

# 網路資料擷取分析

授課老師：邱淑怡

Date:12/11/2023

# 大綱

- ▶ Use yfinance and mplfinance
- ▶ Use pandas read\_html()

## 股價：yfinance

- ▶ yfinance 的資料來源是 Yahoo Finance API
  - ▶ !pip install yfinance
  - ▶ !pip install mplfinance

# Download one ticker from yfinance

```
import yfinance as yf

# for 台股
df = yf.Ticker("2330.TW").history(period="max")
df11 = yf.Ticker("2303.TW").history(period='max')
df12 = yf.Ticker("1605.TW").history(start='2022-01-01',end='2022-09-23')
# for 美股
df111 = yf.Ticker("MSFT")
display(df12)
df112=yf.Ticker("goog")
```

# download- 日期模式

```
import yfinance as yf
df22=yf.download('TSM TSLA',start='2016-01-01',end='2021-01-01')
df33= yf.download(('TSLA')    # 下載特斯拉 TSLA 全部歷史價量資料
```

download() 參數	說明
symbol	股票代號 (字串), 美股例如 'AMD' (超微), 台股後面要加 '.tw', 例如 '0050.tw'
start	起始日期 YYYY-MM-DD (字串), 例如 '2022-08-22'
end	結束日期 YYYY-MM-DD (字串), 例如 '2022-09-06', 注意, 不包含此日資料
period	期間, 可用 d (日), mo(月), y(年), ytd, max(全部), 例如 5d (5 天), 3mo(近三個月)
interval	頻率, 可用 m(分), h(小時), d(日), wk(周), mo(月), 例如 1m(一分線)

ytd:今年以來, 即自年初第一個交易日至今

## download- 日期模式

語法是 `yf.download`

# 股票代號,

# `period`=日期範圍 (1d,5d,1mo,3mo,6mo,1y,2y,5y,10y,ytd,max)

# `interval`=頻率 (1m,2m,5m,15m,30m,60m,90m,1h,1d,5d,1wk,1mo,3mo)

```
yf.download('TSM TSLA',period='6mo',interval='1mo')
```

## download-Period 模式 例子

```
yf.download('TSM TSLA',period='7d',interval='1m')
```

#最近 7 天的日內 ( Intraday ) 數據。透過指定頻率 ( interval ) 為  
1m/5m/10m/90m 等 ( m 代表分鐘 ) ，可以提取到高頻率的股票價格數據

# use mplfinance

```
import mplfinance as mpf
```

```
start = "2022-1-1"
```

```
end = '2022-12-22' # to 2022-12-21
```

```
df3 = yf.download('1605.tw',start,end) # 0050
```

```
#繪製K線圖:candle 也就是我們常講的K線，平均移動線:mav繪製5、20日MA
```

```
mpf.plot(df3, type='candle', mav=(5,20), volume=True, title='1605.TW', savefig='1605_plot.png')
```



# Pandas

- ▶ `read_html()`: 只有傳入網址，就能夠讀取網頁中的<table>表格標籤，且回傳一個dataframe的物件，<tr>標籤就是dataframe資料結構的列(row)，<td>標籤則是欄(column)

# 台灣各城市天氣預報資料截取

```
url = "https://www.cwb.gov.tw/V8/C/W/County/MOD/wf7dayNC_NCSEI/ALL_Week.html?v="
tables = pd.read_html(url, encoding="utf8")
display(tables[0])
```

# 台灣銀行匯率

```
url = "https://rate.bot.com.tw/xrt?Lang=zh-TW"
tables = pd.read_html(url)
df = tables[0]

#選取想要的欄位
df = df[[df.columns[0], df.columns[2]]]
df.columns = ["幣別", "本行賣出"]
display(df)

#處理欄位內容
df["幣別"] = df["幣別"].apply(lambda x: x.split("(")[0].strip())
df.set_index("幣別", inplace=True) #將幣別設定為 Index
print(df.loc["港幣"])
```

# 公開資訊觀測站的資料擷取

```
import pandas as pd
url = 'https://mops.twse.com.tw/server-java/t13sa150_otc?&step=wh'
df_twse = pd.read_html(url,encoding="big5-hkscs", header=1)[0]
display(df_twse)
```

# 今天任務

- ▶ 請擷取 財經資料：  
<http://vip.stock.finance.sina.com.cn/q/go.php/vComStockHold/kind/jjzc/index.phtml> and  
<http://vip.stock.finance.sina.com.cn/q/go.php/vComStockHold/kind/jjzc/index.phtml?p=2>
- ▶ 將兩個dataframe 合併為一個dataframe
  - ▶ Hint use `pd.concat()` function
  - ▶ 並重新建立新的row index