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?

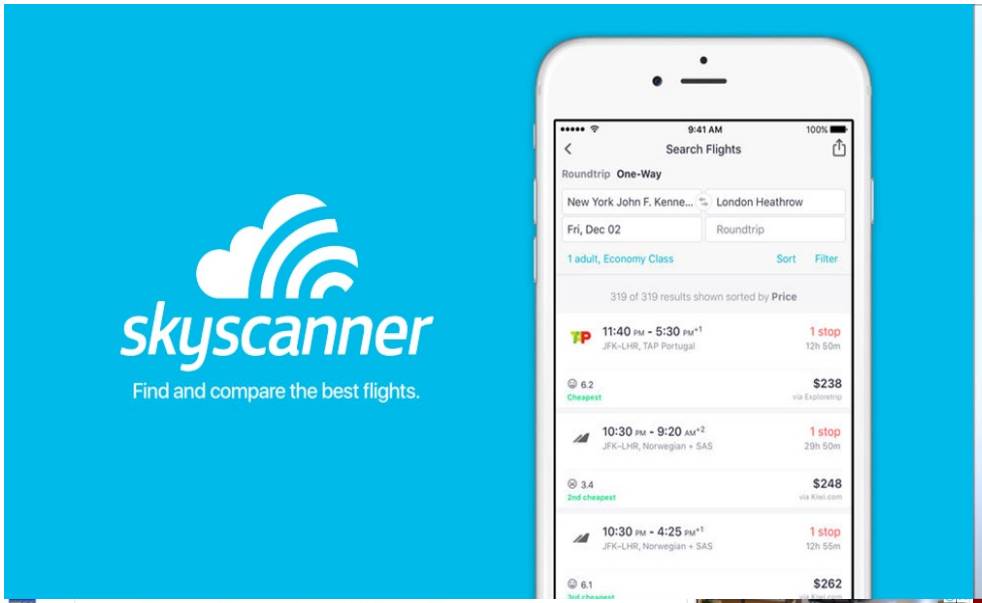
- Also called “Spider”
- 透過程式去追蹤網頁上的超連結，然後不斷往外擴張，以便將全世界中曾經被連結到的網頁全部都抓回到來，這也是 Google、 Yahoo 等網站背後最重要的程式之一

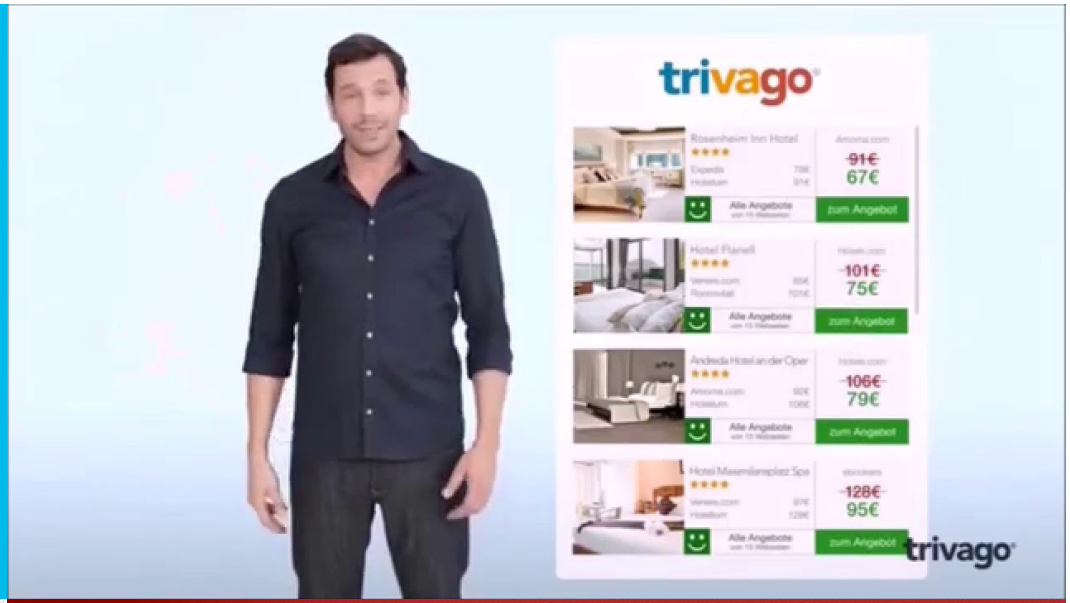
What is crawler?

- 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！


skyscanner
Find and compare the best flights.
9:41 AM
100%
Search Flights
Roundtrip One-Way
New York John F. Kenne...
London Heathrow
Fri, Dec 02
Roundtrip
1 adult, Economy Class
Sort
Filter
319 of 319 results shown sorted by Price
11:40 PM - 5:30 PM+1
JFK–LHR, TAP Portugal
1 stop
12h 50m
6.2
Cheapest
$238
via Expedia
10:30 PM - 9:20 AM+2
JFK–LHR, Norwegian + SAS
1 stop
29h 50m
3.4
2nd cheapest
$248
via Kiwi.com
10:30 PM - 4:25 PM+1
JFK–LHR, Norwegian + SAS
1 stop
12h 55m
6.1
$262


trivago
91€
67€
101€
75€
106€
79€
128€
95€
zum Angebot
trivago


韓國 Ryoe 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 Ryoe 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 Ryoe 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 Ryoe 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 Ryoe 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 Ryoe 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-唯你自在的生活。擁有更美好的自己。
改 - 你 開 始。
沐 / 潤
瞭解詳情
myfone購物 - 今日
$799
Acer SF514-52T-51AA 14吋筆電(i5-8250U/512G/金/福


