



SCU LawTech

Class #04

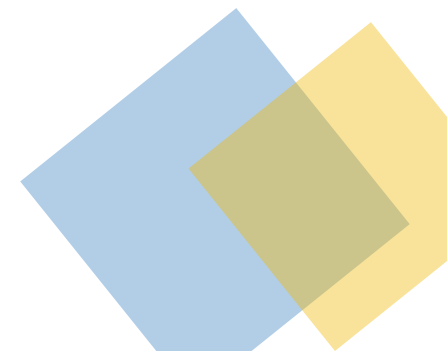
Content

- ◆ 爬虫基础
- ◆ 套件
- ◆ 环境

爬虫基础

概念

- 網路爬蟲 / 網頁蜘蛛 / 網路機器人 / 網頁追逐者
- 藉由網頁之链接（URL / API）對伺服器進行 HTTP 請求，並解析其回傳內容，以獲取所需資料。
- 簡單來說，抓取網頁內容。

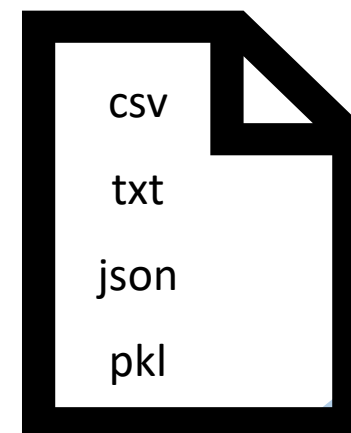
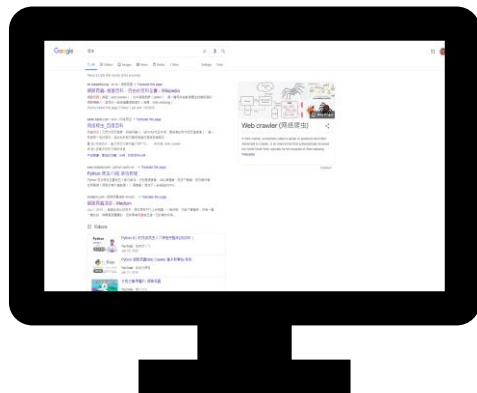


