

網路爬蟲介紹

製作人: 黃宥輔



什麼是網路爬蟲?





- 又稱網路機器人、網路蜘蛛...等
- 自動化模擬使用者向伺服器取得資料







- 自動化蒐集、整理資料
- 資料格式標準化
- 搜尋引擎的根基

補充: Robots 協議



• 和搜尋引擎溝通哪些資訊可以或不可以被爬蟲抓到,是約定俗成的規範

範例:

User-agent: *

Allow:/

Disallow: /tmp/

• 允許所有的機器人抓取,但不能抓取/tmp/路徑下的東西

灰色地帶的爬蟲



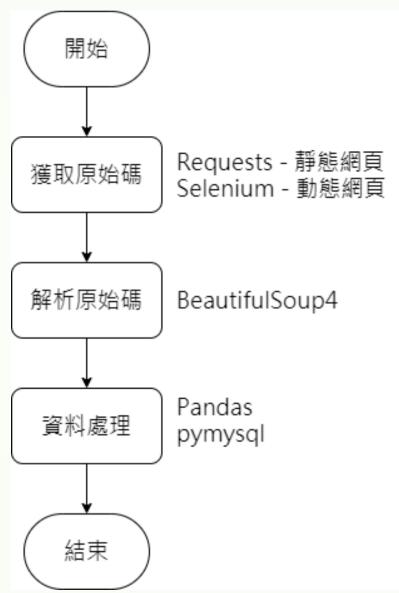


項目	可能的問題	對應法律
利用爬取的資料進行商業活動	可能被視為盜取智慧財產權	著作權法
短時間爬取太多次	對於伺服器造成負擔、癱瘓,如 DDoS 攻擊	《刑法》妨害電腦使用罪
突破某些反爬蟲措施	可能會有刑事風險	
爬取個人、非公開資料	可能會引起爭議	個人資料保護法等

網路爬蟲流程







國立高雄科技大學AI金融科技中心版權所有

網路爬蟲流程





pip install requests

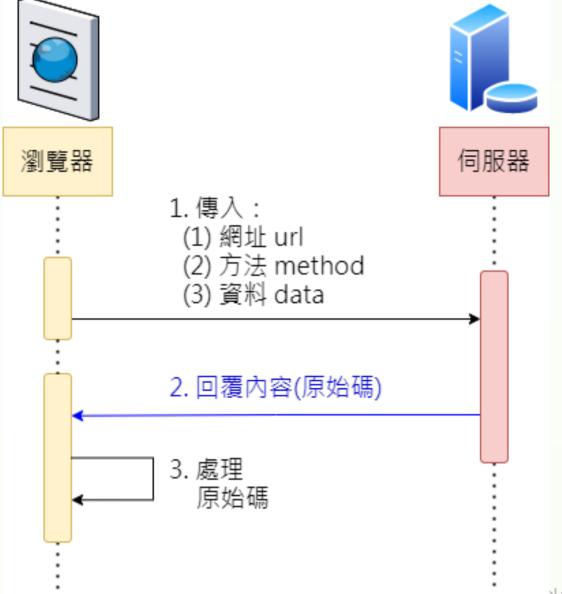
```
■ 選取 系統管理員: Anaconda Prompt (anaconda3)
(base) C:\Windows\system32>pip install requests
Collecting requests
 Using cached requests-2.28.1-py3-none-any.whl (62 kB)
Requirement already satisfied: charset-normalizer<3,>=2 in c:\users\user\anaconda3\lib\site-packages (from requests) (2.0.7)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\user\anaconda3\lib\site-packages (from requests) (1.25.11)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\user\anaconda3\lib\site-packages (from requests) (2020.12.5)
Requirement already satisfied: idna<4,>=2.5 in c:\users\user\anaconda3\lib\site-packages (from requests) (2.8)
Installing collected packages: requests
Successfully installed requests-2.28.1
(base) C:\Windows\system32>
```

瀏覽器如何取得資料?





N 1 全 勁 利 坊 由 心



型土尼北洋技大學 AI 金融科技中心版權所有





- 1. 找到目標網頁 (url)
- 2. 觀察獲取資料模式 (method)
- 3. 找到爬蟲目標元素 (data)
- 4. 撰寫爬蟲

步驟1. 找到目標網頁(url)





- 台灣證券交易所
- https://www.twse.com.tw/zh/



步驟1. 找到目標網頁(url)





交易資訊 > 個股日成交資訊

https://www.twse.com.tw/zh/page/trading/exchange/STOCK DAY.html



步驟1. 找到目標網頁(url)





AI金融科技中心

 \mathcal{C}

https://www.twse.com.tw/zh/page/trading/exchange/STOCK_DAY.html

2













字體大小 A A | 友善服務專區 | 會員專區 | 報表索引 | 日本語ホームページ | English Home | Q

公開資訊觀測站 | 公告快易查 | 基本市況報導 | TWSE 網站:國內業務宣導網站 >

社等职业 關於證交所 交易資訊 指數資訊 上市公司 市場公告 法令規章 輸入時間與代號後, 按下查詢取得資料 交易資訊 ▲ 首頁 ▶ 交易資訊 ▶ 盤後資訊 ▶ 個股日成交資訊 눜 English 盤後資訊 Q 查詢 股票代碼 (瀏覽): 1101 資料日期: 民國 111 年 **∨** 08月 **∨** • 每日收盤行情 ※ 本資訊自民國99年1月4日起開始提供 • 每日市場成交資訊

🚔 列印 / HTML

◆ CSV 下載

111年08月 1101 台泥 各日成交資訊

單位: 元、股

日期 ♦	成交股數 ♦	成交金額 ♦	開盤價 ♦	最高價 ♦	最低價 ♦	收盤價 ♦	漲跌價差 ♦	成交筆數 ♦
111/08/01	17,110,698	671,679,196	39.00	39.50	38.75	39.50	+0.75	9,776
111/08/02	19,955,061	780,513,754	39.05	39.45	39.00	39.15	-0.35	9,799
111/08/03	15,977,710	623,077,746	39.05	39.10	38.75	39.10	-0.05	8,338

- 每日第一上市外國股票成交量值
- 每日創新板股票成交量值
- 每日成交量前一十名證券
- 每5秒委託成交統計
- 各類指數日成交量值
- 個股日成交資訊
- 當日融券賣出與借券賣出成交量值
- 個股日收盤價及月平均價
- 個股月成交資訊
- 個股年成交資訊
- 盤後定價交易

步驟2. 觀察獲取資料模式 (method)





AI.FINIECi

AI金融科技中心



字體大小 A A | 友善服務專區 | 會員專區 | 報表索引 | 日本語ホームページ | English Home | Q

公開資訊觀測站 | 公告快易查 | 基本市況報導 | TWSE 網站: 國內業務宣導網站 ~



Recording network activity...

Perform a request or hit Ctrl + R to record the reload.

