∽ Ч.]字典基本操作

還記得怎麼查國語字典嗎?以查「李」字為例,先由部首目錄找到部首「木」的位置, 剩下來的「子」筆畫為3畫,再於「木部首」3畫的地方就能找到「李」這個字。

Python 中「字典」資料型態與國語字典結構類似,其元素是以「鍵-值」對方式儲存, 運作方式為利用「鍵」來取得「值」。

4.1.1 建立字典

04

串列資料依序排列,若要取得串列內特定資料,必須知道其在串列中的位置,例如 一個水果價格的串列:

list1 = [20, 50, 30] #分別為香蕉、蘋果、橘子的價格

若要得知蘋果的價格,就要知道蘋果價格是串列第2個元素,再使用「list1[1]」取出 蘋果價格,是不是很不方便呢?

9 字典的結構也與串列類似,其元素是以「鍵-值」對方式儲存,這樣就可使用「鍵」
 來取得「值」。有多種方式可以建立字典,第一種方式為將元素置於一對大括號「{}」
 0 中,其語法為:

字典名稱 = {鍵1:值1, 鍵2:值2, ……}

字串、整數、浮點數等皆可做為「鍵」,但以字串做為「鍵」的情況最多。

2 例如將前述水果價格串列建立為字典型態:

dict1 = {"香蕉":20, "蘋果":50, "橘子":30}

建立字典的第二種方式是使用 dict 函式,再將鍵 - 值對置於中括號「[]」中,語法為:

```
字典名稱 = dict([[鍵1,值1], [鍵2,值2], ……])
```

例如:

dict2 = dict([["香蕉",20], ["蘋果",50], ["橘子",30]])

建立字典的第三種方式是使用 dict 函式,只要將鍵值以等號連接起來即可,語法為:

www.gotop.com.tw

字典名稱 = dict(鍵1=值1, 鍵2=值2, ……)

例如:

dict3 = dict(香蕉=20, 蘋果=50, 橘子=30)

第三種建立字典的方式相當簡潔,但特別注意此種方式建立的字典「鍵」不能使用 數值,否則執行時會產生錯誤,例如:

dict1 = dict(1="林大明", 3="李美麗", 5="陳品言") print(dict1[3]) # 錯誤,字典取值的說明參考下一小節

4.1.2 字典取值

可以將字典想像成一個箱子,箱子中許多盒子,每個盒子都貼上標籤,標籤寫著盒子的名稱(鍵),盒子內則裝著指定的物品(值),與串列最大的不同在於串列元素在記憶體中是依序排列,而字典元素則是隨意放置,沒有一定順序。



基本取值方式

既然字典元素没有一定順序,那要如何取得字典元素值呢?其實很簡單,只要依據 標籤 (鍵) 找到存放物品的盒子,就能取得盒子內的物品 (值)。

取得字典元素值的方法是以「鍵」做為索引來取得「值」,語法為:

字典名稱 [鍵]

例如:

dict1 = {"香蕉":20, "蘋果":50, "橘子":30} print(dict1["蘋果"]) #50



02 03

04

當字典的鍵重複時

字典是使用「鍵」做為索引來取得「值」,所以「鍵」必須是唯一,而「值」可以重覆。 如果「鍵」重覆的話,則前面的「鍵」會被覆蓋,只有最後的「鍵」有效,例如:

dict2 = {" 香蕉 ":20, " 蘋果 ":50, " 橘子 ":30, " 香蕉 ":25} print(dict2[" 香蕉 "]) #25, 「 " 香蕉 ":20」 被覆蓋

當字典的鍵不存在時

元素在字典中的排列順序是隨機的,與設定順序不一定相同,例如:

dict1 = {"香蕉":20, "蘋果":50, "橘子":30} print(dict1) #結果: {"蘋果":50, "香蕉":20, "橘子":30}

