**附錄Ｅ**

**安裝第三方模組**

**E-1：命令提示字元**

**E-2：系統多重安裝使用pip**

**E-3：導入模組安裝更新版模組**

**E-4：安裝更新版模組**

**E-5：列出所安裝的模組**

**E-6：刪除模組**

**E-7：找尋更多模組**

**E-8：安裝新版pip**

Python是一個免費的軟體，因此吸引許多公司行號以它作為公司的官方語言，同時也吸引了很多公司行號或個人將所開發的模組放到網頁上供其他人下載使用。通常我們將這些放在網路上可以下載使用的模組稱第三方模組。

**E-1：命令提示字元**

**E-1-1：基本觀念**

安裝第3方模組需要進入Windows作業系統的命令提示字元視窗，這個環境就是早期電腦的DOS環境，如下所示：



如果你的電腦只有一套Python版本，可以直接在上述輸入要安裝的外部模組，例如：如果想要安裝send2trash模組，只要在上述視窗輸入下列指令即可。

pip install send2trash

如果你的電腦有多套Python版本，可以參考E-2節，指定模組要安裝的版本。

**E-1-2:了解你目前電腦安裝多少個Python版本**

可以使用「where python」了解目前電腦安裝多少版本。



**E-1-3：了解命令提示字元視窗的Python版本**

可以使用「python --version」了解目前命令提示字元視窗執行Python的版本。



從上述可以看到筆者電腦命令提示字元視窗執行Python的版本是3.7版，如果有一個程式使用了新版的Python模組，例如：是用Python 3.11版。這個模組如果在命令提示字元視窗執行，就會有錯誤，因為此程式所用的模組Python 3.7不支援，如果我們期待可以在命令提示字元視窗執行，必須指出Python的版本，這時可以使用完整路徑啟動Python，此路徑可以參考前一小節。另一種方法是使用下列方式啟動Python:

py -3.11

**E-2：系統多重安裝使用pip**

筆者電腦先前安裝了Python 3.7，後來安裝了Python 3.8、Python 3.9 … 甚至3.12，發現原先安裝在Python 3.7的模組無法在後來的新版Python環境執行，必須重新到命令提示字元視窗執行安裝模組，這時安裝的語法如下：

py -3.8 -m pip install send2trash

py -3.9 -m pip install send2trash

py -3.12 -m pip install send2trash

上述相當於是在py指令後方加上安裝版本，上述是假設所安裝的模組是send2trash。

**E-3：導入模組安裝更新版模組**

模組安裝完成後，未來可以在程式前面執行import指令導入模組，同時可以測試是否安裝成功，如果沒有錯誤訊息就表示安裝成功了。

import 模組名稱

import send2trash # 導入send2trash為實例

**E-4：列出所安裝的模組**

可以使用list列出所安裝的模組，如果使用’-o’可列出有新版本的模組。

pip list # 列出安裝的模組

pip list –o # 列出有新版本的模組

**E-5：安裝更新版模組**

未來如果有更新版，可用下列方式更新至最新版模組。

pip install -U 模組名稱 # 更新至最新版模組

**E-6：刪除模組**

安裝了模組之後，若是想刪除可以使用uninstall，例如：若是想刪除basemap，可以使用下列指令。

pip uninstall basemap

**E-7：找尋更多模組**

可以進入<https://pypi.org>。

**E-8：安裝新版pip**

安裝好Python後，pip會被自動安裝，如果不小心刪除可以到下列網址下載。

<https://pypi.org/project/pip/>