组长:魏志航

组员:兰天阳，彭尹垚，张世纪，叶翔昊，谢泽川

**魏志航**

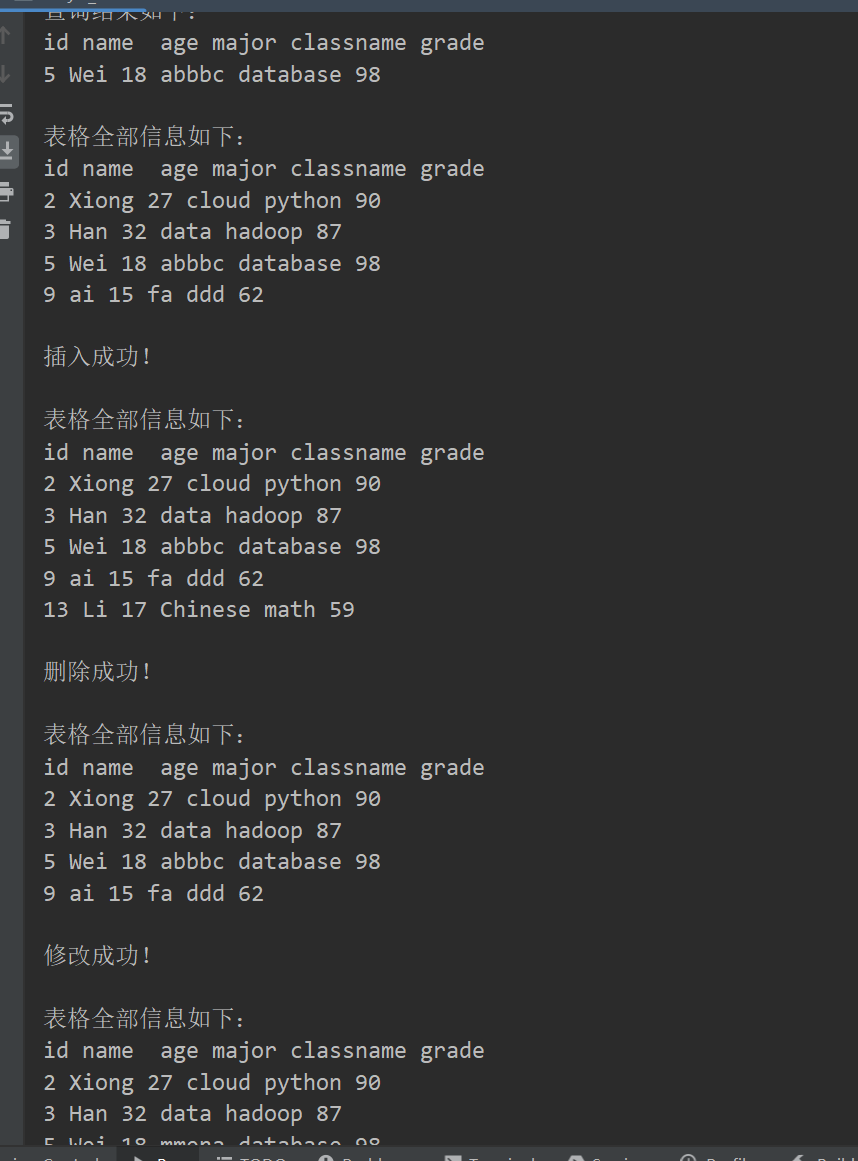
今天主要学习了

1.maven的安装、环境配置与使用

2.spring框架的配置与使用

**任务完成如下：**

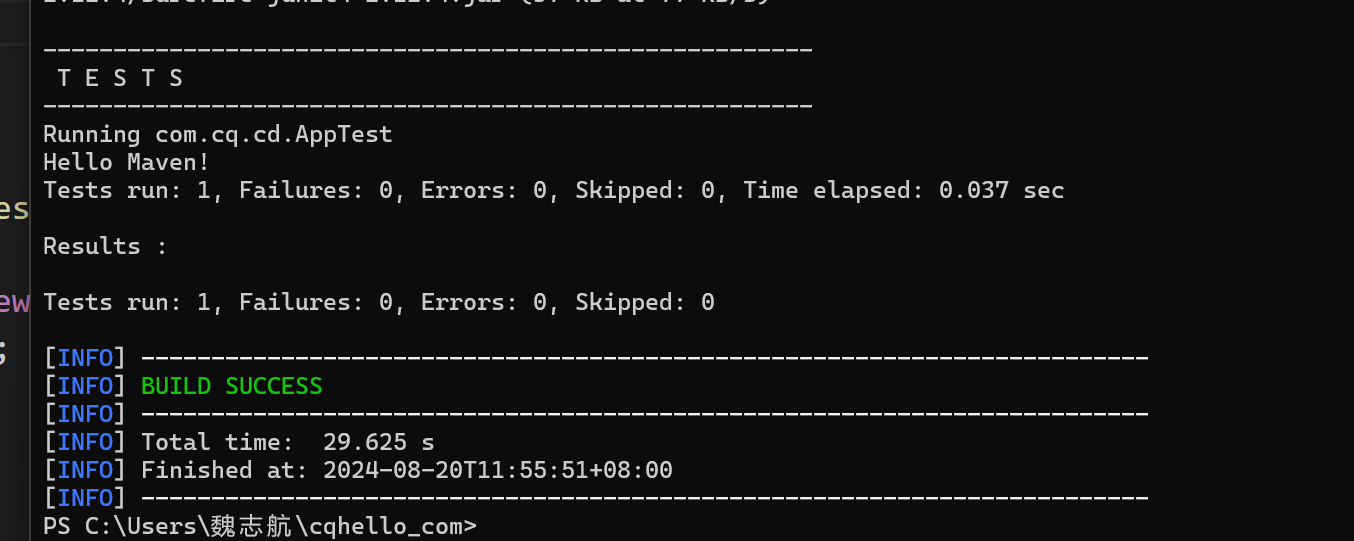
1.补齐了昨天用java封装sql语句的任务

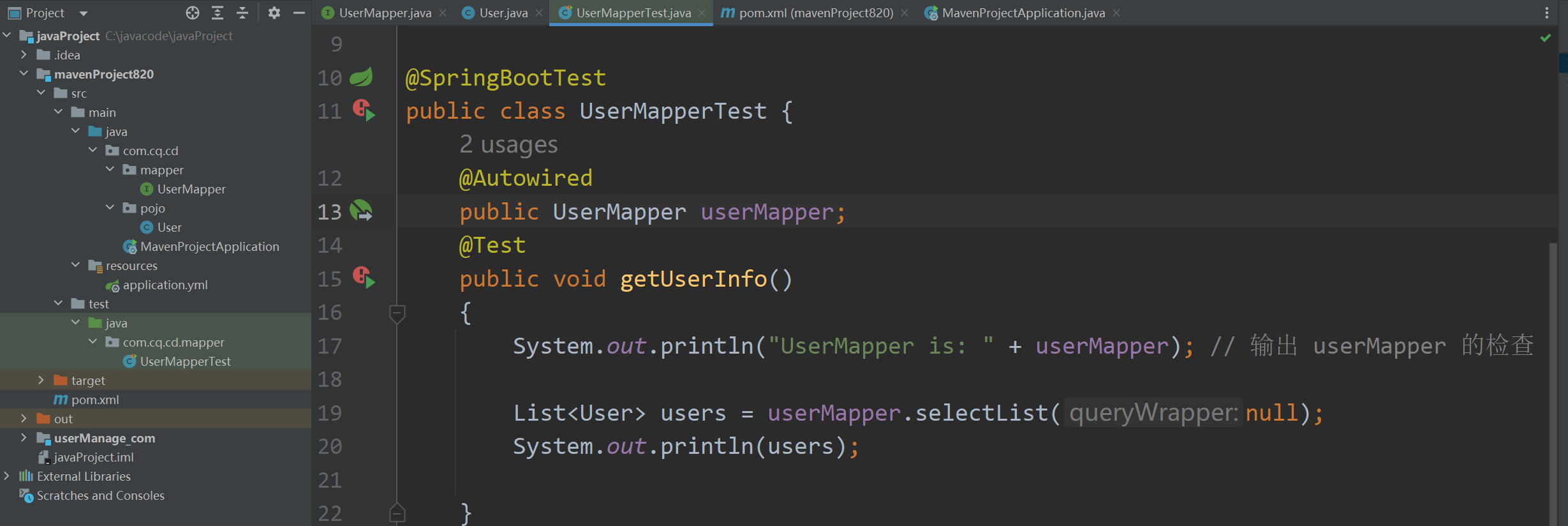


**代码如下：**

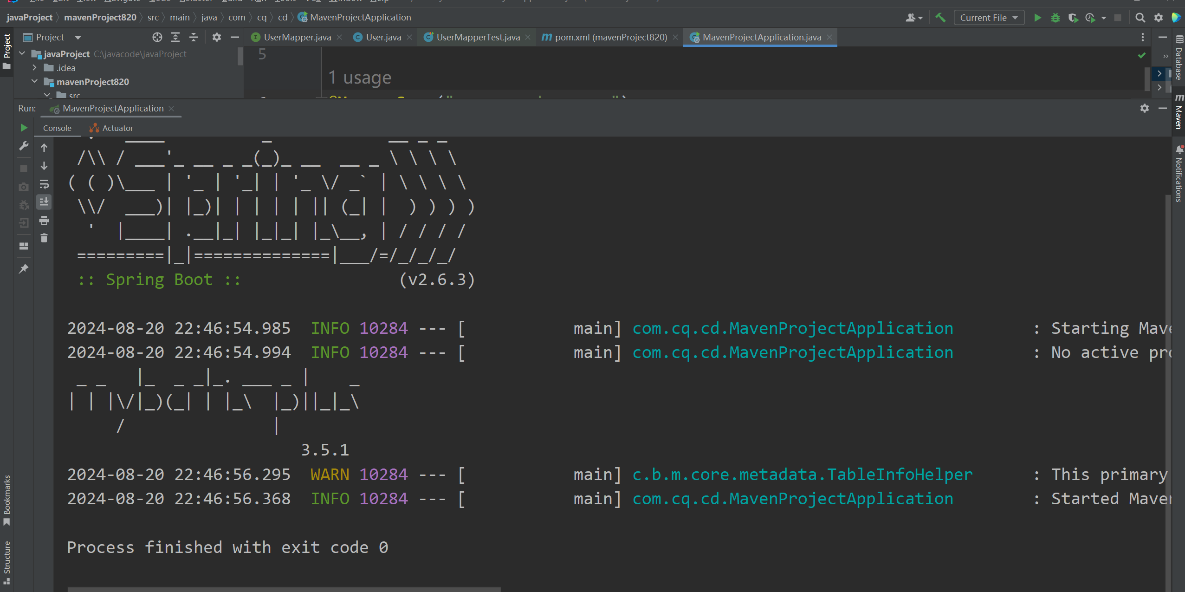
**import java.sql.SQLException;  
import java.sql.Connection;  
import java.sql.DriverManager;  
import java.sql.ResultSet;  
import java.sql.PreparedStatement;  
  
import java.sql.Statement;  
//封装sql语句  
public class Day5\_2  
{  
 static String *url* = "jdbc:mysql://localhost:3306/test";  
 static String *username* = "root";  
 static String *password* = "200414";  
 public static void main(String[] args) throws ClassNotFoundException, SQLException  
 {  
 //加载驱动  
 Class.*forName*("com.mysql.jdbc.Driver");  
 *inquiry*(5);  
 *showAll*();  
 *insert*(13,"Li",17,"Chinese","math",59);  
 *showAll*();  
 *delete*(13);  
 *showAll*();  
 *change*(5,"major","mmena");  
 *showAll*();  
  
