**测试案例**

本项目可以通过**postman**测试，也可通过**浏览器输入url在控制台观察**来测试，也可以实际**上手去玩**进行测试（postman/浏览器均全用**GET**请求（url为127.0.0.1:8000/test/），方便；**实际项目GET**请求仅用来获取页面，具体操作（需要数据库进行增删改）都是通过**POST**来发送（url为上述url去掉test））

项目由Django2.2框架搭建而成，因此需要Python环境，并下载Django2.2

pip install Django==2.2

然后在TreasureHuntGame文件夹下，使用命令行执行

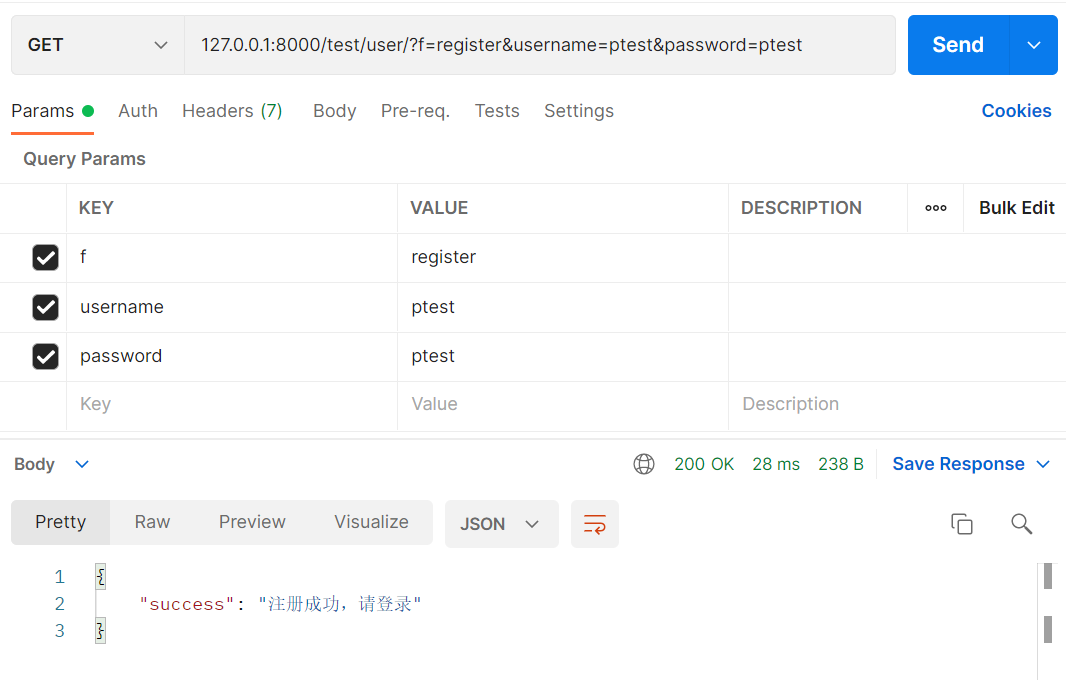
