

## Sample Hibernate Application using Annotations

Let us create a Sample Application using Annotations

1. Create a new Maven Project
2. Add the dependencies
3. Create a package called com.wipro.bean
  - a. Create a class called Student (Persistent Class)
  - b. As Annotated Class

```
package com.wipro.bean;

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

@Entity
@Table(name="STUDENT_TBL")
public class Student {
    @Id
    @Column(name="stdid")
    private int studentId;
    @Column(name="stdName",length=20,nullable=false)
    private String studentName;
    private String course;
    public Student() {
        super();
        // TODO Auto-generated constructor stub
    }
    public Student(int studentId, String studentName, String course) {
        super();
        this.studentId = studentId;
        this.studentName = studentName;
        this.course = course;
    }
    public int getStudentId() {
        return studentId;
    }
    public void setStudentId(int studentId) {
        this.studentId = studentId;
    }
    public String getStudentName() {
        return studentName;
    }
    public void setStudentName(String studentName) {
        this.studentName = studentName;
    }
    public String getCourse() {
        return course;
    }
    public void setCourse(String course) {
        this.course = course;
    }
}
```

```

@Override
public String toString() {
    return "Student [studentId=" + studentId + ", studentName=" +
studentName + ", course=" + course + "]";
}
}

```

4. We need to create the configuration file
  - a. Called **myconfig.cfg.xml** instead of the default 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">

<hibernate-configuration>
<session-factory>
    <!-- Oracle Dialect -->
    <property
name="hibernate.dialect">org.hibernate.dialect.Oracle10gDialect</property>
    <!-- Database Connection Settings -->
    <property
name="hibernate.connection.driver_class">oracle.jdbc.OracleDriver</property>
    <property
name="hibernate.connection.url">jdbc:oracle:thin:@localhost:1521:orcl</property>
    <property name="hibernate.connection.username">scott</property>
    <property name="hibernate.connection.password">tiger</property>
    <!-- To echo / show all executed queries on server output -->
    <property name="hibernate.show_sql">true</property>
    <!-- Used to Create or Alter Table -->
    <!-- Given class structure not present in the database -->
    <property name="hibernate.hbm2ddl.auto">update</property>

    <property name="hibernate.enable_lazy_load_no_trans">true</property>
    <!-- Mapping file entry -->
    <!-- We need to map the class and not the mapping file -->
    <mapping class="com.wipro.bean.Student"/>
</session-factory>
</hibernate-configuration>

```

5. Create the main class called StudentTester to test the created application
  - a. Since we have created configuration with custom name called myconfig.xml we have to pass it as a parameter to the configure() method

```

package com.wipro.service;

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

import com.wipro.bean.Student;

```

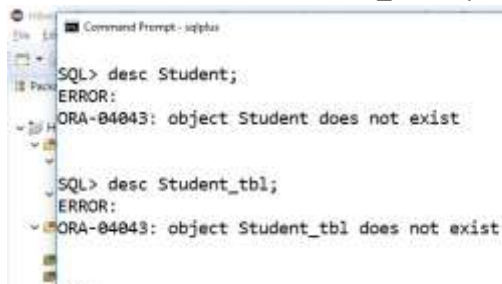
```

public class StudentTester {

    public static void main(String[] args) {
        Configuration cfg = new Configuration().configure("myconfig.cfg.xml");
        SessionFactory sf = cfg.buildSessionFactory();
        Session session = sf.openSession();
        Transaction transaction = session.beginTransaction();
        Student student = new Student(101, "Harry", "Java");
        session.save(student);
        transaction.commit();
        System.out.println("Record inserted");
        session.close();
    }
}

```

6. We check whether the Student\_TBL is present in the database