 }  
  
 public static void insert(int id,String name,int age, String major,String classname,int grade)throws ClassNotFoundException, SQLException  
 {  
 Connection conn = DriverManager.*getConnection*(*url*, *username*, *password*);  
 String sql = "insert into Day5member values("+id+",'"+name+"',"+age+",'"+major+"','"+classname+"',"+grade+")";  
 PreparedStatement preparedStatement = conn.prepareStatement(sql);  
 preparedStatement.executeUpdate();  
 System.*out*.println("插入成功！\n");  
 }  
  
 public static void delete(int id)throws ClassNotFoundException, SQLException  
 {  
 Connection conn = DriverManager.*getConnection*(*url*, *username*, *password*);  
 String sql = "delete from Day5member where id ="+id;  
 PreparedStatement preparedStatement = conn.prepareStatement(sql);  
 preparedStatement.executeUpdate();  
 System.*out*.println("删除成功！\n");  
 }  
  
 public static void inquiry(int id)throws ClassNotFoundException, SQLException  
 {  
 Connection conn = DriverManager.*getConnection*(*url*, *username*, *password*);  
 Statement statement = conn.createStatement();  
 String select\_sql="select \* from Day5member where id ="+id;  
 ResultSet rs = statement.executeQuery(select\_sql);  
 //System.out.println(rs);  
 System.*out*.println("查询结果如下：");  
 System.*out*.println("id name age major classname grade");  
 while(rs.next())  
 {  
  
 String name =rs.getString("name");  
 String idd =rs.getString("id");  
 String age =rs.getString("age");  
 String major =rs.getString("major");  
 String classname =rs.getString("classname");  
 String grade =rs.getString("grade");  
  
 System.*out*.println(idd+" "+name+" "+ age+" "+major+" "+classname+" "+grade);  
 System.*out*.println();  
 }  
 }  
  
 public static void showAll()throws ClassNotFoundException, SQLException  
 {  
 Connection conn = DriverManager.*getConnection*(*url*, *username*, *password*);  
 Statement statement = conn.createStatement();  
 String select\_sql="select \* from Day5member ";  
 ResultSet rs = statement.executeQuery(select\_sql);  
 //System.out.println(rs);  
 System.*out*.println("表格全部信息如下：");  
 System.*out*.println("id name age major classname grade");  
 while(rs.next())  
 {  
 String name =rs.getString("name");  
 String id =rs.getString("id");  
 String age =rs.getString("age");  
 String major =rs.getString("major");  
 String classname =rs.getString("classname");  
 String grade =rs.getString("grade");  
  
 System.*out*.println(id+" "+name+" "+ age+" "+major+" "+classname+" "+grade);  
  
 }  
 System.*out*.println();  
 }  
  
 public static void change(int id, String a,String aim)throws ClassNotFoundException, SQLException  
 {  
 Connection conn = DriverManager.*getConnection*(*url*, *username*, *password*);  
 String sql = "update Day5member set "+a+"= '"+aim+"' where id="+id;  
 PreparedStatement preparedStatement = conn.prepareStatement(sql);  
 preparedStatement.executeUpdate();  
 System.*out*.println("修改成功！\n");  
 }  
  