python manage.py runserver

即可在127.0.0.1:8000启动项目服务

**测试案例这里以postman为例**

注册/登录/退出登陆

注册成功



数据库中记录：

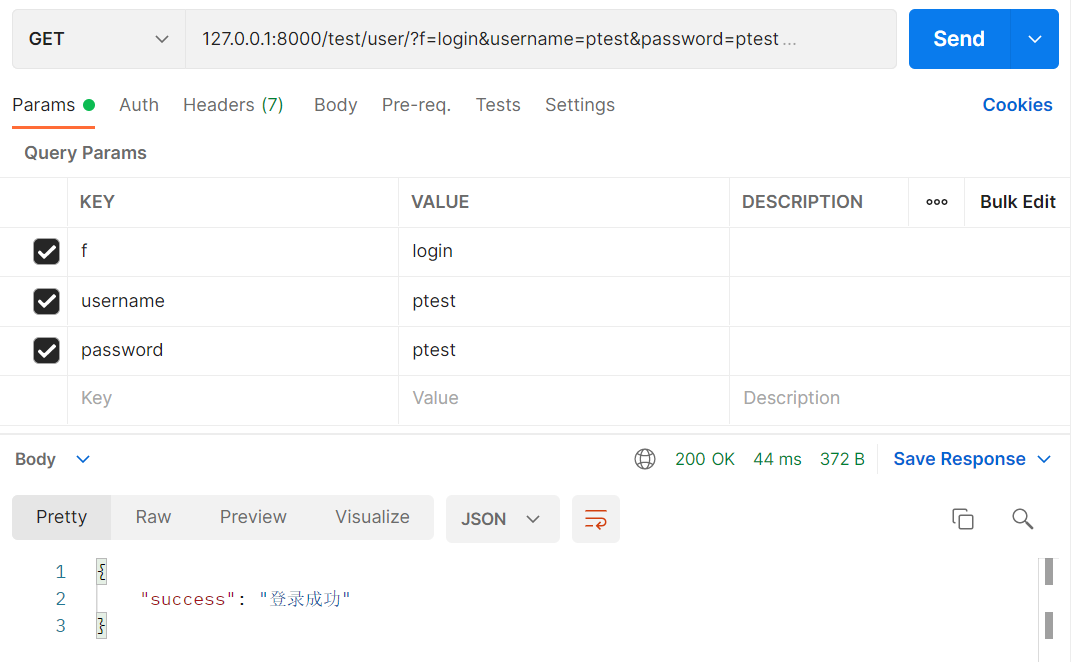


注册失败

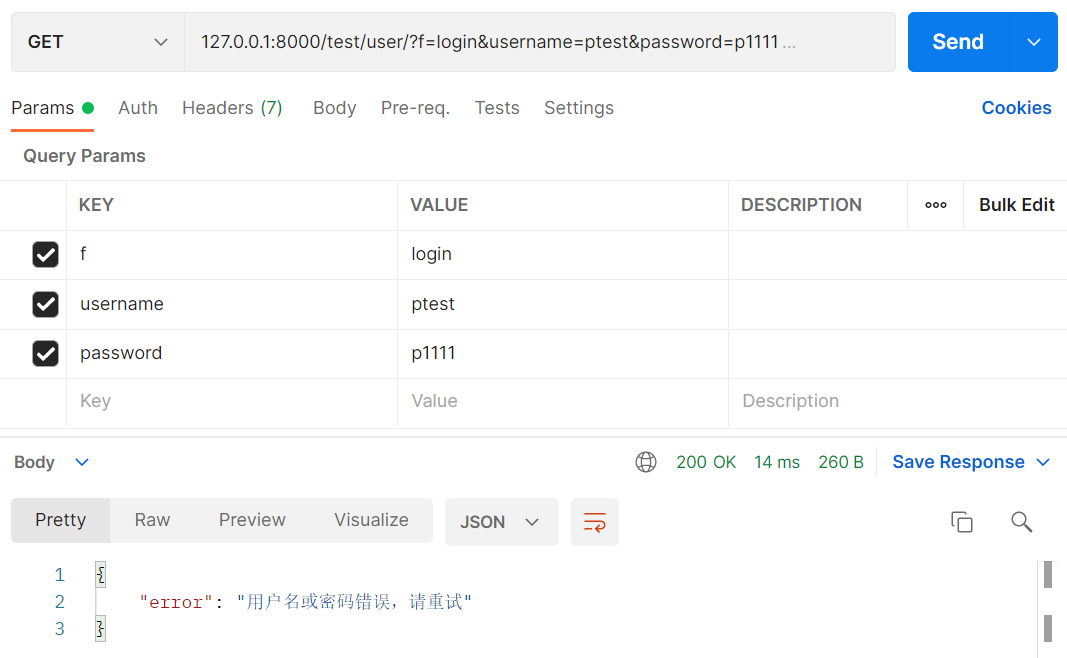


登录

登录成功

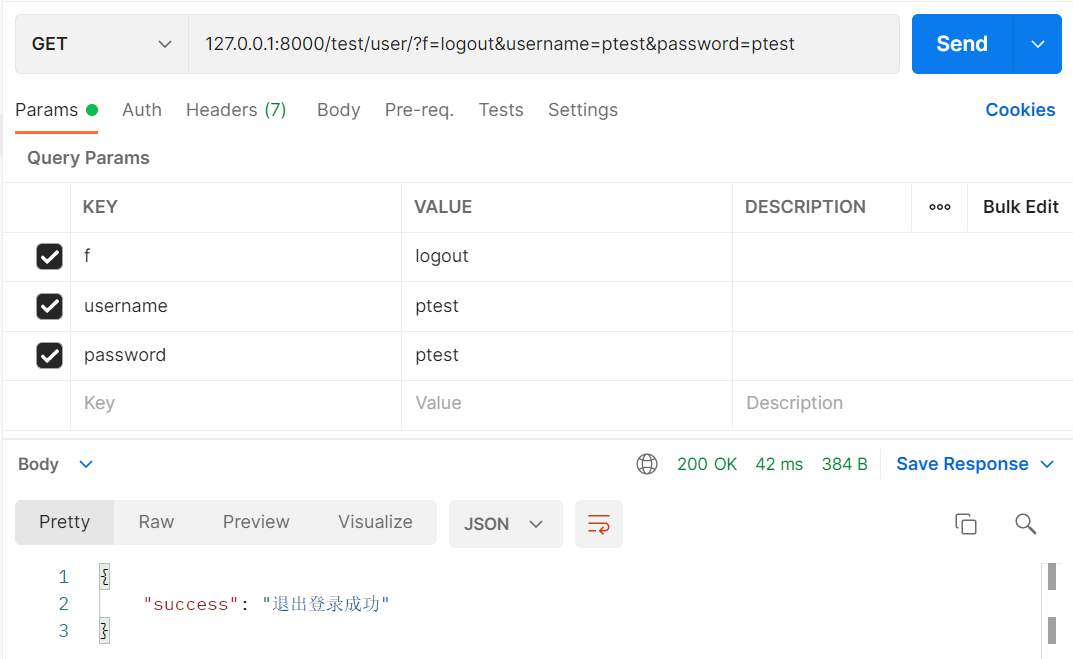


登录失败

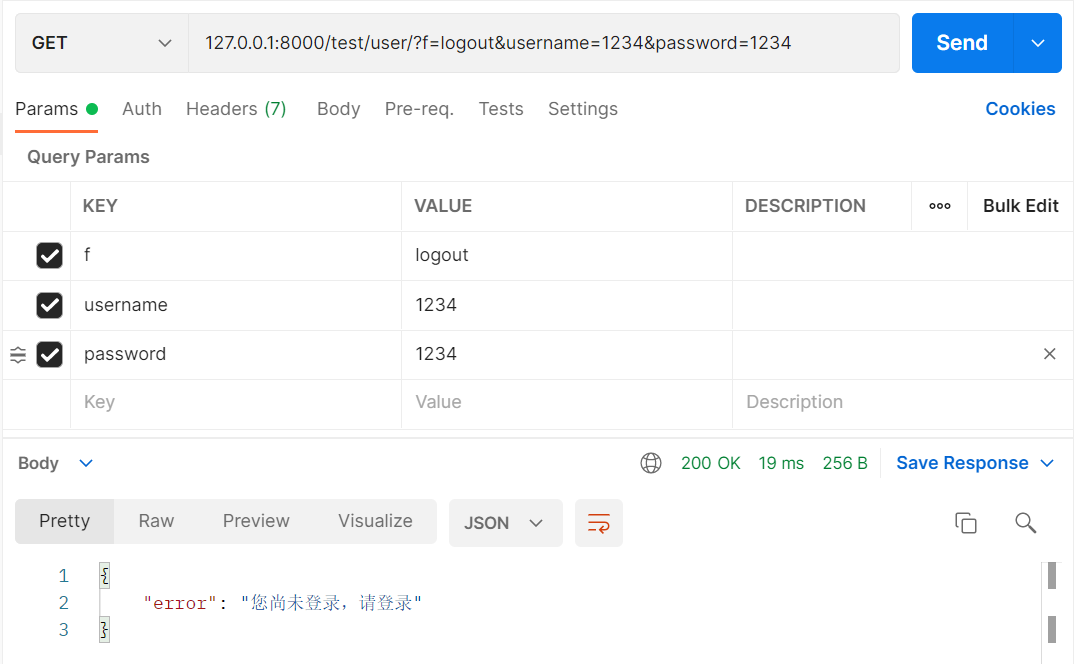


退出登录（这里参数无意义，仅删除数据库所存浏览器session）

