



第3章

認識資料科學神器 pandas 並用網路爬蟲取得資料

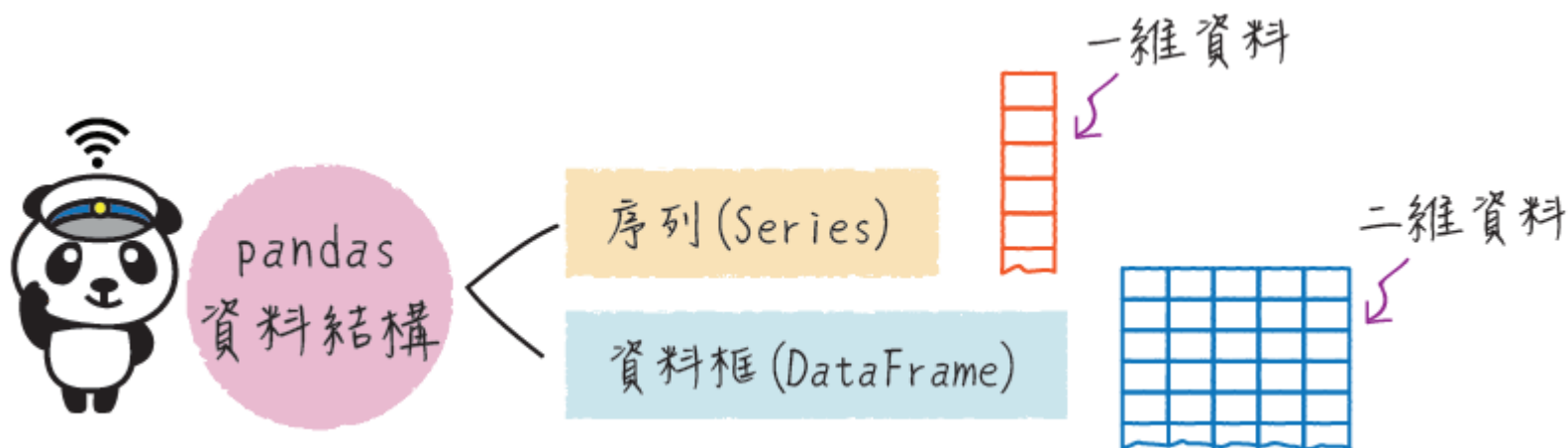
說到資料科學，要先有「資料」，才能衍生出相關的「科學」，因此，如何獲得資料就是首要的步驟。除了靠自己搜集與整理之外，更可以透過直接下載或網路爬蟲取得現成且大量的各類資料。接下來本章將介紹如何以 Python 程式碼來建立、取得及儲存資料。



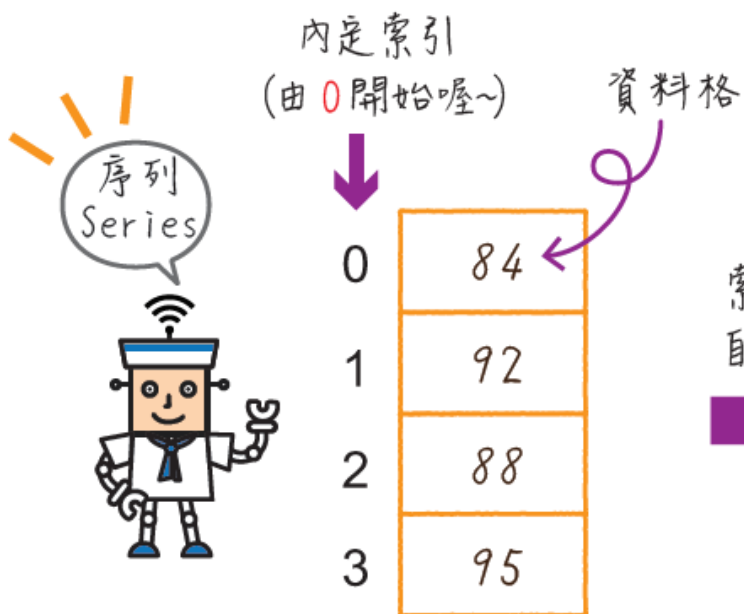
3-1

認識 pandas — 從資料結構看起

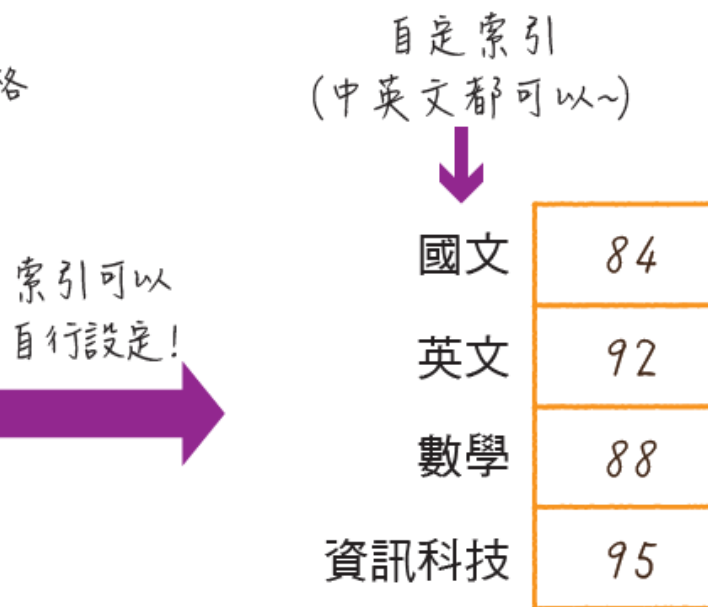
- **pandas**是Python著名的資料處理和分析套件，功用就像Python版的Excel試算表。
- pandas套件提供兩種常用的資料結構如下：