網路
溫度計

ZARA

COLLECTION 19

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

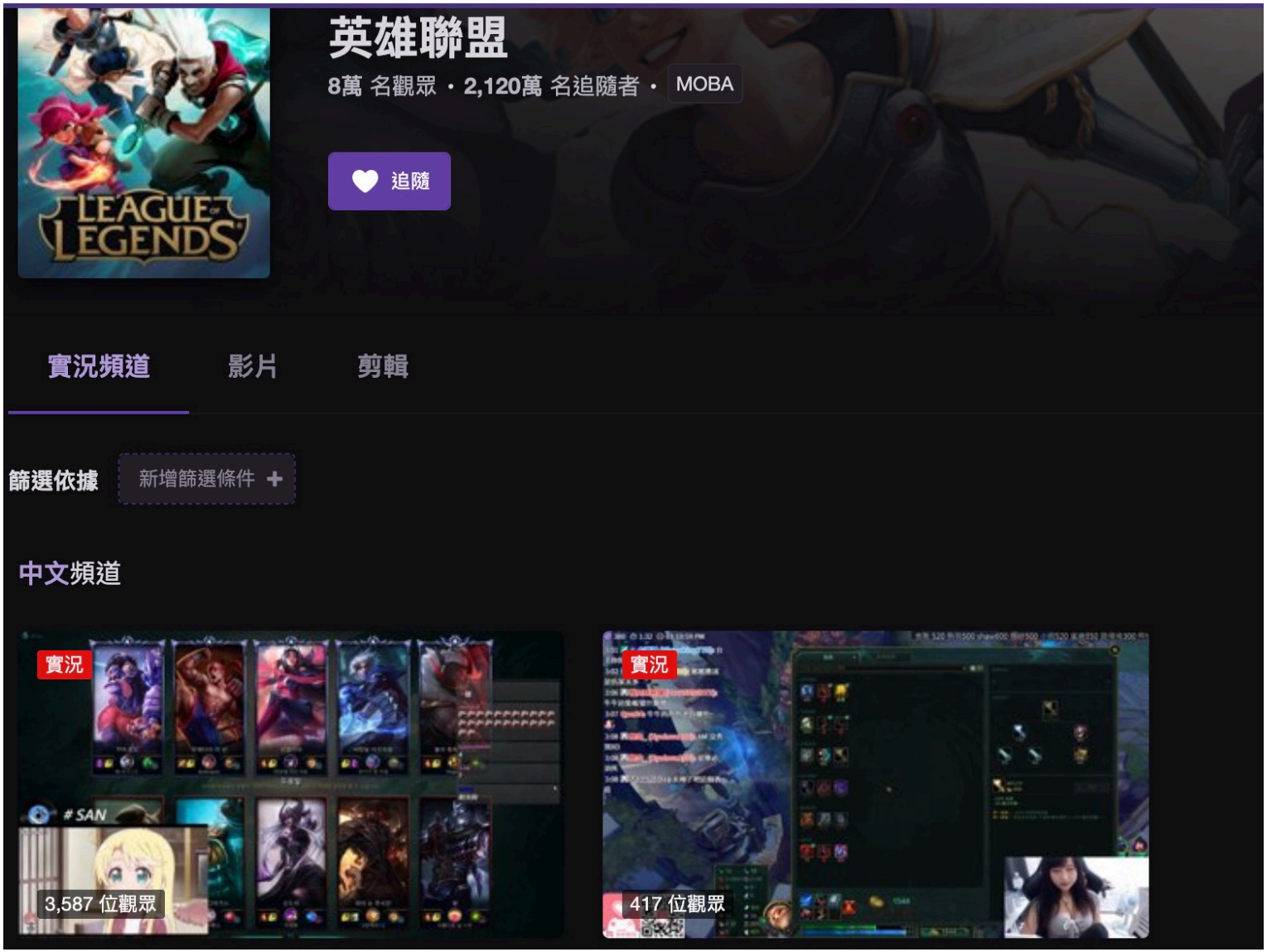
M

L — Coming Soon

XL

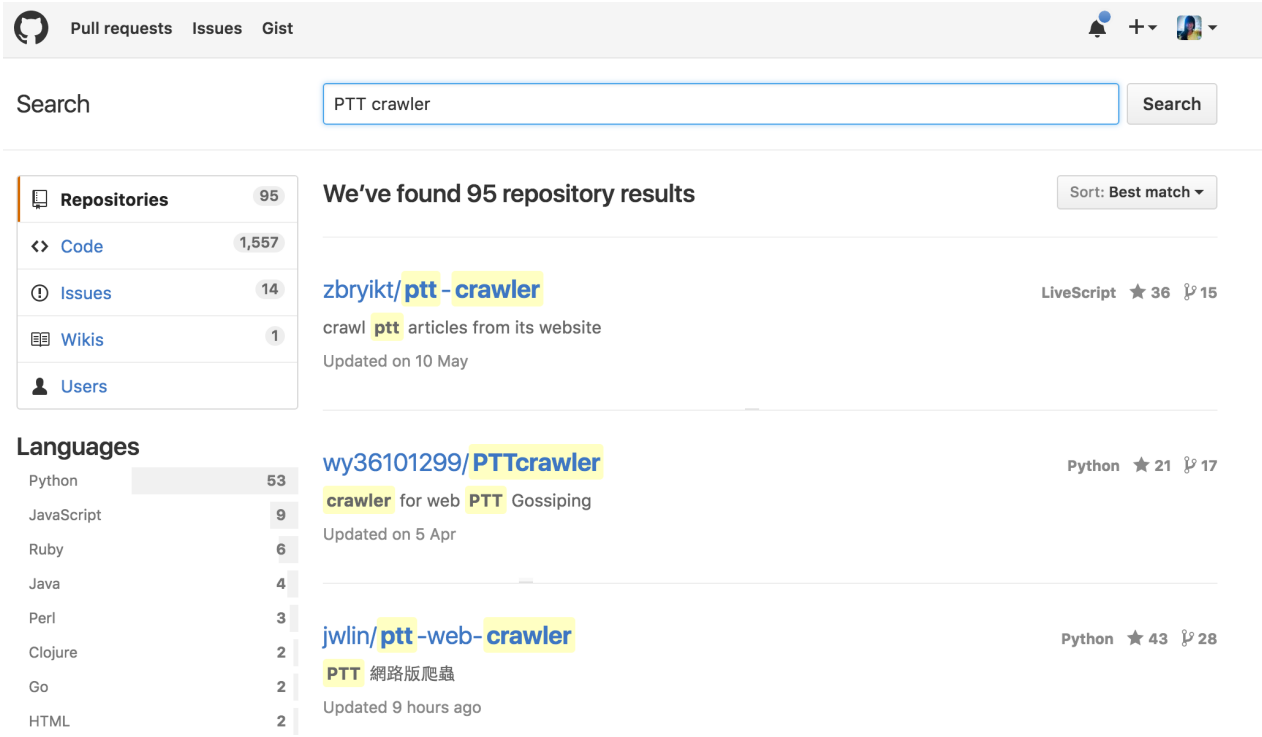
We'll let you know when it's in stock

Twitch


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)



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": 噓文數,
        "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
- 過濾資料內容中之空白、空行及特殊字元
- JSON 格式輸
- 支援 Python

輸出 JSON 格式

```
{
    "article
    "article
    "author"
    "board":
    "content
    "date":
    "ip": 發
    "message_conut": { # 推文
        "all": 總數,
        "boo": 噓文數,
        "count": 推文數-噓文數,
        "neutral": → 數,
        "push": 推文數
    },
    "messages": [ # 推文內容
```

執行方式

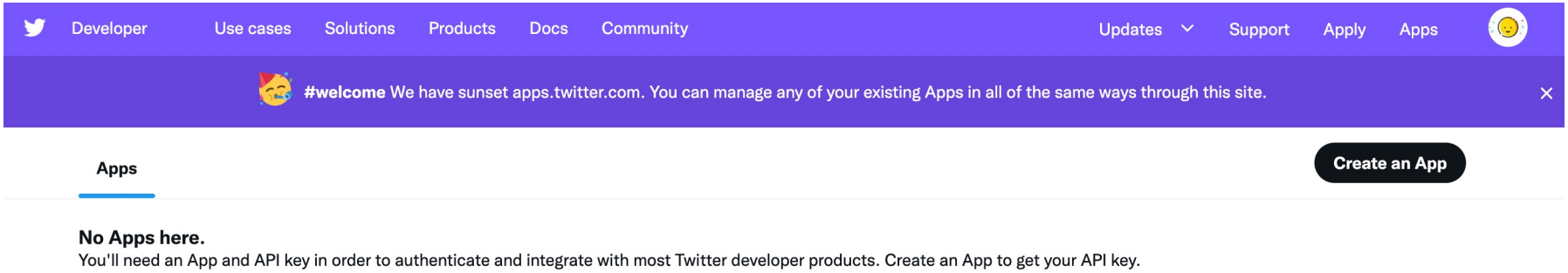
```
python crawler.py -b 看板名稱 -i 起始索引 結束索引 (設為 -1 則自動計算最後一頁)
python crawler.py -b 看板名稱 -a 文章ID
```

範例

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

How to do?