執行流程 – 用戶請求流程

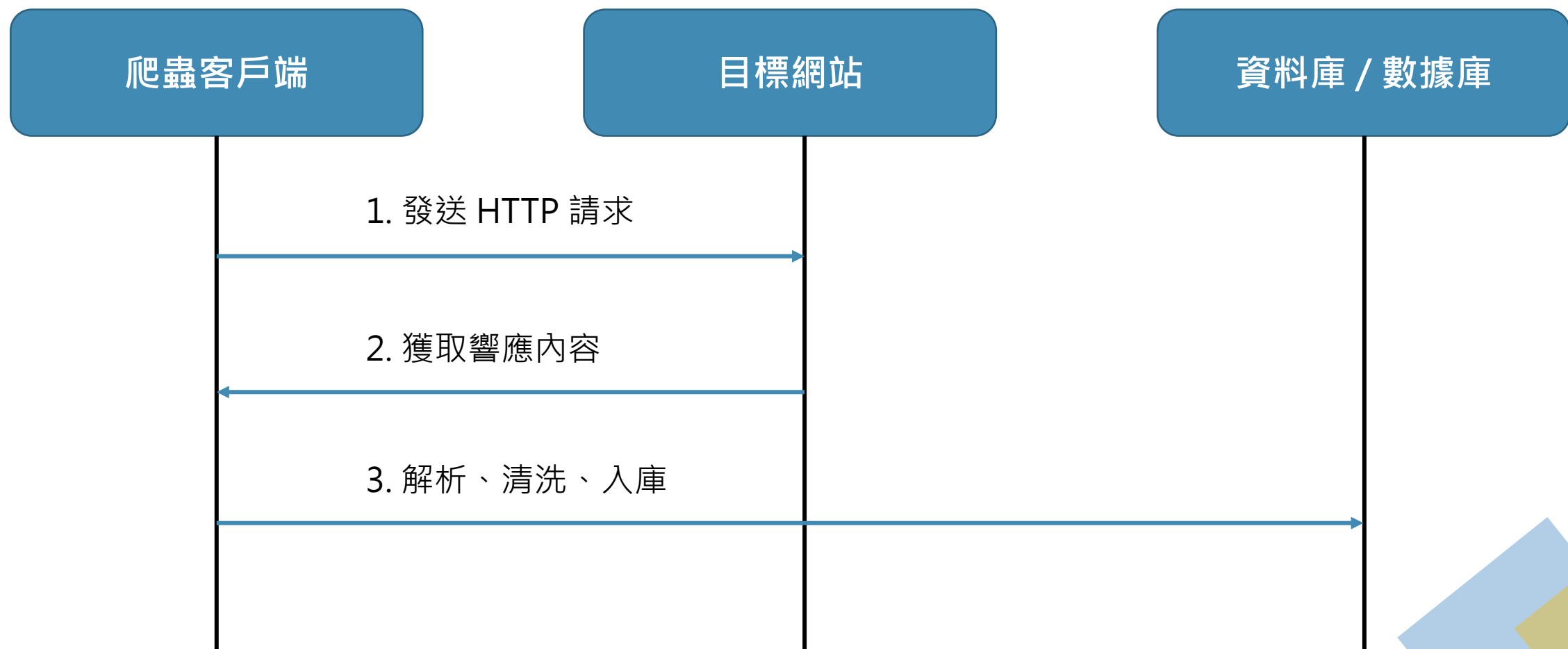
登入網頁

搜尋資料

下載資料



執行流程 – 程式請求流程



第 1 階段 (程式請求流程) — 我的資料在哪？

- 哪個網站有我想要的資料？
- 網站是否需要登錄或身份驗證？
- 內容品質是否可靠？
- 網站內容的更新頻率為何？
- IPO 與資料結構設計

在這個資訊大爆炸的時代，若進行網路爬蟲時，須嚴謹考量目標網站之內容重複程度及完整性，以利於提升分析效率。

第 1 階段 (程式請求流程) — 我的資料在哪？

- 哪個網站有我想要的資料？
- 網站是否需要登錄或身份驗證？
- 內容品質是否可靠？
- 網站內容的更新頻率為何？
- IPO 與資料結構設計

在進行爬蟲時，除了了解目標網站外，須確保在獲取資料前，是否進行登錄行為或任何身份驗證行為，其行為可能為：

- reCAPTCHA
- CAPTCHA
- Session
- Cookies 等

由於上述行為將提升爬蟲難度，因此在撰寫爬蟲前，須將該因素納入考量。

第 1 階段 (程式請求流程) — 我的資料在哪？

- 哪個網站有我想要的資料？
- 網站是否需要登錄或身份驗證？
- 內容品質是否可靠？
- 網站內容的更新頻率為何？
- IPO 與資料結構設計

在確認目標網站後，需確保爬取內容之品質具可靠性。反之，資料複雜高或來源不明的網站將造成分析結果不準確。

第 1 階段 (程式請求流程) — 我的資料在哪？

- 哪個網站有我想要的資料？
- 網站是否需要登錄或身份驗證？
- 內容品質是否可靠？
- 網站內容的更新頻率為何？
- IPO 與資料結構設計

在了解資料可靠性及目標網站後，對於即時性資料需要掌握其更新頻率，除了能夠掌握資料的流動性，還有助於自動化的機制設定。

第 1 階段 (程式請求流程) — 我的資料在哪？

- 哪個網站有我想要的資料？
 - 網站是否需要登錄或身份驗證？
 - 內容品質是否可靠？
 - 網站內容的更新頻率為何？
- IPO 與資料結構設計

Input-Process-Output Model 的應用有助於建立更完整的程式導向流程及程式碼本身的再利用性及易讀性。

在確認規劃儲存資料後，需針對須儲存的資料進行結構設計。除了能有效優化儲存空間，還能夠在完成爬取資料後進行更複雜的操作。

第 2 階段

(程式請求流程)

