Chapter 3

REQUEST+BEAUTIFULSOUP

進行網頁互動

安裝模組

開啟終端機（或 命令提示字元）安裝下列模組：

★安裝requests模組：pip install requests

★安裝beautifulsoup模組：pip install beautifulsoup4

★安裝 lxml模組：pip install lxml

使用python開啟網頁

先確認想要瀏覽的網頁URL網址

匯入webbrowser模組(已內建無須安裝)

```
import webbrowser
webbrowser.open("URL網址")
```

用python查詢google地圖

先去google地圖確認URL網址

再利用webbrowser模組設計查詢地圖程式

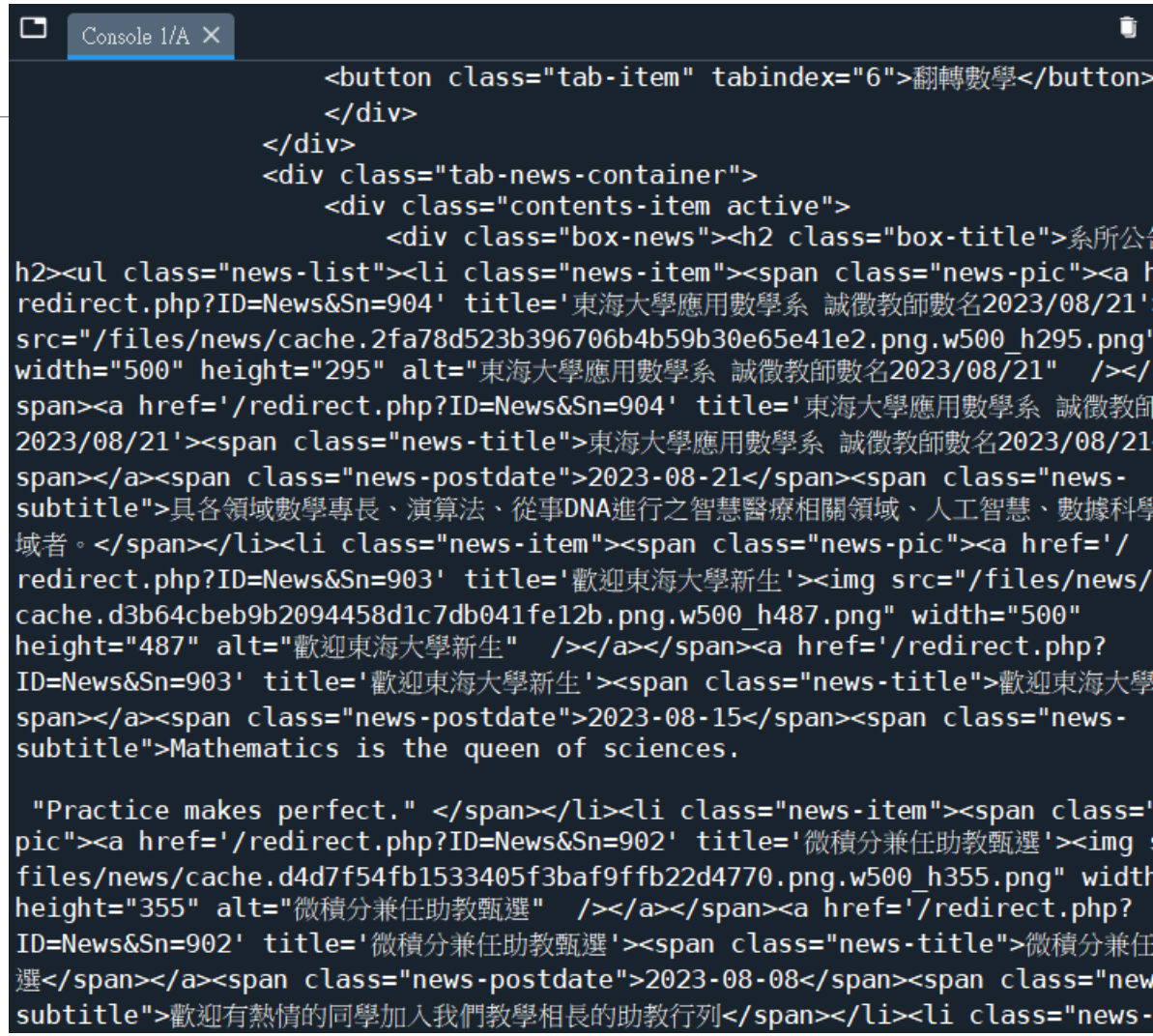
```
import webbrowser
address = input("請輸入地址:")
webbrowser.open('http://www.google.com.tw/maps/place/' + address)
```

讀取HTML網頁資料

爬取東海大學應用數學系網頁

(可開啟記事本ch3-1，複製貼上程式碼)

```
import requests

url = "https://www.math.thu.edu.tw/"
html = requests.get(url)
html.encoding = "utf-8"
print(html.text)
```