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



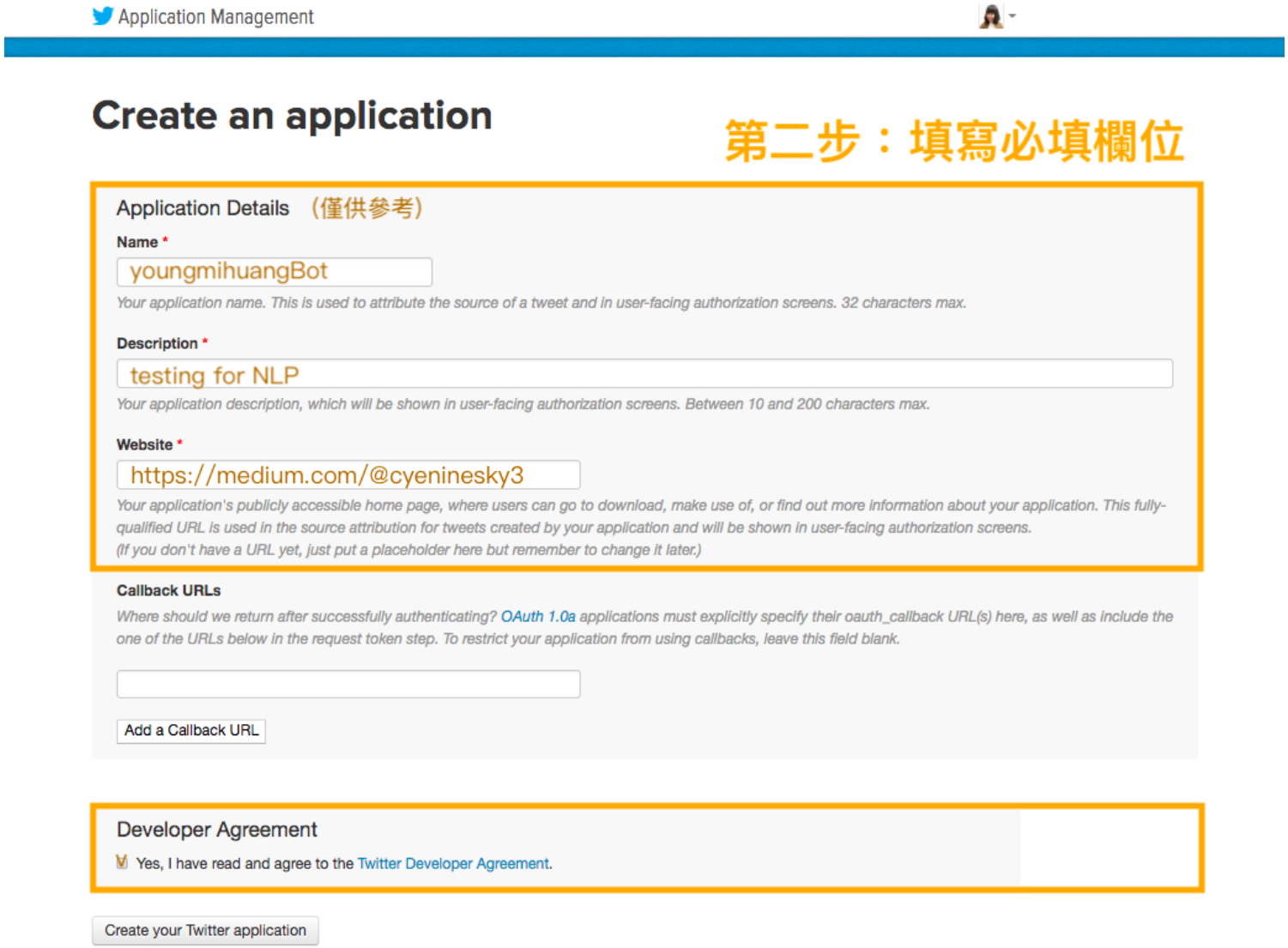
Developer Portal

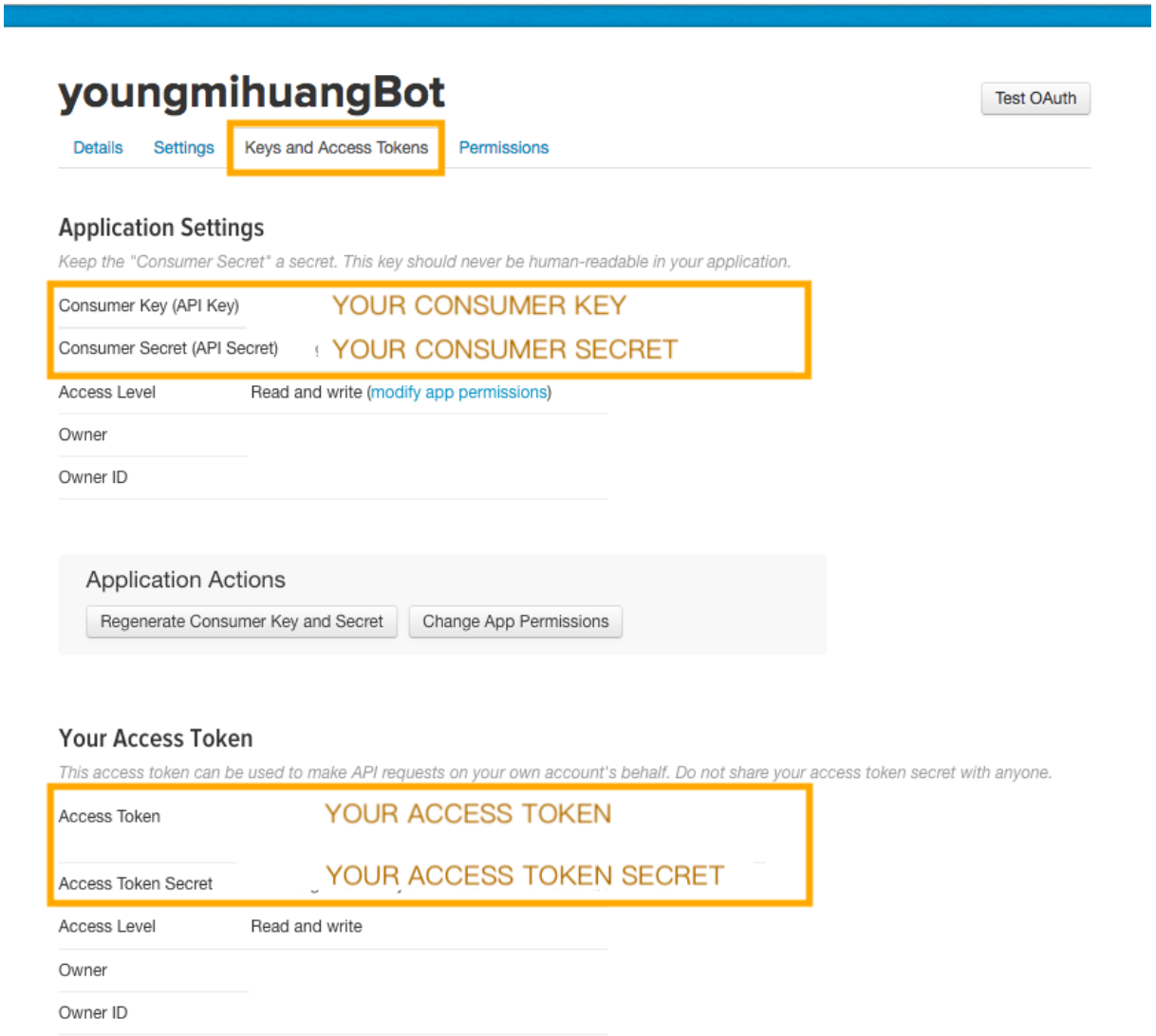
Docs Community Updates Support

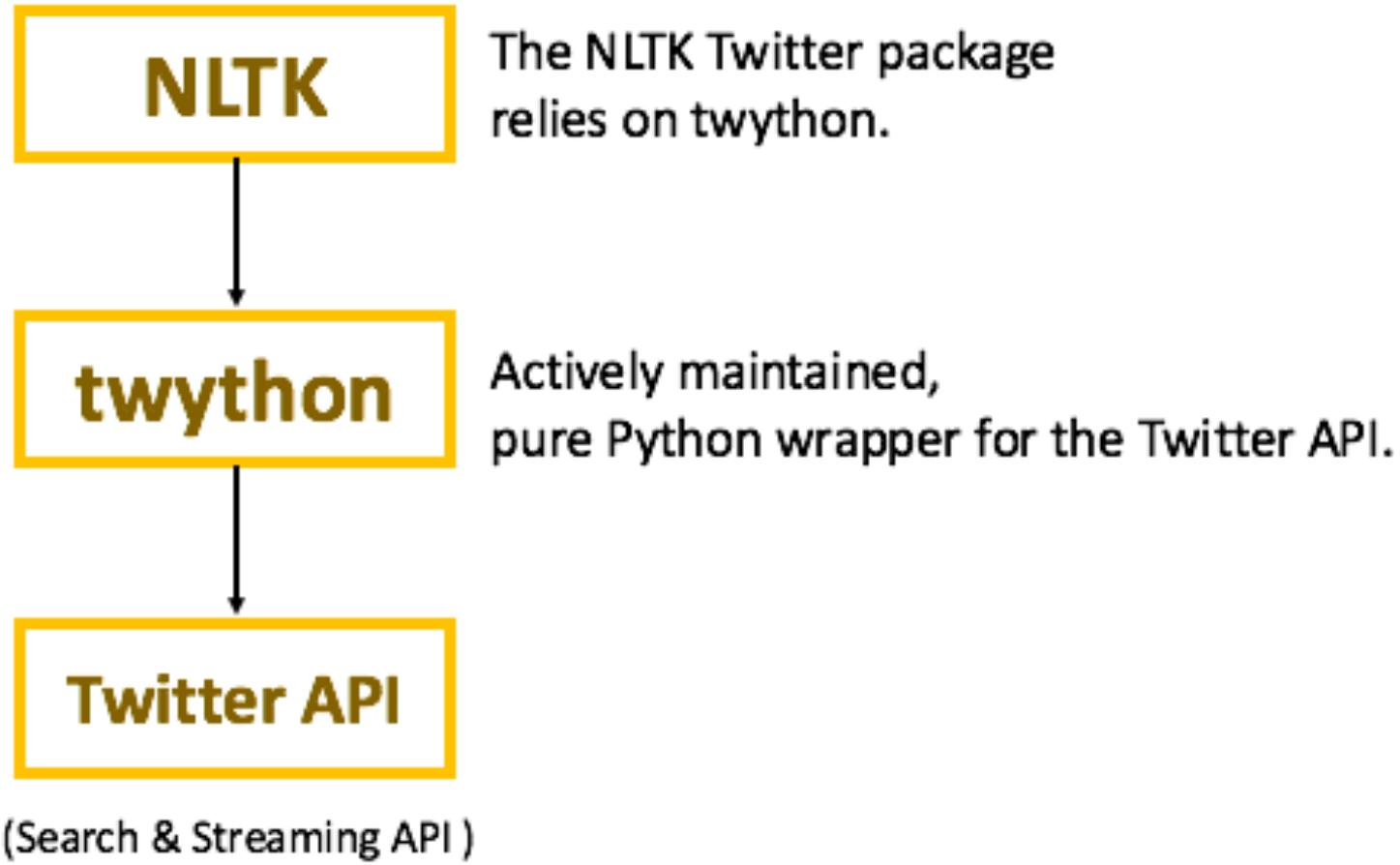
#ApplicationReceived

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.


Application Management
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



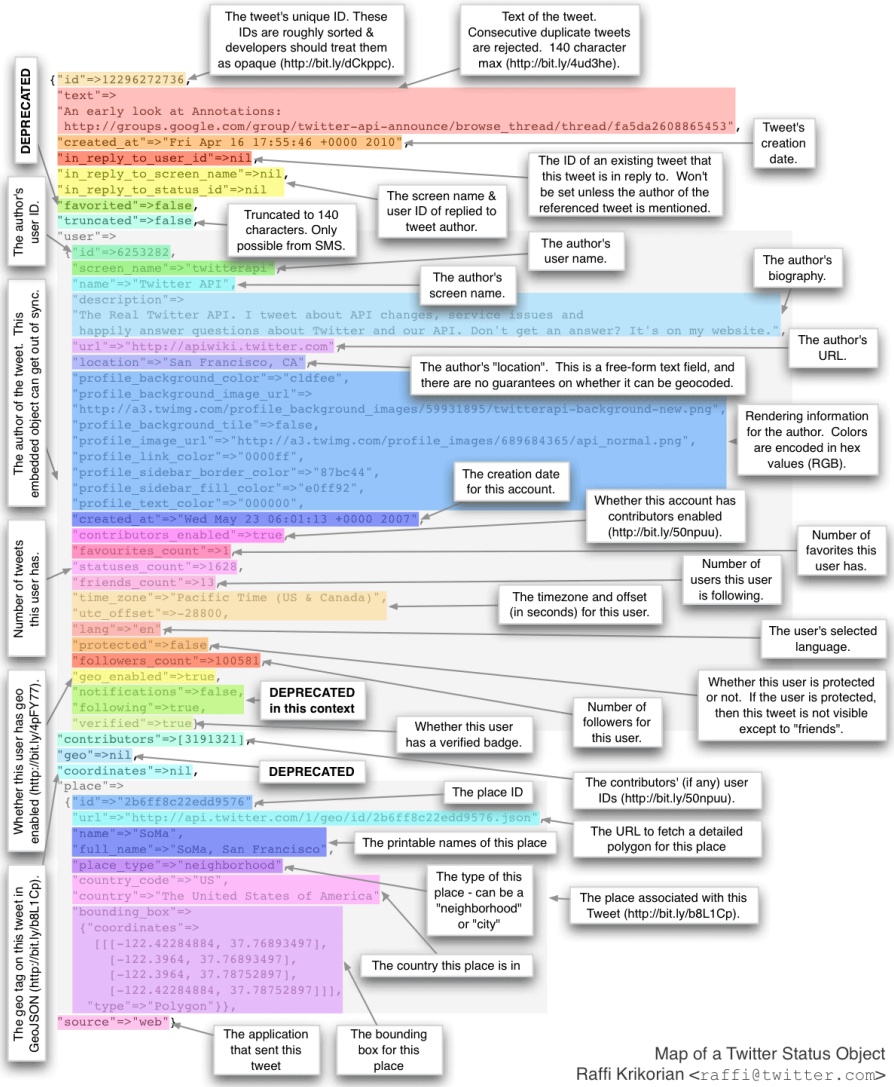
pip install twython

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 Desk
top 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 r
el="preconnect" href="//y.analytics.yahoo.com"><link rel="dns-prefetch" href="//geo.query.yahoo.com"><link rel="preco
nnect" href="//geo.query.yahoo.com"><link rel="dns-prefetch" href="//csc.beap.bc.yahoo.com"><link rel="preconnect" hr
ef="//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="dn
s-prefetch" href="//video-api.yql.yahoo.com"><link rel="preconnect" href="//video-api.yql.yahoo.com"><link rel="dns-p
refetch" href="//yrtas.btrll.com"><link rel="preconnect" href="//yrtas.btrll.com"><link rel="dns-prefetch" href="//sh
im.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

