

Lecture 1: Introduction to Crawler

Speaker: Hong-Han Shuai
ECE, NCTU

Introduction

- **The key component for the crawling here is.....**

Python

- **How to learn**
 - 1. <https://www.python.org/doc/>
 - 2. Google “python tutorial”
 - 3. Python 100 days: <https://github.com/jackfrued/Python-100-Days> (131k stars)

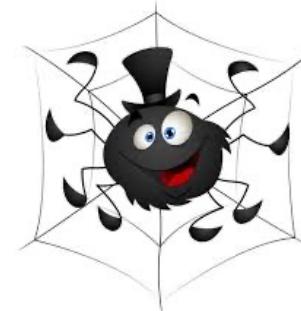
When you encounter problem

- Google
- Ask other people to google for you(?)
- Stackoverflow
- TA hour

Crawler

What is crawler?

- Also called “Spider”
- 透過程式去追蹤網頁上的超連結，然後不斷往外擴張，以便將全世界中曾經被連結到的網頁全部都抓回到來，這也是 Google、Yahoo 等網站背後最重要的程式之一



What is crawler?

- 送出 HTTP Request
- 使用正規表示法(Regular Expression) 解析 Response 接收到的網頁原始碼中的資訊

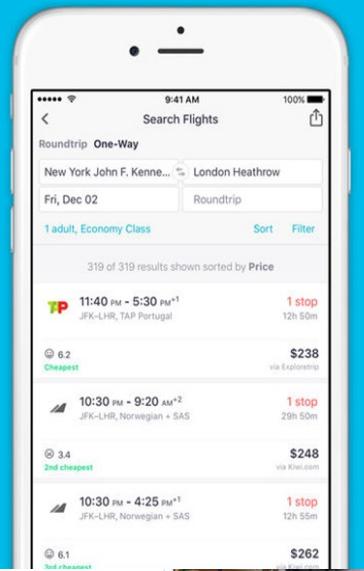


我們的最終目的是...

把別人的資料庫都變成
我家的資料庫！



Find and compare the best flights.



韓國 Ryoé 呂 漢方洗髮精 180ml 【櫻桃飾品】 [24122]
\$ 49 🇹🇼 樂天市場 - 櫻桃飾品 - 2018/11/23
◎商品比價 🔍 相關資訊 ☆加入追蹤 消費額賺 10% 點數



韓國 Ryoé 呂 漢方洗髮精 180ml 【櫻桃飾品】 [24122]
\$ 49 🇹🇼 Yahoo奇摩超級商城 - 櫻桃飾品 - 2018/11/23
◎商品比價 🔍 相關資訊 ☆加入追蹤



韓國 Ryoé 呂 漢方洗髮精 180ml 兩款可選 【櫻桃飾品】 [24122]
\$ 49 🇹🇼 momo摩天商城 - 櫻桃飾品 - 2018/11/22
◎商品比價 🔍 相關資訊 ☆加入追蹤



韓國 Ryoé 呂 漢方洗髮精 180ml 【櫻桃飾品】 [24122]
\$ 49 🇹🇼 momo摩天商城 - 櫻桃飾品 - 2018/11/22
◎商品比價 🔍 相關資訊 ☆加入追蹤



韓國 Ryoé 呂 漢方洗髮精 180ml 兩款可選 【櫻桃飾品】 [24121]
\$ 65 🇹🇼 momo摩天商城 - 櫻桃飾品 - 2018/11/22
◎商品比價 🔍 相關資訊 ☆加入追蹤



韓國 Ryoé 呂 漢方洗髮精 180ml 【櫻桃飾品】 [24121]
\$ 65 🇹🇼 momo摩天商城 - 櫻桃飾品 - 2018/11/22
◎商品比價 🔍 相關資訊 ☆加入追蹤



細心 - 設計師一對一服務
柔靜、愜意, 着你自在的生活。擁用更美好的自己。
改 - 你開始。

沐 / 潤 聽解詳情 >

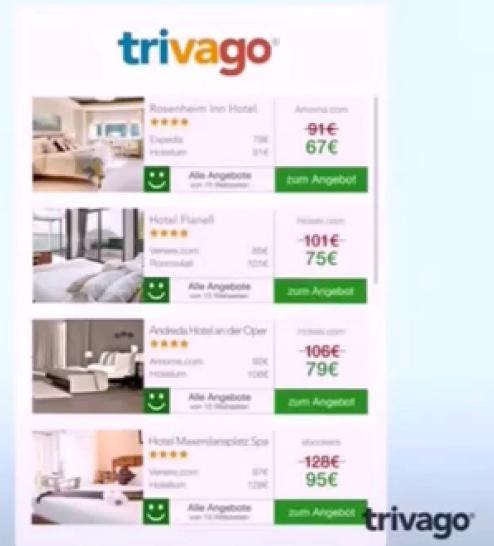
myfone購物 - 今日



\$ 799 前往
暖風量 冷暖熱三
CP 價極高的泡湯
2張組↘【北投
大眾風呂+自助

剛剛有人找

\$ 31,920
Acer SF514-52T-51AA 14吋筆電(i5-
8250U/512G/金/福



網路溫度計





SEARCH _____



FAUX LEATHER MINI SKIRT

NT\$ 990

BLACK - 3046/280

Short A-line skirt featuring front patch pockets with flaps and snap buttons. Zip and button fastening at the front.

HEIGHT OF MODEL: 177 cm. / 69.6"

XS

S

M

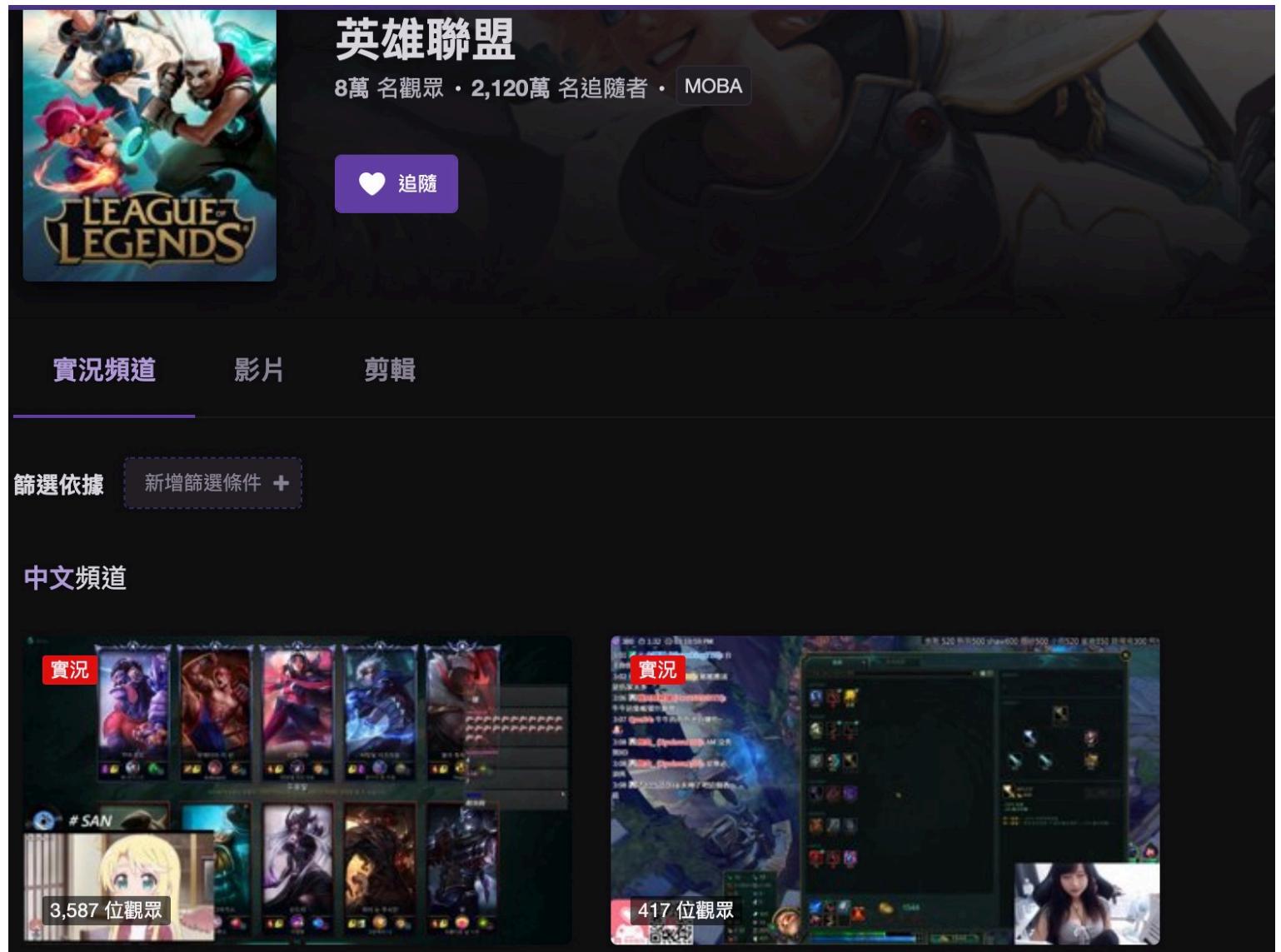
L

✉ Coming Soon

XL

✉ We'll let you know when it's in stock

Twitch



The image shows a Twitch channel page for the game League of Legends. At the top, there's a banner featuring the League of Legends logo and two champions. Below the banner, the title "英雄聯盟" (League of Legends) is displayed, along with statistics: 8萬 名觀眾 (80,000 viewers) and 2,120 萬 名追隨者 (21.2 million followers), categorized as MOBA. A purple "追隨" (Follow) button is visible. Below the banner, there are three tabs: 實況頻道 (Streaming Channels), 影片 (Videos), and 剪輯 (Clips). A "篩選依據" (Filter by) section allows users to add filtering conditions. The main content area is titled "中文頻道" (Chinese Channels) and shows two streaming thumbnails. The left thumbnail, labeled "實況" (Streaming), has 3,587 位觀眾 (3,587 viewers) and features a character from the game. The right thumbnail, also labeled "實況", has 417 位觀眾 (417 viewers) and features a female streamer. Both thumbnails show the League of Legends interface.

英雄聯盟

8萬 名觀眾 · 2,120 萬 名追隨者 · MOBA

追隨

實況頻道 影片 剪輯

篩選依據 新增篩選條件 +

中文頻道

實況 3,587 位觀眾

實況 417 位觀眾

How to do?

- 參考別人寫好的 API (Github search)

The screenshot shows the GitHub search interface with the query "PTT crawler" entered in the search bar. The results page displays 95 repository findings. The top result is "zbryikt/ptt - crawler", which is a LiveScript project that crawls PTT articles. The second result is "wy36101299/PTTcrawler", a Python project for web crawling PTT Gossiping. The third result is "jwlin/ptt - web - crawler", a Python project for crawling PTT 網路版爬蟲.