—

尋找消失的 API / URL, 我要請求咯！

URL / API	溝通方式	常用工具
-----------	------	------

- 你的鏈接不是你要的鏈接？
- 是否有現成套件或 API 提供？
- 是否提供手機版頁面？

在尋找目標網站時，由於使用的 HTTP 溝通方式不同或網頁設計不同，導致搜尋引擎所顯示的網址不完全是目標網站的網址。

因此可以透過使用開發人員工具檢視網頁原始碼來驗證其相關資料的溝通來源及其溝通方式。

第 2 階段

(程式請求流程)



尋找消失的 API / URL,
我要請求咯！

URL / API	溝通方式	常用工具
-----------	------	------

- 你的鏈接不是你要的鏈接？
- 是否有現成套件或 API 提供？
- 是否提供手機版頁面？

在爬蟲前，若相關網站有提供 **API** 或現成套件獲取資料，在限制條件允許下，則可不考慮使用爬蟲形式獲取資料。

第 2 階段

(程式請求流程)

—

尋找消失的 API / URL, 我要請求咯！

URL / API	溝通方式	常用工具
-----------	------	------

- 你的鏈接不是你要的鏈接？
- 是否有現成套件或 API 提供？
- 是否提供手機版頁面？

在檢視 URL 過程中，若原有目標網站在嘗試數次無法破解後，可嘗試尋找該網站之手機版（不是所有網站皆有手機版），並嘗試尋找手機版網站的目標網址。

第 2 階段

(程式請求流程)

—

尋找消失的 API / URL,
我要請求咯！

URL / API

溝通方式

常用工具

OPTIONS

GET

HEAD

POST

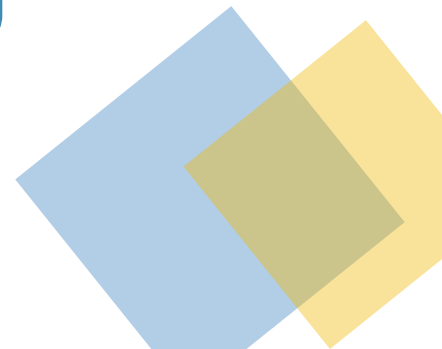
PUT

DELETE

TRACE

CONNECT

***以上溝通方式 (HTTP Method) 定義於 HTTP 1.1 版本



第 2 階段

(程式請求流程)

—

尋找消失的 API / URL,
我要請求咯！

URL / API

溝通方式

常用工具

GET vs POST

	GET	POST
網址差異	網址會帶有 HTML Form 表單的參數與資料。	資料傳遞時，網址並不會改變。
資料傳遞量	由於是透過 URL 帶資料，所以有長度限制。	由於不透過 URL 帶參數，所以不受限於 URL 長度限制。
安全性	表單參數與填寫內容可在 URL 看到。	透過 HTTP Request 方式，故參數與填寫內容不會顯示於 URL。

第 2 階段

(程式請求流程)

—

尋找消失的 API / URL,
我要請求咯！

URL / API

溝通方式

常用工具

GET vs

```
GET /?id=010101 HTTP/1.1
Host: xxx.toright.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-TW; rv:1.9.2.13) Gecko
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-tw,en-us;q=0.7,en;q=0.3
Accept-Encoding: gzip,deflate
Accept-Charset: UTF-8,*
Keep-Alive: 115
Connection: keep-alive
```

第 2 階段

(程式請求流程)

—

尋找消失的 API / URL,
我要請求咯！

URL / API

溝通方式

常用工具

vs POST

```
POST / HTTP/1.1
Host: xxx.toright.com
User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; zh-TW; rv:1.9.2.13) Gecko
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: zh-tw,en-us;q=0.7,en;q=0.3
Accept-Encoding: gzip,deflate
Accept-Charset: UTF-8,*
Keep-Alive: 115
Connection: keep-alive

Content-Type: application/x-www-form-urlencoded
</code><code>Content-Length: 9
id=010101
```

第 2 階段

(程式請求流程)

一

尋找消失的 API / URL,
我要請求咯！

URL / API

溝通方式

常用工具

HTTP 狀態碼 (Status Codes)

