1. 什么是Mybatis-plus

是一个方便的操作数据库功能的代码工具。有一个叫mybatis的技术框架因为经常要自己配置各种文件，进行很多次重复操作，使得常常会不小心出错，这时把经常重复的东西直接封装起来，甚至可以设置属性生成代码能让工作量减少很多，就在原来mybatis基础上整合诞生了一个工具，叫mybatis-plus。

1. 如何使用Mybatis-plus

1.导入依赖

<dependency>

<groupId>com.baomidou</groupId>

<artifactId>mybatis-plus-boot-starter</artifactId>

<version>3.4.1</version>

</dependency>

2.数据库链接配置

spring.datasource.username=root

spring.datasource.password=1234

spring.datasource.url=jdbc:mysql://localhost:3306/mjava9?useSSL=false&useUnicode=true&characterEncoding=utf-8&serverTimezone=GMT%2B8

spring.datasource.driver-class-name=com.mysql.jdbc.Driver

3.在com.huarui下简历需要的表的实体类，添加get set tostring方法

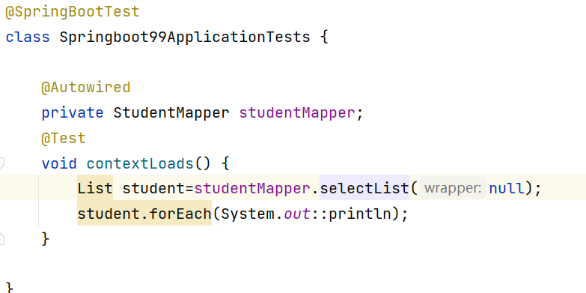
4. 在com.huarui简历Mapper层 如图所示 新建接口



5.在spring的启动类添加注解扫描包 @MapperScan("com.huarui.mapper")

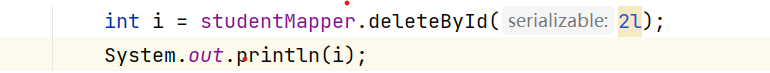
6.然后去测试

1.查询所有



2.删除

1.根据id删除



2.批量删除

int result = stustudentMapper.deleteBatchIds(Arrays.asList(8, 9, 10));  
System.*out*.println(result);

3.简单条件查询删除

HashMap<String, Object> map = new HashMap<>();  
map.put("name", "Helen");  
map.put("age", 18);  
int result = studentMapper.deleteByMap(map);  
System.*out*.println(result);

3.添加

数据库字段一定要自增！

在实体类的id上面加上

@TableId(type = IdType.AUTO)

可以自动生成id

int result = studentMapper.insert(user);

4.修改

Student student=new Student();  
student.setSid(5);  
student.setAge(38);  
int i=studentMapper.updateById(student);

使用mybatis生成mapper servicer conntroller

生成完之后在conntroller把sericerimp注入进来

写自己的sql需要在pom中导入

<resources>  
 <resource>  
 <directory>src/main/java</directory>  
 <includes>  
 <include>\*\*/\*.xml</include>  
 </includes>  
 </resource>  
</resources>

在中添加mybatis-plus.mapper-locations=classpath:com/huarui/mapper/xml/\*.xml

Wrapper ： 条件构造抽象类，最顶端父类，抽象类中提供4个方法西面贴源码展示

AbstractWrapper ： 用于查询条件封装，生成 sql 的 where 条件

AbstractLambdaWrapper ： Lambda 语法使用 Wrapper统一处理解析 lambda 获取 column。

LambdaQueryWrapper ：看名称也能明白就是用于Lambda语法使用的查询Wrapper

LambdaUpdateWrapper ： Lambda 更新封装Wrapper

QueryWrapper ： Entity 对象封装操作类，不是用lambda语法

UpdateWrapper ： Update 条件封装，用于Entity对象更新操作