網頁檢查碼 : 200

```
import requests
url = "https://www.math.thu.edu.tw/"
html = requests.get(url)
html.encoding = "utf-8"
print(html.status_code)
```

回傳 200 表示網頁爬取成功

```
In [2]: runfile('C:/Users/User/.spyder-py3/temp.py', wdir='C:/Users/
User/.spyder-py3')
200
```

檢查是否正確爬取網頁並輸出

若正確爬取網頁則會將網頁程式碼輸出

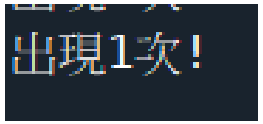
```
import requests


url = "https://www.math.thu.edu.tw/"
html = requests.get(url)
html.encoding = "utf-8"
if html.status_code == requests.codes.ok:

    print(html.text)
```

統計字串出現次數(1/3)

(可開啟記事本ch3-2，複製貼上程式碼)

```
import requests
url = "https://www.math.thu.edu.tw/"
html = requests.get(url)
html.encoding = "utf-8"
htmllist = html.text.splitlines()
print(htmllist)

n = 0
for row in htmllist:
    if "師資" in row: n+=1
    print("出現{}次!".format(n))
```

```
import requests

url = "https://www.math.thu.edu.tw/"
html = requests.get(url)
html.encoding = "utf-8"
htmllist = html.text.splitlines()

print(htmllist)

#統計個數
n = 0
for row in htmllist:
    if "師資" in row: n+=1
    print("出現{}次!".format(n))
```



統計字串出現次數(2/3)

```
import requests
url = "https://www.math.thu.edu.tw/"
html = requests.get(url)
html.encoding = "utf-8"
htmllist = html.text.splitlines()
for row in htmllist:
    print(row)

n = 0
keyword = "獎學金"
for row in htmllist:
    if keyword in row: n+=1
    print("出現{}次!".format(n))
```

```
import requests

url = "https://www.math.thu.edu.tw/"
html = requests.get(url)
html.encoding = "utf-8"
htmllist = html.text.splitlines()

#print(htmllist)

#將HTML原始碼以每一列分割成串列,並除掉跳列字元
for row in htmllist:
    print(row)

#統計個數
n = 0
keyword = "獎學金"
for row in htmllist:
    if keyword in row: n+=1
    print("出現{}次!".format(n))
```

統計字串出現次數(3/3)

互動模式(可開啟ch3-3將程式碼複製貼上)

```
pattern = input("請輸入欲搜尋的字串 : ")   # 讀取字串
    if pattern in htmlfile.text:              # 方法1
        print(f"搜尋 {pattern} 成功")
    else:
        print(f"搜尋 {pattern} 失敗")
    name = re.findall(pattern, htmlfile.text)   # 方法2
    if name:
        print(f"{pattern} 出現 {len(name)} 次")
    else:
        print(f"{pattern} 出現 0 次")
else:
    print("網頁下載失敗")
```

```
import requests
import re

url = 'http://www.mcut.edu.tw'
htmlfile = requests.get(url)
if htmlfile.status_code == requests.codes.ok:
    pattern = input("請輸入欲搜尋的字串 : ")     # 讀取字串
# 使用方法1
    if pattern in htmlfile.text:                # 方法1
        print(f"搜尋 {pattern} 成功")
    else:
        print(f"搜尋 {pattern} 失敗")
    # 使用方法2, 如果找到放在串列name內
    name = re.findall(pattern, htmlfile.text)   # 方法2
    if name:
        print(f"{pattern} 出現 {len(name)} 次")
    else:
        print(f"{pattern} 出現 0 次")
else:
    print("網頁下載失敗")
```

自訂HTTP Headers 偽裝瀏覽器操作(1/2)

有些網站會擋爬蟲程式，所以要加入 header 把程式偽裝成瀏覽器才不會被擋。以momo購物網為例

```
import requests

url = "https://www.momoshop.com.tw/main/Main.jsp"

html = requests.get(url)

html.encoding = "utf-8"

print(html.status_code)
```

```
ConnectionError: ('Connection aborted.', ConnectionResetError(10054, '遠端主機已強制關閉一個現存的連線。', None, 10054, None))
```

自訂HTTP Headers 偽裝瀏覽器操作(2/2)

```
import requests
url = "https://www.momoshop.com.tw/main/Main.jsp"
headers = {'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)'
        'AppleWebKit/537.36 (KHTML, like Gecko)'
        'Chrome/63.0.3239.132 Safari/537.36'}
html = requests.get(url)
html.encoding = "utf-8"
print(html.status_code)
```

若加入程式碼後仍被擋，表示此網站有其他的防爬機制(程式碼在ch3-4)

