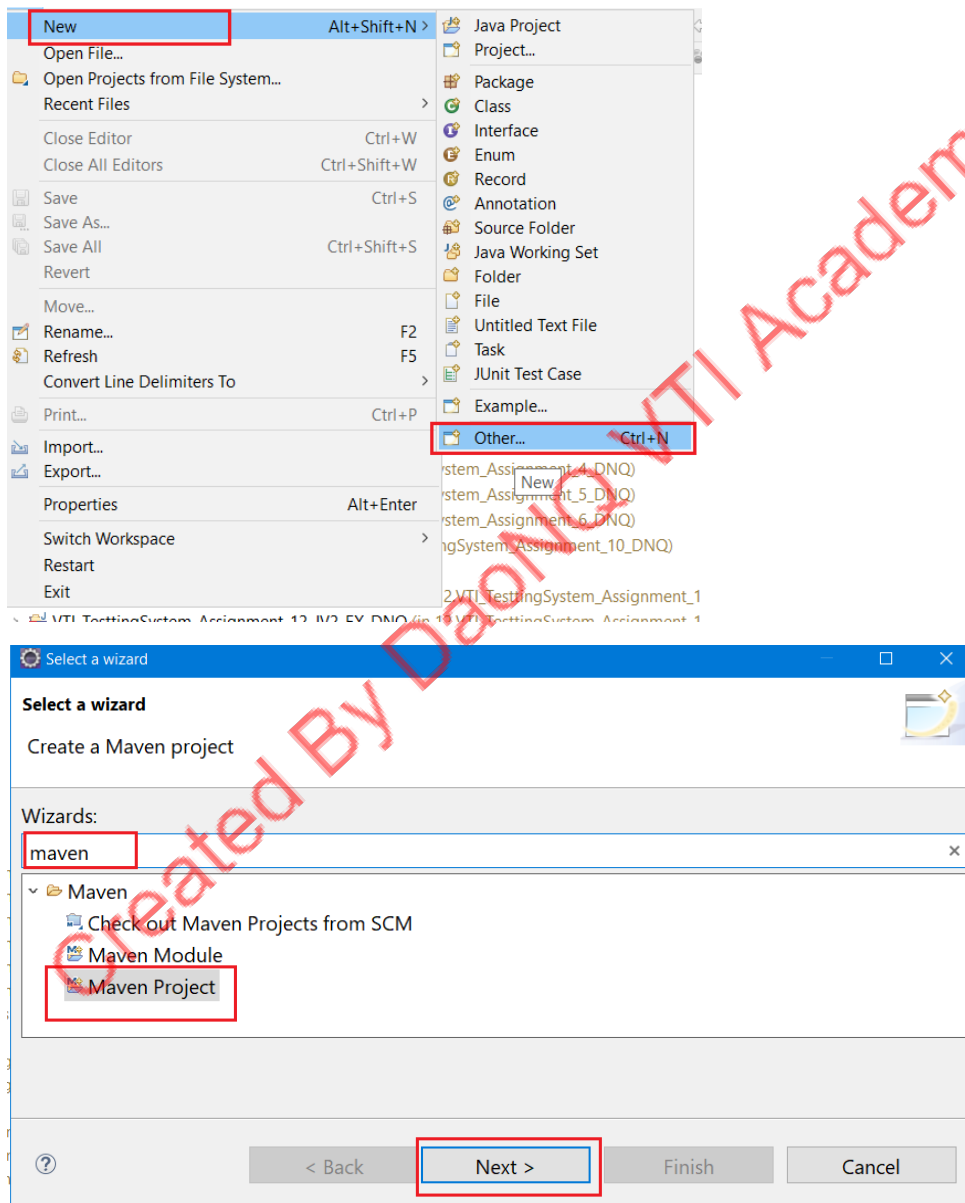


## Tạo ứng dụng test kết nối tới DB TestingSystem bằng Hibernate.

1. **Tạo Project maven:** RainlWay9\_10\_TesttingSystem\_1\_1



**New Maven Project**

Select project name and location

☐ Create a simple project (skip archetype selection)

☒ Use default Workspace location

Location:

☐ Add project(s) to working set

Working set:

► Advanced

**New Maven Project**

Select an Archetype

Catalog:

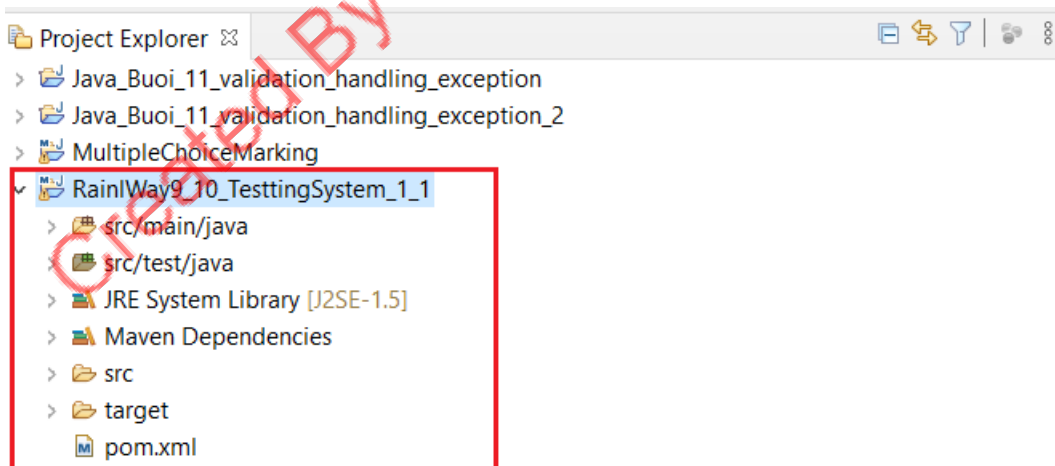
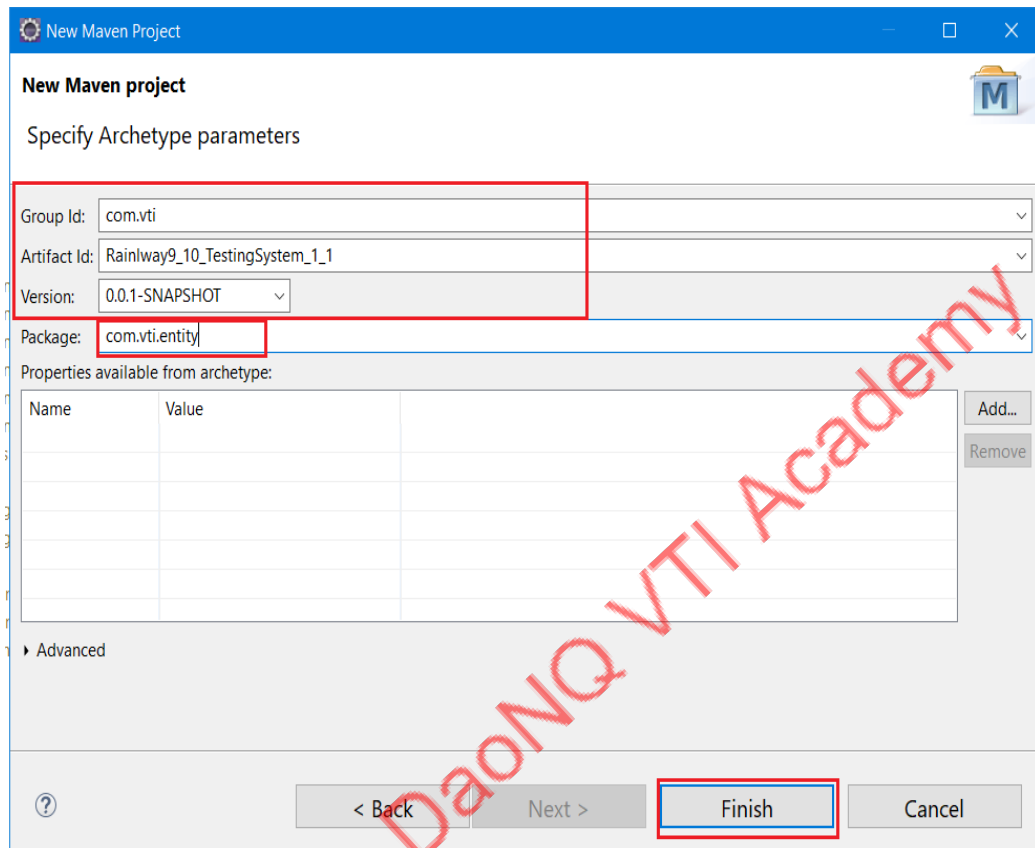
Filter:

Group Id	Artifact Id	Version
org.apache.maven.archetypes	maven-archetype-j2ee-simple	1.0
org.apache.maven.archetypes	maven-archetype-plugin	1.2
org.apache.maven.archetypes	maven-archetype-plugin-site	1.1
org.apache.maven.archetypes	maven-archetype-portlet	1.0.1
org.apache.maven.archetypes	maven-archetype-profiles	1.0-alpha-4
org.apache.maven.archetypes	maven-archetype-quickstart	1.1
org.apache.maven.archetypes	maven-archetype-site	1.1

An archetype which contains a sample Maven project.

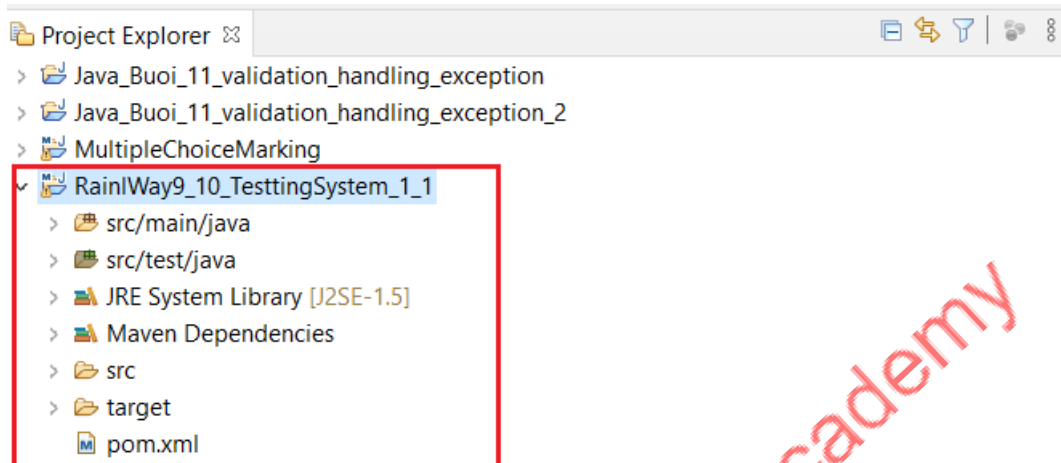
☒ Show the last version of Archetype only ☐ Include snapshot archetypes

