

Mac_Setup_Guide

Mac 上运行 Python 自动化脚本的准备工作

本指南将帮助您在 macOS 系统上设置环境，以运行您的 `main_portable.py` 自动化脚本。我们将使用 Python 虚拟环境来管理项目依赖，确保环境的清洁和可移植性。

1. 安装 Homebrew (如果尚未安装)

Homebrew 是 macOS 上一个非常方便的包管理器，可以简化 Python 和其他工具的安装。

打开您的“终端” (Terminal) 应用，然后运行以下命令：

```
/bin/bash -c "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/HEAD/  
install.sh)"
```

按照终端提示完成安装。

2. 安装 Python 3 (通过 Homebrew)

通过 Homebrew 安装最新版本的 Python 3。

在终端中运行：

```
brew install python
```

安装完成后，您可以通过 `python3 --version` 检查 Python 版本。

3. 创建项目目录和虚拟环境

将您的 `main_portable.py`、`state.json`（如果已存在）和其他相关文件放在一个独立的文件夹中。然后，在该文件夹内创建并激活一个 Python 虚拟环境。

1. **创建项目文件夹**（如果尚未创建，请替换 `YourAutomationProject` 为您的项目名称）：

```
mkdir ~/Documents/YourAutomationProject
cd ~/Documents/YourAutomationProject
```

然后将您的脚本文件复制到此文件夹中。

2. **进入项目文件夹**（如果已在其中，可跳过）：

```
cd /path/to/your/automation/folder
# 例如：cd ~/Documents/YourAutomationProject
```

3. **创建虚拟环境**（名为 `.venv`）：

```
python3 -m venv .venv
```

4. **激活虚拟环境**：

```
source .venv/bin/activate
```

激活成功后，您的终端提示符前面会显示 `(.venv)`，表示您当前正在虚拟环境中工作。

4. 配置脚本

在安装依赖项之前，您需要配置脚本以使用您自己的 API 密钥。

1. 打开 `main_portable.py` 文件。

2. 找到以下行：

```
VOLC_SECRETKEY = "YOUR_VOLC_SECRET_KEY"
```

3. 将其中的 `YOUR_VOLC_SECRET_KEY` 替换为您自己的火山引擎密钥。您可以在火山引擎的[API 密钥管理页面](#)找到您的密钥。

5. 安装所需的 Python 库

在已激活的虚拟环境中，安装脚本所需的所有 Python 库：

```
pip install pandas requests playwright pynput openpyxl
```

6. 安装 Playwright 浏览器驱动

Playwright 需要下载其专用的浏览器驱动才能进行自动化操作。

在已激活的虚拟环境中，运行以下命令：

```
playwright install
```

此命令将下载 Chromium、Firefox 和 WebKit 的驱动。您的脚本明确使用了 Chrome (`channel='chrome'`)，因此此步骤是必需的。

7. 首次运行：登录并保存会话 (`state.json`)

您的脚本需要一个名为 `state.json` 的文件来保存猎聘网的登录状态。如果此文件不存在或登录已过期，您需要执行此步骤。

1. 修改 `main_portable.py` 文件：

- 用文本编辑器打开 `main_portable.py`。
- 找到文件末尾的 `if __name__ == "__main__":` 部分。
- **取消注释**（删除行首的 `#`）
`asyncio.run(save_session())` 这一行。
- **注释掉**（在行首添加 `#`）`run_with_pause_control()` 这一行。

修改后，该部分应如下所示：

```
if __name__ == "__main__":  
    # -----  
    # !!! 重要 !!!  
    # -----  
    # 如果是第一次运行或登录过期，取消注释下一行来保存会话  
    asyncio.run(save_session())  
  
    # 保存会话后，注释掉 save_session()，然后运行 main()  
    # run_with_pause_control()
```

2. 运行脚本进行登录：

在已激活的虚拟环境中，运行：

```
python3 main_portable.py
```

3. 脚本会打开一个浏览器窗口。请**手动登录猎聘网**。
4. 登录成功后，回到终端，按 **Enter** 键。脚本将保存登录状态到 `state.json` 文件并退出。

8. 正常运行脚本

完成登录会话保存后，您需要将 `main_portable.py` 文件改回正常运行模式。

1. 恢复 `main_portable.py` 文件：

- 用文本编辑器再次打开 `main_portable.py`。

- **注释掉** `asyncio.run(save_session())` 这一行。
- **取消注释** `run_with_pause_control()` 这一行。

恢复后，该部分应如下所示：

```
if __name__ == "__main__":  
    # -----  
    # !!! 重要 !!!  
    # -----  
    # 如果是第一次运行或登录过期，取消注释下一行来保存会话  
    # asyncio.run(save_session())  
  
    # 保存会话后，注释掉 save_session()，然后运行 main()  
    run_with_pause_control()
```

2. 运行脚本：

在已激活的虚拟环境中，运行：

```
python3 main_portable.py
```

现在，脚本将开始执行自动化流程。

重要提示：

- **每次开始新终端会话时**，如果您想运行此脚本，请务必先进入您的项目文件夹，然后运行 `source .venv/bin/activate` 来激活虚拟环境。
- 当您完成工作并想退出虚拟环境时，可以在终端中输入 `deactivate`。