

优点

- 项目完成度还是比较高的，可以看出师弟是有花费精力和时间完成的；
- 类设计方面，尽管类的数量比较少，但是做到了考核任务当中规定的大部分业务逻辑实现，可以的；（建议：可以的话建议再将类进行细分）
- 尽管这一个项目在每一次运行环节都只能选择一个身份，但是每一个身份所做到的业务都是十分全面的，这一部分说明师弟或师妹在两个身份之间分配时间还是比较充足的；

优点总结：可以看到师弟或师妹在这一次考核任务当中所付出的时间和精力，整个项目完成度很高；在正常的运行环节是可以运行下去的；也可以看到师弟或师妹在测试阶段做了很多debug调试；总之，整体呈现效果可以的；

缺点

- 类设计环节，类的命名有点小缺陷，在Java的命名规则下，通常要做到见名知义和驼峰原则和第一个字母大写；但是在师弟或师妹的项目当中，可能有部分类没有满足这一种情况；（**需要注意**）
- 没有注册登录环节，是老板还是顾客都是使用该程序的用户的一个选择，不需要任何的校验规则来确保这个用户选择的业务操作符合其身份；
 - 解决方案：使用可以区分老板和顾客的标识来进行操作；
- 在顾客订房环节没有说明用户是手动输入房间类型还是输入1、2、3来选择房间类型，在提示这一方面有一点小缺点；
- 充值环节如果出现了大面额的充值金额，会出现这个东西

您当前余额为200.0

请输入充值金额：

◦

100000000.11

充值成功，您现在余额为：1.0000020011E8

- 注意：这个E8是面向计算机专业的用户可能他认识，但是如果是其他专业的用户那就可能一脸懵逼，你要记得，你做出的程序是面向客户的，而不是面向程序员的；（除非是一些面向计算机专业的服务软件等）
- 当顾客一次性开了多个房间，在查看房号的时候需要给出这多间房间的唯一性标识（房间号），不然用户可能不清楚哪一间是哪一间
- 每一次程序在运行的时候都只能选择一个角色来处理业务，不能看出这两个身份之间的关联性，可以适当增加一个切换账号的操作；

- ...

缺点总结：大部分都是业务运行过程的一些逻辑错误所造成的bug，可能是因为缺少开发房间订购系统的经验；提示语给出的比较少，用户可能在第一次使用这一个程序的时候造成该系统的部分逻辑出错。