Learn more

步驟2. 觀察獲取資料模式 (method)



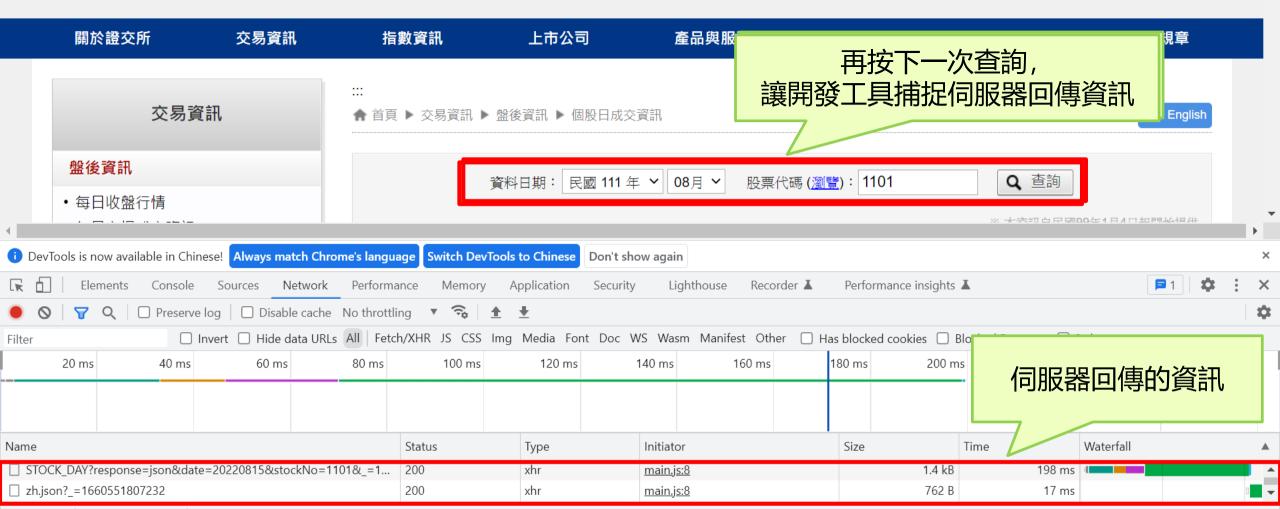


AI金融科技中心



字體大小 A A | 友善服務專區 | 會員專區 | 報表索引 | 日本語ホームページ | English Home | Q

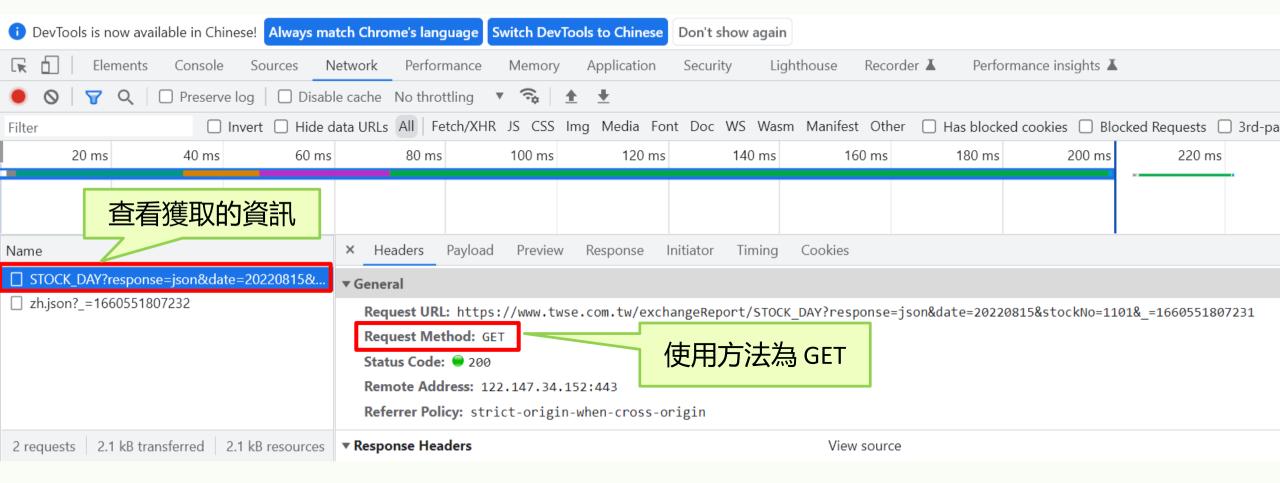
公開資訊觀測站 | 公告快易查 | 基本市況報導 | TWSE 網站: 國內業務宣導網站 v



步驟2. 觀察獲取資料模式 (method)







Get 與 Post 方法



1. Get: 如同明信片,可以直接透過網址取得資料

2. Post:如同信封,需要傳送相關資料才能取得資料









➡️定價,不含鉅額、拍賣、標購。", "ETF證券代號第六碼為K、M、S、C者,表示該ETF以外幣交夠



AI.FINTECH

AI金融科技中心

Data展開有全部頁面上的資料

× Headers Payload Preview Response Initiator Timing Cookies

▼{stat: "OK", date: "20220815", title: "111年08月 1101 台泥 各日成交資訊",...}

▶ data: [["111/08/01", "17,110,698", "671,679,196", "39.00", "39.50", "38.75", "39.50", "+0.75", "9,776"],…]

date: "20220815"

▶ fields: ["日期", "成交股數", "成交金額", "開盤價", "最高價", "還

▶notes: ["符號說明:+/-/x表示漲/跌/不比價", "當日統計資訊含一般▲

stat: "OK"

title: "111年08月 1101 台泥 各日成交資訊"

扁 列印 / HTML ★



111年08月 01 台泥 各日成交資訊

"收盤價", "漲跌價差", "成交筆數"]

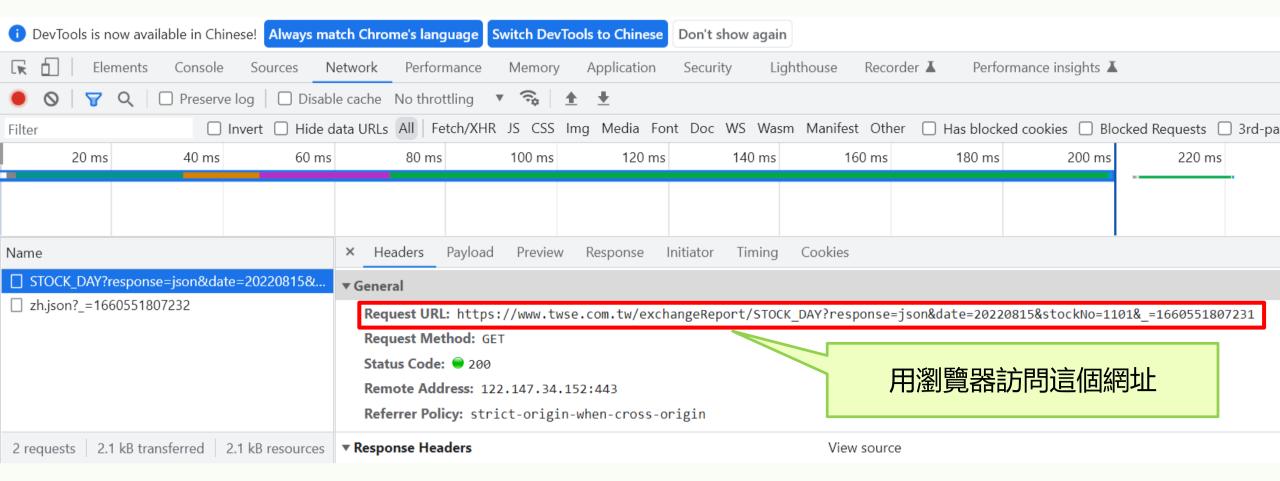
單位:元、股

日期 ♦	成交股數 ♦	成交金額 ♦	開盤價	最高價 ♦	最低價 ♦	收盤價 ♦	漲跌價差 ♦	成交筆數 ♦
111/08/01	17,110,698	671,679,196	39.00	39.50	38.75	39.50	+0.75	9,776
111/08/02	19,955,061	780,513,754	39.05	39.45	39.00	39.15	-0.35	9,799
111/08/03	15,977,710	623,077,746	39.05	39.10	38.75	39.10	-0.05	8,338
111/08/04	18,896,588	733,810,251	39.05	39.20	38.55	38.75	-0.35	10,586
111/08/05	11,583,378	450,508,108	39.10	39.15	38.80	38.85	+0.10	8,390