由於元素在字典中的排列順序是隨機的,所以不能以位置數值做為索引。另外,若 輸入的「鍵」不存在也會產生錯誤,例如:

dict1 = {"香蕉":20, "蘋果":50, "橘子":30} print(dict1[0]) # 錯誤 print(dict1["鳳梨"]) # 錯誤

此種字典取值方式當「鍵」不存在時會因錯誤而讓程式中斷,因此 Python 另外提供 3 get 方法可以取得字典元素值,即使「鍵」不存在也不會產生錯誤,語法為:

字典名稱 .get(鍵[, 預設值])

預設值可有可無。根據是否有傳入預設值及「鍵」是否存在可分為四種情形:

預設值狀況	「鍵」是否存在	返回值
次方庫 λ 預乳店	「鍵」存在	返回鍵對應的值
没有傳八頂設值	「鍵」不存在	返回 None
右値 λ 頚乳店	「鍵」存在	返回鍵對應的值
行诗八頂砹伹	「鍵」不存在	返回預設值



字典與集合的使用 04

當「鍵」不存在時會傳回 None 或預設值,程式執行時不會產生錯誤。例如:

dict1 = {"香蕉":20, "蘋果":50, "橘子":30} print(dict1.get("蘋果")) #50 print(dict1.get("鳳梨")) #None print(dict1.get("蘋果", 80)) #50 print(dict1.get("鳳梨", 80)) #80

範例:血型個性查詢

不同血型的人具有不同的個性:設計程式建立4筆字典資料:「鍵」為血型,「值」 為個性。使用者輸入血型後,若血型存在,就顯示該血型的個性,如果血型不存在, 則顯示沒有該血型的訊息。

IPython console	IPython console	
Cansole 1/A 🖸	Console 1/A	
輸入要查詢的血型:B B 血型的個性為:外向樂觀	輸入要查詢的血型≠P 沒有「P」血型!	2
History log IPython console	History log IPython console	2

	程式碼:dictget.py	
	1 dict1 = {"A":"內向穩重", "B":"外向樂觀", "O":" 堅強自信",	20
	"AB":" 聰明自然 "}	
	2 name = input(" 輸入要查詢的血型:")	27
	3 blood = dict1.get(name)	
	4 if blood == None:	
	5	28
	6 else:	
	7	
程	式說明	29
	1 建立 4 個血型及個性的字典。	
	2 讓使用者輸入血型。	
	3 以 get 取得個性。	21
	4-5 若血型不存在就顯示沒有該血型訊息。	
	6-7 若血型存在就顯示該血型的個性。	32



🕢 Ч.Э 集合

集合 (set) 是一個無序且元素不可重複的元素集,集合元素的內容是不可以改變的, 集合元素的內容可以是整數、浮點數、字串、元組…等,但不可以使用串列、字典、 集合。可以增加或刪除集合元素,但集合元素內容是唯一的。

4.3.1 建立集合

04

可以使用大括號 {} 或是 set() 函式建立集合。

使用大括號建立集合

建立集合最簡單的方法是使用大括號 {}。例如:建立 fruits 集合,包括"香蕉", "蘋果", "橘子"三個字串。

```
fruits = {"香蕉", "蘋果", "橘子"}
print(fruits) # {'橘子', '香蕉', '蘋果'}
print(type(fruits)) # <class 'set'>
```

集合的元素是不可重複的,如果資料重複,多餘的元素將會被捨去。下面的範例中 「香蕉」因為重複,因此只會保留一筆香蕉的資料。

fruits = {"香蕉", "蘋果", "橘子", "香蕉"} print(fruits) # {'橘子', '香蕉', '蘋果'}

可以使用不同型別的集合元素。

fruits = {"香蕉", "蘋果", "橘子", 100, (1,2)} print(fruits) # {(1, 2), 100, '香蕉', '蘋果', '橘子'}

