数据库

version: mysql80

name: mysql80

password: 123456

管理员打开cmd

启动:net start mysql80

然后可打开mysql终端

Python 3.13

Django项目学习

<https://www.bilibili.com/video/BV19spseGE9Y?spm_id_from=333.788.videopod.episodes&vd_source=97bd8bd718a48e29bd9b00dd6b7aee5b&p=2>

注：学习过程中发先Tab可补全预测大量代码，有时可#注释，然后直接tab续写

2025/5/5

Dbeaver数据库管理工具教程

<https://www.bilibili.com/video/BV1G64y197yT/?vd_source=97bd8bd718a48e29bd9b00dd6b7aee5b>

**数据库设置**

DATABASES = {

    'default': {

        'ENGINE': 'django.db.backends.mysql',

        'NAME': 'project54',

        'USER': 'root',

        'PASSWORD': '123456',

        'HOST': 'localhost',

        'PORT': '3306',

        'OPTIONS': {

            'init\_command': "SET sql\_mode='STRICT\_TRANS\_TABLES'",

            'charset': 'utf8mb4',

        }

    }

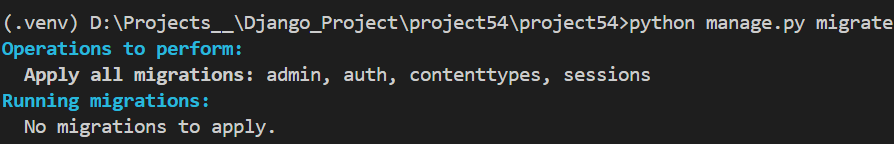
}

完成后记得数据库迁移

1: python manage.py makemigrations

No changes detected

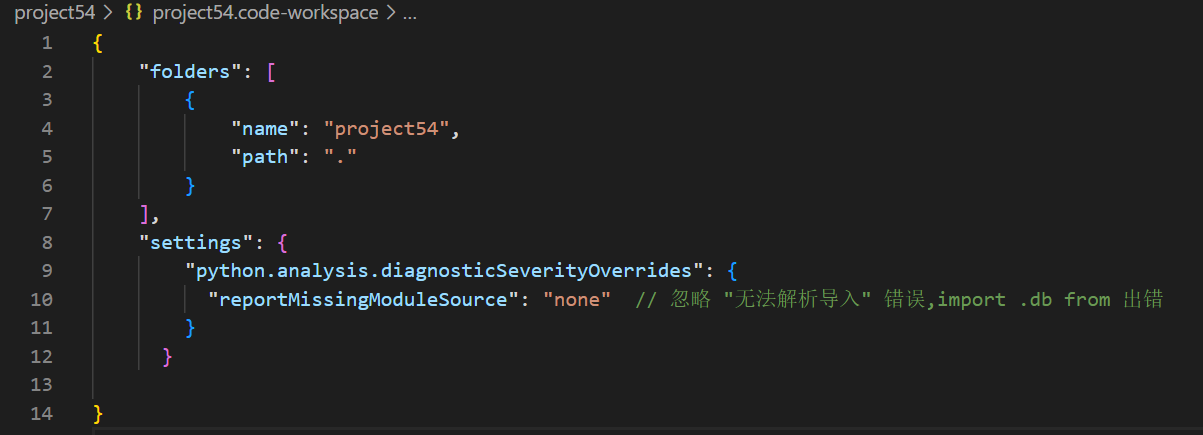
2:python manage.py migrate



编写 users中的models时，tab能自动编写好多 邮箱，头像之类的

Ctrl + shift + p 相当于vscode的终端，reload windows能一键重启vs窗口

忽略了pylance的奇葩警告导入找不到源



https://www.runoob.com/django/django-template.html

额外内容：把 存的list列表 传入视图，加上数据库的话似乎就行了



2025/5/6

开始前端vue

Pip install npm,node no

官网下载包node.js安装，配置环境变量

<https://blog.csdn.net/WHF__/article/details/129362462>

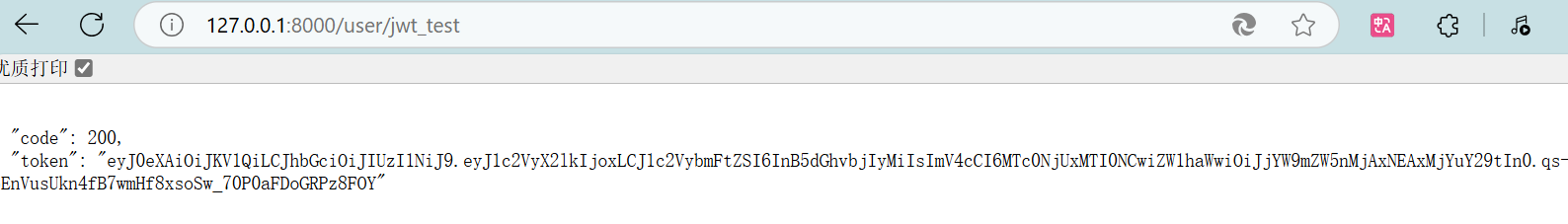
vue前端基础测试成功了(仅在虚拟环境.venv)