```

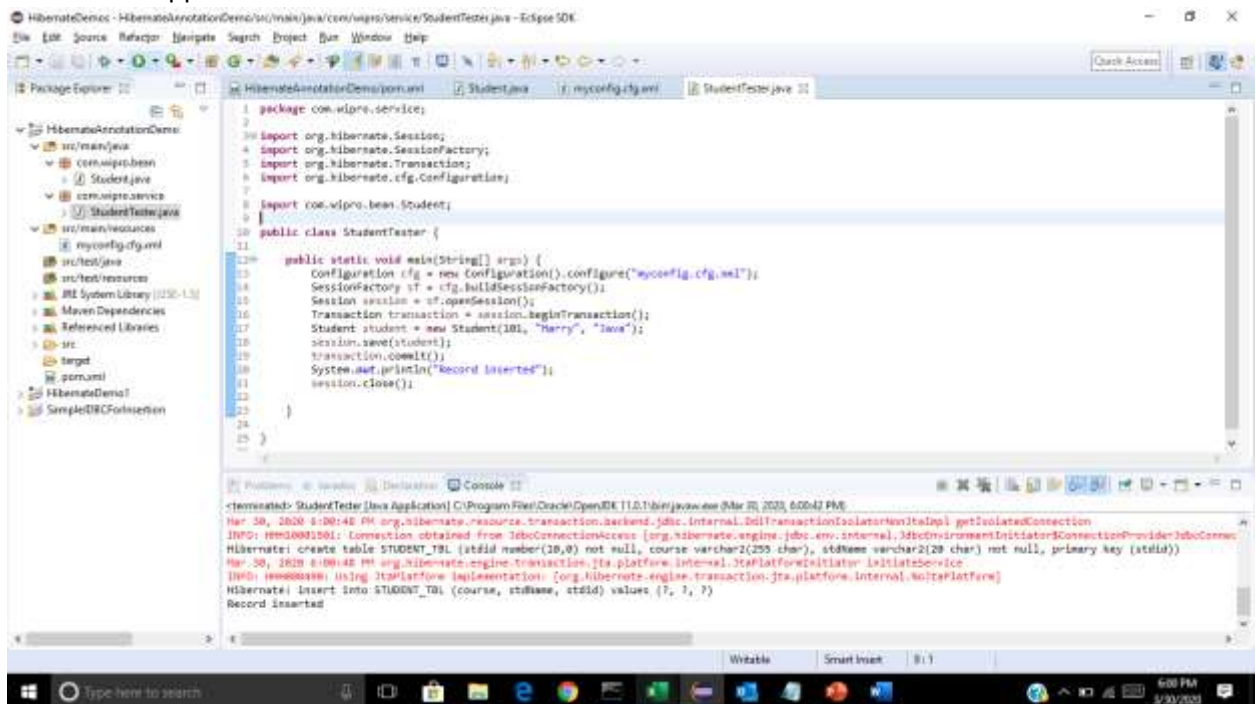
Command Prompt - sqlplus

SQL> desc Student;
ERROR:
ORA-04043: object Student does not exist

SQL> desc Student_tbl;
ERROR:
ORA-04043: object Student_tbl does not exist

```

7. Execute the application



```

package com.wipro.service;

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

import com.wipro.bean.Student;

public class StudentTester {

    public static void main(String[] args) {
        Configuration cfg = new Configuration().configure("myconfig.cfg.xml");
        SessionFactory sf = cfg.buildSessionFactory();
        Session session = sf.openSession();
        Transaction transaction = session.beginTransaction();
        Student student = new Student(101, "Harry", "Java");
        session.save(student);
        transaction.commit();
        System.out.println("Record inserted");
        session.close();
    }
}

```

