Python 3

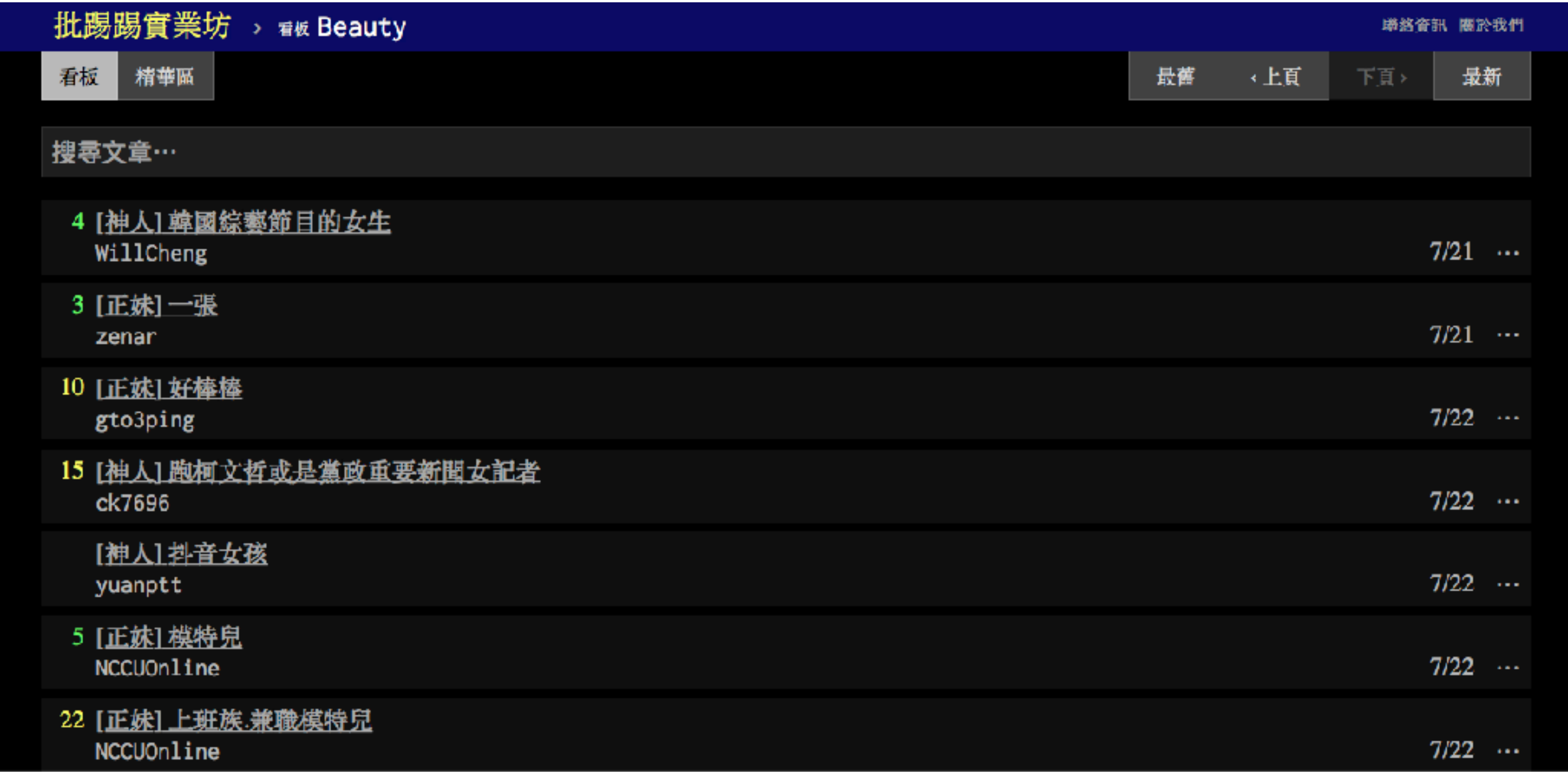
2018 0724

簡報製作：廖家誼

「今天老司機要帶你用Python爬表特板。」

當個愛好和平的肥宅，遠離八卦版。

https://www.ptt.cc/bbs/beauty/index.html



批踢踢實業坊 › 看板 Beauty

聯絡資訊 關於我們

4 [神人] 韓國綜藝節目的女生
WillCheng 7/21 ⋯

3 [正妹] 一張
zenar 7/21 ⋯

10 [正妹] 好棒棒
gto3ping 7/22 ⋯

15 [神人] 跑柯文哲或是黨政重要新聞女記者
ck7696 7/22 ⋯

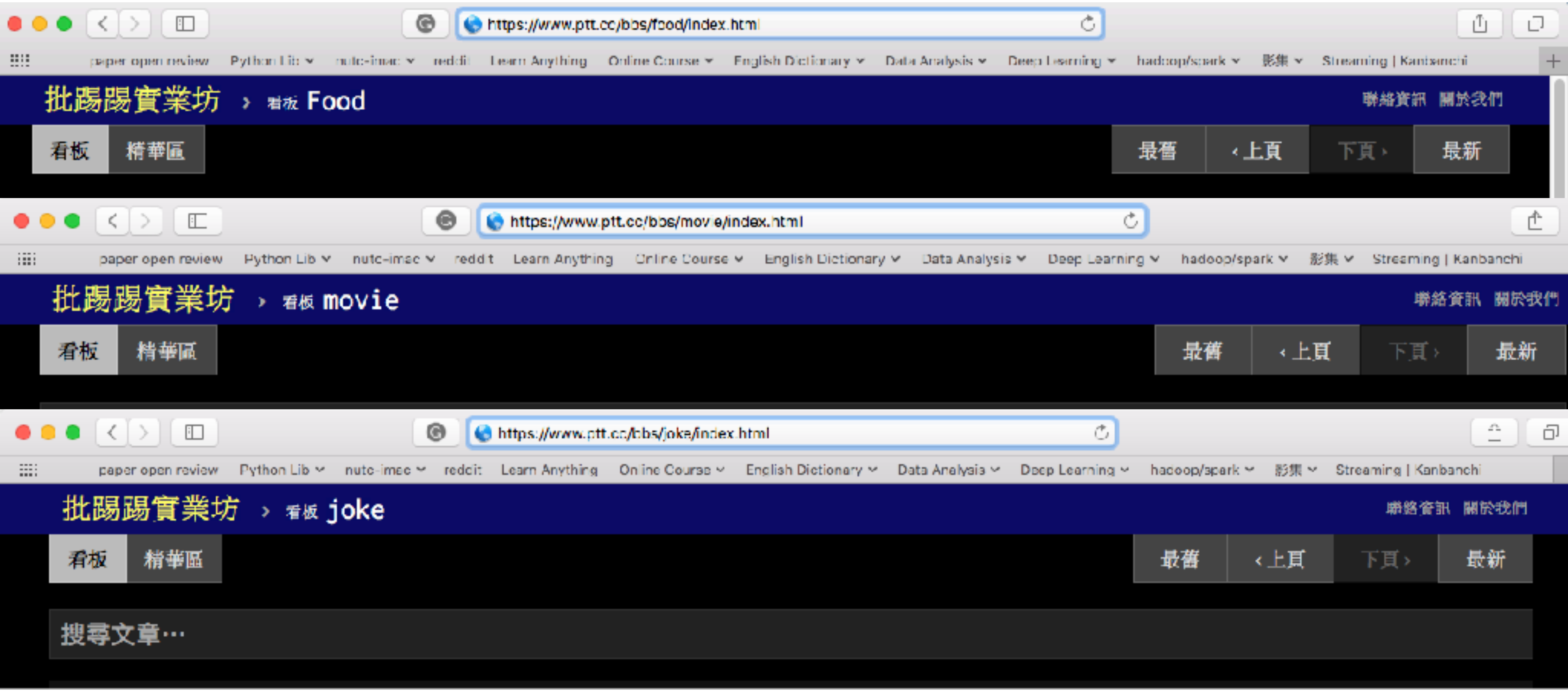
[神人] 抖音女孩
yuanptt 7/22 ⋯

5 [正妹] 模特兒
NCCUOnline 7/22 ⋯

22 [正妹] 上班族.兼職模特兒
NCCUOnline 7/22 ⋯

你會發現PTT的URL網址有規律：

https://www.ptt.cc/bbs/<看板名稱>/index.html



https://www.ptt.cc/bbs/food/index.html

https://www.ptt.cc/bbs/movie/index.html

https://www.ptt.cc/bbs/joke/index.html

總之呢，準備好一個你想爬而且有圖片可以抓的看板。

Before Starting

Before we starting, we have to install some Python libraries:

```
$ pip install requests
```

```
$ pip install bs4
```

```
$ pip install pillow
```