國立高雄科技大學 AI 金融科技中心版權所有











AI金融科技中心

C

a twse.com.tw/exchangeReport/STOCK_DAY?response=json&date=20220815&stockNo=1101&_=1660551807233







```
{"stat":"OK","date":"20220815","title":"111年08月 1101 台記 各日成交資訊","fields":["日期","成交股數","成交金額","開盤價","最高價","最低價","收盤價","漲跌價差","成交筆數"],"data":[["111/08/01","17,110,698","671,679,196","39.00","39.50","38.75","39.50","+0.75","9,776"],
["111/08/02","19,955,061","780,513,754","39.05","39.45","39.00","39.15","-0.35","9,799"],
["111/08/03","15,977,710","623,077,746","39.05","39.10","38.75","39.10","-0.05","8,338"],
["111/08/04","18,896,588","733,810,251","39.05","39.20","38.55","38.75","-0.35","10,586"],
["111/08/05","11,583,378","450,508,108","39.10","39.15","38.80","38.85","+0.10","8,390"],
["111/08/08","12,651,503","487,687,121","38.85","38.85","38.85","+0.10","8,390"],
["111/08/09","14,831,229","569,783,300","38.85","38.85","38.75","+0.05","9,803"],
["111/08/09","14,831,229","569,783,300","38.60","38.75","38.20","38.75","+0.05","9,803"],
["111/08/12","12,305,588","482,415,186","39.15","39.35","39.30","-0.05","8,548"],
["111/08/15","12,173,803","477,458,592","39.30","39.45","39.30","
["111/08/12","12,305,588","482,415,186","39.15","39.35","39.30","-0.05","8,548"],
["111/08/15","12,173,803","477,458,592","39.30","39.45","39.30","
["0.00","6,719"]],"notes":["符號說明:+/-/X表示派),大不比價","當日統計資訊含一般、零股、盤後定價,不含鉅額、拍賣、標購。","ETF證券代號第六碼為K、M、S、C者,表示該ETF以外幣交易。"]}
```

所有資料都在頁面上





A | 全融科技中心

twse.com.tw/exchangeReport/STOCK_DAY?response=json&date=20220815&stockNo=1101&_=1660551807233



```
{"stat":"OK","date":"20220815","title":"111年08月 1101 台泥 各日成交資訊","fields":["日期","成交股數","成交金額","開盤價","最低價","收盤價","涨跌價差","成交筆數"],"data":[["111/08/01","17,110,698","671,679,196","39.00","39.50","39.50","40.75","9,776"],
["111/08/02","19,955,061","780,513,754","39.05","39.45","39.00","39.15","-0.35","9,799"],
["111/08/03","15,977,710","623,077,746","39.05","39.10","38.75","39.10","-0.05","8,338"],
["111/08/04","18,896,588","733,810,251","39.06","39.20","38.55","38.75","-0.35","10,586"],
["111/08/05","11,583,378","450,508,108","39.10","39.15","38.85","38.85","+0.10","83.90"],
["111/08/09","14,831,229","569,783,300","38.60","38.75","38.20","38.75","+0.05","9,803"],
["111/08/09","14,831,229","569,783,300","38.60","38.75","38.20","38.45","39.35","+0.60","8,627"],
["111/08/12","12,305,588","482,415,186","39.15","39.35","39.00","39.30","-0.05","8,548"],
["111/08/12","12,305,588","482,415,186","39.15","39.35","39.00","39.30","-0.05","8,548"],
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","39.30","
["111/08/15","12,773,803","477,458,592","39.30","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","477,458,592","39.30","39.30","39.30","
["111/08/15","12,173,803","4
```



扁 列印 / HTML ★ CSV 下載

111年08月 1101 台泥 各日成交資訊

單位:元、股

	+ 压							平世 76 成
日期 ♦	成交股數 ♦	成交金額 ♦	開盤價 ♦	最高價 ♦	最低價 ♦	收盤價 ♦	漲跌價差 ♦	成交筆數 🛊
111/08/01	17,110,698	671,679,196	39.00	39.50	38.75	39.50	+0.75	9,776
111/08/02	19,955,061	780,513,754	39.05	39.45	39.00	39.15	-0.35	9,799
111/08/03	15,977,710	623,077,746	39.05	39.10	38.75	39.10	-0.05	8,338
111/08/04	18,896,588	733,810,251	39.05	39.20	38.55	38.75	-0.35	10,586
111/08/05	11,583,378	450,508,108	39.10	39.15	38.80	38.85	+0.10	8,390
111/08/08	12,651,503	487,687,121	38.85	38.85	38.30	38.70	-0.15	7,318
111/08/09	14,831,229	569,783,300	38.60	38.75	38.20	38.75	+0.05	9,803
111/08/10	13,005,899	502,018,549	38.60	38.80	38.35	38.75	0.00	6,512
111/08/11	21,760,411	848,981,726	38.45	39.35	38.45	39.35	+0.60	8,627
111/08/12	12,305,588	482,415,186	39.15	39.35	39.00	39.30	-0.05	8,548
111/08/15	12,173,803	477,458,592	39.30	39.45	39.00	39.30	0.00	6,719





https://www.twse.com.tw/exchangeReport/STOCK_DAY?response=json &date=20220815&stockNo=1101& =1660551807233



單位:元、股

日期 ♦	成交股數 ♦	成交金額 🛊	開盤價 ♦	最高價 ♦	最低價 ♦	收盤價 ♦	漲跌價差 ♦	成交筆數 🛊
111/08/01	17,110,698	671,679,196	39.00	39.50	38.75	39.50	+0.75	9,776
111/08/02	19,955,061	780,513,754	39.05	39.45	39.00	39.15	-0.35	9,799
111/08/03	15,977,710	623,077,746	39.05	39.10	38.75	39.10	-0.05	8,338

國北向海州及八字 AI 亚图叶拉中心版權所





- 請觀察以下的資料與網址
- 代號為 2330 的股票
- 時間為 111 年 7 月

JSON 資料格式



- 1. 源自 Javascript 的資料格式
- 2. 如同 Python 的 dict,一個鍵值 (Key) 配上一個值 (Value)

```
資料格式:

{

  "stat": "OK",

  "date": "20210815",

  "title": "111年08月 1101 台泥 各日成交資訊",

  "fields": [...],

  "data": [[...], [...], ...],

}
```



```
1 # -*- coding: utf-8 -*-
  import requests
  # 目標網址
  url = "https://www.twse.com.tw/exchangeReport/STOCK DAY?res;
  res = requests.get(url) # 以 GET 方法取得網頁原始碼
  data = res.text # 取出網頁原始碼的內容
8
  print(data)
  print(type(data)) # 查看型別,為字串型別
```



```
Console 1/A
ر ا ∕کوره ر صوره⊤ ر دد،دد ر د
["111/08/12","12,305,588","482,415,186","39.15","39.35","39.0
0","39.30","-0.05","8,548"],
["111/08/15","12,173,803","477,458,592","39.30","39.45","39.0
0","39.30"," 0.00","6,719"],
["111/08/16","13,078,181","514,161,618","39.50","39.55","39.1
0","39.35","+0.05","6,445"],
["111/08/17","16,463,976","651,431,576","39.40","40.00","39.2
0","40.00","+0.65","9,075"]],"notes":["符號說明:+/-/X表示漲/跌/
不比價","當日統計資訊含一般、零股、盤後定價,不含鉅額、拍賣、標
購。","ETF證券代號第六碼為K、M、S、C者,表示該ETF以外幣交易。"]}
<class 'str'>
```





網頁原始碼部分內容

```
Console 1/A
["111/08/12","12,305,588","482,415,186","39.15","39.35","39.0
0","39.30","-0.05","8,548"],
["111/08/15","12,173,803","477,458,592","39.30","39.45","39.0
0","39.30"," 0.00","6,719"],
["111/08/16","13,078,181","514,161,618","39.50","39.55","39.1
0","39.35","+0.05","6,445"],
["111/08/17","16,463,976","651,431,576","39.40","40.00","39.2
0","40.00","+0.65","9,075"]],"notes":["符號說明:+/-/X表示漲/跌/
不比價","當日統計資訊含一般、零股、盤後定價,不含鉅額、拍賣、標
購。","ETF證券代號第六碼為K、M、S、C者,表示該ETF以外幣交易。"]}
<class 'str'>
```

