https://www.simplilearn.com/what-is-a-web-crawler-article

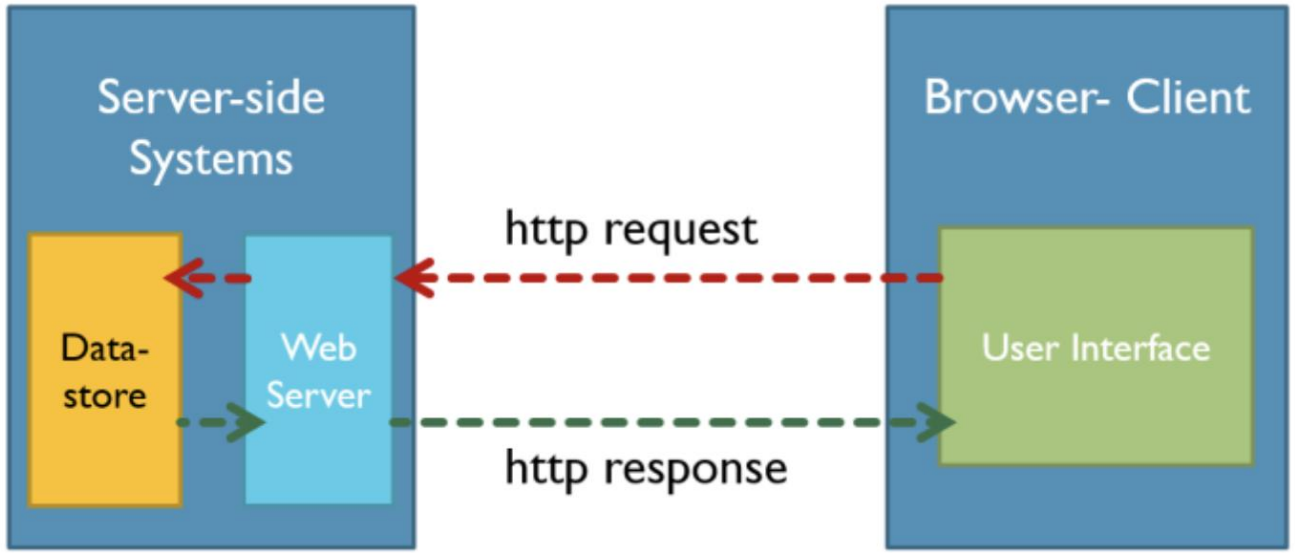
Lesson 5: Web Crawler

Advantages of Web Crawler

- Get user reviews, popular topics, trends on social networking sites
- Obtain the price of the products on the rival website, and then compare and adjust their own prices
- Get the latest movie information, reviews, and theaters
- Get hotel, flight fare money and establish comparative information
- …

Set Up a Web Crawler

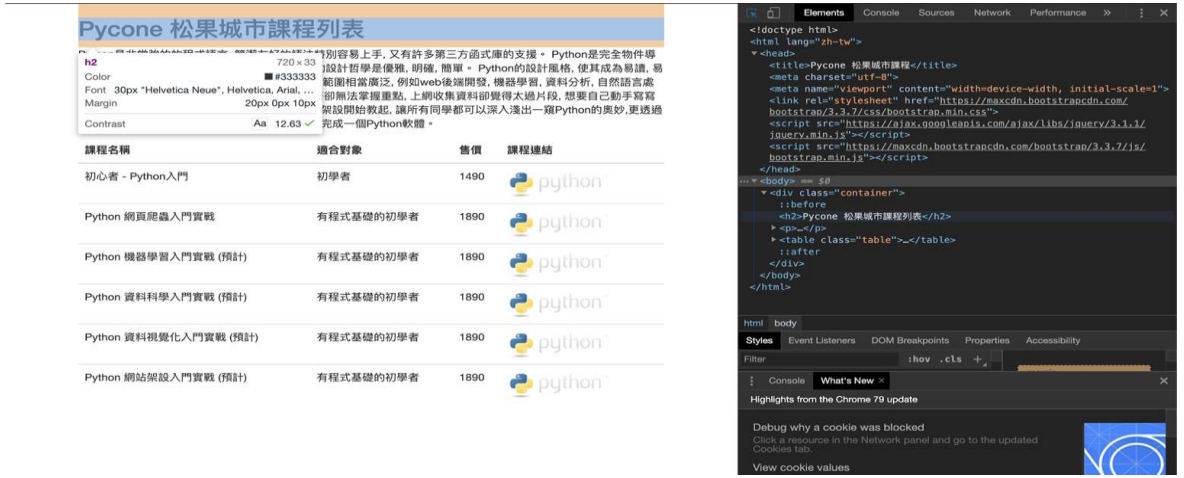
- Select URL
- Send an HTTP request packet to the destination URL to get the HTML page source code



