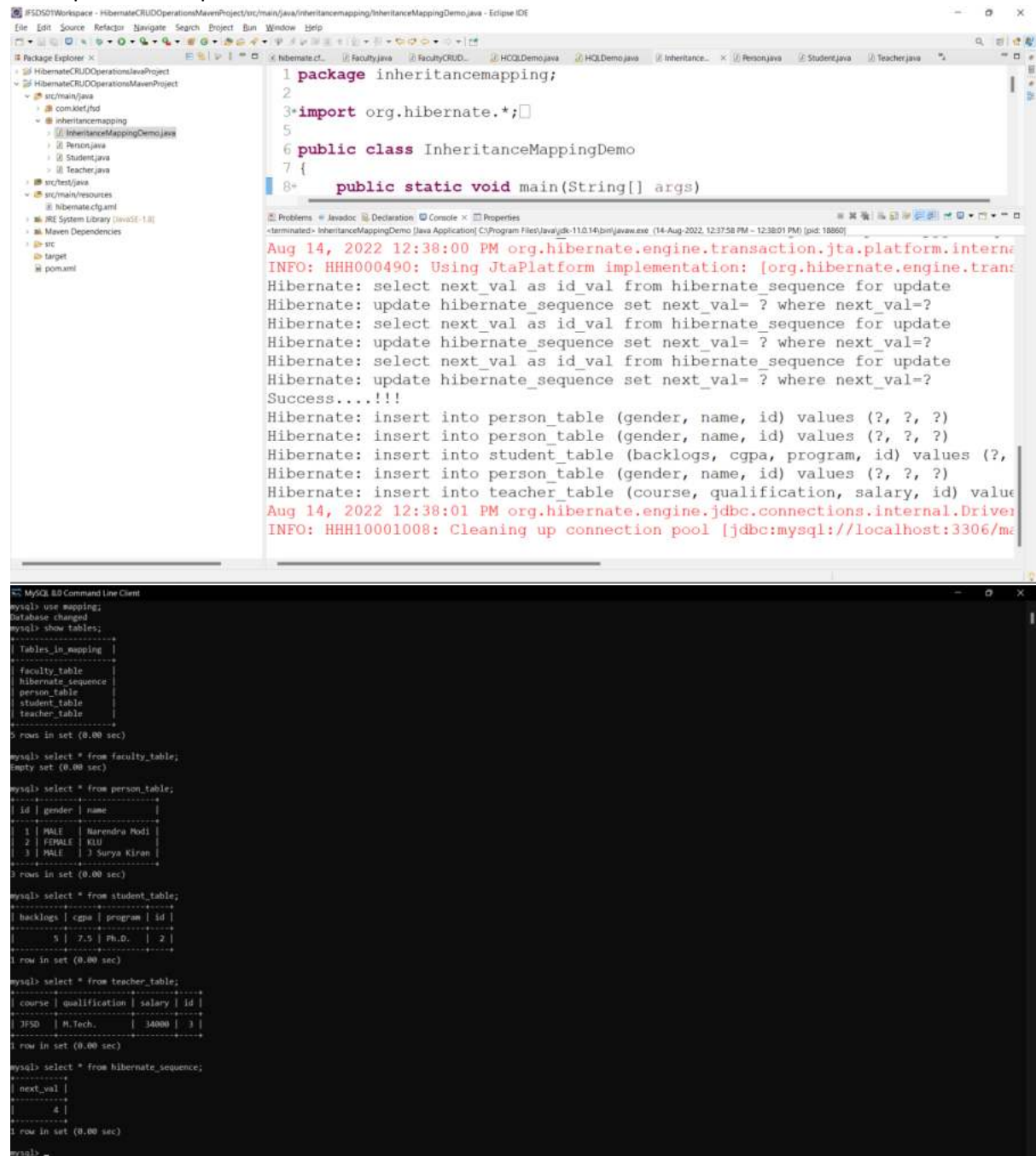
```
1 package inheritanceMapping;  
2  
3 import org.hibernate.*;   
4  
5  
6 public class InheritanceMappingDemo  
7 {  
8     public static void main(String[] args)  
9     {  
10         Configuration configuration=new Configuration();  
11         configuration.configure("hibernate.cfg.xml");  
12  
13  
14         SessionFactory sessionFactory=configuration.buildSessionFactory();  
15         Session session=sessionFactory.openSession();  
16  
17         Transaction transaction=session.beginTransaction();  
18  
19         Person p=new Person();  
20         p.setName("Narendra Modi");  
21         p.setGender("MALE");  
22  
23         Teacher t=new Teacher();  
24         t.setName("J Surya Kiran");  
25         t.setGender("MALE");  
26         t.setQualification("M.Tech.");  
27         t.setSalary(34000.00);  
28     }  
29 }
```



```
23         Teacher t=new Teacher();  
24         t.setName("J Surya Kiran");  
25         t.setGender("MALE");  
26         t.setQualification("M.Tech.");  
27         t.setSalary(34000.00);  
28         t.setCourse("JFSD");  
29  
30         Student s=new Student();  
31         s.setName("KLU");  
32         s.setGender("FEMALE");  
33         s.setProgram("Ph.D.");  
34         s.setCgpa(7.50);  
35         s.setBacklogs(5);  
36  
37         session.save(p);  
38         session.save(s);  
39         session.save(t);  
40  
41         System.out.println("Success...!!!");  
42  
43         transaction.commit();  
44         session.close();  
45         sessionFactory.close();  
46     }  
47 }  
48 }
```