- **序列 (Series)**：一維資料，類似一維串列 (list)，每個元素可以使用**內定索引 (Index)**，也可以改成**自定索引**進行存取。



(a) 內定索引：0,1,2,3

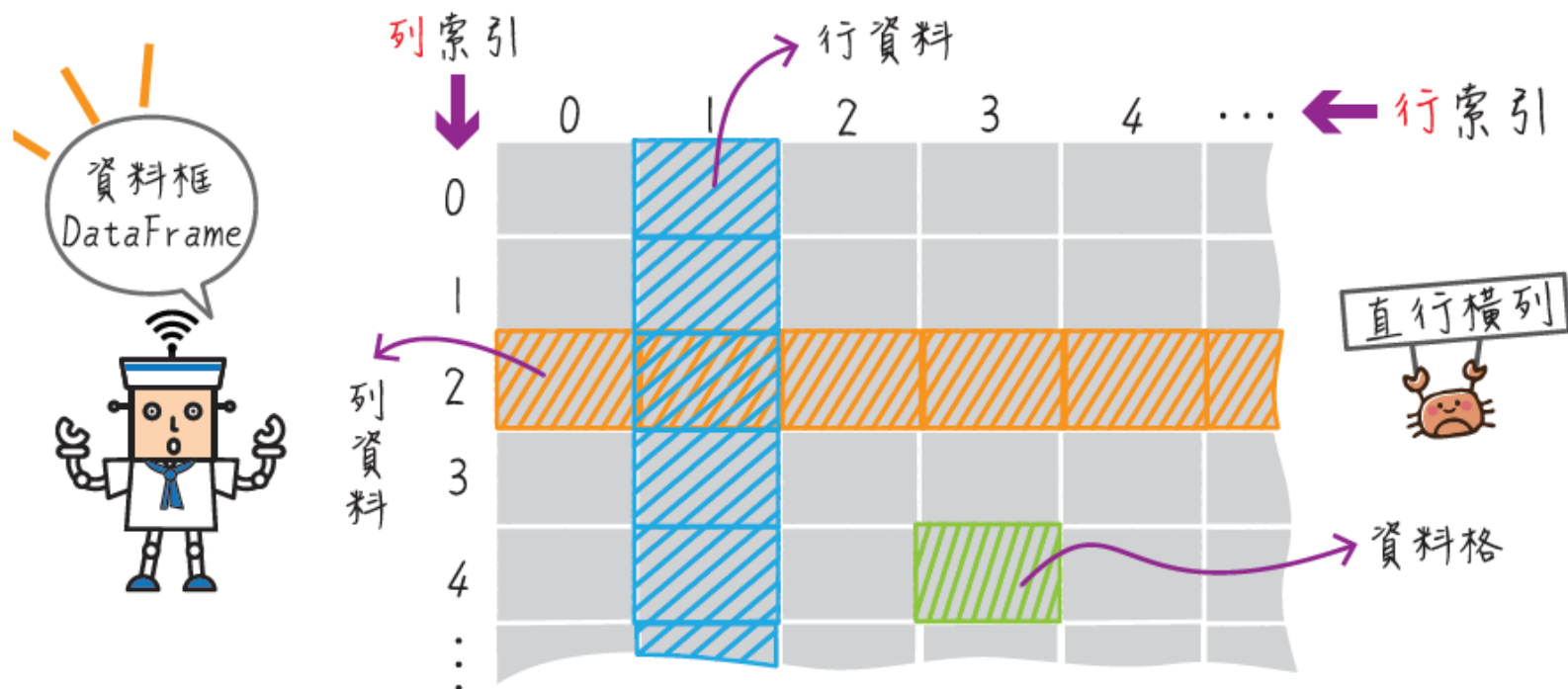


(b) 自定索引：國文,英文,數學,資訊科技



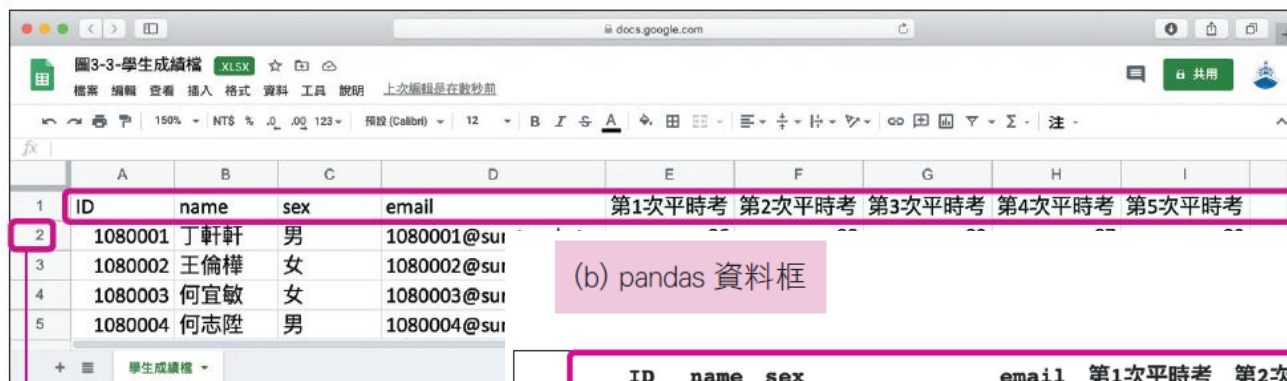
- **資料框 (DataFrame)**：二維資料，類似MS Excel、Google試算表的「**資料表**」或一般的「**表格**」。

★ 內定索引：由 0 開始的數字，可自定改為中英文自訂索引



• 試算表與pandas 資料框的對照：

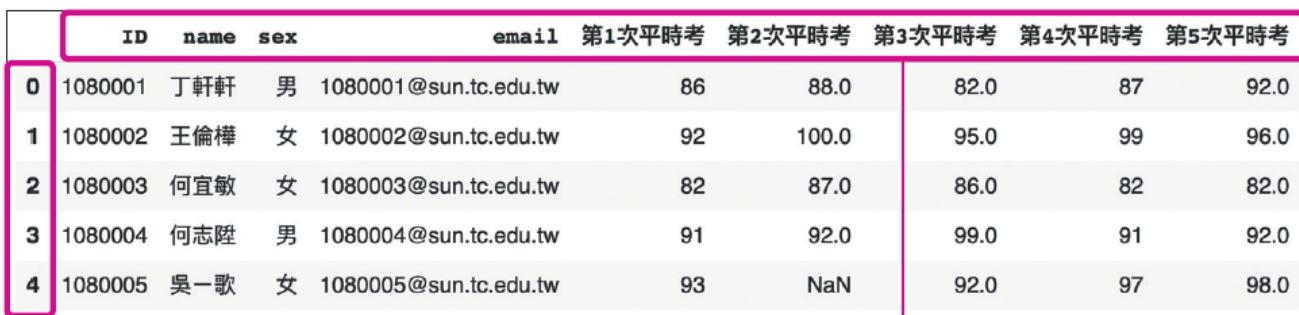
(a) Google 試算表



	A	B	C	D	E	F	G	H	I
1	ID	name	sex	email	第1次平時考	第2次平時考	第3次平時考	第4次平時考	第5次平時考
2	1080001	丁軒軒	男	1080001@sun.tc.edu.tw					
3	1080002	王倫樺	女	1080002@sun.tc.edu.tw					
4	1080003	何宜敏	女	1080003@sun.tc.edu.tw					
5	1080004	何志陞	男	1080004@sun.tc.edu.tw					

試算表的記錄從第 2 列開始

(b) pandas 資料框



	ID	name	sex	email	第1次平時考	第2次平時考	第3次平時考	第4次平時考	第5次平時考
