

Python網路爬蟲入門

-財經為例

David Chiu
2016/05/08

關於我



- 大數軟體有限公司創辦人
- 前趨勢科技工程師
- ywchiu.com
- 粉絲頁
<https://www.facebook.com/largitdata>
- Machine Learning With R Cookbook
<https://www.packtpub.com/big-data-and-business-intelligence/machine-learning-r-cookbook>

課程資料

- 所有課程補充資料、投影片皆位於
 - <https://github.com/ywchiu/pyfinance>

課前預備

安裝Anaconda

Anaconda for Windows

選擇Python 2.7 版

PYTHON 2.7	PYTHON 3.5
WINDOWS 64-BIT GRAPHICAL INSTALLER 349M	WINDOWS 64-BIT GRAPHICAL INSTALLER 361M
Windows 32-bit Graphical Installer 296M	Windows 32-bit Graphical Installer 296M

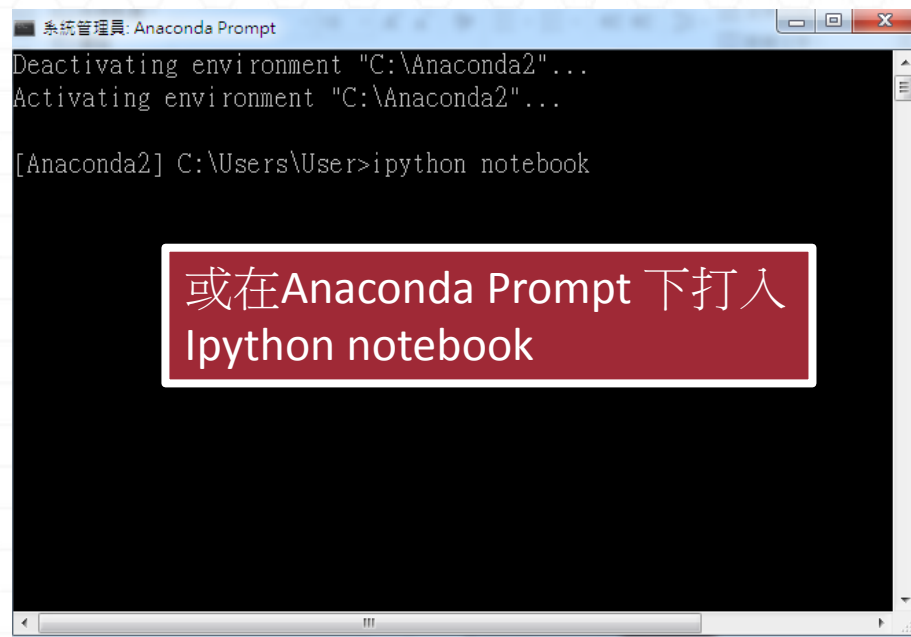
Behind a firewall? Use these [zipped Windows installers](#).

Windows Anaconda Installation

1. Download the graphical installer.
2. Double-click the .exe file to install Anaconda and follow the instructions on the screen.
3. Optional: [Verify data integrity with MD5](#)

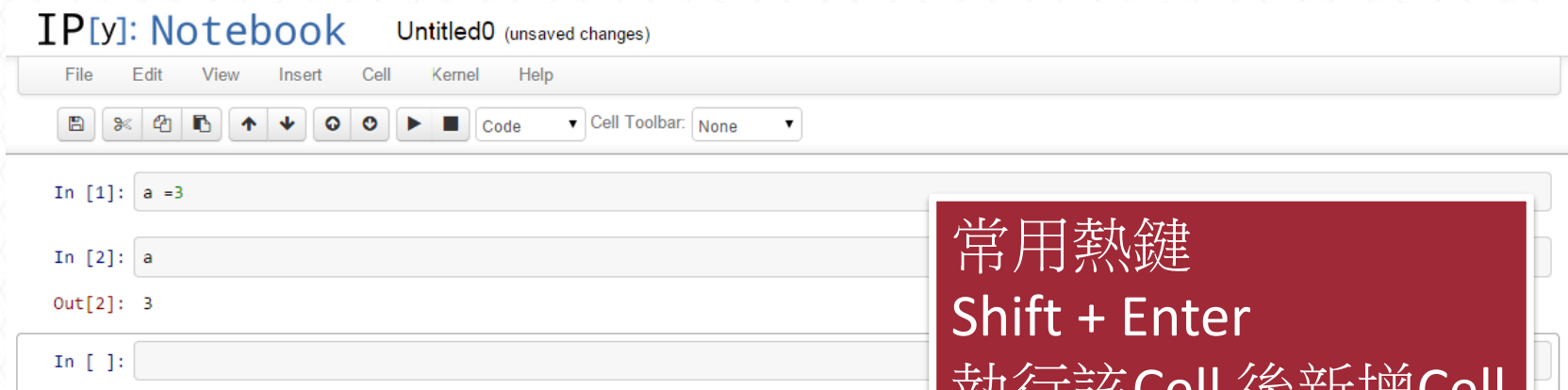
<https://www.continuum.io/downloads>

使用 Jupyter (Ipython Notebook)



啟用 Jupyter (Ipython Notebook)

- 在命令列下打:
 - ipython notebook
 - 自動開啟瀏覽器後便可瀏覽 (預設為localhost:8888)
- 可匯出.ipynb, .py 各種不同格式檔案
- 瀏覽快捷鍵 Help -> Keyboard Shortcuts



常用熱鍵
Shift + Enter
執行該Cell 後新增Cell

Python v.s. Java

Java

```
/* example 1 */  
public static void main(String[]  
args){  
    for(int i=0; i< 10; i++)  
        System.out.print(i);  
}
```

- 執行速度較Python 為快
- 使用{}分隔區塊
- 需要宣告變數型態
- 可以透過Compiler 檢查錯誤
- 使用/**/做註解

Python

```
"example1"  
for i in range(1,11):  
    print i,
```

- 開發速度較快
- 使用indent 替代 {}
- 不須宣告變數型態
- 只能在runtime 檢查錯誤
- 以#與"或" 做註解

資料爬取

透過分析數據擬定策略才能找到聖杯

- 在做任何分析之前，必定要蒐集足夠的數據做分析，才能擬定高勝率策略
- 除了購買數據外，任何人都可以透過撰寫ETL (Extract, Transformation, Loading) 程序自動化蒐集資訊



將非結構化數據轉變為結構化數據

公開資訊觀測站

登錄 | 資訊項目 | 精華版2.0 | 重大訊息 | English

請輸入公司代號或簡稱 搜尋 代號查詢 回首頁

基本資料 負債報表 股東會及股利 公司治理 財務報表 重大訊息與公告 營運概況 投資專區 認購(售)權證 債券 資產證券化

負債報表

基本資料

股東會及股利

TDR股利分派情形(101年起適用)

除權公告

股東會及除權息日曆

法人說明會一覽表

財務報表

按IFRSs後

綜合損益表

資產負債表

財務報表經監事人承認情形

會計師查核(核閱)報告

各產業EPS統計資訊

按IFRSs前

財務預測

綜合損益表

市場別 上市 年度 104 季別 3 搜尋

列印網頁 刷新網頁 問題回報 目上頁

上市公司第三季資料

註：依證券交易法第36條及證券期貨局相關函令規定，財務報告申報期限如下：

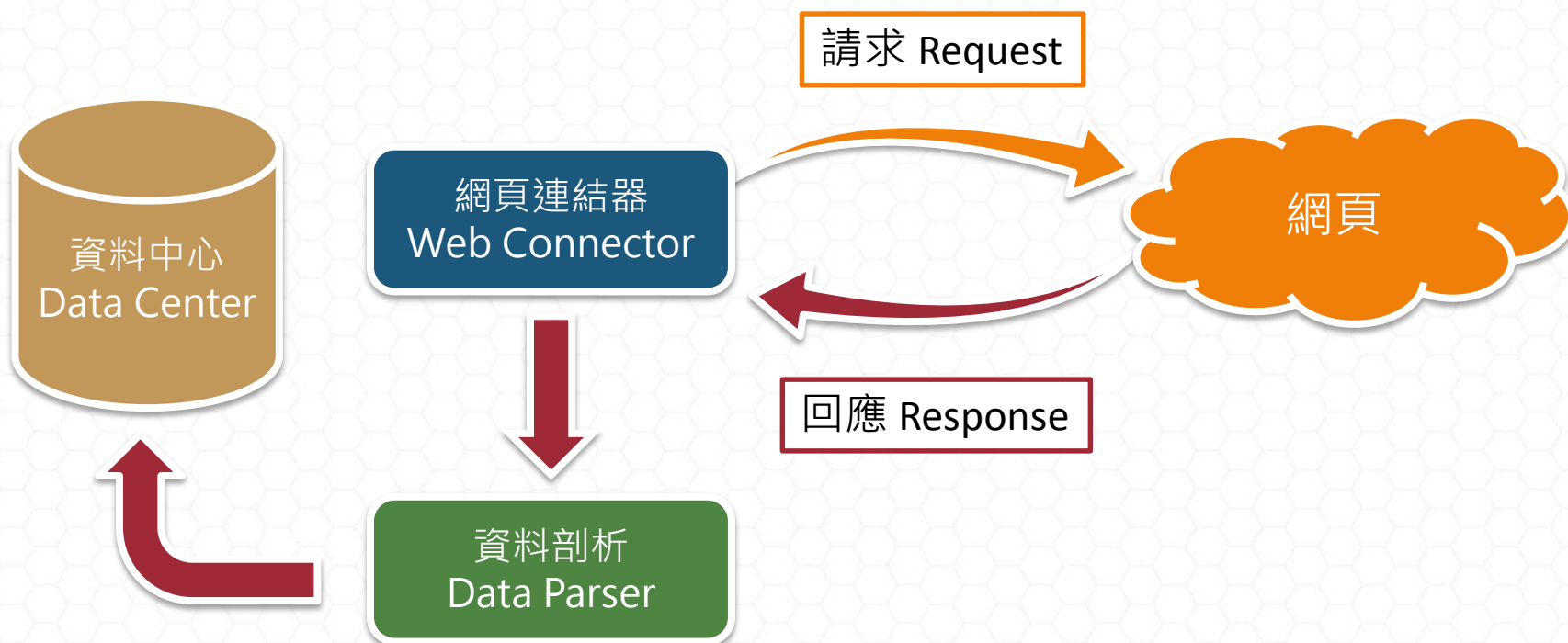
- 1.一般行業申報期限：第一季為5月15日，第二季為8月14日，第三季為11月14日，年度為3月31日。
- 2.金融業申報期限：第一季為5月30日，第二季為8月31日，第三季為11月29日，年度為3月31日。
- 3.銀行及票業申報期限：第一季為5月15日，第二季為8月31日，第三季為11月14日，年度為3月31日。
- 4.保險業申報期限：第一季為5月15日，第二季為8月31日，第三季為11月14日，年度為3月31日。
- 5.證券業申報期限：第一季為5月15日，第二季為8月31日，第三季為11月14日，年度為3月31日。
- 6.申報期限如遇例假日，以證券期貨局公布者為準。