► Advanced



## 2. Tạo khung chương trình:

- Tạo mới project: RainWay9\_10\_TestingSystem\_1\_1



### 3. Thêm các cấu hình cho file: pom.xml

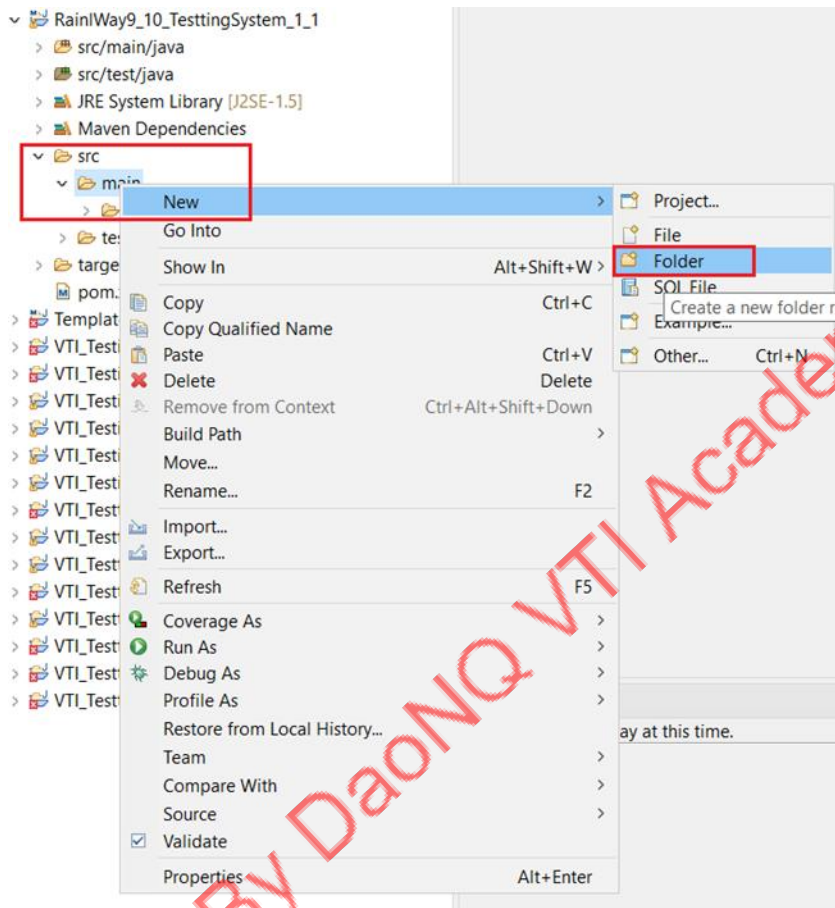
```
<dependency>
    <groupId>org.hibernate</groupId>
    <artifactId>hibernate-core</artifactId>
    <version>5.4.17.Final</version>
</dependency>

<!-- https://mvnrepository.com/artifact/mysql/mysql-connector-java -->
<dependency>
    <groupId>mysql</groupId>
    <artifactId>mysql-connector-java</artifactId>
    <version>8.0.20</version>
</dependency>
```