## Join Operation output:



The screenshot displays the Eclipse IDE environment. The Package Explorer on the left shows the project structure: `hibernateCRUDOperationsMavenProject` with sub-packages `src/main/java` and `src/test/java`. The `src/main/java` package contains `inheritancemapping`, which includes `InheritanceMappingDemo.java`, `Person.java`, `Student.java`, and `Teacher.java`. The `src/test/java` package contains `hibernate.ct`, which includes `hibernate.ctg.xml`. The `src` directory contains `target` and `pom.xml`.

The main editor shows the `InheritanceMappingDemo.java` file with the following code:

```
1 package inheritancemapping;
2
3 import org.hibernate.*;
4
5
6 public class InheritanceMappingDemo
7 {
8     public static void main(String[] args)
```

The console output shows the following log messages:

```
Aug 14, 2022 12:38:00 PM org.hibernate.engine.transaction.jta.platform.internal
INFO: HHH0000490: Using JtaPlatform implementation: [org.hibernate.engine.trans
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Success....!!!
Hibernate: insert into person_table (gender, name, id) values (?, ?, ?)
Hibernate: insert into person_table (gender, name, id) values (?, ?, ?)
Hibernate: insert into student_table (backlogs, cgpa, program, id) values (?,
Hibernate: insert into person_table (gender, name, id) values (?, ?, ?)
Hibernate: insert into teacher_table (course, qualification, salary, id) value
Aug 14, 2022 12:38:01 PM org.hibernate.engine.jdbc.connections.internal.Driver
INFO: HHH10001008: Cleaning up connection pool [jdbc:mysql://localhost:3306/m
```

The MySQL Command Line Client window shows the following output:

```
mysql> use mapping;
Database changed
mysql> show tables;
+-----+
| Tables_in_mapping |
+-----+
| faculty_table      |
| hibernate_sequence |
| person_table       |
| student_table      |
| teacher_table      |
+-----+
5 rows in set (0.00 sec)

mysql> select * from faculty_table;
Empty set (0.00 sec)

mysql> select * from person_table;
+----+-----+-----+
| id | gender | name      |
+----+-----+-----+
| 1  | MALE   | Narendra Modi |
| 2  | FEMALE | KLU      |
| 3  | MALE   | J Surya Kiran |
+----+-----+-----+
3 rows in set (0.00 sec)

mysql> select * from student_table;
+-----+-----+-----+-----+
| backlogs | cgpa | program | id |
+-----+-----+-----+-----+
| 5        | 7.5  | Ph.D.   | 2  |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select * from teacher_table;
+-----+-----+-----+-----+
| course | qualification | salary | id |
+-----+-----+-----+-----+
| JFSD   | M.Tech.       | 34000  | 3  |
+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select * from hibernate_sequence;
+-----+
| next_val |
+-----+
| 4        |
+-----+
1 row in set (0.00 sec)

mysql>
```