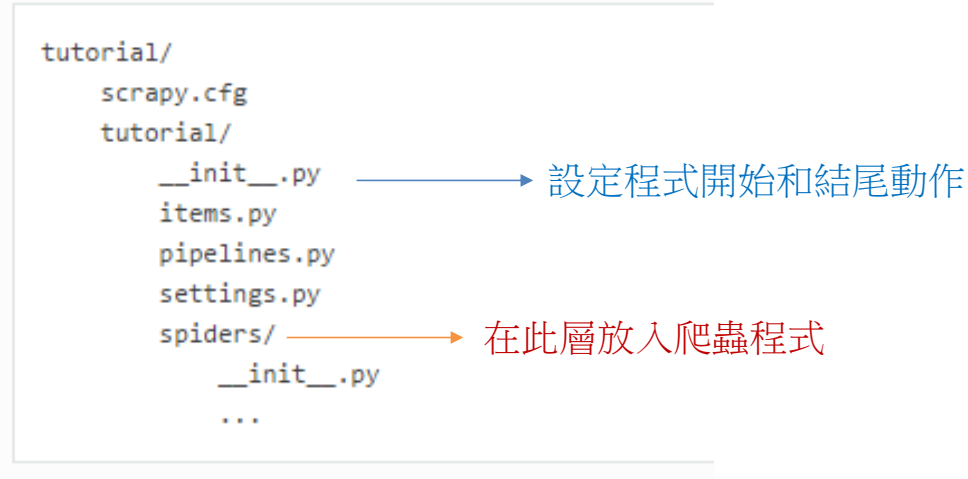
- 介紹

  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:



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, ffffan, fcman'
result = re.findall(pattern, test_string)
print(result)
```

['can', 'man', 'fan', 'ffffan', '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 “lllaaalaa” 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，修喔就鬼，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 te

- 找出有小數點的數字

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