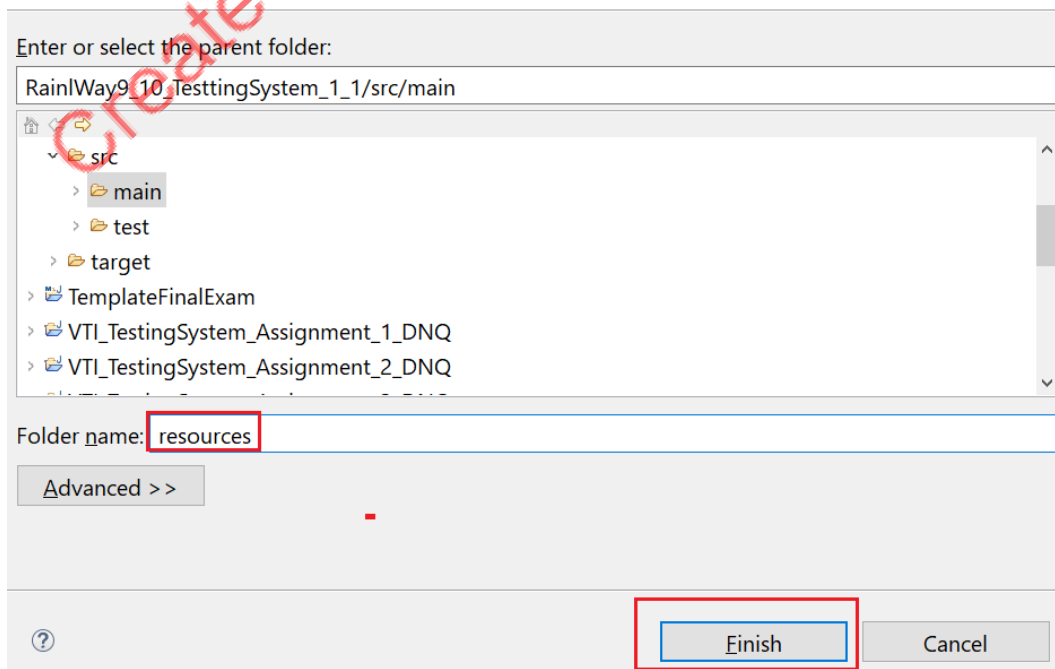


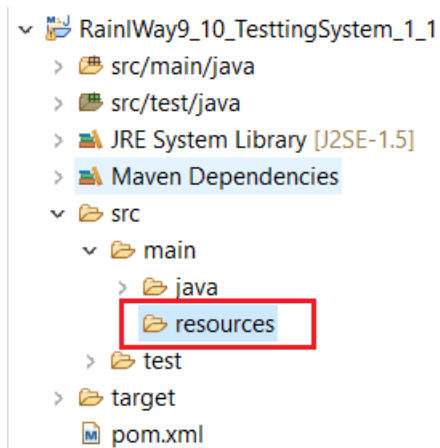
### 4. Thêm các cấu hình cho file: hibernate.cfg.xml trong resources

- Tạo mới folder resources, đường dẫn **src.main.resources**. chú ý viết đúng tên folder.

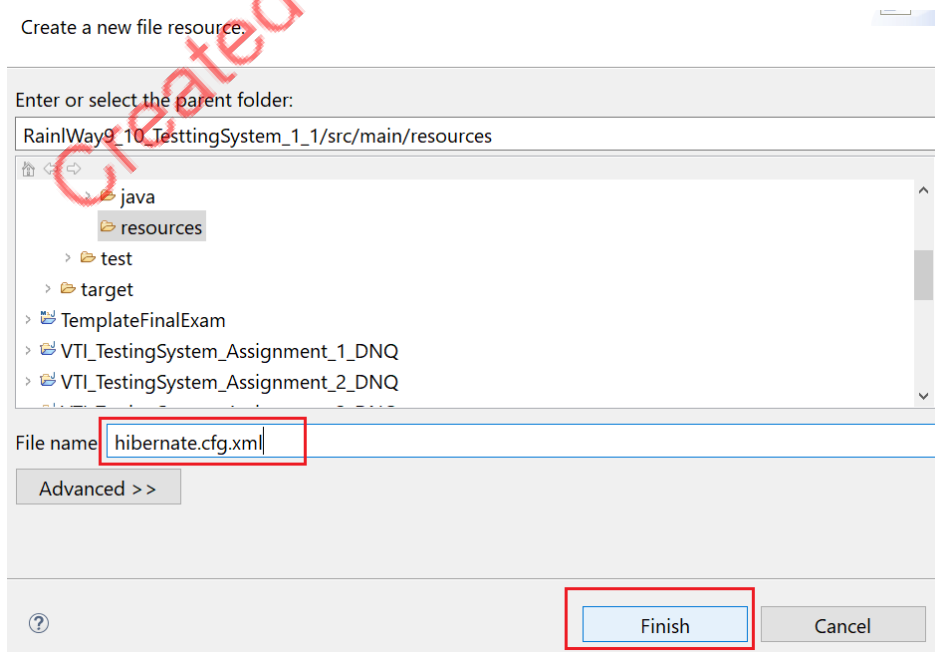
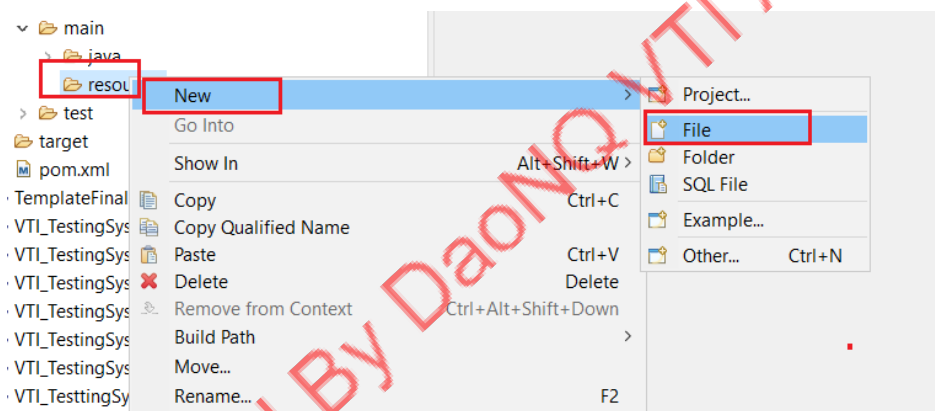