Search: PTT crawler

We've found 95 repository results

Repositories	95
Code	1,557
Issues	14
Wikis	1
Users	

Languages

Python	53
JavaScript	9
Ruby	6
Java	4
Perl	3
Clojure	2
Go	2
HTML	2

zbryikt/ptt - crawler
crawl ptt articles from its website
Updated on 10 May

LiveScript ★ 36 ⚡ 15

wy36101299/PTTcrawler
crawler for web PTT Gossiping
Updated on 5 Apr

Python ★ 21 ⚡ 17

jwlin/ptt - web - crawler
PTT 網路版爬蟲
Updated 9 hours ago

Python ★ 43 ⚡ 28

ptt-web-crawler (PTT 網路版爬蟲) build passing

Live demo

特色

- 支援單篇及多篇文章抓取
- 過濾資料內空白、空行及特殊字元
- JSON 格式輸出
- 支援 Python 2.7 - 3.4

輸出 JSON 格式

```
{
    "article_id": 文章 ID,
    "article_title": 文章標題 ,
    "author": 作者,
    "board": 板名,
    "content": 文章內容,
    "date": 發文時間,
    "ip": 發文位址,
    "message_conut": { # 推文
        "all": 總數,
        "boo": 嘘文數,
        "count": 推文數-噓文數,
        "neutral": → 數,
        "push": 推文數
    },
    "messages": 「 # 推文內容
```

ptt-web-crawler (PTT 網路版爬蟲) build passing

Live demo

特色

- 支援單篇及多篇文章抓取
- 過濾資料內空白、空行及特殊字元
- JSON 格式輸出
- 支援 Python 3

輸出 JSON 格式

執行方式

```
python crawler.py -b 看板名稱 -i 起始索引 結束索引 (設為 -1 則自動計算最後一頁)  
python crawler.py -b 看板名稱 -a 文章ID
```

```
{
    "article": [
        {
            "author": "作者名稱",
            "content": "文章內容",
            "date": "發文時間",
            "ip": "發文IP",
            "message_conut": { # 推文
                "all": 總數,
                "boo": 嘘文數,
                "count": 推文數-噓文數,
                "neutral": → 數,
                "push": 推文數
            },
            "messages": [ # 推文內容
                {
                    "content": "推文內容",
                    "date": "回文時間",
                    "ip": "回文IP",
                    "message_id": "回文ID"
                }
            ]
        }
    ],
    "board": "看板名稱"
}
```

範例

```
python crawler.py -b PublicServant -i 100 200
```

How to do?

- 使用別人寫好的API (Twitter API) <https://apps.twitter.com>

The screenshot shows the Twitter Developer website's 'Apps' section. At the top, there is a purple navigation bar with links for Developer, Use cases, Solutions, Products, Docs, Community, Updates, Support, Apply, Apps, and a user profile icon. A prominent message banner in the center says '#welcome We have sunset apps.twitter.com. You can manage any of your existing Apps in all of the same ways through this site.' with a small emoji icon. Below the banner, the 'Apps' tab is selected, indicated by a blue underline. To the right of the tab is a 'Create an App' button. The main content area below the tabs displays the message: 'No Apps here. You'll need an App and API key in order to authenticate and integrate with most Twitter developer products. Create an App to get your API key.'

Apps

Create an App

No Apps here.

You'll need an App and API key in order to authenticate and integrate with most Twitter developer products. Create an App to get your API key.



#ApplicationReceived

We received your application. We'll let you know when it's done, or if we need any additional information from you by sending an email to ia*****@gm****.com.

In the meantime, you can get a head start by learning about the Twitter API by browsing our [docs](#), [tutorials](#), and [community forums](#).



Create an application

第二步：填寫必填欄位

Application Details (僅供參考)

Name *

youngmihuangBot

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description *

testing for NLP

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

Website *

<https://medium.com/@cyeninesky3>

Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens.
(If you don't have a URL yet, just put a placeholder here but remember to change it later.)

Callback URLs

Where should we return after successfully authenticating? OAuth 1.0a applications must explicitly specify their oauth_callback URL(s) here, as well as include the one of the URLs below in the request token step. To restrict your application from using callbacks, leave this field blank.

Add a Callback URL

Developer Agreement

Yes, I have read and agree to the Twitter Developer Agreement.

Create your Twitter application

youngmihuangBot

[Test OAuth](#)

[Details](#) [Settings](#) [Keys and Access Tokens](#) [Permissions](#)

Application Settings

Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.

Consumer Key (API Key)	YOUR CONSUMER KEY
Consumer Secret (API Secret)	YOUR CONSUMER SECRET

Access Level [Read and write \(modify app permissions\)](#)

Owner

Owner ID

Application Actions

[Regenerate Consumer Key and Secret](#) [Change App Permissions](#)

Your Access Token

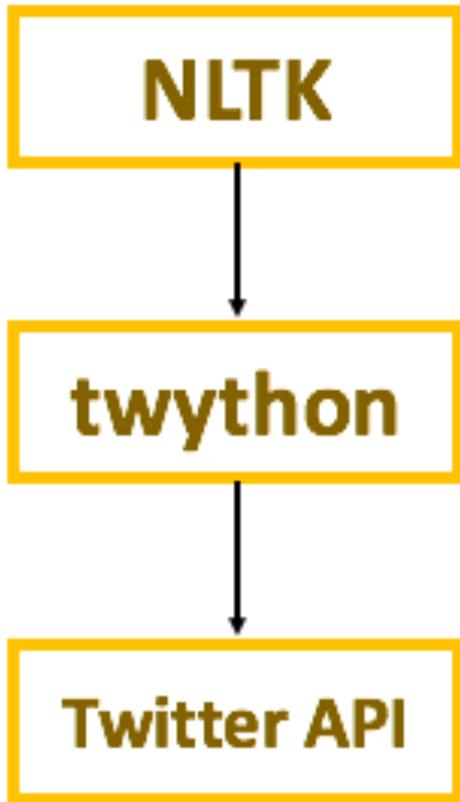
This access token can be used to make API requests on your own account's behalf. Do not share your access token secret with anyone.

Access Token	YOUR ACCESS TOKEN
Access Token Secret	YOUR ACCESS TOKEN SECRET

Access Level [Read and write](#)

Owner

Owner ID



The NLTK Twitter package
relies on twython.

Actively maintained,
pure Python wrapper for the Twitter API.

(Search & Streaming API)

`pip install twython`

Example

搜尋 tweets 可以包含多個字詞 (逗號代表: 'or' 的意義)

```
from nltk.twitter import Twitter
```

```
tw = Twitter()
```

```
tw.tweets(keywords='love, hate', limit=10) # 只取 10 筆資料
```

Result (以 RT 開頭分隔不同則 tweets)

RT @CollinRugg: People are speculating that Elizabeth Warren might take on Trump in 2020.RT @Harry_Styles: Wow, Eight years has passed. Thank you for all the love, thank you for all the support. Thank you for everything.

I love...

RT @lilbaked: i love drinking alcohol i didn't pay for

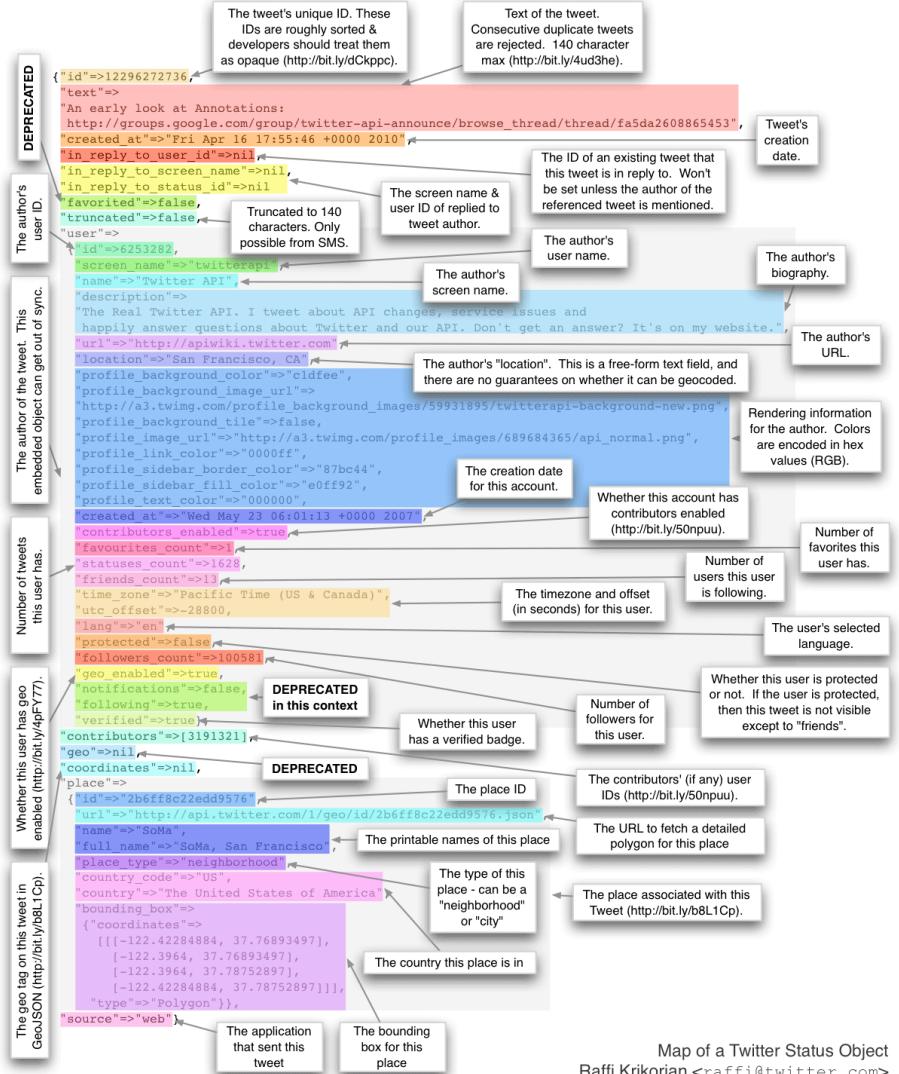
RT @louisvlifestyle: Hate when someone doesn't keep their word

RT @_18RIMA: this is the cutest vid i've ever seen oh my fucking good i love troye so fucking much <https://t.co/DuKLahuzB9>

...

Query

```
client = Query(**oauth) # 歷史資料
tweets = client.search_tweets(keywords='machine learning', limit=10)
tweet = next(tweets) # 取資料
from pprint import pprint
pprint(tweet, depth=1)
```



Map of a Twitter Status Object
Raffi Krikorian <raffi@twitter.com>
18 April 2010

How to do?