但不可以使用串列、字典、集合,下列範例中使用串列當作集合的元素將會出現 「TypeError: unhashable type: 'dict'」的錯誤。

fruits = {"香蕉", "蘋果", "橘子", 100, [1,2]}

使用 set() 函式建立集合

另一種方式是以 set() 函式建立集合, set() 函式的參數內容可以是字串、串列、元組, 這些參數都會轉換成集合元素。例如:將串列轉換為集合。



4-13

www.gotop.com.tw

```
fruits = set(["香蕉", "蘋果", "橘子", "香蕉"])
print(fruits) # {'橘子', '香蕉', '蘋果'}
print(type(fruits)) # <class 'set'>
```

如果集合的參數是字串,集合元素將轉換成字元,同時會去除重複的字元。

```
s = set("Good Boy!")
print(s)  # {'y', '!', 'o', ' ', 'd', 'B', 'G'}
print(type(s))  # <class 'set'>
```

要建立一個空集合必須使用 set() 函式,不可以使用 {},因為 {} 建立的不是集合而是 一個空字典。

```
      s1={}
      # 空字典

      print(type(s1))
      # <class 'dict'>

      s2 = set()
      # 空集合

      print(type(s2))
      # <class 'set'>
```

範例:去除重複的資料

某間知名的企業收集了大量使用者的資料,但仔細分析後發現這些資料很多都是重 複的,必須去除重複的資料。

```
    Console I/A □
    【 '林小明', '曾山水', '鄭美麗']
    [ '林小明 ' 曾山水', '鄭美麗']
    林小明
```

木手	五礁	SP	τ.	nv

```
1 persons = ["林小明","曾山水","鄭美麗","林小明","曾山水","林小明"]
2 s = set(persons) # 串列轉集合
3 print(s) # {'林小明', '曾山水', '鄭美麗']
4 list1 = list(s) # 集合轉串列
5 print(list1) # ['林小明', '曾山水', '鄭美麗']
6 print(list1[0]) # 林小明
```

```
程式說明
```

1	建立串列。	
2-3	將串列轉換成集合並顯示。	32
4-5	將集合轉換成串列並顯示。	
6	集合不可以索引讀取,必須轉換成串列才能以索引來讀取。	

6-17

www.gotop.com.tw

❷ 6.8 打造自己的模組

一個較大型專案,程式是由許多類別或函式組成,為了程式的分工和維護,可以適 度地將程式分割成許多的模組,再匯入並呼叫這些模組。

6.8.1 準備工作

下列程式包含計算兩數相加、兩數相減的兩個函式,可以直接呼叫 add、sub 函式執行兩數相加、相減的運算。



e x
. <i></i>
^

有時為了程式的分工和維護,我們會將程式分割成模組。

6.8.2 打造自己的模組

首先我們將 add、sub 兩個函式建立成一個獨立的模組,模組名稱為 <calculate. py>。

乑	星式碼:calculate.py	31
1	<pre>def add(n1,n2):</pre>	JI
2	return n1+n2	
3		
4	<pre>def sub(n1,n2):</pre>	
5	return n1-n2	
		33

01	6.8.3 匯入自己建立的模組
02	可以使用下列不同的 import 方法匯入並呼叫模組內的函式。
03	import 模組名稱
	以 import 匯入自己建立的模組後,即可以呼叫使用這些模組內的函式。
04	匯入自己建立的模組語法:
	import 模組名稱
	例如:匯入 calculate 模組。
06	import calculate
07	這種方式呼叫函式時,必須加上模組名稱,語法:
07	模組名稱 . 函式名稱
	範例:匯入 <calculate.py> 模組並呼叫模組內的 add、sub 函式。</calculate.py>
09	程式碼:module-2.py import calculate # 匯入 calculate 模組
10	<pre>print(calculate.add(5,2)) # 7 print(calculate.sub(5,2)) # 3</pre>
11	匯入模組內函式
12	每次使用模組內的函式都要輸入模組名稱非常麻煩,下列 import 的方法可改善此種 情況,語法為:
13	from 模組名稱 import 函式名稱 1[, 函式名稱 2,, 函式名稱 n]
14	這種方式呼叫函式時,可以省略模組名稱,直接以函式名稱呼叫。
	範例:匯入 <calculate.py> 模組內的 add、sub 函式,並呼叫模組內 add、sub 函式。</calculate.py>
15	程式碼:module-3.py
16	1 from calculate import add, sub
17	2 3 print(add(5,2)) # 7 4 print(sub(5,2)) # 3