網頁原始碼為字串型別

國立高雄科技大學 AI 金融科技中心版權所有



```
# -*- coding: utf-8 -*-
    import requests, json
   # 目標網址
    url = "https://www.twse.com.tw/exchangeReport/STOCK DAY?response=json&date
    res = requests.get(url) # 以 GET 方法取得網頁原始碼
    data = res.text # 取出網頁原始碼的內容
    data = json.loads(data) # 將資料以 json 套件,轉換成字典格式
    print(data)
10
    print(type(data)) # 查看格式,結果為字典(dict)
```



```
# -*- coding: utf-8 -*-
    import requests, json
                           引用 ison 套件
   # 目標網址
    url = "https://www.twse.com.tw/exchangeReport/STOCK DAY?response=json&date
    res = requests.get(url) # 以 GET 方法取得網頁原始碼
    data = res.text # 取出網頁原始碼的內容
                                                           將文字轉換為
                                                             字典格式
   data = json.loads(data) # 將資料以 json 套件,轉換成字典格式
    print(data)
10
    print(type(data)) # 查看格式,結果為字典(dict)
```



```
Console 2/A
'39.30', '-0.05', '8,548'], ['111/08/15', '12,173,803',
'477,458,592', '39.30', '39.45', '39.00', '39.30', ' 0.00',
'6,719'], ['111/08/16', '13,078,181', '514,161,618', '39.50',
'39.55', '39.10', '39.35', '+0.05', '6,445'], ['111/08/17',
'16,463,976', '651,431,576', '39.40', '40.00', '39.20',
'40.00', '+0.65', '9,075']], 'notes': ['符號說明:+/-/X表示漲/
跌/不比價', '當日統計資訊含一般、零股、盤後定價,不含鉅額、拍賣、
標購。', 'ETF證券代號第六碼為K、M、S、C者,表示該ETF以外幣交
易。'1}
<class 'dict'>
In [3]:
```



部分資料

```
Console 2/A
'39.30', '-0.05', '8,548'], ['111/08/15', '12,173,803'<u>,</u>
'477,458,592', '39.30', '39.45', '39.00', '39.30', ' 0.00',
'6,719'], ['111/08/16', '13,078,181', '514,161,618', '39.50'
'39.55', '39.10', '39.35', '+0.05', '6,445'], ['111/08/17',
'16,463,976', '651,431,576', '39.40', '40.00', '39.20',
'40.00', '+0.65', '9,075']], 'notes': ['符號說明:+/-/X表示漲/
跌/不比價', '當日統計資訊含一般、零股、盤後定價,不含鉅額、拍賣、
標購。', 'ETF證券代號第六碼為K、M、S、C者,表示該ETF以外幣交
易。' ] }
```

<class 'dict'>

字典格式

In [3]:





```
# -*- codina: utf-8 -*-
    import requests, json
    import pandas as pd
   # 目標網址
    url = "https://www.twse.com.tw/exchangeReport/STOCK DAY?response=json&date=20
    res = requests.get(url) # 以 GET 方法取得網頁原始碼
8
    data = res.text # 取出網頁原始碼的內容
    data = json.loads(data) # 將資料以 json 套件,轉換成字典格式
10
11
    print(data.keys()) # 查看字典的鍵(key)
12
13
    print("----")
    print(data["fields"]) # 查看標題
14
    print("----")
15
    print(data["data"]) # 查看資料
16
17
    # 以標題與資料建構DataFrame
18
    df = pd.DataFrame.from_records( data["data"],
19
20
                                columns= data["fields"] )
    print(df) # 查看DataFrame
    df.to_excel("1101_20220815.xlsx", index = False) # 儲存資料到EXCEL
```

"心版權所有





```
# -*- coding: utf-8 -*-
    import requests, json
    import pandas as pd
   # 目標網址
    url = "https://www.twse.com.tw/exchangeReport/STOCK DAY?response=json&date=20
    res = requests.get(url) # 以 GET 方法取得網頁原始碼
8
    data = res.text # 取出網頁原始碼的內容
    data = json.loads(data) # 將資料以 json 套件,轉換成字典格式
11
    print(data.keys()) # 查看字典的鍵(key)
12
                                             輸出結果
    print("----")
13
    print(data["fields"]) # 查看標題
14
    print("----")
15
    print(data["data"]) # 查看資料
                                                     轉換成資料表並儲存
16
17
   # 以標題與資料建構DataFrame
18
    df = pd.DataFrame.from records( data["data"],
19
                               columns= data["fields"] )
20
    print(df) # 查看DataFrame
    df.to_excel("1101_20220815.xlsx", index = False) # 儲存資料到EXCEL
```

1心版權所有



```
Console 4/A
In [1]: runfile('C:/Users/user/Desktop/AIGO 課程/網路爬蟲程式交易最佳化投資組
合分析教材重製/3. TWSE爬蟲 資料儲存.py', wdir='C:/Users/user/Desktop/AIGO 課
程/網路爬蟲程式交易最佳化投資組合分析教材重製')
dict keys(['stat', 'date', 'title', 'fields', 'data', 'notes'])
['日期','成交股數','成交金額','開盤價','最高價','最低價','收盤價','漲跌
價差','成交筆數']
[['111/08/01', '17,110,698', '671,679,196', '39.00', '39.50', '38.75',
'39.50', '+0.75', '9,776'], ['111/08/02', '19,955,061', '780,513,754',
'39.05', '39.45', '39.00', '39.15', '-0.35', '9,799'], ['111/08/03',
'15,977,710', '623,077,746', '39.05', '39.10', '38.75', '39.10', '-0.05',
```



```
Console 4/A
In [1]: runfile('C:/Users/user/Desktop/AIGO 課程/網路爬蟲程式交易最佳化投資組
            合分析教材重製/3.
dict keys(['stat', 'date', 'title', 'fields', 'data', 'notes'])
['日期','成交股數','成交金額','開盤價','最高價','最低價','收盤價','漲跌
價差','成交筆數']
[['111/08/01', '17,110,698', '671,679,196', '39.00', '39.50', '38.75',
'39.50', '+0.75', '9,776'], ['111/08/02', '19,955,061', '780,513,754',
'39.05', '39.45', '39.00', '39.15', '-0.35', '9,799'], ['111/08/03',
'15,977,710', '623,077,746', '39.05', '39.10', '38.75', '39.10', '-0.05',
```

print(data["data"])





	Console 4/A 🗵									≡
	日其	月 成交	股數	成交金額		開盤價		最低價	收盤	^
價	漲跌價差	成交筆數								
0	111/08/01	17,110,698	671,679,196	39.00		38.75	39.50	+0.75	9,776	
1	111/08/02	19,955,061	780,513,754	39.05		39.00	39.15	-0.35	9,799	
2	111/08/03	15,977,710	623,077,746	39.05		38.75	39.10	-0.05	8,338	
3	111/08/04	18,896,588	733,810,251	39.05		38.55	38.75	-0.35	10,586	
4	111/08/05	11,583,378	450,508,108	39.10		38.80	38.85	+0.10	8,390	
5	111/08/08	12,651,503	487,687,121	38.85		38.30	38.70	-0.15	7,318	
6	111/08/09	14,831,229	569,783,300	38.60		38.20	38.75	+0.05	9,803	
7	111/08/10	13,005,899	502,018,549	38.60		38.35	38.75	0.00	6,512	
8	111/08/11	21,760,411	848,981,726	38.45		38.45	39.35	+0.60	8,627	
9	111/08/12	12,305,588	482,415,186	39.15		39.00	39.30	-0.05	8,548	
10	111/08/15	12,173,803	477,458,592	39.30	• • •	39.00	39.30	0.00	6,719	~

print(df)

