

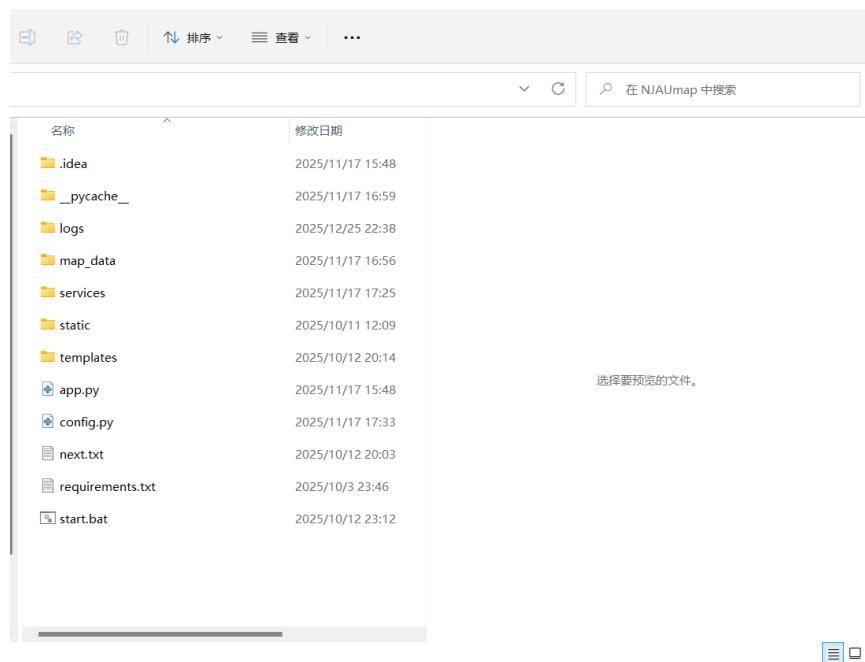
1 准备工作

系统文件，软件：Visual Studio Code，运行环境：python、数据库文件。

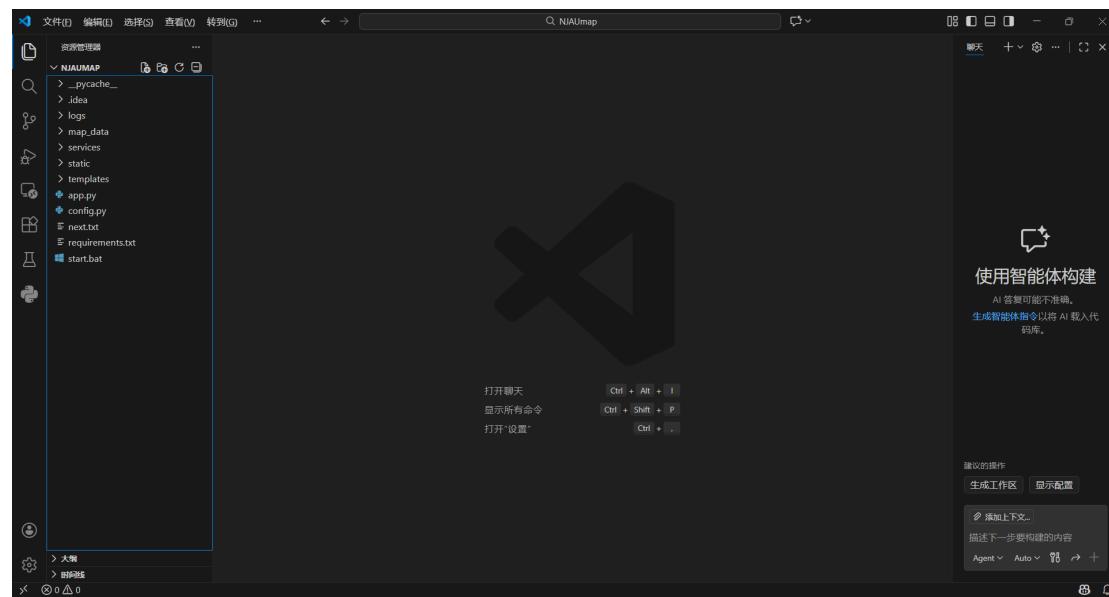
2 部署过程

①在服务器内安装 Visual Studio Code；

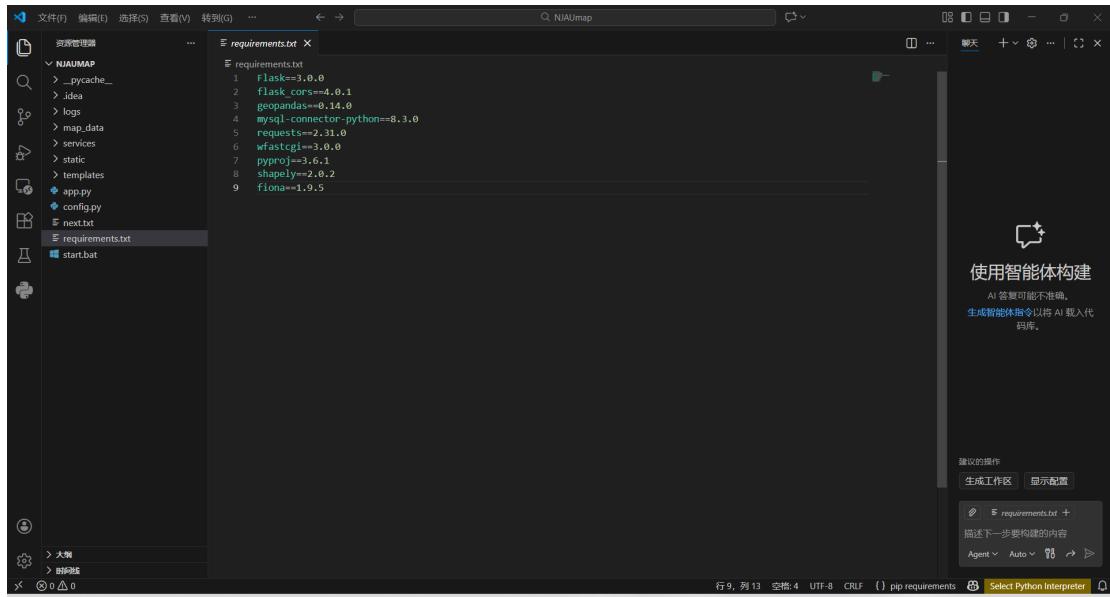
②准备好所需代码



③用 Visual Studio Code 打开代码



④在 Visual Studio Code 中运行 pip install -r requirements.txt，安装所需要的依赖包



⑤准备好高德地图的 API、deepseek API 配置以及数据库配置

```
# 高德地图API配置
AMAP_API_KEY = '25c9b7fc'
AMAP_ROUTE_API_URL = 'ht
BICYCLING_API_URL = 'ht

# 高德路线规划类型
AMAP_ROUTE_TYPES = {
    'walking': '/walking',
    'driving': '/driving',
    'bicycling': '/bicycl
}

# deepseek API配置
DEEPSEEK_API_KEY = 'sk-6
DEEPSEEK_ROUTE_API_URL =
```

```
# mysql数据库配置
DB_CONFIG = {
    "host": "localhost",          # 数据库服务器地址
    "user": "root",              # 数据库用户名
    "password": "",               # 数据库密码
    "database": "",               # 数据库名称
    "port": 3306,                 # MySQL服务端口
}
```

⑥开启服务器；

```
运行 > 启动调试 F5
    1. 以非调试模式运行 Ctrl+F5