0	1080001	丁軒軒	男	1080001@sun.tc.edu.tw	86	88.0	82.0	87	92.0
1	1080002	王倫樺	女	1080002@sun.tc.edu.tw	92	100.0	95.0	99	96.0
2	1080003	何宜敏	女	1080003@sun.tc.edu.tw	82	87.0	86.0	82	82.0
3	1080004	何志陞	男	1080004@sun.tc.edu.tw	91	92.0	99.0	91	92.0
4	1080005	吳一歌	女	1080005@sun.tc.edu.tw	93	NaN	92.0	97	98.0

pandas 將每筆記錄自動
加上列索引 (從 0 開始)

試算表第一列欄位名稱是
pandas 的行索引



3-1-1 建立與存取序列 (Series)



實作

利用 Python 一維串列 (list) 存放資料

EX3-1.1a.ipynb

01

首先利用 Python 「串列」來存放科目名稱 (c) 及分數 (s) 這兩項資料。



```
1 c = ['國文', '英文', '數學', '資訊科技']  
2 s = [84, 92, 88, 95]
```

02

將 c、s 串列印出來看看。



```
1 print(c)  
2 print(s)
```

```
['國文', '英文', '數學', '資訊科技']  
[84, 92, 88, 95]
```

這兩個串列可進一步
用來建立序列 (Series)



資料科學 × 機器學習

實戰探索

Practical Exploration





實作

利用 pandas 套件建立序列 (Series)

EX3-1.1b.ipynb

建立序列
(內定索引)