爬取網頁原始碼

- Requests
- Scrapy

解析網頁原始碼

- 正規表達式 (Regular Expression)
- BeautifulSoup
- XPath



Requests

Requests

- 安裝

pip install requests

- 使用

content 為網頁原始碼

```
1 import requests
2 url = ""
3 r = requests.get(url)
4 content = r.text
```

http://docs.python-requests.org/zh_CN/latest/user/quickstart.html

```
In [1]: import requests  
url = "https://www.ptt.cc/index.html"  
r = requests.get(url)  
print r.text
```

```
<!DOCTYPE html>
<html id="Stencil" lang="zh-Hant-TW" class="StencilRoot my3columns l-out Pos-r https fp fp-default ltr desktop Desktop bkt900">
<head>

    <title>Yahoo奇摩</title><meta http-equiv="x-dns-prefetch-control" content="on"><link rel="dns-prefetch" href="//s.yimg.com"><link rel="preconnect" href="//s.yimg.com"><link rel="dns-prefetch" href="//y.analytics.yahoo.com"><link rel="preconnect" href="//y.analytics.yahoo.com"><link rel="dns-prefetch" href="//geo.query.yahoo.com"><link rel="preconnect" href="//geo.query.yahoo.com"><link rel="dns-prefetch" href="//csc.beap.bc.yahoo.com"><link rel="preconnect" href="//csc.beap.bc.yahoo.com"><link rel="dns-prefetch" href="//geo.yahoo.com"><link rel="preconnect" href="//geo.yahoo.com"><link rel="dns-prefetch" href="//comet.yahoo.com"><link rel="preconnect" href="//comet.yahoo.com"><link rel="dns-prefetch" href="//video-api.yql.yahoo.com"><link rel="preconnect" href="//video-api.yql.yahoo.com"><link rel="dns-prefetch" href="//yrtas.btrll.com"><link rel="preconnect" href="//yrtas.btrll.com"><link rel="dns-prefetch" href="//shim.btrll.com"><link rel="preconnect" href="//shim.btrll.com">    <meta http-equiv="Content-Type" content="text/html; charset=utf-8">

    <meta http-equiv="X-UA-Compatible" content="chrome=1">
    <meta name="description" content="最新Yahoo奇摩首頁，提供最方便的網站搜尋、即時新聞、生活資訊和Yahoo奇摩服務入口"
">
```

Requests

- 其他Http請求

```
>>> r = requests.put("http://httpbin.org/put")
>>> r = requests.delete("http://httpbin.org/delete")
>>> r = requests.head("http://httpbin.org/get")
>>> r = requests.options("http://httpbin.org/get")
```

- 傳遞 URL 參數

```
>>> payload = {'key1': 'value1', 'key2': 'value2'}
>>> r = requests.get("http://httpbin.org/get", params=payload)
>>> print(r.url)
http://httpbin.org/get?key2=value2&key1=value1
```



Scrapy

An open source and collaborative framework
for extracting the data you need from websites.
In a fast, simple, yet extensible way.

Scrapy

- 安裝

```
pip install scrapy
```

- 介紹

Scrapy是一個為了爬取網站數據，提取結構性數據而編寫的應用框架。可以應用於包括數據挖掘，信息處理或存儲歷史數據等一系列的程序中。

<https://doc.scrapy.org/en/latest/intro/install.html>

Scrapy

- 使用

在開始爬取之前，必須先建立一個Scrapy項目。

\$ scrapy startproject tutorial

此命令會建立包含下列內容的 tutorial 目錄:

```
tutorial/
    scrapy.cfg
    tutorial/
        __init__.py      -----> 設定程式開始和結尾動作
        items.py
        pipelines.py
        settings.py
        spiders/         -----> 在此層放入爬蟲程式
            __init__.py
            ...
            ...
```

Scrapy

```
import scrapy

class DmozSpider(scrapy.Spider):
    name = "dmoz"
    allowed_domains = ["dmoz.org"]
    start_urls = [
        "http://www.dmoz.org/Computers/Programming/Languages/Python/Books/",
        "http://www.dmoz.org/Computers/Programming/Languages/Python/Resources/"
    ]

    def parse(self, response):
        filename = response.url.split("/")[-2]
        with open(filename, 'wb') as f:
            f.write(response.body)
```

Scrapy

- 執行

在 tutorial 根目錄下執行

\$ scrapy crawl [crawl name]

```
2014-01-23 18:13:07-0400 [scrapy] INFO: Scrapy started (bot: tutorial)
2014-01-23 18:13:07-0400 [scrapy] INFO: Optional features available: ...
2014-01-23 18:13:07-0400 [scrapy] INFO: Overridden settings: {}
2014-01-23 18:13:07-0400 [scrapy] INFO: Enabled extensions: ...
2014-01-23 18:13:07-0400 [scrapy] INFO: Enabled downloader middlewares: ...
2014-01-23 18:13:07-0400 [scrapy] INFO: Enabled spider middlewares: ...
2014-01-23 18:13:07-0400 [scrapy] INFO: Enabled item pipelines: ...
2014-01-23 18:13:07-0400 [dmoz] INFO: Spider opened
2014-01-23 18:13:08-0400 [dmoz] DEBUG: Crawled (200) <GET http://www.dmoz.org/Computers/Progra...
2014-01-23 18:13:09-0400 [dmoz] DEBUG: Crawled (200) <GET http://www.dmoz.org/Computers/Progra...
2014-01-23 18:13:09-0400 [dmoz] INFO: Closing spider (finished)
```

Regular Expression

Regular expression

- 又稱正則表達式、正規表示法、正規運算式、規則運算式、常規表示法
- 使用單個字串來描述、符合一系列符合某個句法規則的字串。在很多文字編輯器裡，正則運算式通常被用來檢索、取代那些符合某個模式的文字。
- 簡單來說，就是一種字串的篩選器。

Let's try an example

- Find the email from the document!

[A-Za-z0-9._]+@[A-Za-z.]+\.(com|edu)+\.\tw

Python 的 re package

```
import re
pat = '[a-zA-Z]+'
text = 'Hello, hm...this is Tom speaking, who are you?'
r = re.findall(pat, text)
print r

['Hello', 'hm', 'this', 'is', 'Tom', 'speaking', 'who', 'are', 'you']
```

字元	描述
[xyz]	字元集合。匹配所包含的任意一個字元
[^xyz]	不含字元集合。
[a-z]	字元範圍。
+	匹配前面的子運算式一次或多次
*	匹配前面的子運算式零次或多次

Example

```
import re
pattern = "l+o*ve"
test_string = 'love, loooove, dove, leave, lve'
result = re.findall(pattern, test_string)
print(result)
```

`['love', 'loooove', 'lve']`

Example

```
import re
pattern = "[cmf]+an"
test_string = 'can, man, fan, damn, fffffan, fcman'
result = re.findall(pattern, test_string)
print(result)
```

```
['can', 'man', 'fan', 'fffffan', 'fcman']
```

Regular expression

字元	描述
\d	匹配一個數位字元。等價於[0-9]。
\D	匹配一個非數位字元。等價於[^0-9]。
\w	匹配包括底線的任何單詞字元。等價於 “[A-Za-z0-9_]”。
\W	匹配任何非單詞字元。等價於 “[^A-Za-z0-9_]”。
{n}	比對前一個字元 n 次，n 為一個正整數。例如： /a{3}/ 可比對 “lllaaala” 其中的 “aaa”，但不可比對 “aa”
(n,m)	至少出現n次,至多出現m次
\	避開特殊字元。
.	代表任意字元(符號、空格、數字)
x y	匹配x或y。例如，“z food”能匹配“z”或“food”

Example

```
import re
pattern = "修.{2,3}"
test_string = '修身，修罵長，修理紗窗紗門換玻璃，\
老不修，修嘅就鬼，修各樓，修但幾勒'
result = re.findall(pattern, test_string)
print(result)

['修身, ', '修罵長, ', '修理紗窗', '修, 修', '修各樓, ', '修但幾勒']
```

Let's try an example

- Find the email from the document!

[A-Za-z0-9._]+@[A-Za-z.]+\.(com|edu)+\.\tw



密碼驗證

- 至少有一個數字
- 至少有一個小寫英文字母
- 至少有一個大寫英文字母
- 字串長度在 6 ~ 20 個字母之間

```
In [116]: import re
pattern = "^(?=.*\d)(?=.*[a-zA-Z]).{6,20}$"
password = raw_input("Enter string to test: ")
result = re.findall(pattern, password)
print result
if (result):
    print "Valid password"
else:
    print "Password not valid"
```

```
Enter string to test: sd1Ads5
['sd1Ads5']
Valid password
```

- $(?=.*\d)$: 這是 Positive Lookahead , 用來判斷右邊緊接著的字元是否符合比對條件 , 如果符合條件才會繼續比對下去。拿這個實例來說 , 右邊的字元必須包含一個數字才算符合這個條件。
- ^ : 表示字首 , 意思就是說後面的字必須是從字首開始。
- \$: 表示字尾 , 表示前面的字要是字尾。

Regular expression test

- 試著找出這些要求的pattern

- 找出有小數點的數字

- text = “87.87 % people”

- 找出email

- text = “email:1234@yahoo.com.tw,email:4567@gmail.com”

- 找出URL

- text = “奇摩網頁”

Regular expression techniques

- 找出有小數點的數字
- 找出email
- 找出URL

```
import re
pat = '[0-9]+\.[0-9]+'
text = '87.87% people'
r = re.findall(pat, text)
print r
```

['87.87']

```
import re
pat = '[a-zA-Z0-9_]+@[a-zA-Z0-9\._]+'
text = 'email:1234@yahoo.com.tw,email:4567@gmail.com'
r = re.findall(pat, text)
print r
```

['1234@yahoo.com.tw', '4567@gmail.com']

```
import re
pat = 'http://[a-zA-Z0-9\._]+'
text = '<a href = "http://yahoo.com.tw">奇摩網頁</a>'
r = re.findall(pat, text)
print r
```

['http://yahoo.com.tw']

BeautifulSoup



Introduction

- 他是一個可以從HTML或XML文件中提取數據的Python package，提供一些簡單的函數來解析網頁結構。
- BeautifulSoup 將HTML變成一個複雜的樹狀結構，所有結構可以分為四對象：
 - Tag
 - NavigableString
 - BeautifulSoup
 - Comment

http://beautifulsoup.readthedocs.io/zh_CN/v4.4.0/

What is tag ?

- 簡單來講就是HTML中的一個個標籤
- 例如：

```
<title>Yahoo奇摩</title>
```

```
<div id="darla-assets-top"> </div>
```

```
<a href="https://tw.news.yahoo.com/weather/">氣象 </a>
```

```
<p class="MouseOver-TextDecoration My-0 Cur-p Ov-h E11">才開季一天 完美罰球出現了 </p>
```

- BeautifulSoup 可以快速讓我們獲取tag 資訊

BeautifulSoup

- 安裝

```
pip install beautifulsoup4
```

- 使用

抓出第一個< p ></ p >裡的資訊

```
In [1]: from bs4 import BeautifulSoup
import requests
url = "http://yahoo.com.tw"
r = requests.get(url)
content = r.text

soup = BeautifulSoup(content, 'html.parser')
print soup.p
```

```
<p class="MouseOver-TextDecoration My-0 Cur-p Ov-h E11">花媽公開指責 他不敢擺攤了</p>
```

