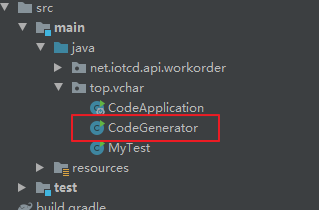
# 快速上手myBatis plus

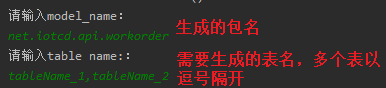
myBatis plus是对myBatis的增强，使用myBatis plus可以快速开发单表的CUID，并且通过与Lambda表达式的结合，基本可以实现单表操作的所有而不需要写sql或配置xml文件，如果遇到联表查询，则还是需要写Sql语句

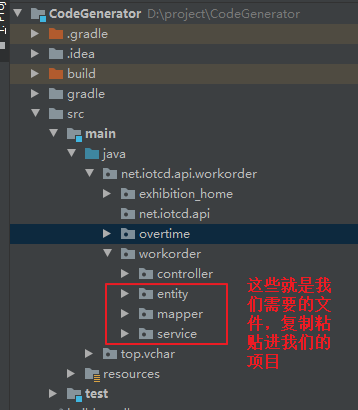
## 生成Entity，Mapper和Service文件

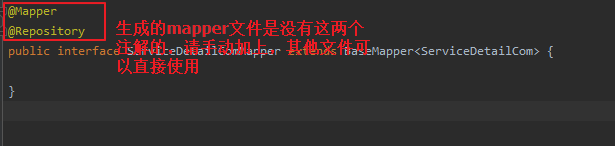
* + myBatis plus提供了自动生成这些文件的方法，可以不用再写这些文件
  + 找到CodeGenerator项目，进入CodeGenerator.java文件







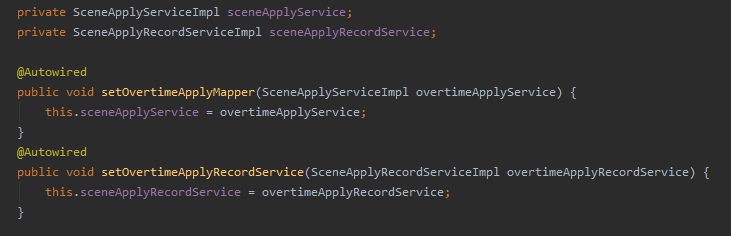




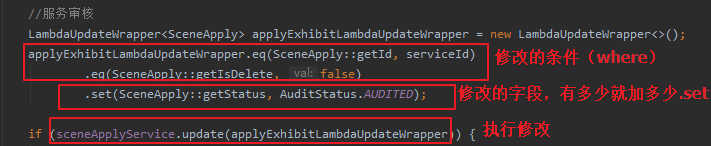
* 单表的增删查改

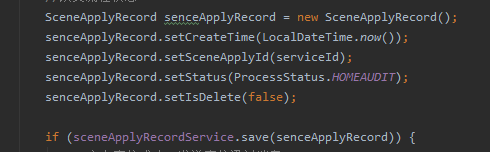
因为myBatis plus的封装，service也可以直接进行单表的增删查改，所以使用service和mapper看自己喜好，这里以mapper的为主

* + 依赖注入你需要的mapper

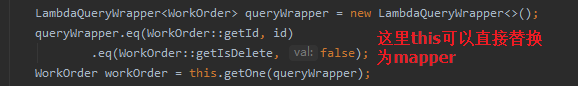


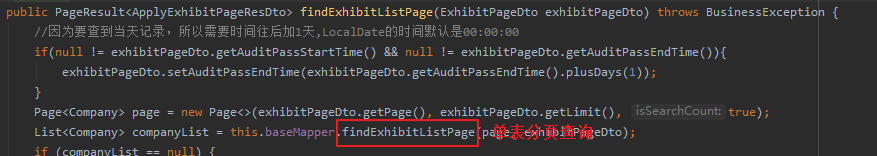
* + 单表增加，修改





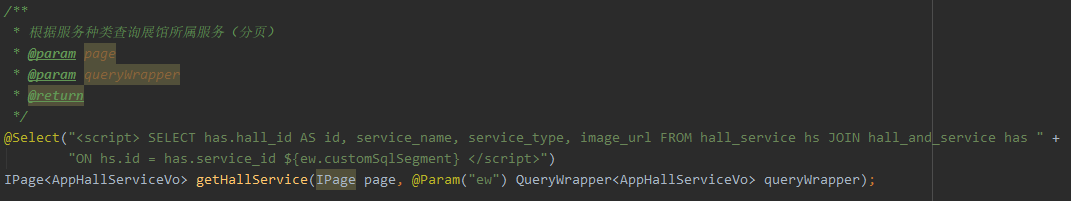
* + 单表查询，单表分页



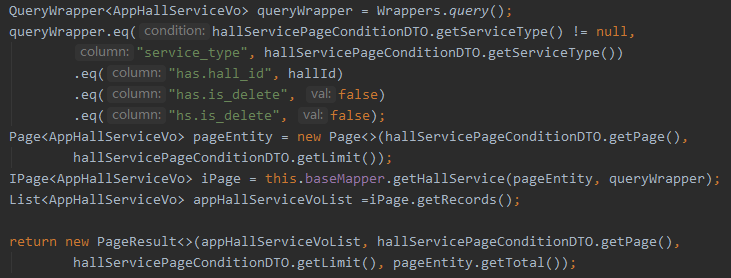




* + 联表分页
    - 编写mapper，在自己写的mapper方法中加入wapper参数，在sql语句中插入${ew.customSqlSegment}(这句话之后，不用再写后面的查询条件，mybatis-plus会根据wapper来进行条件的插入)

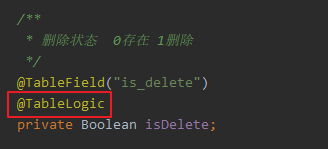


* + - 方法调用：首先，需要编写wapper信息来进行条件查询；其次编写Page信息；然后通过iPage信息得到返回列表；最后将列表信息封装城PageResult返回



* + 逻辑删除

项目中通常删除不会是真正的删除某条记录，而是以某个字段作为逻辑删除字段（比如is\_delete:true删除 false不删除），如果每次都在查询，加入未删除这个会很麻烦，所以可以使用mybatis-plus的自带注解（新增不起效果，还是需要手动新增或者给数据库设默认值）



* + 自动获取某些字段

之前项目的当前用户信息都是通过存入Header中的某个属性，假设我们要存用户-xx表的用户id信息，每次都必须从Header中取，比较麻烦。可以使用mybatis-plus的自带工具，这样加入一个注解就可以自动存入。比较适合一些可以自动设置字段的值