```

<terminated>: StudentTester [Java Application] C:\Program Files\Oracle\OpenIDE\T10\bin\javaw.exe (Mar 30, 2020, 6:00:42 PM)
Mar 30, 2020 6:00:42 PM org.hibernate.resource.transaction.backend.jdbc.internal.DdlTransactionIsolatorNonJtaImpl getIsolatedConnection
INFO: HHH10001501: Connection obtained from JdbcConnectionAccess [org.hibernate.engine.jdbc.env.internal.JdbcEnvironmentInitiator$ConnectionProviderJdbcConnectionAccess]
Mar 30, 2020 6:00:42 PM org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformInitiator initializeService
INFO: HHH0000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
Hibernate: Insert into STUDENT_TBL (course, stdName, stdId) values (?, ?, ?)
Record inserted

```

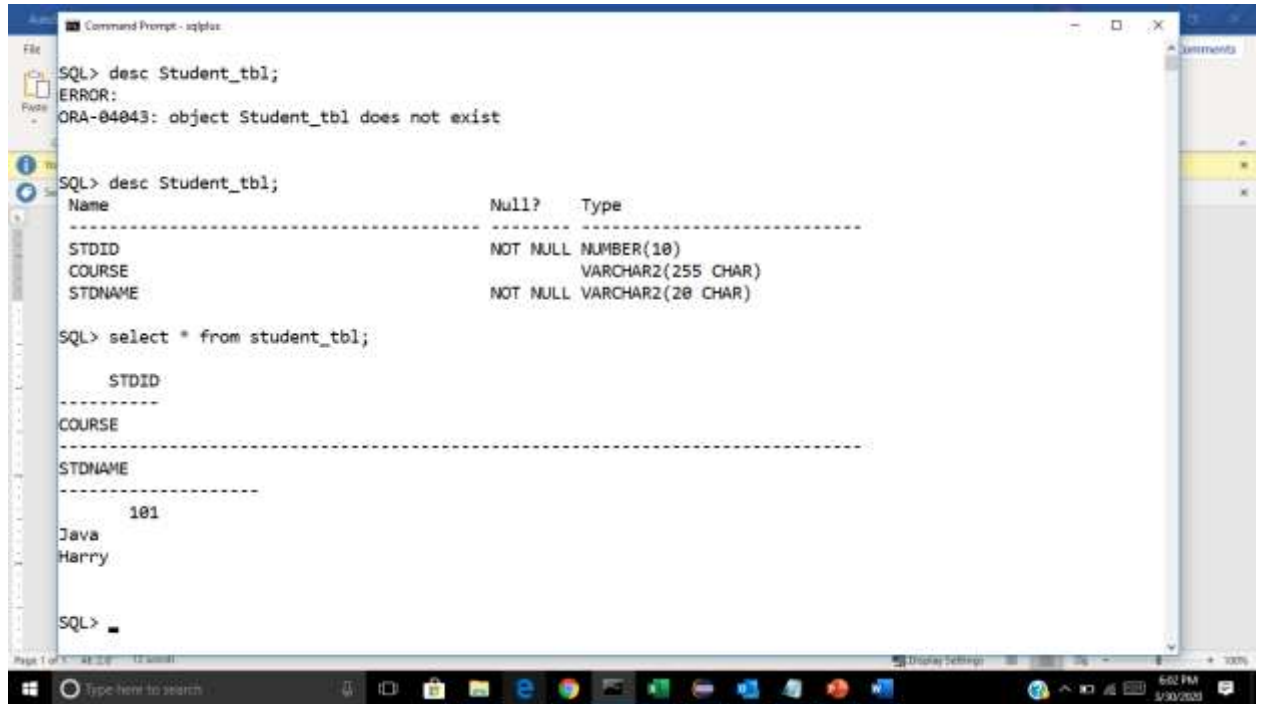
- a. As the table is not present in the database and we have given `hibernate.hbm2ddl.auto` property as update, It creates a new table.

```

INFO: HHH10001501: Connection obtained from JdbcConnectionAccess [org.hibernate.engine.jdbc.env.internal.JdbcEnvironmentInitiator$ConnectionProviderJdbc
Hibernate: create table STUDENT_TBL (stdId number(10,0) not null, course varchar2(255 char), stdName varchar2(20 char) not null, primary key (stdId))
Mar 30, 2020 6:00:48 PM org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform

```

- b. Then inserts the data into the table.
8. Check the solution in database



The screenshot shows a Windows Command Prompt window titled "Command Prompt - sqlplus". The user has entered the following SQL commands:

```
SQL> desc Student_tbl;
ERROR:
ORA-04043: object Student_tbl does not exist

SQL> desc Student_tbl;
Name                               Null?    Type
-----
STDID                               NOT NULL NUMBER(10)
COURSE                             VARCHA2(255 CHAR)
STDNAME                             NOT NULL VARCHA2(255 CHAR)

SQL> select * from student_tbl;

   STDID
-----
COURSE
-----
STDNAME
-----
      101
Java
Harry

SQL>
```

The output of the second `desc` command shows the table structure with columns `STDID`, `COURSE`, and `STDNAME`. The output of the `select` command shows a single row with values 101, Java, and Harry.