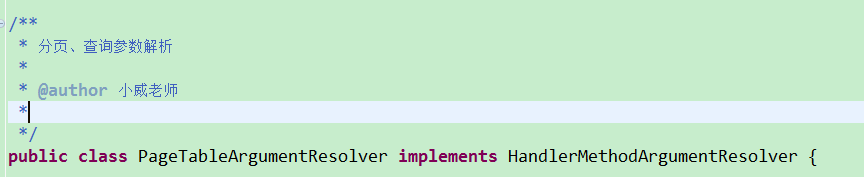
在该项目中，由于用的是mybatis，表字段名我们默认采用驼峰，这样就跟java中的成员变量保持一致了，省去了mybatis的转换。

有的同学项目中，表字段规则可能是下划线风格的。这样datatables自动排序时会出现问题。

原因为，datatables排序是将java的字段名传向后台的，我的解析类

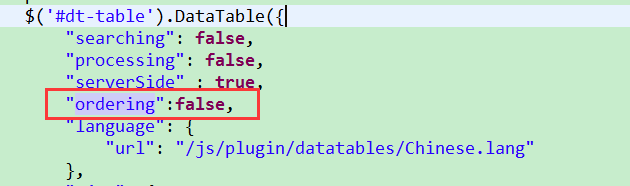


直接把java字段拼到order by 里的，这样如果您的表字段名是下划线风格的，sql肯定会报错的，如你本该是order by create\_time，程序拼成了order by createTime。

# 排序解决办法：

## 粗暴解决

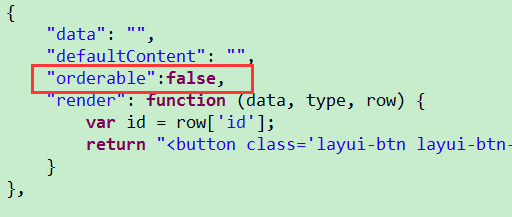
### 放弃全部点击表头排序



这样的话，整个表头都不会出现点击排序的图标

### 放弃单列排序

在不写"ordering":false的情况下



在单列加参数"orderable":false

这样datatables就不会传排序到后台了。

## 优雅解决

### 第一步

添加相应的mapper.xml文件，添加一个resultMap

如，我们做个简单例子，这里就用两个字段举例，这个mapper可以手写，也可以用mybatis的自动生成插件生成，不会的可以百度mybatis generator，百度了再不会用的，可放弃改行了，程序猿不适合你。

<mapper namespace=*"com.boot.security.server.dao.DictDao "* >

<resultMap id=*"BaseResultMap"* type=*"com.boot.security.server.model.Dict"* >

<result column=*"id"* property=*"id"* jdbcType=*"INTEGER"* />

<result column=*"create\_time"* property=*"createTime"* jdbcType=*"DATE"* />

</resultMap>

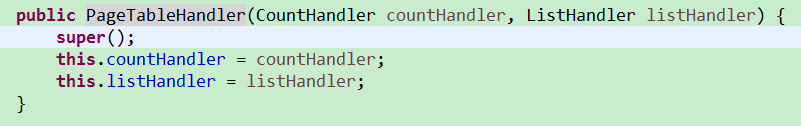
</mapper>

这样就有了java的成员变量和表字段的对应关系

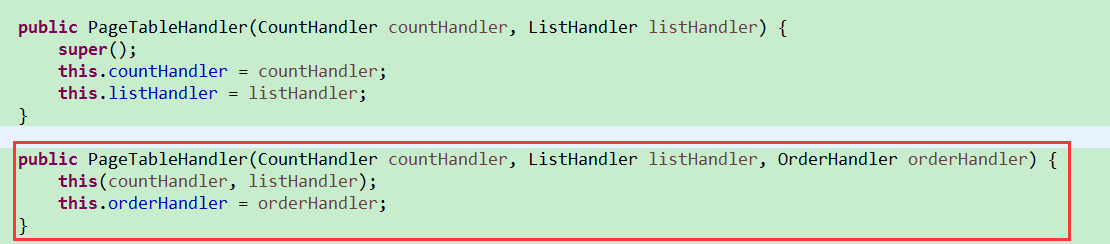
### 第二步

修改我们的

方法，我们默认用的是



，下面要改成



如

改成



其实就是将java的成员变量，根据mapper文件里的对应关系，替换成表字段名，