

Windows 定时运行 Python 脚本指南（以 limitup_tracker.py 为例）

① 打开任务计划程序

1. 按 **Win + S** 搜索 **任务计划程序**，点击打开。
 2. 左侧选择 **任务计划程序库**。
 3. 右侧点击 **创建任务....**。
-

② 基本信息设置

- **名称**：**LimitUp Tracker**
 - **描述（可选）**：每日自动获取涨停数据并生成 Excel
 - **勾选**：
 - “使用最高权限运行”（确保脚本在后台也能执行）
 - “配置适用于：Windows 10/11”
-

③ 触发器设置（Triggers）

1. 点击 **新建...**
2. **开始任务**：选择 **按计划**
3. **设置**：
 - 每天执行
 - 时间：例如 **09:00**（可根据交易日需求调整）
4. **高级设置**：
 - 勾选 **如果错过计划时间，尽快运行任务**（保证关机后开机也能补跑）
5. 点击 **确定** 保存触发器。

⚠ 注意：每天定时触发依赖电脑开机，如果关机，只有勾选“错过计划时间立即运行”才能补执行。

④ 操作设置（Actions）

1. 点击 **新建...**
2. **操作**：选择 **启动程序**
3. **程序/脚本**：

- 写 Python 的绝对路径，例如：

```
C:\Python310\python.exe
```

4. 添加参数：

- Python 脚本路径，例如：

```
"C:\Users\Jason\limitup_tracker.py"
```

注意加双引号，避免路径中有空格导致找不到文件

5. 点击 确定 保存操作。

5 条件设置 (Conditions)

- 默认勾选“仅在交流电源时启动”，如果电脑笔记本可调节是否勾选。
 - 可根据需求取消勾选“仅在计算机空闲时启动”。
-

6 设置 (Settings)

- 勾选：
 - 如果任务失败，重新启动（可设置尝试次数）
 - 按需重新启动直到成功
 - 停止任务如果运行超过 xx 小时（可选）
 - 点击 确定 完成任务创建。
-

7 测试任务

1. 在任务计划程序里找到新建任务 **LimitUp Tracker**。
 2. 右键选择 **立即运行**，确认脚本能成功生成 Excel 文件。
 3. 检查 **limitup_tracker** 文件夹，确保文件生成正常。
-

8 日常注意事项

- 脚本路径、Python 路径使用绝对路径，避免找不到文件。
- 可以在脚本里添加日志输出：

```
with open("limitup_tracker/log.txt", "a") as f:  
    f.write(f"{datetime.now()} 执行完成\n")
```

- 脚本执行失败时可查看日志或任务计划程序中的历史记录。
-

这样设置后：

- 每天固定时间自动运行，获取涨停数据。
 - 关机后开机也能补跑，不会错过交易日数据。
 - Excel 文件按日期保存，不会覆盖历史数据。
-