}**

2.maven的安装和成功测试

  
3.用maven创建项目并调用spring框架

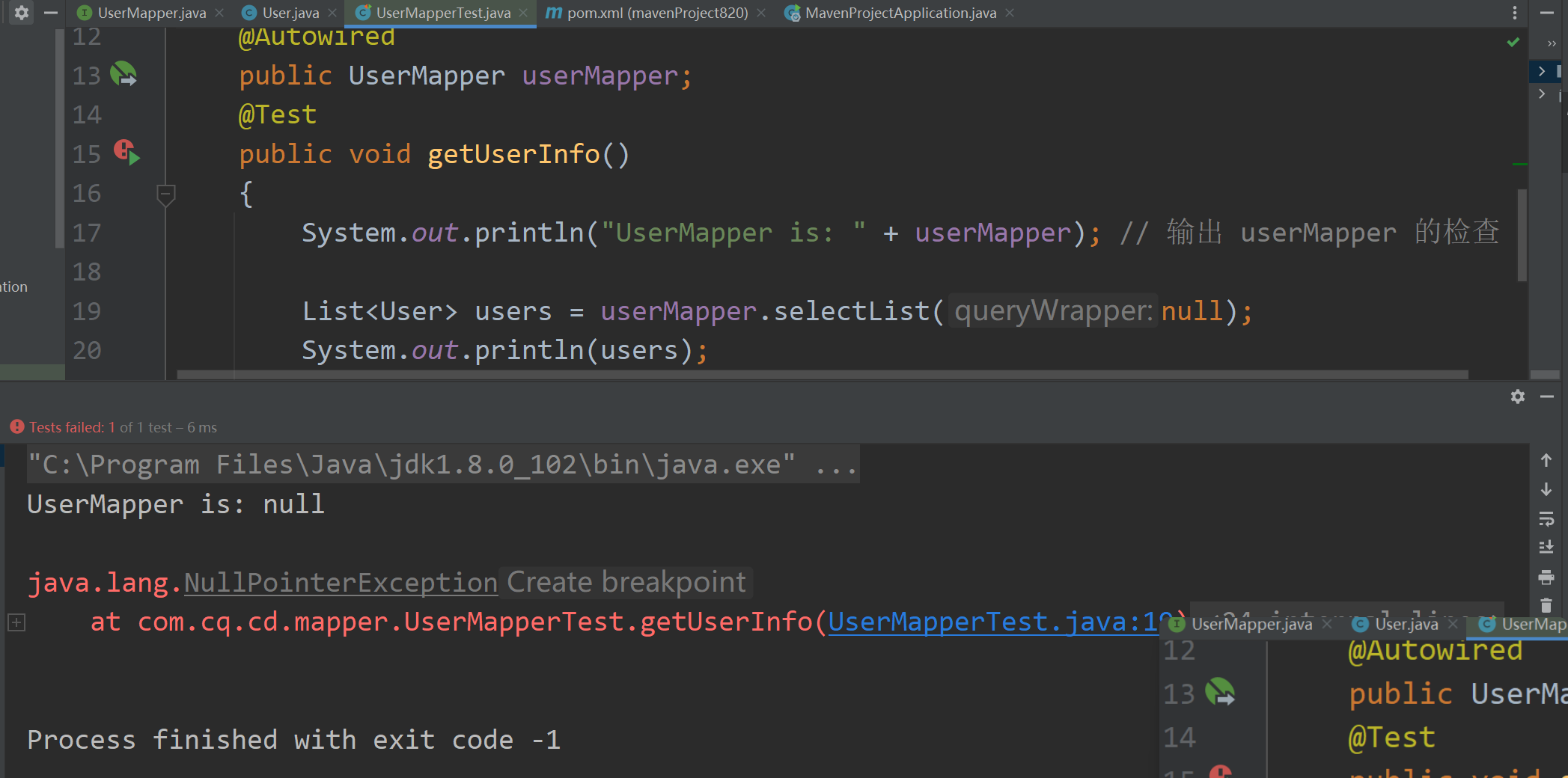


4. SpringBootApplication运行成功

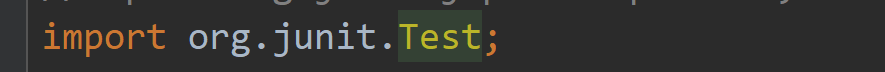


**5.报错：运行UserMapperTest测试时UserMapper为空**

****



**报错原因：Test包导入错误**

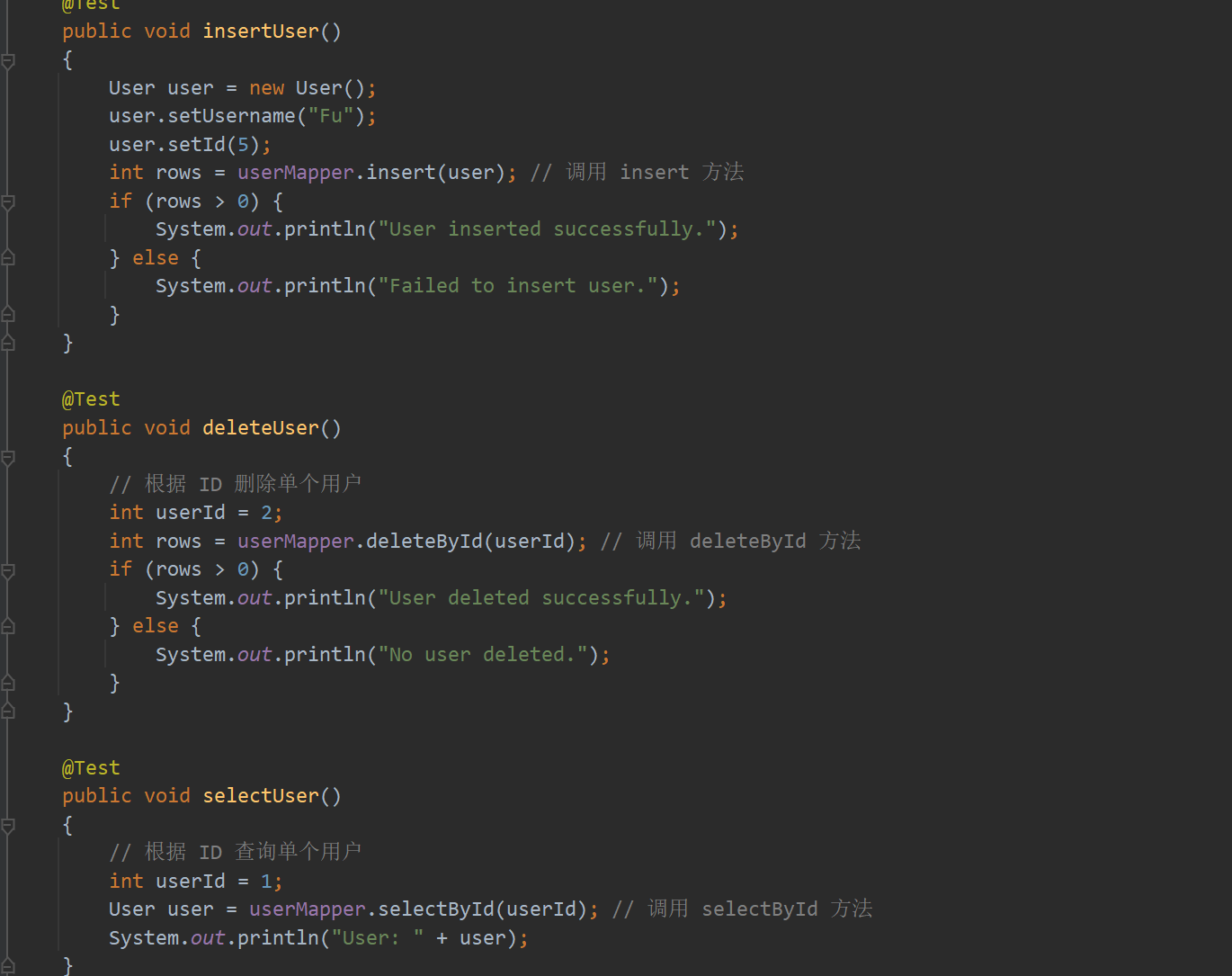
错误的：

正确的:

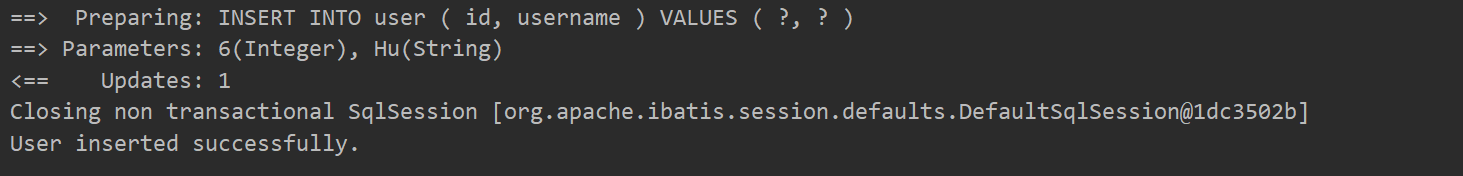
**修正后成功：**

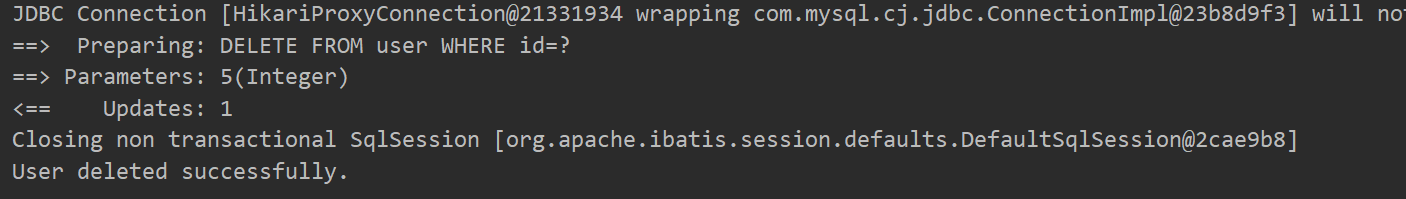


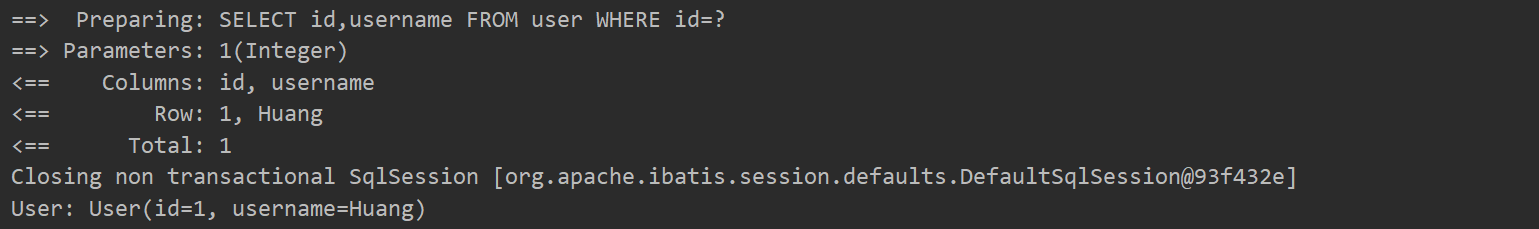
测试Mapper中的增删查改均成功：

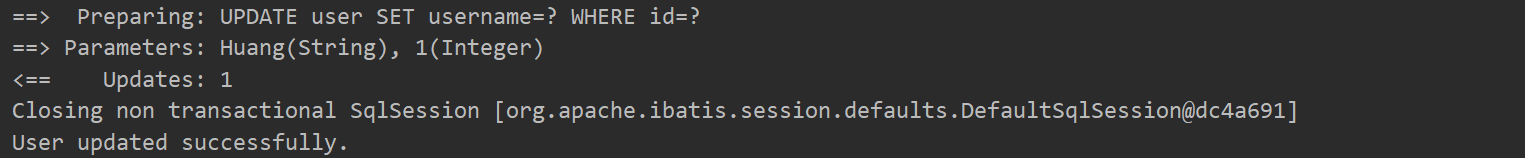












**彭尹垚**

Java学习日志

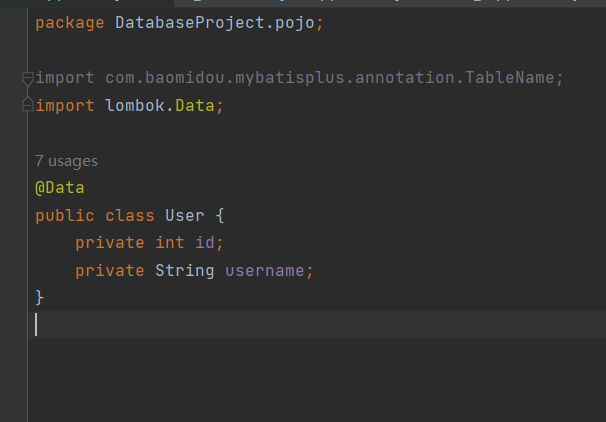
今天主要学习了在idea中编辑数据库，用idea集成maven，搭建了mybais-plus和springboot框架

在实际操作中遇到并解决了大部分问题

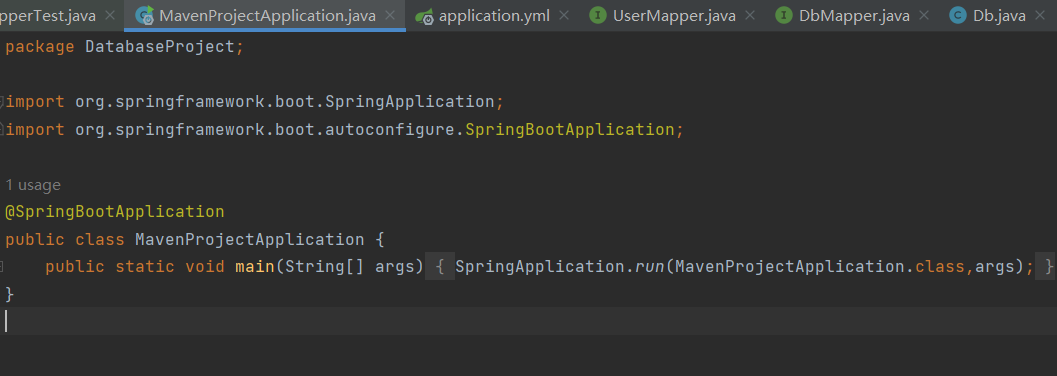
搭建好了运行依赖

构建了UserMapper 接口

User类的创建（映射到user表）：



启动文件



测试代码：



添加成功后成员来到4个，原2个

**谢泽川**

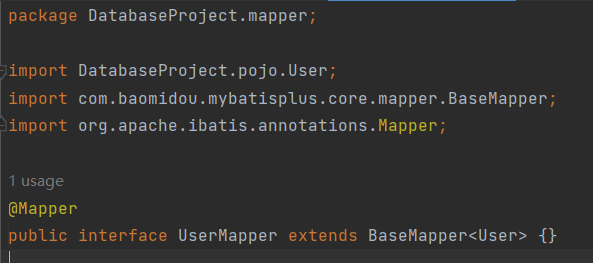
##### Java学习日志 - JAVA数据库的mybais-plus的学习和springboot框架的搭建

##### 日期：2024年8月20日

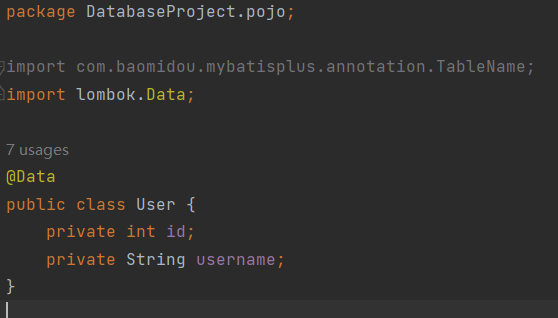
##### 学习内容：

学习了 JAVAJAVA数据库的mybais-plus的学习和springboot框架

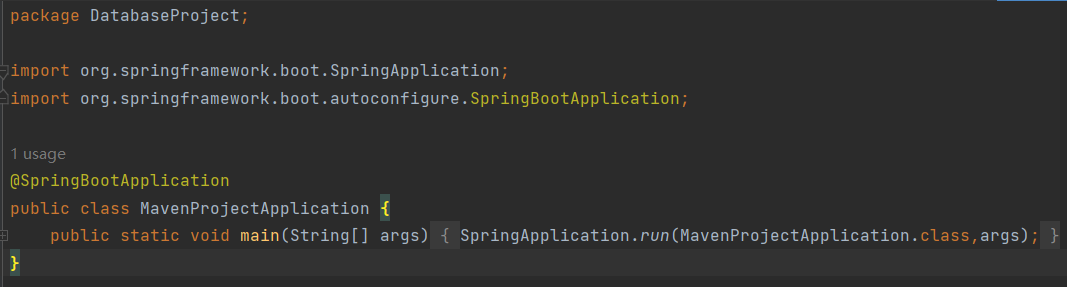
UserMapper的创建：



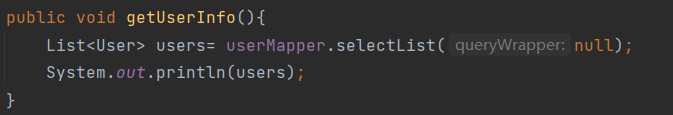
User类的创建（映射到user表）：



Springboot的启动类：

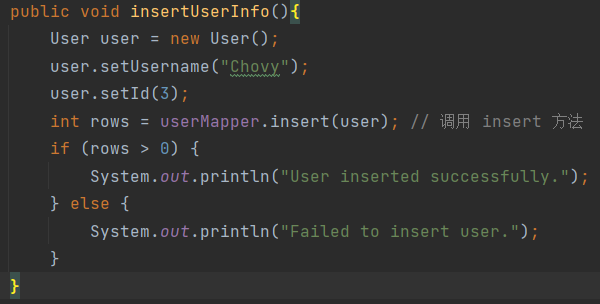


数据库查询方法的测试：





插入方法的测试：

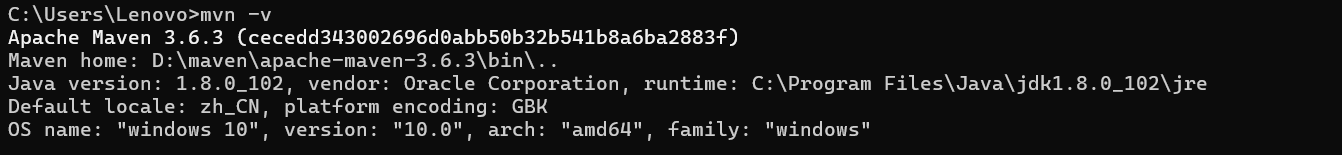


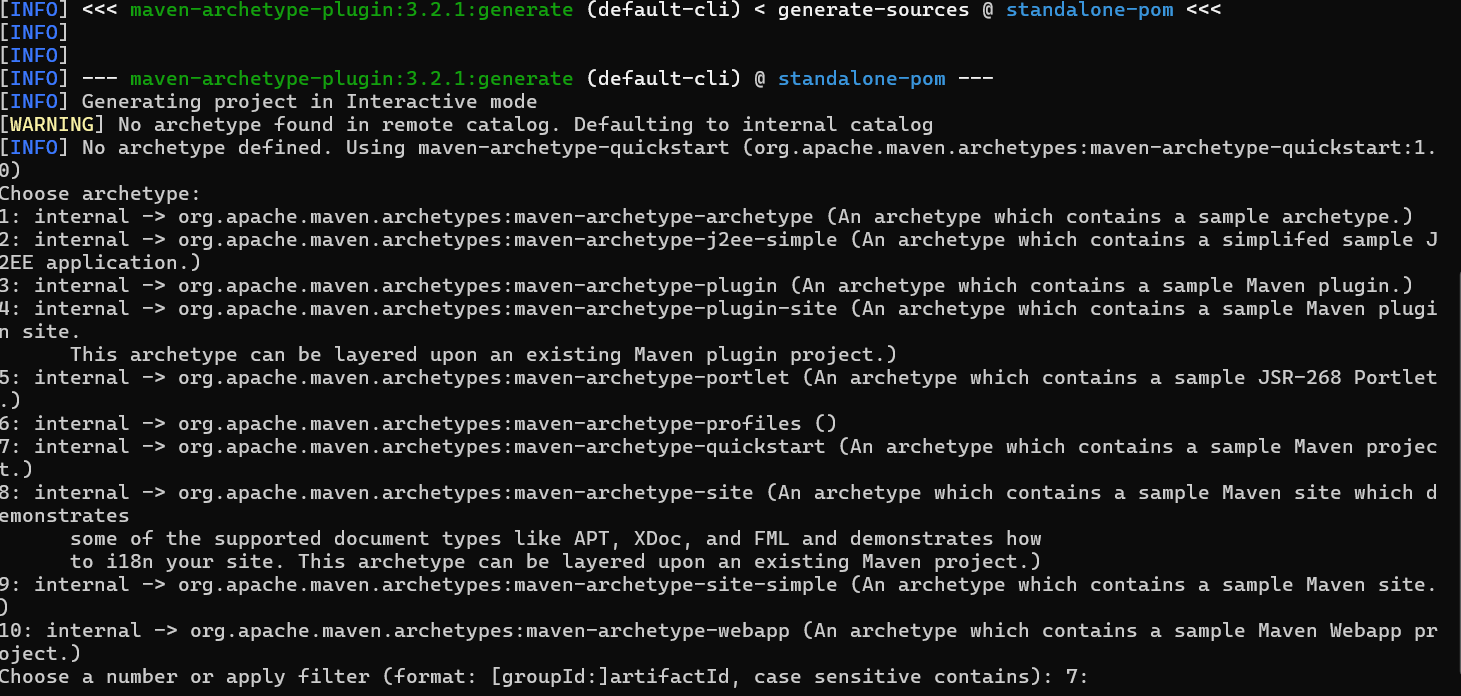




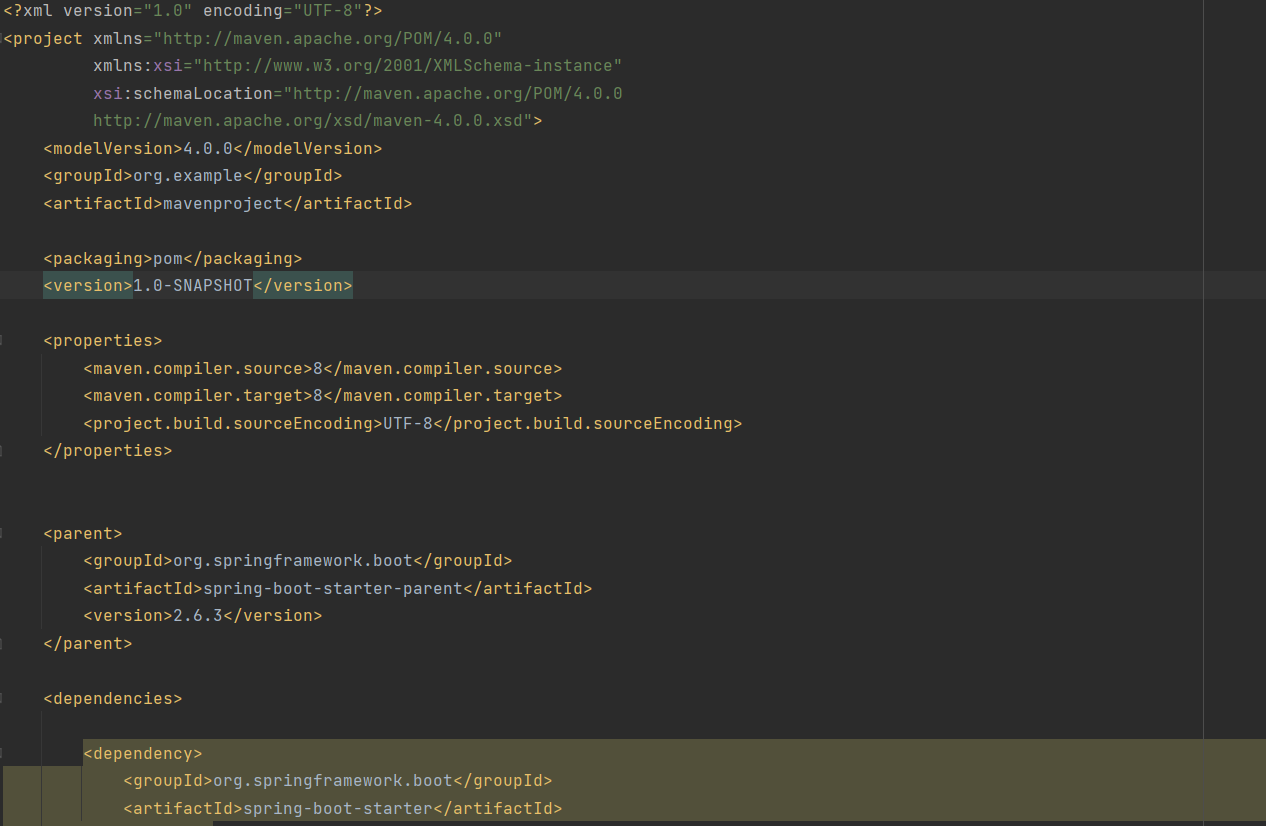
**张世纪**

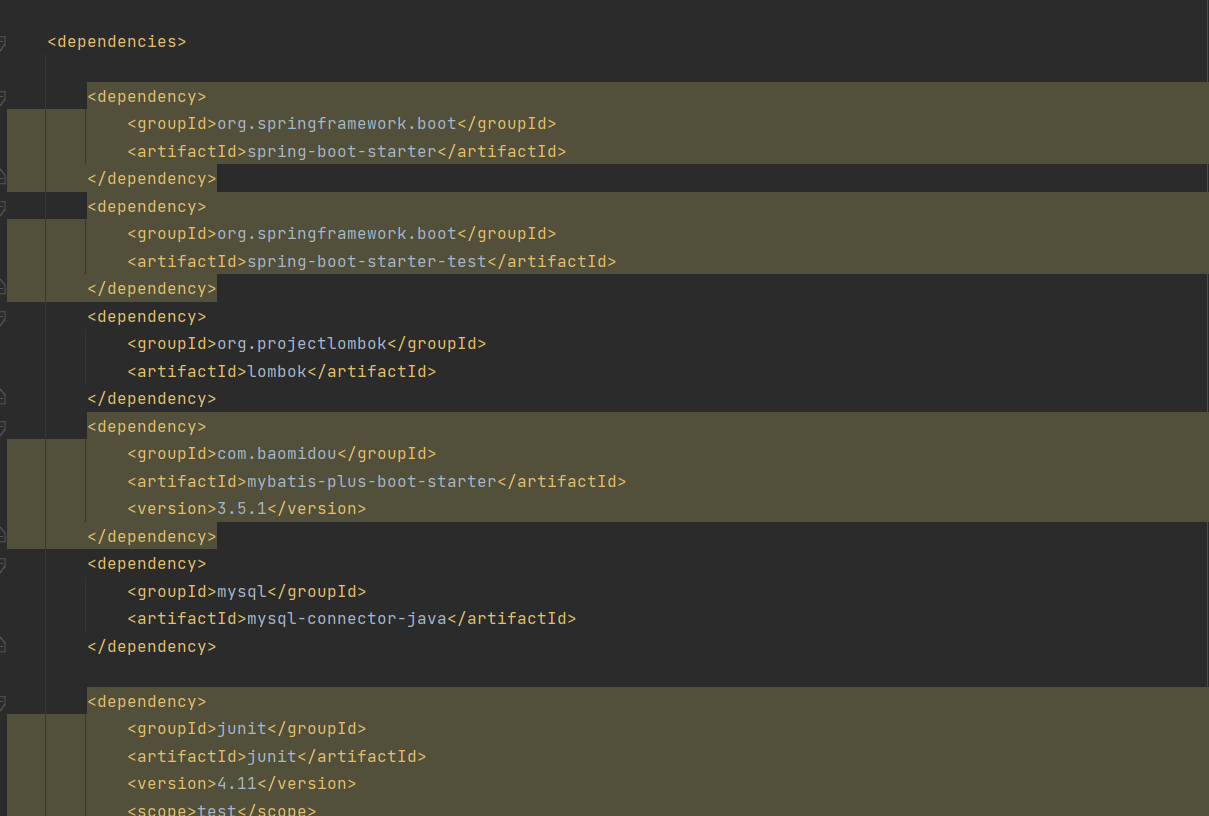
1.配置完成meven





2.成功添加mapper包





**叶翔昊**

8·20实训日志

总体情况：任务基本完成，代码编写成功。

代码：

DbMapper.java

package DatabaseProject.mapper;  
  
import DatabaseProject.pojo.Db;  
import DatabaseProject.pojo.User;  
import com.baomidou.mybatisplus.core.mapper.BaseMapper;  
import org.apache.ibatis.annotations.Mapper;  
  
@Mapper  
public interface DbMapper extends BaseMapper<Db> {}

UserMapper.java

package DatabaseProject.mapper;  
  
import DatabaseProject.pojo.User;  
import com.baomidou.mybatisplus.core.mapper.BaseMapper;  
import org.apache.ibatis.annotations.Mapper;  
  
@Mapper  
public interface UserMapper extends BaseMapper<User> {}

Db.java

package DatabaseProject.pojo;  
  
  
import com.baomidou.mybatisplus.annotation.TableName;  
import lombok.Data;  
  
@Data  
@TableName("test")  
public class Db {  
 private int id;  
 private String name;  
}

User.java

package DatabaseProject.pojo;  
  
import com.baomidou.mybatisplus.annotation.TableName;  
import lombok.Data;  
  
@Data  
