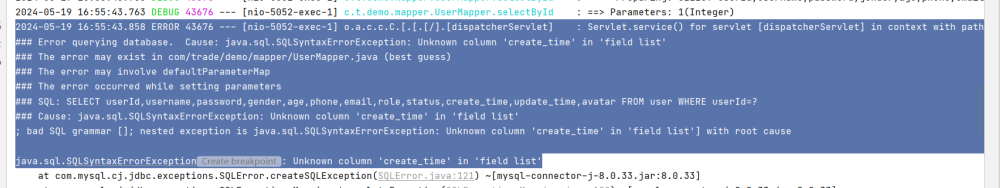
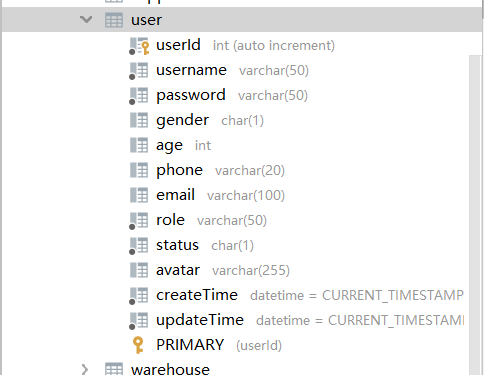
# Debug参考书



这个错误表明你的数据库中 **user** 表没有 **create\_time** 列，而实体类 **User** 和/或 MyBatis-Plus 查询试图访问该列。为了解决这个问题，需要确保数据库列名和实体类属性名之间的一致性。



我的user表是这样的,的确没有create\_time,出现错误的原因是,mybatis对实体类中属性中的大写字母对应数据库中的"\_小写字母". 例如, 我数据表中属性有一个为e\_mail,那么,java后端的实体类就应填写eMail, 这样,mybatis解析eMail才会对应e\_mail。

解决方法一：修改数据表结构，改为create\_time,update\_time

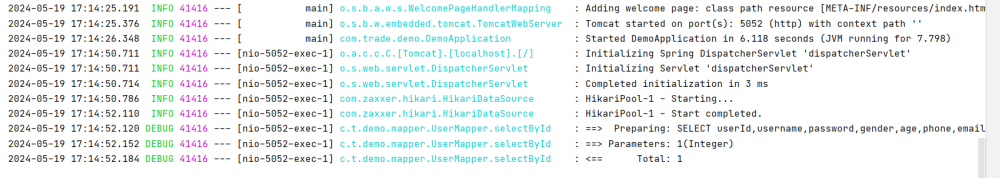
解决方法二:修改实体类和数据库表的映射关系

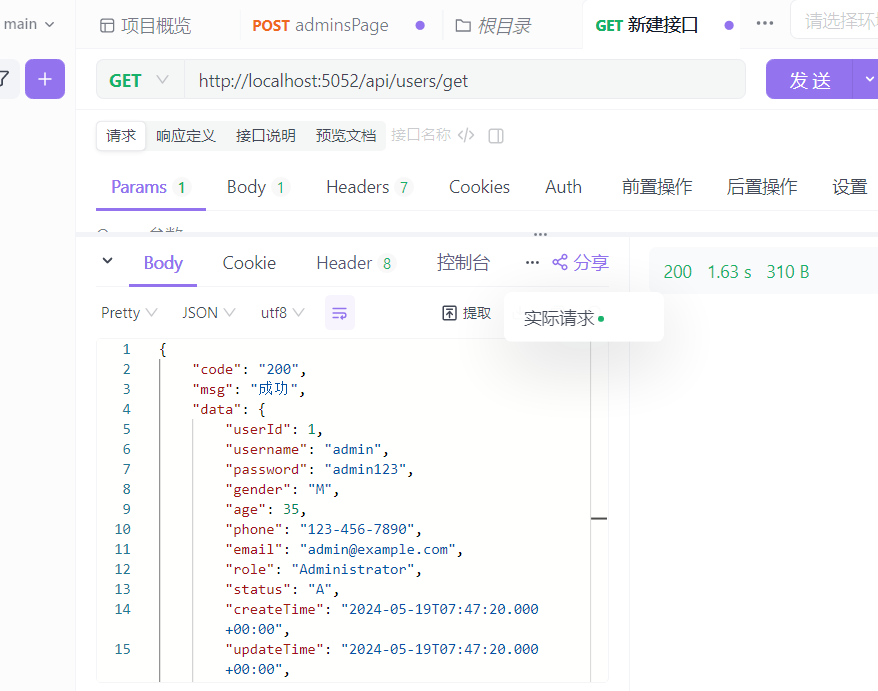
使用**@TableField** 注解来指定实体类字段与数据库表列之间的映射关系

