推薦序

推薦給熟悉 Excel，又希望自己數據分析能力再更上好幾層的你

這是一本特別的 Excel + Python 的書，甚至是一本特別的爬蟲與數據分析的書。

相信大家對 Excel 都不陌生，也知道怎麼樣做基本的資料整理、繪製圖表等工作。有些進階的使用者，可能會使用一點點的 VBA 讓 Excel 更有彈性、更能自動完成許多複雜的事，這些基本 Excel 使用，甚至 VBA 入門，在市面上都有不少的書。

這本書最特別的地方有兩個，第一個是舉了非常多實務上的例子，而且手把手帶著你完成。聽起來好像平凡無奇，但是大多數（即使那些稱為「實戰」的書）都不會這樣從頭帶到尾。因為這中間有很多細節，比如說我們要收集數據資料，大多有帶實務例子的，都直接把清理好的檔案給你。你並不會知道要做到這樣子是多麼辛苦，雖然這麼做常常也是擔心怕非常個別化的狀況隱藏了真正要的主題。但本書可以說又做到手把手教學，又不會過度專注在細節上，跟著做可以讓自己覺得真的完成一個完整的專案。

前面提到，我們已經有不少 VBA 的書籍。但即使 VBA 用得再熟，還是沒有像 Python 這樣的程式語言有彈性、有那麼多在數據分析乃至機器學習的套件。而這本書教你怎麼樣可以讓 Python 幫助你的 Excel 如虎添翼，讓你可以很方便的用現在熱門的數據分析、機器學習的方法，分析你的數據。這樣子的資訊，真的很難找到相關書籍的介紹。

相信這本書對於 Excel 做數據分析的朋友，可以再更上一層發揮 Excel 的潛力！你會驚訝得發現原來 Excel 可以做到這麼酷炫的事！推薦給熟悉 Excel，又希望自己數據分析能力再更上好幾層的你。

蔡炎龍
國立政治大學應用數學系副教授
http://moocs.nccu.edu.tw/instructor/yenlung

推薦序

從 Excel VBA 到 Python 爬蟲之路

許多投資人是從複製網頁資料貼到 Excel 工作表，開始進行投資分析的，一開始在資料種類及數量不多的狀況下，利用複製貼上勉強堪用。在對資料的種類及數量的需求越來越大，利用複製貼上這種枯燥無聊的動作，就變成一種沉重的負擔。為了要解決這個困境不得不透過 VBA 撰寫爬蟲程式，來協助快速取得資料。

VBA 自從 1993 年面市以來，就沒有大幅度更新其功能，在今天面對海量資料的情況下，對大數據資料處理能力有限，欠缺機械學習及深度學習能力，讓使用者不得不思考是否有更好的程式語言來輔助或偕同作業，我想 Python 應該會是明確的選擇。

對非資訊相關科系的人來說，學會 Excel VBA 撰寫爬蟲程式，已經是很不簡單的事，要從 Excel VBA 到 Python 爬蟲之路顯然有其難度，是否存在快速的捷徑呢？ Facebook 好友廖敏宏這本新書肯定是解方之一，從 Python 環境安裝、編輯環境、Excel 與 Python 相互操作等，從基礎入門到實戰演練鉅細靡遺。

從實際操作網頁、判斷網頁資料所在位置、分析封包尋找資料來源、URL 與傳送參數驗證，然後分別提供 Excel VBA 抓資料及 Python 抓資料的完整程式碼，讓 VBA 的使用者，參照兩種語言的寫法，進而快速掌握 Python 爬蟲的關鍵能力。

非常開心有機會寫序來推薦本書，有了這本工具書肯定會節省大量自行摸索的時間，推薦給想從 Excel VBA 邁入 Python 爬蟲之路的讀者們，這是你唯一的選擇！

Tivo Chang

Tivo168 教你 Excel 輕鬆投資理財

https://facebook.com/tivo168.Excel.Invest

推薦序

讓網路機器人自動幫你做每天繁瑣的工作

認識敏宏是因為我為了學習爬蟲而找到他的部落格「iInfo 資訊交流」，雖然以前在國防部的時間大部分也是從事資訊相關，但對於透過爬蟲程式自動蒐集的技術，我是全然門外漢，由於他無私地分享透過爬蟲自動蒐集資訊的技術，讓我在建置價值投資分析的 i-stock 可以突破資料蒐集的障礙，也因為透過大量的資料蒐集與整合分析，讓我在投資的路上可以看到更多一般人看不到的面向。