序列名稱 = `pd.Series(串列)`

傳入串列

匯入 `pandas` 套件時，設定別名為 `pd`，
之後程式中 `pandas` 就可用 `pd` 代替！

`s` 須大寫，否則會產生錯誤

建立方式1. 直接輸入串列內容：

```
ss = pd.Series([84, 92, 88, 95])
```

建立方式2. 使用串列名稱：

```
s = [84, 92, 88, 95]  
ss = pd.Series(s)
```

內定索引
(由 0 開始)

取出序列值

↓		↓
0	84	ss[0]
1	92	ss[1]
2	88	ss[2]
3	95	ss[3]



資料科學 × 機器學習

實戰探索

Practical Exploration





利用 pandas 套件建立序列 (Series)

EX3-1.1b.ipynb

01

匯入 pandas 套件，並設定別名為「pd」。



```
1 import pandas as pd
```

02

建立二個串列 (c、s)，存放科目名稱及分數這兩項資料，準備用來放入序列中。



```
1 c = ['國文', '英文', '數學', '資訊科技']  
2 s = [84, 92, 88, 95]
```





利用 pandas 套件建立序列 (Series)

EX3-1.1b.ipynb

03

呼叫 `pd.Series()` 函式建立兩個序列 (cs、ss)，並將其內資料設定為序列的內容。完成後把兩個序列印出來檢視。

用串列 `c` 建立 `cs` 序列



```
1 cs = pd.Series(c)
2 ss = pd.Series(s)
3 print(cs)
4 print(ss)
```

用串列 `s` 建立 `ss` 序列

```
0  國文
1  英文
2  數學
3  資訊科技
dtype: object
0   84
1   92
2   88
3   95
dtype: int64
```

cs 序列

ss 序列





利用 pandas 套件建立序列 (Series)

EX3-1.1b.ipynb

04

接下來印出序列某一筆元素的內容。



```
1 print(cs[0])  
2 print(ss[0])
```

利用內定索引取出序列的值



國文

84





建立自定索引的序列 (Series)

EX3-1.1c.ipynb

建立序列
(自定索引)

序列名稱 = `pd.Series(串列, index=自定索引串列)`

自定索引



國文	84	→ <code>sc['國文']</code>
英文	92	→ <code>sc['英文']</code>
數學	88	→ <code>sc['數學']</code>
資訊科技	95	→ <code>sc['資訊科技']</code>

取出序列值



```
c = ['國文', '英文', '數學', '資訊科技']  
s = [84, 92, 88, 95]  
sc = pd.Series(s, index=c)
```



```
sc = pd.Series([84, 92, 88, 95], index=['國文', '英文', '數學', '資訊科技'])
```





建立自定索引的序列 (Series)

EX3-1.1c.ipynb

01

匯入 pandas 套件，並將科目及分數分別放到兩個串列 (c、s) 中。



```
1 import pandas as pd  
2 c = ['國文', '英文', '數學', '資訊科技']  
3 s = [84, 92, 88, 95]
```





建立自定索引的序列 (Series)

EX3-1.1c.ipynb

02

建立一個序列，將科目串列 (c) 當成序列的「自定索引」、分數串列 (s) 當成序列的元素，完成後印出建立的序列。



```
1 sc = pd.Series(s, index=c)  
2 print(sc)
```



```
國文      84  
英文      92  
數學      88  
資訊科技  95  
dtype: int64
```





建立自定索引的序列 (Series)

EX3-1.1c.ipynb

03

分別試試利用「內定索引」和「自定索引」兩種方式來存取及設定序列元素。

```
1 #使用內定索引
2 print(sc[1])
3 sc[1] = 100
4 print(sc[1])
5
6 #使用自定索引
7 print(sc['英文'])
8 sc['英文'] = 80
9 print(sc['英文'])
10
```

可以用字串 '英文' 當索引！

取出並設定「內定索引 1」(即英文科目)的分數

```
☐→ 92
     100
     100
     80
```

利用自定索引('英文'字串)取出並設定英文分數





序列(Series)存取

存取多個
序列元素

內定索引
(由 0 開始)



0	84
1	92
2	88
3	95

ss[0]
ss[1]
ss[2]
ss[3]

連續的元素，序列參數寫成範圍，如 0:3

ss[0:3]

包含 0, 1, 2 三個元素

中括號要用 2 個

ss[[1, 3]]

不連續的元素，序列參數寫成串列，如 [1, 3]

自定索引



國文	84
英文	92
數學	88
資訊科技	95

sc['國文']
sc['英文']
sc['數學']
sc['資訊科技']

sc['國文': '數學']

sc[['英文', '資訊科技']]

3-1-2 建立與存取資料框



實作

利用 Python 建立二維串列

EX3-1.2a.ipynb

一維串列

索引

	0	1	2	3
s	國文	英文	數學	資訊科技

串列內容

```
s = ['國文', '英文', '數學', '資訊科技']
```

```
len(s) → 4
```

s串列的元素共4個

len()