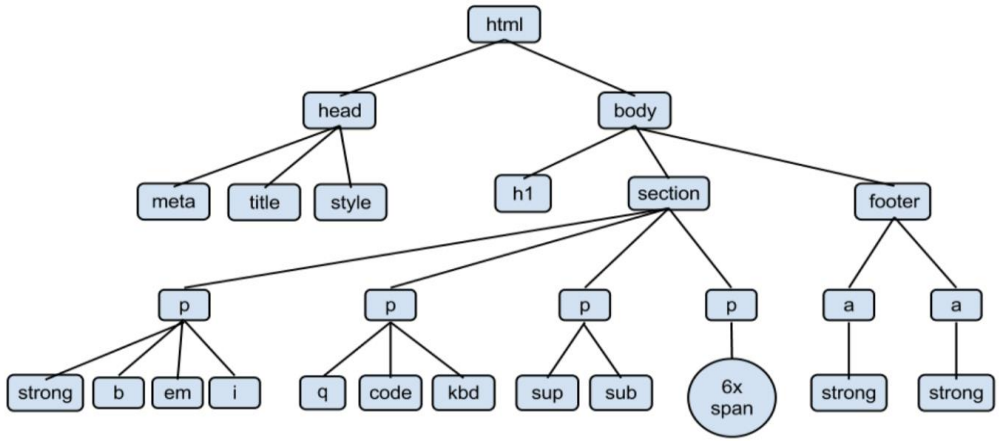
html request, and response packets

https://medium.com/@gordonfang_85054/python-%E7%88%AC%E8%9F%B2%E7%AD%86%E8%A8%98-1-15fdec38393c

- Analyze HTML pages



https://medium.com/@gordonfang_85054/python-%E7%88%AC%E8%9F%B2%E7%AD%86%E8%A8%98-1-15fdec38393c



html tree structure

https://medium.com/@gordonfang_85054/python-%E7%88%AC%E8%9F%B2%E7%AD%86%E8%A8%98-1-15fdec38393c

- Fetch data

Web Crawler Module

Requests Module

```
# 下載安裝 Python 3 的 requests 模組
pip3 install requests
# 導入 requests 模組
import requests
# 使用 GET 方法下載網頁資料
r = requests.get('https://www.google.com.tw/')
```

Beautiful Soup Module

```
# 下載安裝 Python 3 的 requests 模組
pip3 install BeautifulSoup
# 導入 BeautifulSoup 模組
from bs4 import BeautifulSoup
```

