

# WP脚本连续执行方法

## 方法一

利用python的sys库，模拟cmd执行。以下代码为例：

```
import os
import sys

path = r'D:\ATC\WPATC\test'
ret = []
for root, dirs, files in os.walk(path):
    for filepath in dirs:
        ret.append(os.path.join(filepath))
ret.sort(key=lambda x: int(x.split('.')[0]))

for i in ret:
    print(i)
    os.system('python -m airtest run test\\' + i + ' --device Windows:///')
#os.system('python -m airtest run 渲染摄像头到1号位置.air --device Windows:///')
```

通过方法：`os.walk(path)` 遍历目标路径下的所有.air文件夹，并返回list。

根据[airtest官方文档](#)提供的信息，可以在命令中输入以下指令，执行.air文件：

```
airtest run test.air --device Windows:///?title_re=Unity.*
```

其中 `test.air` 是要执行的.air文件名，`--device Windows:///?title_re=Unity.*` 则是指定的需要连接的窗口，在WP脚本使用中，如果是直接连接Windows窗口而不指定特定窗口，则使用 `--device Windows:///` 即可。

需要说明的是，在WP脚本中存在着 **有序脚本** 的操作，这会对.air文件的 **命名** 提出要求。由于 `os.walk(path)` 方法，获取的列表顺序是无序列表，如文件名中带有要实现有序执行的.air文件，则在文件夹进行命名时，以有序数字开头，并使用一下示例方法

```
filelist.sort(key=lambda x: int(x.split('.')[0]))
```

对获取到的文件列表按照文件夹开头的数字进行排序。

## 方法二

使用WP连跑小工具进行连跑操作。

环境要求：

```
1.python
2.pip install airtest
```

小工具界面如下：



该工具使用pyqt5开发，具有以下功能：

- 1.检索使用Airtest开发的ui自动化脚本，能被检索到的文件夹名格式为 `T*.air`；
- 2.对已获取的文件夹列表进行刷新；
- 3.勾选需要进行连跑的脚本，并开始运行，运行中的Log将打印在右侧Log窗口中；
- 4.对Log进行检查，直接显示脚本运行结果为 OK/FAIL（有bug，修改中）；
- 5.将Log窗口中的所有内容保存为txt，并自动命名为 `{%Y-%m-%d-%H-%M}测试log.txt`；
- 6.停止功能;