在投資的領域，我發現很多投資績效不錯的朋友，不但具備財商知識專業外，往往也具有一定的資訊整合與分析能力，所以敏宏成立的 Line 群組也匯集了不少這樣的高手，敏宏的前一本著作《Excel VBA 實戰技巧｜金融數據 x 網路爬蟲》就是值得推薦的寶典，我除了買紙本書外，也買了電子書可以隨時查閱，而此次推出的這本著作，更是鉅細靡遺的詳細說明如何透過 Excel 與 Python 來實作十大金融網站的資訊擷取，透過一步步詳細的說明，引領我們學習爬蟲的技能，只要熟悉這十大範例，相信足以可以應用在國內九成以上的網站資訊擷取。

很多朋友常問我說，我不是資訊本科系的，適不適合讀這本書？我認為，如果你這輩子要上班，必須常與資訊為伍的話，花點時間學會讓網路機器人自動幫你做每天繁瑣的工作，未來就能夠節省更多的時間休閒，敏宏的書會是你學習爬蟲很好的工具書，相信無論未來應用在工作、生活或投資，都會有很大的幫助。

價值投資人：算利教官
https://facebook.com/ezlifeInstructor/

03 Excel與Python相互操作

Excel 與 Python 常被拿來相互比較，這兩個工具彼此也是常被用來相互輔助，在不同的領域下各有其愛好者與擁戴者。以下列表僅透過幾個領域比較 Excel 與 Python 兩者間的差異，若需要更詳盡了解各領域內容，請讀者們自行研究。

	Excel	Python
軟體安裝	Office 成員	需下載安裝
數據分析	內建功能	安裝套件 Pandas、Numpy、SciPy…等
視覺化圖表	內建功能	安裝套件 Matplotlib、Seaborn、Plotly
大數據資料處理能力	有限	可完全處理
機械學習、深度學習	需藉由外部程式輔助	安裝套件 Scikit-learn、Keras、Tensorflow、PyTorch
網站建置	無	安裝套件 Flask、Django 等
Console 操作支援	有	有

本章介紹如何在 Excel 與 Python 之間的操作，以「Excel 操作 Python」、「Python 操作 Excel」、「Excel、Python 相互操作」三個方向，讓讀者們了解透過 Excel 與 Python 彼此相互操作可做到更多的可能。

3.1 Excel 操作 Python

「元件物件模型」（Component Object Model, COM）是微軟在 1993 年所提出的一個規範，在此規範下，COM 元件需要有統一的介面與方法，讓不同程式語言進行呼叫操作，如此可做到跨程式語言的行程通訊（Inter-process communication, IPC），與動態建立物件（如 Excel VBA 的「前期綁定」與「後期綁定」）。目前 COM 已在微軟多項產品導入，如 IE、OutLook、Excel

等，COM 同時也是微軟多項技術的基礎，如 OLE、OLE、ActiveX、COM+、DCOM、Windows shell、DirectX、Windows Runtime。

3.1.1 Excel 呼叫 Python 的 COM

Excel VBA 欲使用非內建的函數功能，如 IE 瀏覽器連線、DOM 網頁解析…等，可使用「前期綁定」與「後期綁定」方式來輔助，「前期綁定」與「後期綁定」是透過 COM 執行外部函數。若要使用 Python 函數或模組，如 Python 開發影像辨識、機械學習，則可將這些 Python 函數或模組依據 COM 規範來封裝開發，最後再註冊至 Windows 系統登錄檔中，如此 Excel VBA 或其他程式語言（C++、C#、VB、JavaScript）就可透過 COM 執行 Python 這些函數與模組。

下面步驟介紹如何以 Python 建立 COM 元件，最後再註冊至 Windows 中，供系統與其他語言操使用。

1 宣告 Python 套件。

```
001  import pythoncom, win32com.server.register, sys
```

程式碼說明：

pythoncom：提供 Windows COM 元件操作與設定的 Python 套件。

win32com.server.register：註冊 COM 元件到 Windows 的 Python 套件。

2 建立「PythonTestCOM」類別，並加入指定 COM 介面設定。

```
001 class PythonTestCOM:
002     #將COM物件註冊為exe元件，或是DLL元件(CLSCTX_INPROC_SERVER)
003     _reg_clsctx_ = pythoncom.CLSCTX_LOCAL_SERVER
004
005     #建立COM元件的名稱
006     _reg_progid_ = "PythonTestCOM"
008     #建立一個在Windows註冊GUID用來代表COM元件
009     _reg_clsid_ = '{83E67D90-57C4-46A2-8010-35EBC7DFE9A6}'
010
```

