APPENDIX

B

Python 文字檔案存取與字串處理

- B-1 Python 文字檔案存取
- B-2 Python 字串處理

B-1 Python 文字檔案存取

Python 提供檔案處理（File Handling）的內建函數，可以讓我們將資料寫入文字檔案，和讀取文字檔案的資料。

☆ 將 Request 取得的回應內容寫成檔案：appb-1-1.py

我們準備將網站內容的 HTML 標籤字串儲存成 Example.txt 檔案，如下所示：

```
import requests

r = requests.get("https://fchart.github.io/Example.html")

fp = open("Example.txt", "w", encoding="utf8")
fp.write(r.text)
print("寫入檔案Example.txt...")
fp.close()
```

上述程式碼使用 open() 函數開啟檔案，close() 函數關閉檔案，如下所示：

```
fp = open("Example.txt", "w", encoding="utf8")
```

上述函數的回傳值是檔案指標，第 1 個參數是檔案名稱或檔案完整路徑，如果內含路徑「\」符號，Windows 作業系統需要使用逸出字元「\\」，第 2 個參數是檔案開啟的模式字串，支援的開啟模式字串說明，如下表所示：

模式字串	當開啟檔案已經存在	當開啟檔案不存在
r	開啟唯讀的檔案	產生錯誤
w	清除檔案內容後寫入	建立寫入檔案
a	開啟檔案從檔尾後開始寫入	建立寫入檔案
r+	開啟讀寫的檔案	產生錯誤
w+	清除檔案內容後讀寫內容	建立讀寫檔案
a+	開啟檔案從檔尾後開始讀寫	建立讀寫檔案

最後的 encoding 參數指定使用的編碼，以此例是 utf8，其執行結果可以看到寫入檔案的訊息文字，如下所示：

```
寫入檔案 Example.txt...
```

☆ 讀取檔案的全部內容（一）：appb-1-1 a.py

讀取和顯示 appb-1-1.py 建立的 Example.txt 檔案內容，如下所示：

```
fp = open("Example.txt", "r", encoding="utf8")
str = fp.read()
print("檔案內容:")
print(str)
```

上述 open() 函數的模式字串是 "r"，即讀取檔案內容，然後呼叫 read() 函數，當函數沒有參數時，就是讀取檔案全部內容，其執行結果可以顯示檔案內容，如下所示：

```
檔案內容:
<!DOCTYPE html>

<html>

<head>

<meta charset="utf-8"/>

<title>Example.html</title>

<style type="text/css">

.blue { color: blue; }

.red { color: red; }

.green { color: green; }

.line { border: 1px solid #333; }
```

```
</style>
<script src="jquery-3.1.0.min.js"></script>
<script>
$(document).ready(function() {
   $('*').addClass('line');
   $('p').addClass('blue');
   $('#list').addClass('red');
   $('.item').addClass('green');
});
</script>
</head>
<body>
<p>Python 網路資料擷取</p>
<p>建立網路爬蟲程式</p>
<ol id="list">
  <li>CSS選擇器</li>
  <li class='item'>XPath表達式</li>
  <li>正規表達式</li>
  <li class='item'>DOM瀏覽方法</li>
</ol>
</body>
</html>
```

☆ Python 的 with/as 程式區塊：appb-1-1b.py

請注意！Python 檔案處理需要在處理完後自行呼叫 close() 函數來關閉檔案，對於一些需要善後的操作，如果擔心忘了執行這些工作，我們可以改用 with/as 程式區塊讀取檔案內容，如下所示：

```
with open("Example.txt", "r", encoding="utf8") as fp:
    str = fp.read()
    print("檔案內容:")
    print(str)
```

上述程式碼建立讀取檔案內容的程式區塊（不要忘了 fp 後的「:」冒號），當執行完程式區塊，就會自動關閉檔案。

☆ 讀取檔案的全部內容（二）：appb-1-1c.py

讀取和顯示 appb-1-1.py 建立的 Example.txt 檔案內容，如下所示：

```
with open("Example.txt", "r", encoding="utf8") as fp:
    list1 = fp.readlines()
    for line in list1:
        print(line, end="")
```

上述程式碼是使用 readlines() 函數讀取檔案內容成為 list1 串列，每一行是一個項目，然後使用 for/in 迴圈顯示每一行的檔案內容，因為檔案中的每一行有換行，所以在 print() 函數就不需要換行。

B-2 Python 字串處理

資料清理的主要工作是處理從網路爬蟲取得的資料，這些資料都是字串資料，我們可以使用 Python 字串處理的函數和切割運算子來處理取得的字串資料。

B-2-1 建立 Python 字串

Python「字串」（Strings）是使用「'」單引號或「"」雙引號括起的一序列 Unicode 字元，這是一種不允許更改（Immutable）內容的資料型態，所有字串的變更事實上都是建立了一個全新的字串。