BeautifulSoup

- 幾個簡單的瀏覽結構化數據的方法：

```
In [15]: print soup.title  
<title>Yahoo奇摩</title>
```

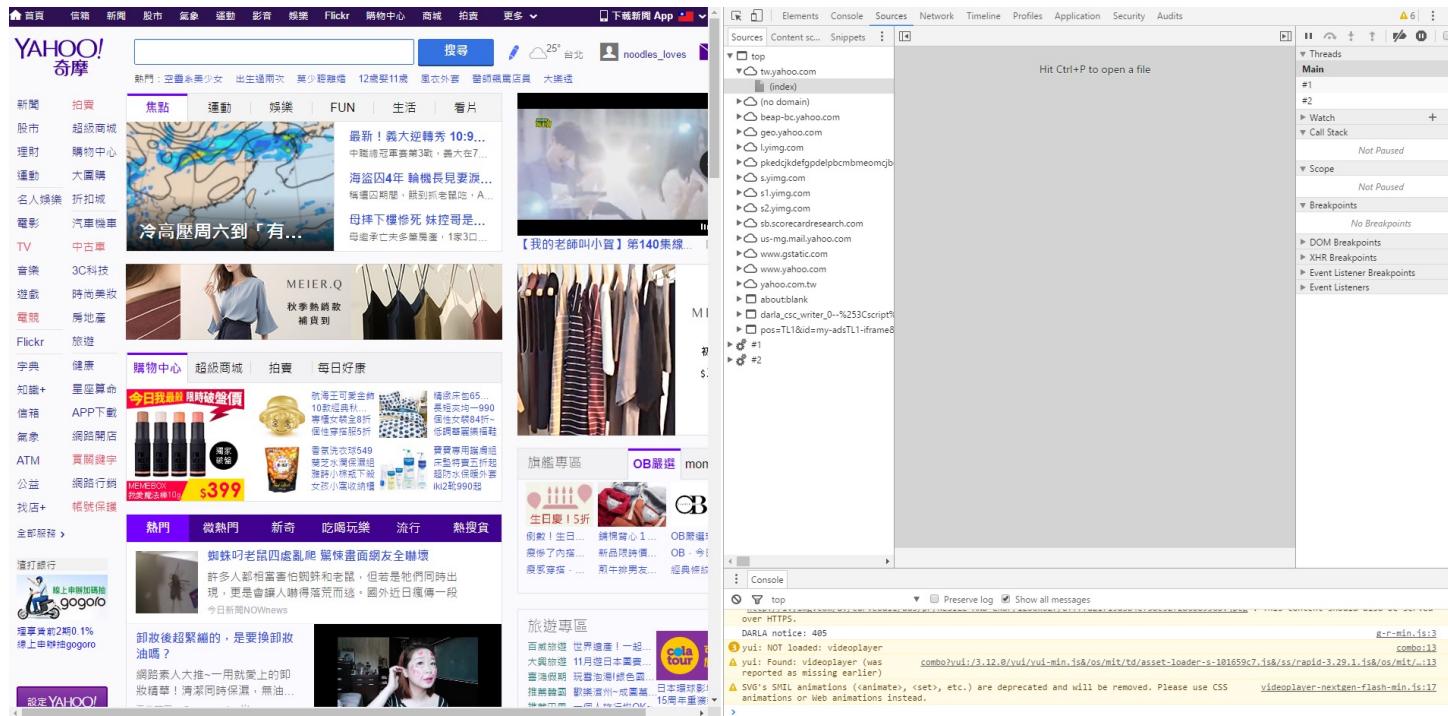
```
In [16]: print soup.title.string  
Yahoo奇摩
```

```
In [17]: print soup.find_all('h2')  
[<h2 class="Grid-U App-Title Pstart-12 Pos-r Z-1 Fz-1 Pt-8 Mt-0 Mb-12">\u65d7\u8266\u5c08\u5340</h2>, <h2 class="Grid-U App-Title Pstart-12 Pos-r Z-1 Fz-1 Pt-8 Mt-0 Mb-12">\u751f\u6d3b\u60c5\u5831</h2>, <h2 class="Grid-U App-Title Pstart-12 Pos-r Z-1 Fz-1 Pt-8 Mt-0 Mb-12">\u4eba\u6c23\u54c1\u724c</h2>]
```

```
In [11]: print soup.find(id="nav")  
Out[11]: <div class="Bxz-bb Pos-r Lh-1" id="nav" style="width:145px;">\n<div class="mod_view_default">\n<div class="bd type_navrail type_navrail_default">\n<div class="legacy W-50 Fl-start">\n<ul class="navlist My-0 Mend-12 Mstart-0 Ov-h Bd-b ne ws">\n<li class="Fz-m MouseOver">\n<a class="drag-item D-b Td-n" href="https://tw.news.yahoo.com/">\n<span class="Cur-p D-ib Py-8 C-blue MouseOver-TextDecoration" style="color:; ">\u65b0\u805e</span>\n</a>\n</li>\n<li class="Fz-m MouseOver">\n<a class="drag-item D-b Td-n" href="https://tw.stock.yahoo.com/">\n<span class="Cur-p D-ib Py-8 C-blue MouseOver-TextDecoration" style="color:; ">\u180a1\u1100</span>\n</a>\n</li>
```

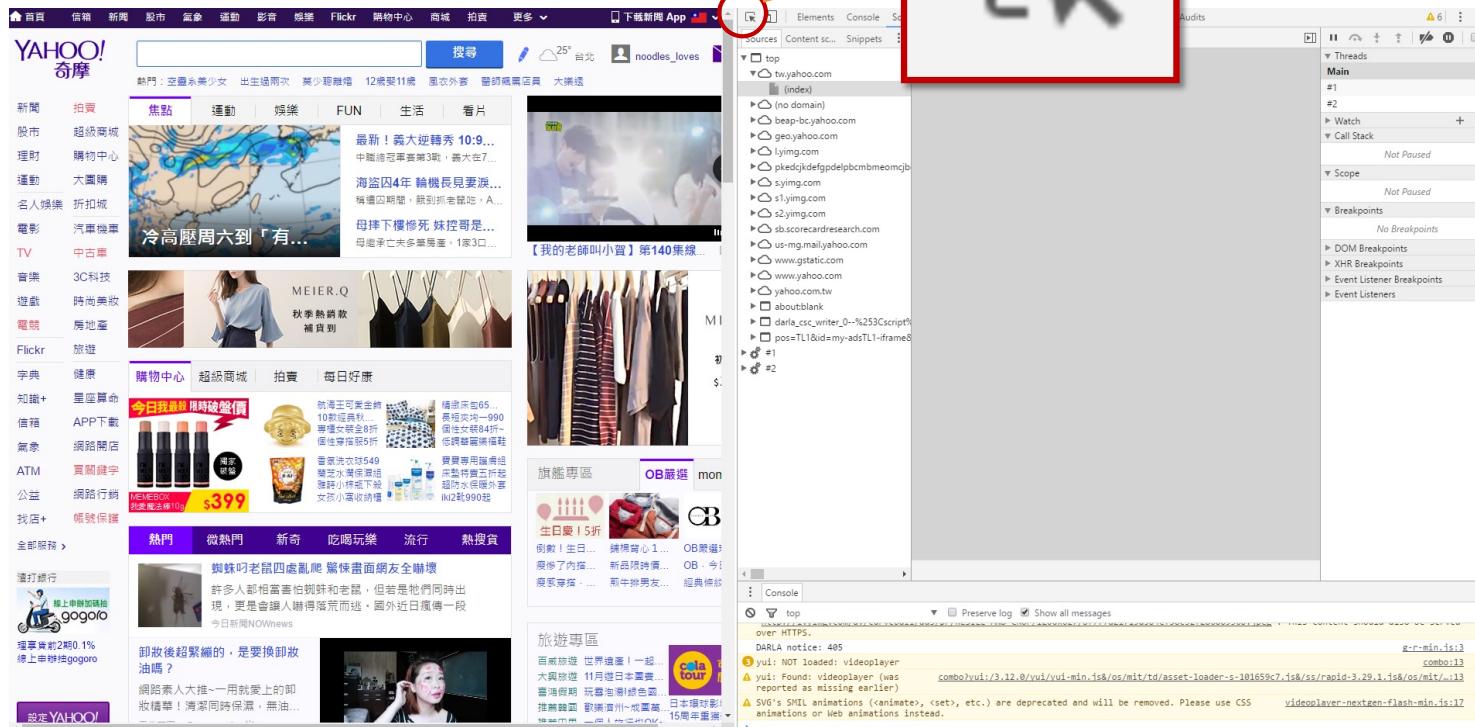
快速查看網頁元件資訊

- 進入網頁按F12，或者點右鍵按檢查 Mac 使用 Command+Option+I



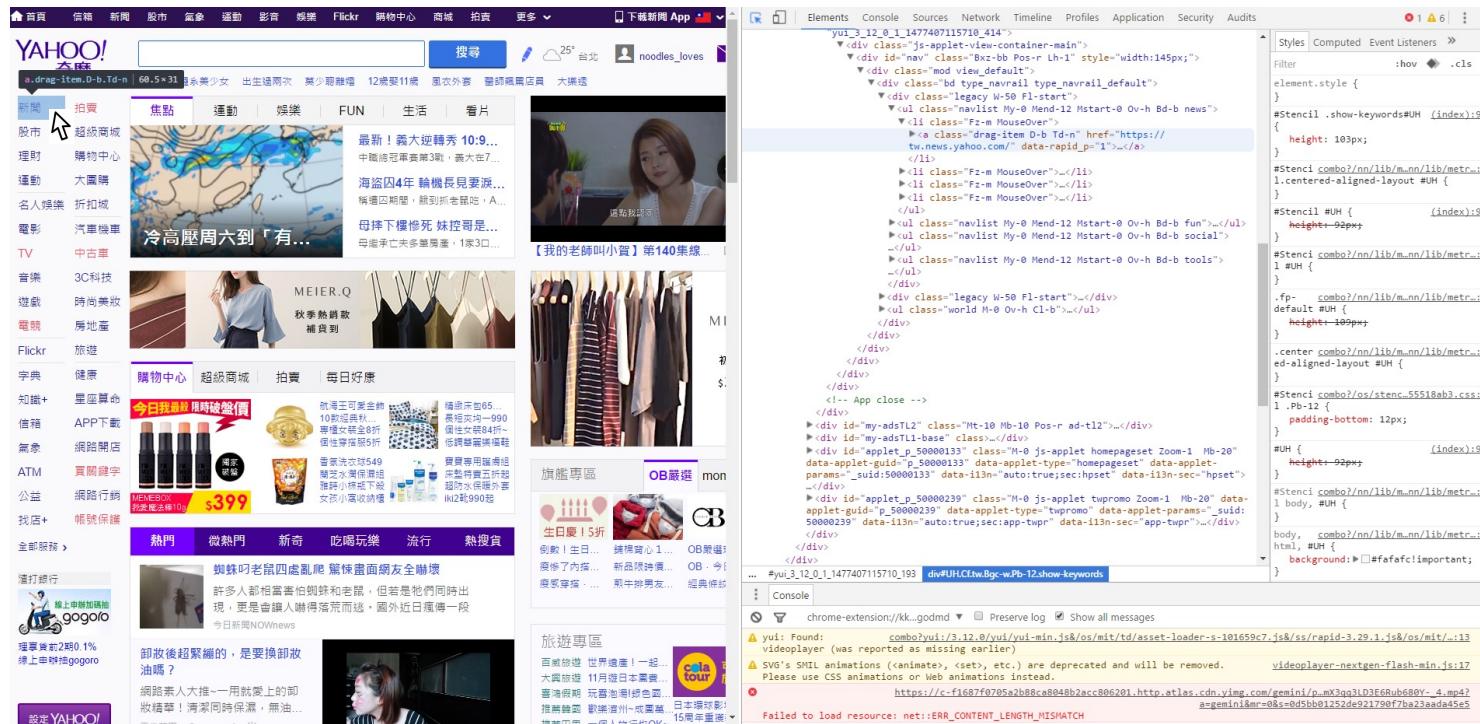
快速查看網頁元件資訊

- 點他 >>>>>



快速查看網頁元件資訊

- 把滑鼠移到想要查詢的元件



XPath

XPath

- 為XML路徑語言（ XML Path Language ），它是一種用來確定 XML文檔中某部分位置的語言。
- 基於XML的樹狀結構，提供在資料結構樹中找尋節點的能力。被開發者採用來當作小型查詢語言。
- Example:

```
▼ <div class="section" id="id2">
  ► <h2>...</h2>
    <p>使用 Requests 发送网络请求非常简单。</p>
    <p>一开始要导入 Requests 模块：</p> == $0
```

```
//*[@id="id2"]/p[2]
```

<https://zh.wikipedia.org/wiki/XPath>

XPath

The screenshot shows a web browser window with the Yahoo homepage loaded. The developer tools are open, specifically the Elements tab, which displays the HTML structure of the page. A context menu is open over a specific element, showing options like 'Copy' and 'Copy XPath'. The 'Copy XPath' option is highlighted.