Review (1/3)

```
import requests
```

let’s try to get a webpage. For this example

```
r = requests.get('https://google.com')
```

`Response` object called `r`. We can get all the information we need from this object.

```
print("status: “,r.status_code)
print(r.url)
print(r.encoding)
```

We can pass parameters in URLs

```
payload = {"q":"python"}
r = requests.get(“https://www.google.com.tw/search?", params = payload)
```

Review (2/3)

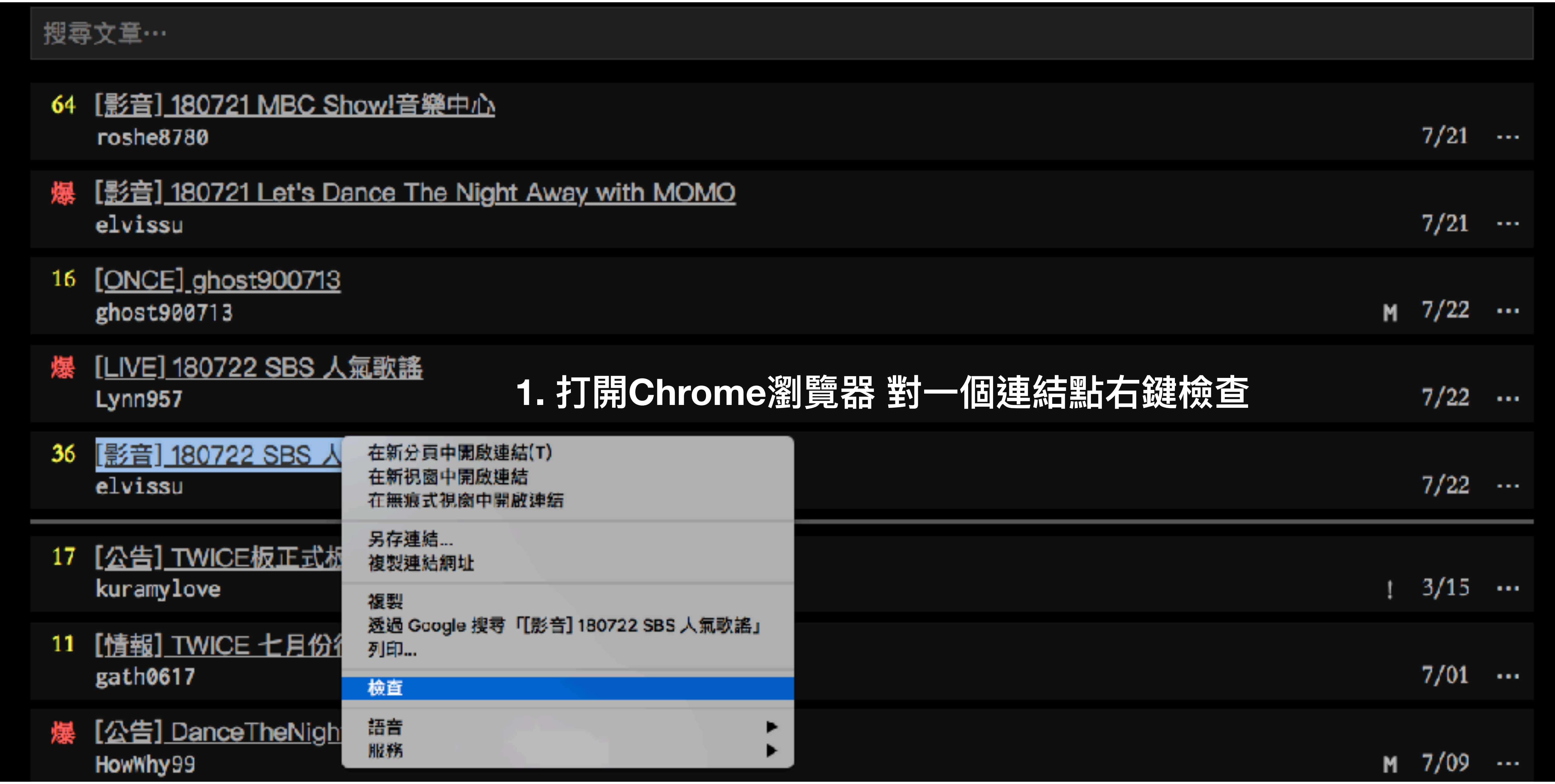
If you want to save a image from binary data returned by a request:

```
from PIL import Image
import requests
from io import BytesIO

r = requests.get(‘You_want_download_ImageUrl')
i = Image.open(BytesIO(r.content))
i.save(‘圖片名稱.副檔名’,'副檔名')
```