補充說明

除了上述列表內容，還有三個在執行上要注意的地方。

1. 利用 Anaconda 建置 Excel 執行 Python 環境，上述操作方法在 Anaconda3-5.3.1 之後的版本無法適用，但原生 Python、ActivePython 無此問題。
2. Pywin32.exe 安裝時，自動把「Lib\site-packages\pywin32_system32」資料夾底下的 DLL 複製到「C:\WINDOWS\System32」資料夾底下，並執行 Python 檔案「Lib\site-packages\win32comext\axscript\client\pyscript.py」，檔案功能是註冊 Python COM 元件至 Windows 登錄檔，讓 Windows 任意程式語言執行 Python 程式碼使用。
3. Excel VBA 透過「MSScriptControl.ScriptControl」操作 Python 組成必要條件：
   (1) Pywin32 安裝。
   (2) 登錄檔登錄 Python 安裝資訊。
   (3) DLL 複製到「C:\WINDOWS\System32」。
   (4) pyscript.py 執行註冊 COM 元件到登錄檔。

3.2 Python 操作 Excel

Python 有很多第三方套件可用來操作 Excel，如 xlrd/xlwt、xlutils、openpyxl、xlwings、pyxll 與 Pandas 中 xlsxwriter 等，這些套件在讀寫方面各具優勢，但就控制 Excel 方面，Pywin32 中 win32com 提供完整的 Excel VBA 操控，也有豐富的 Windows API 可以使用，以 win32com 來做為 Python 操作 Excel 套件為最合適。

以下說明如何使用 win32com 套件的 win32com.client.Dispatch、win32com.client.DispatchEx 與 win32com.client.gencache.EnsureDispatch，並在 Python 建立 Excel 與控制 Excel VBA 程式。

- **win32com.client.Dispatch**：新增啟動 Excel，若系統中存在 Excel 程式，則會直接使用已存在的程式，不會另外新增 Excel。
- **win32com.client.DispatchEx**：新增啟動一個 Excel 程式。

Excel VBA、Python 與腳本語言操作 04

JavaScript 是網頁主要使用的腳本語言（Script Language），幾乎包辦了所有網頁特效、前端與後端同步 / 非同步溝通的主要媒介，在爬蟲的操作上，除 Python、Excel VBA、R、PHP…等，JavaScript 也是最常被用來做為主要、次要或輔助的爬蟲語言，在 Python、Excel VBA 可透過套件或內建類別函式庫，直接或間接透過 JavaScript 與 HTML、伺服器進行操作，有時使用 JavaScript 直接進行網頁操作會比在 Python、Excel VBA 上操作來得更快速、更方便。

4.1 Excel VBA 與 JavaScript

Excel VBA 中，操作 JavaScript 可由以下內建函式庫進行。

函式庫	前期綁定（前期綁定物件名稱）	後期綁定
IE	Microsoft Internet Controls (SHDocVw.InternetExplorer)	InternetExplorer.Application
DOM	Microsoft HTML Object Libaray (MSHTML.HTMLDocument)	Htmlfile
Script	Microsoft Script Control 1.0	MSScriptControl.ScriptControl

在「IE」與「DOM」函式庫，可使用「eval」函數與「execScript」方法執行 JavaScript 程式。

「eval」函數

執行「腳本語言」與取回變數數值。

```
eval(字串)
字串：JavaScript 函數名稱或變數名稱。
```

「execScript」方法

執行「腳本語言」。

```
execScript (字串, 腳本語言)
字串：「腳本語言」函數名稱。
腳本語言：JavaScript、Jscript、VBScript等。
```

觀念說明

「execScript」方法目前在各家瀏覽器上皆不支援，雖說是微軟 IE 上的產物，但從 IE11 開始之後的版本也已不再支援該功能。

參考微軟官網相關說明：

1. Compatibility changes in IE11
   https://docs.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/dev-guides/bg182625(v=vs.85)?redirectedfrom=MSDN#legacyAPIs
2. execScript method
   https://docs.microsoft.com/en-us/previous-versions/windows/internet-explorer/ie-developer/platform-apis/aa741364(v%3Dvs.85)

4.1.1 「瀏覽器」函式庫操作 JavaScript

「瀏覽器」函式庫僅支援「execScript」方法，若是需取回網頁上數值，可搭配寫入 HTML 節點，再以 DOM 取回數值資料。

```
001  Sub 輸出字串到瀏覽器()
002      With CreateObject("InternetExplorer.Application")
003          .Navigate "about:blank"
004          .Visible = True
005          Do While .readyState <> 4: DoEvents: Loop
006          .Document.parentWindow.execScript "var msg = '測試';",
"JavaScript"
007          .Document.parentWindow.execScript "document.writeln(msg);",
"JavaScript"
008          '.Quit
009      End With
010  End Sub
```