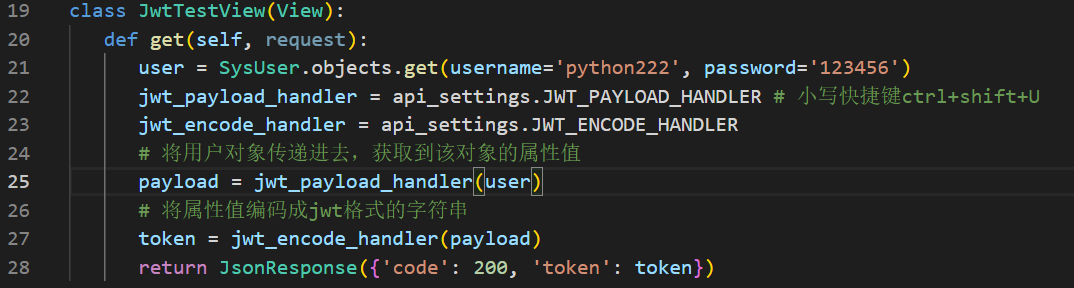
看到acm网站使用了springboot(待学)

使用了jwt（json web token）返回一个时刻不同的token, **(登录验证)**

若username和password存在于数据库，

则返回一个token,否则token无（false）





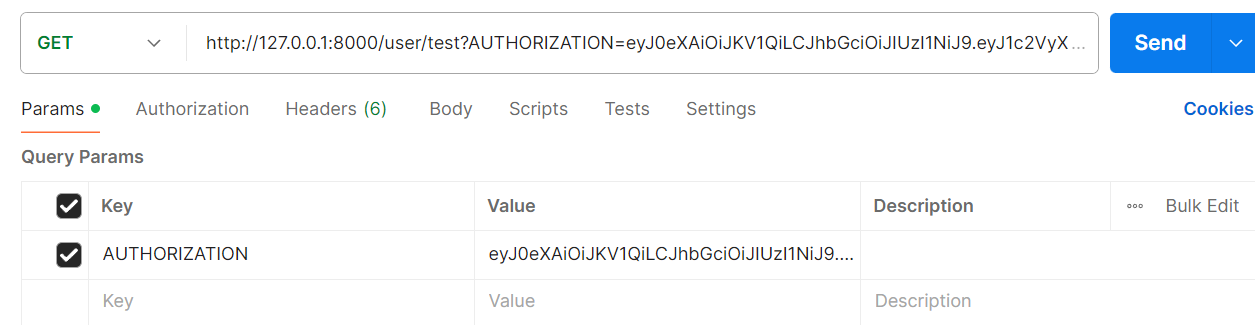
**postman使用**

用户名xxapple

[1271332306@qq.com](mailto:1271332306@qq.com)