@TableField("createTime")  
@ApiModelProperty(value = "创建时间")  
private Date createTime;  
@TableField("updateTime")  
@ApiModelProperty(value = "更新时间")  
private Date updateTime;

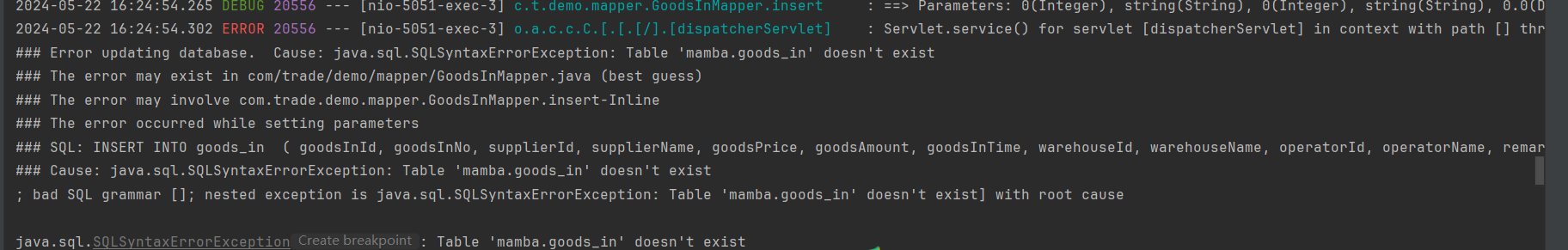
在 MyBatis-Plus 中，默认情况下，它会将实体类的驼峰命名法（CamelCase）转换为数据库表的下划线命名法（snake\_case）。因此，如果你的数据库表中的列名是 **create\_time**，而你的实体类属性名是 **createTime**，MyBatis-Plus 会自动进行映射。然而，如果数据库表中的列名不遵循这种命名规则（例如 **e\_mail**），那么需要明确地指定映射关系。