4.2 Python 與 JavaScript

網頁爬蟲的過程會遇到需要處理 JavaScript 情況，像是傳送參數或回傳數據經過 JavaScript 加密、JavaScript 解密、JavaScript 混淆、JavaScript 壓縮、JavaScript 美化 … 等，Python 中用來處理 JavaScript 有 4 種方法，如 Pyexecjs、PyV8、Js2Py、Node.js，以下選擇 PyExecjs、Node.js 作為 JavaScript 在 Python 的操作。

PyExecJS 是最常被使用的套件，它透過本機的 JavaScript 環境執行 JavaScript 程式碼，所以無須再安裝任何的 JavaScript 套件，但缺點之一就是執行效能差，原因是執行 JavaScript 程式碼前需先啟動 JavaScript 環境，所以執行 JavaScript 緩慢。

PyV8 是將 Google Chrome V8 引擎用 Python 封裝的套件，和 PyExecJS 相比是個輕量型套件，V8 是一個開源 JavaScript 引擎，負責把 JavaScript 程式碼解析轉換成機器碼並執行，因此不需要額外安裝、啟動 JavaScript 環境，就能獨立於瀏覽器之外並快速執行 JavaScript 程式碼，目前 PyV8 只支援 Python 2.6、Python 3.3，無法適用全部 Python 版本。

Js2Py 由純 Python 開發的直譯器，將 JavaScript 程式碼轉換為 Python 程式碼執行。這種方式可以擺脫呼叫 JS 環境的瓶頸，若遇到很長的混淆 JS 程式碼，轉換成 Python 程式碼的過程很有可能會出錯。

NodeJS 是一個獨立於瀏覽器之外的 JavaScript 執行環境，採用 Google 的 V8 引擎將 JavaScript 轉換成機器語言，以便在機器上直接運行。

執行 JavaScript 函數與抓取變數數值

```
001  import execjs
002
003  jscode = '''
004          function Hello(msg = 'World!!')
005          {
006              return 'Hello ' + msg;
007          }
008
009          var a = "Hello world";
010          var b = 1, c = 2;
```

05 側錄發送封包

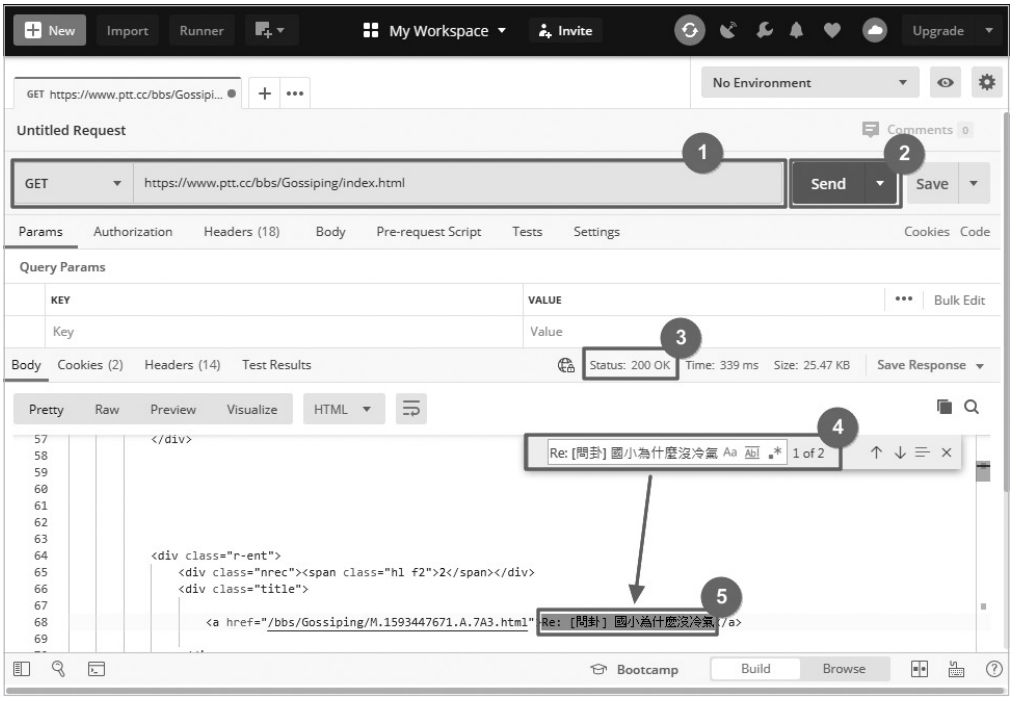
網路爬蟲就是使用程式由用戶端（Client）發送含有正確參數的封包給伺服器（Server），伺服器（Server）成功驗證參數後，回傳封包資料給用戶端（Client）的過程，若參數錯誤還可以修正重試，若遇到非參數問題，如 Content-Type 內容與編碼錯誤、或明明已填妥了 User-Agent、Referer、Cookie…等，最後還是被伺服器拒絕服務，其實這些問題都可以透過 Fiddler、Wireshark…等封包側錄工具來輔助解決。