public class User {  
 private int id;  
 private String username;  
}

MavenProjectApplication.java

package DatabaseProject;  
  
import org.springframework.boot.SpringApplication;  
import org.springframework.boot.autoconfigure.SpringBootApplication;  
  
@SpringBootApplication  
public class MavenProjectApplication {  
 public static void main(String[] args) {  
 SpringApplication.*run*(MavenProjectApplication.class,args);  
 }  
}

application.yml

spring:  
 datasource:  
 driver-class-name: com.mysql.cj.jdbc.Driver  
 url: jdbc:mysql://localhost:3306/db1?characterEncoding=utf-8  
 username: soulandwind  
 password: "1366"  
 type: com.zaxxer.hikari.HikariDataSource

UserMapperTest.java

package DatabaseProject.mapper;  
  
import DatabaseProject.pojo.Db;  
import DatabaseProject.pojo.User;  
import org.junit.jupiter.api.Test;  
import org.springframework.beans.factory.annotation.Autowired;  
import org.springframework.boot.test.context.SpringBootTest;  
  
import java.util.List;  
  
@SpringBootTest  
public class UserMapperTest {  
 @Autowired  
 public UserMapper userMapper;  
 @Autowired  
 public DbMapper dbMapper;  
 @Test  
 public void getUserInfo(){  
 List<User> users= userMapper.selectList(null);  
 System.*out*.println(users);  
 }  
 @Test  
 public void insertUserInfo(){  
 User user = new User();  
 user.setUsername("soulandwind");  
 user.setId(8);  
 int rows = userMapper.insert(user); // 调用 insert 方法  
 if (rows > 0) {  
 System.*out*.println("User inserted successfully.");  