zhangjiahui.123

使用生成的token似乎可以一直登录，



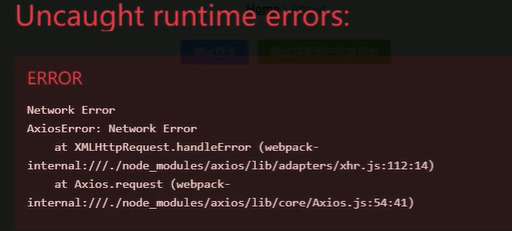
5/7

在前端vue的src下添加目录utl，添加request.Js. 按教程复制了request。Js

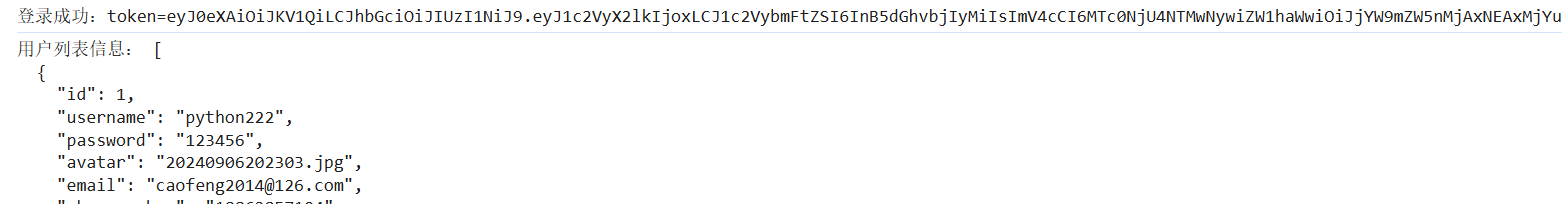


添加click时间





**跨域问题**解决，设置了一下settings

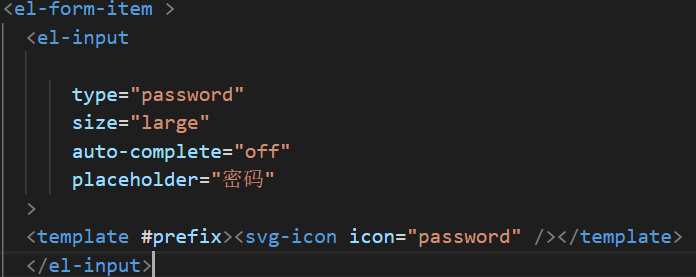


输了写了好多vue的代码，

实质上是.vue代替了.html文件，然后调用格式似乎方便些

第七讲：

学习了svg icon，矢量图标应用（设置较为麻烦，有warn的是因为版本快不支持sass，不用管）。设置好后， 更改图标只要修改 icon=” (这里)” 即可



八讲：登录验证

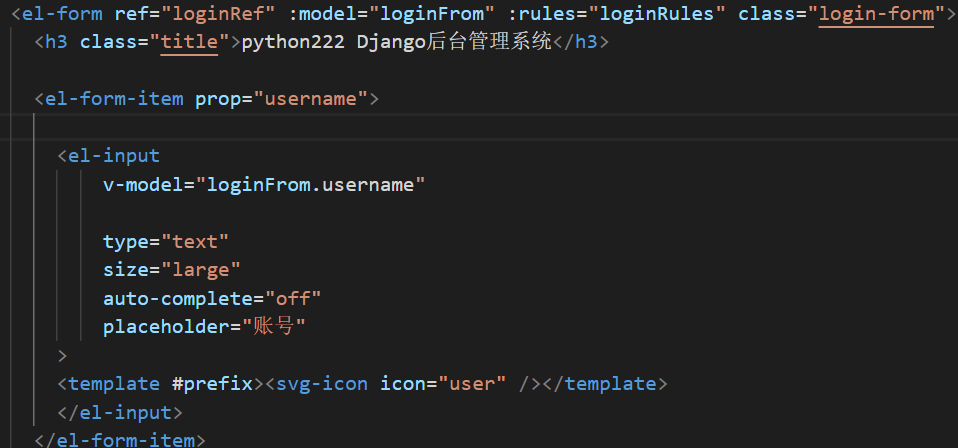
后端只要获取下username,password (修改view文件以及urls)，然后数据库比对返回Y /N