透由簡單的SQL語句
從結構化資料中
達到簡單的分析目的

Prices						
Date	Open	High	Low	Close	Volume	Adj Close*
Mar 25, 2016	158.50	159.00	157.00	158.00	10,175,000	158.00
Mar 24, 2016	158.00	159.00	157.00	158.50	24,853,000	158.50
Mar 23, 2016	158.50	159.50	158.00	159.50	27,478,000	159.50
Mar 22, 2016	159.50	159.50	157.00	158.50	25,809,000	158.50
Mar 21, 2016	160.00	160.00	158.00	160.00	26,100,000	160.00
Mar 18, 2016	158.50	159.50	158.50	159.50	55,975,000	159.50
Mar 17, 2016	159.50	160.00	157.50	158.50	48,193,000	158.50
Mar 16, 2016	155.50	156.00	154.00	156.00	30,962,000	156.00
Mar 15, 2016	155.00	156.50	153.00	154.50	28,689,000	154.50
Mar 14, 2016	156.50	157.50	155.50	156.00	32,751,000	156.00
Mar 11, 2016	154.50	155.00	153.00	155.00	29,566,000	155.00
Mar 10, 2016	153.00	154.50	151.50	154.50	28,302,000	154.50
Mar 9, 2016	152.00	153.00	150.50	153.00	24,004,000	153.00
Mar 8, 2016	151.00	152.00	149.50	152.00	35,683,000	152.00
Mar 7, 2016	152.50	153.50	151.00	152.00	23,906,000	152.00
Mar 4, 2016	153.00	153.50	151.50	152.50	32,794,000	152.50
Mar 3, 2016	154.00	154.50	153.00	154.00	28,822,000	154.00
Mar 2, 2016	154.00	154.50	153.00	153.00	36,010,000	153.00

爬蟲是怎麼運作的



使用開發人員工具

■ 於網頁上點選右鍵 -> 檢查

Navigation bar: 首頁 | 投資組合 | 當日行情 | 大盤 | 類股 | 期權 | 港澳

Search bar: 股票代號/名稱 2330 | 當日個股股價 | 查詢

凱基客戶專區：委託成交 庫存報價

股票代號	時間	成交	買進	賣出	漲跌	張數	昨收
2330台積電 加到投資組合	14:30	158.0	157.5	158.0	▽0.5	10,180	158.5

凱基證券下單 ☐ 買 ☐ 賣 張 零股交易

Context Menu:

- 上一頁(B) Alt+向左鍵
- 下一頁(F) Alt+向右鍵
- 重新載入(R) Ctrl+R
- 另存新檔(A)... Ctrl+S
- 列印(P)... Ctrl+P
- 翻譯成中文(繁體)(T)
- AdBlock
- Evernote Web Clipper
- OneTab
- 檢視網頁原始碼(V) Ctrl+U
- 檢查(N) Ctrl+Shift+I**

點選檢查
或使用ctrl + shift + i

觀察HTTP 請求與返回內容

1. 點選Network

2. 點選Doc

3. 點選連結

Filter

Name

q?s=2330

1 / 120 requests | 18.2 KB / 48.3 KB transferred | Finish: 9.99 s | DOMContentL...

URLs All XHR JS CSS Img Media Font Doc WS Manifest Other

Headers Preview Response Cookies Timing

General

Request URL: <https://tw.stock.yahoo.com/q/q?s=2330>

Request Method: GET

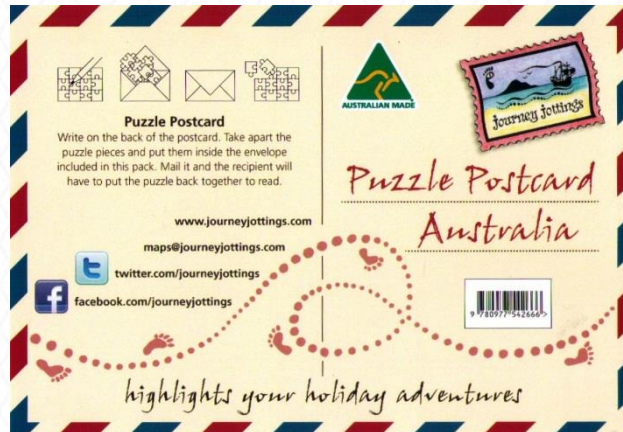
Status Code: 200 OK

Remote Address: 27.123.201.252:443

Response Headers view source

Age: 0

什麼是GET?



GET
內容寫在上頭

<https://tw.stock.yahoo.com/q/q?s=2330>

Python 抓取網頁的主流套件

■ Urllib2

- ▣ 提供獲取URLs(Uniform Resource Locators)的函式及類別

■ Requests

- ▣ 改善Urllib2 的缺點，讓使用者以最簡單的方式獲取網路資源
- ▣ 使用**REST** 操作，可以調用GET,POST, PUT, DELETE

使用GET 抓取頁面資訊

```
import requests
```

```
res = requests.get('https://tw.stock.yahoo.com/q/q?s=2330')
```

```
print res.text
```

```
▼ <tr>
  ▶ <td align="center" width="105">...</td>
    <td align="center" bgcolor="#FFFFFF" nowrap>14:30</td>
  ▶ <td align="center" bgcolor="#FFFFFF" nowrap>...</td>
    <td align="center" bgcolor="#FFFFFF" nowrap>157.5</td>
    <td align="center" bgcolor="#FFFFFF" nowrap>158.0</td>
  ▶ <td align="center" bgcolor="#FFFFFF" nowrap>...</td>
    <td align="center" bgcolor="#FFFFFF" nowrap>10,180</td>
    <td align="center" bgcolor="#FFFFFF" nowrap>158.5</td>
    <td align="center" bgcolor="#FFFFFF" nowrap>158.5</td>
    <td align="center" bgcolor="#FFFFFF" nowrap>159.0</td>
    <td align="center" bgcolor="#FFFFFF" nowrap>157.0</td>
  ▶ <td align="center" width="137" class="tt">...</td>
</tr>
```

使用Help 與 dir 查詢套件與函式

```
import requests
```

```
help(requests)
```

使用help 查詢文件

```
dir(requests)
```

使用dir 表列可用屬性
與方法

```
help(requests.get)
```

```
? requests.get
```

不確定該方法的功能
使用help 或 ?

抓取三大法人買賣超日報


臺灣證券交易所

[ENGLISH](#)
[日本語](#)
[t](#)
[f](#)
[p](#)

[線上支援](#)
[相關服務平台](#)

[關於證交所](#)
[公司治理中心](#)
[交易資訊](#)
[上市公司](#)
[產品與服務](#)
[結算服務](#)
[市場公告](#)
[法令規章](#)

[盤後資訊](#)
[臺灣跨市場指數](#)
[TWSE自行編製指數](#)
[與FTSE合作編製指數](#)
[與銳聯合作編製指數](#)
[與S&PDJI合作編製指數](#)
[升降幅度/首五日無漲跌幅](#)
[變更交易](#)
[當日沖銷交易標的及統計](#)
[融資融券與可融券賣出額度](#)
[標借](#)

三大法人

[三大法人買賣金額統計表](#)
[三大法人買賣超日](#)

[首頁](#) > [交易資訊](#) > [三大法人](#) > [三大法人買賣超日報](#)
[回首頁](#)



資料日期：
分類項目：

☒ 依買賣超股數排列
 ☐ 依證券代號排列

本資訊自民國101年5月2日起提供

證券代號	證券名稱	外資買進股數	外資賣出股數	外資買賣超股數	投信買進股數	投信賣出股數	投信買賣超股數	自營商買進股數	自營商賣出股數 (自行買賣)	自營商買賣超股數 (自行買賣)
1102	亞泥	4,774,896	4,036,737	738,159	0	0	0	185,000	0	1,00
1109	信大	6,000	10,000	-4,000	0	0	0	0	0	
1110	東泥	0	19,000	-19,000	0	0	0	0	0	
1104	環泥	1,000	85,000	-84,000	70,000	6,000	64,000	0	0	
1108	幸福	1,000	46,000	-45,000	0	0	0	23,000	23,000	