Elements tab content (partial HTML):

```
<!-- App open -->
<style>...</style>
<div class="App-Bd">
  <div class="App-Main yui3-app-views" data-region="main" id="yui_3_12_0_1_1477929977691_532">
    <div class="js-applet-view-container-main">
      <div id="nav" class="Bxz-bb Pos-r Lh-1" style="width:145px;">
        <div class="mod view_default">
          <div class="bd type_nar nail type_narail_default">
            <div class="legacy W-50 Fl-start">...</div>
            <div class="legacy W-50 Fl-start">
              <ul class="navlist My-0 Mend-12 Mstart-0 Ov-h Bd-b pcshopping">
                <li class="Fz-n MouseOver">
                  <a class="drag-item D-b Td-n" href="https://tw.bid.yahoo.com/?co_servername=aME&co_servername2=aME" data-rapid_p="19">
                    <span class="Cur-p D-ib Py-8 C-blue MouseOver-TextDecoration" style="color: #EF4E5E; ">拍賣 /span> == $0
                  </a>
                </li>
                <li class="Fz-n">
                <li class="Fz-n">
                <li class="Fz-n">
                </ul>
              <ul class="navlist My-0 Mend-12 Mstart-0 Ov-h Bd-b pcshopping">
                <li class="Fz-n MouseOver">
                  <a class="drag-item D-b Td-n" href="https://tw.bid.yahoo.com/?co_servername=aME&co_servername2=aME" data-rapid_p="19">
                    <span class="Cur-p D-ib Py-8 C-blue MouseOver-TextDecoration" style="color: #EF4E5E; ">拍賣 /span> == $0
                  </a>
                </li>
                <li class="Fz-n">
                <li class="Fz-n">
                <li class="Fz-n">
                </ul>
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>
</div>
```

Console tab content:

- 2 ► A Parser-blocking, cross-origin script https://s.yimg.com/bf/homerun/hotspot/ may be blocked by the browser if the connection is lost.
- 3 ► yui: NOT loaded: videoplayer
- 4 ► yui: Found: videoplayer (was reported as missing earlier)

Xpath example

```
In [8]: from lxml import html
import requests
url = "http://yahoo.com.tw"
r = requests.get(url)
content = r.text

tree = html.fromstring(content)
p_text = tree.xpath('//*[@id="nav"]/div/div[2]/ul[1]/li[1]/a/span/text()')
print p_text[0]
```

拍賣

收藏本站 | 阅读客户端 | 免费 | 搜索 ▾

— 百度文学旗下 —

纵横中文网 原创精品
www.zongheng.com

广告

英雄打天下

关闭

首页 | 花语女生 | 图书频道 | 书库 | 动漫 | 小说排行榜 | 资讯 | 作者专区 | 个人中心 | 社区 | 游

会员 | 免费小说 | 全本 | 名著 | 奇幻·玄幻 | 武侠·仙侠 | 历史·军事 | 都市·娱乐 | 竞技·同人 | 科幻·游戏 | 悬疑·灵异 |

专题 9月月票榜冠军：食堂包子《祭炼山河》

升级 唐文 玄幻 综合 ▾ 万域之王

众神之战 畅世来袭 纵横游戏中心

在线客服

首页 > 排行榜

排行榜 索引

综合排行榜

综合排行榜

百度小说月票榜 规则说明

名次	作品	票数
1	超品战兵	10120
2	十方神王	7048
3	在中原行镖的日子	6622
4	儒武争锋	5948
5	霉运阴阳眼	2462
6	书剑长安	2345
7	火帝神尊	2200
8	纯禽记者	940
9	最强狂兵	789

言情小说点击榜

名次	作品	日	周	月
1	神犬小七	135		
2	重生毒女谋天下	35		
3	王妃如此多娇	33		
4	一爱千年：魔君的心头...	27		
5	盛世婚宠：老公送上门	27		
6	陆少的暖婚新妻	24		
7	诱妻入怀：前夫，请温柔	23		
8	重生之凰女驾到	19		
9	汉家美人谋	17		

Elements Console Sources Network Timeline Profile

```
</div>
</div>
<!-- 女主笔点击榜 --&gt;
<!-- 女主笔点击榜 --&gt;
<div class="box ph_list mar_right10 tabcontainer">


综合
月票榜
人气榜
评论榜
推荐榜
新书榜
综合



综合排行榜



| 名次 | 作品       | 票数    |
|----|----------|-------|
| 1  | 超品战兵     | 10120 |
| 2  | 十方神王     | 7048  |
| 3  | 在中原行镖的日子 | 6622  |
| 4  | 儒武争锋     | 5948  |
| 5  | 霉运阴阳眼    | 2462  |
| 6  | 书剑长安     | 2345  |
| 7  | 火帝神尊     | 2200  |
| 8  | 纯禽记者     | 940   |
| 9  | 最强狂兵     | 789   |



言情小说点击榜



| 名次 | 作品            | 日   | 周 | 月 |
|----|---------------|-----|---|---|
| 1  | 神犬小七          | 135 |   |   |
| 2  | 重生毒女谋天下       | 35  |   |   |
| 3  | 王妃如此多娇        | 33  |   |   |
| 4  | 一爱千年：魔君的心头... | 27  |   |   |
| 5  | 盛世婚宠：老公送上门    | 27  |   |   |
| 6  | 陆少的暖婚新妻       | 24  |   |   |
| 7  | 诱妻入怀：前夫，请温柔   | 23  |   |   |
| 8  | 重生之凰女驾到       | 19  |   |   |
| 9  | 汉家美人谋         | 17  |   |   |



言情小说月票榜



| 名次 | 作品            | 票数  |
|----|---------------|-----|
| 1  | 神犬小七          | 135 |
| 2  | 重生毒女谋天下       | 35  |
| 3  | 王妃如此多娇        | 33  |
| 4  | 一爱千年：魔君的心头... | 27  |
| 5  | 盛世婚宠：老公送上门    | 27  |
| 6  | 陆少的暖婚新妻       | 24  |
| 7  | 诱妻入怀：前夫，请温柔   | 23  |
| 8  | 重生之凰女驾到       | 19  |
| 9  | 汉家美人谋         | 17  |


```

縱橫中文網實戰!!!

```
In [98]: import requests
from bs4 import BeautifulSoup
url = "http://book.zongheng.com/rank.html"
re = requests.get(url)
content = re.text

soup = BeautifulSoup(content,"html.parser")
book_list = soup.find_all(class_="book_list")

for link in book_list[1].find_all('a'):
    title = link.string
    url_link = link.get('href')

    print title
    print url_link
```

神犬小七
<http://huayu.baidu.com/book/585151.html>
王妃如此多娇
<http://huayu.baidu.com/book/599859.html>
重生毒女谋天下
<http://huayu.baidu.com/book/598174.html>
盛世婚宠：老公送上门
<http://huayu.baidu.com/book/318263.html>

登录 注册



首页 | 排行榜 | 女生书库 | 古代言情 | 都市言情 | 幻想时空 | 耽美同人 | 全本频道 | 个人

VIP作品 | 书讯 | 动漫 | 社区 | VIP充值 | 客服帮助 | 和阅读女生

关键字: 书名

您的位置: 花语女生网 → 女生书库 → 都市言情 → 神犬小七

神犬小七

作者: 月涵 +关注 A级签约

总点击: 792330

总字数: 362083

总红票: 5920

作品积分: 1350

更新: 2016-10-30

连载中

读者

无鞋

书友1

师座1

师座2

...

本书

排名

1

2

3

4



一键转帖: MSN 移动微博 搜狐 QQ空间 新浪 腾讯

作品简介:

爱狗成痴的林宛白觉得要男人不如要条狗，而当男人跟狗结合在一起时……完美！

湖南卫视电视剧《神犬小七》同名小说。7月17日《神犬小七》金鹰独播剧场精彩上映。

作者标签: 萌宠养成 冷酷总裁 豪门世家 欢喜冤家

本书共60条评论

[查看评论]

普通阅读

收藏本书

投红票

订阅VIP

```
Elements Console Sources Network Timeline Profiles Application  
追妻成狂  


追妻成狂



# 猎爱小军医



桃爆看



普通阅读



collect favorite



thp voteRecommend tp="0"



baoyue dbby



订阅VIP

<!-- 图片及简介 结束 --&gt;<div><!-- 作品相关 结束 --&gt;<div class="book_wrap">
```

Console

top

Preserve log

A Parser-blocking, cross-origin script.

縱橫中文網實戰!!!

```
In [99]: import requests
from bs4 import BeautifulSoup
url = "http://book.zongheng.com/rank.html"
re = requests.get(url)
content = re.text

soup = BeautifulSoup(content,"html.parser")
book_list = soup.find_all(class_="book_list")

for link in book_list[1].find_all('a'):
    title = link.string
    url_link = link.get('href')

    result = requests.get(url_link)
    result_content = result.text
    soup2 = BeautifulSoup(result_content,"html.parser")
    button_list = soup2.find_all(class_="button")
    read_url = button_list[0].find('a').get('href')

    article_url = "http://book.zongheng.com" + read_url
    print article_url
    break
```

<http://book.zongheng.com/showchapter/585151.html>

登录 注册

纵横中文网 女生网首页 排名



追妻成狂
LIE AI XIAO JUN YE

猎爱小军医

桃夭未央 [作品]
爆笑军婚
看首长宠妻日常

花语女生网 | 加入书架 | 投红票 | 全文阅读(新版 旧版) | 下载本书 | 返回书页 | 返回书目

神犬小七 作者: 月涵

《神犬小七》正文章节 全文阅读[新版 旧版] ★ 收藏本书 下载本书

卷号200 正文 [分卷阅读]

章节名/章节提要	章节字数	页数
第一章 宫翎是条狗!	3585	2C
第二章 夜半，宫小夜到访！	3533	2C
第三章 狗是我的第二种人格！	3536	2C
第四章 狗，是我内心深处的召唤！	3165	2C
第五章 就你，也配跟我抢女人？	2255	2C
第六章 宫翎霸道强吻！	3729	2C
第七章 女人，你当我摆设吗？	3034	2C
第八章 没有狗，怎么能称之为狗仔！	2570	2C