www.gotop.com.tw

II.3.3 實戰: Facebook 自動登入

2 在這個專題中將要使用 Selenium 的功能來控制瀏覽器,進行自動登入 Facebook 網站的動作。

範例:自動登入 Facebook

- ▲ 要登入 Facebook 網站的步驟如下:
 - 1. 輸入網址「https://www.facebook.com/」。
 - 2. 輸入帳號、密碼。
 - 3. 按**登入**鈕。
 - 現在,我們要改用 Python 程式自動完成登入 Facebook 網站的動作。
- 7 在開發之前要先查詢在 Facebook 登入頁面上的欄位,以便等一下使用 Selenium 模 組時可以找到要輸入電子郵件、密碼欄位及登入鈕的位置。
 - 開啟 Chrome 開發人員工具,找尋 電子郵件或電話 欄位原始碼的位置,如下圖可以在標籤中找到 id 值為「email」。



2. 找尋密碼欄位原始碼的位置,可以在標籤中找到 id 值為「pass」。

← C • fa	icebook.com					04		0	黑肌肉:	¢	÷
	1014 TTR	RE	Ð	Eléments	Console	Sources	>>		a.2	1	×
註冊	Input#pass.inputtext.login_form_in 150+242 put.box	in.	D.		<pre>* <td ing ing test *</td </pre>	nul type puttext l c pass Lid roya <th>passo ogin_j id "pi 1_pass</th> <th>ward form_ ass</th> <th>inpot deta - 50</th> <th>bax</th> <th></th>	passo ogin_j id "pi 1_pass	ward form_ ass	inpot deta - 50	bax	

3. 找尋 登入 鈕原始碼的位置,可以在標籤中找到 name 值為「login」。

🗧 😋 🖬 lacebook.com		
	* 🕼 🔂 Elements	Console Sources Network Performance Memory Applic
人 经	-	> button value 1 class _42ft _4jy0 _61th _4jy6 _4jy1 selected _51sy name login _cata-testid royal login but
\$ F 8 4,7		ton type submit id u_0_0_000/button == ED (/div)



11

數據資料的爬取

1-29

www.gotop.com.tw

完成登入時最重要的三個欄位資訊查詢後,就可以動手寫程式了喔!

en 11		
程式	、碼:loginFacebook.py	
1	from selenium import webdriver	
2	# 設定 facebook 登入資訊	
3	url = 'https://www.facebook.com/'	
4	email='你的 faceook 電子郵件 '	
5	password=' 你的 faceook 密碼 '	
6	# 建立瀏覽器物件	
7	driver = webdriver.Chrome()	
8	# 最大化視窗後開啟 facebook 網站	
9	driver.maximize_window()	
10	driver.get(url)	
11	# 執行自動登入動作	
12	driver.find_element_by_id('email').send_keys(email) # 輸入郵件	
13	driver.find_element_by_id('pass').send_keys(password)#輸入密碼	
14	driver.find_element_by_name('login').click() # 按登入鈕	
記說	明	_
3-4	設定 Facebook 登入資訊,請將電子郵件及密碼置換成你的資料。	

- 9-10 最大化視窗後開啟 Facebook 網站。
- 12 找到電子郵件輸入的欄位後,用 send_keys()方法輸入郵件資料。
- 13 找到密碼輸入的欄位後,用 send_keys()方法輸入密碼資料。
- 14 找到登入按鈕後用 click() 方法按下按鈕將表單送出,執行登入的動作。

當執行程式時,Selenium 會自動開啟瀏覽器,並且前往 Facebook 的頁面,在自動輸入完電子郵件及密碼後,程式會按下登入鈕,就能成功進入 Facebook 的個人頁面了,是不是很神奇啊!