退出登录成功

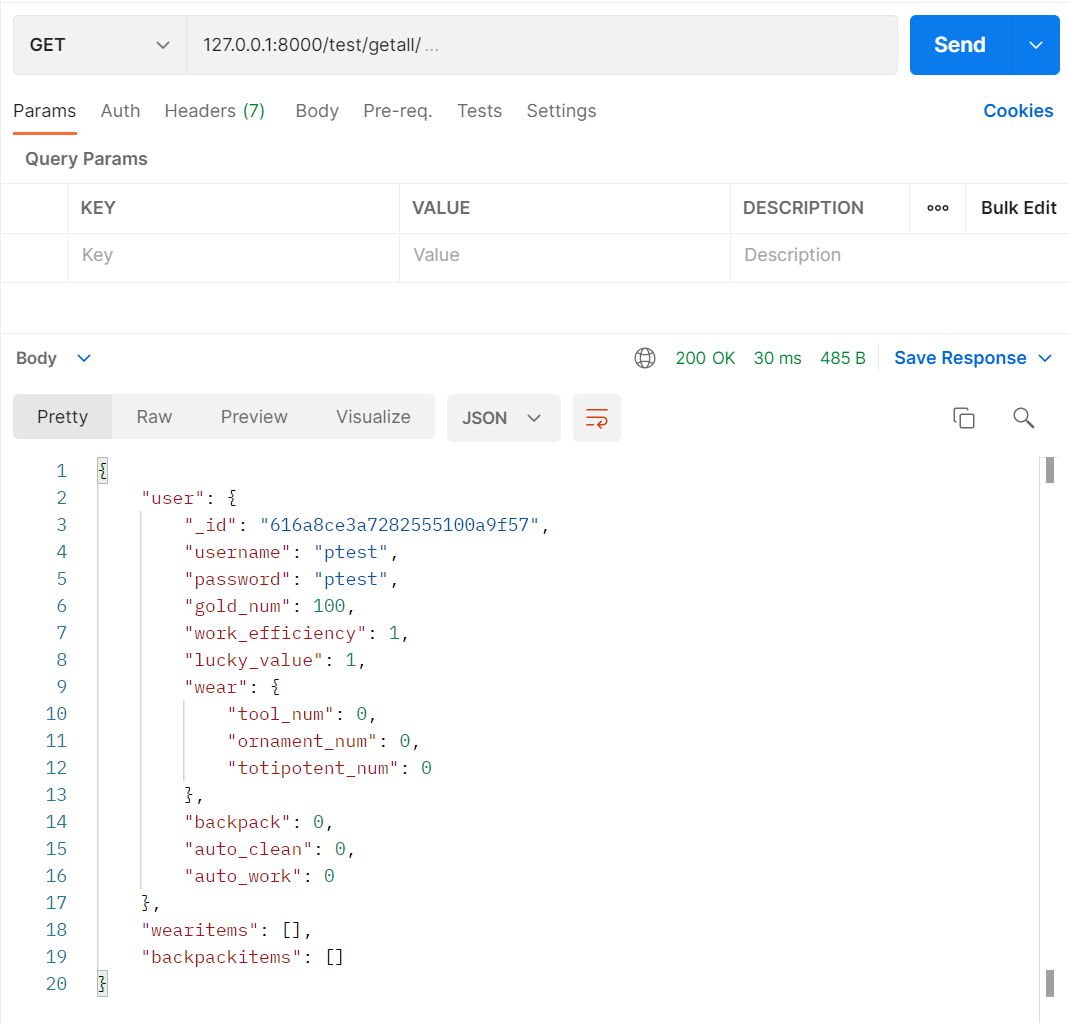


退出登录失败

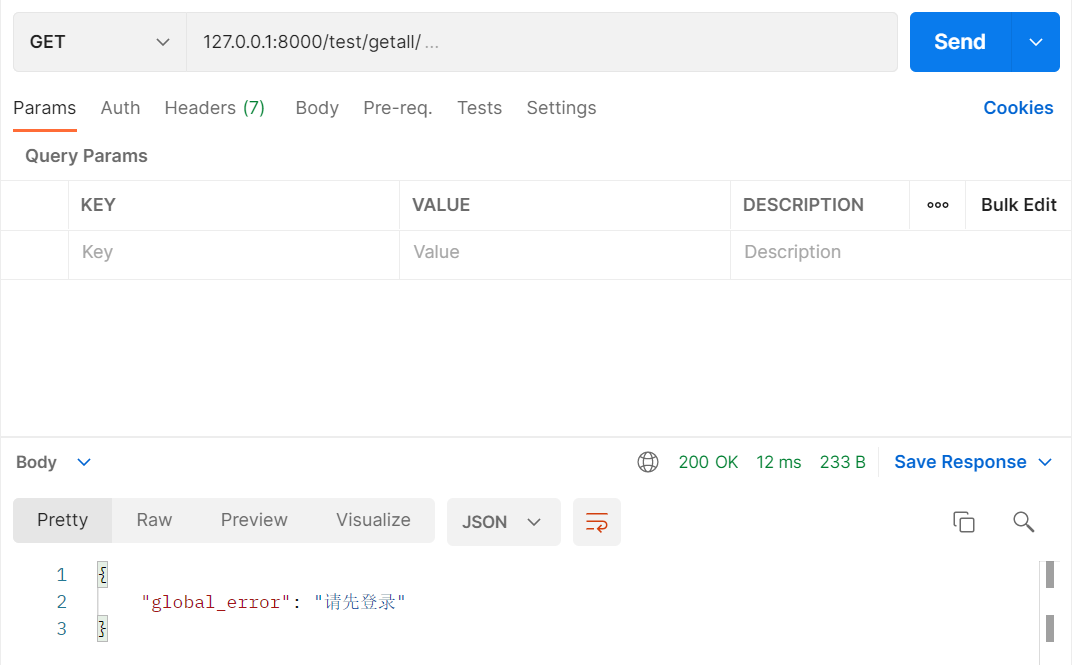


获取个人信息

获取成功

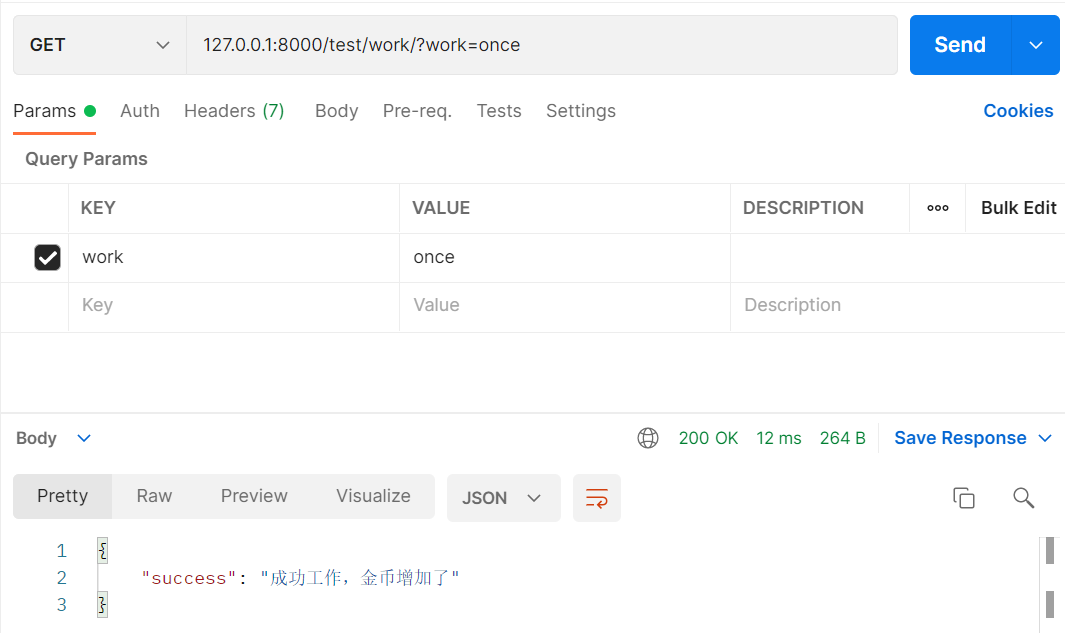


获取失败



工作一次

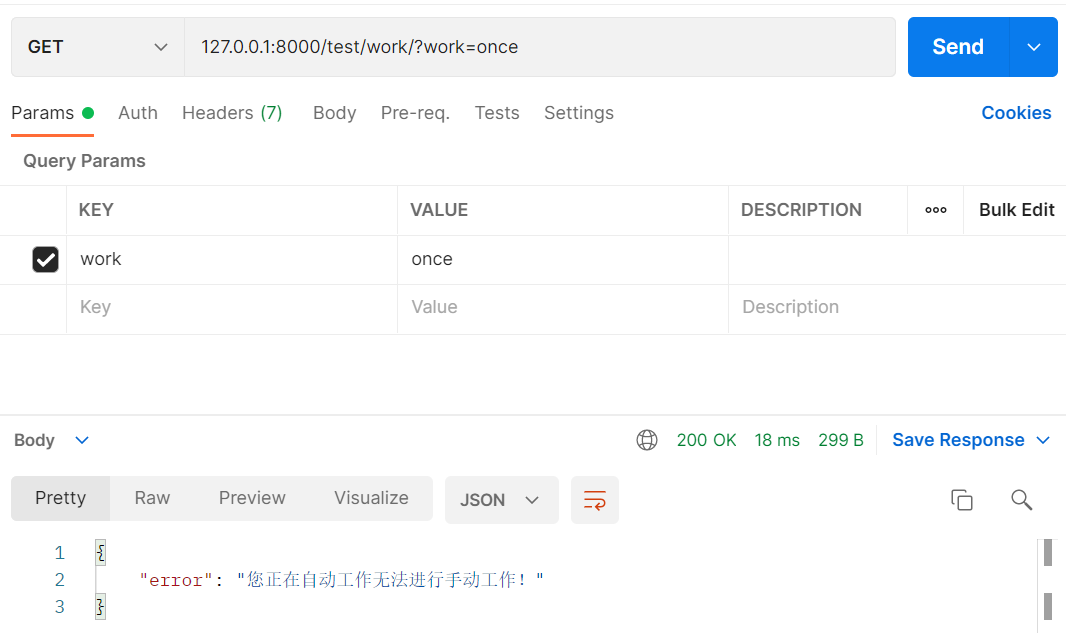
成功



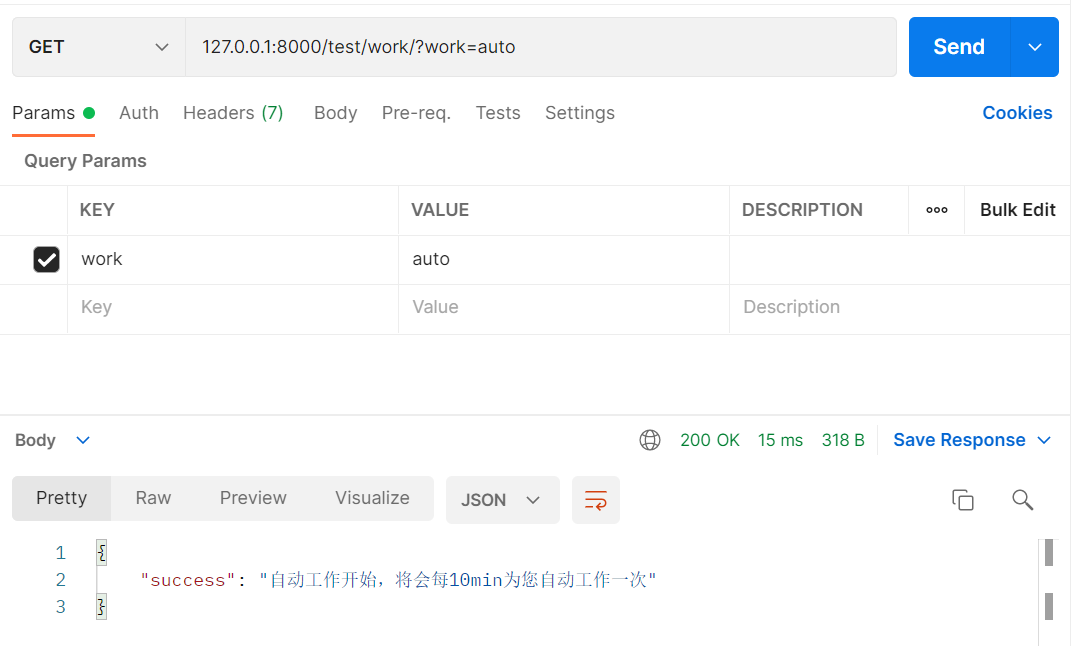
数据库金币更新



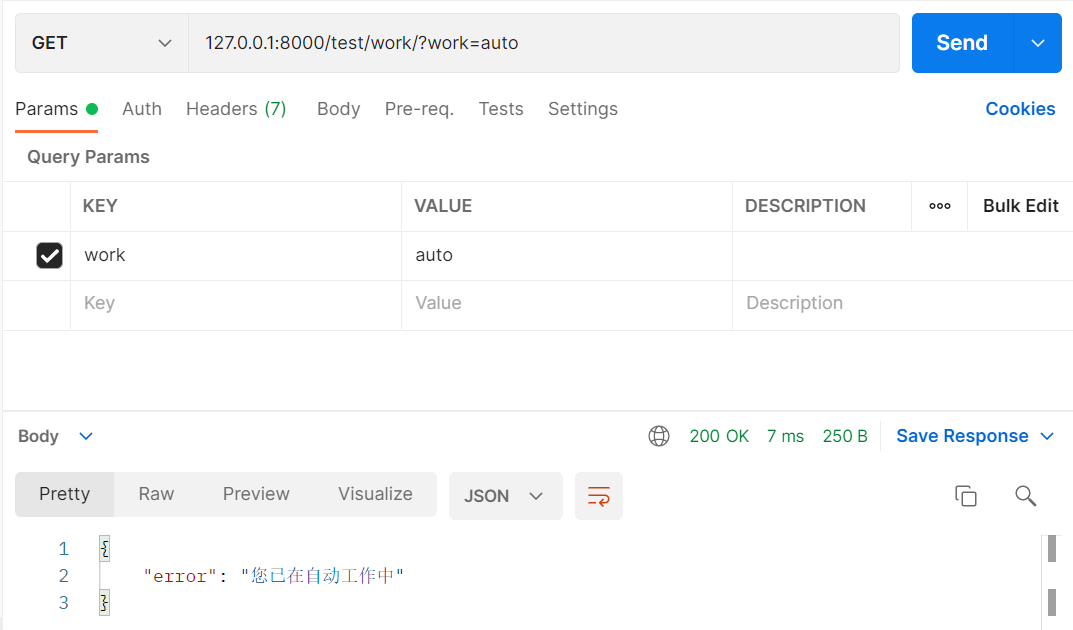
若设置了自动工作则导致失败