「Chrome 開發人員工具」、「Fiddler」與「Wireshark」皆具有側錄封包功能，其使用的差異如下。

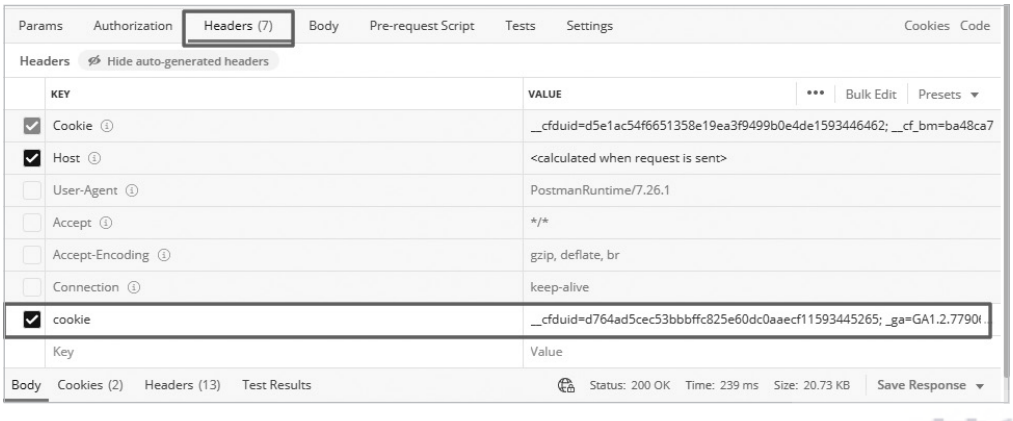
Chrome 開發人員工具	只能擷取到由「Chrome」瀏覽器發送的封包與無法側錄由其他程式發送的封包，如 Excel VBA、Python、Postman 等。
Fiddler	透過將自身轉成代理 (Proxy) 伺服器，將電腦內的所有封包轉發出去，能擷取 HTTP/HTTPS 封包，並對 HTTPS 封包解密。
Wireshark	透過擷取系統網路卡的所有封包，能擷取 HTTP/HTTPS 封包，但不能解密 HTTPS，需搭配網站 session key 才能對 HTTPS 解密察看內容。

本章介紹如何由環境設定，讓程式發送可被「Fiddler」側錄到的封包，下頁上圖為開啟「Fiddler」前，程式對外發送封包的線路連結圖；下圖為開啟「Fiddler」後，程式對外發送封包的線路連結圖。

6 匯入封包資訊後，按「Send」按鈕發送封包測試，確認「Status: 200 OK」表示成功取回資料，而「Re: [問卦] 國小為什麼沒冷氣」存在於回傳資料中，如此以該封包所有資訊為主，進行後續的參數刪減取得必要參數。



7 切換「Headers」與「Params」頁籤，將封包 Header 參數設定與傳送變數逐一刪減再進行封包發送測試，用來找出主要的設定與傳送變數。



8.5 PTT 八卦版文章載入流程

連線至「PTT 八卦版」網站觀看站內文章一定會遇到「網頁分級」畫面，點擊「我同意、我已滿十八歲進入」按鈕，進到網站後端驗證成功後，最後再轉進「PTT 八卦版」文章頁面，這樣整個過程就是網站登入標準流程，舉凡需要輸入帳號與密碼才行登入查看的網站皆可以此作為基礎，至於中間帳密如何驗證與網頁轉跳就依據開發人員設計，以下就來分析「PTT 八卦版」如何運作。

不同與以往，由於登入類型網站在帳號密碼輸入後，要進行帳密驗證與網頁轉跳，因此側錄封包就需要改用「Network」下的「All」頁籤來觀察所有封包的執行過程進而分析。

8.5.1 進入「PTT 八卦版」如同網站登入流程

網頁操作

1 開啟「Chrome 開發人員工具」，清除封包列表，準備側錄封包。

❶ 點擊「Network」頁籤。