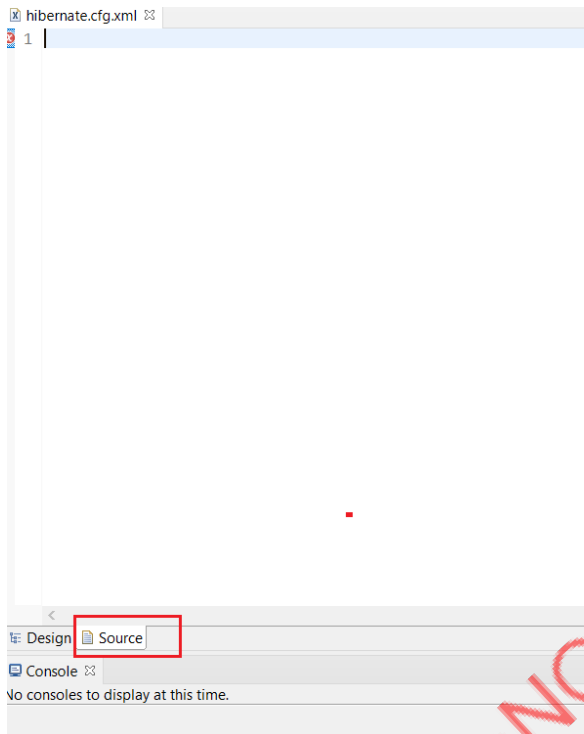
Create a new folder resource.





- Tạo mới file: **hibernate.cfg.xml** trong folder vừa tạo.





- Copy nội dung bên dưới vào file vừa tạo.

```
<?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>
        <!-- Database connection settings -->
        <property name="connection.driver_class">com.mysql.cj.jdbc.Driver</property>
        <property
name="connection.url">jdbc:mysql://localhost:3306/TestingSystem</property>
        <property name="connection.username">root</property>
        <property name="connection.password">root</property>

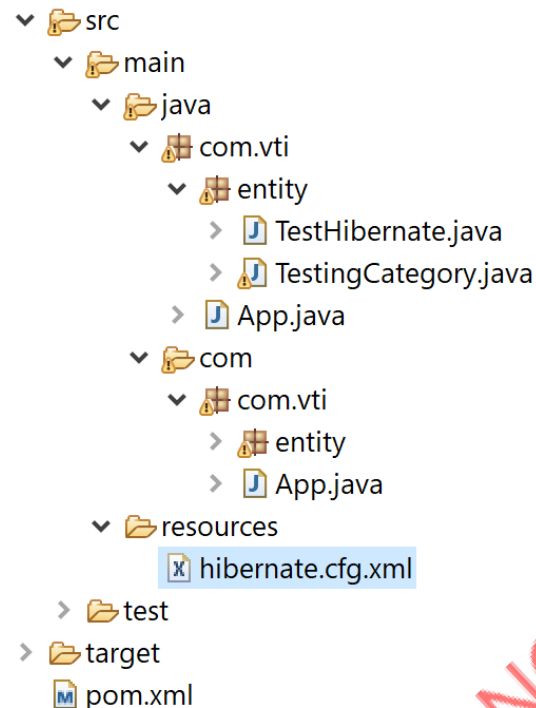
        <!-- format code SQL -->
        <property name="show_sql">true</property>
        <property name="hibernate.format_sql">true</property>
        <property name="connection.pool_size">10</property>

        <!-- other -->
        <property name="hibernate.connection.characterEncoding">utf8</property>

    </session-factory>
</hibernate-configuration>
```



hibernate.cfg.xml



Tạo DB TestingSystem để Demo:

```
-- create database
DROP DATABASE IF EXISTS TestingSystem;
CREATE DATABASE TestingSystem;
USE TestingSystem;

-- create table 1: Department
DROP TABLE IF EXISTS Testing_Category;
CREATE TABLE Testing_Category(
    id TINYINT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    `name` NVARCHAR(50) NOT NULL UNIQUE KEY
);

INSERT INTO Testing_Category(`name`)
VALUES
    ('Category 1'),
    ('Category 2'),
    ('Category 3'),
    ('Category 4'),
    ('Category 5'),
    ('Category 6'),
    ('Category 7');
```