Review (3/3)

```
import requests
from bs4 import BeautifulSoup

r = requests.get(page_url)
soup = BeautifulSoup(r.text,”html.parser")


soup.findAll(‘a’)
soup.findAll(“div”,{"class":"title"})
soup.select(‘tag.class’)
soup.select(‘tag#id’)

Example :
soup.select(‘div.class_value > a')
```

Get Hrefs (1/3)



搜尋文章⋯

64 [影音] 180721 MBC Show!音樂中心
roshe8780 7/21 ⋯

爆 [影音] 180721 Let's Dance The Night Away with MOMO
elvissu 7/21 ⋯

16 [ONCE] ghost900713
ghost900713 M 7/22 ⋯

爆 [LIVE] 180722 SBS 人氣歌謠
Lynn957 7/22 ⋯

36 [影音] 180722 SBS 人
elvissu 7/22 ⋯

17 [公告] TWICE板正式板
kuramylove ! 3/15 ⋯

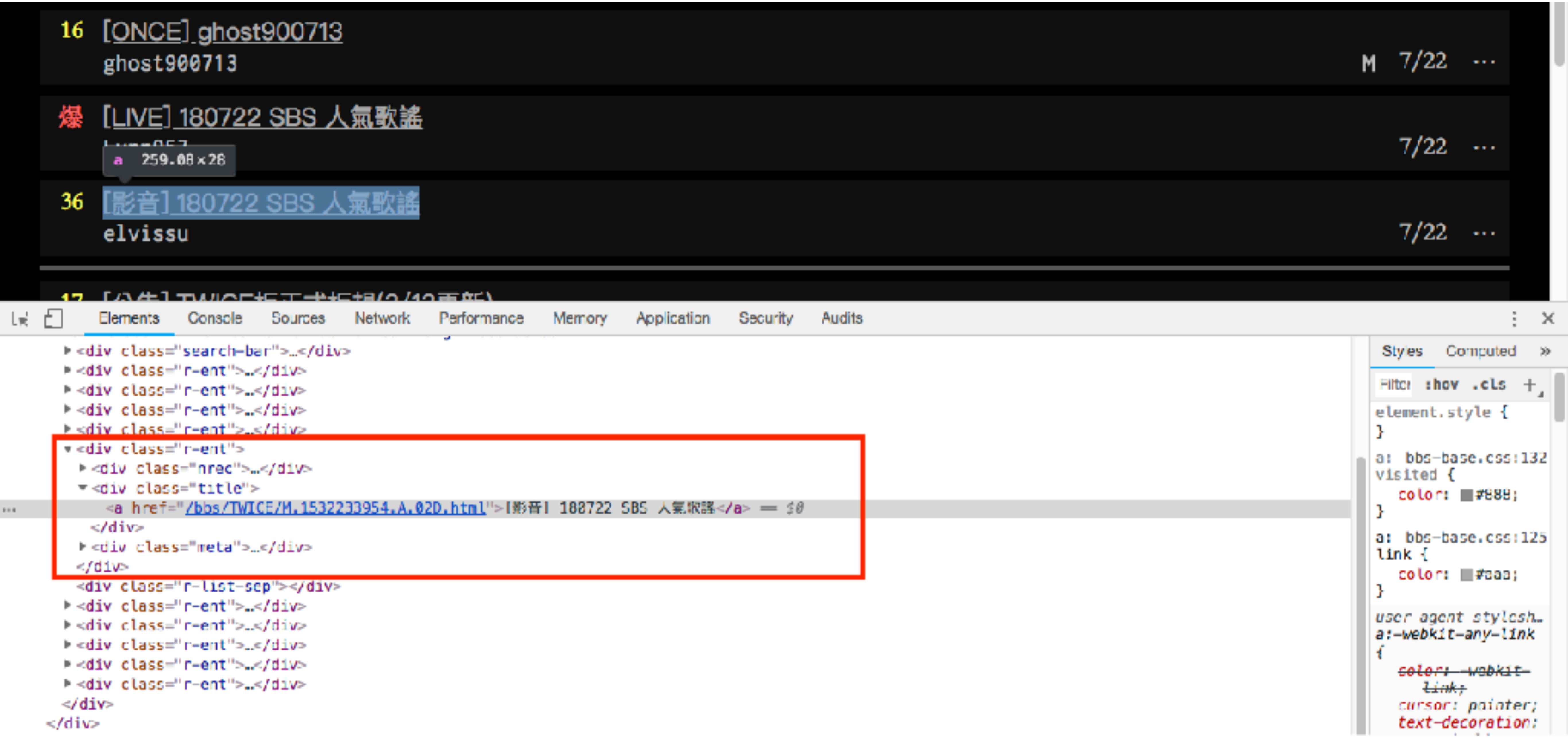
11 [情報] TWICE 七月份行
gath0617 7/01 ⋯

爆 [公告] DanceTheNigh
HowWhy99 M 7/09 ⋯

1. 打開Chrome瀏覽器 對一個連結點右鍵檢查

Get Hrefs (2/3)

可以看到連結（href）與標題：



16 [ONCE] ghost900713