<http://www.twse.com.tw/ch/trading/fund/T86/T86.php>

找尋抓取三大法人買賣超日報資訊

■ 填入資訊後按查詢

資料日期： 105/05/06

分類項目： 全部

查詢

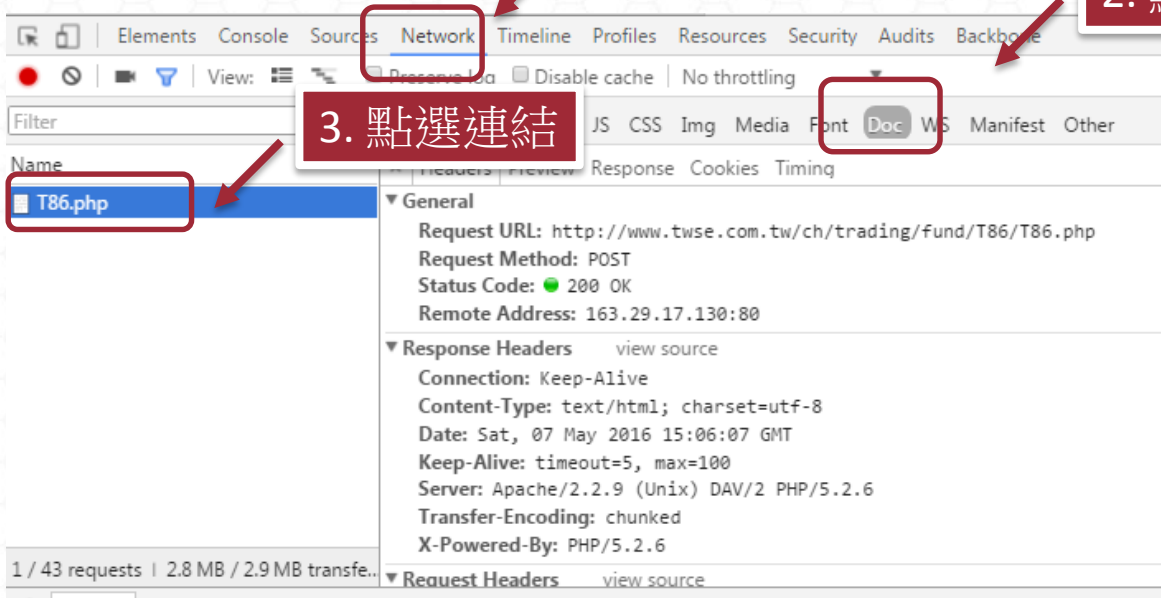
☒ 依買賣超股數排列 ☐ 依證券代號排列

本資訊自民國101年5月2日起提供

1. 點選Network

2. 點選DOC

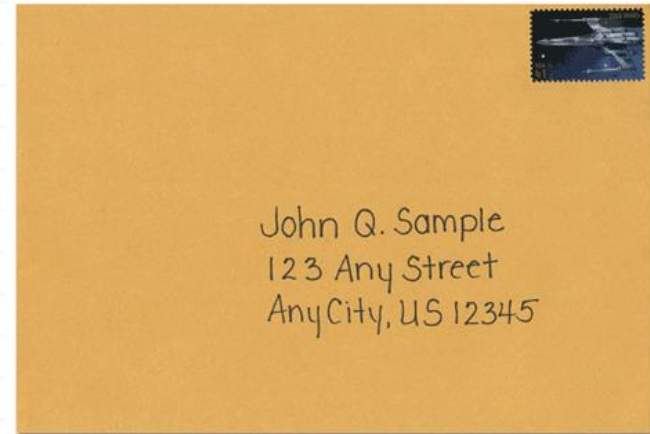
3. 點選連結



什麼是POST?

download:
qdate: 105/05/06
select2: ALL
sorting: by_issue

<http://www.twse.com.tw/ch/trading/fund/T86/T86.php>



POST
內容寫在信紙，包在信封內

使用POST 取得三大法人買賣超日報資訊

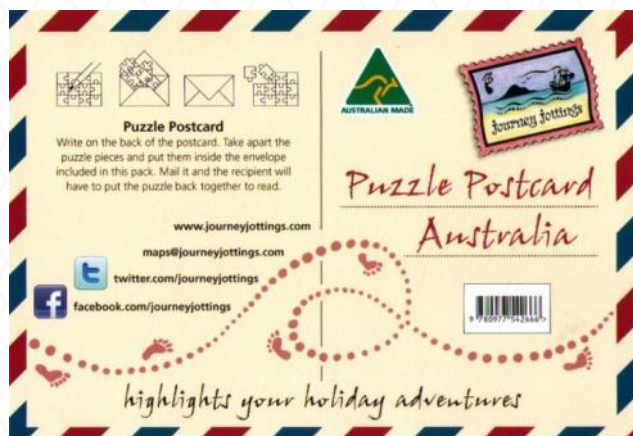
```
import requests
payload = {
    'qdate':'105/05/06',
    'select2':'ALL',
    'sorting':'by_issue'
}

res = requests.post('http://www.twse.com.tw/ch/trading/fund/T86/T86.php',
                    data=payload)
#print res.text
```

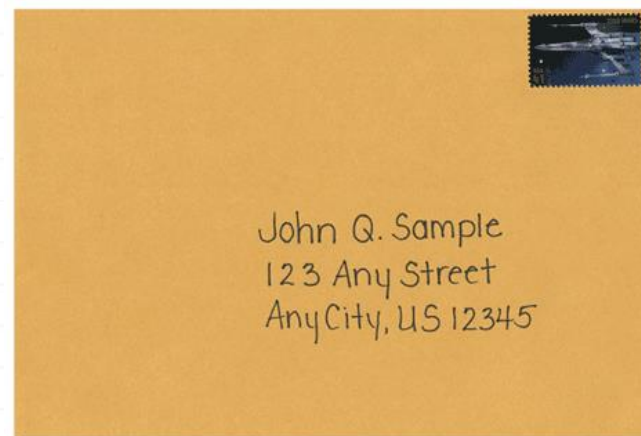
Python 字典(Dictionary)

- 其他語言有相同的操作
 - Java: HashMap, HashTable...
 - C++: hashmap
 - C#: Dictionary...
- `dic = {key : value}`
- 其中key 為唯一不重複值

GET V.S. POST



GET
內容寫在上頭



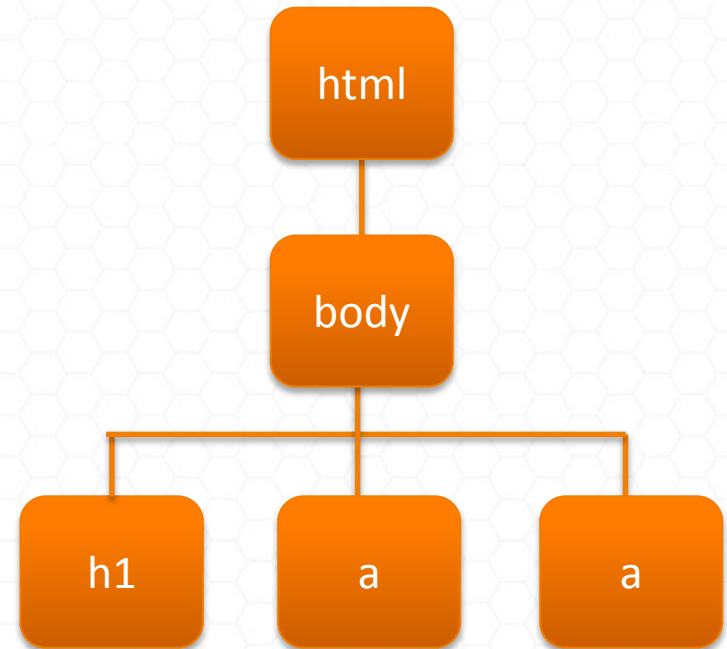
POST
內容寫在信紙，包在信封內

資料剖析

DOM Tree

```
<html>
<body>
<h1 id="title">Hello World</h1>
<a href="#" class="link">This is link1</a>
<a href="# link2" class="link">This is link2</a>
</body>
</html>
```

Document Object Model



使用BeautifulSoup4

- 可以用來剖析及萃取 HTML的內容
- 會自動將讀入的內容轉換成UTF-8編碼
- 底層使用lxml及html5lib，可以使用不同的剖析函式以取得速度與彈性的平衡
 - BeautifulSoup(html_sample, 'html.parser')



可抽換Parser

BeautifulSoup 範例

■ 將網頁讀進BeautifulSoup 中

```
from bs4 import BeautifulSoup
html_sample = '''
<html>
  <body>
    <h1 id="title">Hello World</h1>
    <a href="#" class="link">This is link1</a>
    <a href="# link2" class="link">This is link2</a>
  </body>
</html>'''

soup = BeautifulSoup(html_sample, 'html.parser')
print soup.text
```


找出所有含a tag 的HTML 元素

- 使用Select 找出(第一個)含有a tag 的元素

```
soup = BeautifulSoup(html_sample, 'html.parser')  
alink = soup.select('a')  
print alink
```

Select 的結果會存放在list 中

取得含有特定ID的元素

- 使用Select 找出所有id為title的元素

```
alink = soup.select('#title')  
print alink
```

ID 前面必須加上 #

取得含有特定class的元素

- 使用Select 找出所有class為link的元素

```
soup = BeautifulSoup(html_sample, 'html.parser')  
for link in soup.select('.link'):  
    print link
```

Class 前面必須加上 .

取得所有a tag 內的連結

使用select找出所有a tag 的href 連結

```
alinks = soup.select('a')
```

```
for link in alinks:
```

```
    print link['href']
```


試著抓取Yahoo 股市資訊

2. 點選要抓取的區塊

1. 點選觀察元素

3. 檢視css path

凱基客戶專區：委託成交 庫存報價 資料日期: 105/03/27

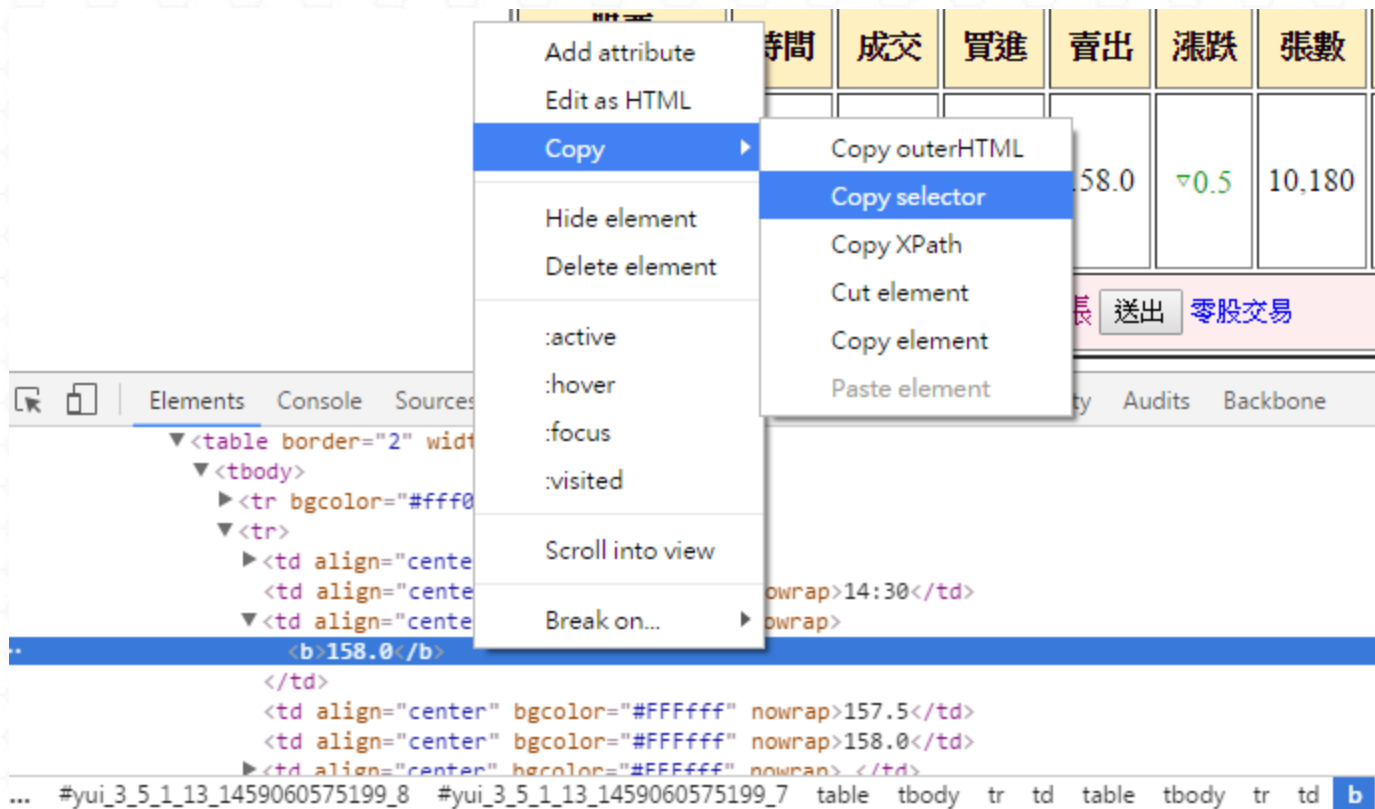
股票代號	時間	成交	買進	賣出	漲跌	張數	昨收	開盤	最高	最低	個股資料
00220 台積電 投資組合	14:30	158.0	157.5	158.0	↑0.5	10,180	158.5	158.5	159.0	157.0	成交明細 技術 新聞 基本 籌碼 個股健診