## 5. Tạo Class TestingCategory trong package com.vti.entity:

```
package com.vti.entity;

import java.io.Serializable;

import javax.persistence.Column;
import javax.persistence.Entity;
import javax.persistence.GeneratedValue;
import javax.persistence.GenerationType;
```



```

import javax.persistence.Id;
import javax.persistence.Table;

import net.bytebuddy.dynamic.loading.ClassReloadingStrategy.Strategy;

@Entity
@Table(name = "Testing_Category", catalog = "TestingSystem")
public class TestingCategory implements Serializable {
    @Column(name = "id")
    @Id
    @GeneratedValue(strategy = GenerationType.IDENTITY)
    private short id;
    @Column(name = "name", length = 50, nullable = false, unique = true)
    private String name;

    public TestingCategory() {
        super();
    }

    /**
     * @return the id
     */
    public short getId() {
        return id;
    }

    /**
     * @param id the id to set
     */
    public void setId(short id) {
        this.id = id;
    }

    /**
     * @return the name
     */
    public String getName() {
        return name;
    }

    /**
     * @param name the name to set
     */
    public void setName(String name) {
        this.name = name;
    }
}

```



TestingCategory.java

## 6. Tạo Class TestHibernate.java trong package com.vti.entity

```
package com.vti.frontend;
```

```

import java.util.List;

import org.hibernate.Session;
import org.hibernate.SessionFactory;
import org.hibernate.boot.registry.StandardServiceRegistryBuilder;
import org.hibernate.cfg.Configuration;
import org.hibernate.query.Query;
import org.hibernate.service.ServiceRegistry;

import com.vti.entity.TestingCategory;

public class TestHibernate {
    public static void main(String[] args) {
        Session session = null;
        try {
            session = buildSessionFactory().openSession();
            session.beginTransaction();
            // Insert dữ liệu:
            TestingCategory testingcategory = new TestingCategory();
            testingcategory.setName("Category Test");

            session.save(testingcategory);
            System.out.println("Create Success!!!");

            // Select dữ liệu cơ bản:
            Query<TestingCategory> query = session.createQuery("FROM
TestingCategory");

            List<TestingCategory> listResul = query.list();
            for (TestingCategory testingCategory2 : listResul) {
                System.out.println(testingCategory2.getId() + " " +
testingCategory2.getName());
            }

            session.getTransaction().commit();
        } finally {
            if (session != null) {
                session.close();
            }
        }
    }

    private static SessionFactory buildSessionFactory() {
        Configuration configuration = new Configuration();
        configuration.configure("hibernate.cfg.xml");
        configuration.addAnnotatedClass(TestingCategory.class);
        ServiceRegistry serviceRegistry = new
StandardServiceRegistryBuilder()

        .applySettings(configuration.getProperties()).build();
        return configuration.buildSessionFactory(serviceRegistry);
    }
}

```



TestHibernate.java

## 7. Chạy chương trình để kiểm tra:

```
Console
<terminated> TestHibernate [Java Application] C:\Program Files\Java\jdk-11.0.10\bin\javaw.exe (Jun 5, 2021, 4:43:17 PM - 4:43:21 PM)
INFO: HHH10001003: Autocommit mode: false
Jun 05, 2021 4:43:19 PM org.hibernate.engine.jdbc.connections.internal.DriverManagerConnectionProviderImpl$PooledConnections <init>
INFO: HHH000115: Hibernate connection pool size: 10 (min=1)
Jun 05, 2021 4:43:20 PM org.hibernate.dialect.Dialect <init>
INFO: HHH000400: Using dialect: org.hibernate.dialect.MySQL8Dialect
Jun 05, 2021 4:43:20 PM org.hibernate.engine.transaction.jta.platform.internal.JtaPlatformInitiator initiateService
INFO: HHH000490: Using JtaPlatform implementation: [org.hibernate.engine.transaction.jta.platform.internal.NoJtaPlatform]
Hibernate:
insert
into
TestingSystem.Testing_Category
(name)
values
(?)
Create Success!!!
```