以下前端的



前端即可添加限制 上图为script 中的,下图进行引用

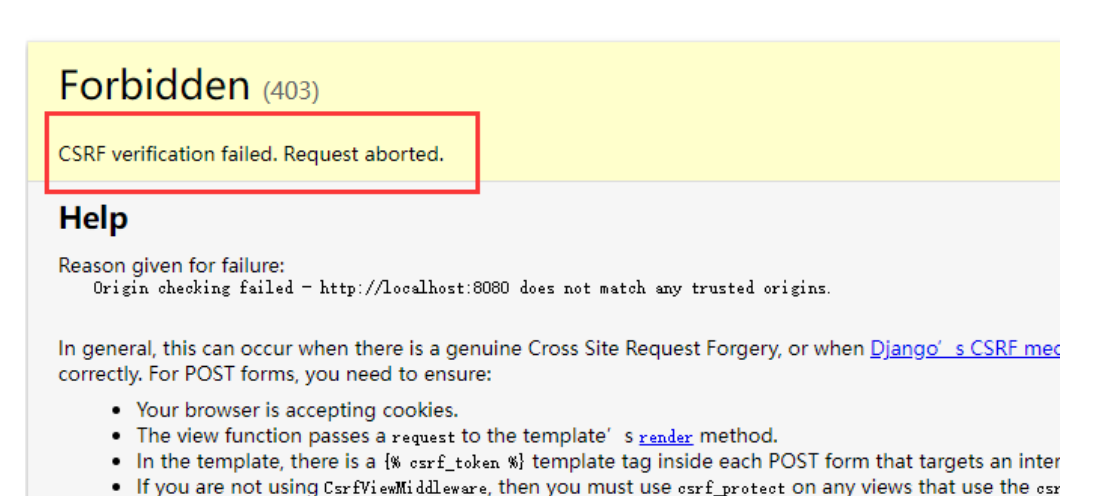
Rules : loginRules -> 下面添加 username对应

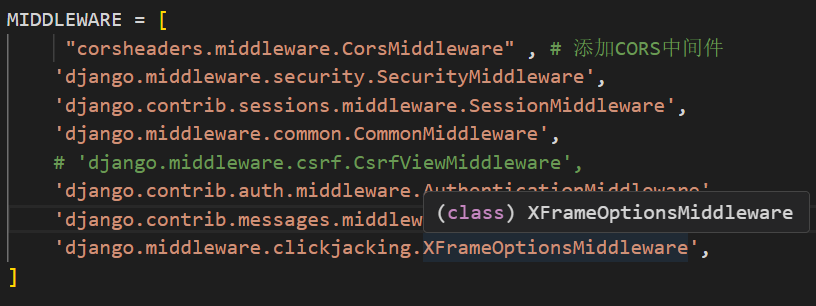




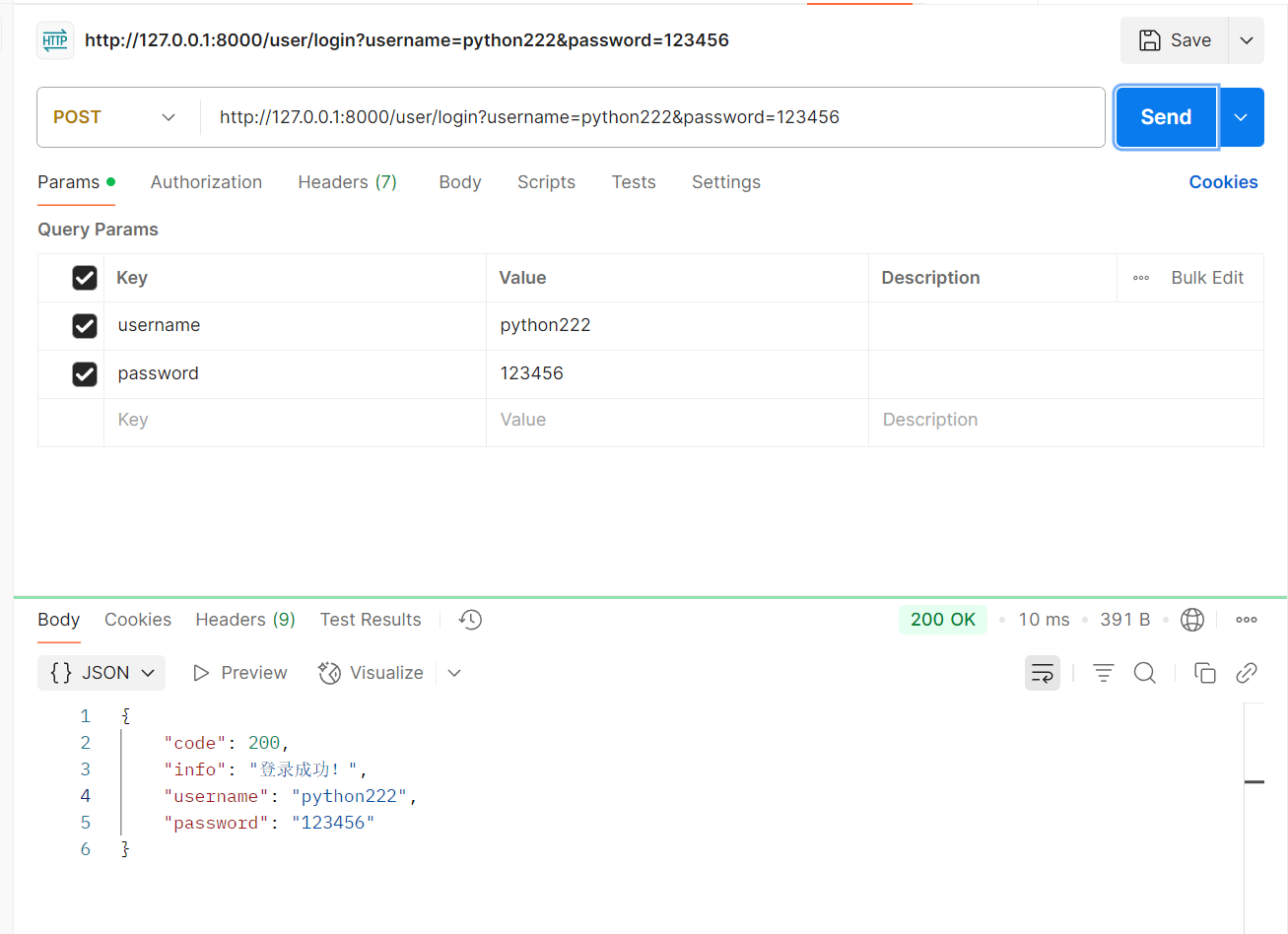