Elements Console Sources Network Timeline Profiles

```
<div class="cl"></div>
<div class="wraph">
  <div class="book_nav pink">...</div>
  <div class="cl"></div>
  <div class="book_title">...</div>
  <div class="cl"></div>
  <div class="bookchaplist clearmar">
    <!-- 正文章节 -->
    <div class="book_chattit">...</div>
    <!-- 正文章节 结束 -->
    <div class="cl"></div>
    <!-- 章节列表 -->
    <div class="book_chapter chapter_newwidth">
      <div class="pinklow">...</div>
      <div class="con bordec_notop">
        <div class="chapter_head">...</div>
        <ul>
          <li>
            <span class="chapname">
              <a href="/chapter/585151/33339803.html" target="_blank">第一章 宫翎是条狗！</a>
              ==>
              <a href="/chapter/585151/33339803.html" target="normal" target="_blank">
                </a>
            </span>
            <span class="chapcount">3585</span>
            <span class="chaptim" style="font-size: small;">2016-07-27 05:13:26</span>
          </li>
        </ul>
      </div>
    </div>
  </div>
</div>
```

html body div div div div div ul li span.chapname a

Console

縱橫中文網實戰!!!

```
In [100]: import requests
from bs4 import BeautifulSoup
url = "http://book.zongheng.com/rank.html"
re = requests.get(url)
content = re.text

soup = BeautifulSoup(content,"html.parser")
book_list = soup.find_all(class_="book_list")

for link in book_list[1].find_all('a'):
    title = link.string
    url_link = link.get('href')

    result = requests.get(url_link)
    result_content = result.text
    soup2 = BeautifulSoup(result_content,"html.parser")
    button_list = soup2.find_all(class_="button")
    read_url = button_list[0].find('a').get('href')

    article_url = "http://book.zongheng.com" + read_url
    article = requests.get(article_url)
    article_content = article.text
    soup3 = BeautifulSoup(article_content,"html.parser")
    artricle_list = soup3.find(class_ = "con_bordec_notop")
    for l in artricle_list.find_all('a'):
        print l
    break

<a href="/chapter/585151/33339803.html" target="_blank">第一章 宮翎是条狗！</a>
<a class="normal" href="/chapter/585151/33339803.html" target="_blank">
</a>
<a href="/chapter/585151/33346746.html" target="_blank">第二章 夜半，宮小祓到訪！</a>
<a class="normal" href="/chapter/585151/33346746.html" target="_blank">
```

登录 注册



花语女生网 | 加入书签 | 投红票 | 全文阅读(新版 旧版) | 下载本书 | 返回书页 | 返回书目

纵横中文网

Elements Console Sources Network Timeline Profiles Application

```
...</uiv>
<!-- 登录弹出框 -->
<a name="top"></a>
<div class="headwrap">...</div>
<div class="c1"></div>
<div class="wrap">...</div>
<div class="c1"></div>
<div class="wrap">...</div>
<div class="wrap book_reader">
  <h1>...</h1>
  <div class="tc">...</div>
  <div class="tc">[更新时间] 2016-07-27 05:13:26 [字数]&nbsp;3585</div>
  <div class="book_con">
    <p>...</p> = $0
    <p>...</p>
    <n>...</n>
  </div>

```

“Encore! ”

“Encore! ! ”

“Encore! ! ! ”

“Encore.....”

在歌迷们热情的呼喊声中，一束灯光打向舞台。

```
In [102]: import requests
from bs4 import BeautifulSoup
url = "http://book.zongheng.com/rank.html"
re = requests.get(url)
content = re.text

soup = BeautifulSoup(content,"html.parser")
book_list = soup.find_all(class_="book_list")

for link in book_list[1].find_all('a'):
    title = link.string
    url_link = link.get('href')

    result = requests.get(url_link)
    result_content = result.text
    soup2 = BeautifulSoup(result_content,"html.parser")
    button_list = soup2.find_all(class_="button")
    read_url = button_list[0].find('a').get('href')

    article_url = "http://book.zongheng.com" + read_url
    article = requests.get(article_url)
    article_content = article.text
    soup3 = BeautifulSoup(article_content,"html.parser")
    artricle_list = soup3.find(class_ = "con_bordec_notop")
    for l in artricle_list.find_all('a'):
        content_url = "http://book.zongheng.com" + l.get('href')
        c = requests.get(content_url)
        soup4 = BeautifulSoup(c.text,"html.parser")
        for con in soup4.find(class_="book_con").find_all("p"):
            remove = con.find("span").get_text()
            temp = con.get_text()
            print temp.replace(remove,"")
        break
    break

"Encore ! "
"Encore ! ! "
"Encore ! ! ! "
```

Scrapy + XPath

Findprice 實戰!!!

商品 圖書 車價網 團購 供應商

刊登採購訊息 加入供應商 登入

FindPrice

商品搜尋 目錄搜尋

熱門搜尋：[X5](#) [水晶](#) [水漾](#) [好神](#) [老爺酒店](#) [光陽機車 many](#) [金莎巧克力](#) [GP 125](#) [YAMAHA山葉機車 SMAX](#) [more...](#)

品牌總覽：[寶貝](#) [視聽](#) [相機](#) [贈品](#) [食品](#) [服飾](#) [通訊](#) [眼鏡](#) [電器](#) [婦幼](#) [廚具](#) [美妆](#)

供應廠商：[福睿得生活事業有限公司](#) [寶欣有限公司](#) [博優創藝有限公司](#) [馬來西亞商會益補國際股份有限公司台灣分公司](#) > [更多供應商](#)

採購訊息：[60W傳統捲線燈泡 數量:2000 有效期限:2016/11/12](#) > [更多採購訊息](#)

今日好康：[beauty88時尚美人【MORINO摩力諾】★團購超值價★素色隱形襪\(10雙\) \\$399](#) > [更多好康](#)

[f](#) [讚 3,838](#) [G](#) [G+ 146](#)

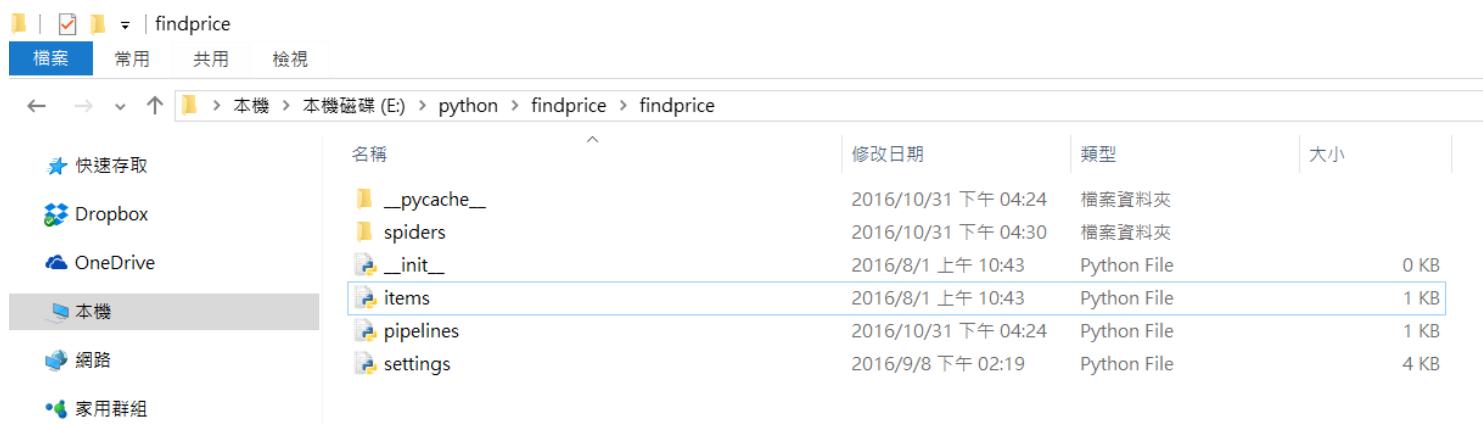
©2016 FindPrice - [商品](#) [圖書](#) [車價網](#) [團購](#) [供應商](#) [服務條款](#) [隱私權政策](#) [行動版](#)

Findprice 實戰!!!

- 建專案

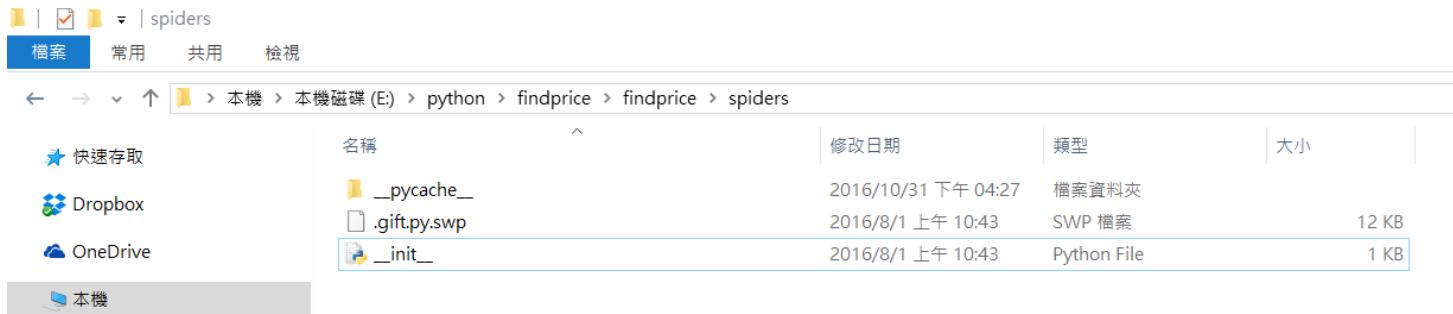
```
E:\python
λ scrapy startproject findprice
```

- 完成後會出現findprice資料夾

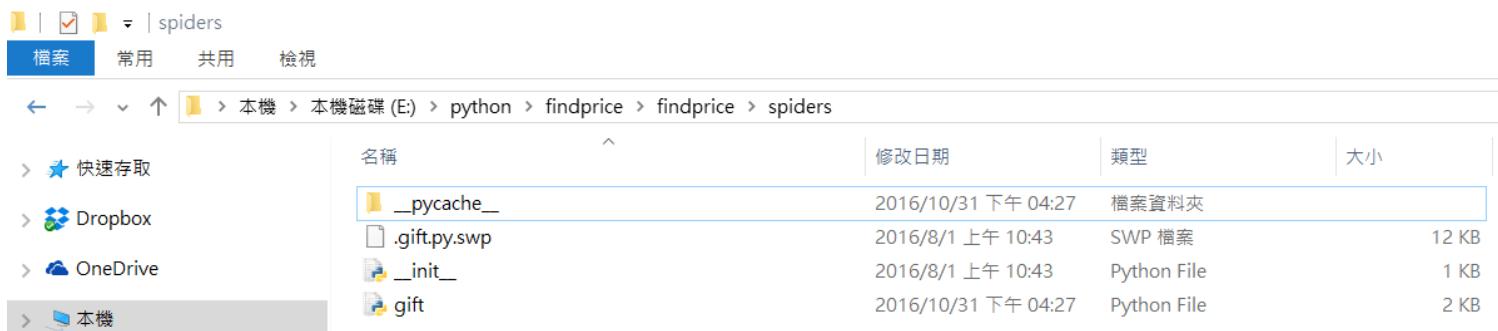


Findprice 實戰!!!

- 進入spiders資料夾



- 手動建一個.py檔



Gift.py

```
1 # -*- coding: utf-8 -*-
2 import scrapy
3 import re
4 import sys
5
6 from scrapy.http import Request
7
8 class GiftSpider(scrapy.Spider):
9     name = "gift"
10    allowed_domains = ["findprice.com.tw"]
11
12    gift_list = [
13        u"卡比獸+抱枕",
14        u"小火龍+娃娃"
15    ]
16    target_url = u'https://www.findprice.com.tw/datalist.aspx?s=g&q='
17
18    start_urls = [
19        target_url + gift_list[0],target_url + gift_list[1]
20    ]
21    print(start_urls)
22
23    def start_requests(self):
24        requests = []
25        for item in self.start_urls:
26            requests.append(Request(url=item, headers={'Referer': 'https://www.findprice.com.tw/'}))
27        return requests
28
29    def parse(self, response):
30        pass
```

Copy Xpath

The screenshot shows a web browser window with the FindPrice search results for "卡比獸 抱枕". The search bar at the top has the query "卡比獸 抱枕". Below the search bar, there are filters for sorting by price (低至高排序) and price range (不限). The results section displays several items, including a Philips air purifier and a pair of shoes from ONE BOY.