步驟4. 撰寫爬蟲

儲存到 Excel 中的結果





	А	В	С	D	Е	F	G	Н	I
1	日期	成交股數	成交金額	開盤價	最高價	最低價	收盤價	漲跌價差	成交筆數
2	111/08/01	17,110,698	671,679,19	39.00	39.50	38.75	39.50	+0.75	9,776
3	111/08/02	19,955,06	780,513,75	39.05	_	39.00	39.15	-0.35	9,799
4		15,977,710			39.10	38.75	39.10	-0.05	8,338
5	1	18,896,588			_	_	38.75	-0.35	10,586
6		11,583,378		_	_	_	38.85	+0.10	8,390
7		12,651,503		_	38.85	38.30	38.70	-0.15	7,318
8		14,831,229		_		_	38.75	+0.05	9,803
9		13,005,899			38.80	38.35	38.75	0.00	6,512
10		21,760,41			_	_	39.35	+0.60	8,627
11		12,305,588		_	39.35	39.00	39.30	-0.05	8,548
12	111/08/15	12,173,803	477,458,59	39.30	39.45	39.00	39.30	0.00	6,719
13		13,078,18			39.55		39.35	+0.05	6,445
14	111/08/17	16,463,970	651,431,57	39.40	40.00	39.20	40.00	+0.65	9,075

國立高雄科技大學AI金融科技中心版權所有





```
# -*- coding: utf-8 -*-
    import requests, json, time, datetime
 3
    import pandas as pd
 4
 5
    def main(date, stockNo):
 6
        # 目標網址
        url = f"https://www.twse.com.tw/exchangeReport/STOCK_DAY?response=json&date
8
        res = requests.get(url) # 以 GET 方法取得網頁原始碼
 9
10
        data = res.text # 取出網頁原始碼的內容
        data = json.loads(data) # 將資料以 json 套件,轉換成字典格式
11
12
13
        # 以標顋與資料建構DataFrame
        df = pd.DataFrame.from_records( data["data"],
14
15
                                     columns= data["fields"] )
```





- 請抓取的資料
- 代號為 2330 的股票
- 時間為 111 年 7 月

5

6

8

9

10

12

13

14

15





將原本的程式包裝成函數,依據傳入 的date與stockNo參數改變網址

```
# -*- coding: utf-8 -*-
import requests, json, time, datetime
import pandas as pd
```

```
def main(date, stockNo):
   # 目標網址
   url = f"https://www.twse.com.tw/exchangeReport/STOCK_DAY?response=json&date
   res = requests.get(url) # 以 GET 方法取得網頁原始碼
   data = res.text # 取出網頁原始碼的內容
   data = json.loads(data) # 將資料以 json 套件,轉換成字典格式
   # 以標顋與資料建構DataFrame
   df = pd.DataFrame.from_records( data["data"],
                              columns= data["fields"] )
```





url =
f"https://www.twse.com.tw/exchangeReport/STOCK_DAY?respo
nse=json&date={date}01&stockNo={stockNo}&_=166055180723
3"

def main(date, stockNo):

將傳入的參數帶入網址中



```
17
        # 將所有欄位當作字串處理,把短號取代成空字串(去除短號)
        for c in df.columns:
18
            if c == "日期": # 將民國轉西元如 104/01/01 變成 2015-01-01
19
20
                df[c] = df[c].str.replace("/","") # 去除 /
21
                df[c] = df[c].astype(int) + 19110000 # 民國轉西元 ,年 + 1911
22
               # 轉換成 datetime 格式
                df[c] = pd.to datetime(df[c].astype(str),
23
24
                                     format = "%Y%m%d".
25
                                     errors = "coerce")
26
            else:
27
               # 日期以外的欄位,則賺換成數字格式
                df[c] = df[c].str.replace(",","")
28
                df[c] = pd.to numeric(df[c] ,
29
30
                                    errors = "coerce")
31
32
        file_name = f"{stockNo}_{date}.xlsx"
        df.to_excel(f"股價資料/{file_name}", index = False) # 儲存資料到EXCEL
33
```

33

將抓取的資歷進行整理



```
17
          將所有欄位當作字串處理,把逗號取代成空字串(去除逗號)
18
        for c in df.columns:
            if c == "日期": # 將民國轉西元如 104/01/01 變成 2015-01-01
19
               df[c] = df[c].str.replace("/","") # 去除 /
20
21
               df[c] = df[c].astype(int) + 19110000 # 民國轉西元 ,年 + 1911
22
               # 轉換成 datetime 格式
               df[c] = pd.to datetime(df[c].astype(str),
23
24
                                     format = "%Y%m%d",
25
                                     errors = "coerce")
26
            else:
27
               # 日期以外的欄位,則賺換成數字格式
               df[c] = df[c].str.replace(",","")
28
               df[c] = pd.to_numeric(df[c] ,
29
                                                               儲存資料
30
                                    errors = "coerce")
31
32
```

```
file_name = f"{stockNo}_{date}.xlsx"

df.to_excel(f"股價資料/{file_name}", index = False) # 儲存資料到EXCEL
```





```
date list = [] # 日期列表
35
    current date = 202208
36
    for y in range(2015, 2023):
37
        for m in range(1, 13):
38
           date = v * 100 + m
39
           if date <= current date:</pre>
40
               date list.append(str(date))
41
42
43
    # 0050成分股列表
    "2227", "2207", "301", "1326", "1326", "2002", "2207", "2227"
44
45
    # 跑回圈抓取所有期間50檔股票的資料
46
47
    for date in date list:
48
        for stockNo in stockNo list:
           print(date, stockNo, datetime.datetime.now())
49
50
           try:
51
               main(date, stockNo)
           except KeyError: # 沒有資料可能會出錯
52
53
               pass
54
55
           time.sleep(5) # 避免被誤認為DDOS攻擊,每次造訪後都休息 5 秒
```





```
date list = [] # 日期列表
35
    current date = 202208
36
    for y in range(2015, 2023):
37
38
        for m in range(1, 13):
            date = v * 100 + m
39
            if date <= current date:</pre>
40
                date list.append(str(date))
41
42
43
    # 0050成分股列表
    stockNo_list = ["1101" ,"1216" , "1301" ,"1303" ,"1326" ,"2002" ,"2207" ,"2227"
44
45
    # 跑回圈抓取所有期間50檔股票的資料
46
    for date in date list:
47
48
        for stockNo in stockNo list:
            print(date, stockNo, datetime.datetime.now())
49
50
            try:
51
                main(date, stockNo)
52
            except KeyError: # 沒有資料可能會出錯
53
                pass
54
55
            time.sleep(5) # 避免被誤認為DDOS攻擊,每次造訪後都休息 5 秒
```

