**CRUD**

在架设好utils,pojo,resources里的mybatis-config.xml里以后，我们只需要在Mapper里添加接口，UserMapper.xml实装接口就行，使用就是测试类，只要改这三个

三步：

Mapper添加对应接口方法，Select多条需要return List,一条就return 单个，其他的return int

Mapper.xml实现·接口，先要用namespace找到对应interface,

**Select:**

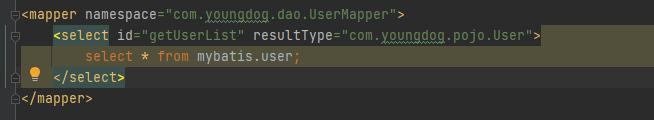
id：对应接口名字，

resultType：对应实体类

parameterType: 如果是限定查询可能需要参数，参数的类

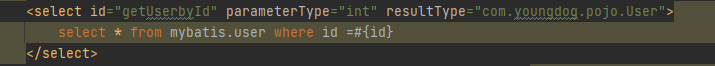
return多个就是List，return一个条目就是单个







他通过id select到，因此只return一个条目，所以是User



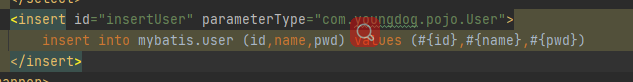
parameter Type就是跟着接口的parameter一路写下去

引用接口的parameter就是#{},

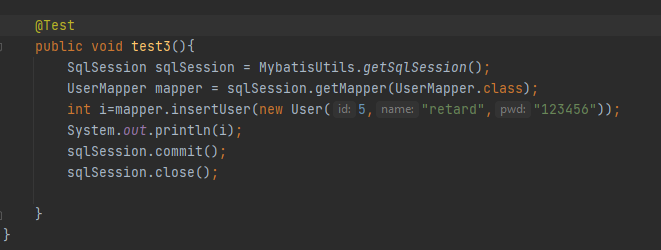
**Insert**

增删改 有一个特殊点，他们return的都是int,如果return 1，代表这个操作成功没有语法上的错误，如果return 0代表语法有问题

 return类型为int



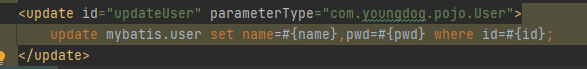
Mapper接口是public的，所以可以直接引用user的id，name,pwd//引用object的属性可以直接提取出来

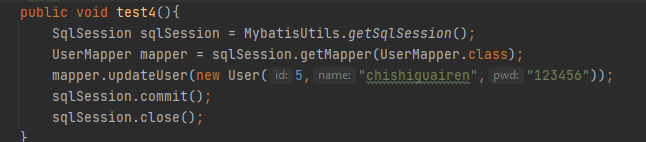


增删改还必须要commit

**Update**

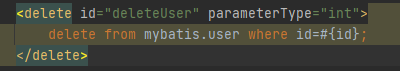
****

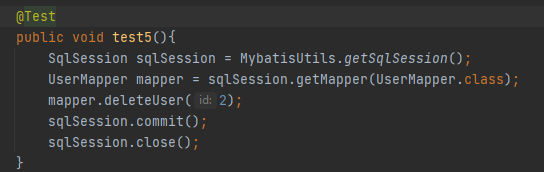
****

****

**Delete**

****

****

****

**Map**

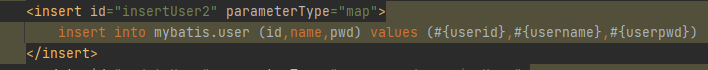
**两种用法**

1.有的时候pojo类/数据库特长特复杂，我们可以用Map作为参数，

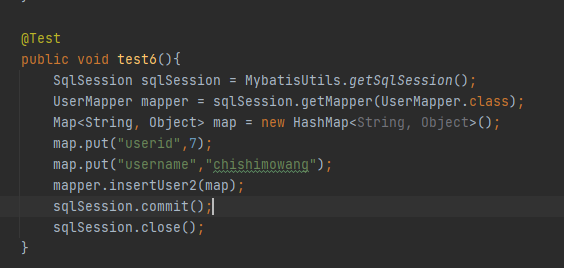


key是String,用来手写字段

Object是因为String,Integer都能作为Object

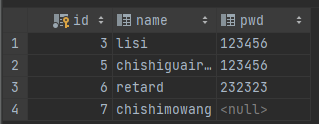


userid,username,userpwd就是 key



可以看出我们并没有put pwd，

也就是说他把pwd设置成了null,如果实体类/数据库字段参数过多，这样可以少填很多可以为Null或者默认的数据



2.比如说select,我们同时搜索两个int类型的参数，这时可以使用map,



例如这里getUserbyId, 如果id是可以重复的，我们需要另外一个int子id才能确认，这时可以用map

总结：

Map传递参数，直接在sql中取出key即可！ 【parameter=“map”】

对象传递参数，直接在sql中取出对象的属性即可！ 【parameter=“Object”】 //不用管private

只有一个基本类型参数的情况下，可以直接在sql中取到

多个参数用Map , **或者注解！**