The developer tools are open, specifically the Elements tab. A context menu is displayed over one of the search results. The menu options include:

- Add attribute
- Edit as HTML
- Copy
- Copy outerHTML
- Copy selector
- Copy XPath
- Hide element
- Delete element
- Expand all
- Collapse all
- Cut element
- Copy element
- Paste element

The "Copy XPath" option is highlighted in the menu. The right-hand panel of the developer tools shows the selected element's CSS selector: ".rec-gname".

告訴parse要抓網頁的哪個物件資料

```
def parse(self, response):
    target = response.xpath('//*[@id="q"]/@value').extract()
    page = response.xpath('//*[@id="pg"]/strong/text()').extract()
    if page == []:
        page = 1

    for i in response.xpath('//*[@id="GoodsGridDiv"]/table/tr'):
        name = i.xpath('td/a/text()').extract()
        price = i.xpath('td/font/b/text()').extract()
        print("".join(name))
        print(price)

    if len(response.xpath('//*[@id="pg-next"]')).extract() > 0:
        path = response.xpath('//*[@id="pg-next"]/@href').extract()
        target_url = u'https://www.findprice.com.tw/'
        url = target_url + path[0]
        page = path[0].split("i=")[1].split("&")[0]
        if(int(page) <= 5):
            yield scrapy.Request(url, self.parse)
```

執行scrapy

```
E:\python\findprice
$ scrapy crawl gift
['https://www.findprice.com.tw/datalist.aspx?s=g&q=卡比獸+抱枕', 'https://www.findprice.com.tw/datalist.aspx?s=g&q=小火龍+娃娃']
2016-10-31 16:52:42 [scrapy] INFO: Scrapy 1.2.0 started (bot: findprice)
2016-10-31 16:52:42 [scrapy] INFO: Overridden settings: {'BOT_NAME': 'findprice', 'ROBOTSTXT_OBEY': True, 'NEWSPIDER_MODULE': 'findprice.spiders', 'SPIDER_MODULES': ['findprice.spiders']}
2016-10-31 16:52:42 [scrapy] INFO: Enabled extensions:
['scrapy.extensions.telnet.TelNetConsole',
 'scrapy.extensions.logstats.LogStats',
 'scrapy.extensions.corestats.CoreStats']
2016-10-31 16:52:42 [scrapy] INFO: Enabled downloader middlewares: cl=item, headers={'Referer': 'https://www.findprice.com.tw/datalist.aspx?s=g&q=卡比獸+抱枕'}
['scrapy.downloadermiddlewares.robotstxt.RobotsTxtMiddleware',
 'scrapy.downloadermiddlewares.httpauth.HttpAuthMiddleware',
 'scrapy.downloadermiddlewares.downloadtimeout.DownloadTimeoutMiddleware',
 'scrapy.downloadermiddlewares.defaultheaders.DefaultHeadersMiddleware',
 'scrapy.downloadermiddlewares.useragent.UserAgentMiddleware',
 '@id="pg"]/@value').extract()
'scrapy.downloadermiddlewares.retry.RetryMiddleware',
 'scrapy.downloadermiddlewares.redirect.MetaRefreshMiddleware',
 '@id="pg"]/strong/text()').extract()
'scrapy.downloadermiddlewares.httpcompression.HttpCompressionMiddleware',
 'scrapy.downloadermiddlewares.redirect.RedirectMiddleware',
 'scrapy.downloadermiddlewares.cookies.CookiesMiddleware',
 'scrapy.downloadermiddlewares.chunked.ChunkedTransferMiddleware',
 'scrapy.downloadermiddlewares.stats.DownloaderStats',
 '@id="GoodsGridDiv"]/table/tr');
'scrapy.downloadermiddlewares.httpcache.HttpCacheMiddleware']
2016-10-31 16:52:42 [scrapy] INFO: Enabled spider middlewares:
['scrapy.spidermiddlewares.httperror.HttpErrorMiddleware',
 '@id="pg"]/@value').extract()
'scrapy.spidermiddlewares.offsite.OffsiteMiddleware',
 'scrapy.spidermiddlewares.referrer.ReferrerMiddleware',
 'scrapy.spidermiddlewares.urllength.UrlLengthMiddleware',
 'scrapy.spidermiddlewares.depth.DepthMiddleware']
2016-10-31 16:52:42 [scrapy] INFO: Enabled item pipelines: ['@id="pg-next"]').extract() > 0 :
['findprice.pipelines.FindpricePipeline']
2016-10-31 16:52:42 [scrapy] INFO: Spider opened
2016-10-31 16:52:42 [scrapy] INFO: Crawled 0 pages (at 0 pages/min), scraped 0 items (at 0 items/min)
2016-10-31 16:52:42 [scrapy] DEBUG: Telnet console listening on 127.0.0.1:6023
2016-10-31 16:52:42 [scrapy] DEBUG: Crawled (200) <GET https://www.findprice.com.tw/robots.txt> (referer: None) ['cached']
2016-10-31 16:52:42 [scrapy] DEBUG: Crawled (200) <GET https://www.findprice.com.tw/datalist.aspx?s=g&q=%E5%8D%A1%E6%AF%94%E7%8D%B8+%E6%8A%B1%E6%9E%95>
2016-10-31 16:52:42 [scrapy] DEBUG: Crawled (200) <GET https://www.findprice.com.tw/datalist.aspx?s=g&q=%E5%B0%8F%E7%81%AB%E9%BE%8D+%E5%A8%83%E5%A8%83>
    神奇寶貝口袋妖怪公仔寵物小精靈毛絨玩具玩偶動漫周邊
['179']      神奇寶貝公仔毛絨玩具玩偶靠墊動漫周邊生日禮物女
['238']      精靈寶可夢 Pokemon GO 神奇寶貝 可愛超柔絨毛娃娃 30CM ☆現貨供應☆【宇庭飾品店】
['288']      公仔娃娃 50公分 精靈寶可夢 Pokemon GO 【RS487】 (4.4折)
```

執行scrapy

```
E:\python\findprice
$ scrapy crawl gift
['https://www.findprice.com.tw/datalist.aspx?s=g&q=卡比獸+抱枕', 'https://www.findprice.com.tw/datalist.aspx?s=g&q=小火龍+娃娃']
2016-10-31 16:52:42 [scrapy] INFO: Scrapy 1.2.0 started (bot: findprice)
2016-10-31 16:52:42 [scrapy] INFO: Overridden settings: {'BOT_NAME': 'findprice', 'ROBOTSTXT_OBEY': True, 'NEWSPIDER_MODULE': 'findprice.spiders', 'SPIDER_MODULES': ['findprice.spiders']}
2016-10-31 16:52:42 [scrapy] INFO: Enabled extensions:
['scrapy.extensions.telnet.TelNetConsole',
 'scrapy.extensions.logstats.LogStats',
 'scrapy.extensions.corestats.CoreStats']
2016-10-31 16:52:42 [scrapy] INFO: Enabled downloader middlewares: cl=item, headers={'Referer': 'https://www.findprice.com.tw/datalist.aspx?s=g&q=卡比獸+抱枕'}
['scrapy.downloadermiddlewares.robotstxt.RobotsTxtMiddleware',
 'scrapy.downloadermiddlewares.httpauth.HttpAuthMiddleware',
 'scrapy.downloadermiddlewares.downloadtimeout.DownloadTimeoutMiddleware',
 'scrapy.downloadermiddlewares.defaultheaders.DefaultHeadersMiddleware',
 'scrapy.downloadermiddlewares.useragent.UserAgentMiddleware',
 'scrapy.downloadermiddlewares.retry.RetryMiddleware',
 'scrapy.downloadermiddlewares.redirect.MetaRefreshMiddleware',
 'scrapy.downloadermiddlewares.httpcompression.HttpCompressionMiddleware',
 'scrapy.downloadermiddlewares.redirect.RedirectMiddleware',
 'scrapy.downloadermiddlewares.cookies.CookiesMiddleware',
 'scrapy.downloadermiddlewares.chunked.ChunkedTransferMiddleware',
 'scrapy.downloadermiddlewares.stats.DownloaderStats',
 'scrapy.downloadermiddlewares.httpcache.HttpCacheMiddleware']
2016-10-31 16:52:42 [scrapy] INFO: Enabled spider middlewares:
['scrapy.spidermiddlewares.httperror.HttpErrorMiddleware',
 'scrapy.spidermiddlewares.offsite.OffsiteMiddleware',
 'scrapy.spidermiddlewares.referrer.ReferrerMiddleware',
 'scrapy.spidermiddlewares.urllength.UrlLengthMiddleware',
 'scrapy.spidermiddlewares.depth.DepthMiddleware']
2016-10-31 16:52:42 [scrapy] INFO: Enabled item pipelines: ['findprice.pipelines.FindpricePipeline']
2016-10-31 16:52:42 [scrapy] INFO: Spider opened
2016-10-31 16:52:42 [scrapy] INFO: Crawled 0 pages (at 0 pages/min), scraped 0 items (at 0 items/min)
2016-10-31 16:52:42 [scrapy] DEBUG: Telnet console listening on 127.0.0.1:6023
2016-10-31 16:52:42 [scrapy] DEBUG: Crawled (200) <GET https://www.findprice.com.tw/robots.txt> (referer: None) ['cached']
2016-10-31 16:52:42 [scrapy] DEBUG: Crawled (200) <GET https://www.findprice.com.tw/datalist.aspx?s=g&q=%E5%8D%A1%E6%AF%94%E7%8D%B8+%E6%8A%B1%E6%9E%95>
2016-10-31 16:52:42 [scrapy] DEBUG: Crawled (200) <GET https://www.findprice.com.tw/datalist.aspx?s=g&q=%E5%8F%E7%81%AB%E9%BE%8D+%E5%A8%83%E5%A8%83>
['179']      神奇寶貝口袋妖怪公仔寵物小精靈毛絨玩具玩偶動漫周邊
['238']      神奇寶貝公仔毛絨玩具玩偶靠墊生日禮物女
['288']      精靈寶可夢 Pokemon GO 神奇寶貝 可愛超柔絨毛娃娃 抱枕30CM ☆現貨供應☆【宇庭飾品店】
['288']      公仔娃娃 50公分 精靈寶可夢 Pokemon GO 【RS487】 (4.4折)
```



執行scrapy

```
2016-10-31 16:52:42 [scrapy] INFO: Closing spider (finished)
2016-10-31 16:52:42 [scrapy] INFO: Dumping Scrapy stats:
{'downloader/request_bytes': 2768,
 'downloader/request_count': 7,
 'downloader/request_method_count/GET': 7, response):
 'downloader/response_bytes': 127969,
 'downloader/response_count': 7,
 'downloader/response_status_count/200': 7,
 'finish_reason': 'finished',
 'finish_time': datetime.datetime(2016, 10, 31, 8, 52, 42, 970945),
 'httpcache/hit': 7,
 'log_count/DEBUG': 8,
 'log_count/INFO': 7, for i in response.xpath('//*[@id="content"]//td[@class="list-item"]'):
 'request_depth_max': 3,
 'response_received_count': 7,
 'scheduler/dequeued': 6,   name = i.xpath('td/a/text()').extract_first(),
 'scheduler/dequeued/memory': 6,   price = i.xpath('td/font/b/text()').extract_first(),
 'scheduler/enqueued': 6,   print(name)
 'scheduler/enqueued/memory': 6,
 'start_time': datetime.datetime(2016, 10, 31, 8, 52, 42, 433705)}
2016-10-31 16:52:42 [scrapy] INFO: Spider closed (finished)
```

記得加delay time
以免被誤以為是
惡意攻擊程式

Steam Crawling



Steam is a digital distribution platform developed by Valve Corporation offering digital rights management (DRM), multiplayer gaming and social networking services. The Steam platform is considered to be the largest digital distribution platform for PC gaming

A screenshot of the Steam homepage. At the top, there's a banner for '近期更新過 主要產品更新' (Recently Updated Main Products) featuring 'Rise of the Tomb Raider™ 20 Year Celebration' and 'Zup!'. Below this is a '为您推薦' (Recommended for You) section with a thumbnail for 'N.E.R.O.: Nothing Ever Re...' and a link to '登入以查看個人化推薦並自訂您的 Steam 頁面' (Log in to view personalized recommendations and customize your Steam page). Further down, there's a 'Steam 繫賞家 推薦關注的鑑賞家' (Steam Critics Recommended密切关注的鉴赏家) section with thumbnails for 'Bahamut' and 'Reimu Maga's evil shrine'. At the bottom, there are sections for '熱門新品' (Hot New Releases), '暢銷商品' (Best-Selling Products), '即將發行' (Upcoming Releases), and '特惠' (Sales).