```
['87.87']
```

- 找出email

```
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']
```

- 找出URL

```
import re
pat = 'http://[a-zA-Z0-9\./_]+'
text = '<a herf = "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 Ell">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 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 Ell">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')
```

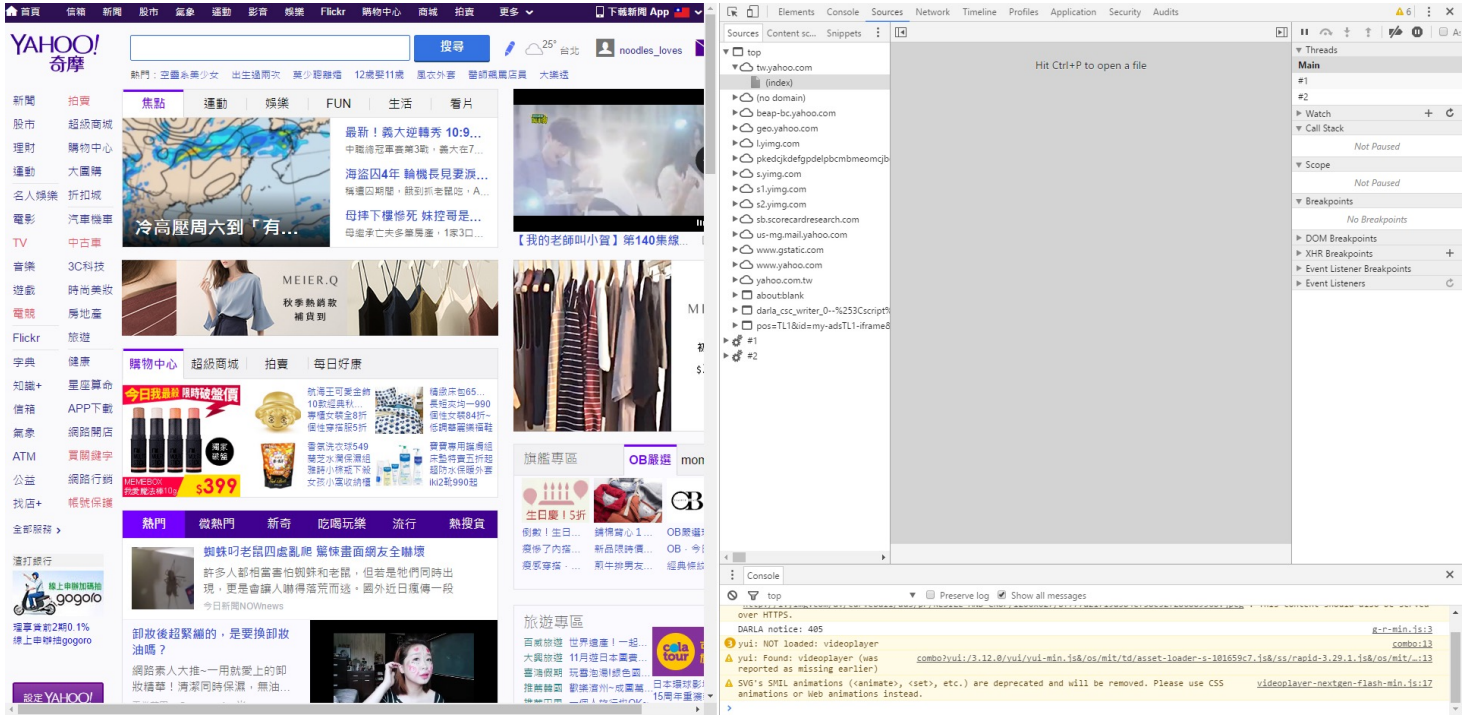
```
[<h2 class="Grid-U App-Title Pstart-12 Pos-r Z-1 Fz-l Pt-8 Mt-0 Mb-12">\u65d7\u
8266\u5c08\u5340</h2>, <h2 class="Grid-U App-Title Pstart-12 Pos-r Z-1 Fz-l Pt-
8 Mt-0 Mb-12">\u751f\u6d3b\u60c5\u5831</h2>, <h2 class="Grid-U App-Title Pstart
-12 Pos-r Z-1 Fz-l Pt-8 Mt-0 Mb-12">\u4eba\u6c23\u54c1\u724c</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-TextDecorat
ion" style="color:;">\u65b0\u805e</span>\n</a>\n</li>\n<li class="Fz-m MouseOve
r">\n<a class="drag-item D-b Td-n" href="https://tw.stock.yahoo.com/">\n<span c
lass="Cur-p D-ib Py-8 C-blue MouseOver-TextDecoration" style="color:;">\u80a1\u
```

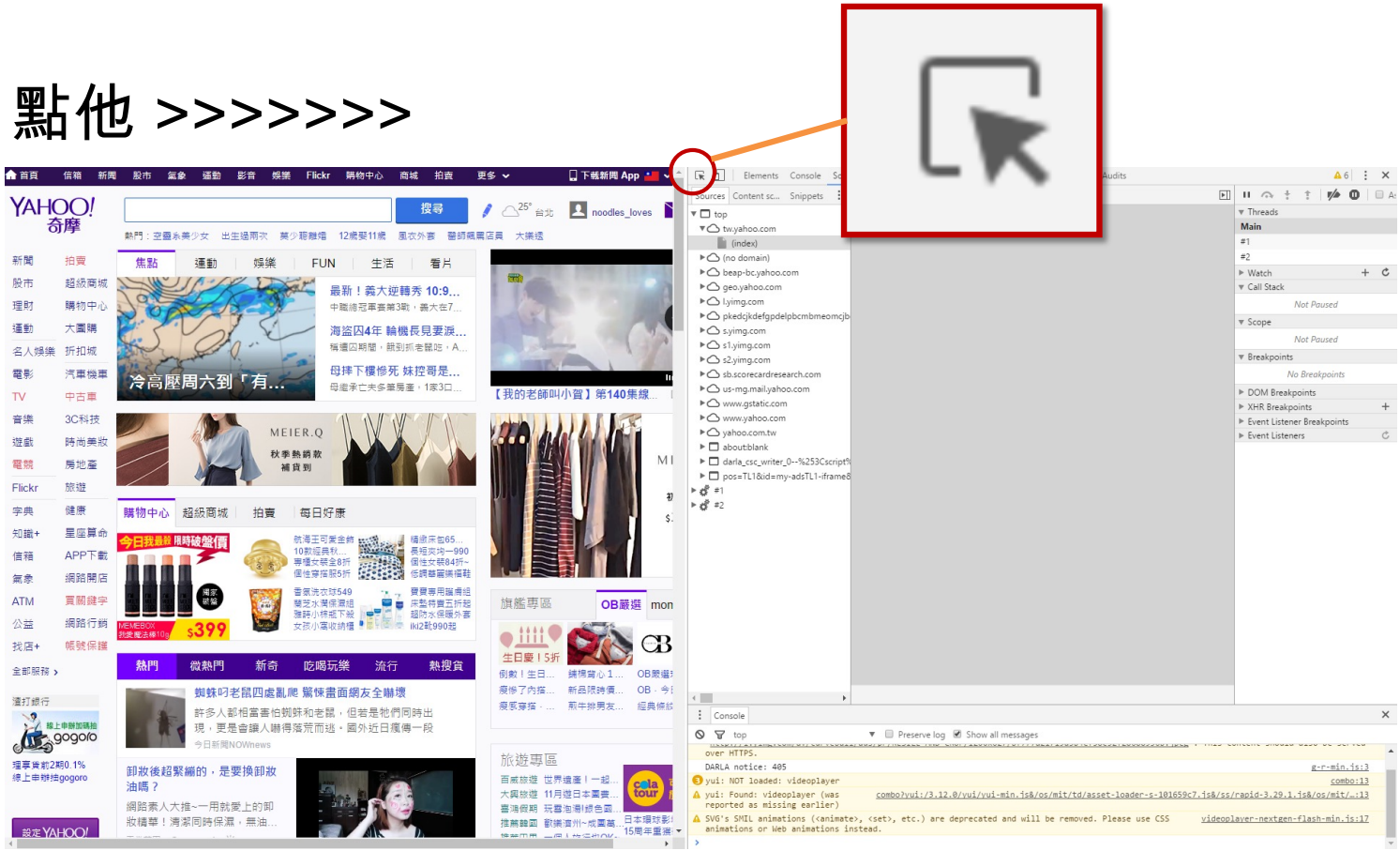
快速查看網頁元件資訊

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



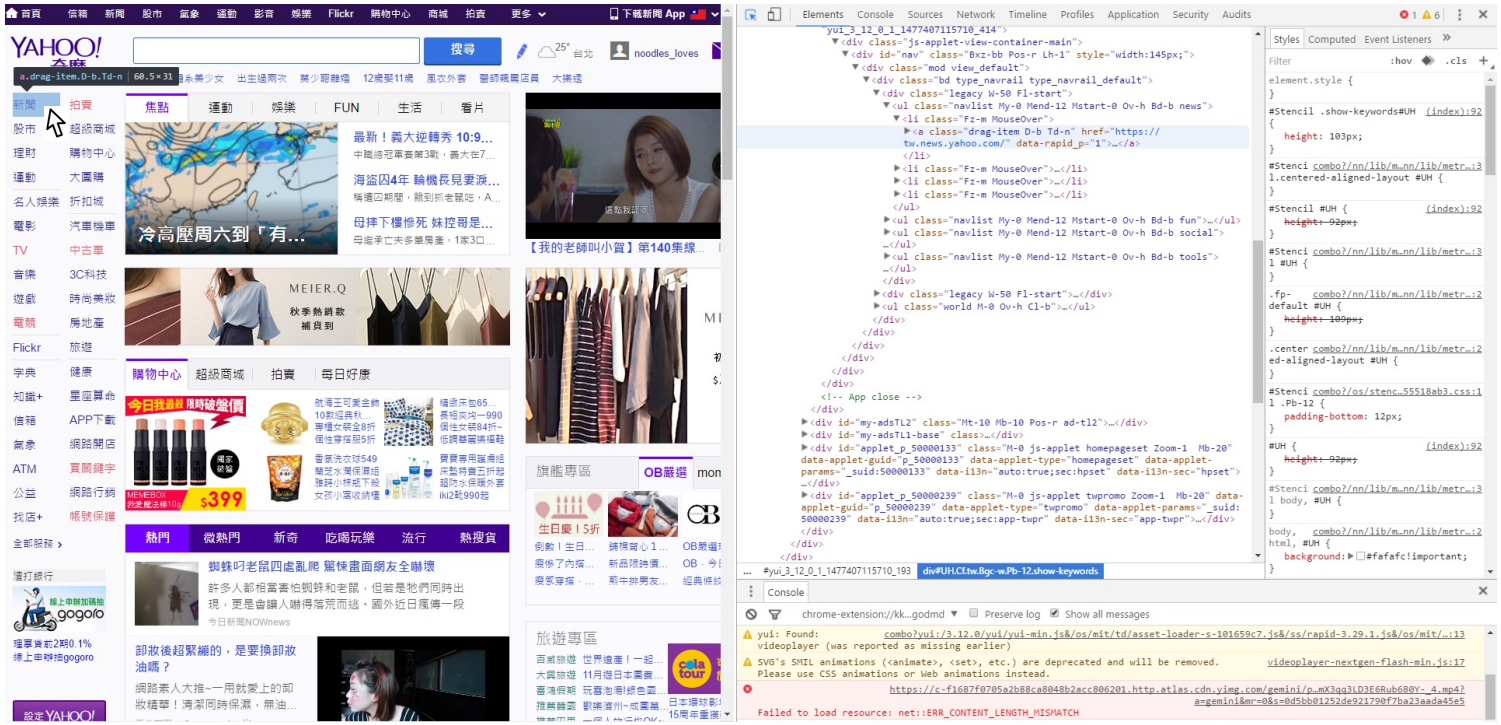
快速查看網頁元件資訊

- 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

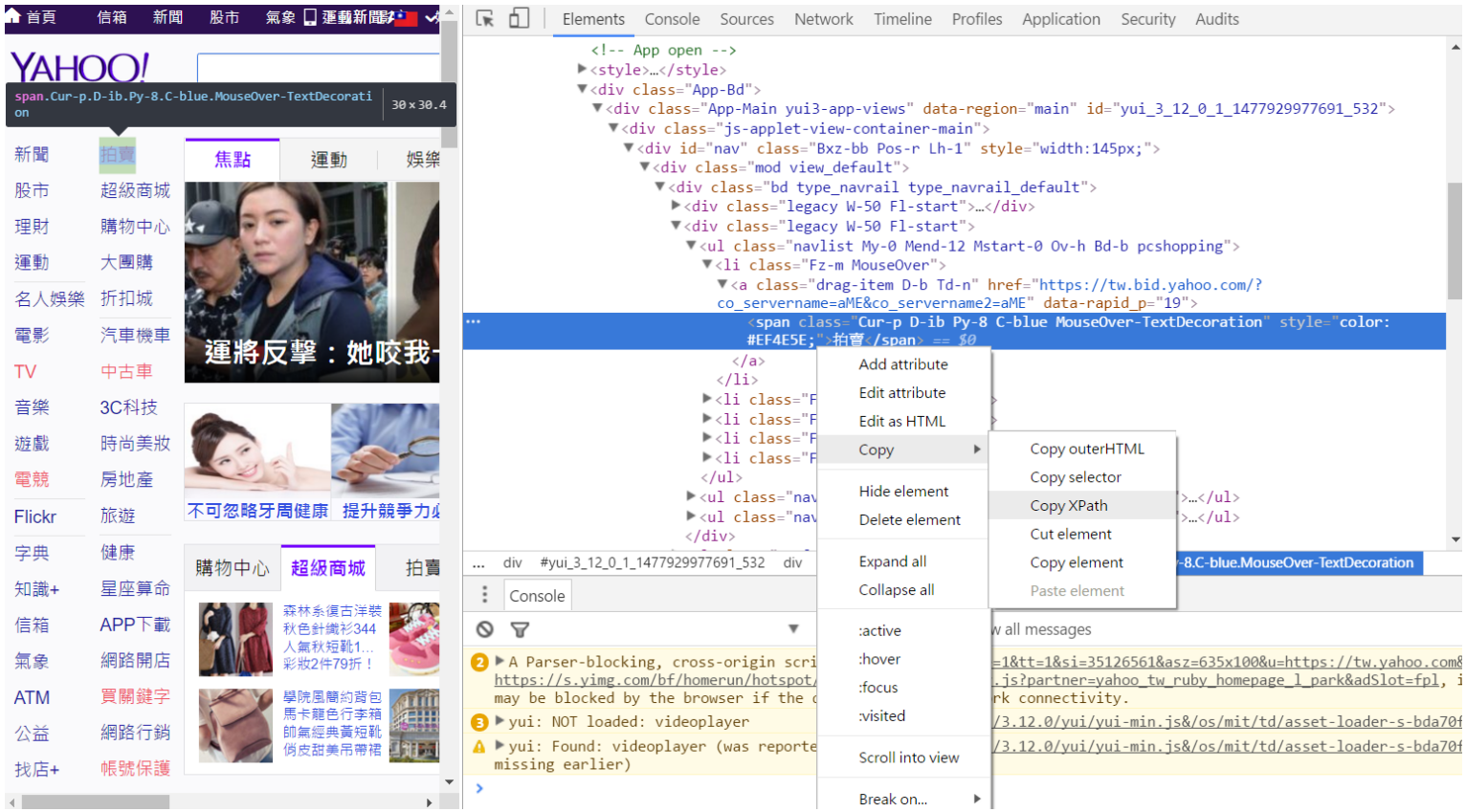
XPath

- 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:


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

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

XPath


首頁 信箱 新聞 股市 氣象 運動新聞
YAHOO!
span.Cur-p.D-ib.Py-8.C-blue.MouseOver-TextDecoration
30 × 30.4
新聞 拍賣
股市 超級商城
理財 購物中心
運動 大團購
名人娛樂 折扣城
電影 汽車機車
TV 中古車
音樂 3C科技
遊戲 時尚美妝
電競 房地產
Flickr 旅遊
字典 健康
知識+ 星座算命
信箱 APP下載
氣象 網路開店
ATM 買關鍵字
公益 網路行銷
找店+ 帳號保護
焦點 運動 娛樂
運將反擊：她咬我
不可忽略牙周健康 提升競爭力
購物中心 超級商城 拍賣
森林系復古洋裝
秋色針織衫344
人氣秋短靴1...
彩妝2件79折！
學院風簡約背包
馬卡龍色行李箱
帥氣經典黃短靴
俏皮甜美吊帶裙
Elements Console Sources Network Timeline Profiles Application Security Audits
<!-- App open -->
<style>…</style>

…

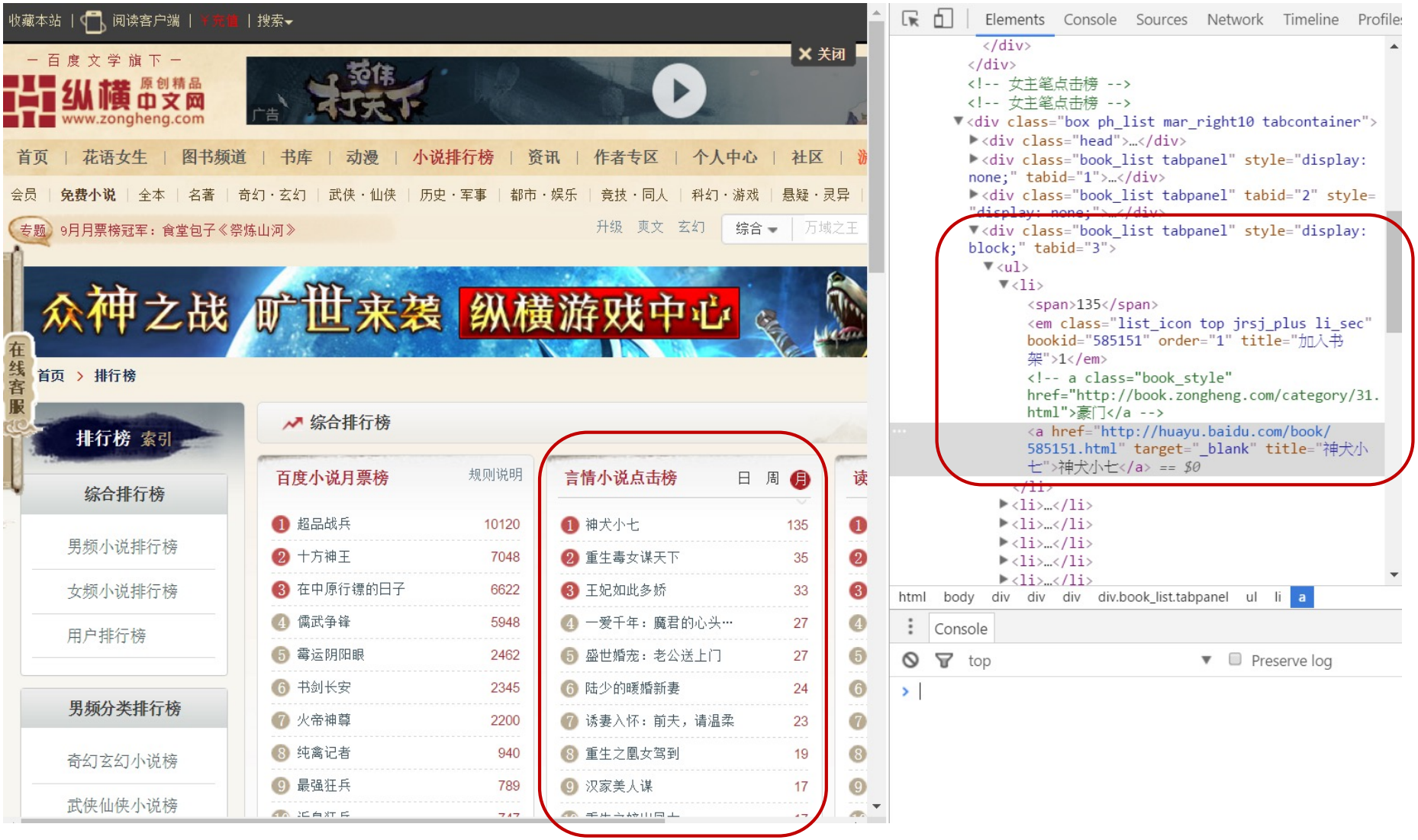
<ul class="navlist My-0 Mend-12 Mstart-0 Ov-h Bd-b pcshopping">
<li class="Fz-m MouseOver">
<a class="drag-item D-b Td-n" href="https://tw.bid.yahoo.com/?
co_servername=aME&co_servername2=aME" data-rapid_p="19">
<span class="Cur-p D-ib Py-8 C-blue MouseOver-TextDecoration" style="color:
#EF4E5E;">拍賣 == $0

Add attribute
Edit attribute
Edit as HTML
Copy
Hide element
Delete element
Expand all
Collapse all
:active
:hover
:focus
:visited
Scroll into view
Break on...
Copy outerHTML
Copy selector
Copy XPath
Cut element
Copy element
Paste element
div #yui_3_12_0_1_1477929977691_532 div
Console
A Parser-blocking, cross-origin scri
https://s.yimg.com/bf/homerun/hotspot/
may be blocked by the browser if the
yui: NOT loaded: videoplayer
yui: Found: videoplayer (was reporte
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/div[2]/ul[1]/li[1]/a/span/text()')
print p_text[0]
```

```
拍賣
```