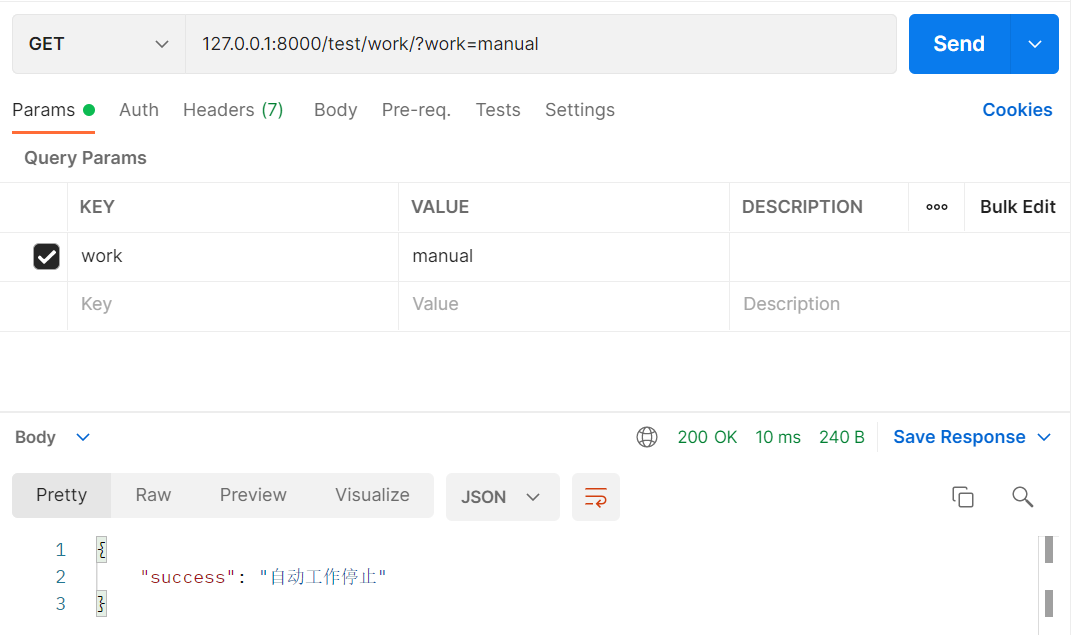
设置自动工作成功



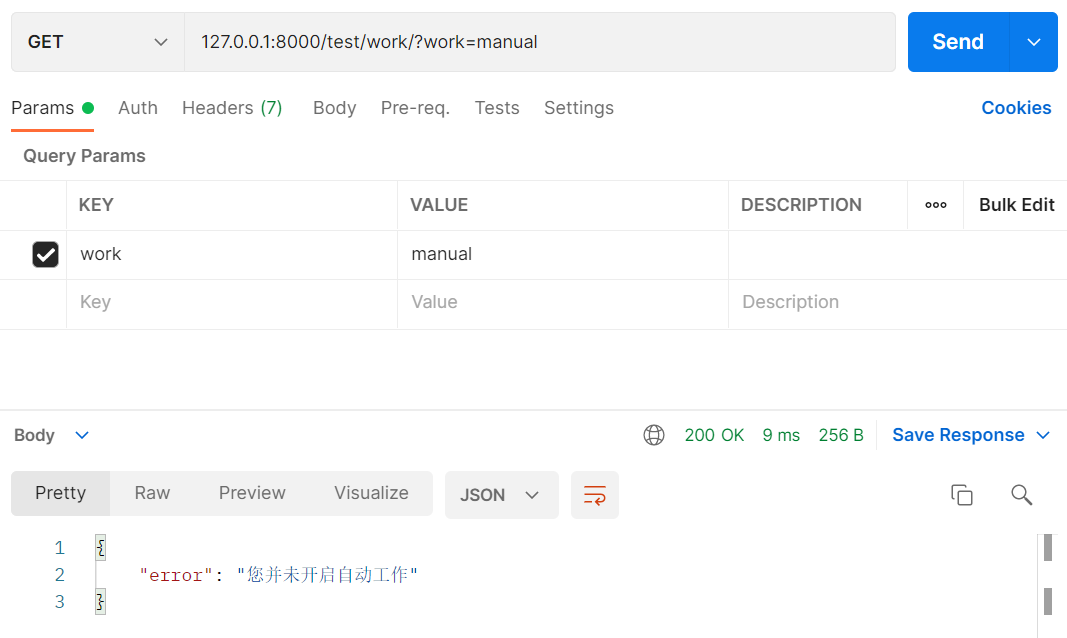
设置自动工作失败



设置手动工作成功

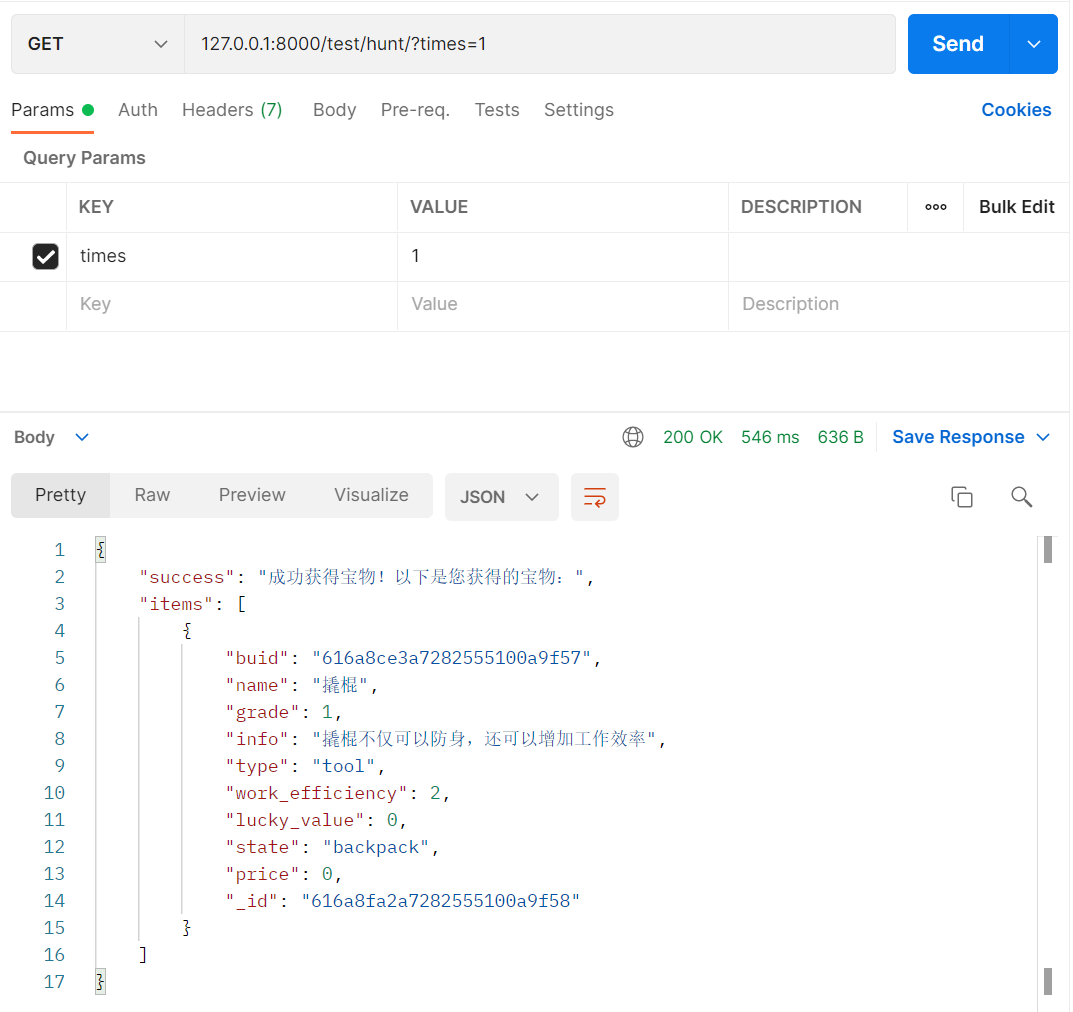


设置手动工作失败

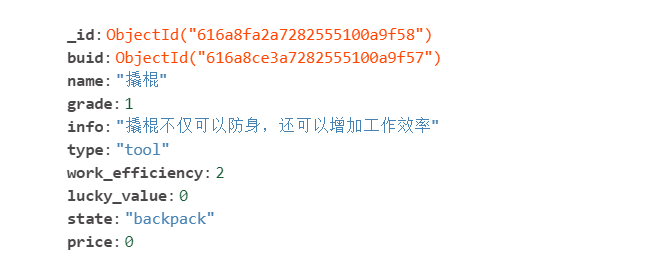


寻宝

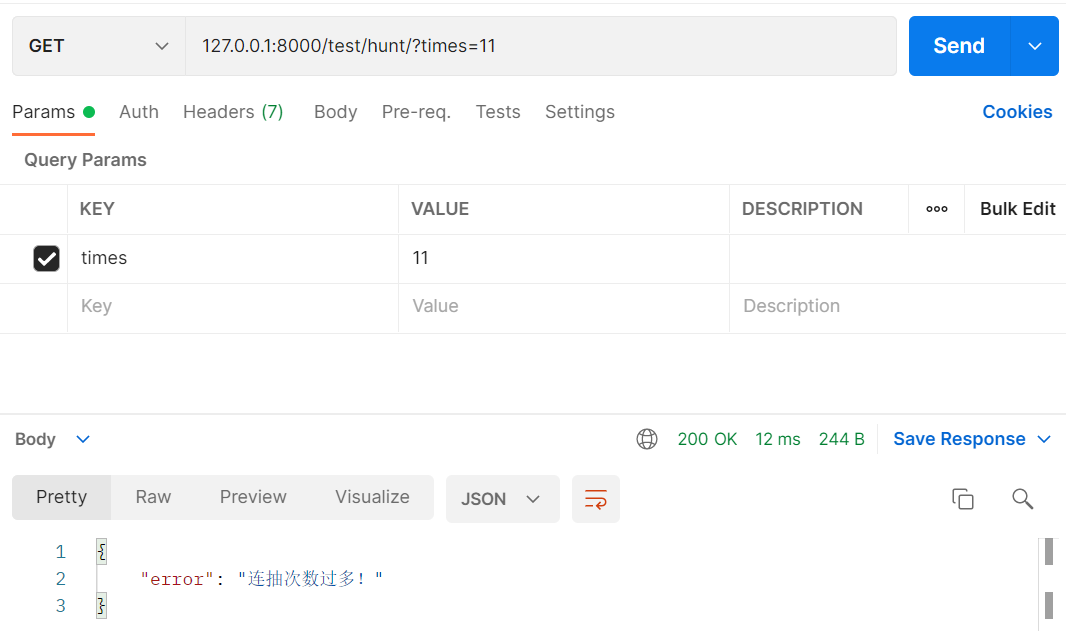
寻宝成功



数据库中

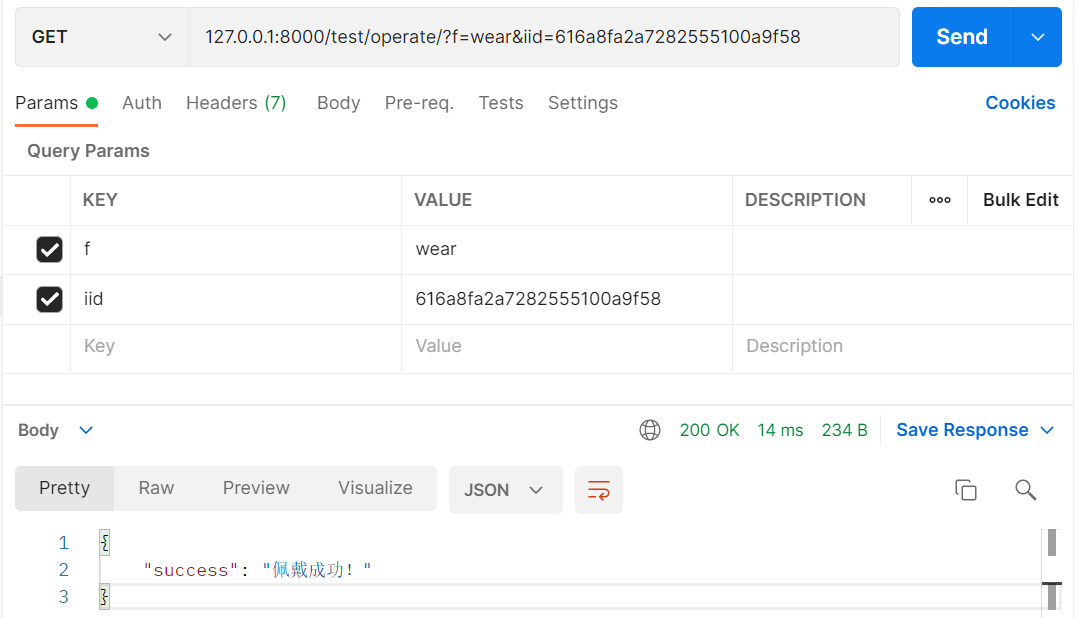


寻宝失败（连抽过多）

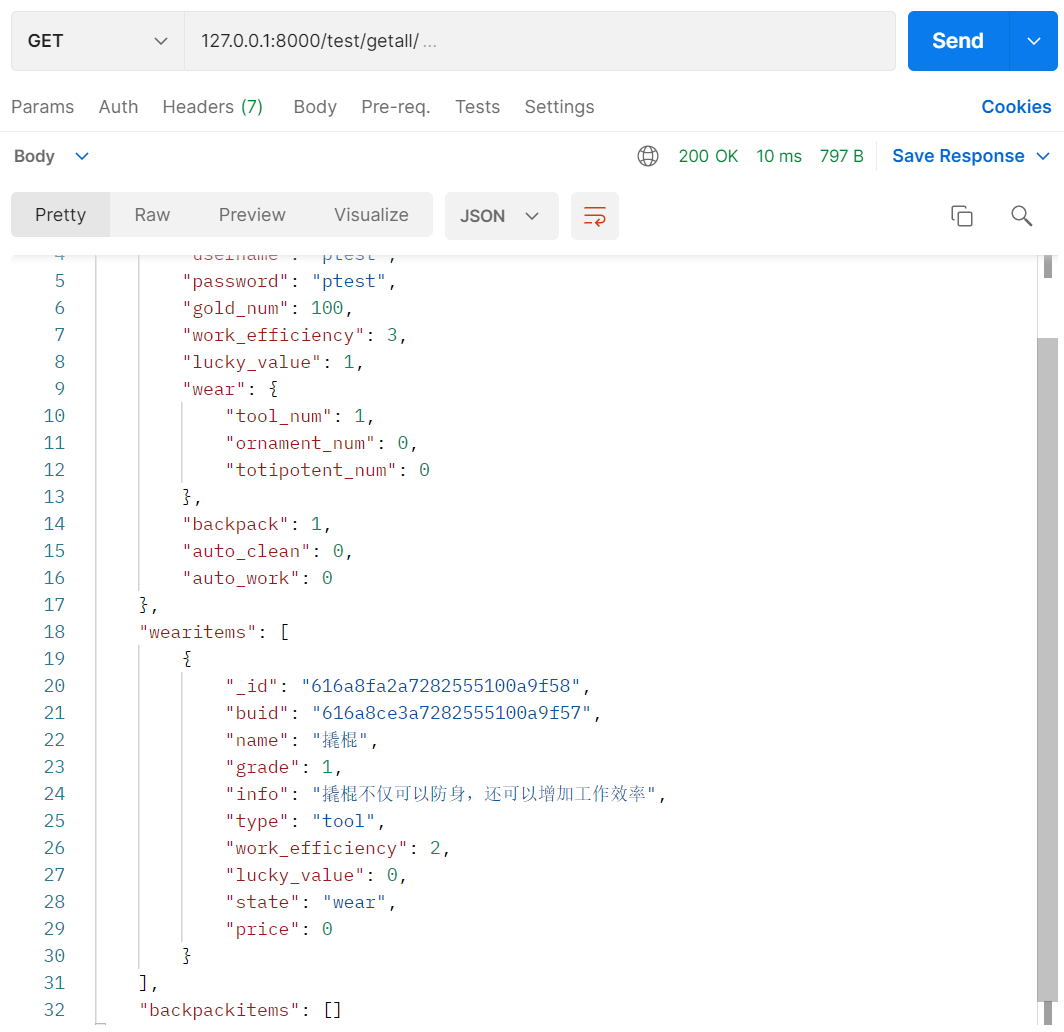