A screenshot of the Steam homepage featuring a large banner for 'HungerDungeon' with a 75% discount. To the right, there's a promotional image for 'EUROPA UNIVERSALIS IV' with a 75% discount. Below these, there are sections for '搶先體驗已推出' (Early Access Released) and '新推出 精選新品' (Newly Released Selected New Products), showing thumbnails for 'Medieval Tycoon', 'Battle.net Battle Pass', and 'Tom Clancy's Rainbow Six Siege'.

VALVE Developer Community

https://developer.valvesoftware.com/wiki/Steam_Web_API

The screenshot shows a dark-themed Wikipedia-style page for the Steam Web API. At the top, there's a navigation bar with the Valve logo, a search bar, and links for "Page", "Discussion", "View", and "View source". On the left, a sidebar contains links for "Navigation" (Main Page, Source SDK index, Recent changes, Random page), "Support" (Getting help, Source SDK FAQ, Level Design FAQ, SDK Help Forums), "Steam Community" (Source SDK Hub, Steam Games), and "Tools" (What links here, Related changes, Special pages, Printable version, Permanent link, Page information). The main content area is titled "Steam Web API" and features a "Contents [hide]" section with a tree view of API documentation. The tree includes sections for "License and further documentation", "Formats" (JSON, XML, VDF, CSV), "Interfaces and methods" (Game interfaces and methods, GetNewsForApp, GetGlobalAchievementPercentagesForApp, GetGlobalStatsForGame, GetPlayerSummaries), and detailed arguments for each method.

Get Player Summaries

The image displays two side-by-side screenshots of the Steam Player Summary interface for a user named 'Robin'.

Left Screenshot (Player Summary Overview):

- User Info:** Robin Walker, Bellevue, Washington, United States.
- 等级:** 等級 20 (Level 20)
- Valve 員工:** 1,000 髰驗值 (Experience points)
- Recent Activity:** 最近一次上線 1 小時, 53 分鐘前 (Last logged in 1 hour, 53 minutes ago)
- Challenges:** 徵章 7 (Achievements 7) - Alien (100 髰驗值), 13, Alien, and another achievement icon.
- Gameplay Statistics:** 遊戲 474, 物品庫 17, 萤幕捕圖 17, 評論 3.
- Groups:** 群組 34 (Valve 230 位成員), DotA 15 位成員, garry's mod 70,630 位成員.
- Friends:** 好友 301 (McVee 在線, EricS 最近一次上線 3 小時, 11 分鐘前, >>The Heartsman--> 在線, bOtter 最近一次上線 7 小時, 19 分鐘前, Alden 最近一次上線 12 小時, 10 分鐘前, MaxOS2D 在線).

Right Screenshot (Detailed Game Activity):

- Gameplay Statistics:** 遊戲時數 45, 成就 31 (Alien 100 髰驗值). Achievement progress bar at 31/70.
- Recent Games:**
 - BATTLE BROTHERS:** 遊戲時數共 135 小時, 最後執行於 10 月 11 日.
 - SMASH+GRAB:** 遊戲時數共 0.9 小時, 最後執行於 10 月 8 日.
 - SUBNAUTICA:** 遊戲時數共 18.0 小時, 最後執行於 10 月 3 日.
- Friends:** 好友 301 (McVee 在線, EricS 最近一次上線 3 小時, 11 分鐘前, >>The Heartsman--> 在線, bOtter 最近一次上線 7 小時, 19 分鐘前, Alden 最近一次上線 12 小時, 10 分鐘前, MaxOS2D 在線).

Step

1. Apply the key

<https://steamcommunity.com/dev>

Obtaining an Steam Web API Key

All use of the Steam Web API requires the use of an API Key. You can acquire one by [filling out this form](#). Use of the APIs also requires that you agree to the [Steam API Terms of Use](#).

You can type in localhost in domain name for your own test.



Step

2.Get User ID

<http://steamid.co/>

Steam ID Finder / Steam ID Converter
We have already generated 2.474.330 profiles!

https://steamcommunity.com/id/robinwalker

Steam profile of: Robin

Robin Offline
Account is 13 years old

STEAMID.CO Trust Score* 7.1

STEAM PROFILE STEAMREP

Player Reputation

STEAM_0:0:849010 > Steam ID
76561197960435530 > Steam 64 ID
[U:1:1698020] > Steam 3 ID
<http://steamid.co/u/robinwalker> > Steamid.co Link

User has no VAC bans
User is not tradebanned
User has no community bans

No STEAMID.CO STATUS ?
VALVE ADMIN ✓



Result

Example URL:

<http://api.steampowered.com/ISteamUser/GetPlayerSummaries/v0002/?key=KEY&steamids=U>



<https://steamcommunity.com/id/robinwalker/>

```
{
    "response": {
        "players": [
            {
                "steamid": "76561197960435530",
                "communityvisibilitystate": 3,
                "profilestate": 1,
                "personaname": "Robin",
                "lastlogoff": 1476242584,
                "profileurl": "http://steamcommunity.com/id/robinwalker/",
                "avatar": "https://steamcdn-a.akamaihd.net/steamcommunity/p",
                "avatarmedium": "https://steamcdn-a.akamaihd.net/steamcommu",
                "avatarfull": "https://steamcdn-a.akamaihd.net/steamcommunit",
                "personastate": 0,
                "realname": "Robin Walker",
                "primaryclanid": "103582791429521412",
                "timecreated": 1063407589,
                "personastateflags": 0,
                "loccountrycode": "US",
                "locstatecode": "WA",
                "loccityid": 3961
            }
        ]
    }
}
```

Python

```
import requests  #$ pip install requests
import json
SteamKey="XXXXXXXXXX"
SteamId="XXXXXXXXXX"

#get web data
res =
requests.get("http://api.steampowered.com/ISteamUser/GetPlayerSummaries
/v0002/?key=%s&steamids=%s"%(SteamKey, SteamId))

feature = json.loads(res.text) # decode Json data

for x in feature['response']['players'][0].keys(): #print data
    print x,":",feature['response']['players'][0][x]
```

Result

```
steamid : [REDACTED]
primaryclanid : [REDACTED]
realname : Stan
personaname : Stan.
personastate : 0
personastateflags : 0
communityvisibilitystate : 3
loccountrycode : TW
profilestate : 1
profileurl : http://steamcommunity.com/profiles/[REDACTED]/
timecreated : 1096621321
avatar : https://steamcdn-a.akamaihd.net/steamcommunity/public/images/avatars/f9/[REDACTED].jpg
commentpermission : 1
avatarfull : https://steamcdn-a.akamaihd.net/steamcommunity/public/images/avatars/f9/[REDACTED]_
ull.jpg
avatarmedium : https://steamcdn-a.akamaihd.net/steamcommunity/public/images/avatars/f9/[REDACTED]_
medium.jpg
lastlogoff : 1519585688
```

You can also get...

1. User achievement
2. Owned Game
3. Friend List
4. Global Stats For Game

....

Please refer to the Steam Web API.

selenium

Step 1: Install

- **pip** install selenium
- PhantomJS
- <https://phantomjs.org/>
- Firefox browser driver:
- <https://github.com/mozilla/geckodriver/releases>
- Google chrome browser driver:
- <http://chromedriver.storage.googleapis.com/index.html?path=2.33/>

PhantomJS

```
from selenium import webdriver
driver = webdriver.PhantomJS(executable_path=r'請
輸入路徑') # PhantomJs
driver.get('https://www.zara.com/tw/en/') # 輸入範
例網址，交給瀏覽器
pageSource = driver.page_source # 取得網頁原始碼
print(pageSource) driver.close() # 關閉瀏覽器
```

Chrome

```
import time
from selenium import webdriver
from bs4 import BeautifulSoup
driver = webdriver.Chrome(executable_path=r'請輸入路徑') # chrome瀏覽器
time.sleep(3)
driver.get('https://hahow.in/courses')
for i in range(10): # 進行十次
    driver.execute_script('window.scrollTo(0, document.body.scrollHeight);') #
    重複往下捲動
    time.sleep(1) # 每次執行打瞌睡一秒
driver.close() # 關閉瀏覽器
```

Print Screen

```
from selenium import webdriver
driver = webdriver.Chrome(executable_path=r'請輸入路徑')
driver.get('http://www.pixiv.net/')
driver.save_screenshot('儲存位置/檔案名稱.png') # 保存截圖
driver.close()
```

Set header

```
headers =  
{'Accept': 'text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8',  
 'Accept-Encoding': 'gzip,deflate,sdch',  
 'Accept-Language': 'en,zh-CN;q=0.8,zh;q=0.6',  
 'Cache-Control': 'max-age=0',  
 'Host': 'www.xxx.com',    #此處為財經網的主頁  
 'Connection': 'keep-alive',  
 'Upgrade-Insecure-Requests': '1',  
 'Content-Type': 'application/x-www-form-urlencoded',  
 'User-Agent': 'Mozilla/5.0 (Windows NT 6.1; WOW64)  
 AppleWebKit/537.36 (KHTML, like Gecko) Chrome/34.0.1847.131  
 Safari/537.36'  
}  
  
response = requests.get("http://www.xxxxxx.com", headers=headers) #請求的地址
```

How to find header?

- <https://ithelp.ithome.com.tw/articles/10191165>

The screenshot shows the Network tab in the Chrome DevTools interface. A single request named 'doc/' is listed. The 'Headers' tab is selected, displaying the following details:

Server: Apache/2.4.18 (Ubuntu) OpenSSL/1.0.2g mod_wsgi/4.3.0 Python/2.7.12
Vary: Accept-Encoding

Request Headers (view source)

- Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8
- Accept-Encoding: gzip, deflate, br
- Accept-Language: en-US,en;q=0.8
- Cache-Control: max-age=0
- Connection: keep-alive
- Host: www.crummy.com
- If-Modified-Since: Sun, 07 May 2017 14:01:53 GMT
- If-None-Match: "448df-54eef9092e5cd-gzip"
- Referer: https://www.google.com.tw/
- Upgrade-Insecure-Requests: 1
- User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/59.0.3071.115 Safari/537.36

At the bottom left, it says '1 / 6 requests | 39.9 KB / 39.9 ...'