Web Crawler: Example #1

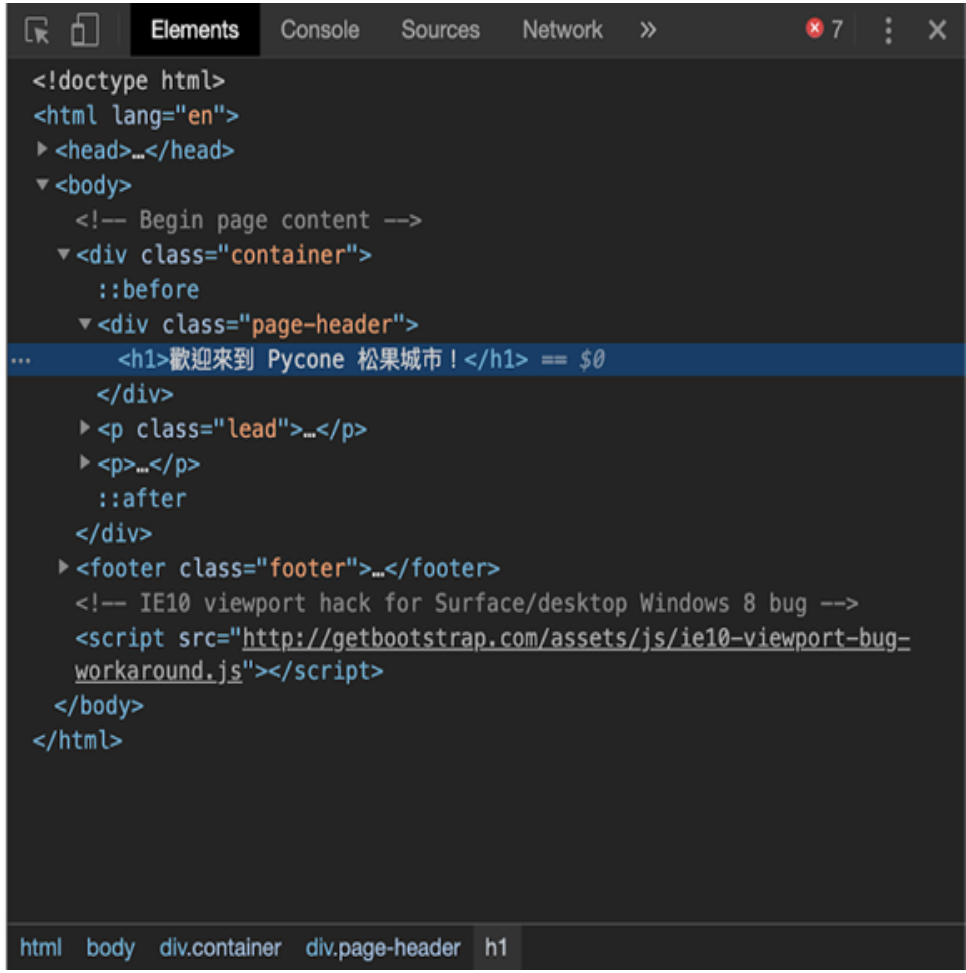
目標網址:http://blog.castman.net/web-crawler-tutorial/ch1/connect.html

歡迎來到 Pycone 松果城市！

Python是非常強的的程式語言, 簡潔友好的語法特別容易上手, 又有許多第三方函式庫的支援。Python是完全物件導向的語言, 有益於減少程式碼的重複性。Python的設計哲學是優雅, 明確, 簡單。 Python的設計風格, 使其成為易讀, 易維護且具有廣泛用途的程式語言。Python的應用範圍相當廣泛, 例如web後端開發, 機器學習, 資料分析, 自然語言處理, 網頁爬蟲與遊戲等等。如果自己常常翻閱書籍卻無法掌握重點, 上網收集資料卻覺得太過片段, 想要自己動手寫寫看卻不知道如何開始。 這們課會從最基本的環境架設開始教起, 讓所有同學都可以深入淺出一窺Python的奧妙,更透過實務專題練習的方式,使學生可以應用課堂所學來完成一個Python軟體。

了解更多

Pycone (c) 2017



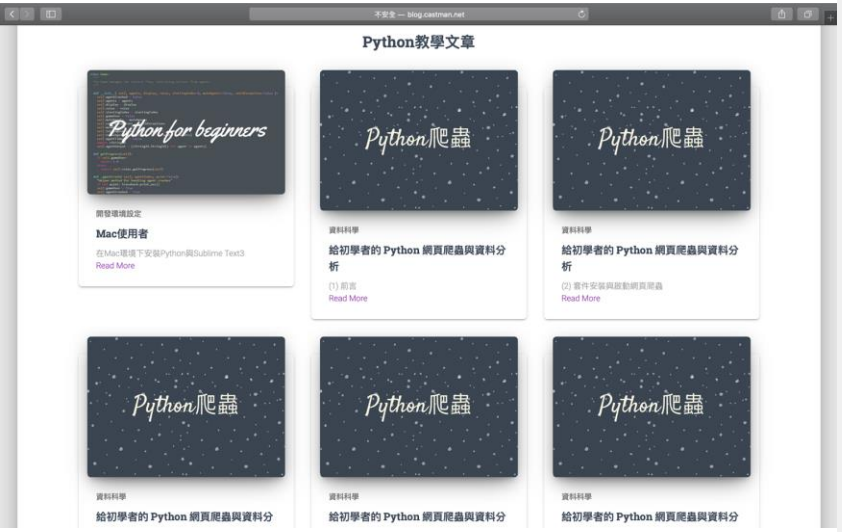
Web Crawler: Example #1

```
import requests
from bs4 import BeautifulSoup


def main():
    resp = requests.get('http://blog.castman.net/web-crawler-tutorial/ch1/connect.html')
    soup = BeautifulSoup(resp.text, 'html.parser')
    print(soup.find('h1').text)


if __name__ == '__main__':
    main()
```