凱基證券下單 ☐ 買 ☐ 賣 張 送出 零股交易

```
<table border="2" width="750">
  <tbody>
    <tr bgcolor="#fff0c1">...</tr>
    <tr>
      <td align="center" width="105">...</td>
      <td align="center" bgcolor="#FFFFFF" nowrap>14:30</td>
      <td align="center" bgcolor="#FFFFFF" nowrap>
        <b>158.0</b>
      </td>
      <td align="center" bgcolor="#FFFFFF" nowrap>157.5</td>
      <td align="center" bgcolor="#FFFFFF" nowrap>158.0</td>
      <td align="center" bgcolor="#FFFFFF" nowrap>...</td>
    </tr>
  </tbody>
</table>
```

```
font-weight: bold;
}
Inherited from table
table {
  display: table;
}
```

複製css selector



```
#yui_3_5_1_13_1459060575199_7 > table:nth-child(13) > tbody > tr > td > table > tbody  
> tr:nth-child(2) > td:nth-child(3) > b
```

抓取成交價格

```
import requests
from bs4 import BeautifulSoup as bs
res =
requests.get('https://tw.stock.yahoo.com/q/q?s=23
30')
soup = bs(res.text, 'html.parser')
print soup.select('b')
print soup.select('b')[0].text
```

尋找CSS 的定位

- Chrome 開發人員工具

- Firefox 開發人員工具

- InfoLite

- ▣ <https://chrome.google.com/webstore/detail/infolite/ipjbadabbpedegielkhgpiekdImfpgal>

使用InfoLite 點選抓取區域

YAHOO! 股市 奇摩

Q 搜尋

首頁 投資組合 當日行情 大盤 類股 期權 港

股票代號/名稱 2330 當日個股股價 查詢

InfoLite Username Password Login X ?

Submit

b Clear (1) Add

凱基客戶專區：委託成交 庫存報價 資料日期: 105/03/27

股票代號	時間	成交	買進	賣出	漲跌	張數	昨收	開盤	最高	最低	個股資料
2330台積電 加到投資組合	14:30	158.0 td b	157.5	158.0	↑0.5	10,180	158.5	158.5	159.0	157.0	成交明細 技術 新聞 基本 籌碼

凱基證券下單 買 賣 張 送出 零股交易

綠色: 目前選取得區塊
黃色: 符合樣式的區塊
紅色: 排除的曲塊
Clear旁邊的數字: 符合區塊的數目

HTML 表格

```
<table>
<thead>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
</thead>
<tbody>
  <tr> <td>January</td> <td>$100</td></tr>
  <tr> <td>February</td> <td>$80</td></tr>
</tbody>
<tfoot>
  <tr> <td>Sum</td> <td>$180</td> </tr>
</tfoot>
</table>
```

Month	Savings
January	\$100
February	\$80
Sum	\$180

如何有效抓取HTML表格資料？

使用Pandas處理表格資料

Pandas

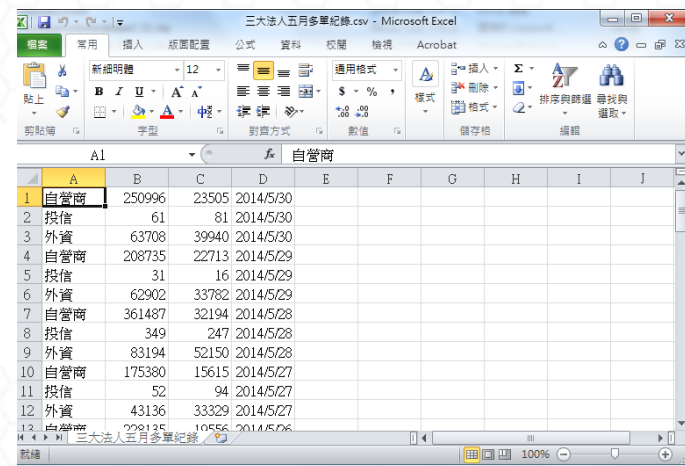
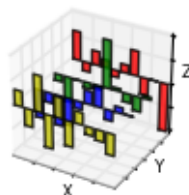
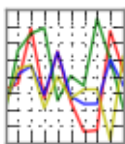
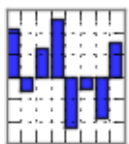
■ Python for Data Analysis

- 源自於R

- Table –Like 格式

- 提供高效能、簡易使用的資料格式(Data Frame)讓使用者可以快速操作及分析資料

pandas

$$y_{it} = \beta' x_{it} + \mu_i + \epsilon_{it}$$
A screenshot of a Microsoft Excel spreadsheet titled "三大陸人五月多單紀錄.csv". The spreadsheet contains a table with columns A through J. The data is organized into rows, with the first column (A) containing labels like "自營商", "投信", and "外資". The subsequent columns (B, C, D) contain numerical values and dates. The table is displayed in a standard Excel format with a ribbon at the top and a status bar at the bottom.

	A	B	C	D	E	F	G	H	I	J
1	自營商	250996	23505	2014/5/30						
2	投信	61	81	2014/5/30						
3	外資	63708	39940	2014/5/30						
4	自營商	208735	22713	2014/5/29						
5	投信	31	16	2014/5/29						
6	外資	62902	33782	2014/5/29						
7	自營商	361487	32194	2014/5/28						
8	投信	349	247	2014/5/28						
9	外資	83194	52150	2014/5/28						
10	自營商	175380	15615	2014/5/27						
11	投信	52	94	2014/5/27						
12	外資	43136	33329	2014/5/27						
13	自營商	228125	10556	2014/5/26						

使用read_html 讀取表格

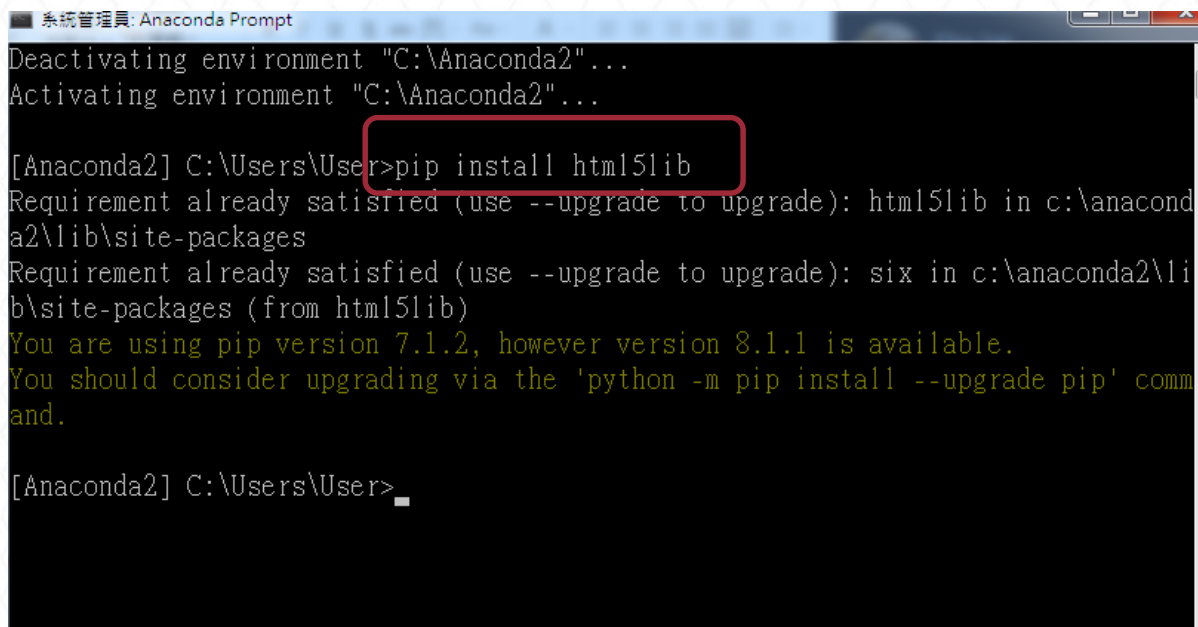
```
table = """
<table>
  <thead>
    <tr>
      <th>Month</th>
      <th>Savings</th>
    </tr>
  </thead>
  <tbody>
    <tr> <td>January</td> <td>$100</td></tr>
    <tr> <td>February</td> <td>$80</td></tr>
  </tbody>
  <tfoot>
    <tr> <td>Sum</td> <td>$180</td> </tr>
  </tfoot>
</table>
"""
```

```
import pandas as pd
dfs = pd.read_html(table)
dfs[0]
```

	Month	Savings
0	January	\$100
1	February	\$80
2	Sum	\$180

安裝html5lib

- 在Anaconda Prompt 下打
 - `pip install html5lib`



```
系統管理員: Anaconda Prompt
Deactivating environment "C:\Anaconda2"...
Activating environment "C:\Anaconda2"...

[Anaconda2] C:\Users\User>pip install html5lib
Requirement already satisfied (use --upgrade to upgrade): html5lib in c:\anaconda2\lib\site-packages
Requirement already satisfied (use --upgrade to upgrade): six in c:\anaconda2\lib\site-packages (from html5lib)
You are using pip version 7.1.2, however version 8.1.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

[Anaconda2] C:\Users\User>
```

- 必須將Jupyter Notebook 重啟

使用read_html 讀取Yahoo 股市表格

```
import pandas as pd
table = soup.select('table + table table')[0]
dfs = pd.read_html(table.prettify('utf-8'), encoding=
'utf-8', header=0)
dfs[0]
```