狀態碼	說明 (英)	說明 (中)
200	OK	成功
301	Moved Permanently	目標網頁移到新網址(永久轉址)
302	Found(Moved Temporarily)	暫時轉址
304	Not Modified	已讀取過的圖片或網頁，由瀏覽器緩存中讀取
400	Bad Request	伺服器無法處理這個 Request
401	Unauthorized	需身分驗證，如 SSL key or htaccess password

第 2 階段

(程式請求流程)



尋找消失的 API / URL,
我要請求咯！

URL / API	溝通方式	常用工具
-----------	------	------

HTTP 狀態碼 (Status Codes)

狀態碼	說明 (英)	說明 (中)
403	Forbidden	無讀取權限，可能是 IP 被阻檔或是伺服器限制
404	Not Found	伺服器未找到目標網址，檔案不存在
500	Internal Server Error	伺服器發生錯誤：可能是 htaccess 有錯
502	Bad Gateway	伺服器的某個服務沒有正確執行
503	Service Unavailable	伺服器當掉
504	Gateway Timeout	伺服器上的服務沒有回應

第 2 階段

(程式請求流程)

—

尋找消失的 API / URL,
我要請求咯！

URL / API

溝通方式

常用工具

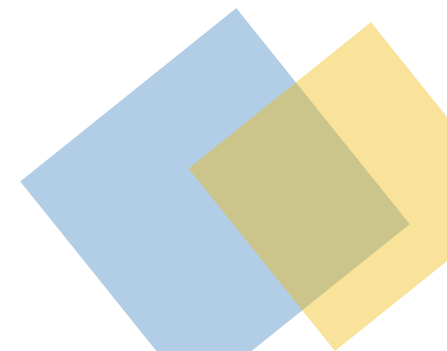
回傳資料格式

HTML

XML

JSON

特定項目格式



第 2 階段

(程式請求流程)

—

尋找消失的 API / URL,
我要請求咯！

URL / API

溝通方式

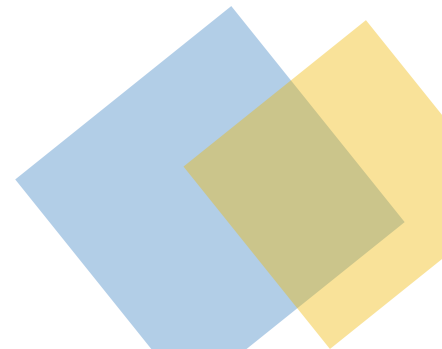
常用工具

程式
請求

urllib3 / requests

模擬
請求

selenium



(程式請求流程)

是的，

我要斷開一切的牽連~

	年度	字號	案號	類型	裁判日期	裁判案由	金額
0	109	台上	1719	刑事判決	109.04.15	違反證券交易法等罪	61785382
1	108	台上	4056	刑事判決	109.04.15	違反證券交易法等罪	573105930
2	108	台上	16	刑事判決	109.01.16	違反證券交易法等罪	372933046
3	109	台抗	46	刑事裁定	109.01.09	違反證券交易法等罪不服再執行羈押	898763655
4	107	台上	846	民事判決	108.03.27	請求損害賠償	75615095

[illegible]

資料解析：清整、提取的過程

第 3 階段

(程式請求流程)

— 是的，
我要斷開一切的牽連~

資料解析工具

工具
語言

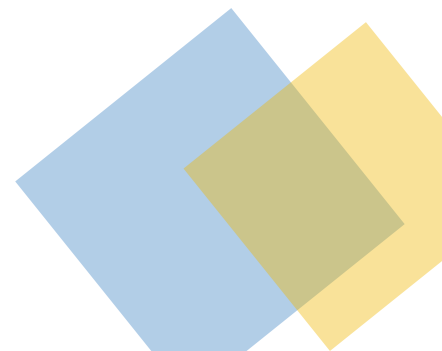
BeautifulSoup /
PyQuery /
JSON / Scrapy

CSS
選擇器

selenium

路徑
語言

Selenium /
lxml



第 3 階段

(程式請求流程)

