

Appendix

# C

## 將 Python 打包成執行檔

許多人想要將完成的 Python 應用程式分享給其他人使用，但麻煩的是不是每一台電腦都會安裝 Python 以及相關的模組。這時，就必須要使用包裝工具將 Python 應用程式打包成 **exe** 執行檔，才可以在其他機器上執行。

**PyInstaller** 有兩種製作 **exe** 檔的方式：

第一種方式是將製作出的檔案皆放在同一個目錄下，這是預設的方式，稱為 **onedir**。

第二種方式是加上「**-F**」參數將製作出的檔案包裝成一個獨立的 **exe** 執行檔，稱為 **onefile**。

Python 初學特訓班





## C.1 打包前的準備工作

想要將完成的 Python 應用程式，分享給自己的好友，卻發現好友的電腦中並沒有安裝 Python 以及相關的模組。這時，就必須要使用包裝工具將 Python 應用程式打包成 **exe** 執行檔，才可以在其他機器上執行。

### C.1.1 安裝 PyInstaller

PyInstaller 可以將 Python 程式打包成執行檔，首先必須安裝 PyInstaller，如下：

```
pip install https://github.com/pyinstaller/pyinstaller/archive/develop.zip
```

我們嘗試以「`pip install pyinstaller`」安裝，但實作會出現錯誤，並無法順利安裝。

建議以「<https://github.com/pyinstaller/pyinstaller/archive/develop.zip>」從網站中安裝較新的 **pyinstaller** 版本。

### C.1.2 PyInstaller 使用方式

PyInstaller 有兩種製作 **exe** 檔的方式。

#### onedir 方式：

第一種方式是將製作出的檔案皆放在同一個目錄下，這是預設的方式，稱為 **onedir**。語法：

```
pyinstaller 應用程式
```

例如：

```
pyinstaller Hello.py
```

#### onefile 方式：

第二種方式是加上「**-F**」參數將製作出的檔案包裝成一個獨立的 **exe** 執行檔，稱為 **onefile**。語法：

```
pyinstaller -F 應用程式
```

例如：

```
pyinstaller -F Hello.py
```

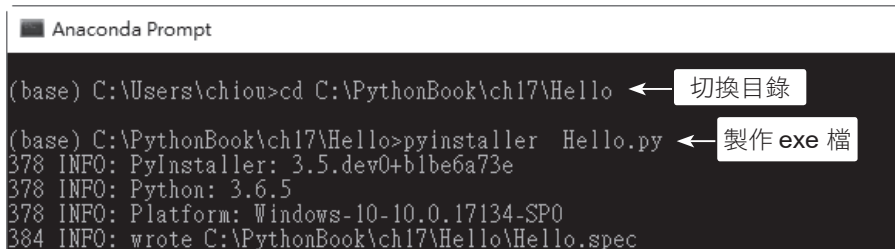
## C.2 實作 exe 執行檔

我們先以 <chC\Hello\Hello.py> 這個較簡單的程式來實作。程式內容如下：

```
程式碼：chC\Hello\Hello.py  
print("Hello Python")  
a=input("請按任意鍵結束!")
```

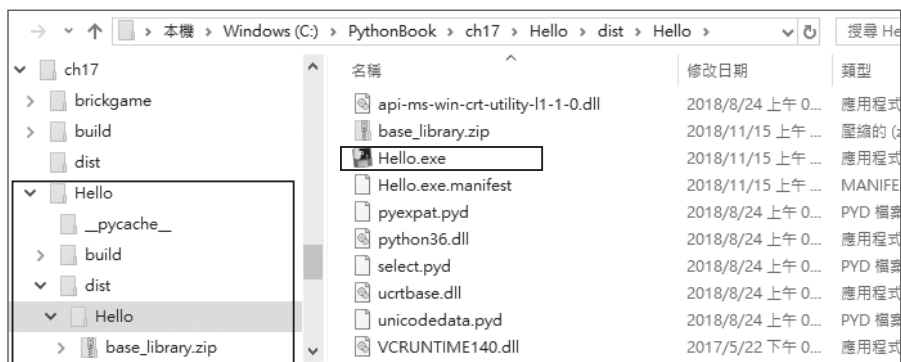
### C.2.1 實作 onedir 的 exe 執行檔

首先開啟 Anaconda Prompt 視窗，切換到 <Hello.py> 應用程式的目錄，然後以「pyinstaller Hello.py」將 <Hello.py> 應用程式打包成 onedir 的 exe 執行檔。



```
Anaconda Prompt  
(base) C:\Users\chiou>cd C:\PythonBook\ch17\Hello  
(base) C:\PythonBook\ch17\Hello>pyinstaller Hello.py  
378 INFO: PyInstaller: 3.5.dev0+b1be6a73e  
378 INFO: Python: 3.6.5  
378 INFO: Platform: Windows-10-10.0.17134-SP0  
384 INFO: wrote C:\PythonBook\ch17\Hello\Hello.spec
```