佩戴（佩戴上述撬棍）

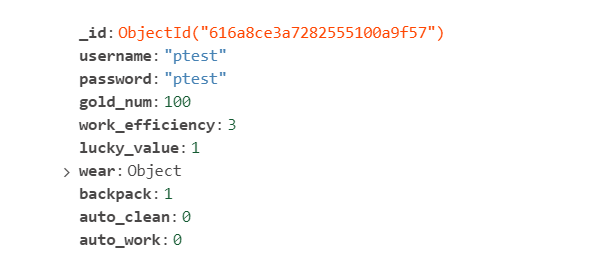
佩戴成功



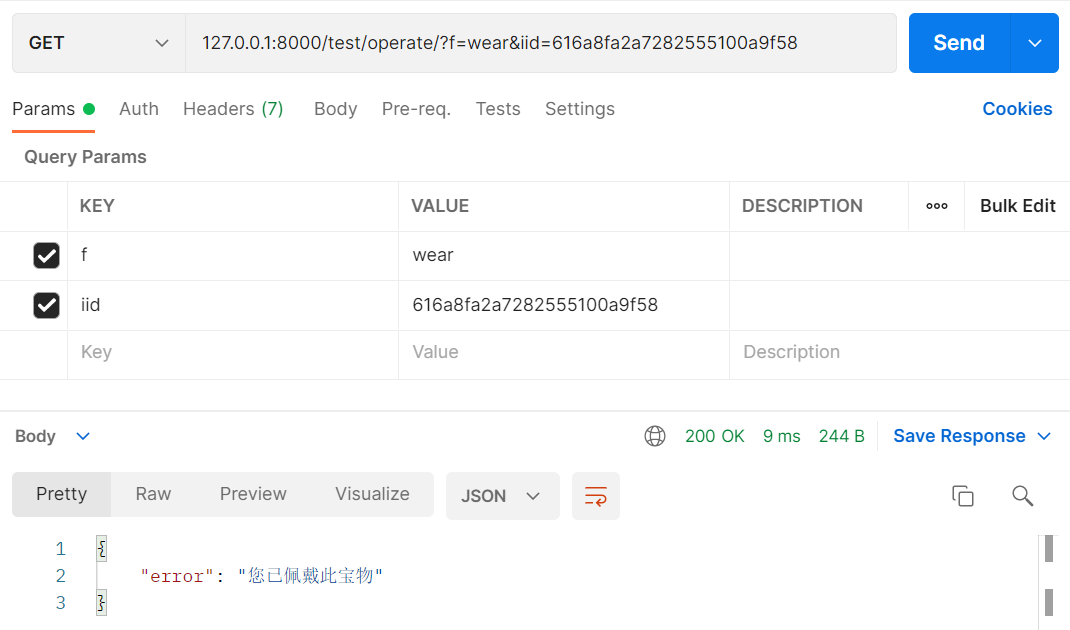
佩戴成功后getall

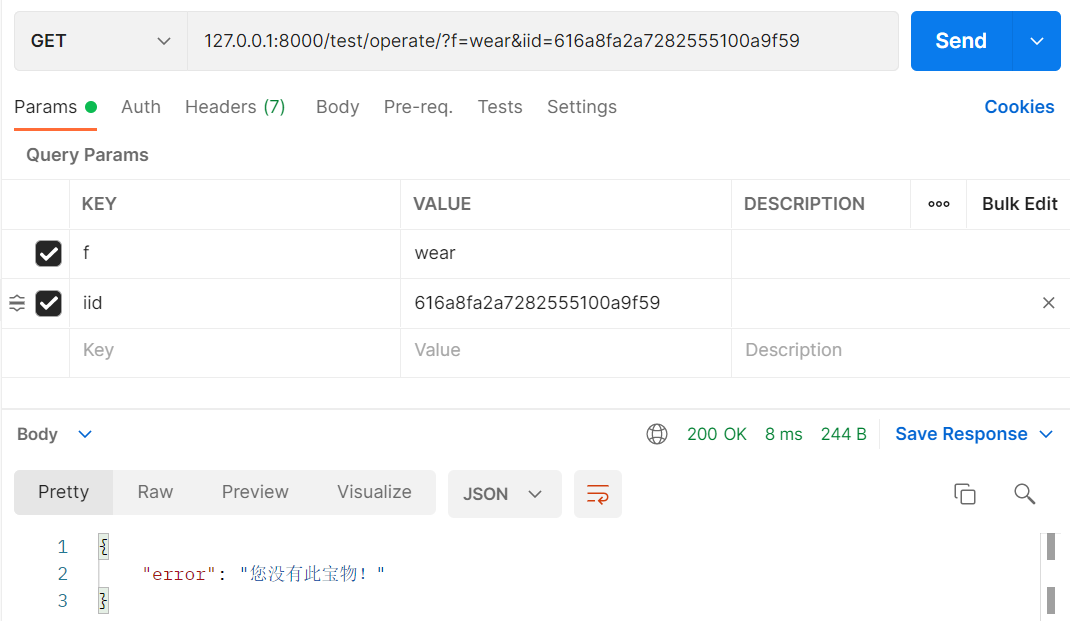


佩戴成功后数据库



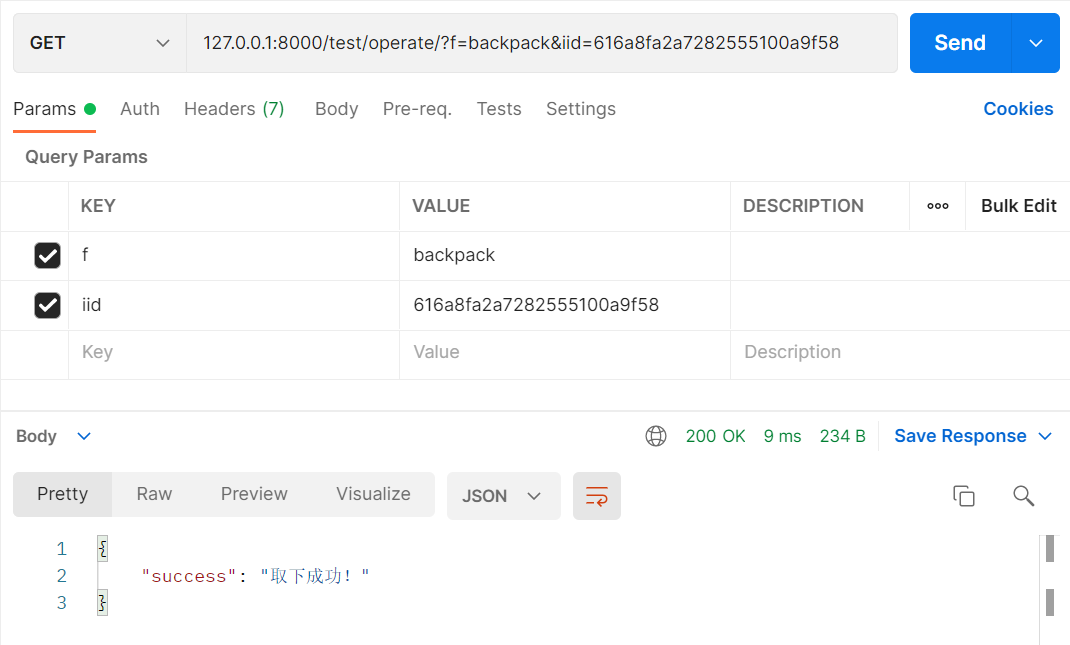
佩戴失败



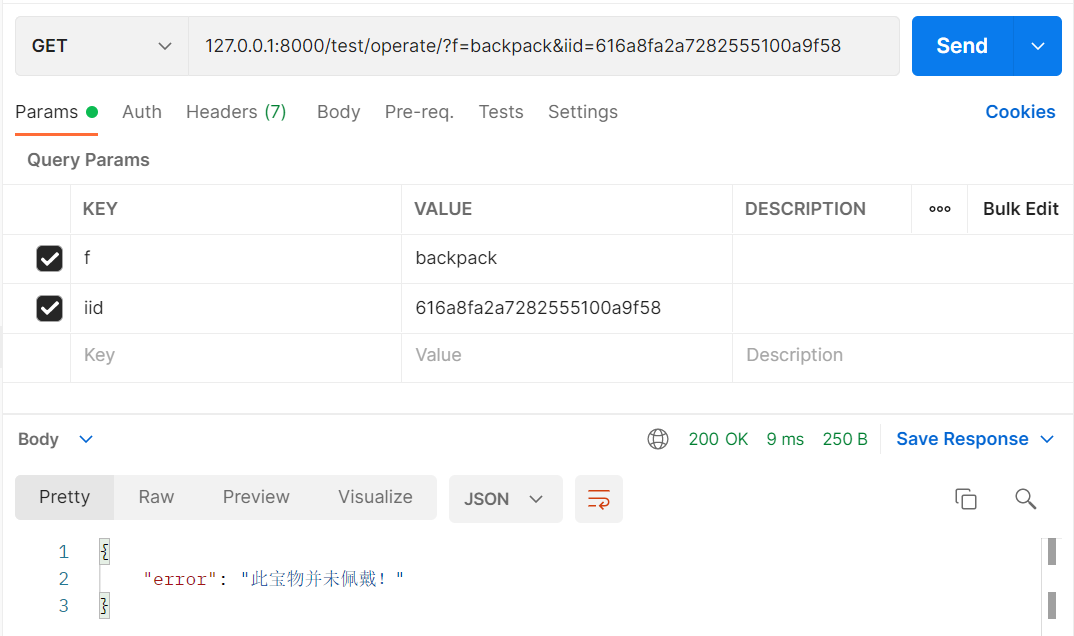


取下

取下成功



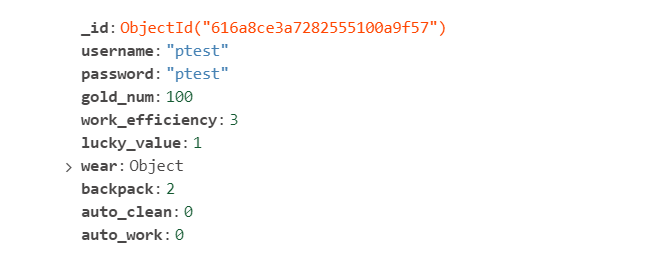
取下失败