	股票 代號	時間	成交	買進	賣出	漲跌	張數	昨收	開盤	最高	最低	個股資料
0	2330台積電 加到投資組合	14:30	158	157.5	158	▽0.5	10180	158.5	158.5	159	157	成交明細 技術 新聞 基本 籌碼 個股健診
1	凱基證券下單	買 賣 張 零股交易	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

```
print dfs[0]['成交'].decode('utf-8')[0]
```

Python DataFrame 範例

```
df = pd.DataFrame([[ 'frank', 'M', 29], [ 'mary', 'F', 23], [ 'tom',  
'M', 35], [ 'ted', 'M', 33], [ 'jean', 'F', 21], [ 'lisa', 'F', 20]])
```

```
df.columns = [ 'name', 'gender', 'age' ]
```

```
df
```

	name	gender	age
0	frank	M	29
1	mary	F	23
2	tom	M	35
3	ted	M	33
4	jean	F	21
5	lisa	F	20

6 rows × 3 columns

進行簡單的統計分析

`df.describe()`

	age
count	6.000000
mean	26.833333
std	6.400521
min	20.000000
25%	21.500000
50%	26.000000
75%	32.000000
max	35.000000

8 rows × 1 columns

存取元素與切割 (Indexing & Slicing)

➤ `df.ix[1]`

name mary

gender F

age 23

Name: 1, dtype: object

➤ `df.ix[1:4]`

	name	gender	age
1	mary	F	23
2	tom	M	35
3	ted	M	33
4	jean	F	21

4 rows × 3 columns

存取元素與切割 (Indexing & Slicing)

➤ `df['name']`

```
0    frank
1    mary
2    tom
3    ted
4    jean
5    lisa
Name: name, dtype: object
```

➤ `df[['name', 'age']]`

	name	age
0	frank	29
1	mary	23
2	tom	35
3	ted	33
4	jean	21
5	lisa	20

6 rows × 2 columns

存取元素與切割 (Indexing & Slicing)

➤ `df['gender'] == 'M'`

```
0    True
1   False
2    True
3    True
4   False
5   False
```

Name: gender, dtype: bool

➤ `df[df['gender'] == 'M']`

	name	gender	age
0	frank	M	29
2	tom	M	35
3	ted	M	33

3 rows × 3 columns

取男女年齡平均

➤ `df[df['gender'] == 'M'].mean()`

age 32.333333

dtype: float64

➤ `df[df['gender'] == 'F'].mean()`

age 21.333333

dtype: float64

如果今天性別有很多個？
是否要根據不同性別不斷取平均？

使用SQL統計資料

```
SELECT gender, AVERAGE(age) FROM df  
GROUP BY gender
```

=

```
df.groupby('gender')['age'].mean()
```

```
gender
```

```
F      21.333333
```

```
M      32.333333
```

```
Name: age, dtype: float64
```

整理三大法人買賣超日報資訊

整理三大法人買賣超日報資訊


臺灣證券交易所

[ENGLISH](#)
[日本語](#)
[t](#)
[f](#)
[p](#)

[線上支援](#)
[相關服務平台](#)

[關於證交所](#)
[公司治理中心](#)
[交易資訊](#)
[上市公司](#)
[產品與服務](#)
[結算服務](#)
[市場公告](#)
[法令規章](#)

- 盤後資訊
- 臺灣跨市場指數
- TWSE自行編製指數
- 與FTSE合作編製指數
- 與銳聯合作編製指數
- 與S&PDJI合作編製指數
- 升降幅度/首五日無漲跌幅
- 變更交易
- 當日沖銷交易標的及統計
- 融資融券與可融券賣出額度
- 標借
- 三大法人
 - 三大法人買賣金額統計表
 - 三大法人買賣超口

[首頁](#) > [交易資訊](#) > [三大法人](#) > [三大法人買賣超日報](#)
[回首頁](#)



資料日期：
 分類項目：

☒ 依買賣超股數排列
 ☐ 依證券代號排列

本資訊自民國101年5月2日起提供

105年05月06日 三大法人買賣超日報										
證券代號	證券名稱	外資買進股數	外資賣出股數	外資買賣超股數	投信買進股數	投信賣出股數	投信買賣超股數	自營商買賣超股數	自營商買進股數(自行買賣)	自營商賣出股數(自行買賣)
1102	亞泥	4,774,896	4,036,737	738,159	0	0	0	185,000	0	1,00
1109	信大	6,000	10,000	-4,000	0	0	0	0	0	
1110	東泥	0	19,000	-19,000	0	0	0	0	0	
1104	環泥	1,000	85,000	-84,000	70,000	6,000	64,000	0	0	
1108	幸福	1,000	46,000	-45,000	0	0	0	23,000	23,000	

抓取三大法人買賣超日報資訊

```
import requests
payload = {
    'qdate':'105/05/06',
    'select2':'ALL',
    'sorting':'by_issue'
}
```

```
res = requests.post('http://www.twse.com.tw/ch/trading/fund/T86/T86.php', data=payload)
#print res.text
```

使用Pandas 讀取資料

```
from bs4 import BeautifulSoup as bs
import pandas as pd
soup = bs(res.text, 'html5lib')
tbl = soup.select('#tbl-sortable')[0]
dfs = pd.read_html(tbl.prettify('utf-8'), encoding='utf-8')
stockdf = dfs[0]
```

Out[30]:

	證券代號	證券名稱	外資買進股數	外資賣出股數	外資買賣超股數	投信買進股數	投信賣出股數	投信買賣超股數	自營商買超股數	自營商買進股數(自行買賣)	自營商賣出股數(自行買賣)	自營商買賣股數(自行買賣)
0	00632R	T50反1	350000	391000	-41000	0	5243000	-5243000	65286000	1195000	1030000	165
1	042800	永豐EX	0	45000	-45000	0	0	0	9533000	0	0	0
2	2349	鍊德	9648000	997000	8651000	0	0	0	0	0	0	0
3	2345	智邦	2484000	804000	1680000	1854000	0	1854000	509000	472000	66000	406

猜猜哪隻股票外資買賣超最多？

```
stockdf[stockdf['外資 買賣超股數'.decode('utf-8')]==stockdf['外資 買賣超股數'.decode('utf-8')].max()]
```

	證券代號	證券名稱	外資買進股數	外資賣出股數	外資買賣超股數	投信買進股數	投信賣出股數	投信買賣超股數	自營商買賣超股數	自營商買進股數(自行買賣)	自營商賣出股數(自行買賣)	自營商買賣超股數(自行買賣)	自營商買進股數(避險)	自營商賣出股數(避險)	自營商買賣超股數(避險)	三大法人買賣超股數
2	2349	鍊德	9648000	997000	8651000	0	0	0	0	0	0	0	0	0	0	8651000

根據買賣超排序

```
stockdf.sort_values( by= '外資 買賣超股數'  
' .decode('utf-8'), ascending=False).head()
```

	證券代號	證券名稱	外資買進股數	外資賣出股數	外資買賣超股數	投信買進股數	投信賣出股數	投信買賣超股數	自營商買賣超股數	自營商買進股數(自行買賣)	自營商賣出股數(自行買賣)	自營商買賣超股數(自行買賣)	自營商買進股數(避險)	自營商賣出股數(避險)
2	2349	鍊德	9648000	997000	8651000	0	0	0	0	0	0	0	0	0
6	3023	信邦	4756000	1194000	3562000	214000	0	214000	-164000	118000	348000	-230000	340000	274
18	1312	國喬	3830000	713000	3117000	0	1326000	-1326000	198000	75000	0	75000	249000	126
5	1605	華新	3451000	1257000	2194000	1500000	0	1500000	-61000	1000	100000	-99000	210000	172

由大到小做排序

定義函式

```
def getTradingVolume(date):  
    payload['qdate'] = date  
    res = requests.post('http://www.twse.com.tw/ch/trading/fund/T86/T86.php', data=payload)  
    soup = bs(res.text, 'html5lib')  
    tbl = soup.select('#tbl-sortable')[0]  
    dfs = pd.read_html(tbl.prettify('utf-8'), encoding='utf-8')  
    stockdf = dfs[0]  
    stockdf['ymd'] = date  
    return stockdf
```

增加日期欄位

時間跟字串轉換

```
from datetime import datetime
currenttime = datetime.now()
print currenttime.strftime("%Y-%m-%d")
```

```
a = '2014-05-03 14:00'
print datetime.strptime(a, "%Y-%m-%d %H:%M")
```

產生日期

```
from datetime import date, datetime, timedelta
currenttime = datetime.now()
for i in range(1,3):
    dt = currenttime - timedelta(days = i)
    print dt
    print dt.strftime('%Y/%m/%d')
```

但是必須要民國時間

產生民國日期

```
from datetime import date,datetime, timedelta
currenttime = datetime.now()
for i in range(1,3):
    dt = currenttime - timedelta(days = i)
    year = int(dt.strftime('%Y')) - 1911
    monthdate = dt.strftime('%m/%d')
    print '{}/{}'.format(year, monthdate)
```


增加日期轉換函式

```
def getTWDate(dt):  
    year = int(dt.strftime('%Y')) - 1911  
    monthdate = dt.strftime('%m/%d')  
    ymd = '{}/{}'.format(year, monthdate)  
    return ymd
```

修改原本函式

```
def getTradingVolume(dt):  
    payload['qdate'] = getTWDate(dt)  
    res = requests.post('http://www.twse.com.tw/ch/trading/fund/T86/T86.php',  
data=payload)  
    soup = bs(res.text, 'html5lib')  
    tbl = soup.select('#tbl-sortable')[0]  
    dfs = pd.read_html(tbl.prettify('utf-8'), encoding='utf-8')  
    stockdf = dfs[0]  
    stockdf['ymd'] = dt  
    return stockdf
```

批次執行30天的資料

```
dfs = []  
currenttime = datetime.now()  
for i in range(1,30):  
    dt = currenttime.date() - timedelta(days = i)  
    print dt,  
    dfs.append(getTradingVolume(dt))
```

合併所有的Data Frame

```
stockdf = pd.concat(dftotal, ignore_index=True)  
len(stockdf)
```

	證券代號	證券名稱	外資買進股數	外資賣出股數	外資買賣超股數	投信買進股數	投信賣出股數	投信買賣超股數	自營商買賣超股數	自營商買進股數(自行買賣)	自營商賣出股數(自行買賣)	自營商買賣超股數(自行買賣)	自營商買進股數(避險)	自營商賣出股數(避險)	自買股數
0	3474	華亞科	5865836	3726000	2139836	0	0	0	-4000	0	0	0	0	4000	-4
1	2408	南亞科	2755000	1194000	1561000	0	0	0	104000	1000	0	1000	106000	3000	10
2	2449	京元	6490000	5054000	1436000	144000	443000	-299000	131000	42000	77000	-35000	509000	343000	16