建構日期列表

建構股票代號列表

走訪日期、股票代號 列表, 抓取資料





AI.FINTECH

A I 金融科技中心

名稱	修改日期	類型	大小
1101_201501.xlsx	2022/8/17 下午 06:17	Microsoft Excel 工作	7 KB
1101_201502.xlsx	2022/8/17 下午 06:22	Microsoft Excel 工作	7 KB
1101_201503.xlsx	2022/8/17 下午 06:26	Microsoft Excel 工作	7 KB
1101_201504.xlsx	2022/8/17 下午 06:30	Microsoft Excel 工作	7 KB
1101_201505.xlsx	2022/8/17 下午 06:35	Microsoft Excel 工作	7 KB
1101_201506.xlsx	2022/8/17 下午 06:39	Microsoft Excel 工作	7 KB
1101_201507.xlsx	2022/8/17 下午 06:43	Microsoft Excel 工作	7 KB
1101_201508.xlsx	2022/8/17 下午 06:48	Microsoft Excel 工作	7 KB
1101_201509.xlsx	2022/8/17 下午 06:52	Microsoft Excel 工作	7 KB
1101_201510.xlsx	2022/8/17 下午 06:56	Microsoft Excel 工作	7 KB
1101_201511.xlsx	2022/8/17 下午 07:01	Microsoft Excel 工作	7 KB
1101_201512.xlsx	2022/8/17 下午 07:05	Microsoft Excel 工作	7 KB
1101_201601.xlsx	2022/8/17 下午 07:09	Microsoft Excel 工作	7 KB
1101_201602.xlsx	2022/8/17 下午 07:14	Microsoft Excel 工作	7 KB
1101_201603.xlsx	2022/8/17 下午 07:18	Microsoft Excel 工作	7 KB
1101_201604.xlsx	2022/8/17 下午 07:22	Microsoft Excel 工作	7 KB
1101_201605.xlsx	2022/8/17 下午 07:26	Microsoft Excel 工作	7 KB
1101_201606.xlsx	2022/8/17 下午 07:31	Microsoft Excel 工作	7 KB
1101_201607.xlsx	2022/8/17 下午 07:35	Microsoft Excel 工作	7 KB
1101_201608.xlsx	2022/8/17 下午 07:39	Microsoft Excel 工作	7 KB
1101_201609.xlsx	2022/8/17 下午 07:44	Microsoft Excel 工作	7 KB
1101_201610.xlsx	2022/8/17 下午 07:48	Microsoft Excel 工作	7 KB 温中
	名稱 1101_201501.xlsx 1101_201502.xlsx 1101_201503.xlsx 1101_201504.xlsx 1101_201506.xlsx 1101_201506.xlsx 1101_201508.xlsx 1101_201509.xlsx 1101_201510.xlsx 1101_201511.xlsx 1101_201512.xlsx 1101_201601.xlsx 1101_201603.xlsx 1101_201605.xlsx 1101_201605.xlsx 1101_201605.xlsx 1101_201607.xlsx 1101_201607.xlsx 1101_201607.xlsx 1101_201608.xlsx 1101_201608.xlsx 1101_201608.xlsx 1101_201608.xlsx 1101_201608.xlsx 1101_201608.xlsx 1101_201609.xlsx	1101_201501.xlsx 2022/8/17 下午 06:17 1101_201502.xlsx 2022/8/17 下午 06:22 1101_201503.xlsx 2022/8/17 下午 06:26 1101_201504.xlsx 2022/8/17 下午 06:30 1101_201505.xlsx 2022/8/17 下午 06:35 1101_201506.xlsx 2022/8/17 下午 06:39 1101_201507.xlsx 2022/8/17 下午 06:43 1101_201509.xlsx 2022/8/17 下午 06:48 1101_201510.xlsx 2022/8/17 下午 06:50 1101_201511.xlsx 2022/8/17 下午 07:01 1101_201512.xlsx 2022/8/17 下午 07:05 1101_201601.xlsx 2022/8/17 下午 07:05 1101_201603.xlsx 2022/8/17 下午 07:14 1101_201605.xlsx 2022/8/17 下午 07:18 1101_201605.xlsx 2022/8/17 下午 07:22 1101_201605.xlsx 2022/8/17 下午 07:35 1101_201608.xlsx 2022/8/17 下午 07:35 1101_201608.xlsx 2022/8/17 下午 07:35 1101_201609.xlsx 2022/8/17 下午 07:34 1101_201609.xlsx 2022/8/17 下午 07:35 1101_201609.xlsx 2022/8/17 下午 07:34 1101_201609.xlsx 2022/8/17 下午 07:44 1	### 1101_201501.xlsx 2022/8/17 下午 06:17 Microsoft Excel 工作 1101_201502.xlsx 2022/8/17 下午 06:22 Microsoft Excel 工作 1101_201503.xlsx 2022/8/17 下午 06:26 Microsoft Excel 工作 1101_201503.xlsx 2022/8/17 下午 06:26 Microsoft Excel 工作 1101_201504.xlsx 2022/8/17 下午 06:30 Microsoft Excel 工作 1101_201505.xlsx 2022/8/17 下午 06:35 Microsoft Excel 工作 1101_201506.xlsx 2022/8/17 下午 06:39 Microsoft Excel 工作 1101_201507.xlsx 2022/8/17 下午 06:43 Microsoft Excel 工作 1101_201508.xlsx 2022/8/17 下午 06:48 Microsoft Excel 工作 1101_201509.xlsx 2022/8/17 下午 06:52 Microsoft Excel 工作 1101_201510.xlsx 2022/8/17 下午 06:56 Microsoft Excel 工作 1101_201510.xlsx 2022/8/17 下午 07:01 Microsoft Excel 工作 1101_201511.xlsx 2022/8/17 下午 07:05 Microsoft Excel 工作 1101_201601.xlsx 2022/8/17 下午 07:09 Microsoft Excel 工作 1101_201601.xlsx 2022/8/17 下午 07:14 Microsoft Excel 工作 1101_201603.xlsx 2022/8/17 下午 07:18 Microsoft Excel 工作 1101_201603.xlsx 2022/8/17 下午 07:18 Microsoft Excel 工作 1101_201604.xlsx 2022/8/17 下午 07:26 Microsoft Excel 工作 1101_201605.xlsx 2022/8/17 下午 07:26 Microsoft Excel 工作 1101_201605.xlsx 2022/8/17 下午 07:31 Microsoft Excel 工作 1101_201607.xlsx 2022/8/17 下午 07:31 Microsoft Excel 工作 1101_201608.xlsx 2022/8/17 下午 07:31 Microsoft Excel 工作 1101_201608.xlsx 2022/8/17 下午 07:39 Microsoft Excel 工作 1101_201608.xlsx 2022/8/17 下午 07:39 Microsoft Excel 工作 1101_201609.xlsx 2022/8/17 下午 07:39 Microsoft Excel 工作 1101_201609.xlsx 2022/8/17 下午 07:39 Microsoft Excel 工作 1101_201609.xlsx 2022/8/17 下午 07:34 Microsoft Excel 工作 1101_201609.xlsx 2022/8/17 下午 07:44 Microsoft Excel 工作 1101_201609.xlsx 2022/8/17 下午 07:44 Microsoft Excel 工作 1101_20160





date_list

```
Console 1/A
             Console 2/A
['201501', '201502', '201503', '201504', '201505', '201506', '201507',
'201508', '201509', '201510', '201511', '201512', '201601', '201602',
'201603', '201604', '201605', '201606', '201607', '201608', '201609',
'201610', '201611', '201612', '201701', '201702', '201703', '201704',
'201705', '201706', '201707', '201708', '201709', '201710', '201711',
'201712', '201801', '201802', '201803', '201804', '201805', '201806',
'201807', '201808', '201809', '201810', '201811', '201812', '201901',
'201902', '201903', '201904', '201905', '201906', '201907', '201908',
'201909', '201910', '201911', '201912', '2020001', '202002', '202003',
'202004', '202005', '202006', '202007', '202008', '202009', '202010',
'202011', '202012', '202101', '202102', '202103', '202104', '202105',
'202106', '202107', '202108', '202109', '202110', '202111', '202112',
'202201', '202202', '202203', '202204', '202205', '202206', '202207',
'202208']
```





stockNo_list

公司簡稱	股票代號	公司簡稱	股票代號	公司簡稱	股票代號	公司簡稱	股票代號	公司簡稱	股票代號
台泥	1101	鴻海	2317	中華電	2412	開發金	2883	欣興	3037
統一	1216	國巨	2327	聯發科	2454	玉山金	2884	台灣大	3045
台塑	1301	台積電	2330	長榮	2603	元大金	2885	日月光投控	3711
南亞	1303	佳世達	2352	陽明	2609	兆豐金	2886	遠傳	4904
台化	1326	華碩	2357	華航	2610	台新金	2887	上海商銀	5876
中鋼	2002	瑞昱	2379	萬海	2615	中信金	2891	合庫金	5880
和泰車	2207	廣達	2382	彰銀	2801	第一金	2892	台塑化	6505
裕日車	2227	研華	2395	華南金	2880	統一超	2912	力積電	6770
聯電	2303	南亞科	2408	富邦金	2881	大立光	3008	南電	8046
台達電	2308	友達	2409	國泰金	2882	聯詠	3034	豐泰	9910