二維串列

		0	1	2	3
sc[0]	0	國文	英文	數學	資訊科技
sc[1]	1	84	92	88	95

```
sc = [['國文', '英文', '數學', '資訊科技'], [84, 92, 88, 95]]
```

```
len(sc) → 2
```

sc包含了2個一維串列

```
len(sc[0]) → 4
```

sc[0]是一維串列，串列的元素共4個



資料科學 × 機器學習

實戰探索

Practical Exploration





利用 Python 建立二維串列

EX3-1.2a.ipynb

01

建立一個二維串列，用來存放科目和分數，完成後印出串列內容。

```
[ ] 1 sc = [['國文', '英文', '數學', '資訊科技'], [84, 92, 88, 95]]  
    2 print(sc)
```

```
 [['國文', '英文', '數學', '資訊科技'], [84, 92, 88, 95]]
```





02

印出二維串列內的串列個數，可以使用 `len()` 函式來處理。



```
1 print(len(sc))
```



2

sc 這個二維串列包含了 2 個一維串列





利用 Python 建立二維串列

EX3-1.2a.ipynb

03

試著用逐一走訪的方式來顯示每一筆元素的內容（註：「一一讀取出來」的動作就稱為**走訪**）。



```
1 for i in range(len(sc[0])):  
2     print(sc[0][i],sc[1][i])
```



```
國文 84  
英文 92  
數學 88  
資訊科技 95
```





實作

呼叫 pandas 的 DataFrame() 函式建立資料框

EX3-1.2b.ipynb

建立資料框
DataFrame

資料框名稱 = `pd.DataFrame()`

```
c = ['國文', '英文', '數學', '資訊科技']  
s = [84, 92, 88, 95]  
df = pd.DataFrame()  
df['科目'] = c  
df['分數'] = s
```

行索引

列索引

	科目	分數
0	國文	84
1	英文	92
2	數學	88
3	資訊科技	95

df['科目'][1]

df['分數'][3]

資料框索引值：
先行(直)後列(橫)



資料科學 × 機器學習

實戰探索

Practical Exploration





實作

呼叫 pandas 的 DataFrame() 函式建立資料框

EX3-1.2b.ipynb

01

匯入 pandas 套件，分別建立科目 (c) 和分數 (s) 二個串列。



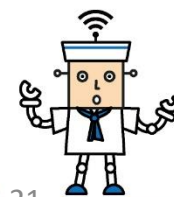
```
1 import pandas as pd
2 c = ['國文', '英文', '數學', '資訊科技']
3 s = [84, 92, 88, 95]
```



資料科學 × 機器學習

實戰探索

Practical Exploration





實作

呼叫 pandas 的 DataFrame() 函式建立資料框

EX3-1.2b.ipynb

02

呼叫 pandas 的 DataFrame() 函式建立空白的 df 資料框。



```
1 df = pd.DataFrame()  
2 print(df)
```



```
Empty DataFrame  
Columns: []  
Index: []
```





實作

呼叫 pandas 的 DataFrame() 函式建立資料框

EX3-1.2b.ipynb

03

將 c 串列加入到 df 資料框中，並給予自定行索引為「科目」；再將 s 串列加入到 df 資料框，並給予自定行索引為「分數」。



```
1 df['科目'] = c  
2 df['分數'] = s
```



資料科學 × 機器學習

實戰探索

Practical Exploration





實作

呼叫 pandas 的 DataFrame() 函式建立資料框

EX3-1.2b.ipynb

04

將 df 印出來檢視其內容。



1 df



	科目	分數
0	國文	84
1	英文	92
2	數學	88
3	資訊科技	95



資料科學 × 機器學習

實戰探索

Practical Exploration





設定資料框
列索引

資料框名稱.index = 串列

```
df.index = ['第1科', '第2科', '第3科', '第4科']
```

或

```
r = ['第1科', '第2科', '第3科', '第4科']  
df.index = r
```

自定列索引

第1科
第2科
第3科
第4科

科目	分數
國文	84
英文	92
數學	88
資訊科技	95

行索引





自定資料框的列索引

EX3-1.2c.ipynb

01

使用 `df.index` 設定列索引，完成後印出 df 資料框。

```
1 import pandas as pd
2 c = ['國文', '英文', '數學', '資訊科技']
3 s = [84, 92, 88, 95]
4 df = pd.DataFrame()
5 df['科目'] = c
6 df['分數'] = s
7 df.index = ['第1科', '第2科', '第3科', '第4科']
8 df
```

自訂列索引

	科目	分數
第1科	國文	84
第2科	英文	92
第3科	數學	88
第4科	資訊科技	95



資料科學 × 機器學習

實戰探索

Practical Exploration





DataFrame() 參數設定

資料框
設定
行列索引

資料框名稱 = `pd.DataFrame(二維串列, index = 列索引串列, columns = 行索引串列)`

columns=行索引串列

index=列索引串列

	國文	英文	數學	社會	自然
丁軒軒	86	88	82	87	92
王倫華	92	100	95	99	96
何宜敏	82	87	86	82	82
陳志昇	91	92	99	91	92