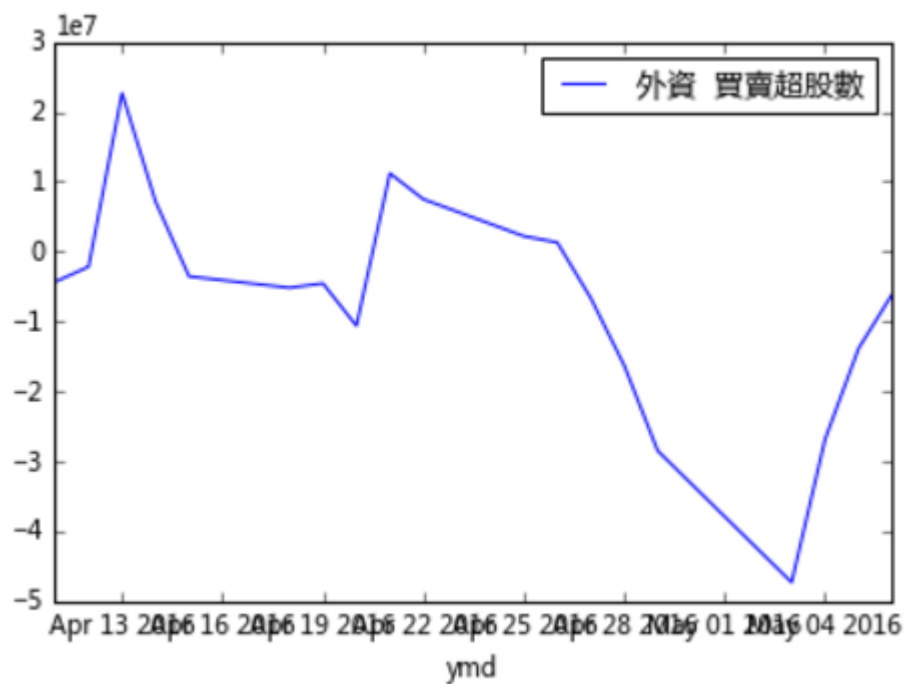
篩選出台積電股票

```
stockdf[stockdf['證券代號'.decode('utf-8')] == 2330]
```

	證券代號	證券名稱	外資買進股數	外資賣出股數	外資買賣超股數	投信買進股數	投信賣出股數	投信買賣超股數	自營商買賣超股數	自營商買進股數(自行買賣)	自營商賣出股數(自行買賣)	自營商買賣超股數(自行買賣)	自營商買賣超股數(自營商買賣)
66	2330	台積電	30192917	36329176	-6136259	11000	19000	-8000	123000	270000	202000	68000	337000
137	2330	台積電	23891785	37758202	-13866417	127000	53000	74000	1451000	1437000	98000	1339000	307000
209	2330	台積電	23512540	50354331	-26841791	10000	165000	-155000	1010000	1045000	37000	1008000	404000

繪製折線圖

```
df2330 = stockdf[stockdf['證券 代號'.decode('utf-8')] == 2330]  
df2330.plot.line(x = 'ymd', y= '外資 買賣超股數'.decode('utf-8'))
```



如何讓matplotlib 出現中文？

■ 切換到matplotlib 設定目錄下

- C:\Anaconda2\Lib\site-packages\matplotlib\mpl-data

■ 修改matplotlibrc (將註解#拿掉)

- font.family : sans-serif

- font.sans-serif : Microsoft YaHei, ...

增加微軟雅黑字體

資料儲存 (SQLITE)

課前預備

■ 安裝SQLite

- 請至官網<http://sqlite.org/download.html> 下載適合自己作業系統的版本安裝

■ 安裝SQLite Manager

- 打開Firefox 後至下列網址
<https://addons.mozilla.org/en-US/firefox/addon/sqlite-manager/> 下載適合自己作業系統的版本安裝

課前知識

■ 特性

- self-contained
- serverless
- zero-configuration
- Transactional

■ ACID 資料庫

■ 支援 SQL 92 語法

■ 開源

使用python 連結SQLite

```
import sqlite3 as lite
import sys
con = None
try:
    con = lite.connect(dbname) # connect to database
    cur = con.cursor() # create cursor
    cur.execute('SELECT SQLITE_VERSION()') # selece database version
    data = cur.fetchone() # fetch one data at a time
    print "SQLite version: %s" % data
except lite.Error, e:
    print "Error %s:" % e.args[0] sys.exit(1)
finally:
    if con:
        con.close()
```

使用python 連結SQLite (2)

with con:

```
cur = con.cursor()
```

```
# Read Meta Information
```

```
cur.execute("PRAGMA table_info(PhoneAddress)")
```

```
rows = cur.fetchall()
```

```
for row in rows:
```

```
    print row
```


透過SQLite 做資料新增、查詢

```
import sqlite3 as lite
import sys
con = lite.connect("test.db")
with con:
    cur = con.cursor() # Drop Table If Exisits
    cur.execute("DROP TABLE IF EXISTS PhoneAddress")
    cur.execute("CREATE TABLE PhoneAddress(phone CHAR(10) PRIMARY KEY, address TEXT, name TEXT
unique, age INT NOT NULL)")
    cur.execute("INSERT INTO PhoneAddress VALUES('0912173381','United State','Jhon Doe',53)")
    cur.execute("INSERT INTO PhoneAddress VALUES('0928375018','Tokyo Japan','MuMu Cat',6)")
    cur.execute("INSERT INTO PhoneAddress VALUES('0957209108','Taipei','Richard',29)")
    cur.execute("SELECT phone,address FROM PhoneAddress")
    data = cur.fetchall()
    for rec in data:
        print rec[0], rec[1]
con.close()
```

fetchone v.s. fetchall

```
rows = cur.fetchall()
for row in rows:
    print row
```

```
data = cur.fetchone()
print data[0], data[1]
```

使用Pandas 將資料塞進資料庫

```
import sqlite3 as lite
with lite.connect('finance.sqlite') as db:
    stockdf.to_sql(name='trading_volume',
index=False, con=db, if_exists='replace')
```

開啟SQLite Manager



使用SQLite Manager瀏覽資料

檢視所有塞入的資料

SQLite Manager - C:\Users\User\pyfinance\finance.sqlite

Database (D) Table (T) Index (I) View (V) Trigger (T) Tools (O) Help

Directory (Select Profile Database) Go

finance.sqlite

Master Table (1)
Tables (1)
enterprise_eps
Views (0)
Indexes (0)
Triggers (0)

Structure Browse & Search Execute SQL DB Settings

TABLE enterprise_eps Search (H) Show All Add (A) Duplicate (D) Edit (E) Delete (L)

rowid	公司代號	公司名稱	產業別	基本每股盈...	普通股每股...	營業收入	營業利益	營業外收入...	稅後淨利	民
1	1102	亞洲水泥股...	水泥工業	0.4	新台幣 10.00...	13931550	339801.0	1250044.0	1371559	
2	1101	台灣水泥股...	水泥工業	0.38	新台幣 10.00...	24114047	2026729.0	314060.0	1999624	
3	1104	環球水泥股...	水泥工業	0.3	新台幣 10.00...	1248072	30247.0	156012.0	183441	
4	1108	華僑水泥股...	水泥工業	0.17	新台幣 10.00...	1203671	98223.0	-13612.0	63869	
5	1103	嘉新水泥股...	水泥工業	0.13	新台幣 10.00...	741189	-149811.0	183613.0	59637	
6	1110	東南水泥股...	水泥工業	0.11	新台幣 10.00...	460227	51795.0	7706.0	61097	
7	1109	信大水泥股...	水泥工業	0.09	新台幣 10.00...	916242	35919.0	7715.0	35090	
8	1256	鮮活控股股...	食品工業	2.59	新台幣 10.00...	341578	57968.0	7039.0	45783	
9	1232	大統益股份...	食品工業	1.32	新台幣 10.00...	4882485	210352.0	49243.0	212608	
10	1227	佳格食品股...	食品工業	1.23	新台幣 10.00...	5076330	749193.0	93050.0	699669	
11	1231	聯華食品工...	食品工業	0.91	新台幣 10.00...	1482501	117267.0	16905.0	111496	
12	1233	天仁茶業股...	食品工業	0.76	新台幣 10.00...	532755	78968.0	4005.0	68672	
13	1216	統一企業股...	食品工業	0.75	新台幣 10.00...	104634790	5759096.0	1223048.0	5739800	
14	1702	南僑化學工...	食品工業	0.6	新台幣 10.00...	2917694	234161.0	-12219.0	148329	
15	1236	宏亞食品股...	食品工業	0.58	新台幣 10.00...	726305	74045.0	624.0	62845	
16	1201	味全食品工...	食品工業	0.58	新台幣 10.00...	6700847	376786.0	-25129.0	247114	
17	1210	大成興城企...	食品工業	0.45	新台幣 10.00...	21357129	250873.0	87210.0	291215	
18	1203	味王股份有...	食品工業	0.35	新台幣 10.00...	1643325	132636.0	8902.0	120200	
19	1215	台灣卜蜂企...	食品工業	0.31	新台幣 10.00...	4380233	97982.0	-8828.0	68825	
20	1220	聯華食品股...	食品工業	0.3	新台幣 10.00...	1482501	117267.0	16905.0	111496	

1 to 100 of 9825

使用Pandas 下SQL 查詢資料

```
import sqlite3 as lite
with lite.connect('finance.sqlite') as db:
    df = pd.read_sql_query('SELECT count(1)
FROM trading_volume;', db)
df
```

AJAX 資料抓取

抓取財報狗的資訊



<https://statementdog.com/analysis/tpe#2330>

找出呼叫資料的進入點

1. 點選Network

2. 點選XHR

3. 點選連結

event?a=4677131045&d=45091211
?data=evJldmVudCt6ICJtcF9uWdl
1?_=1459090955106
?_=1459090955110

半導體 | 上市公司
董事長 張忠謀
執行長 劉德音及魏哲家
依客戶之訂單與其提供之產品設計說明，以從事製造與銷售積體電路以及其他晶圓半導體裝置，提供... 查看更多

4 / 25 requests | 24.9 KB / 101 KB transferred | Finish: ...