A I 金融科技中心

名稱	修改日期	類型	大小
1101_201501.xlsx	2022/8/17 下午 06:17	Microsoft Excel 工作	7 KB
1101_201502.xlsx	2022/8/17 下午 06:22	Microsoft Excel 工作	7 KB
1101_201503.xlsx	2022/8/17 下午 06:26	Microsoft Excel 工作	7 KB
1101_201504.xlsx	2022/8/17 下午 06:30	Microsoft Excel 工作	7 KB
1101_201505.xlsx	2022/8/17 下午 06:35	Microsoft Excel 工作	7 KB
1101_201506.xlsx	2022/8/17 下午 06:39	Microsoft Excel 工作	7 KB
1101_201507.xlsx	2022/8/17 下午 06:43	Microsoft Excel 工作	7 KB
1101_201508.xlsx	2022/8/17 下午 06:48	Microsoft Excel 工作	7 KB
1101_201509.xlsx	2022/8/17 下午 06:52	Microsoft Excel 工作	7 KB
1101_201510.xlsx	2022/8/17 下午 06:56	Microsoft Excel 工作	7 KB
1101_201511.xlsx	2022/8/17 下午 07:01	Microsoft Excel 工作	7 KB
1101_201512.xlsx	2022/8/17 下午 07:05	Microsoft Excel 工作	7 KB
1101_201601.xlsx	2022/8/17 下午 07:09	Microsoft Excel 工作	7 KB
1101_201602.xlsx	2022/8/17 下午 07:14	Microsoft Excel 工作	7 KB
1101_201603.xlsx	2022/8/17 下午 07:18	Microsoft Excel 工作	7 KB
1101_201604.xlsx	2022/8/17 下午 07:22	Microsoft Excel 工作	7 KB
1101_201605.xlsx	2022/8/17 下午 07:26	Microsoft Excel 工作	7 KB
1101_201606.xlsx	2022/8/17 下午 07:31	Microsoft Excel 工作	7 KB
1101_201607.xlsx	2022/8/17 下午 07:35	Microsoft Excel 工作	7 KB
1101_201608.xlsx	2022/8/17 下午 07:39	Microsoft Excel 工作	7 KB
1101_201609.xlsx	2022/8/17 下午 07:44	Microsoft Excel 工作	7 KB
1101_201610.xlsx	2022/8/17 下午 07:48	Microsoft Excel 工作	7 KB
			/ 、

改良程式 - 整理資料

```
import pandas as pd
    from os import listdir
    path = "股價資料/"
    save path = "股價資料 整合/"
    file list = listdir(path) # 讀取所有資料
    file dict = {}
10
11
    # 將檔案名稱根據代號放入字典
    for i in file list:
12
        Stock id = i.split("_")[0]
13
        if Stock id not in file dict.keys():
14
            file dict[Stock id] = [i]
15
16
        else:
17
            file dict[Stock id].append(i)
18
19
    # 根據字典的鍵進行資料合併
    for Stock id in file dict.keys():
20
21
        stock_file_list = file_dict[Stock_id]
        concated df = pd.DataFrame() # 整合的DF
22
23
24
        # 讀取所有資料
25
        for i in stock file list:
            df = pd.read_excel(f"{path}{i}") # 資料讀取
26
            concated df = pd.concat([concated df, df], axis = 0) # 上下合併資料
27
28
        concated_df.to_excel(f"{save_path}{Stock_id}.xlsx", index = False) # 儲存資料
29
        print(Stock id)
30
```

改良程式 -整理資料





I.FINTECH

AI金融科技中心

名稱

- 1101 201501.xlsx
- 1101 201502.xlsx
- 1101 201503.xlsx
- 1101 201504.xlsx
- 1101 201505.xlsx
- 1101 201506.xlsx
- 1101 201507.xlsx
- 1101 201508.xlsx
- 1101 201509.xlsx
- 1101_201510.xlsx
- 1101 201511.xlsx
- 1101 201512.xlsx
- 1101 201601.xlsx
- 1101_201602.xlsx
- 1101_201603.xlsx
- 1101 201604.xlsx
- 1101_201605.xlsx
- 1101_201606.xlsx
- 1101 201607.xlsx
- 1101_201608.xlsx
- 1101_201609.xlsx
- 1101_201610.xlsx



- 1101.xlsx
- 1216.xlsx
- 1301.xlsx
- 1303.xlsx
- 1326.xlsx
- 2002.xlsx
- 2207.xlsx
- 2227.xlsx
- 2303.xlsx
- 2308.xlsx
- 2317.xlsx
- 2327.xlsx
- 2330.xlsx
- 2352.xlsx
- 2357.xlsx
- 2379.xlsx
- 2382.xlsx
- 2395.xlsx
- 2408.xlsx
- 2409.xlsx
- 2412.xlsx
- 2454.xlsx
- 2603.xlsx
- 2609.xlsx
- 2610.xlsx

全融科技中心版權所有

划立高机





```
Anaconda Prompt (anaconda3)
                                                                                                                       X
                                                        pip install yfinance
(base) C:\Users\user>pip install vfinance
Collecting vfinance
 Downloading vfinance-0.1.74-pv2.pv3-none-anv.whl (27 kB)
Requirement already satisfied: multitasking>=0.0.7 in c:\users\user\anaconda3\lib\site-packages (from yfinance) (0.0.10)
Requirement already satisfied: lxml>=4.5.1 in c:\users\user\anaconda3\lib\site-packages (from yfinance) (4.6.3)
Requirement already satisfied: requests>=2.26 in c:\users\user\anaconda3\lib\site-packages (from yfinance) (2.28.1)
Requirement already satisfied: numpy>=1.15 in c:\users\user\anaconda3\lib\site-packages (from yfinance) (1.20.1)
Requirement already satisfied: pandas>=0.24.0 in c:\users\user\anaconda3\lib\site-packages (from yfinance) (1.2.4)
Requirement already satisfied: python-dateutil>=2.7.3 in c:\users\user\anaconda3\lib\site-packages (from pandas>=0.24.0-
>vfinance) (2.8.1)
Requirement already satisfied: pytz>=2017.3 in c:\users\user\anaconda3\lib\site-packages (from pandas>=0.24.0->yfinance)
 (2021.1)
Requirement already satisfied: six>=1.5 in c:\users\user\anaconda3\lib\site-packages (from python-dateutil>=2.7.3->panda
s = 0.24.0 - \text{yfinance} (1.15.0)
Requirement already satisfied: idna<4,>=2.5 in c:\users\user\anaconda3\lib\site-packages (from requests>=2.26->yfinance)
 (2.8)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\user\anaconda3\lib\site-packages (from requests>=2.26->
vfinance) (1.25.11)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\user\anaconda3\lib\site-packages (from requests>=2.26->yfi
nance) (2020.12.5)
Requirement already satisfied: charset-normalizer<3,>=2 in c:\users\user\anaconda3\lib\site-packages (from requests>=2.2
6->yfinance) (2.0.7)
Installing collected packages: yfinance
Successfully installed yfinance-0.1.74
(base) C:\Users\user>
```



```
# -*- coding: utf-8 -*-
   import yfinance as yf
    import pandas as pd
    import time, datetime
5
6
    save path = "YF股價資料/"
    def main( stockNo, period ):
        # Yahoo Finance 的 代號為台灣的代號 + .TW
8
        stock id = str( stockNo ) + ".TW"
9
        data = yf.Ticker( stock_id ) # 指定K線資料
10
        df = data.history( period = period ) # 下載近 N 個月的資料
```





```
# -*- coding: utf-8 -*-
    import yfinance as yf
    import pandas as pd
                              引用模組
    import time, datetime
                                                 向函數傳入股票代號與
 5
                                                    期間抓取資料
                               儲存資料的路徑
    save_path = "YF股價資
6
    def main( stockNo, period ):
        # Yahoo Finance 的 代號為台灣的代號 + .TW
8
        stock id = str( stockNo ) + ".TW"
9
        data = yf.Ticker( stock_id ) # 指定K線資料
10
        df = data.history( period = period ) # 下載近 N 個月的資料
11
```





```
13
       df = df.round(2) # 四捨五入到小數第二位
       df.reset index(inplace=True) # 將index的日期資料變成欄位,以便處理
14
15
16
       # 重新命名欄位
        rename dict = {
17
           'Open': '開盤價',
18
           'High': '最高價',
19
           'Low': '最低價',
20
           'Close': '收盤價',
21
           'Volume': '成交股數',
22
23
           'Date': '日期'
24
       df.rename(columns = rename dict, inplace = True)
25
26
       # 將日期轉換為 datetime 格式
27
       df["日期"] = pd.to_datetime(df["日期"], "%Y-%m-%d")
28
29
       df.to_excel(f"{save_path}{stockNo}.xlsx", index = False) # 儲存檔案
30
```