完成之後會在 <Hello.py> 程式所在的目錄產生 <\_\_pycache\_\_>、<build> 和 <dist> 三個目錄，其中 <dist> 目錄中建立了 <Hello> 目錄，<Hello> 目錄中產生許多如 .dll 的相關檔案以及 <Hello.exe> 執行檔，只要將 <Hello> 整個目錄複製到其他的電腦，就可以在其他的電腦上執行。





## C.2.2 實作 onefile 的 exe 執行檔

接著以範例 <chC\Hello2\Hello2.py> 程式實作。程式內容如下：

```
程式碼：chC\Hello2\Hello2.py
```

```
print("Hello Python")
```

```
a=input(" 請按任意鍵結束!")
```

請開啟 Anaconda Prompt 視窗，切換到 <Hello2.py> 應用程式的目錄，然後以

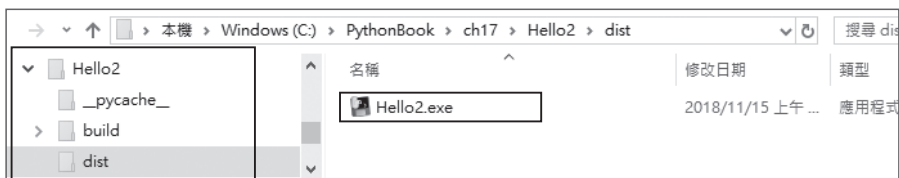
「pyinstaller -F Hello2.py」將 <Hello2.py> 程式打包成 onefile 的 exe 執行檔。

```

Anaconda Prompt

(base) C:\PythonBook\ch17>cd C:\PythonBook\ch17\Hello2 ← 切換目錄
(base) C:\PythonBook\ch17\Hello2>pyinstaller -F Hello2.py ← 製作 exe 檔
63 INFO: PyInstaller: 3.5.dev0+b1be6a73e
63 INFO: Python: 3.6.5
64 INFO: Platform: Windows-10-10.0.17134-SP0
69 INFO: wrote C:\PythonBook\ch17\Hello2\Hello2.spec
```

完成之後會在 <Hello2.py> 程式所在的目錄產生 <\_\_pycache\_\_>、<build> 和 <dist> 三個目錄，其中 <dist> 目錄中只建立了一個 <Hello2.exe> 執行檔，因為 pyinstaller 已將所有相關模組都包含在 <Hello2.exe> 檔中，只要將 <Hello2.exe> 檔複製到其他的電腦，就可以在其他的電腦上執行。



## C.3 包裝專題執行檔

實際應用上，Python 應用程式通常不會這麼簡單，可能還包含許多的模組，或是相關的資源如圖片、音效檔，甚至以 `selenium` 執行 `webdriver` 時，必須使用 `<ChromeDriver.exe>` 建立 Google Chrome 瀏覽器等。

### C.3.1 實作 onefile 的 mp3Play 音效播放執行檔

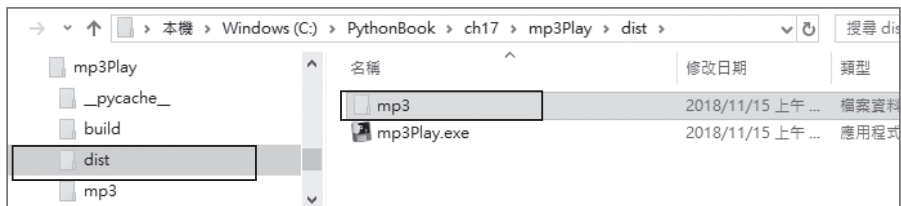
首先以 `<chC\mp3Play>` 的 `<mp3Play.py>` 為例，它是複製原來書附光碟中第 `<ch15>` 的 `<mp3Play.py>` 和 `<mp3>` 音效檔案。程式部份內容如下，這個程式中包含 `tkinter`、`pygame` 和 `glob` 模組，同時也需要 `mp3` 的檔案。

程式碼：`chC\mp3Play\mp3Play.py`

```
...略
52  ### 主程式從這裡開始 ###
53
54  import tkinter as tk
55  from pygame import mixer
56  import glob
57
58  mixer.init()
59  win=tk.Tk()
60  win.geometry("640x380")
61  win.title("mp3 播放器")
...略
69  source_dir = "mp3/"
70  mp3files = glob.glob(source_dir+"*.mp3")
```