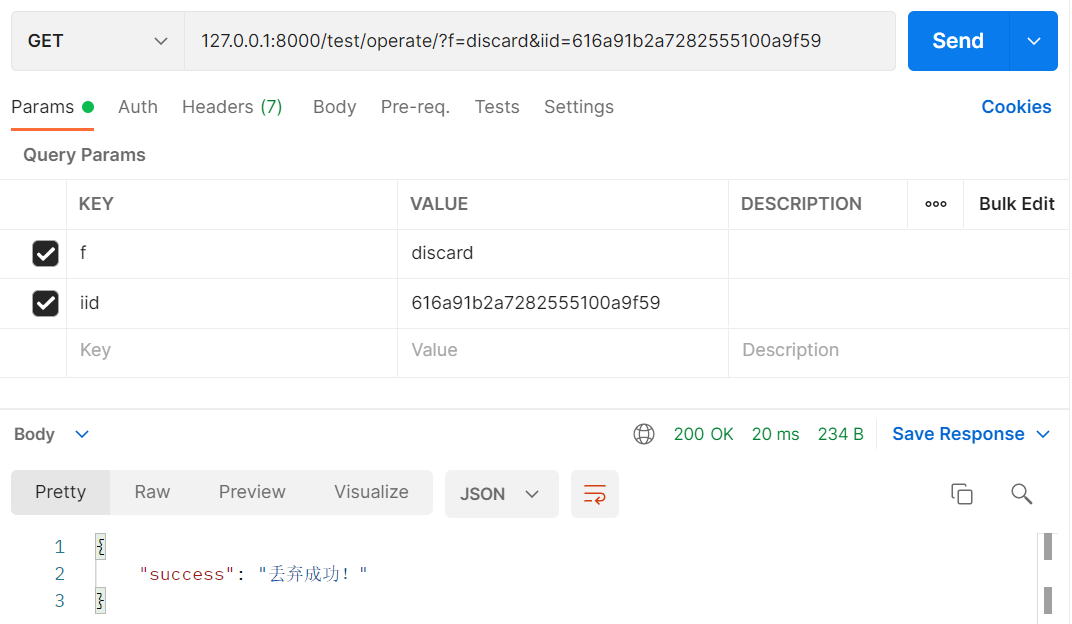


丢弃

丢弃成功（丢弃的是另一个寻到的宝物）

丢弃前数据库：

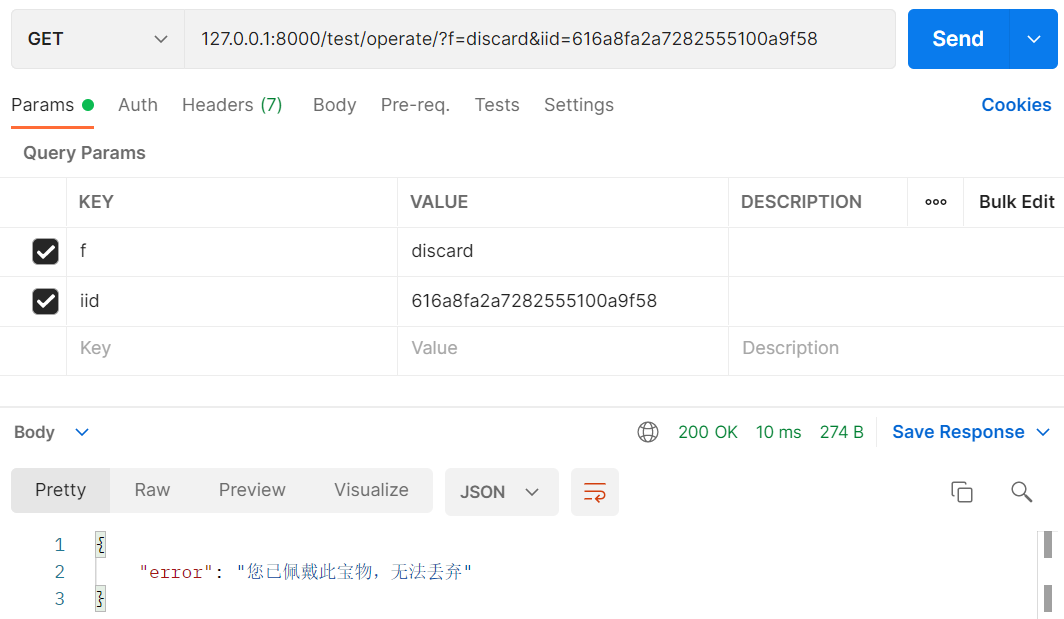


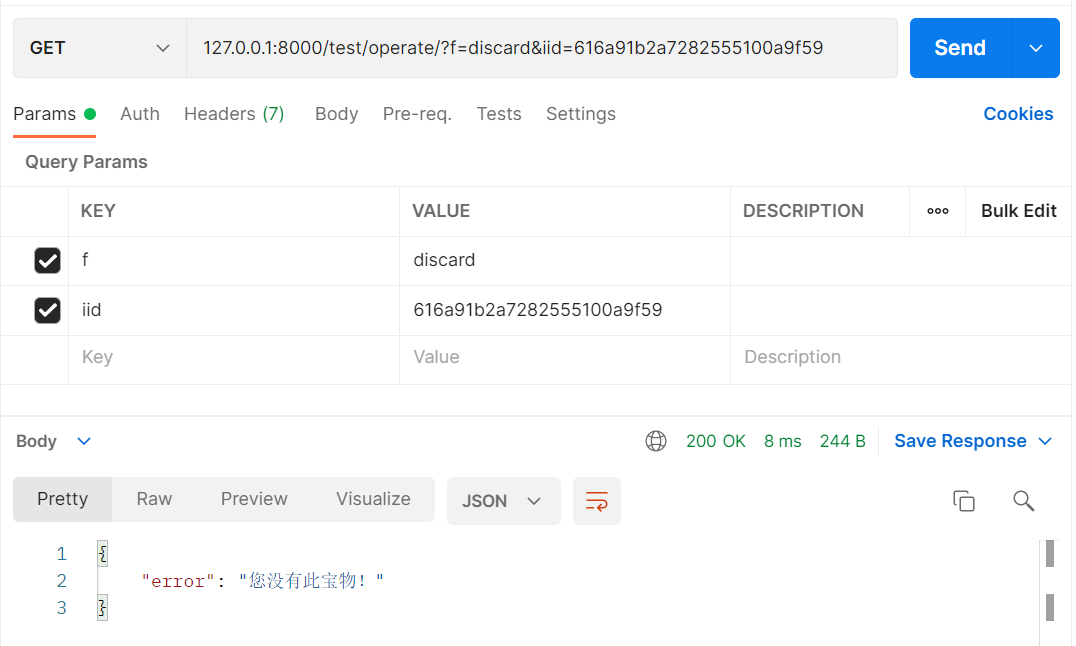


丢弃后数据库

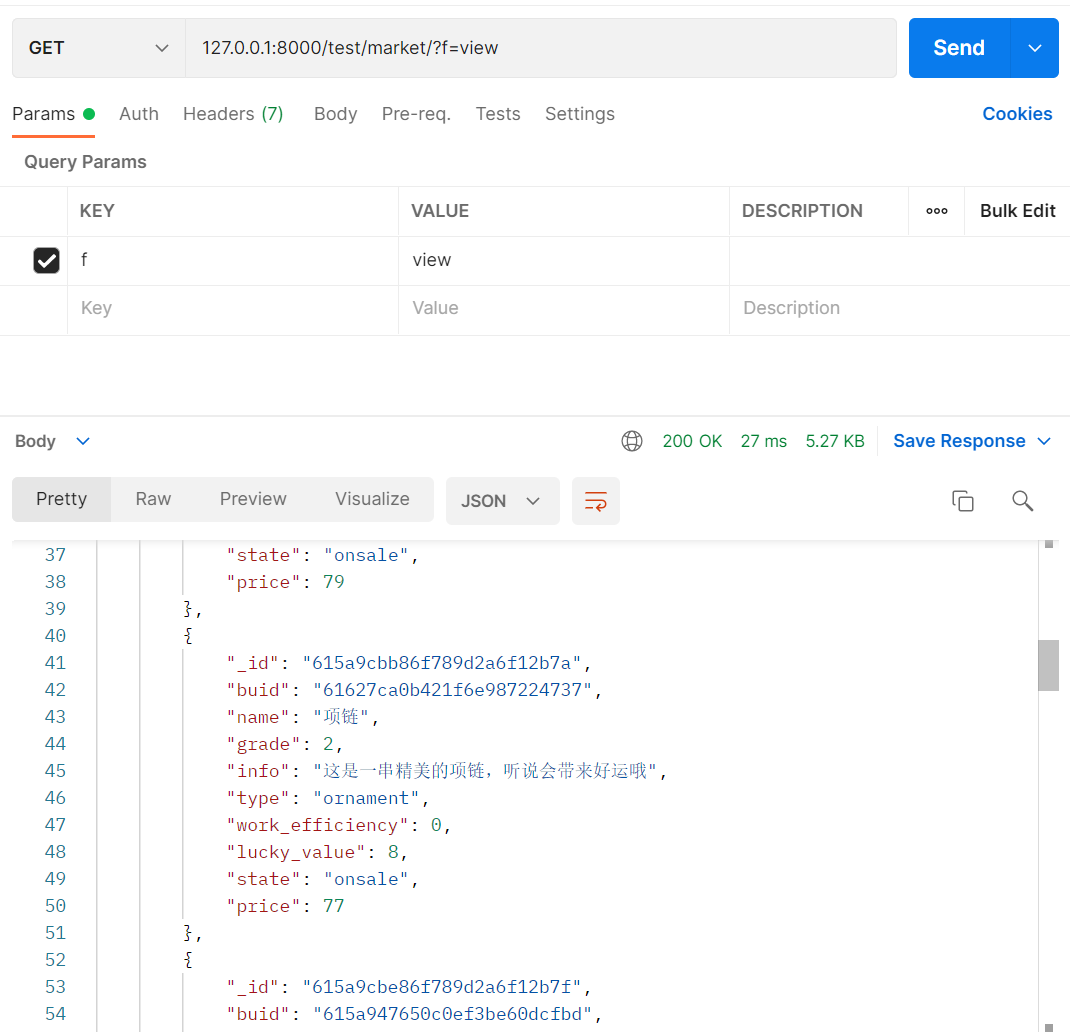


丢弃失败





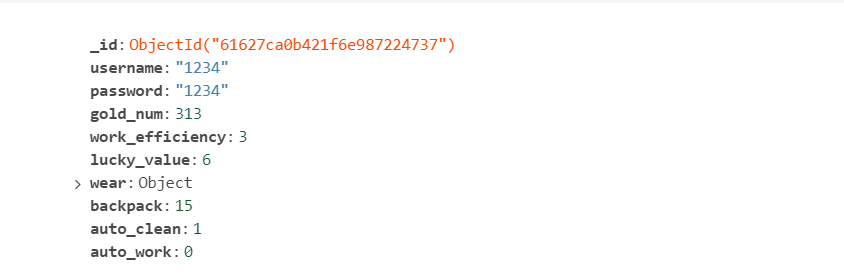
查看市场



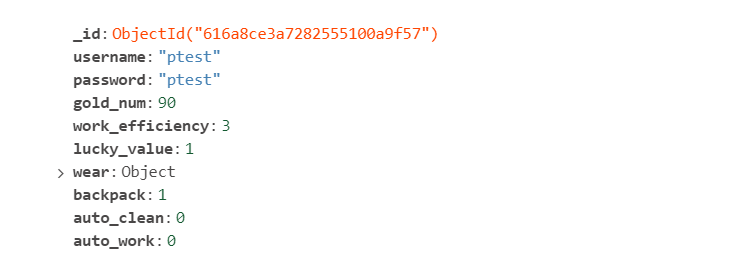
购买（购买上述项链）

购买前数据库

