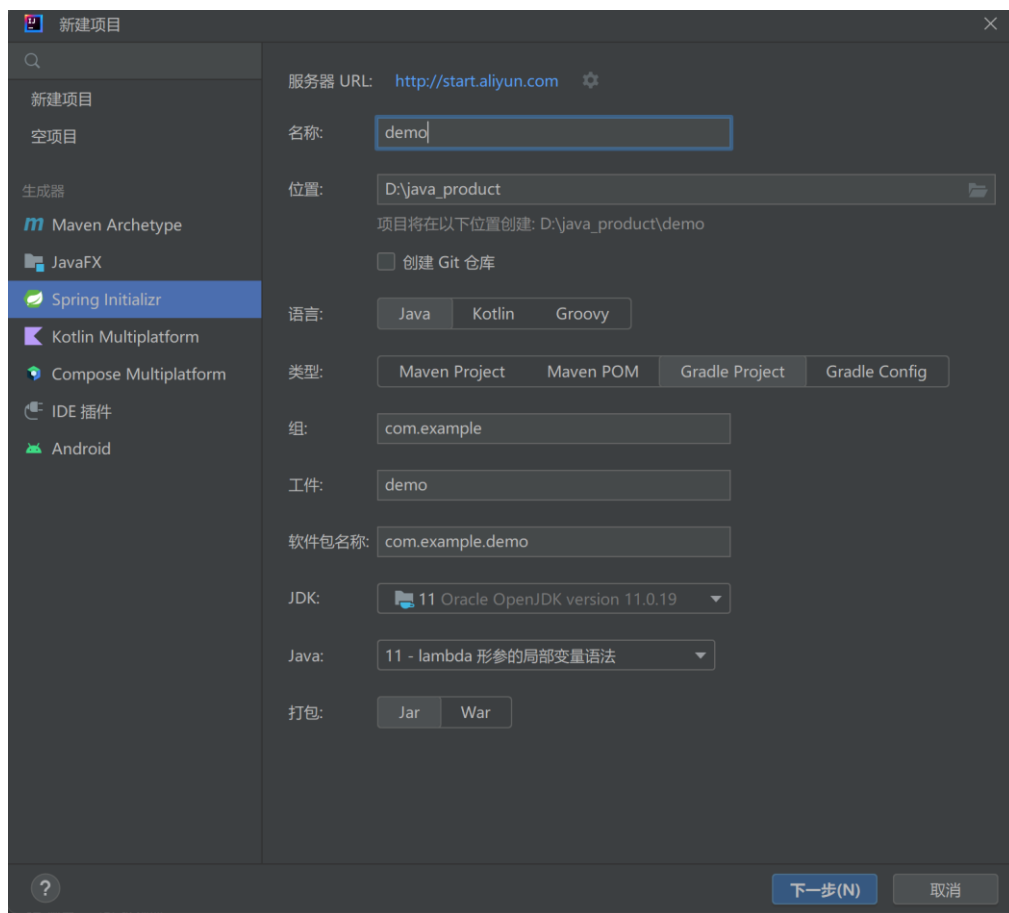


JAVA 学习笔记一——

springboot+mybatisplus 框架

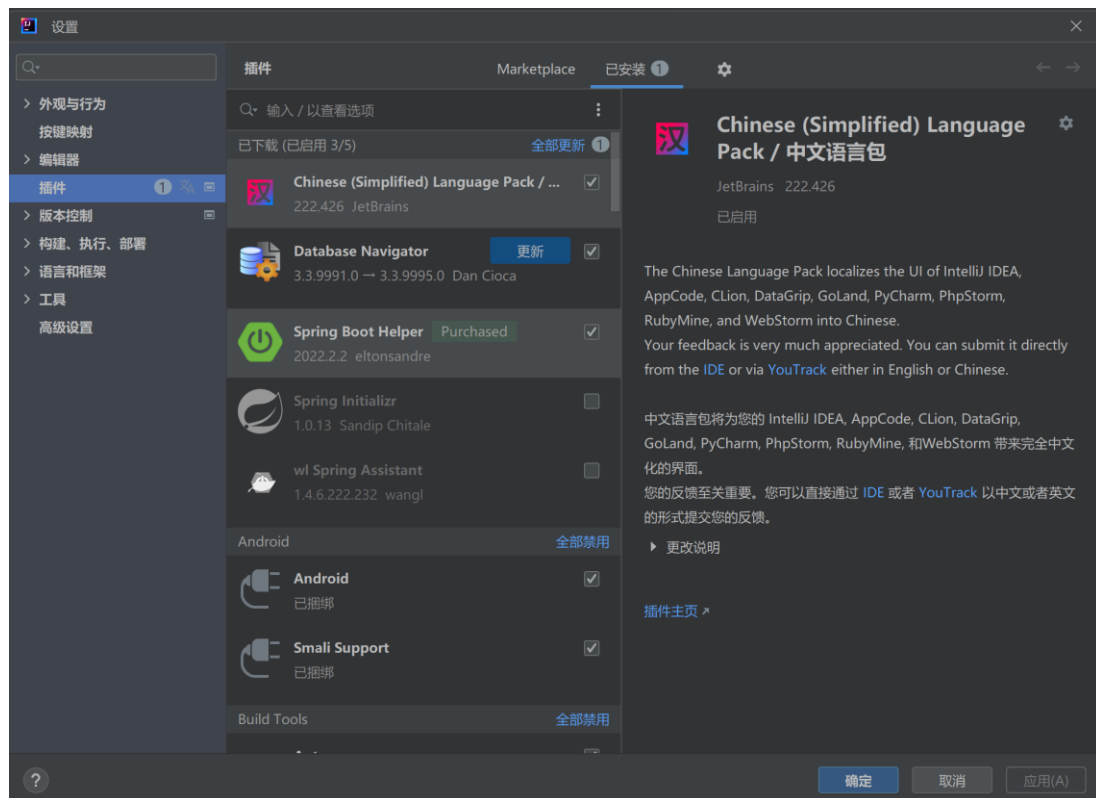
本文将利用 springboot+mybatisplus 框架, 实现 JAVA 对于 mysql 数据库的增查服务。

首先是环境的配置, 打开 IDEA, 笔者使用的是 IDEA2022 社区版, 选择 Spring Initializr, 尤其是这里的 JDK 版本, 推荐选择 JDK11。服务器 URL 推荐使用国内阿里云镜像, 国外的下载链接非常的慢。