上图为发送表单以及添加url

但是会有报错，表单被抛弃，注释一下setting的这一行即可

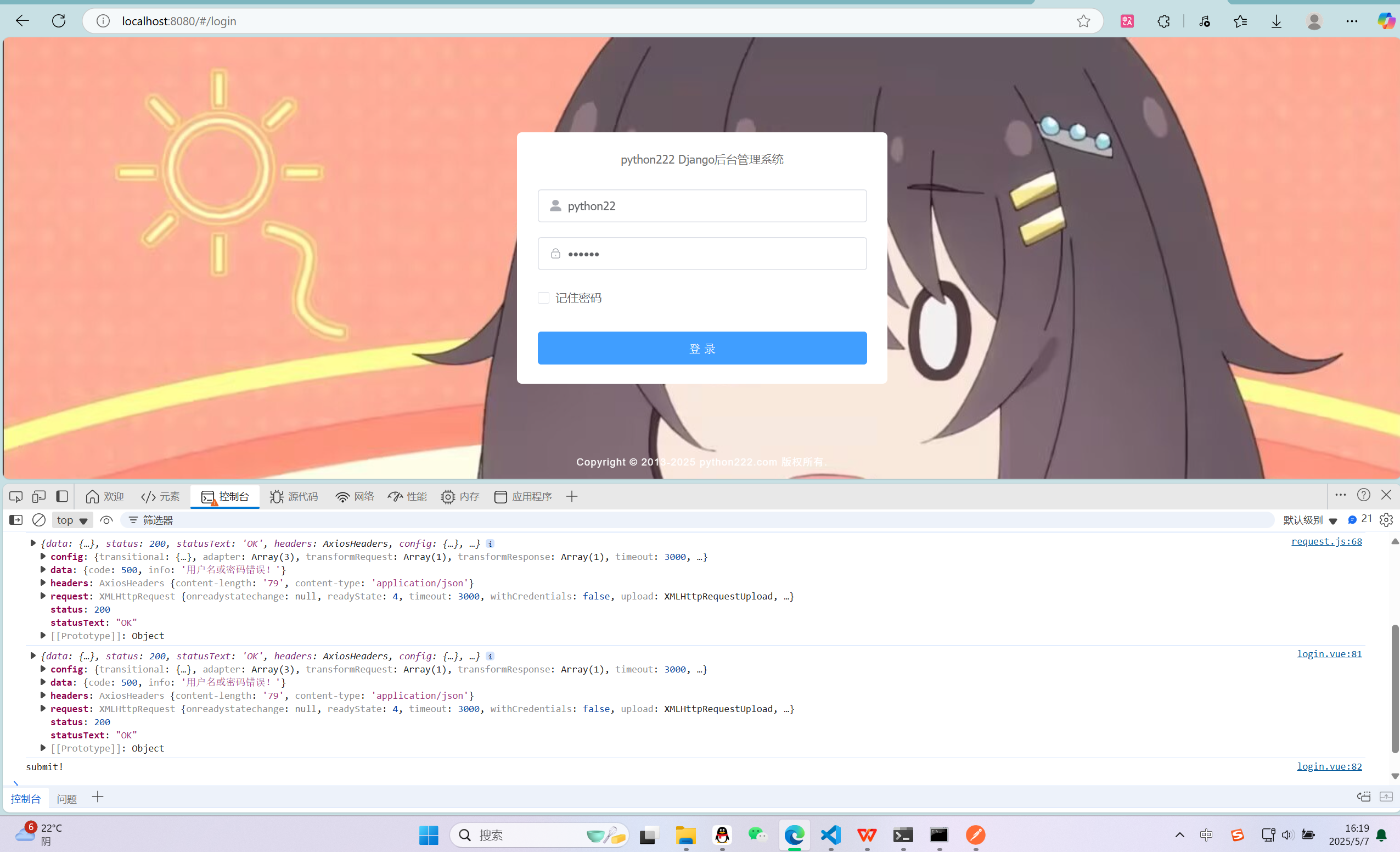




使用postman（postman仅验证后端，很方便） 验证后端没问题



前端测试也无误



上传Github

临时关闭了ssl安全验证

git config --global http.sslVerify false