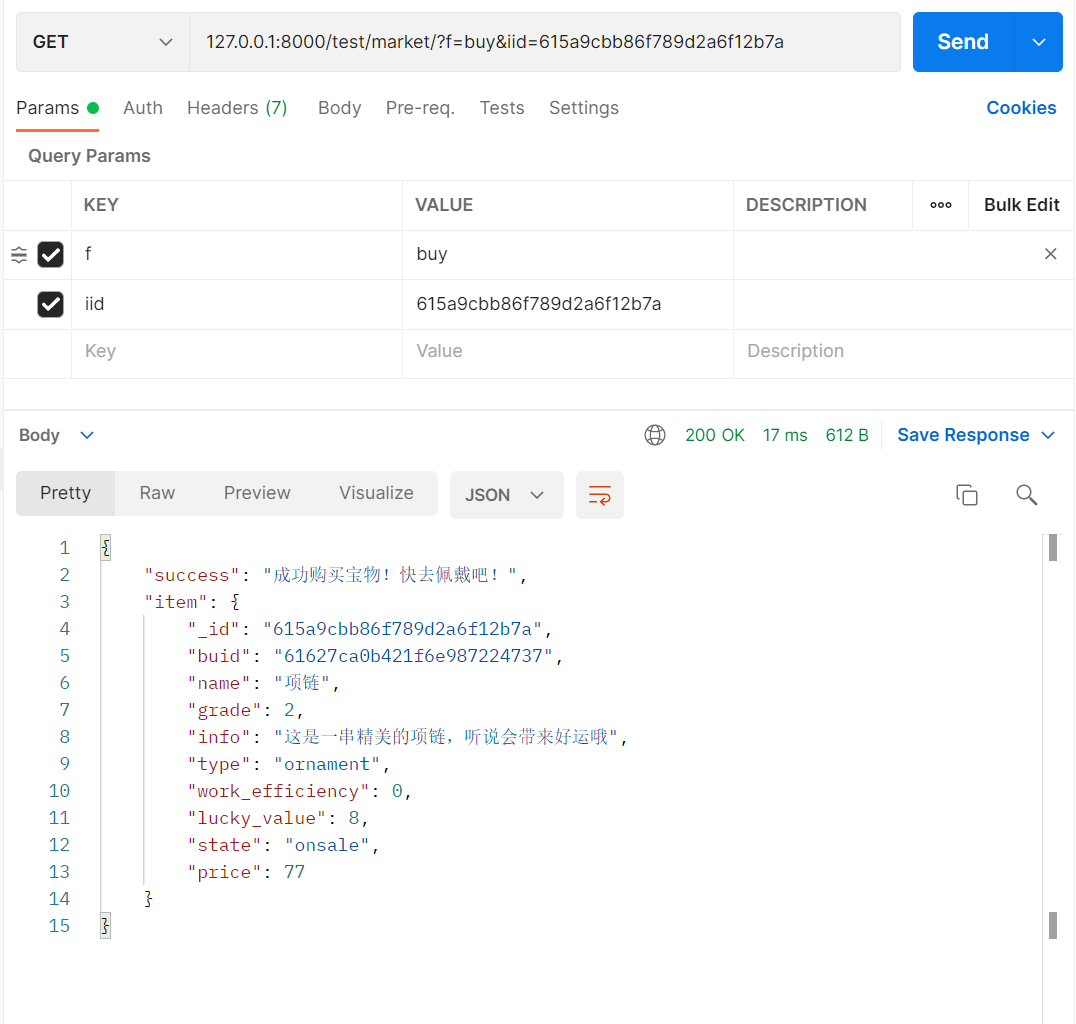
卖家



买家

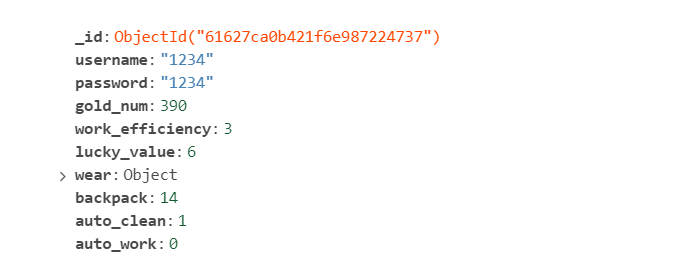


购买成功

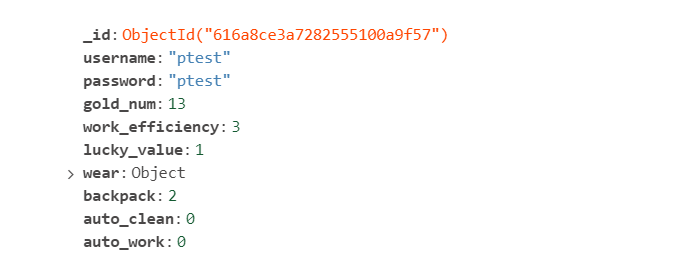


购买后数据库

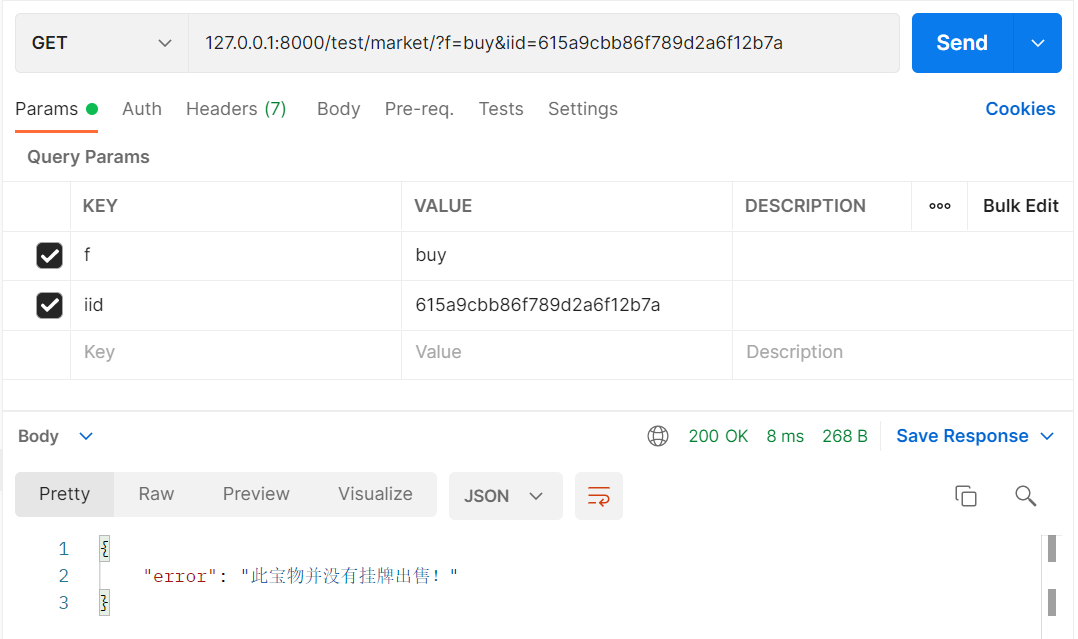
卖家



买家

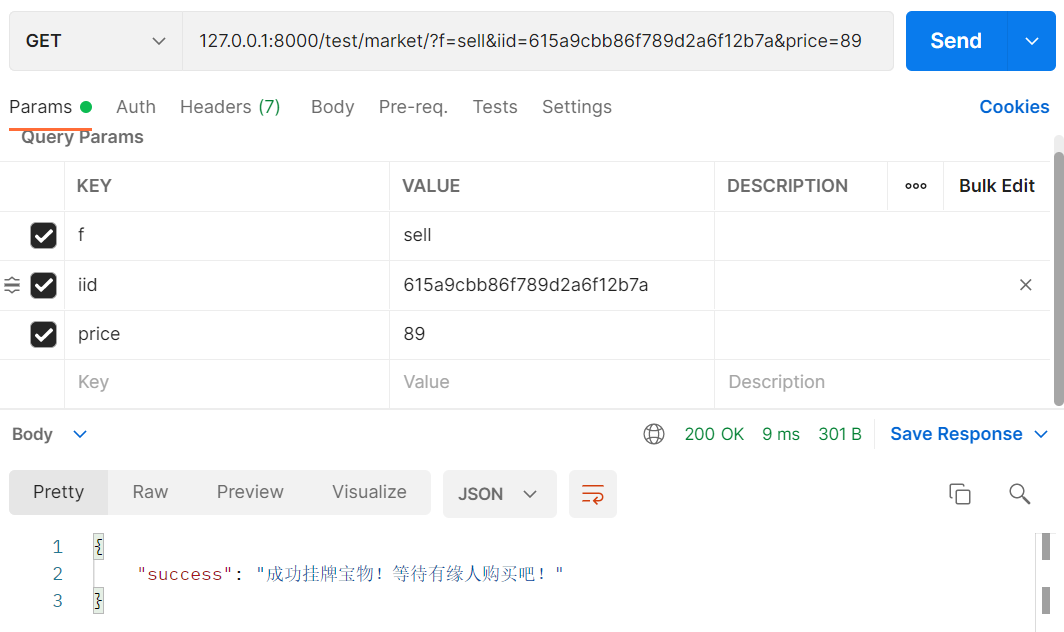


购买失败

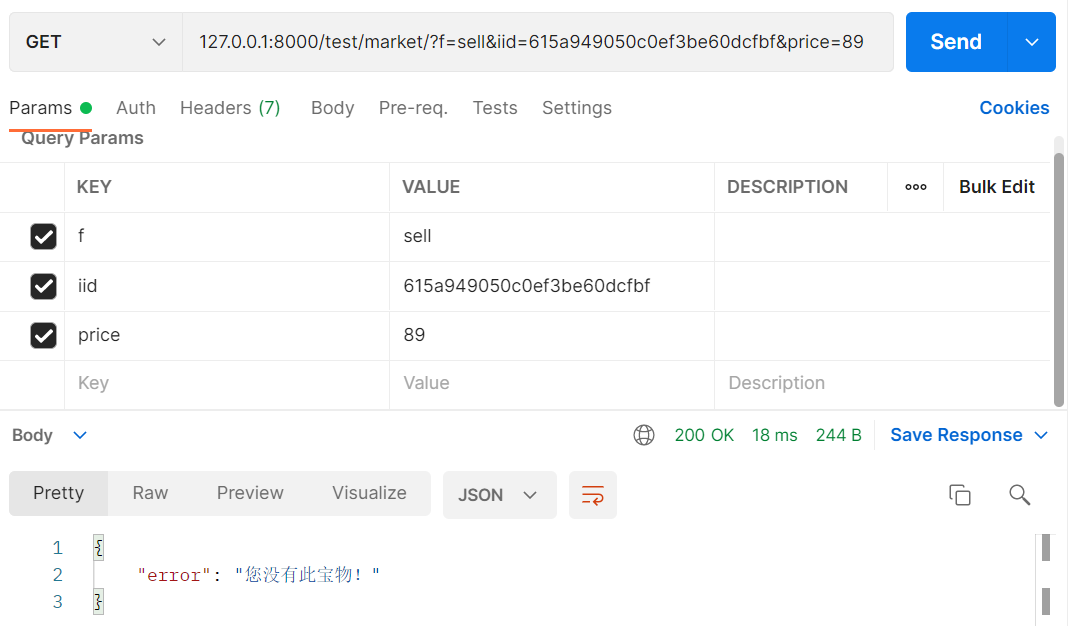


出售

出售成功

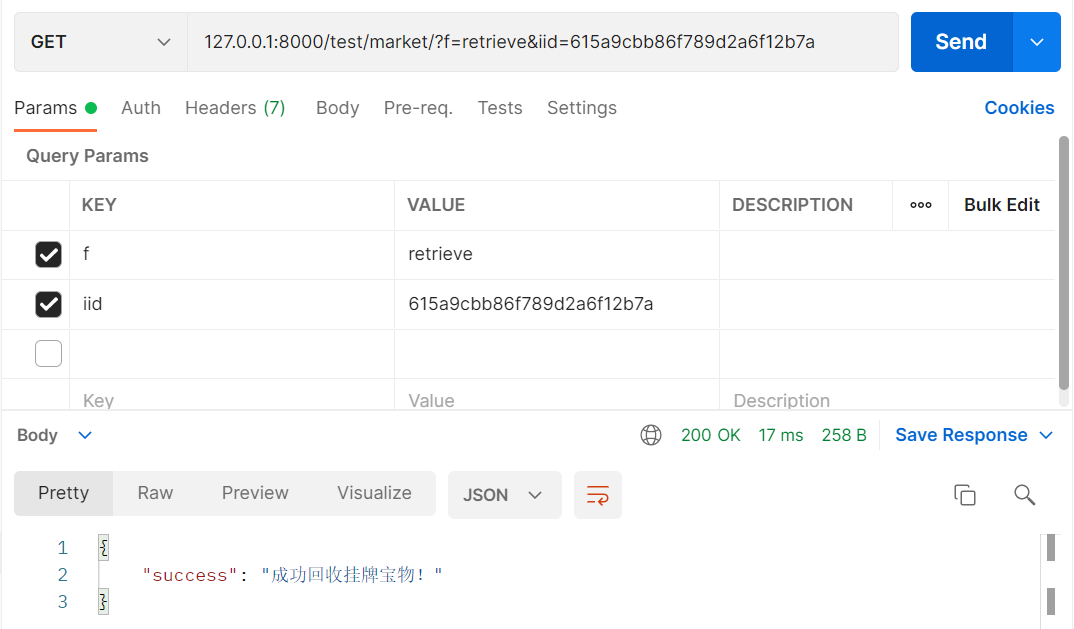


出售失败

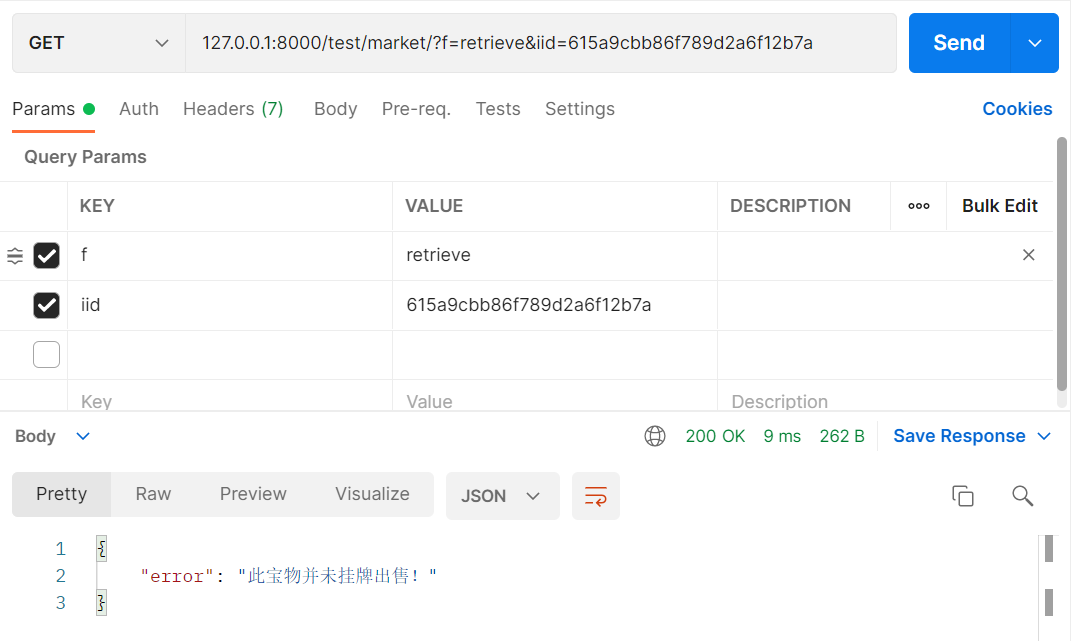


回收宝物

回收成功

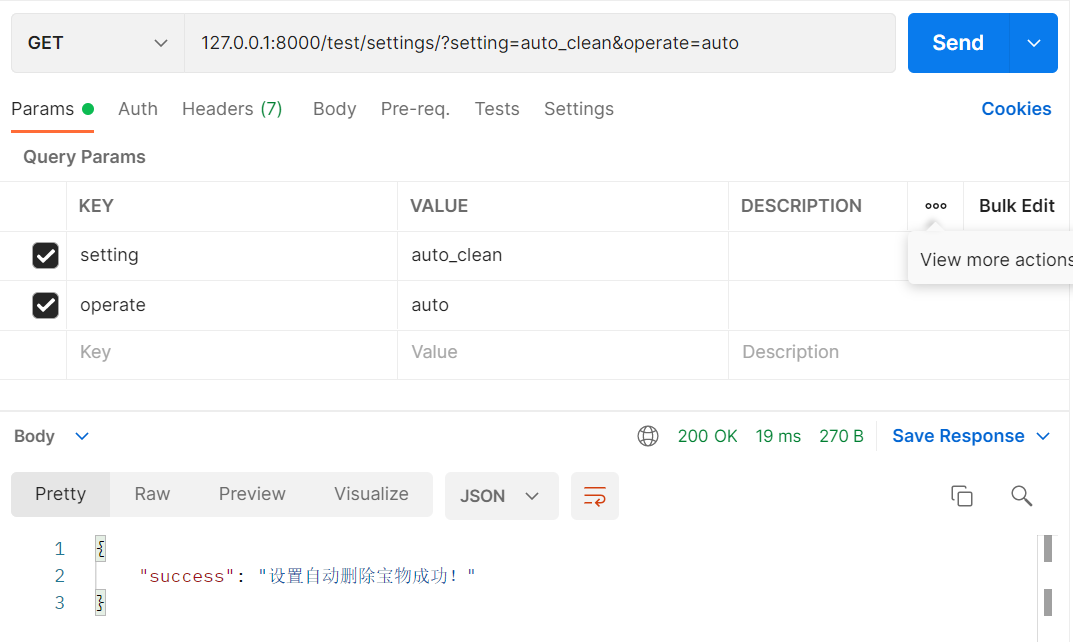


回收失败

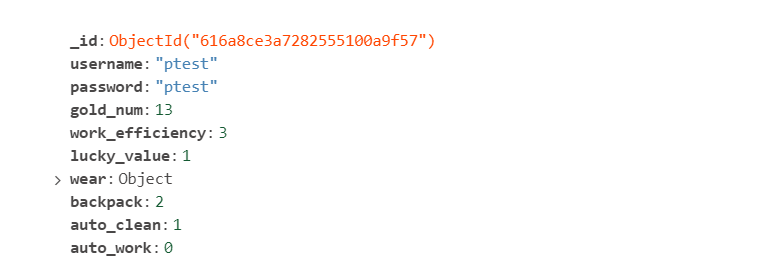


自动清理 开/关

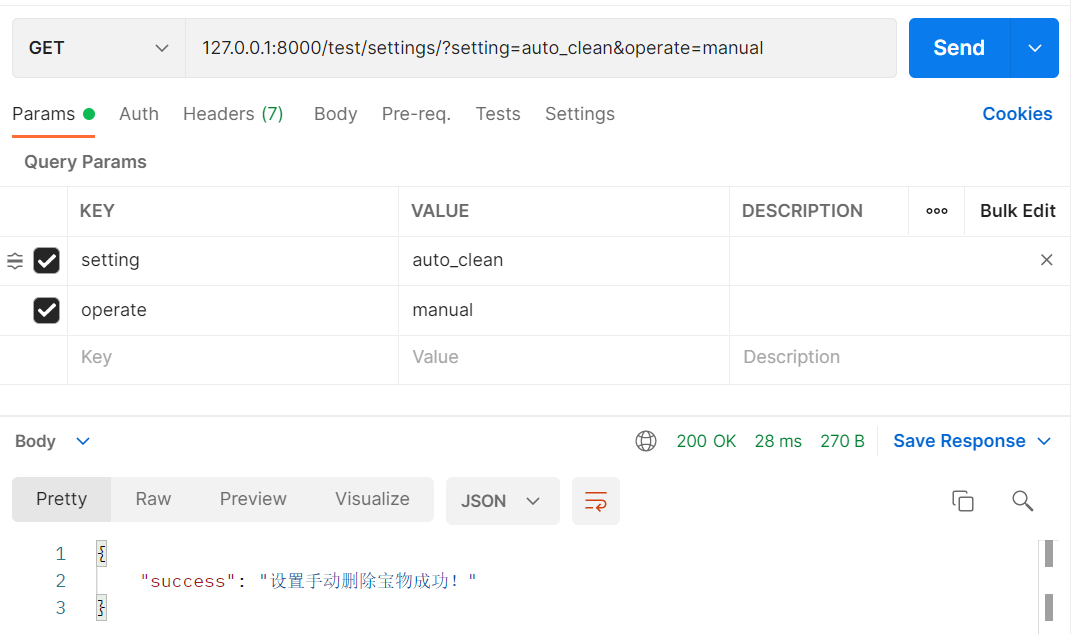
自动



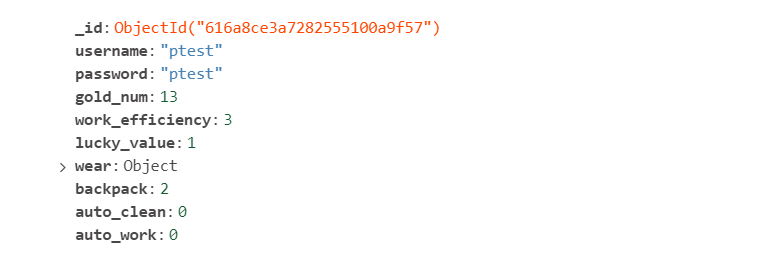
数据库



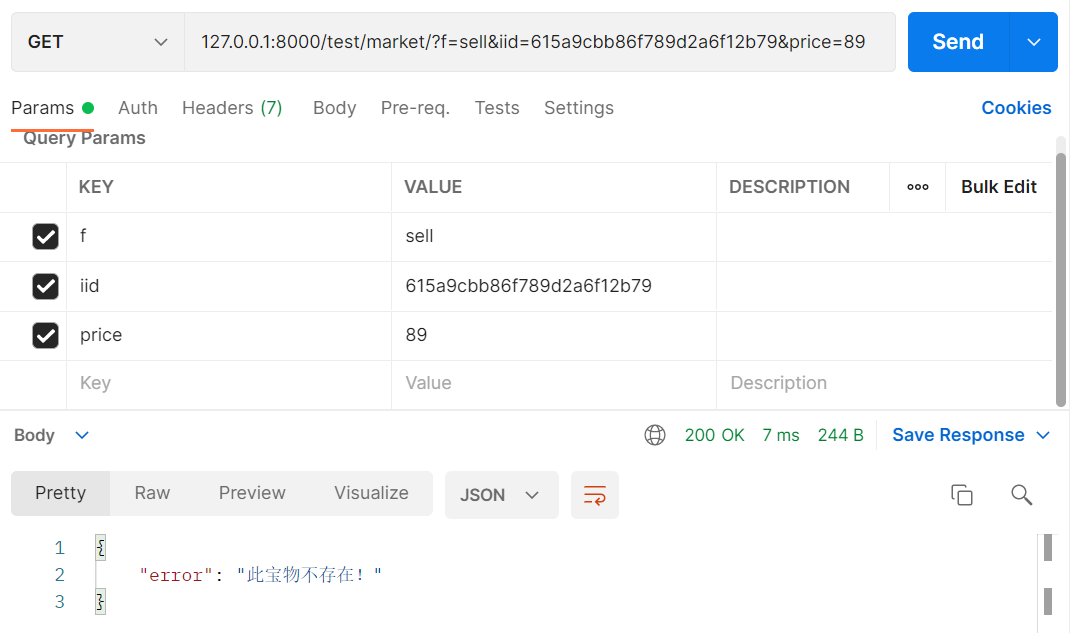
手动



数据库



若没有宝物则



若格式不对（缺少关键json数据）则返回