## TABLE\_PER\_CLASS Strategy:

The screenshot shows the Eclipse IDE with the `InheritanceMappingDemo.java` file open. The code in the editor is:

```
31 s.setName("KLU");
```

The console output shows the following log messages:

```
<terminated> InheritanceMappingDemo [Java Application] C:\Program Files\Java\jdk-11.0.14\bin\java.exe (14-Aug-2022, 12:42:18 PM - 12:42:20 PM) [pid: 15012]
Aug 14, 2022 12:42:20 PM org.hibernate.resource.transaction.backend.jdbc.internal.JdbcConnectionAccess [org.hibernate.resource.transaction.backend.jdbc.internal.JdbcConnectionAccess]
INFO: HHH10001501: Connection obtained from JdbcConnectionAccess [org.hibernate.resource.transaction.backend.jdbc.internal.JdbcConnectionAccess]
Hibernate: create table hibernate_sequence (next_val bigint) engine=MyISAM
Hibernate: insert into hibernate_sequence values ( 1 )
Hibernate: create table person_table (id integer not null, gender varchar(255)) engine=MyISAM
Hibernate: create table student_table (id integer not null, gender varchar(255)) engine=MyISAM
Hibernate: create table teacher_table (id integer not null, gender varchar(255)) engine=MyISAM
Hibernate: alter table faculty_table add constraint UK_trveb37dbf50hs8gckoj8c
Aug 14, 2022 12:42:20 PM org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformImpl [org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformImpl]
INFO: HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformImpl]
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Success.....!!!
Hibernate: insert into person_table (gender, name, id) values (?, ?, ?)
Hibernate: insert into student_table (gender, name, backlogs, cgpa, program, salary) values (?, ?, ?, ?, ?, ?)
Hibernate: insert into teacher_table (gender, name, course, qualification, salary) values (?, ?, ?, ?, ?)
Aug 14, 2022 12:42:20 PM org.hibernate.engine.jdbc.connections.internal.DriverBasedConnectionProvider [org.hibernate.engine.jdbc.connections.internal.DriverBasedConnectionProvider]
INFO: HHH10001008: Cleaning up connection pool [jdbc:mysql://localhost:3306/maven]
```

Output:

## SINGLE TABLE STRATEGY

The screenshot shows the Eclipse IDE with the `InheritanceMappingDemo.java` file open. The code in the editor is:

```
1 package inheritanceMapping;
2
3 import org.hibernate.*;
4
5
6 public class InheritanceMappingDemo
```

The console output shows the following log messages:

```
<terminated> InheritanceMappingDemo [Java Application] C:\Program Files\Java\jdk-11.0.14\bin\java.exe (14-Aug-2022, 2:27:06 PM - 2:27:08 PM) [pid: 8460]
Hibernate: create table hibernate_sequence (next_val bigint) engine=MyISAM
Hibernate: insert into hibernate_sequence values ( 1 )
Hibernate: create table person_table (type varchar(100) not null, id integer not null) engine=MyISAM
Hibernate: alter table faculty_table add constraint UK_trveb37dbf50hs8gckoj8c
Aug 14, 2022 2:27:08 PM org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformImpl [org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformImpl]
INFO: HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformImpl]
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Hibernate: select next_val as id_val from hibernate_sequence for update
Hibernate: update hibernate_sequence set next_val= ? where next_val=?
Success.....!!!
Hibernate: insert into person_table (gender, name, type, id) values (?, ?, 'person', ?)
Hibernate: insert into person_table (gender, name, backlogs, cgpa, program, type) values (?, ?, ?, ?, ?, ?)
Hibernate: insert into person_table (gender, name, course, qualification, salary, type) values (?, ?, ?, ?, ?, ?)
Aug 14, 2022 2:27:08 PM org.hibernate.engine.jdbc.connections.internal.DriverBasedConnectionProvider [org.hibernate.engine.jdbc.connections.internal.DriverBasedConnectionProvider]
INFO: HHH10001008: Cleaning up connection pool [jdbc:mysql://localhost:3306/maven]
```



1