C Inner-Alzint × + + = C A Intercolum	× B 0 1	🖬 entriestaat 🔺 👎 b 12 's fanstaaksen	* 8 0 1
facebook	tisinati ta a	Const Brading Condition	Dana ak wi 45 P
Facebook · 國際和國際的安全的問題 · 國際分 會活用中的第一日 ·	#### #### ######		

14.9.2 繪製長條圖、橫條圖、堆疊圖

在範例中新增公司北中南三區 2015 到 2019 年的分區銷售資料,分別繪製長條圖、 橫條圖及堆疊圖:

程式	碼:plot1.py
1	import pandas as pd
2	import matplotlib.pyplot as plt
3	# 設定中文字型及負號正確顯示
4	<pre>plt.rcParams["font.sans-serif"] = "mingliu"</pre>
5	<pre>plt.rcParams["axes.unicode_minus"] = False</pre>
6	
7	df = pd.DataFrame([[250,320,300,312,280],
	[280,300,280,290,310],
	[220,280,250,305,250]],
	index=['北部 ',' 中部 ',' 南部 '],
	columns=[2015,2016,2017,2018,2019])
8	
9	g1 = df.plot(kind='bar', title=' 長條圖 ', figsize=[10,5])
10	g2 = df.plot(kind='barh', title=' 橫條圖 ', figsize=[10,5])
11	g3 = df.plot(kind='bar', stacked=True, title=' 堆疊圖 ',
	figsize=[10,5])

執行結果:





程式說明	18
 1-2 匯入 pandas、matplotlib.pyplot 模組並設定別名。 4-5 利用 matplotlib 模組的功能修正中文顯示問題。 7 建立公司分隔各年度銷售資料 DataFrame。 	19
 ■ 9 繪製長條圖。 ■ 10 繪製橫條圖。 	20
■ 11 繪製堆疊圖,其實就是長條圖加上 stacked=True 參數。	21
14.9.3 繪製折線圖 範例中將公司北中南三區 2015 到 2019 年的分區銷售資料分別繪製成折線圖:	22
程式碼:plot2.py	23
<pre> 7 g1 = df.iloc[0].plot(kind='line', legend=True,</pre>	24
title='公司分區年度銷售表', figsize=[10,5]) 8 g1 = df.iloc[1].plot(kind='line',	25
<pre>legend=True,</pre>	26
legend=True, xticks=range(2015,2020))	27
執行結果:	28
	29
20	30
☆	31
 7 以 df.iloc[0] 調出北區資料進行折線圖繪製。 8 以 df.iloc[1] 調出中區資料進行折線圖繪製。 	32
■ 9 以 df.iloc[2] 調出南區資料進行折線圖繪製。	33
	14-39

14.9.4 繪製圓餅圖

範例中將公司北中南三區 2015 到 2019 年的分區銷售資料分別繪製成圖餅圖:



www.gotop.com.tw

רו-די

www.gotop.com.tv



如果目標值為連續數值時需使用迴歸演算法進行機器學習訓練及預測,線性迴歸演 算法是最間單的迴歸演算法。

23.4.] 線性迴歸演算法原理

線性迴歸演算法是在已知資料點中找出規律,畫出一條最接近資料點的直線,再以 此直線對未知資料進行預測的演算法。

以下面的例子說明:先生想購買一顆 1.1 克拉的鑽石送給太太,作為結婚十週年的禮物。到珠寶店後發現無現貨可買,但店家可幫忙代購。先生向店家詢問價錢以便提 23款,但珠寶店因未賣過此重量鑽石無法回答,於是先生向珠寶店索取過去鑽石重量與價錢資料自行估算,將鑽石重量與價錢資料繪製圖形如下,再自行畫一條最接近 24 資料點的直線,於是估計 1.1 克拉鑽石大約 26 萬元。



此方法的關鍵是機器學習如何找到最接近資料點的直線呢?

損失函數

找到最接近資料點直線的方法是利用損失函數計算預測值與真實值的誤差,誤差最 小時就是最佳直線。最常用的損失函數為最小方差。

數學上的直線方程式為:

$$y = wx + b$$

w 稱為權重,b 稱為偏置:此時問題變成如何找到最適當直線的 w 及 b 呢?方法是 將資料點 (x) 逐一代入直線方程式求出預測值 (y),然後計算每一個預測值與真實值 差的平方總和,此總和稱為「方差」。

方差 = \sum (預測值 - 真實值)²

不斷改變 w 及 b 的值, 方差最小時的 w 及 b 值就是最佳直線。

上面是一個特徵的情況,如果是多個特徵時該如何呢?以前面鑽石的例子來說,影 響鑽石價格的因素不是只有重量而已,鑽石的顏色、純度、切割等都會影響價格。 在多特徵的線性迴歸演算法中,只要將所有特徵結合在一起即可,例如多個特徵值 分別為 x₁、x₂、……,則其方程式為:

 $y = w_1 x_1 + w_2 x_2 + \cdots + b$

同樣將所有資料點代入方程式,找到最小方差的 w₁、w₂、……、b 即可。

線性迴歸演算法中找到最小方差權重值及偏置值常用的方法,有正規方程法及梯度 下降法。

23.4.2 正規方程法線性迴歸

正規方程法是最簡單的線性迴歸演算法,它利用特徵值矩陣的轉置及反矩陣複雜運 算直接計算出最佳權重值及偏置值。

正規方程法線性迴歸僅適用於特徵數量較少的情況,因反矩陣運算的複雜度很高, 當特徵較多時,正規方程法會耗費非常龐大的計算資源與時間,並可能無法計算出 最佳權重值及偏置值。

6 使用線性迴歸演算法時,各特徵的數值大小會影響結果,因此訓練前需將各特徵值 進行標準化。

在 scikit-learn 要使用正規方程法線性迴歸,首先匯入模組:

from sklearn.linear_model import LinearRegression

接著建立 LinearRegression 物件, 語法為:

正規變數 = LinearRegression()



25

5-15

www.gotop.com.tw

25.4 實戰:市場股價預測

本節將建立循環神經網路模型,先從證券交易所取得股票資料訓練模型,然後儲存 訓練完成的模型,最後利用模型預測股價。

25.4.] 蒐集股票資料

本章範例中附上已從證券交易所取得的股票資料:<twstock20xx.csv> 為全年股票 資料,例如<twstock2018.csv> 為 2018 年股票資料;<twstock_all.csv> 則為 2015 到 2018 年全部股票資料,本範例是以 <twstock_all.csv> 為訓練及測試資料。(取 得股票資料的方法請參考基峰資訊出版的《Python 機器學習與深度學習特訓班》或 《Python 大數據特訓班》)。

股票資料檔案的結構為:

G	▲ tw	stock_all.csv ×								
	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↓
I	2	2015-01-05,	25369082,	2210737580,	87.8,	87.8,	86.9,	87.2,	-0.7,	12695↓
I	- 3	2015-01-06,	65967122,	5611395850,	86.0,	86.0,	84.5,	84.5,	-2.7,	33773↓
I	- 4	2015-01-07,	33872596,	2884075500,	84.3,	85.8,	84.3,	85.2,	0.7,	15794↓
l	5	2015-01-08,	31001877,	2683506287,	85.9,	87.0,	85.7,	86.9,	1.7,	14395↓
I	6	2015-01-09,	22801642,	1984878503,	87.3,	87.8,	86.5,	86.5,	-0.4,	11266↓

本範例是以「收盤價」做為訓練及預測的資料。

25.4.2 股票資料前處理

取得收盤價資料

由證券交易所取得的股票資料包含的資料繁多,本範例只使用其中收盤價資料做為 訓練及預測,因此先將收盤價資料抽取出來。

首先利用 pandas 模組讀取全部股票資料:

```
import pandas as pd
df = pd.read_csv('twstock_all.csv', encoding='big5')
```

然後取得其中「收盤價」欄位的資料存於 dfprice 變數中:

```
dfprice=pd.DataFrame(df['收盤價'])
```





程式說明

30	載入模型。
3 1	進行預測。
32-33	將預測值還原為原始值。
■ 34	將真實資料還原為原始值。
36-39	繪製預測值及直實資料值圖形。

執行結果:

14 15 16





· CB.7 實戰:大量圖片處理

在電腦檔案的操作中,經常會有需要將大量的檔案重新命名,再分別整理到指定資料 夾的需求。例如製作網頁時,為了網路傳輸速度考量,會將圖片的解析度降低、更 改所有的圖檔為相同大小,或是將圖片轉換成另一種格式的檔案、圖片轉灰階等等。 其實只要利用 pillow 超強的圖片處理功能,即可輕鬆的完成這些任務。

範例:大量圖片處理

將 pic 目錄中所有圖片,全部轉換為 800*600 大小,再另存為.bmp 檔 (存在 bmp_photo 目錄),同時也將全部圖片灰階處理後依編號存成 gray001.jpg、gray002.jpg… 等檔案 (存在 gray_photo 目錄)。

1	def emptydir(dirname):
2	if os.path.isdir(dirname):
3	shutil.rmtree(dirname)
4	sleep(1) # 需延遲, 否則會出錯
5	os.mkdir(dirname)
6	
7	import PIL
8	from PIL import Image
9	import glob
10	import shutil, os
11	from time import sleep
12	
13	image_dir="pic"
14	target_dir = 'bmp_photo'
15	target_dir2 = 'gray_photo'
16	emptydir(target_dir)
17	emptydir(target_dir2)
18	<pre>files=glob.glob(image_dir+"*.jpg") + glob.glob(image_dir+"*.png")</pre>
19	for i, f in enumerate(files):
20	<pre>img = Image.open(f)</pre>
21	img_new = img.resize((800, 600), PIL.Image.ANTIALIAS)
22	path,filename = f.split("\\") #路徑、檔名
23	name,ext = filename.split(".") #主檔名、副檔名
24	# 以 bmp 格式存檔
25	img_new.save(target_dir+'/' + name + '.bmp')



28-19

www.gotop.com.tw

26		
27	# 轉換為灰階	
28	<pre>img_gray = img_new.convert('L')</pre>	1
29	# gray001.jpg、gray002.jpg	
30	outname = str("gray") + str('{:0>3d}').format(i+1) + '.jpg'	
31	<pre>img_gray.save(target_dir2+'/'+outname)</pre>	
32	print("{} 複製完成!".format(f))	
33	<pre>img.close()</pre>	
34		2
35	print('轉換尺寸及灰階處理結束!')	

執行結果:

左下圖為主檔名不變,在 <bmp_photo> 目錄另存為 .bmp 檔,右下圖為先轉換為灰 階後,在 <bmp_photo> 目錄另存為 gray001.jpg、gray002.jpg…檔。



程式說明

1-5	emptydir 自訂函式建立空的資料夾,若資料夾已存在,就先刪除再建 立新資料夾。	28
2-3	若資料夾已存在,就先刪除該資料夾。	
4	刪除資料夾需一些時間,所以延遲1秒。	29
5	建立資料夾。	
7-11	匯入相關的模組。	30
13	設定來源圖片目錄。	
14-15	設定圖片轉換後儲存的目錄。	31
16-17	建立目的資料夾。	
18	讀取來源資料夾中所有圖片檔。	32
19-33	逐一將圖片檔案轉換尺寸、另存副檔名、轉灰階、另存為 jpg 檔。	
	17 12	