Console Search

抓取圖表資料

```
import requests
headers = {
    'X-Requested-With': 'XMLHttpRequest',
    'User-Agent': 'Mozilla/5.0 (Windows NT 6.1; WOW64)
    AppleWebKit/537.36 (KHTML, like Gecko) Chrome/49.0.2623.87
    Safari/537.36',
}
rs = requests.session()
res =
rs.get('https://statementdog.com/analysis/analysis_ajax/2330/2011/1/20
16/4/1', headers = headers)
json = res.text
```

使用Pandas讀取JSON

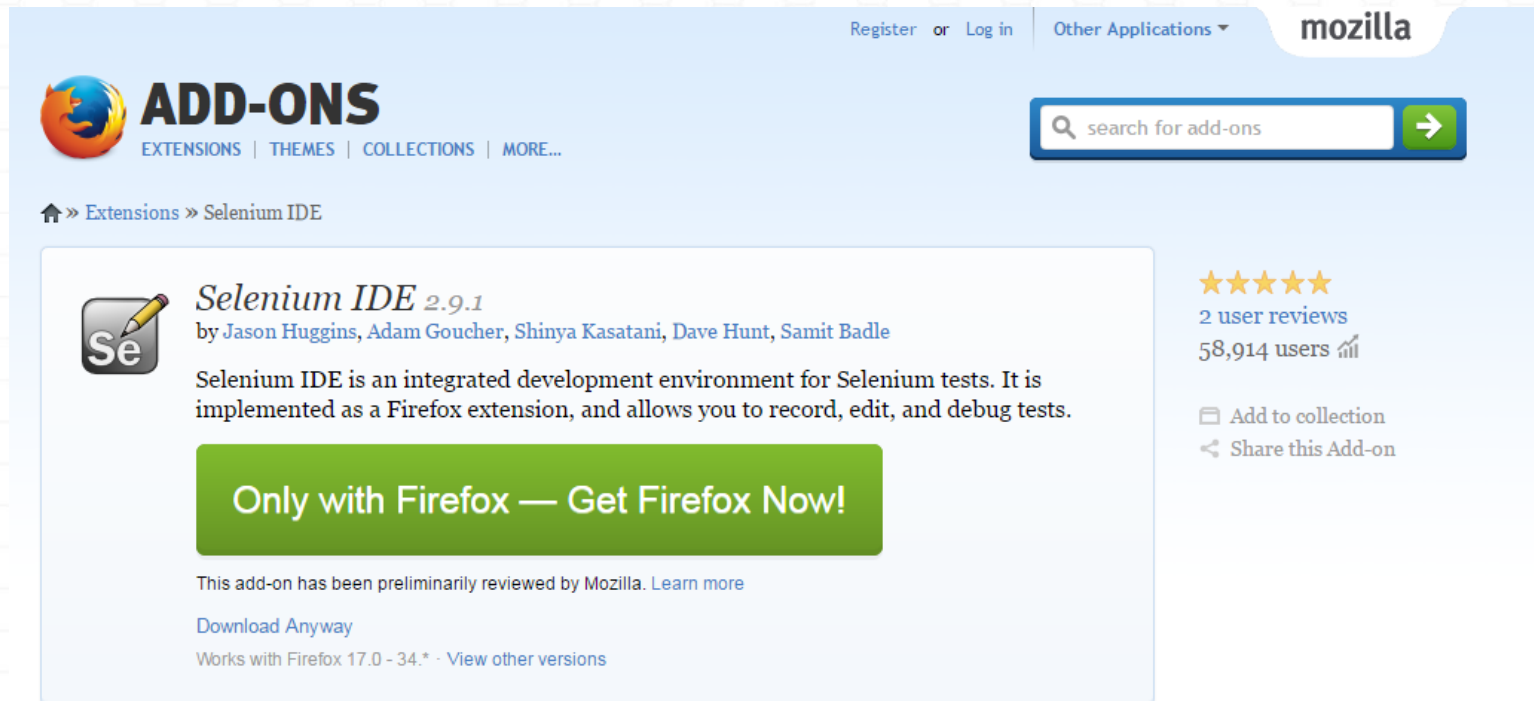
```
jdf = pd.read_json(json)
```

```
jdf
```

	0.0	1.0	10.0	100.0	101.0	102.0	103.0	104.0	105.0	106.0	...	90.0	91.0	92.0	93.0	94.0	95.0	96.0	97.0
data	[0, success]	[2330 台積電, 半導體, 2016-03-25 158.0, 上市, 依客戶之訂單與其...	[]	[[0, 24.01], [1, 21.97], [2, 20.58], [3, 18.09...]]	[[0, 28.81], [1, 30.12], [2, 25.18], [3, 22.21...]]	[[0, 23.87], [1, 29.43], [2, 28.18], [3, 34.31...]]	[]	[]	[]	[]	...	[[0, 155.04], [1, 175.04], [2, 181.41], [3, 23...]]	[[0, 2.35], [1, 2.38], [2, 2.32], [3, 2.42], [...]]	[[0, 1.78], [1, 1.88], [2, 2.16], [3, 2.29], [...]]	[[0, 0.26], [1, 0.24], [2, 0.22], [3, 0.22], [...]]	[[0, 0.14], [1, 0.14], [2, 0.14], [3, 0.14], [...]]	[[0, 49.03], [1, 46.02], [2, 42.04], [3, 44.74...]]	[[0, 37.16], [1, 34.27], [2, 29.67], [3, 31.45...]]	[[0, 34.5], [1, 32.6], [2, 28.58], [3, 30.14]]
label	Return	StockInfo	最新彼得林區評價	近四季ROA	近四季ROE	盈再率	15%股利折現	10%股利折現	5%股利折現	0%股利折現	...	營業現金對稅後淨利比	應收帳款周轉	存貨週轉	固定資產週轉	總資產週轉	毛利率	營業利率	稅後利率

使用Selenium 抓資料

使用Selenium Plugin




The screenshot shows the Mozilla Add-ons website for Selenium IDE. The page has a light blue header with the Mozilla logo and navigation links. The main content area features the Selenium IDE add-on card, which includes its icon, name, version, authors, description, and a prominent green button to get Firefox. To the right of the card, there are star ratings, user reviews, and user count. Below the card, there are links to download the add-on and view other versions.

Register or Log in Other Applications ▼ mozilla

ADD-ONS
EXTENSIONS | THEMES | COLLECTIONS | MORE...

🔍 search for add-ons →

🏠 » Extensions » Selenium IDE

 **Selenium IDE 2.9.1**
by Jason Huggins, Adam Goucher, Shinya Kasatani, Dave Hunt, Samit Badle

Selenium IDE is an integrated development environment for Selenium tests. It is implemented as a Firefox extension, and allows you to record, edit, and debug tests.

Only with Firefox — Get Firefox Now!

This add-on has been preliminarily reviewed by Mozilla. [Learn more](#)

[Download Anyway](#)

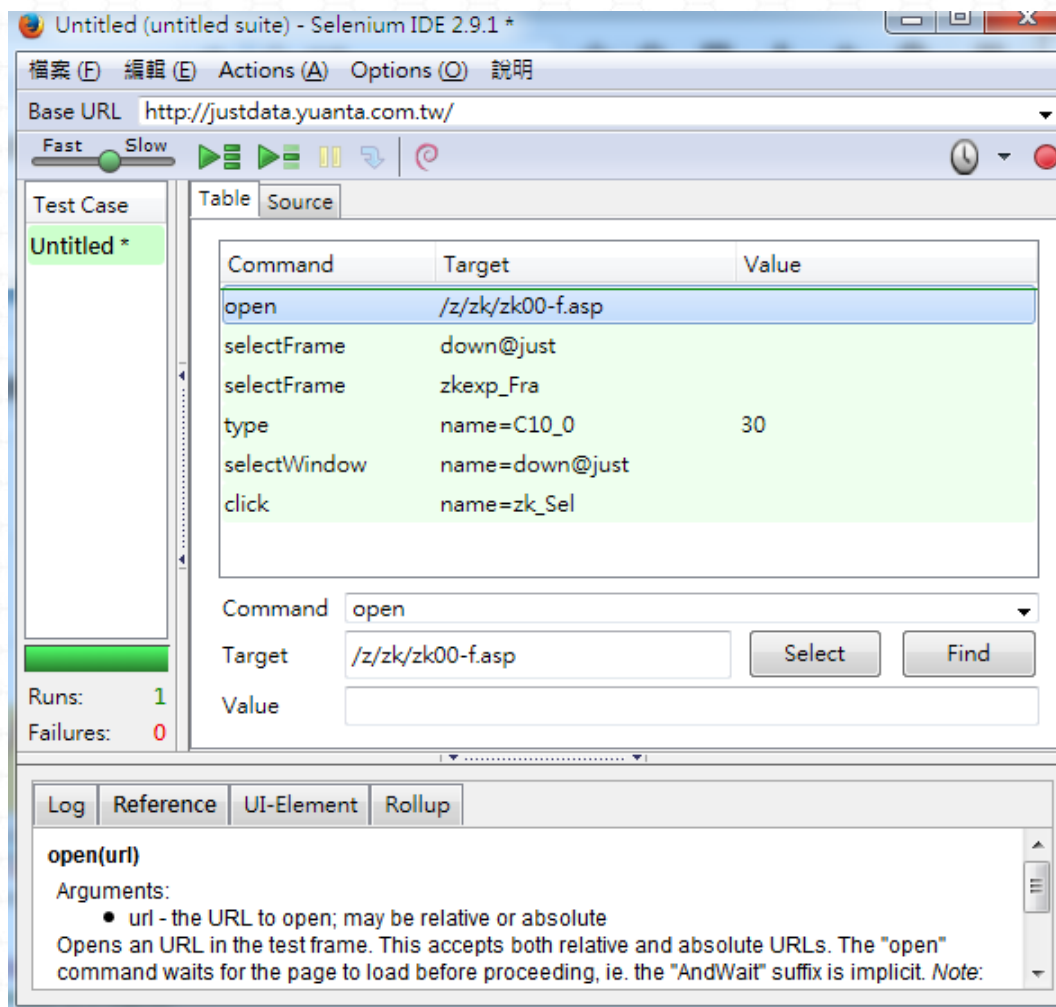
Works with Firefox 17.0 - 34.* · [View other versions](#)