☆ 建立 Python 字串：appb-2-1.py

Python 程式可以指定變數值是一個字串，如下所示：

```
str1 = "學習 Python 語言程式設計"
str2 = 'Hello World!'
ch1 = "A"
```

上述前 2 列程式碼是建立字串，最後 1 列是字元（這是只有 1 個字元的字串，我們可以將它視為字元），我們也可以使用物件方式建立字串，如下所示：

```
name1 = str()
name2 = str("陳會安")
```

上述第 1 列程式碼建立空字串，第 2 列建立內容是 "陳會安" 的字串物件。在建立字串後，可以使用 print() 函數輸出字串變數，如下所示：

```
print(str1)
print(str2)
```

在 print() 函數也可以使用字串連接運算式來輸出字串變數，因為是字串變數，所以不需要呼叫 str() 函數轉換成字串型態，如下所示：

```
print("ch1 = " + ch1)
print("name1 = " + name1)
print("name2 = " + name2)
```

☆ 走訪 Python 字串的每一個字元：appb-2-1a.py

字串就是一序列 Unicode 字元，我們一樣可以使用 for/in 迴圈來走訪顯示每一個字元，正式的說法是迭代（Iteration），如下所示：

```
str3 = 'Hello'

for e in str3:
    print(e)
```

上述 for/in 迴圈在 in 關鍵字後的是字串 str3，每執行一次 for/in 迴圈，就從字串第 1 個字元開始，取得一個字元指定給變數 e，並且移至下一個字元，直到最後 1 個字元為止，其操作如同從字串的第 1 個字元走訪至最後 1 個字元，可以依序輸出 H、e、l、l 和 o。

B-2-2 字串函數

Python 提供多種字串函數來幫助我們處理字串。如果是使用物件的字串函數，我們需要使用物件變數加上「.」句號來呼叫，如下所示：

```
str1 = 'welcome to python'
print(str1.islower())
```

上述程式碼建立字串 str1 後，呼叫 islower() 函數檢查內容是否都是小寫英文字母，請注意！字串函數不只可以使用在字串變數，也可以直接使用在字串字面值來呼叫（因為 Python 都是物件），如下所示：

```
print("1000".isdigit())
```

☆ Python 內建的字串函數：appb-2-2.py

Python 內建字串函數可以取得字串長度、在字串中的最大和最小字元，其說明如下表所示：

字串函數	說明
len()	回傳參數字串的長度，例如：len(str1)
max()	回傳參數字串的最大字元，例如：max(str1)
min()	回傳參數字串的最小字元，例如：max(str1)

☆ 檢查字串內容的函數：appb-2-2a.py

字串物件提供檢查字串內容的相關函數，其說明如下表所示：

字串函數	說明
isalnum()	如果字串內容是英文字母或數字，回傳 True；否則為 False，例如：str1.isalnum()
isalpha()	如果字串內容只有英文字母，回傳 True；否則為 False，例如：str1.isalpha()
isdigit()	如果字串內容只有數字，回傳 True；否則為 False，例如：str1.isdigit()
isidentifier()	如果字串內容是合法的識別字，回傳 True；否則為 False，例如：str1.isidentifier()
islower()	如果字串內容是小寫英文字母，回傳 True；否則為 False，例如：str1.islower()
isupper()	如果字串內容是大寫英文字母，回傳 True；否則為 False，例如：str1.isupper()
isspace()	如果字串內容是空白字元，回傳 True；否則為 False，例如：str1.isspace()

☆ 搜尋子字串函數：appb-2-2b.py

字串物件關於搜尋子字串的函數，其說明如下表所示：

字串函數	說明
endswith(str1)	如果字串內容是以參數字串 str1 結尾，回傳 True；否則為 False，例如：str2.endswith(str1)
startswith(str1)	如果字串內容是以參數字串 str1 開頭，回傳 True；否則為 False，例如：str2.startswith(str1)
count(str1)	回傳字串內容出現多少次參數字串 str1 的整數值，例如：str2.count(str1)
find(str1)	回傳字串內容出現參數字串 str1 的最小索引位置值，沒有找到傳回 -1，例如：str2.find(str1)
rfind(str1)	回傳字串內容出現參數字串 str1 的最大索引位置值，沒有找到傳回 -1，例如：str2.rfind(str1)