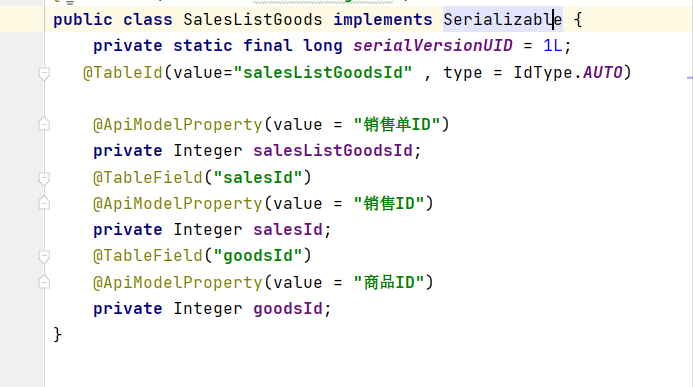
效果





发现也有表名的匹配的问题

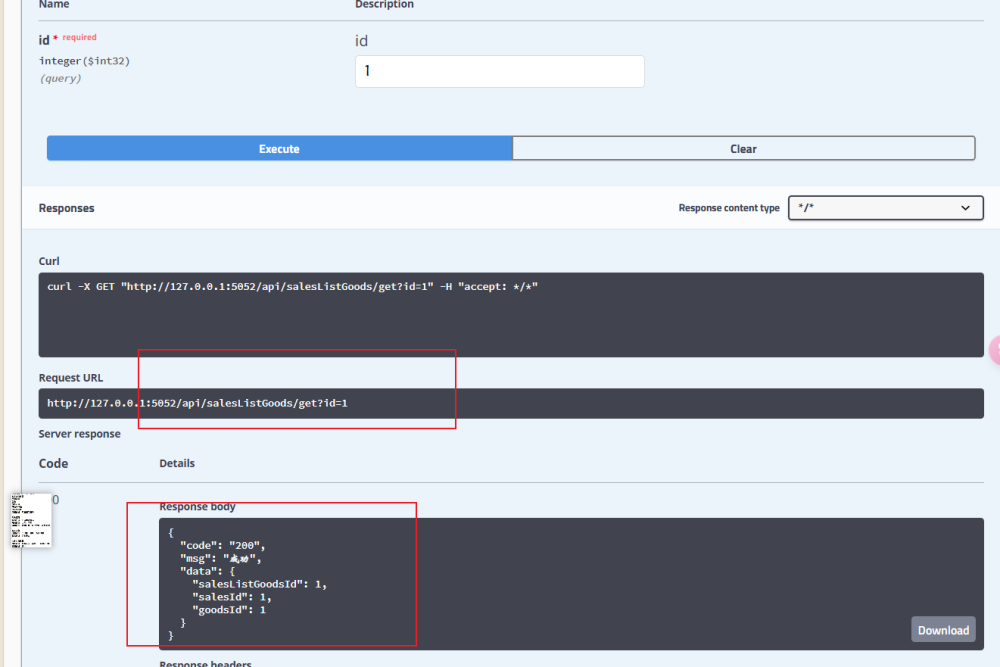


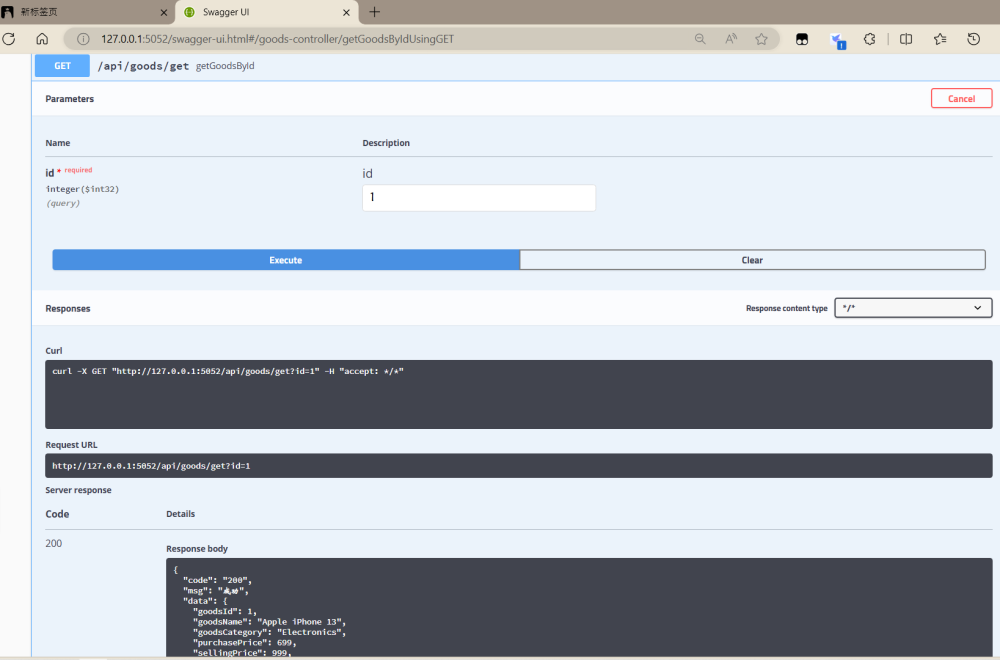


意外发现重修定义也有问题, 差点大改数据库,解决方案是给类加上表的注解,值为数据库中表的值



成功:

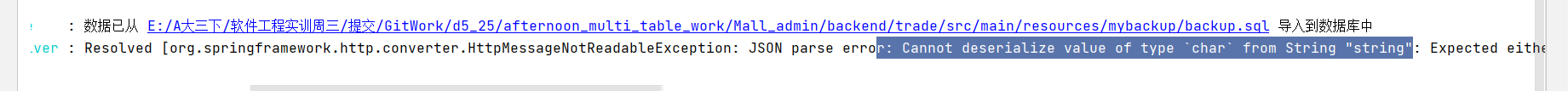


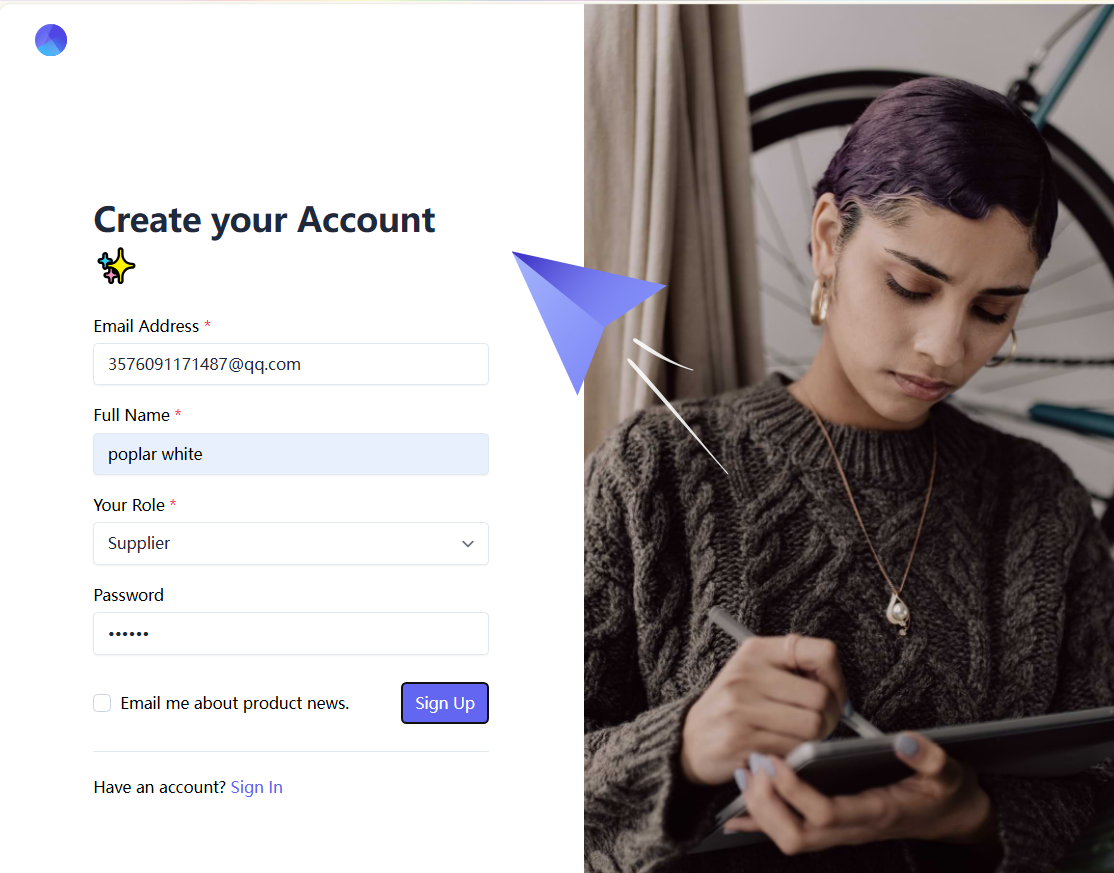
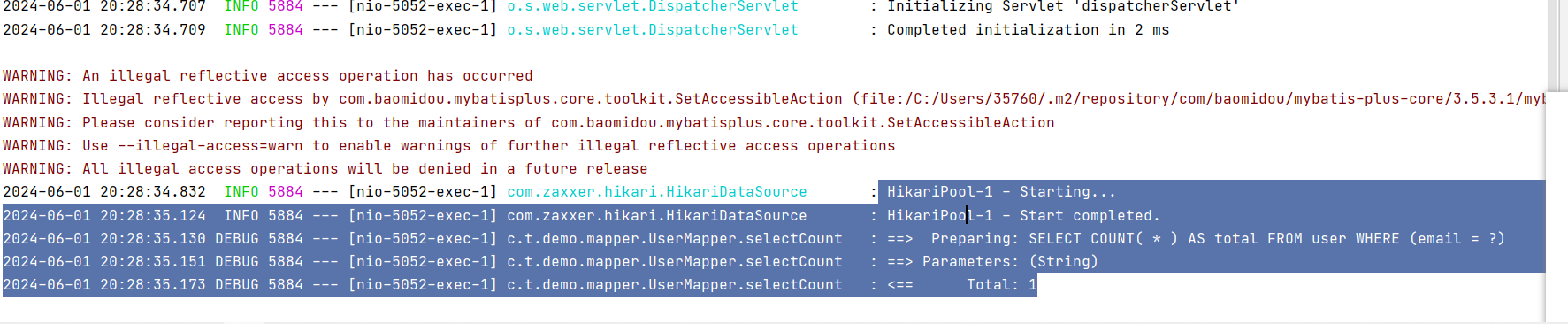


对于在Swagger UI进行测试时,尤其是进行插入,update操作时,配置的样例数据的json是将属性自动填满了的,但有的类型不匹配。如status是char类型的，如果直接执行这个样例代码，会失败并后端idea查看可以知道错误类型(swagger错误认为status是string的)



后端提示：





前端在传入参数时,似乎出现错误

2024-06-01 20:28:35.130 DEBUG 5884 --- [nio-5052-exec-1] c.t.demo.mapper.UserMapper.selectCount : ==> Preparing: SELECT COUNT( \* ) AS total FROM user WHERE (email = ?)

2024-06-01 20:28:35.151 DEBUG 5884 --- [nio-5052-exec-1] c.t.demo.mapper.UserMapper.selectCount : ==> Parameters**: (String)**

2024-06-01 20:28:35.173 DEBUG 5884 --- [nio-5052-exec-1] c.t.demo.mapper.UserMapper.selectCount : <== Total: 1

可见传参并没有传入输入的数据。

这是由于前端form表单没有绑定v-model进行信息的绑定,导致表单提交时能够正确获取数据

解决部分代码如下:

