

TkMapper

1. TkMapper简介

基于MyBatis提供了很多第三方插件，这些插件通常可以完成数据操作的方法的封装，数据库逆向生成等等。

- MyBatisPlus
- tkMapper

tkMapper就是一个MyBatis插件，是在MyBatis的基础上提供了很多工具，让开发变得简单，提高开发效率。

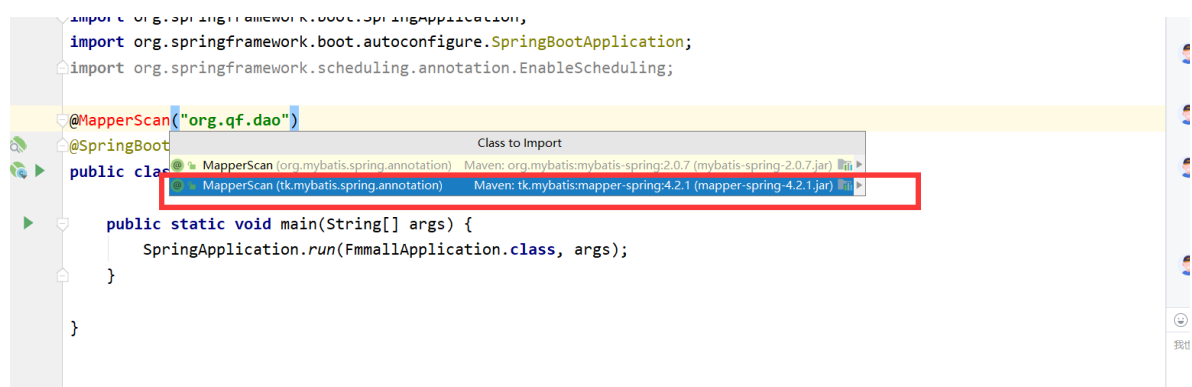
- 提供了针对单表通用的数据库操作方法
- 逆向工程

2. TkMapper整合Springboot

2.1 添加依赖

```
<dependency>
    <groupId>tk.mybatis</groupId>
    <artifactId>mapper-spring-boot-starter</artifactId>
    <version>4.2.1</version>
</dependency>
```

2.2 修改启动类的@MapperScan注解包



2.3 tkMapper的使用

1. 创建dao接口

tkMapper已经完成了对单表的通用操作的封装，封装在Mapper接口和MySqlMapper接口，因此如果我们要完成对单表的操作，只需要自定义Dao接口继承Mapper和MySqlMapper接口

```
7
8 @Component
9 public interface UsersMapper extends Mapper<Users>, MySqlMapper<Users> {
10
11     /**
12      * 注册功能
13      * @param users
```

对于单表来说，接口无需定义，那么mapper.xml无需使用

2. 业务层的实现

```
26 //注册
27 @Override
28 public ResultVO userResgit(String name, String pwd) {
29     //1. 根据用户查询，这个用户是否已经注册
30     Example example=new Example(Users.class);
31     Example.Criteria criteria=example.createCriteria();
32     criteria.andEqualTo( property: "username",name);
33     List<Users> users = usersMapper.selectByExample(example); //通过用户名进行条件查询的结果
34     //2. 如果没有注册则进行保存操作
35     if(users.size()==0){
36         //密码加密 md5
37         String md5Pwd=MD5Utils.md5(pwd);
38         Users user=new Users();
39         user.setUsername(name);
40         user.setPassword(md5Pwd);
```