二維串列內容

3-2

資料取得

3-2-1 讀取資料檔

▼ pandas 用來讀取常見資料檔的函式

函式	說明
<code>read_csv</code> (檔名)	讀取 csv 格式的檔案
<code>read_json</code> (檔名)	讀取 json 格式的檔案
<code>read_html</code> (檔名)	讀取 html 格式的檔案
<code>read_excel</code> (檔名)	讀取 excel 格式的檔案





掛接 Google 雲端硬碟

EX3-2.1a.ipynb

mount()
函式

掛接Google
雲端硬碟

插入USB 隨身碟，掛到電腦



1. `mount()` 是 `drive` 模組下的一個函式，而 `drive` 隸屬於 `google.colab` 套件，所以使用 `mount()` 函式前必須先匯入 `google.colab` 套件。

2. 要掛接上 Google 雲端硬碟，需要在 `mount()` 使用「`/content/MyGoogleDrive`」參數。

3. 掛上個人 Google 雲端硬碟後，根資料夾是位於虛擬機的「`/content/MyGoogleDrive/My Drive`」資料夾內。



資料科學 × 機器學習

實戰探索

Practical Exploration





01

呼叫 `mount()` 函式掛接 Google 雲端硬碟，成為**虛擬磁碟機**。



```
1 from google.colab import drive  
2 drive.mount('/content/MyGoogleDrive')
```





02

使用個人 Google 雲端硬碟時需要通過帳號、密碼的**認證程序**，執行程式時會有幾個認證的交談視窗。先以滑鼠點選連結。

```
+ 程式碼 + 文字
```

```
1 from google.colab import drive
2 drive.mount('/content/MyGoogleDrive')
```

... Go to this URL in a browser: https://accounts.google.com/o/oauth2/auth?client_id=9473

Enter your authorization code:

點選此連結





03

選擇使用 Google 雲端硬碟的使用者帳戶。





04

允許存取 Google 帳戶。



「Google Drive File Stream」 想要存取您的 Google 帳戶

titanic2020.wang@gmail.com

這麼做將允許「Google Drive File Stream」進行以下操作：

-  查看、編輯、建立及刪除您的所有 Google 雲端硬碟檔案 ⓘ
-  查看 Google 相簿中的相片、影片和相簿 ⓘ
-  查看 Google 使用者資訊，例如個人資料和聯絡人 ⓘ
-  查看、編輯、建立及刪除您的任何 Google 雲端硬碟文件 ⓘ

確認「Google Drive File Stream」是您信任的應用程式

這麼做可能會將您的機密資訊提供給這個網站或應用程式。想瞭解「Google Drive File Stream」會如何處理您的資料，請參閱該用戶端的《服務條款》和《隱私權政策》。您隨時可以前往 [Google 帳戶](#) 頁面查看或移除存取權。

[瞭解潛在風險](#)

[取消](#) [允許](#)





05

複製授權碼。





06

將複製的授權碼貼至 02 的授權碼空白文字方塊後，按 **Enter** 鍵。



```
1 from google.colab import drive  
2 drive.mount('/content/MyGoogleDrive')
```

... Go to this URL in a browser: <https://accounts.google.com/o/oauth2/auth?client>

Enter your authorization code:



將授權碼貼至此空白方塊內





07

完成掛接 Google 雲端硬碟。

```
1 from google.colab import drive  
2 drive.mount('/content/MyGoogleDrive')
```

Mounted at /content/MyGoogleDrive

已完成掛接至雲端硬碟





用 pandas 讀取 CSV 檔及印出資料

EX3-2.1b.ipynb

read_csv()
函式

讀取csv檔案到程式中

資料框名稱=pd.read_csv(路徑+檔案名稱,
[encoding = 'utf-8'])

1. 若要讀取的csv檔和程式檔在同一資料夾，則路徑名稱可省略。

2. **encoding**是csv檔的**字元編碼**，預設為**utf-8**，可省略。

如果csv檔是其他編碼(例如：Windows應用軟體常用的big5編碼)，則需特別指定相對應的編碼(例如：**encoding = 'big5'**)才能順利開啟。



```
df = pd.read_csv('/content/MyGoogleDrive/My Drive/Ch03/Iris.csv')
```

路徑名稱

檔案名稱

head()
函式

印出資料框前幾列數的內容

資料框名稱.head(列數)

列數若省略，
預設會印
出5列資料



資料科學×機器學習

實戰探索

Practical Exploration





用 pandas 讀取 CSV 檔及印出資料

EX3-2.1b.ipynb

01

匯入 pandas 套件並命名為 pd，使用 read_csv() 函式讀取 csv 資料檔，再設定給 df 變數。df 會是資料框的資料型別。



```
1 from google.colab import drive
2 drive.mount('/content/MyGoogleDrive')
3 import pandas as pd
4 df = pd.read_csv('/content/MyGoogleDrive/My Drive/Python-for-Titanic/Ch03/Iris.csv')
```