★★★★★
2 user reviews
58,914 users 📈

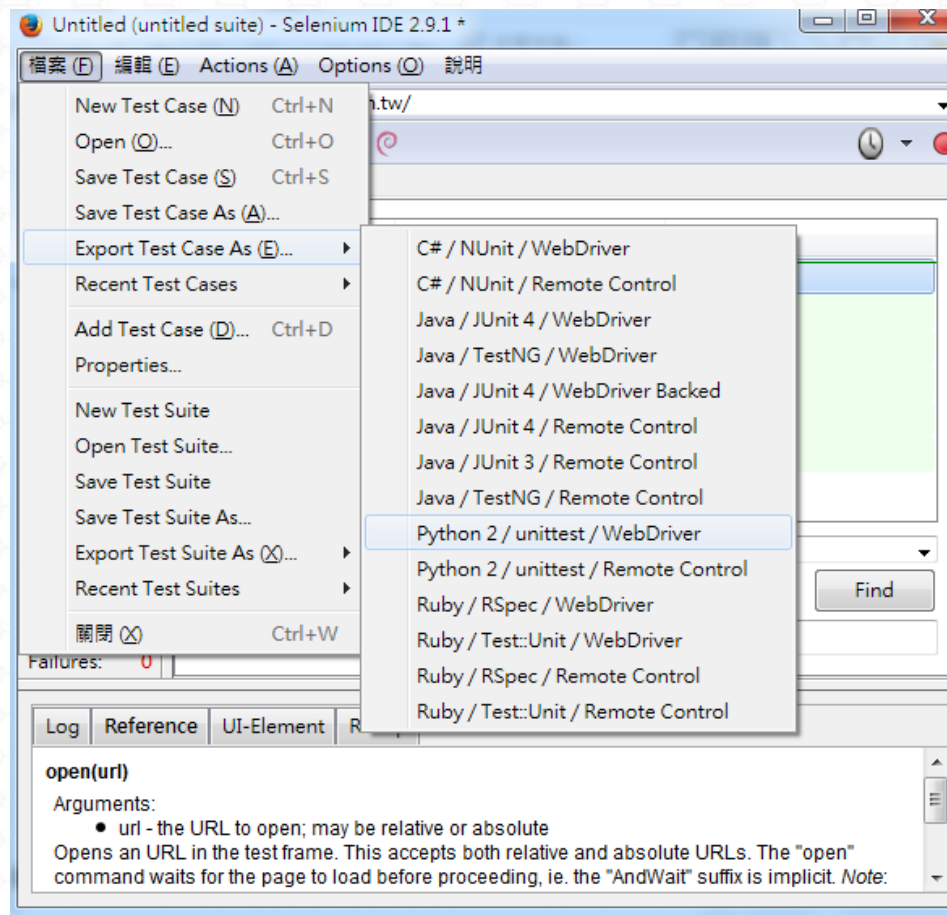
📁 Add to collection
↔ Share this Add-on

<https://addons.mozilla.org/en-US/firefox/addon/selenium-ide/>

使用Selenium Plugin 錄製動作



匯出Selenium 腳本



執行Selenium 腳本

```
# -*- coding: utf-8 -*-
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.support.ui import Select
from selenium.common.exceptions import NoSuchElementException
from selenium.common.exceptions import NoAlertPresentException
from bs4 import BeautifulSoup
import unittest, time, re

driver = webdriver.Firefox()
driver.implicitly_wait(30)
driver.get("http://justdata.yuanta.com.tw/z/zk/zk00-f.asp")

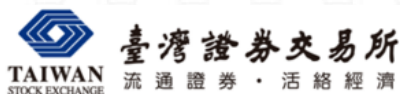
soup = BeautifulSoup(driver.page_source)
print soup

driver.close()
```

需要執行pip install selenium

買賣日報表資料抓取

破解買賣日報表查詢系統



<<買賣日報表查詢系統>>

☐ 一般交易 證券代號:

☐ 鉅額交易



輸入圖形中5碼文數字:

此資料不得逕自散布或販售，
並請詳閱「使用條款」

資料日期:2016/03/25

歡迎使用

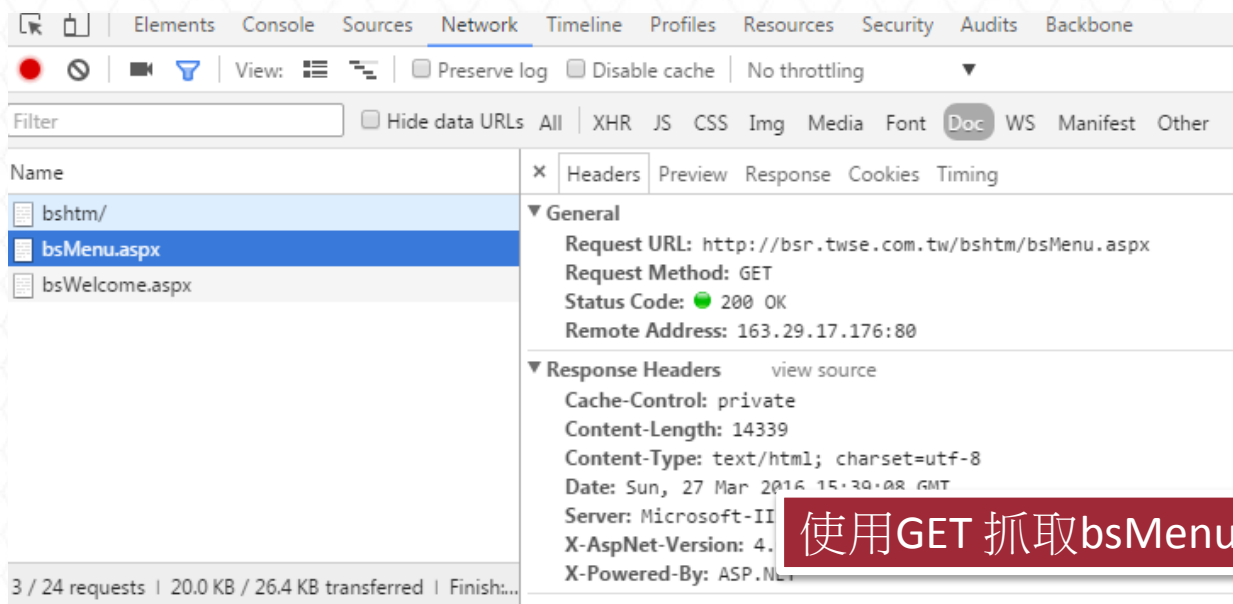
買賣日報表查詢系統

- 本系統僅提供集中市場當日交易資料
- 一般、零股交易報表產製時間:每交易日下午4時
- 鉅額交易報表產製時間:每交易日下午5時30分
- 簡易使用說明:[使用說明](#)
- 《重要提醒公告》本系統自103年12月1日起調整查詢方式,查詢每一檔證券前均輸入驗證碼

<http://bsr.twse.com.tw/bshtm/>

抓取bsMenu.aspx

```
headers = {  
    'User-Agent': 'Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko)  
    Chrome/48.0.2564.116 Safari/537.36'  
}  
rs = requests.session()  
r2 = rs.get('http://bsr.twse.com.tw/bshtm/bsMenu.aspx', headers = headers)
```



```
payload = {
    '__EVENTTARGET': '',
    '__EVENTARGUMENT': '',
    '__LASTFOCUS': '',
    'RadioButton_Normal': 'RadioButton_Normal',
    'TextBox_Stkno': '2330',
    'CaptchaControl1': 'E3QL8',
    'btnOK': '查詢'
}

for inp in soup.select('input[type==hidden]'):
    payload[inp['id']] = inp['value']
```

必須從上一頁的Hidden Value 擷取

```
r3 = rs.post('http://bsr.twse.com.tw/bshtm/bsMenu.aspx', data=payload, headers = headers)
```



OCR 辨認裡面數字

- pytesser

- <https://code.google.com/p/pytesser/>

- ocropus

- <https://code.google.com/p/ocropus/>

- Google Vision API

- <https://cloud.google.com/vision/>

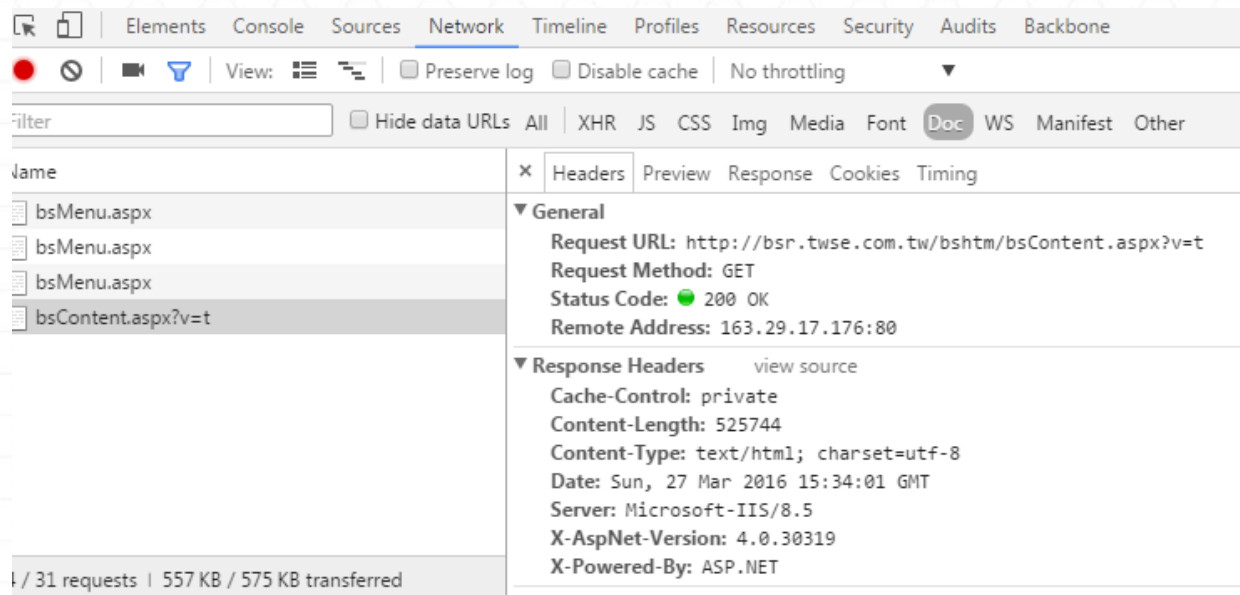
大數學堂

<http://largitdata.com/course/37>

<http://largitdata.com/course/38>

使用GET 取得分點進出資訊

```
r4 =  
rs.get('http://bsr.twse.com.tw/bshtm/bsContent.aspx?v=t',  
headers = headers)  
print r4.text
```



The background features a light blue hexagonal grid pattern. Overlaid on this is a large, faint, circular graphic composed of concentric rings and radial lines, resembling a stylized sun or a target. The text "THANK YOU" is centered in a bold, dark blue, sans-serif font.

THANK YOU