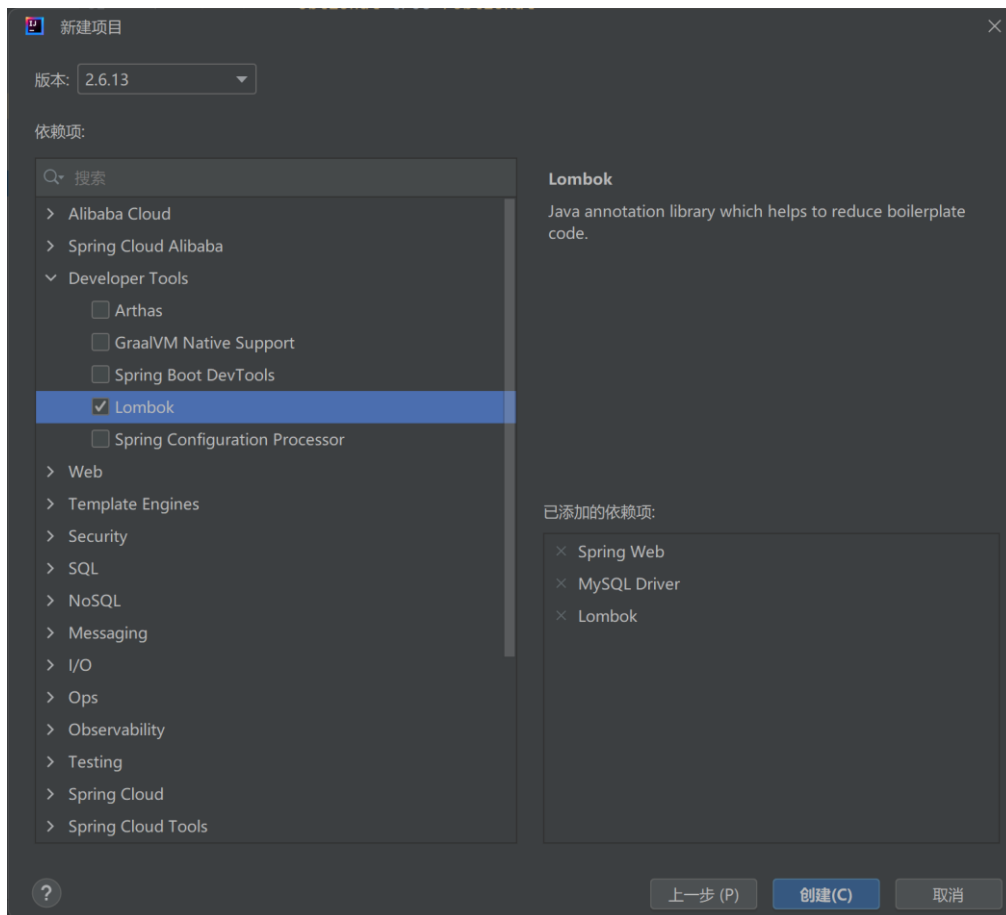


如果找不到 Spring Initializer 的话，需要在文件-设置-插件里面搜索，

然后打勾激活插件



选择需要加载的工具时，只需要选择如下图的 3 个即可：



由于本项目要用到 mybatisplus 框架，所以需要手动在 pop.xml 插入以下依赖，然后刷新一下即可。Maven 将自动下载依赖文件。

```
<dependency>
  <groupId>com.baomidou</groupId>
  <artifactId>mybatis-plus-boot-starter</artifactId>
  <version>3.5.3.1</version>
</dependency>
```

至此，springboot+mybatisplus 的框架已经搭建完成，接下来实现 JAVA 与 mysql 数据库的互通过程。

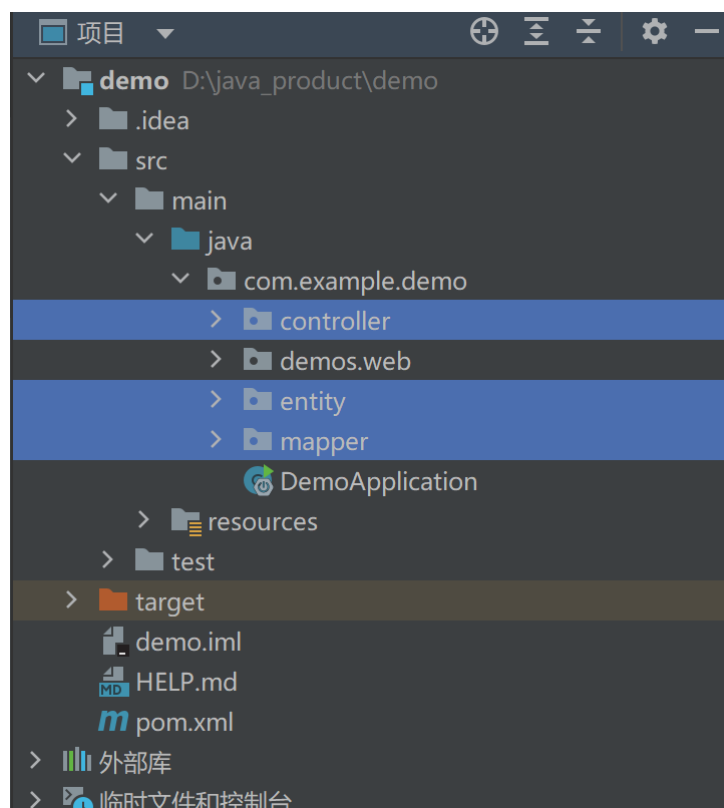
首先在 resources 文件夹下方找到 application.properties 文件，将其改名为 application.yml，然后在里面插入如下内容：

```

server:
  port: 8081    #连接开放的端口号
  session:
    timeout: 10    #连接超时时间
  tomcat:
    uri-encoding: UTF-8    #编码格式
spring:
  datasource:
    driver-class-name: com.mysql.cj.jdbc.Driver    #mysql8.0 及以上填这个，不需要改动
    password: xxx    #数据库密码
    url: jdbc:mysql://localhost:3306/test    #数据库的ip地址，'/'后面填写链接的database名称
    username: xxx    #数据库账号

```

然后需要在起始文件夹下方新建三个文件夹，存放三个层，分别是 controller, entity 和 mapper 层。



接着先来配置 mapper 层，在 mapper 下方创建接口 usermapper:



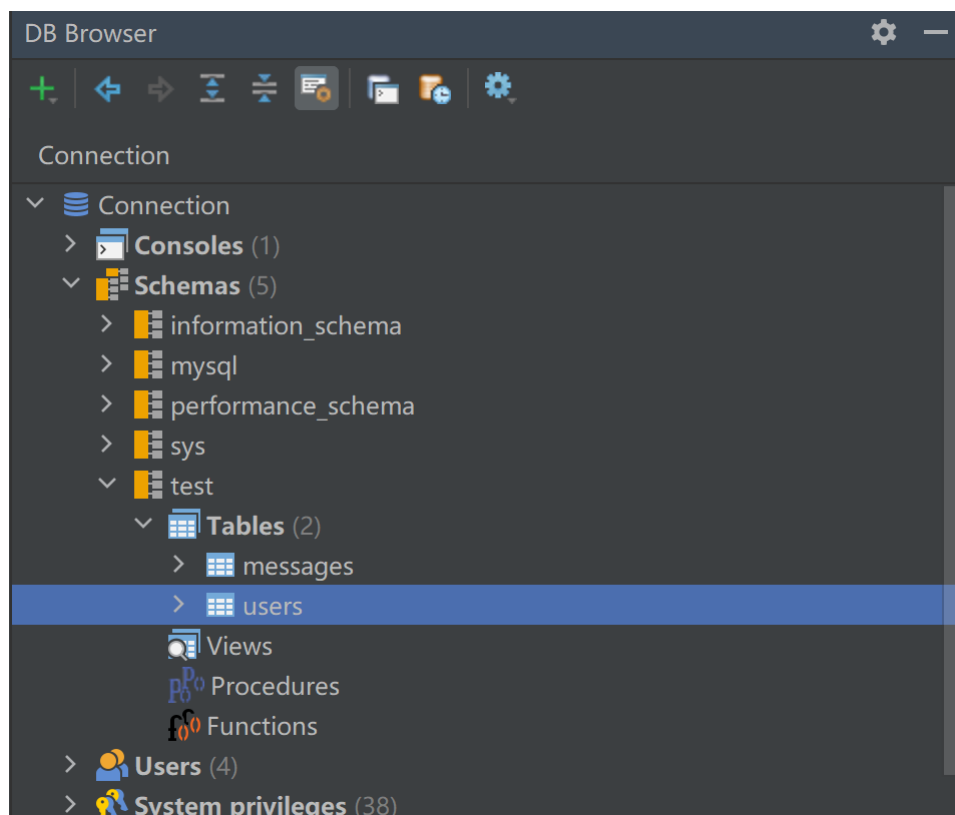
放入以下代码即可：

```
package com.example.demo.mapper;  
  
import com.baomidou.mybatisplus.core.mapper.BaseMapper;  
import com.example.demo.entity.users;  
  
public interface usersmapper extends BaseMapper<users>{  
  
}  

```

切记，这里的 extends BaseMapper<>，中括号内部的内容需要放置数据库内的

表名，笔者这里 mysql 数据库的表名称为 users，所以填入内容为 users。



然后配置 entity 层，在 entity 文件夹下方创建 users 类



填入以下内容：

```
package com.example.demo.entity;  
  
import lombok.AllArgsConstructor;  
import lombok.Data;  
import lombok.NoArgsConstructor;  
  
@Data  
@AllArgsConstructor  
@NoArgsConstructor  
public class users {  
    private String username;  
    private String password;  
}
```

注意，这里 users 类里面添加的变量应该与表内部的变量类型相同，我的表内部

只有 username 和 password 两个变量，存储的类型都为 String 类型，所以代码

如上

	id	username	password
•	NULL	NULL	NULL

然后在 controller 文件夹下新建 userscontroller 类，增删改查的所有功能都

在此文件中实现：

好比一个简单的插入语句，就可以这样写：

```
package com.example.demo.controller;

import com.baomidou.mybatisplus.core.conditions.query.QueryWrapper;
import com.example.demo.entity.users;
import com.example.demo.mapper.usersmapper;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

import java.util.List;

@RestController
public class userscontroller {

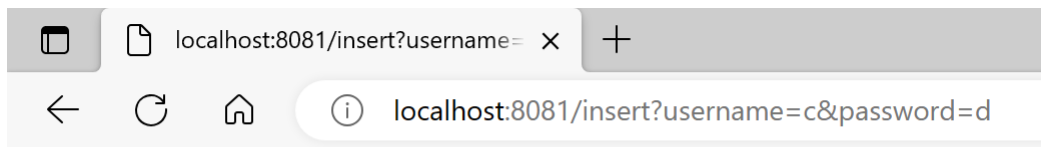
    @Autowired
    usersmapper usermapper;

    @RequestMapping("/insert") //插入数据的接口
    public String insert(String username,String password)
    {
        QueryWrapper queryWrapper1 = new QueryWrapper();
        queryWrapper1.eq("username", username);

        List<String> list = usermapper.selectList(queryWrapper1); //相当于
        于 select * from users where username = username
        if (list.size()>0) //判断数据库是否已经存在该数据
        {
            return "insert repeat";
        }
        else//不存在该数据，则可以进行插入操作
        {
            return usermapper.insert(new users(username, password)) >
0 ? "insert success" : "insert fail";
        }
    }

}
```

然后运行文件，就可以访问接口，完成对于数据库的插入数据操作



insert success

数据库内部也确实收到了消息：