— 是的，
我要斷開一切的牽連~

資料儲存

Excel
儲存

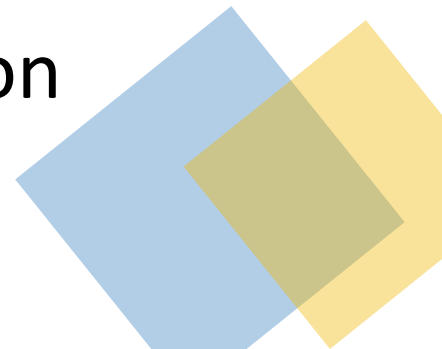
pandas /
built-in write
function

資料庫
儲存

Mysql-connector /
API

TXT
儲存

built-in write
function



第 4 階段

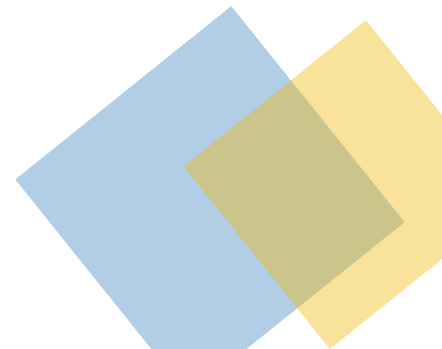
(程式請求流程)

— 不是完了嗎？
怎麼還要繼續呀！

優化程式

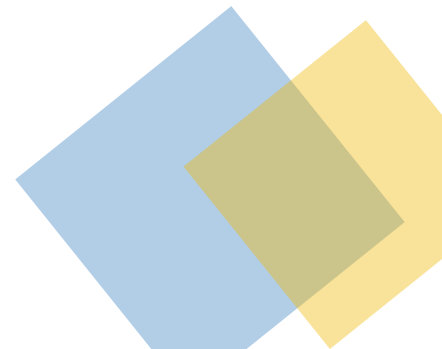
完成一個爬取的流程僅僅只是最小可行性 (MVP) 的實現，我們還需按照對於資料的需求進行一系列的規劃與設計：

- 模組化程式設計 (Design Pattern)
- 定時定期的爬取觸發機制 (Cron job)
- 異常處理設計 (Error Handling)
-



爬蟲的禮貌

1. 不要太快進行下個請求，請善用休息函式（`time.sleep(休息秒數)`）
2. 不要把不需要的資料也爬下來（若請求允許彈性查詢）
3. 不要違反 Robots 協議和網站規則，請在請求時附上該有的參數



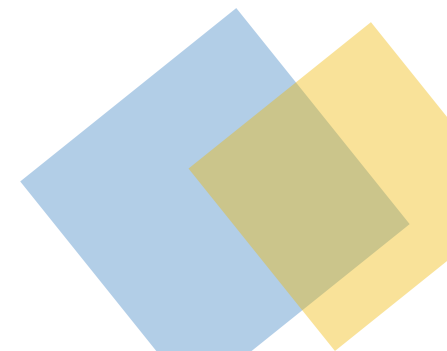
爬蟲的應用場域

項目
預訂

交易
策略

資料
分析

應用
程序



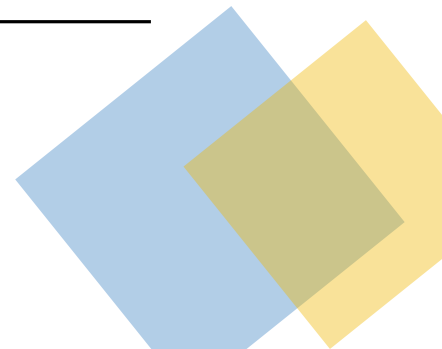
法律看爬蟲！

- 你的爬蟲會送你進監獄嗎？

<https://www.mdeditor.tw/pl/2qTa/zh-tw>

- 靠挖掘別人家的資料數據來賺錢，「網路爬蟲」這個行為合法嗎？

<https://www.techbang.com/posts/75284-is-the-internet-crawler-legal-china-and-the-united-states-have-different-views#top>