    2. 停止调试 Shift+F5
    3. 重启调试 Ctrl+Shift+F5
    4. REM 设置日志文件路
    5. set "LOG_DIR=logs"
    6. set "%TIMESTAMP%=%date%-%time::~-2%%time::~-3%%time::~-6%%"
    7. set "LOG_FILE=%LOG_DIR%\%LOG_TIMESTAMP%.log"
    8. 遍过程 F10
    9. REM 检查Logs文件夹
   10. if not exist "%LOG_DIR%" (
   11.     REM 记录启动时间
   12.     echo Program start time: %date% %time% >> "%LOG_FILE%"
   13.     echo Running app...
   14.     echo To exit, press Ctrl+C
   15.     REM 运行app.py并捕获输出
   16.     python app.py >> "%LOG_FILE%"
   17.     REM 检查程序是否正常结束
   18.     if %errorlevel% eq 0 (
   19.         echo app.py started successfully! Log file: %LOG_FILE%
   20.     ) else (
   21.         echo app.py startup failed, please check the log file: %LOG_FILE%
   22.     )
   23. 
   24. 
   25. 
   26. 
   27. 
   28. REM 记录结束时间
   29. echo Program end time: %date% %time% >> "%LOG_FILE%"
   30. 
   31. REM 按任意键退出
   32. pause
```

使用智能体构建
AI 答复可能不准确。
生成智能体指令以将 AI 载入代码。

⑦测试项目是否成功在服务器上运行。