收藏本站 | 阅读客户端 | ¥充值 | 搜索
关闭
纵横中文网
www.zongheng.com
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
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》
众神之战 旷世来袭 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
男频小说排行榜
女频小说排行榜
用户排行榜
男频分类排行榜
奇幻玄幻小说榜
武侠仙侠小说榜
综合排行榜
百度小说月票榜 规则说明
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 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 Profiles

<!-- 女主笔点击榜 -->
<!-- 女主笔点击榜 -->

…
…
…

135
1
<!-- a class="book_style" href="http://book.zongheng.com/category/31.html">豪门</a -->
神犬小七 == $0

…
…
…
…
…
html body div div div div.book_list.tabpanel ul li a
Console
top
Preserve log

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
花语女生网
huayu.baidu.com
追妻成狂
猎爱小军医
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:
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
连载中
总点击：792330
总字数：362083
总红票：5920
作品积分：1350
更新：2016-10-30
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
普通阅读
收藏本书
投红票
订阅VIP
本书共60条评论
[查看评论]
读者
本书
排名
Elements
Console
Sources
Network
Timeline
Profiles

…

…
…

普通阅读 == $0
…
…
…

<!-- 图片及简介 结束 -->
<script type="text/javascript">…</script>
<!-- 作品相关 结束 -->
<!-- cacheTime: 2016-11-01 11:53:57 677 -->
…
<!-- 小喇叭 开始 -->
…
<!-- 小喇叭 结束 -->
html body div div div.marr10 div.bookinfo div.wz div.button span.normal a
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
追妻成狂
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
第一章 宫翎是条狗！ 3585
第二章 夜半，宫小筱到访！ 3533
第三章 狗是我的第二种人格！ 3536
第四章 狗，是我内心深处的召唤！ 3165
第五章 就你，也配跟我抢女人？ 2255
第六章 宫翎霸道强吻！ 3729
第七章 女人，你当我摆设吗？ 3034
第八章 没有狗，怎么能称之为狗仔！ 2570
Elements
Console
Sources
Network
Timeline
Profiles

…

…

<!-- 正文章节 -->
…
<!-- 正文章节 结束 -->

<!-- 章节列表 -->

…

…

<a href="/chapter/585151/33339803.html"
target="_blank">第一章 宫翎是条狗！ ==
<a href="/chapter/585151/33339803.html
class="normal" target="_blank">

3585
2016-07-27
05:13:26

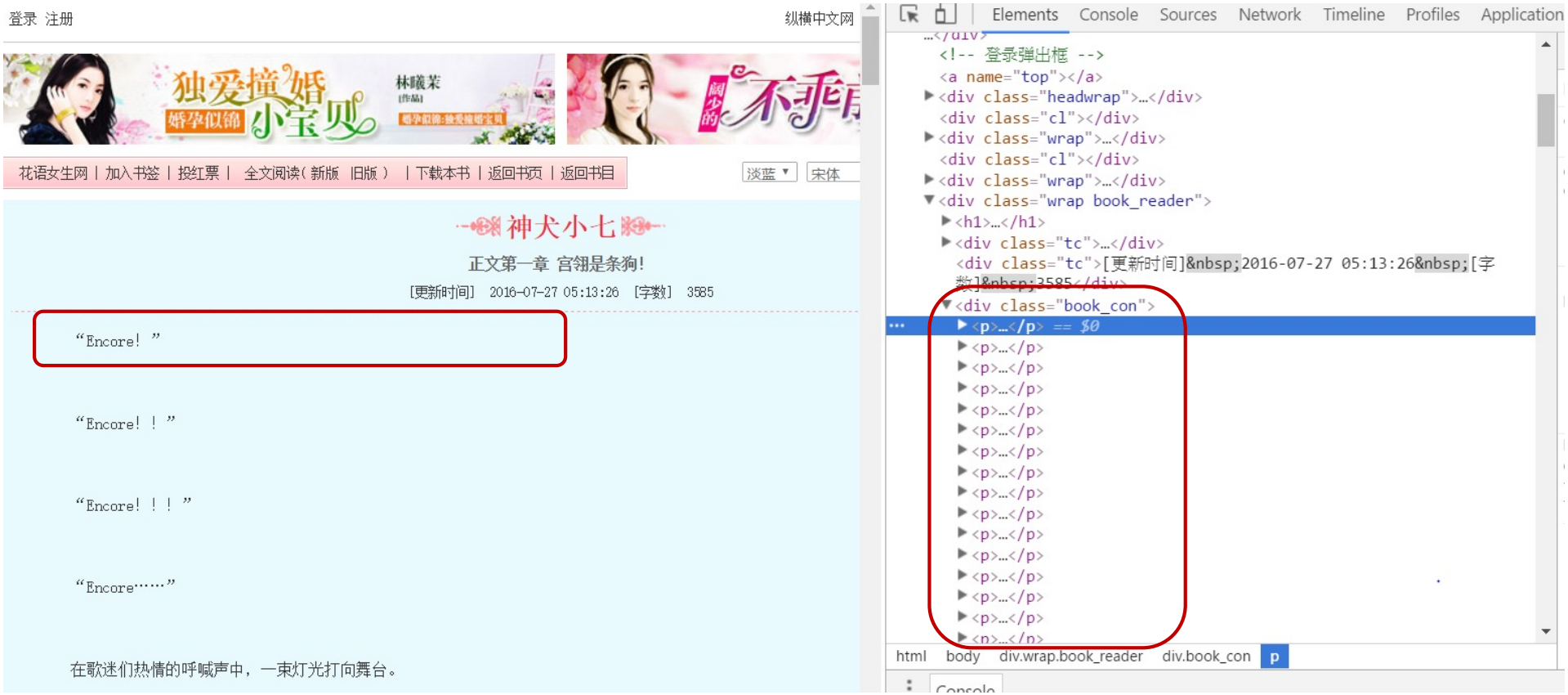
html body div div div div ul li span.chapname a

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 宫翎是条狗！</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，宫小筱到访！</a>
<a class="normal" href="/chapter/585151/33346746.html" target="_blank">
```


登录 注册
纵横中文网
独爱撞婚 小宝贝
林曦茉
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
淡蓝
宋体
神犬小七
正文第一章 宫翎是条狗！
[更新时间] 2016-07-27 05:13:26 [字数] 3585
“Encore！”
“Encore！！”
“Encore！！！”
“Encore……”
在歌迷们热情的呼喊声中，一束灯光打向舞台。
Elements
Console
Sources
Network
Timeline
Profiles
Application
<!-- 登录弹出框 -->

…

…

…

<h1>…</h1>
…
[更新时间] 2016-07-27 05:13:26 [字数] 3585

… == $0
html
body
div.wrap.book_reader
div.book_con
p

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
```

```
<p>My mother has <span style="color:blue">blue
</span> eyes.</p>
```

```
“Encore ! ”

“Encore ! ! ”

“Encore ! ! ! ”
```