歡迎來到 Pycone 松果城市！

Web Crawler: Example #2

目標網址:http://blog.castman.net/web-crawler-tutorial/ch2/blog/blog.html




```
<div class="main main-raised">
    <!-- courses introduction -->
    <div class="blogs-3">

    <div class="container">
        <div class="row">

            <div class="section">
            <h3 class="title text-center">Python教學文章</h3>
            <br>
            <div class="row">
                <div class="col-md-4">
                    <div class="card card-blog">
                        <div class="card-image">
                            <a href="http://www.pycone.com/blogs#pablo">
                                <img class="img img-raised" src="static/python-for-beginners.png">
                            </a>
                        </div>

                        <div class="content">
                            <h6 class="category text-muted">開發環境設定</h6>
                            <h4 class="card-title">
                                <a href="http://www.pycone.com/blogs#pablo">Mac使用者</a>
                            </h4>
                            <p class="card-description" id='mac-p'>
                                在Mac環境下安裝Python與Sublime Text3<a href="http://www.pycone.com/blogs/mac-python-environment" data-foo='mac-foo'> <br>Read More </a>
                            </p>

                        </div>
                    </div>
                </div>

                <div class="col-md-4">
                    <div class="card card-blog">
                        <div class="card-image">
                            <a href="http://www.pycone.com/blogs#pablo">
                                <img class="img img-raised" src="static/python_crawler.png">
                            </a>
                        </div>
                        <div class="content">
                            <h6 class="category text-muted">
                                資料科學
                            </h6>
                            <h4 class="card-title">
                                <a href="http://www.pycone.com/blogs#pablo">給初學者的 Python 網頁爬蟲與資料分析</a>
                            </h4>
                            <p class="card-description">
                                (1) 前言<a href="http://www.pycone.com/blogs/python-data-science-tutorial-1"> <br>Read More </a>
                            </p>
                        </div>
                    </div>
                </div>
```

Web Crawler: Example #2

```python
import requests
from bs4 import BeautifulSoup

def main():
    resp = requests.get('http://blog.castman.net/web-crawler-tutorial/ch2/blog/blog.html')
    soup = BeautifulSoup(resp.text, 'html.parser')

    # 取得第一篇 blog (h4)
    print(soup.find('h4'))
    print(soup.h4)  # 與上一行相等

    # 取得第一篇 blog 主標題
    print(soup.h4.a.text)

    # 取得所有 blog 主標題, 使用 tag
    main_titles = soup.find_all('h4')
    for title in main_titles:
        print(title.a.text)
if __name__ == '__main__':
    main()
```

```
<h4 class="card-title">
<a href="http://www.pycone.com/blogs#pablo">Mac使用者</a>
</h4>
<h4 class="card-title">
<a href="http://www.pycone.com/blogs#pablo">Mac使用者</a>
</h4>
Mac使用者
Mac使用者
給初學者的 Python 網頁爬蟲與資料分析
給初學者的 Python 網頁爬蟲與資料分析
給初學者的 Python 網頁爬蟲與資料分析
給初學者的 Python 網頁爬蟲與資料分析
給初學者的 Python 網頁爬蟲與資料分析
```

Web Crawler: Example #2

```python
import requests
from bs4 import BeautifulSoup

def main():
    resp = requests.get('http://blog.castman.net/web-crawler-tutorial/ch2/blog/blog.html')
    soup = BeautifulSoup(resp.text, 'html.parser')

    # 使用 key=value 取得元件
    print(soup.find(id='mac-p'))

    # 當 key 含特殊字元時, 使用 dict 取得元件
    # print(soup.find(data-foo='mac-foo'))  # 會導致 SyntaxError
    print(soup.find('', {'data-foo': 'mac-foo'}))

     # 取得各篇 blog 的所有文字
    divs = soup.find_all('div', 'content')
    for div in divs:
        # 方法一, 使用 text (會包含許多換行符號)
        #print(div.text)
        # 方法二, 使用 tag 定位
        #print(div.h6.text.strip(), div.h4.a.text.strip(), div.p.text.strip())
        # 方法三, 使用 .stripped_strings
        print([s for s in div.stripped_strings])
if __name__ == '__main__':
    main()
```

```
<p class="card-description" id="mac-p">
                                在Mac環境下安裝Python與Sublime Text3<a data-foo="mac-foo" href="http://www.pycone.com/blogs/mac-python-environ
ment"> <br/>Read More </a>
</p>
<a data-foo="mac-foo" href="http://www.pycone.com/blogs/mac-python-environment"> <br/>Read More </a>
['開發環境設定', 'Mac使用者', '在Mac環境下安裝Python與Sublime Text3', 'Read More']
['資料科學', '給初學者的 Python 網頁爬蟲與資料分析', '(1) 前言', 'Read More']
['資料科學', '給初學者的 Python 網頁爬蟲與資料分析', '(2) 套件安裝與啟動網頁爬蟲', 'Read More']
['資料科學', '給初學者的 Python 網頁爬蟲與資料分析', '(3) 解構並擷取網頁資料', 'Read More']
```