☆ 轉換字串內容的函數：appb-2-2c.py

字串物件支援轉換字串內容的相關函數，可以輸出英文大小寫轉換的字串，或取代字串內容，其說明如下表所示：

字串函數	說明
capitalize()	回傳只有第 1 個英文字母大寫的字串，例如：str1.capitalize()
lower()	回傳小寫英文字母的字串，例如：str1.lower()
upper()	回傳大寫英文字母的字串，例如：str1.upper()
title()	回傳字串中每 1 個英文字的第 1 個英文字母大寫的字串，例如：str1.title()
swapcase()	回傳英文字母大寫變小寫；小寫變大寫的字串，例如：str1.swapcase()
replace(old, new)	將字串中參數 old 的舊子字串取代成參數 new 的新字串，例如：str1.replace(old_str, new_str)

B-2-3 字串切割運算子

Python 程式碼不只可以使用「[]」索引運算子取出指定索引位置的字元，索引運算子還是一種「切割運算子」（Slicing Operator），可以從原始字串切割出所需的子字串。

☆ 使用索引運算子取得字元：appb-2-3.py

Python 字串可以使用「[]」索引運算子取出指定位置的字元，索引值是從 0 開始，而且可以是負值，如下所示：

```
str1 = 'Hello'

print(str1[0])    # H
print(str1[1])    # e
print(str1[-1])   # o
print(str1[-2])   # l
```

上述程式碼依序顯示字串 str1 的第 1 和第 2 個字元，-1 是最後 1 個，-2 是倒數第 2 個。

☆ 切割字串：appb-2-3a.py

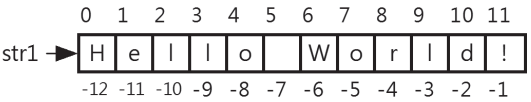
切割運算子（Slicing Operator）的基本語法，如下所示：

```
str1[start:end]
```

上述 [] 語法中使用「:」冒號分隔 2 個索引位置，可以取回字串 str1 從索引位置 start 開始到 end-1 之間的子字串，如果沒有 start，就是從 0 開始；沒有 end 就是到字串的最後 1 個字元。例如：本節範例字串 str1 的字串內容，如下所示：

```
str1 = 'Hello World!'
```

上述字串的索引位置值可以是正，也可以是負值，如下圖所示：



現在，就讓我們來看一些切割字串的範例，如下表所示：

切割字串	索引值範圍	取出的子字串
str1[1:3]	1~2	"el"
str1[1:5]	1~4	"ello"
str1[:7]	0~6	"Hello W"
str1[4:]	4~11	"o World!"
str1[1:-1]	1~(-2)	"ello World"
str1[6:-2]	6~(-3)	"Worl"

B-2-4 切割字串成為串列和合併字串

Python 字串可以使用 split() 函數將字串切換成串列，反過來，我們可以使用 join() 函數將串列以指定連接字串合併成一個字串。

☆ 切割字串成為串列：split() 函數

字串物件提供相關函數可以使用分隔字元，將字串內容以分隔字元切割字串成為串列，其說明如下表所示：

字串函數	說明
split()	沒有參數是使用空白字元切割字串成為串列，我們也可以指定參數的分隔字元
splitlines()	使用新行字元「\n」切割字串成為串列

例如：使用 split() 函數將一個英文句子的每一個單字切割成串列（Python 程式：appb-2-4.py），如下所示：

```
str1 = "This is a book."
list1 = str1.split()
print(list1)     # ['This', 'is', 'a', 'book.']
```

我們也可以指定 split() 函數使用參數「,」的分隔字元來切割字串成為串列，如下所示：

```
str2 = "Tom,Bob,Mary,Joe"
list2 = str2.split(",")
print(list2)     # ['Tom', 'Bob', 'Mary', 'Joe']
```

如果是從檔案讀取的字串，因為其中的每一行是使用「\n」新行字元來分隔，除了呼叫 split("\n") 函數，也可以直接呼叫 splitlines() 函數，將字串切割成串列，如下所示：

```
str3 = "23\n12\n45\n56"
list3 = str3.splitlines()
print(list3)     # ['23', '12', '45', '56']
```

上述字串內容是使用「\n」新行字元分隔的數字資料，在切割字串建立成串列後，可以看到串列項目都是數值字串，並不是整數。

☆ 合併串列成為字串：join() 函數

Python 字串的 join() 函數可以將串列的每一個元素使用連接字串連接成單一字串（Python 程式：appb-2-4a.py），如下所示：

```
str1 = "-"
list1 = ['This', 'is', 'a', 'book.']
print(str1.join(list1))     # 'This-is-a-book.'
```

上述程式碼的 str1 是連接字串，list1 是欲連接的串列，可以顯示連接後的字串內容：'This-is-a-book.'。