Scrapy + XPath

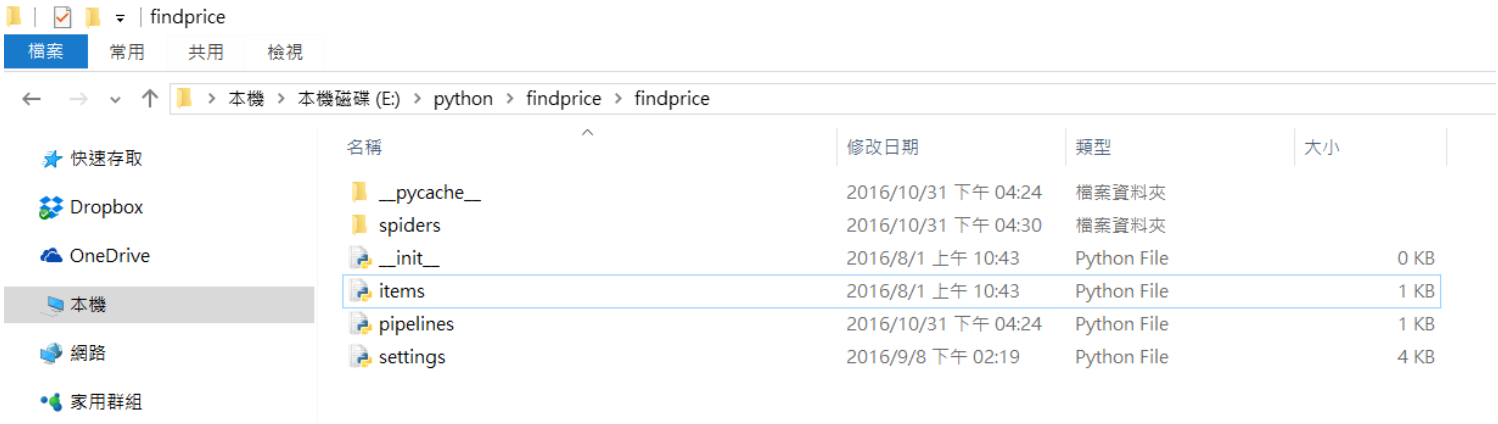
Findprice 實戰!!!

Findprice 實戰!!!

- 建專案

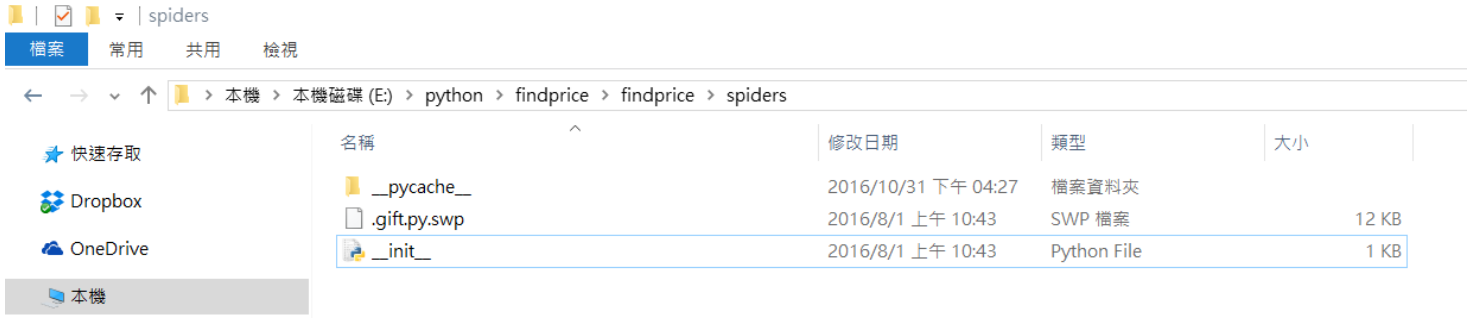
```
E:\python
λ scrapy startproject findprice
```

- 完成後會出現findprice資料夾

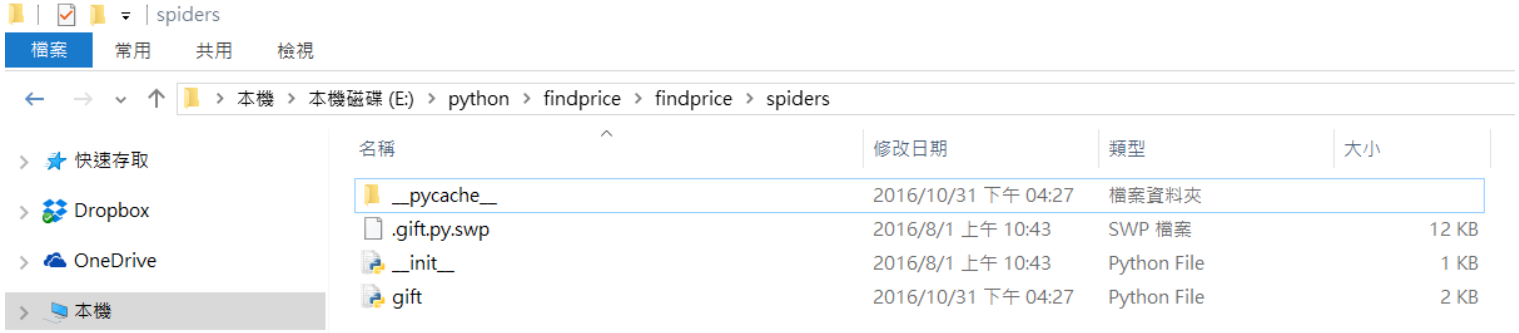


Findprice 實戰!!!

- 進入spiders資料夾



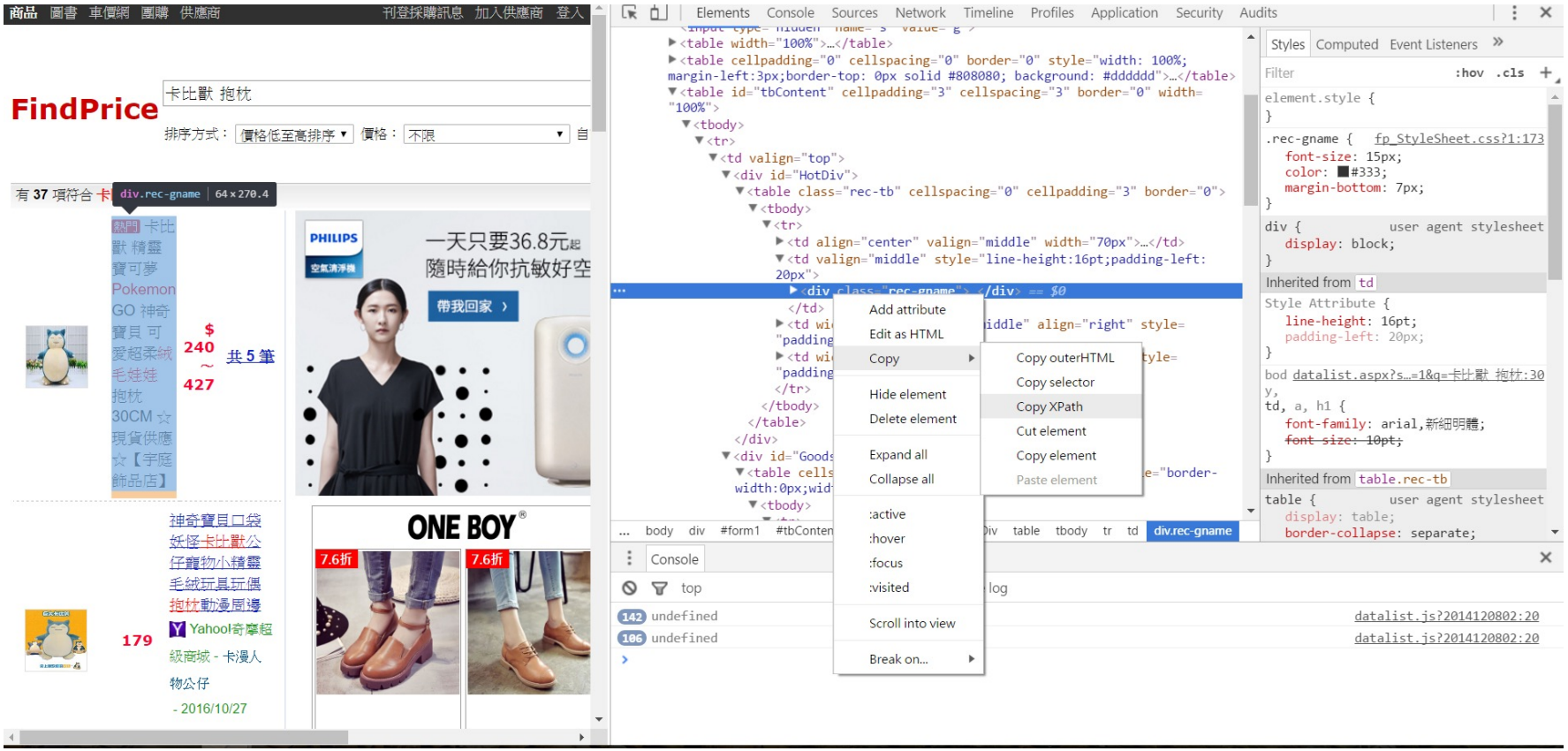
- 手動建一個.py檔



Gift.py

```python
# -*- coding: utf-8 -*-
import scrapy
import re
import sys

from scrapy.http import Request

class GiftSpider(scrapy.Spider):
    name = "gift"
    allowed_domains = ["findprice.com.tw"]

    gift_list = [
        u"卡比獸+抱枕",
        u"小火龍+娃娃"
    ]
    target_url = u'https://www.findprice.com.tw/datalist.aspx?s=g&q='

    start_urls = [
        target_url + gift_list[0],target_url + gift_list[1]
    ]
    print(start_urls)

    def start_requests(self):
        requests = []
        for item in self.start_urls:
            requests.append(Request(url=item, headers={'Referer': 'https://www.findprice.com.tw/'}))
        return requests

    def parse(self, response):
        pass
```