Web Crawler: Example #3

目標網址:http://blog.castman.net/web-crawler-tutorial/ch2/table/table.html

Pycone 松果城市課程列表

Python是非常強的的程式語言，簡潔友好的語法特別容易上手，又有許多第三方函式庫的支援。 Python是完全物件導向的語言，有益於減少程式碼的重複性。Python的設計哲學是優雅，明確，簡單。 Python的設計風格，使其成為易讀，易維護且具有廣泛用途的程式語言。 Python的應用範圍相當廣泛，例如web後端開發，機器學習，資料分析，自然語言處理，網頁爬蟲與遊戲等等。如果自己常常翻閱書籍卻無法掌握重點，上網收集資料卻覺得太過片段，想要自己動手寫寫看卻不知道如何開始。 這們課會從最基本的環境架設開始教起，讓所有同學都可以深入淺出一窺Python的奧妙,更透過實務專題練習的方式,使學生可以應用課堂所學來完成一個Python軟體。

課程名稱	適合對象	售價	課程連結
初心者 - Python入門	初學者	1490	python
Python 網頁爬蟲入門實戰	有程式基礎的初學者	1890	python
Python 機器學習入門實戰 (預計)	有程式基礎的初學者	1890	python
Python 資料科學入門實戰 (預計)	有程式基礎的初學者	1890	python
Python 資料視覺化入門實戰 (預計)	有程式基礎的初學者	1890	python
Python 網站架設入門實戰 (預計)	有程式基礎的初學者	1890	python

```
<!DOCTYPE html>
<html lang="zh-tw">
<head>
  <title>Pycone 松果城市課程</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
  <script src="https://ajax.googleapis.com/ajax/libs/jquery/3.1.1/jquery.min.js"></script>
  <script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>
</head>
<body>

<div class="container">
  <h2>Pycone 松果城市課程列表</h2>
  <p>Python是非常強的的程式語言，簡潔友好的語法特別容易上手，又有許多第三方函式庫的支援。 Python是完全物件導向的語言，有益於減少程式碼的重複性。Python的設計哲學是優雅，明確，簡單。 Python的設計風格，使其成為易讀，易維護且具有廣泛用途的程式語言。 Python的應用範圍相當廣泛，例如web後端開發，機器學習，資料分析，自然語言處理，網頁爬蟲與遊戲等等。如果自己常常翻閱書籍卻無法掌握重點，上網收集資料卻覺得太過片段，想要自己動手寫寫看卻不知道如何開始。 這們課會從最基本的環境架設開始教起，讓所有同學都可以深入淺出一窺Python的奧妙,更透過實務專題練習的方式,使學生可以應用課堂所學來完成一個Python軟體。</p>
  <table class="table">
    <thead>
      <tr><th>課程名稱</th><th>適合對象</th><th>售價</th><th>課程連結</th></tr>
    </thead>
    <tbody>
      <tr><td>初心者 - Python入門</td><td>初學者</td><td>1490</td><td><a href="http://www.pycone.com"><img class="python-logo" src="img/python-logo.png" alt="python™"></a></td></tr>
      <tr><td>Python 網頁爬蟲入門實戰</td><td>有程式基礎的初學者</td><td>1890</td><td><a href="http://www.pycone.com"><img class="python-logo" src="img/python-logo.png" alt="python™">
</a></td></tr>
      <tr><td>Python 機器學習入門實戰 (預計)</td><td>有程式基礎的初學者</td><td>1890</td><td><a href="http://www.pycone.com"><img class="python-logo" src="img/python-logo.png"
alt="python™"></a></td></tr>
      <tr><td>Python 資料科學入門實戰 (預計)</td><td>有程式基礎的初學者</td><td>1890</td><td><a href="http://www.pycone.com"><img class="python-logo" src="img/python-logo.png"
alt="python™"></a></td></tr>
      <tr><td>Python 資料視覺化入門實戰 (預計)</td><td>有程式基礎的初學者</td><td>1890</td><td><a href="http://www.pycone.com"><img class="python-logo" src="img/python-logo.png"
alt="python™"></a></td></tr>
      <tr><td>Python 網站架設入門實戰 (預計)</td><td>有程式基礎的初學者</td><td>1890</td><td><a><img class="python-logo" src="img/python-logo.png" alt="python™"></a></td></tr>
    </tbody>
  </table>
</div>

</body>
</html>
```