請填實際的資料夾名稱及檔案路徑，這裡是讀取本章範例內的 Iris.csv 資料檔

接下行

↳ Drive already mounted at /content/MyGoogleDrive; to attempt to forcibly remount, call drive.mount("/content/MyGoogleDrive")





02

呼叫 head() 函式印出前 5 列的資料。



```
1 df.head()
```

head() 內可指定幾筆，省略的話，內定是 5



	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa



3-2-2 儲存 (匯出) 資料檔

- CSV 格式每個欄位之間會以逗號隔開，每筆資料之間則以換行來分隔。

▼ pandas 用來儲存常見資料檔的函式

函式	說明
<code>to_csv</code> (檔名)	儲存成 csv 格式的檔案
<code>to_json</code> (檔名)	儲存成 json 格式的檔案
<code>to_html</code> (檔名)	儲存成 html 格式的檔案
<code>to_excel</code> (檔名)	儲存成 excel 格式的檔案





將資料框儲存成 CSV 檔

EX3-2.2.ipynb

資料框 (DataFrame)

	國文	英文	數學	社會	自然
丁軒軒	86	88	82	87	92
王倫樺	92	100	95	99	96
何宜敏	82	87	86	82	82
何志陞	91	92	99	91	92

to_csv() 函式



Google 雲端硬碟





將資料框儲存成 CSV 檔

EX3-2.2.ipynb

01

掛上 Google 雲端硬碟，程式執行時也會如前述出現 Google 帳密的認證步驟。

```
1 from google.colab import drive  
2 drive.mount('/content/MyGoogleDrive')
```

↳ Drive already mounted at /content/MyGoogleDrive; to attempt to forc





02

匯入 pandas 套件，建立列索引、行索引的串列，以及各科分數的二維串列，呼叫 DataFrame() 函式建立 df 資料框。

```
1 import pandas as pd
2 n = ['丁軒軒', '王倫樺', '何宜敏', '何志陞']
3 c = ['國文', '英文', '數學', '社會', '自然']
4 s = [[86, 88, 82, 87, 92], [92, 100, 95, 99, 96],
       [82, 87, 86, 82, 82], [91, 92, 99, 91, 92]]
5 df = pd.DataFrame(s, index=n, columns=c)
6 df
```

列索引串列

行索引串列

接下行

各科分數的二維串列

	國文	英文	數學	社會	自然
丁軒軒	86	88	82	87	92
王倫樺	92	100	95	99	96
何宜敏	82	87	86	82	82
何志陞	91	92	99	91	92





將資料框儲存成 CSV 檔

EX3-2.2.ipynb

03

指定存檔的路徑及檔名，呼叫 `to_csv()` 函式將資料框的內容存到 Google 雲端硬碟中。

```
1 o_filepath = '/content/MyGoogleDrive/My Drive/Colab 接下行  
Notebooks/資料檔/完成檔/Ch03/'  
2 GooglePath = o_filepath  
3 filename='五科成績.csv'  
4 df.to_csv(GooglePath + filename)
```

路徑

檔名



3-2-3 由網址讀取資料檔



由網址讀取資料檔

EX3-2.3.ipynb

01

匯入 pandas 套件並命名為 pd，呼叫 `pd.read_csv()` 函式由「PM2.5 日均值」檔案網址「https://data.epa.gov.tw/api/v1/aqx_p_322?limit=1000&api_key=9be7b239-557b-4c10-9775-78cadfc555e9&format=csv」讀取 CSV 資料檔，再指定給 df 變數。此時 df 就是資料框的資料結構。

```
1 import pandas as pd
2 df=pd.read_csv('https://data.epa.gov.tw/api/v1/ 接下行
                 aqx_p_322?limit=1000&api_key=9be7b239-557b-4c10-97
3 df.head()
```

	SiteId	SiteName	County	ItemId	ItemName	ItemEngName	ItemUnit	MonitorDate	Concentration
0	1	基隆	基隆市	33	細懸浮微粒	PM2.5	µg/m3	2021-04-05 00:00:00	13
1	2	汐止	新北市	33	細懸浮微粒	PM2.5	µg/m3	2021-04-05 00:00:00	14
2	3	萬里	新北市	33	細懸浮微粒	PM2.5	µg/m3	2021-04-05 00:00:00	16
3	4	新店	新北市	33	細懸浮微粒	PM2.5	µg/m3	2021-04-05 00:00:00	12
4	5	土城	新北市	33	細懸浮微粒	PM2.5	µg/m3	2021-04-05 00:00:00	11



資料科學 × 機器學習

實戰探索

Practical Exploration





02

參照 3-2-2 節的方法，將此 df 資料框儲存成 CSV 檔並上傳至個人的 Google 雲端硬碟。

```
1 o_filepath = '/content/MyGoogleDrive/My Drive/Colab Notebooks/資料檔/完成檔/Ch03/'  
2 GooglePath = o_filepath  
3 filename='PM25.csv'  
4 df.to_csv(GooglePath + filename)
```