ᅴ.].닉 Colab 連接 Google Drive 雲端硬碟

由於 Colab 檔案存於 Google Drive 雲端硬碟,因此如果有程式中需要使用的檔案, 必須上傳到 Google Drive 雲端硬碟。

上傳檔案到 Google Drive 雲端硬碟

在 Google Drive 雲端硬碟中切換到 <Colab Notebooks> 資料夾,按左上方 新增 鈕, 再點選 檔案上傳,於 開啟 對話方塊選擇要上傳的檔案就可將該檔案上傳到雲端硬碟 的 <Colab Notebooks> 資料夾。例如上傳本章範例 <ATM00625_20200513155248. csv> 檔。

1 資料夾	e硬碟 > Colab Notebook	5	⊞ () ∎
 檔案上傳 資料夾上傳 	◎ 開啟 ← - · · ↑ 3 · · · 本機磁碟 (C:) > exa	ample - ర	ク 授尋 example
■ Google 文件 Google 試算表	组合管理 ▼ 新増資料夾 名稱 ^	修改日期	 ・ 3 ・ 3 ・ 4 ・ 4
Google 簡報 更多	ATM00625_20200513155248.csv	2021/5/8下午12:17	Microsoft Excel 痘 7
日前使用豪 . 5 2 6 9			
間配額: 15 GB)	檔案名稱(N): ATM00	625 20200513155248 csv v	所 右 檔 宏 (* *)

<ATM00625_20200513155248.csv> 檔是全台 PM2.5 測站的檢測資料。上傳後可在 Google Drive 雲端硬碟看到該檔案。

4 雲端硬碟	Q、 在雲端硬碟中搜尋		- 😢 🗇 🏟 🏭	6
→ 新増	我的雲端硬碟 > Colab Notebooks	7	Ⅲ ()	
▼ △ 我的雲端硬碟	- 名磺 ↑	擁有奇	上次修改時間	ŀ
▶ ■ 雜項部落格	imgTest	我	2020年5月27日 我	Ľ
▶ In AndroidStudio部落格	ATM00625_20200513155248,csv	我	下年8.48	
Colab Notebooks	autoColor.ipynb	我	2020年5月24日 我	
▶ GoogleAppScript部落格 目前便用量・33 GB (結応会)	carplate.ipynb	我	2020年7月2日 我	
間配額: 15 GB)	firstlab.ipynb	我	下午6:38 我	



31-9

www.gotop.com.tw

、原始	的格式上傳					
傳檔	案到 Google	Drive 雲端硬行	蹀時,需確保 ④ ✿ ■ = 、	是以原始格式	式上傳,否則 日,於 乳 中	则在 Colab
消核	電楽町首座王 巽 將已上傳 6	り檔案轉換成(四	扁輯器格式 項		
4	雲端硬碟	Q、 在雲端硬碟中打	要尋	÷	00	
+	新增	我的雲端硬碟 >	Colab Notebooks 🔻	設定	(D 🔤
* 🛆	我的雲端硬碟	名稱 个		建盤快速鍵		
	設定	invnb checknoin	ts	32 2021	1年7月21日 我	完成
	—殿		購買儲存空間 2	檢視項目佔用的儲存空間		
	通知 管理應用程式 軒	转上傳檔案	网络巴上傳的檔	窦轉换成 Google 文件編輯器科	街式	

Colab 連接 Google Drive 雲端硬碟

Colab 要使用 Google Drive 雲端硬碟的檔案,首先要讓 Colab 連接上 Google Drive 雲端硬碟:點按 Colab 左方資料夾管理 □鈕,再按 掛接雲端硬碟 № 圖示。

CO L firstlab.ipynb ☆ 檔案 编輯 檢視畫面 插入 執行術	跟 工具 說明 <u>已儲存所有變更</u>	目留言 🏎 共用 🌣 👜
∈ 檔案 ×	+ 程式碼 + 文字	✓ RAM ★ / 編輯 ▲
	<pre>[1] 1 for i in range(5): 2</pre>	
🖵 🕨 sample_data	1	↑ 0 0 1 1 1
	示範串列使用	

按 連線至 Google 雲端硬碟 鈕。





33

F-FF

www.gotop.com.tw