套件

Python 套件庫

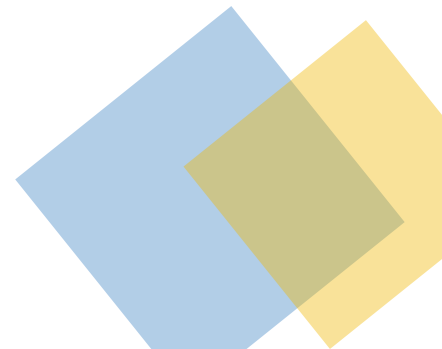


<https://pypi.org/>

Python生態系的 PyPI (The Python

Package Index) 是一個軟體套件儲存庫

(software package repository)。



套件管理器

類別	conda	pip
管理	二進位	wheel 或源碼
需要編譯器	No	Yes
語言	Any	Python
虛擬環境	支持	通過 virtualenv 或 venv 等支持
套件來源	Anaconda Repo 和 Cloud	PyPi

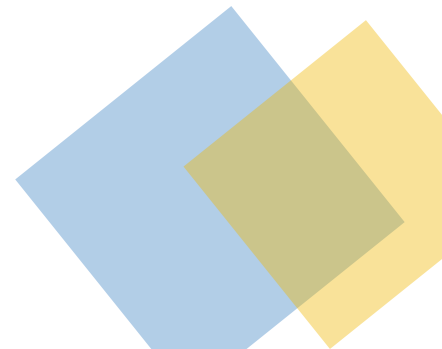
Python 套件安裝語法

- 方法 1：在 Cell 中輸入 `!pip install <套件名稱>`，然後執行

```
In [ ]: !pip install pandas
```

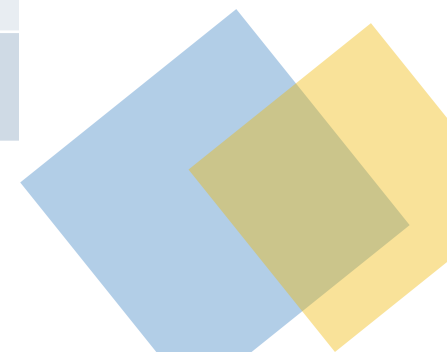
- 方法 2：在終端機中輸入 `pip install <套件名稱>`，然後按 ENTER 鍵

```
(base) C:\Users\sefx5>pip install pandas
```