Copy Xpath


FindPrice
卡比獸 抱枕
排序方式：
價格低至高排序
價格：
不限
有 37 項符合
div.rec-gname | 64 x 270.4
PHILIPS
一天只要36.8元起
隨時給你抗敏好空
帶我回家
ONE BOY
7.6折
7.6折
Elements
Console
Sources
Network
Timeline
Profiles
Application
Security
Audits
Styles
Computed
Event Listeners
Add attribute
Edit as HTML
Copy
Hide element
Delete element
Expand all
Collapse all
:active
:hover
:focus
:visited
Scroll into view
Break on...
Copy outerHTML
Copy selector
Copy XPath
Cut element
Copy element
Paste element
Console
undefined
undefined

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
λ 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', 'SP
ue}
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:
['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.referer.RefererMiddleware',
 '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%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
λ 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', 'SP
ue}
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:
['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.referer.RefererMiddleware',
 '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
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)
```

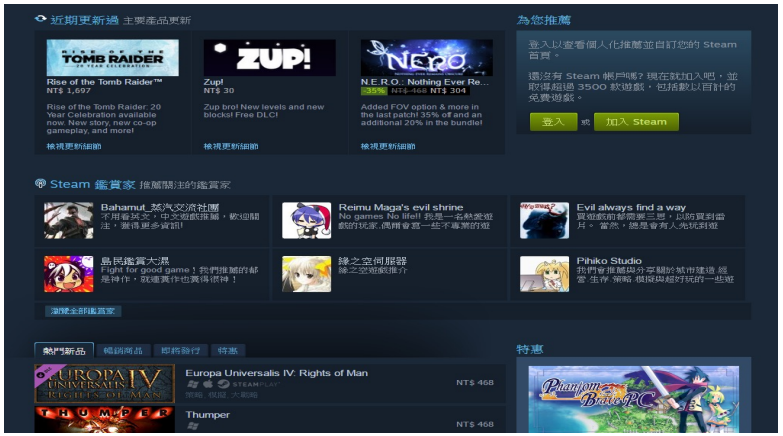
執行scrapy

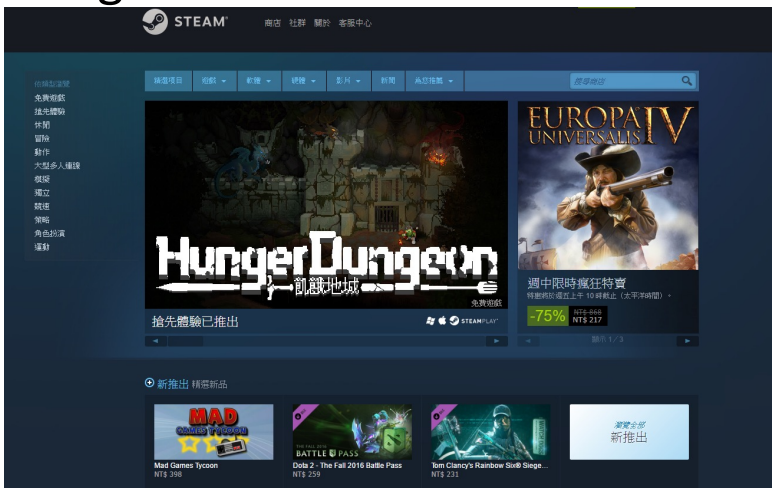
```
2016-10-31 16:52:42 [scrapy] INFO: Closing spider (finished)
2016-10-31 16:52:42 [scrapy] INFO: Dumping Scrapy stats:
{'downloader/request_bytes': 2768,
 'downloader/request_count': 7,
 'downloader/request_method_count/GET': 7,
 'downloader/response_bytes': 127969,
 'downloader/response_count': 7,
 '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,
 'request_depth_max': 3,
 'response_received_count': 7,
 'scheduler/dequeued': 6,
 'scheduler/dequeued/memory': 6,
 'scheduler/enqueued': 6,
 '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

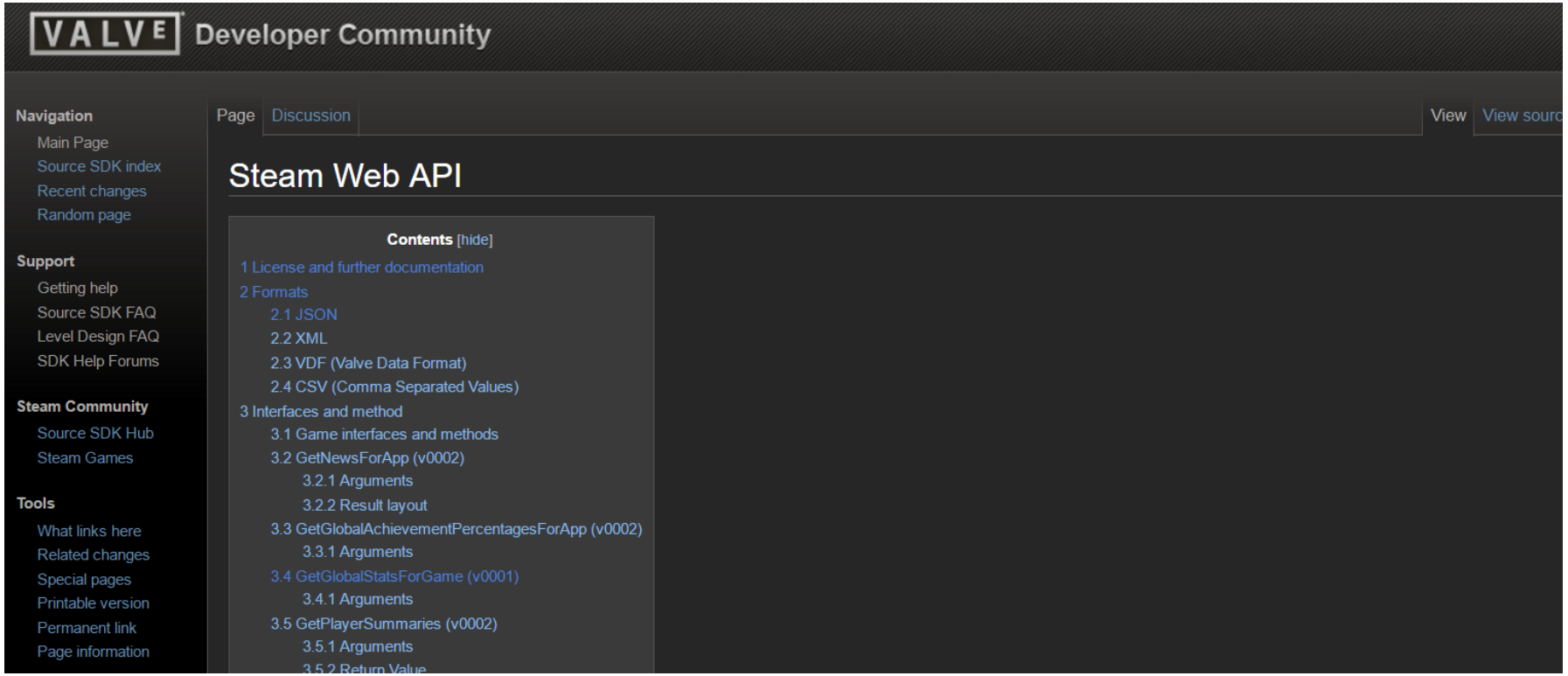
Steam is a digital distribution platform developed by Valve Corporation offering digital rights management (DRM), multiplayer gaming and social networking services. The Steam platform is considered to be the largest digital distribution platform for PC gaming



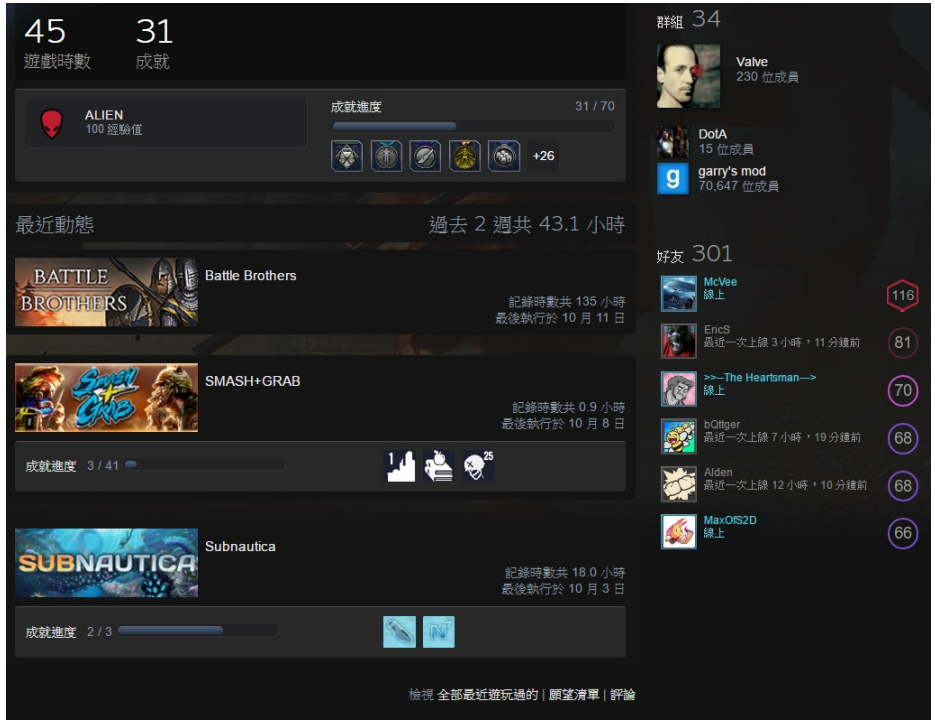


VALVE Developer Community

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



Get Player Summaries



Step

1.Apply the key

https://steamcommunity.com/dev

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

Step

2.Get User ID

http://steamid.co/

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/pu
                "avatarmedium": "https://steamcdn-a.akamaihd.net/steamcommu
                "avatarfull": "https://steamcdn-a.akamaihd.net/steamcommuni
                "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=“XXXXXXXX"
SteamId=“XXXXXXXX"

#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,imag
e/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