

# Windows 下製作可攜式 Python 的方法

[#python](#) [#portable](#)

## [portable \(3 Part Series\)](#)

[1 製作可攜版本的 Arduino IDE](#) [2 製作可攜式版本的 Visual Studio Code](#) [3 Windows 下製作可攜式 Python 的方法](#)

從 Python 3.5 開始，官方提供有 Windows 下的 [embeddable package](#) 版本，這是為了讓你可以將 Python 內嵌在自己的應用程式而特別設計的最精簡版本 Python 執行環境，原本的用途是讓你：

1. 幫應用程式加上腳本功能，可以讓使用者用 Python 撰寫你自己應用程式的腳本。
2. 讓你用 Python 寫的程式可以成為單獨的應用程式，不需要使用者自己安裝 Python 環境。

這也表示我們可以使用這個版本製作一個獨立可攜的 Python 執行環境，有點像是虛擬環境，但是更獨立可攜。以下就來說明如何使用這個版本的 Python。

## 下載並解開 Python Embeddable Package

在個別版本的下載頁面中，都會有 Windows embeddable package 檔案可以下載，以下我們以 [3.8.10](#) 版為例，下載 [Windows embeddable package \(64-bit\)](#) 後，直接解開在你想要放置的資料夾即可，底下以 D:\temp\python-3.8.10-embed-amd64 為例，只要切換到該資料夾下，就可以執行 Python 環境了：

```
# cd \temp
# cd .\python-3.8.10-embed-amd64\
# ./python
Python 3.8.10 (tags/v3.8.10:3d8993a, May 3 2021, 11:48:03) [MSC v.1928
64 bit (AMD64)] on win32
>>>
```

不過因為是精簡版本，所以並不包含 Tcl/Tk 以及使用該套件衍生的其他工具，像是 IDLE 等等，也沒有包含說明檔案，更重要的是，也沒有 pip 工具，由於本來就不是要拿來給使用者開發程式，所以拿掉上述這些也是合情合理。

## 安裝 pip 工具

理論上只要把剛剛解開的資料夾複製到其他機器上就可以執行 Python 程式，不過如果想要將其當成可攜式的 Python 環境，當然得有 pip 工具，才能安裝相依的套件。要安裝 pip 工具，只要下載對應版本的 [get-pip.py](https://bootstrap.pypa.io/get-pip.py) 檔案後執行即可，以下採用 [curl](https://curl.hq.apache.org/) 工具程式將下載檔案存放在與 Python 相同的資料夾下：

```
# curl https://bootstrap.pypa.io/get-pip.py -O
```

使用資料夾內的 Python 執行 **get-pip.py**：

```
# ./python get-pip.py
Collecting pip
  Using cached pip-22.3.1-py3-none-any.whl (2.1 MB)
Collecting setuptools
  Using cached setuptools-65.6.3-py3-none-any.whl (1.2 MB)
Collecting wheel
  Using cached wheel-0.38.4-py3-none-any.whl (36 kB)
Installing collected packages: wheel, setuptools, pip
  WARNING: The script wheel.exe is installed in
'D:\temp\python-3.8.10-embed-amd64\Scripts' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress
this warning, use --no-warn-script-location.
  WARNING: The scripts pip.exe, pip3.8.exe and pip3.exe are installed in
'D:\temp\python-3.8.10-embed-amd64\Scripts' which is not on PATH.
  Consider adding this directory to PATH or, if you prefer to suppress
this warning, use --no-warn-script-location.
Successfully installed pip-22.3.1 setuptools-65.6.3 wheel-0.38.4
```

注意：若是下載回來的 **get-pip.py** 版本太新，與 Python 版本不相容，可能會看到以下的錯誤：

```
# .\python.exe .\get-pip.py
ERROR: This script does not work on Python 3.6 The minimum supported Python
version is 3.7. Please use https://bootstrap.pypa.io/pip/3.6/get-pip.py
instead.
```

不用擔心，因為錯誤訊息裡面已經有提供對應 Python 版本的 **get-pip.py** 下載網址了，只要重新下載後執行即可。

安裝完成後 `pip` 執行檔會放在 `Scripts` 資料夾下，不過如果直接執行 `pip`，會看到以下的錯誤訊息：

```
# .\Scripts\pip3 list
Traceback (most recent call last):
  File "runpy.py", line 194, in _run_module_as_main
  File "runpy.py", line 87, in _run_code
  File
"D:\temp\python-3.8.10-embed-amd64\Scripts\pip3.exe\__main__.py", line
4, in <module>
ModuleNotFoundError: No module named 'pip'
```

這是因為預設並不會啟用自行安裝的套件，還要進行額外的設定。

## 啟用自行安裝的套件

在解開的資料夾下有一個 `python38._pth` 檔，這個檔案的主檔名結尾的數字代表 Python 的版本編號，像是 `3.8.x` 版就是 `38`，若是 `3.10.x` 版就會是 `310`。預設的內容如下：

```
python38.zip
.

# Uncomment to run site.main() automatically
#import site
```

最後一行必須取消註解，讓 `import site` 將 `Lib/site-package` 加入系統搜尋模組的路徑中，Python 才能在該路徑下找到我們自行安裝的套件，包含剛剛安裝的 `pip`，請將該檔案修改如下後存檔：

```
python38.zip
.

# Uncomment to run site.main() automatically
import site
```

這樣就可以正常執行 `pip` 了：

```
# .\Scripts\pip3 list
Package    Version
```

```
-----
pip          22.3.1
setuptools  65.6.3
wheel        0.38.4
```

要安裝模組也沒有問題：

```
# .\Scripts\pip3 install pyquery
Collecting pyquery
  Using cached pyquery-1.4.3-py3-none-any.whl (22 kB)
Collecting lxml>=2.1
  Downloading lxml-4.9.2-cp38-cp38-win_amd64.whl (3.9 MB)
    _____ 3.9/3.9 MB 4.4
MB/s eta 0:00:00
Collecting cssselect>0.7.9
  Downloading cssselect-1.2.0-py2.py3-none-any.whl (18 kB)
Installing collected packages: lxml, cssselect, pyquery
Successfully installed cssselect-1.2.0 lxml-4.9.2 pyquery-1.4.3
```

確認安裝的模組可以正常運作：

```
# .\python.exe
Python 3.8.10 (tags/v3.8.10:3d8993a, May  3 2021, 11:48:03) [MSC v.1928
64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> import pyquery
>>>
```

## 結語

利用上述的方法，就可以自行製作特定版本的可攜式 **Python** 環境，像是我自己常會需要把寫好的 **Python** 程式交給非開發人員使用，這時搭配這樣的可攜式環境就可以簡化交付流程，只要將整個資料夾壓縮後提供給使用者解開即可，非常方便。

補充：另一種可攜式 **Python** 環境的選擇是使用 [WinPython](#)。