開啟 Anaconda Prompt 視窗，切換到 `<mp3Play.py>` 應用程式的目錄，然後以「`pyinstaller -F mp3Play.py`」將 `<mp3Play.py>` 應用程式打包成 `onefile` 的 `exe` 執行檔，這個程式較複雜，編譯需要幾分鐘，請耐心等待。

完成之後會在 `<mp3Play.py>` 目錄產生 `<__pycache__>`、`<build>` 和 `<dist>` 參個目錄，其中 `<dist>` 目錄中建立了 `<mp3Play.exe>` 執行檔，但這時執行 `<mp3Play.exe>` 並無法正常播放 `mp3` 檔，因為它找不到 `mp3` 的檔案，必須將 `<mp3Play.py>` 目錄中的 `<mp3>` 目錄全部複製到 `<mp3Play.exe>` 執行檔目錄中，執行時才有 `mp3` 檔可播放。



現在您可以執行 `<mp3Play.exe>` 檔，您一定會很驚訝，這麼複雜的程式，`pyinstaller` 竟然辦到了，而且除了 `mp3` 歌曲檔案，其它相關的模組都包含在 `<mp3Play.exe>` 執行檔中。請記得要在別的電腦中執行，必須複製 `<dist>` 整個目錄，包括 `<mp3>` 的歌曲檔案也必須複製，當然您可以使用不同的 `mp3` 檔，播放不同的歌曲，也可以將 `<dist>` 名稱改為較有意義的名稱，例如 `<MyMp3Player>`。



### C.3.2 實作 onefile 的 PM2.5 下載執行檔

最後再以 `<chC\loginPM25>` 的 `<loginPM25.py>` 為例，它是複製原來書附光碟中第 `<ch06>` 的 `<loginPM25.py>` 檔案。程式部份內容如下：

程式碼：`chC\loginPM25\loginPM25.py`

```
1 from selenium import webdriver
2 from time import sleep
3 from selenium.webdriver.common.keys import Keys
4
5 url = 'https://opendata.epa.gov.tw/'
6
7 driver = webdriver.Chrome()
8 driver.maximize_window()
9 driver.get(url)
```

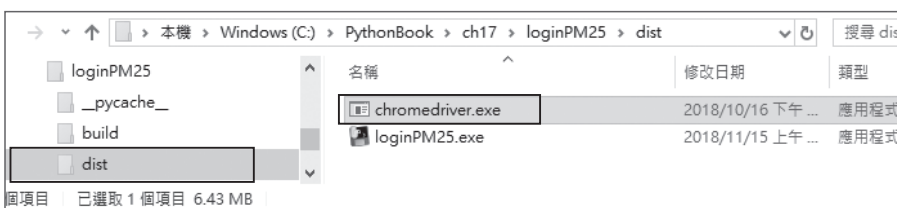
同 17.3.1 的操作，一樣以「`pyinstaller -F loginPM25.py`」將 `<loginPM25.py>` 應用程式打包成 onefile 的 `<loginPM25.exe>` 執行檔。

這個程式中必須使用 `<ChromeDrvier.exe>` 建立瀏覽器，在第 6 章中我們是將 `<ChromeDrvier.exe>` 複製到 `<C:\ProgramData\Anaconda3>` 目錄或目前專案的工作目錄中，但別人的電腦未必會安裝 Anaconda3，這個方法不可行。

其實我們將 `<ChromeDrvier.exe>` 複製到 `<C:\ProgramData\Anaconda3>` 目錄中是因為電腦的環境變數 `path` 已在安裝 Anaconda3 時加入了「`path=C:\ProgramData\Anaconda3`」的路徑搜尋設定。



因此，只要將 `<ChromeDrvier.exe>` 檔複製到 `path` 已設定的路徑即可，最簡單的方式就是將 `<ChromeDrvier.exe>` 檔複製到 `<loginPM25.exe>` 相同的目錄中，這樣就可以省略 `path` 的設定。也請記得要在別的電腦中執行，必須複製 `<dist>` 整個目錄，包括 `<ChromeDrvier.exe>` 檔，同時將 `<dist>` 目錄名稱改為較有意義的名稱。



## This image shows a full page of a handwriting practice worksheet. It consists of multiple sets of three horizontal dashed lines spaced evenly down the page, providing a guide for letter height and placement. The background is plain white, and there are no other markings or text present.