3-2-4 網路爬蟲

- **網路爬蟲** (web crawler) 是一種用來自動瀏覽全球資訊網 (WWW) 的網路機器人，可以直接從 html 網頁擷取所需的資料。
- 可以藉由設定時間**自動取得資料**，並進一步儲存或做資料處理。





01

找到有表格的網頁，例如：台灣彩券大樂透各期中獎號碼網頁「<https://www.taiwanlottery.com.tw/Lotto/Lotto649/history.aspx>」。

[關於台灣彩券](#)
[威力彩](#)
[大樂透](#)
[今彩539](#)
[雙贏彩](#)
[BINGO BINGO賓果賓果](#)
[3星彩](#)
[4星彩](#)
[38樂合彩](#)
[49樂合彩](#)
[39樂合彩](#)
[「加開獎項」加碼活動專區](#)
[刮刮樂](#)
[新聞及公告](#)
[彩券資訊通](#)

各期獎號與開獎結果

其他遊戲查詢

請選擇欲查詢的期別或開獎日期
● 第 期的中獎號碼(輸入九碼期別，例：103000001期)
● ☐ 中華民國 年 月的中獎號碼

大樂透

期別	開獎日	兌獎截止(註6)	銷售金額	獎金總額				
109000091	109/10/16	110/01/18	140,227,550	78,527,428				
獎號				特別號				
開出順序	07	44	28	25	47	33	19	
大小順序	07	25	28	33	44	47	19	
獎金分配								
項目	頭獎	貳獎	參獎	肆獎	伍獎	陸獎	柒獎	普獎
對中獎號數	6個	任5個 +特別號	任5個	任4個 +特別號	任4個	任3個 +特別號	任2個 +特別號	任3個
中獎注數	0	1	37	98	2,401	3,420	33,625	44,355
單注獎金	0	2,542,372	73,998	17,960	2,000	1,000	400	400
累至次期獎金	32,073,010	0	0	0				

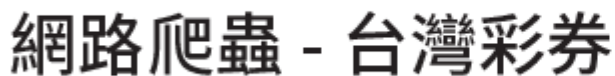
網頁中的部分表格
(某期的中獎號碼)

資料科學 × 機器學習

實戰探索

Practical Exploration

48



呼叫 pandas 的 read_html() 函式讀取網頁包含的所有表格。

```
1 import pandas as pd
2 url='https://www.taiwanlottery.com.tw/Lotto/Lotto649/history.aspx'
3 df = pd.read_html(url)
4 df
```

```
[
0  .style3 {color: #FF0000} .style5 {color: #FF00...,
0
0  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000091 109/10/...
1  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000090 109/10/...
2  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000089 109/10/...
3  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000088 109/10/...
4  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000087 109/10/...
5  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000086 109/09/...
6  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000085 109/09/...
7  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000084 109/09/...
8  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000083 109/09/...
9  期別 開獎日 兌獎截止(註6) 銷售金額 獎金總額 109000082 109/09/...
]
0      1      2      ...      7      8      9
0      期別      開獎日      開獎日      ...      獎金總額      獎金總額      獎金總額
1      109000091 109/10/16 109/10/16 ... 78527428 78527428 78527428
2      獎號      獎號      獎號      ...      獎號      特別號      特別號
3      開出順序      開出順序      07 ...      33      19      19
4      大小順序      大小順序      07 ...      47      19      19
5      獎金分配      獎金分配      獎金分配 ...      獎金分配      獎金分配      獎金分配
6      項目      頭獎      頭獎      ...      陸獎      柒獎      普獎
```

用 pandas 將各期中獎號碼讀入資料框內





03

網頁中如果包含了許多表格，使用 pandas 的 `read_html()` 函式會依讀取的次序將讀到的表格由 0,1.. 來編號，不過有些可能是我們不需要的，必須稍做檢視。

1 df[2]

	0	1	2	3	4	5	6	7	8	9
0	期別	開獎日	開獎日	兌獎截止(註6)	兌獎截止(註6)	銷售金額	銷售金額	獎金總額	獎金總額	獎金總額
1	109000091	109/10/16	109/10/16	110/01/18	110/01/18	140227550	140227550	78527428	78527428	78527428
2	獎號	獎號	獎號	獎號	獎號	獎號	獎號	獎號	特別號	特別號
3	開出順序	開出順序	07	44	28	25	47	33	19	19
4	大小順序	大小順序	07	25	28	33	44	47	19	19
5	獎金分配	獎金分配	獎金分配	獎金分配	獎金分配	獎金分配	獎金分配	獎金分配	獎金分配	獎金分配
6	項目	頭獎	頭獎	貳獎	參獎	肆獎	伍獎	陸獎	柒獎	普獎
7	對中獎號數	6個	6個	任5個 + 特別號	任5個	任4個 + 特別號	任4個	任3個 + 特別號	任2個 + 特別號	任3個
8	中獎注數	0	0	1	37	98	2401	3420	33625	44355
9	單注獎金	0	0	2542372	73998	17960	2000	1000	400	400
10	累至次期獎金	32073010	32073010	0	0	0	NaN	NaN	NaN	NaN

編號2這個資料框是我們需要的