BeautifulSoup功能：網頁解析

BeautifulSoap模組的功能，是將讀取的網頁原始解析為一個個結構化的物件，讓程式能夠快速取得其中的內容。要先安裝 pip install beautifulsoup4

```
# 解析網頁
from bs4 import BeautifulSoup
# 建立BeautifulSoup型別物件sp, 其中「html.parser」內建的解析器
sp = BeautifulSoup(html.text, 'html.parser')
# sp.body.div.a.text
sp.a.text
```

BeautifulSoup常用的屬性

常用的屬性有

(1) 標籤名稱：傳回標籤內容

(2) text：傳回去除所有HTML標籤後的網頁文字內容

在HTML中每個標籤都是DOM結構中的結點，使用BeautifulSoup 物件.標籤名稱即可取得該節點中的內容(包含HTML標籤)，在取得的內容加上text屬性，則可去除HTML標籤，取得標籤區域內的文字。

```
# 取得屬性
print(sp.title) # 傳回title標籤內容
print(sp.title.text) # 傳回title內容
print(sp.h1)
print(sp.p)
```

```
# 使用get取得物件屬性
sp.a.get('href')
sp.a['href']
sp.img.get('src')
sp.img['src']
```

BeautifulSoup常用的方法

(1) find():尋找第一個符合條件的標籤，以字串傳回

(2) find_all():尋找所有符合條件的標籤，以串列傳回

(3) select():尋找指定CSS選擇器如id或class的內容, 以串列傳回

加入標籤屬性為搜尋條件,若有多個屬性條件, 則加到後方

find(標籤名稱, 屬性名稱=屬性內容)

find("img", width = 20)

若是屬性為class類別時，因為是保留字，所以要設為 _class=:

sp.find_all("p",class_=:'red')

```
# 發出原始資料
import requests
from bs4 import BeautifulSoup
url = "http://liangyuh.neocities.org/python/demo2.html"
html = requests.get(url)
html.encoding = "utf-8"
print(html.text)
```

```
# 拆解資料
sp = BeautifulSoup(html.text, 'html.parser')
# CSS中id編號是唯一的, 讀取時最明確
# sp.find('p', id='p2').text
# sp.find('li', class_="even").a.text
# 將資料存為串列
datas = sp.find_all('p')
for data in datas:
    print(data.text)

datas = sp.find_all('a')
for data in datas:
    print(data.get('href'))
```

以google首頁為例(1/2)

爬取google首頁的網頁內容(ch3-5)

```
from bs4 import BeautifulSoup
import requests
r = requests.get('https://www.google.com/')
if(r.status_code == requests.codes.ok):
    soup = BeautifulSoup(r.text, 'html.parser')
print(soup.prettify())
```

以google首頁為例(2/2)

★爬取google首頁的特定內容：

```
print(soup.title)
```

★若只需要標籤內的文字：

```
print(soup.title.string)
```

★爬取特定節點：

```
a_tags = soup.find_all('a')
for tag in a_tags:
  print(tag.string)
```

實例 : 爬取水庫容量

台灣水庫即時水情(程式碼 ch3-6)

```
import requests
from bs4 import BeautifulSoup

url = 'https://water.taiwanstat.com/'
web = requests.get(url)                    # 取得網頁內容
soup = BeautifulSoup(web.text, "html.parser")  # 使用 html.parser 解析器轉換成標籤樹
reservoir = soup.select('.reservoir')     # 取得所有 class 為 reservoir 的 tag
for i in reservoir:
  print(i.find('div', class_='name').get_text(), end=' ')  # 取得內容的 class 為 name 的 div 文字
  print(i.find('h5').get_text(), end=' ')   # 取得內容 h5 tag 的文字
  print()
```

實例 : google 搜尋

```
import requests
from bs4 import BeautifulSoup
google_url = 'https://www.google.com.tw/search'
my_params = {'q':'開學'}
r = requests.get(google_url, params = my_params)
if r.status_code == requests.codes.ok:
    soup = BeautifulSoup(r.text, 'html.parser')
    print(soup.prettify())
    items = soup.select('div.kCrYT > a[href^="/url"]')
    for i in items:
        print("標題:"+ i.text)
        print("網址:"+ i.get('href'))
```

實例：威力彩當期號碼

```
import requests

from bs4 import BeautifulSoup

url = "https://www.taiwanlottery.com.tw/index_new.aspx"

html = requests.get(url)

sp = BeautifulSoup(html.text,'html.parser') # 拆解格式

sp.title

datas = sp.find('div',class_="contents_box02") # 找到的第1筆

print(datas.find('div',class_="contents_mine_tx02").find('span',class_="font_black15").text)

nums = datas.find_all('div',class_='ball_tx ball_green')

# nums

print('開出順序: ')

for  i in range(0,6):

    print(nums[i].text, end=' ')

print('\n大小順序: ')

for  i in range(6,12):

    print(nums[i].text, end=' ')

print('\n特別號: ', datas.find('div',class_="ball_red").text)
```