 } else {  
 System.*out*.println("Failed to insert user.");  
 }  
 }  
// @Test  
// public void getDbInfo(){  
// List<Db> dbs=dbMapper.selectList(null);  
// System.out.println(dbs);  
// }  
}

Pom.xml

<?xml version="1.0" encoding="UTF-8"?>  
<project xmlns="http://maven.apache.org/POM/4.0.0"  
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"  
 xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">  
 <modelVersion>4.0.0</modelVersion>  
  
 <groupId>org.example</groupId>  
 <artifactId>mavenProject</artifactId>  
 <version>1.0-SNAPSHOT</version>  
 <parent>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-parent</artifactId>  
 <version>2.6.3</version>  
 </parent>  
  
 <dependencies>  
  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter</artifactId>  
 </dependency>  
  
 <dependency>  
 <groupId>org.springframework.boot</groupId>  
 <artifactId>spring-boot-starter-test</artifactId>  
 </dependency>  
  
 <dependency>  
 <groupId>org.projectlombok</groupId>  
 <artifactId>lombok</artifactId>  
 </dependency>  
  
 <dependency>  
 <groupId>com.baomidou</groupId>  
 <artifactId>mybatis-plus-boot-starter</artifactId>  
 <version>3.5.1</version>  
 </dependency>  
  
 <dependency>  
 <groupId>mysql</groupId>  
 <artifactId>mysql-connector-java</artifactId>  
 </dependency>  
  
 <dependency>  
 <groupId>junit</groupId>  
 <artifactId>junit</artifactId>  
 <version>4.11</version>  
 <scope>test</scope>  
 </dependency>  
 </dependencies>  
</project>

**兰天阳**

2024.8.20