Web Crawler: Example #3

```
import requests
from bs4 import BeautifulSoup

def main():
    resp = requests.get('http://blog.castman.net/web-crawler-tutorial/ch2/table/table.html')
    soup = BeautifulSoup(resp.text, 'html.parser')

    # 計算課程均價
    # 取得所有課程價錢: 方法一, 使用 index
    prices = []
    rows = soup.find('table', 'table').tbody.find_all('tr')
    for row in rows:
        price = row.find_all('td')[2].text
        prices.append(int(price))
    print(sum(prices)/len(prices))

    # 取得所有課程價錢: 方法二, <a> 的 parent (<td>) 的 previous_sibling
    prices = []
    links = soup.find_all('a')
    for link in links:
        price = link.parent.previous_sibling.text
        prices.append(int(price))
    print(sum(prices) / len(prices))

if __name__ == '__main__':
    main()
```

```
1823.3333333333333
1823.3333333333333
```

Web Crawler: Example #3

```python
import requests
from bs4 import BeautifulSoup

def main():
    resp = requests.get('http://blog.castman.net/web-crawler-tutorial/ch2/table/table.html')
    soup = BeautifulSoup(resp.text, 'html.parser')

    # 取得每一列所有欄位資訊: find_all('td') or row.children
    rows = soup.find('table', 'table').tbody.find_all('tr')
    for row in rows:
        all_tds = row.find_all('td')
        # 以下執行時會報錯, 因為最後一列的 <a> 沒有 'href' 屬性
        # print(all_tds[0].text, all_tds[1].text, all_tds[2].text, all_tds[3].a['href'], all_tds[3].a.img['src'])
        # 取得 href 屬性前先檢查其是否存在
        if 'href' in all_tds[3].a.attrs:
            href = all_tds[3].a['href']
        else:
            href = None
        print(all_tds[0].text, all_tds[1].text, all_tds[2].text, href, all_tds[3].a.img['src'])

        # 取得每一列所有欄位文字資訊: stripped_strings
    rows = soup.find('table', 'table').tbody.find_all('tr')
    for row in rows:
        print([s for s in row.stripped_strings])

if __name__ == '__main__':
    main()
```

```
初心者 – Python入門 初學者 1490 http://www.pycone.com img/python-logo.png
Python 網頁爬蟲入門實戰 有程式基礎的初學者 1890 http://www.pycone.com img/python-logo.png
Python 機器學習入門實戰 (預計) 有程式基礎的初學者 1890 http://www.pycone.com img/python-logo.png
Python 資料科學入門實戰 (預計) 有程式基礎的初學者 1890 http://www.pycone.com img/python-logo.png
Python 資料視覺化入門實戰 (預計) 有程式基礎的初學者 1890 http://www.pycone.com img/python-logo.png
Python 網站架設入門實戰 (預計) 有程式基礎的初學者 1890 None img/python-logo.png
['初心者 – Python入門', '初學者', '1490']
['Python 網頁爬蟲入門實戰', '有程式基礎的初學者', '1890']
['Python 機器學習入門實戰 (預計)', '有程式基礎的初學者', '1890']
['Python 資料科學入門實戰 (預計)', '有程式基礎的初學者', '1890']
['Python 資料視覺化入門實戰 (預計)', '有程式基礎的初學者', '1890']
['Python 網站架設入門實戰 (預計)', '有程式基礎的初學者', '1890']
```


Q&A