本學年需下載的指定套件

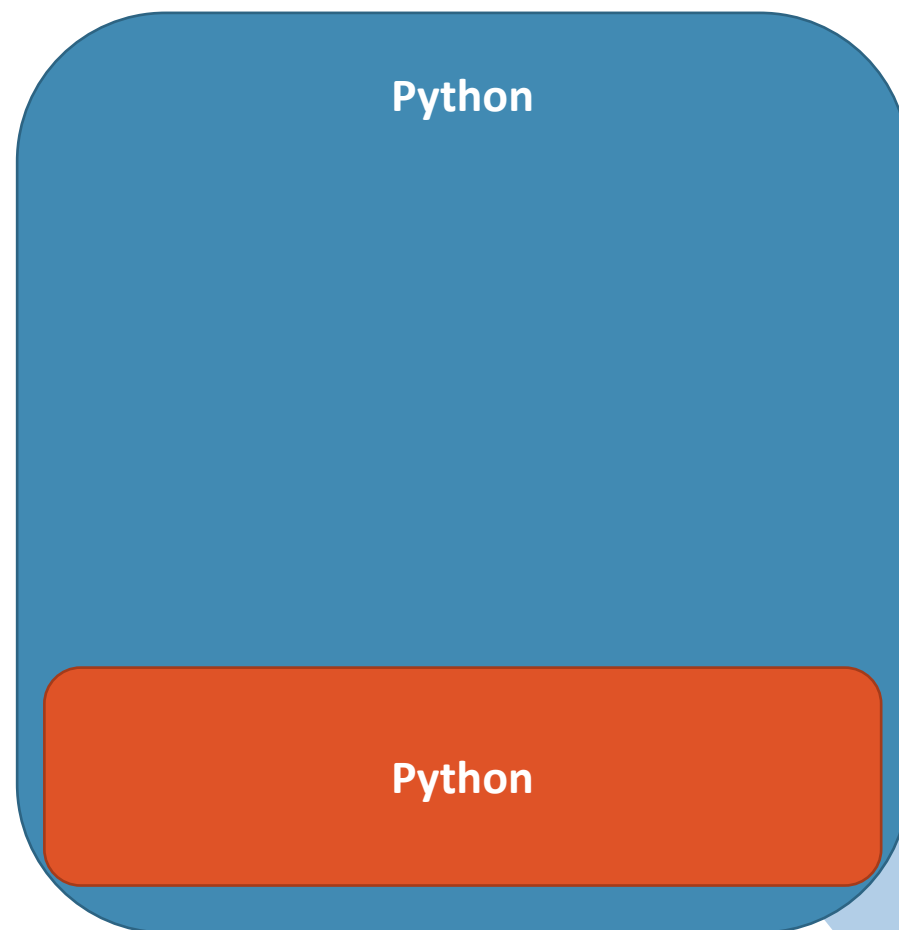
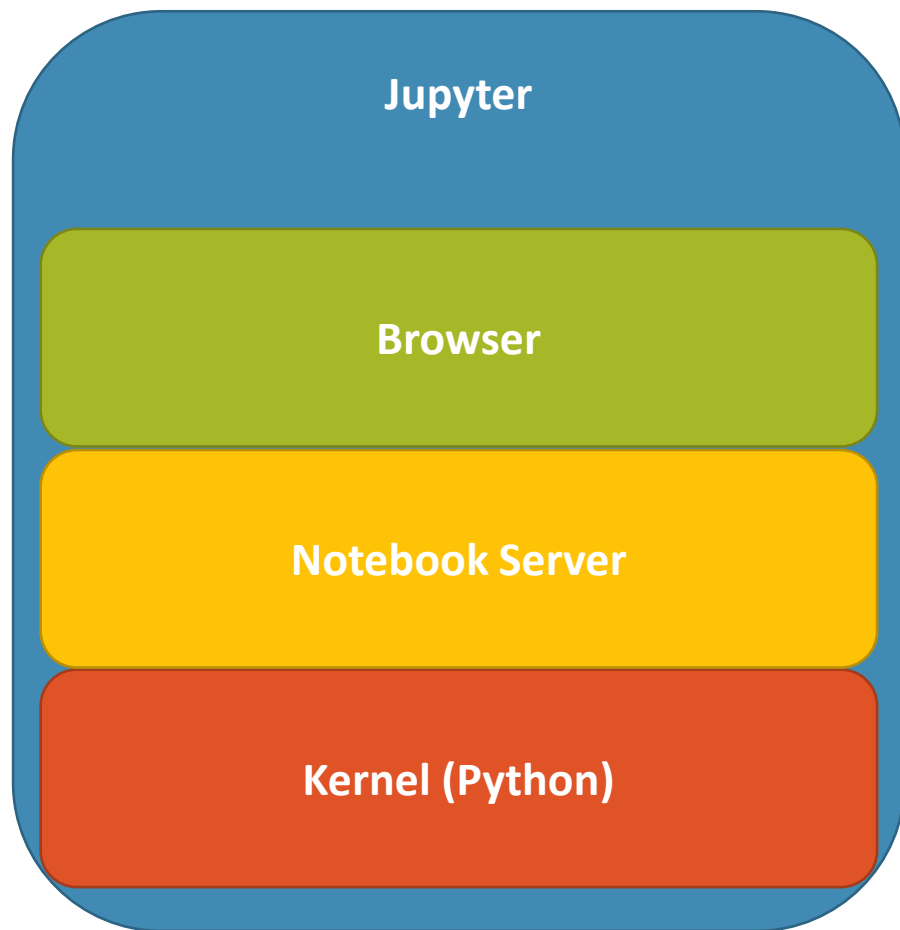
安裝語法（套件）	用途
<code>pip install pandas</code>	資料處理、探勘
<code>pip install numpy</code>	向量處理
<code>pip install requests</code>	爬蟲
<code>pip install selenium</code>	爬蟲
<code>pip install lxml</code>	爬蟲定位工具
<code>pip install beautifulsoup4</code>	爬蟲定位工具
<code>pip install plotly</code>	繪圖





環境

Jupyter 與 python 環境





THANKS