整理資料

儲存檔案

```
df = df.round(2) # 四捨五入到小數第二位
13
       df.reset index(inplace=True) # 將index的日期資料變成欄位,以便處理
14
15
         重新命名欄位
16
       rename dict = {
17
           'Open': '開盤價',
18
           'High': '最高價',
19
                                                       更動欄位名稱
           'Low': '最低價',
20
           'Close': '收盤價',
21
           'Volume': '成交股數',
22
           'Date': '日期'
23
24
       df.rename(columns = rename dict, inplace = True)
25
26
                                                         將日期欄位轉換為
         將日期轉換為 datetime 格式
27
                                                           datetime格式
       df["日期"] = pd.to_datetime(df["日期"], "%Y-%m-%d")
28
29
       df.to_excel(f"{save_path}{stockNo}.xlsx", index = False) # 儲存檔案
30
```

國立高雄科技大學 AI 金融科技中心版權所有



```
# 0050成分股列表

stockNo_list = ["1101" ,"1216" ,"1301" ,"1303" ,"1326" ,"2002" ,"2207"

for stockNo in stockNo_list:
    # 1mo = 1個月,max 可以把所有期間的資料都下載
    main( stockNo, period = "100mo") # 抓取近 100 個月的資料

print(stockNo, datetime.datetime.now())
    time.sleep(3) # 每次抓取休息 3 秒
```



建構股票代號列表

```
# 0050成分股列表

stockNo_list = ["1101" ,"1216" ,"1301" ,"1303" ,"1326" ,"2002" ,"2207"

for stockNo in stockNo_list:
    # 1mo = 1個月,max 可以把所有期間的資料都下載
    main( stockNo, period = "100mo") # 抓取近 100 個月的資料

print(stockNo, datetime.datetime.now())

time.sleep(3) # 每次抓取休息 3 秒
```

走訪所有股票代號 抓取近 100 個月的資料







日期	開盤價	最高價	最低價	收盤價	成交股數	Dividendstock	Split
2014-04-21 00:00:00	22.45	22.59	22.32	22.55	3148614	0	0
2014-04-22 00:00:00	22.55	22.55	22.32	22.39	5043765	0	0
2014-04-23 00:00:00	22.41	22.71	22.27	22.29	8889806	0	0
2014-04-24 00:00:00	22.27	22.27	22.09	22.11	7571086	0	0
2014-04-25 00:00:00	22.04	22.11	21.33	21.35	12798385	0	0
2014-04-28 00:00:00	21.35	21.95	21.28	21.86	13986592	0	0
2014-04-29 00:00:00	21.86	22.09	21.7	21.97	8854459	0	0
2014-04-30 00:00:00	21.97	22.13	21.95	22	8071384	0	0
2014-05-02 00:00:00	22	22.11	21.7	21.79	7221693	0	0
2014-05-05 00:00:00	21.79	21.86	21.49	21.56	6601758	0	0
2014-05-06 00:00:00	21.51	21.61	21.42	21.49	8099933	0	0
2014-05-07 00:00:00	21.4	21.42	21.12	21.12	16072074	0	0
2014-05-08 00:00:00	21.17	21.22	20.89	20.99	16992459	0	0
2014-05-09 00:00:00	20.89	20.99	20.55	20.8	18013448	0	0
2014-05-12 00:00:00	20.62	20.66	20.2	20.3	18561328	0	0
2014-05-13 00:00:00	20.3	20.46	20.14	20.2	26519874	0	0
2014-05-14 00:00:00	20.25	20.37	20.16	20.34	24433033	0	0
2014-05-15 00:00:00	20.34	20.85	20.27	20.8	14168766	0	0

名稱	修改日期	類型	大小
1101.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	69 KE
1216.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	69 KE
1301.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	69 KE
1303.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	68 KE
1326.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	69 KE
2002.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	67 KE
2207.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	69 KE
2227.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	65 KE
2303.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	69 KE
2308.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	68 KE
2317.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	71 KE
2327.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	72 KE
2330.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	70 KE
2352.xlsx	2022/8/18 下午 05:25	Microsoft Excel 工作	67 KE
2357.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	68 KE
2379.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	70 KE
2382.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	69 KE
2395.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	69 KE
2408.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	71 KE
2409.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	69 KE
2412.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	64 KE
2454.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	71 KE
2603.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	70 KE
2609.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	69 KE
2610.xlsx	2022/8/18 下午 05:26	Microsoft Excel 工作	69 KE

回家作業



- 請抓取 0050 中 50 檔成分股, 2022年7月的股價資料
- 完成後請截圖展示部分檔案, 範例如下:

名稱	修改日期	類型	大小
1101_201501.xlsx	2022/8/17 下午 06:17	Microsoft Excel 工作	7 KB
1101_201502.xlsx	2022/8/17 下午 06:22	Microsoft Excel 工作	7 KB
1101_201503.xlsx	2022/8/17 下午 06:26	Microsoft Excel 工作	7 KB
1101_201504.xlsx	2022/8/17 下午 06:30	Microsoft Excel 工作	7 KB
1101_201505.xlsx	2022/8/17 下午 06:35	Microsoft Excel 工作	7 KB
1101_201506.xlsx	2022/8/17 下午 06:39	Microsoft Excel 工作	7 KB
1101_201507.xlsx	2022/8/17 下午 06:43	Microsoft Excel 工作	7 KB
1101_201508.xlsx	2022/8/17 下午 06:48	Microsoft Excel 工作	7 KB
1101_201509.xlsx	2022/8/17 下午 06:52	Microsoft Excel 工作	7 KB
1101_201510.xlsx	2022/8/17 下午 06:56	Microsoft Excel 工作	7 KB
1101_201511.xlsx	2022/8/17 下午 07:01	Microsoft Excel 工作	7 KB
1101_201512.xlsx	2022/8/17 下午 07:05	Microsoft Excel 工作	7 KB
1101_201601.xlsx	2022/8/17 下午 07:09	Microsoft Excel 工作	7 KB
1101_201602.xlsx	2022/8/17 下午 07:14	Microsoft Excel 工作	7 KB
1101_201603.xlsx	2022/8/17 下午 07:18	Microsoft Excel 工作	7 KB
1101_201604.xlsx	2022/8/17 下午 07:22	Microsoft Excel 工作	7 KB
1101_201605.xlsx	2022/8/17 下午 07:26	Microsoft Excel 工作	7 KB
1101_201606.xlsx	2022/8/17 下午 07:31	Microsoft Excel 工作	7 KB
1101_201607.xlsx	2022/8/17 下午 07:35	Microsoft Excel 工作	7 KB
1101_201608.xlsx	2022/8/17 下午 07:39	Microsoft Excel 工作	7 KB
1101_201609.xlsx	2022/8/17 下午 07:44	Microsoft Excel 工作	7 KB
1101_201610.xlsx	2022/8/17 下午 07:48	Microsoft Excel 工作	7 KB 🖠

ACCITION TO THE TOTAL TOTAL TO THE TOTAL TOTAL TO THE TOTAL TOTAL TO THE TOTAL TOTAL



章節到此結束,有任何問題歡迎提出來討論!