ghost900713 M 7/22 ⋯

爆 [LIVE] 180722 SBS 人氣歌謠
7/22 ⋯

a 259.08 × 26

36 [影音] 180722 SBS 人氣歌謠
elvissu 7/22 ⋯

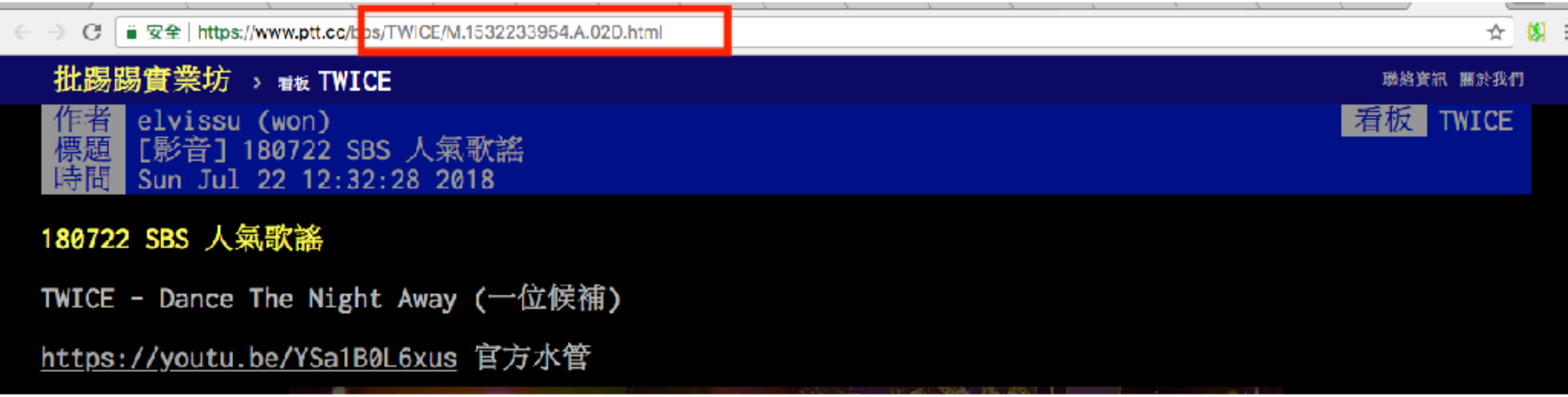
17 [公告] TWICE板正式板規(2/13更新)

```
▶<div class="search-bar">…</div>
▶<div class="r-ent">…</div>
▶<div class="r-ent">…</div>
▶<div class="r-ent">…</div>
▶<div class="r-ent">…</div>
▼<div class="r-ent">
  ▶<div class="nrec">…</div>
  ▼<div class="title">
      <a href="/bbs/TWICE/M.1532233954.A.02D.html">[影音] 180722 SBS 人氣歌謠</a> == $0
    </div>
  ▶<div class="meta">…</div>
 </div>
 <div class="r-list-sep"></div>
▶<div class="r-ent">…</div>
▶<div class="r-ent">…</div>
▶<div class="r-ent">…</div>
▶<div class="r-ent">…</div>
▶<div class="r-ent">…</div>
</div>
</div>
```

```
element.style {
}
a: bbs-base.css:132
visited {
   color: ■#888;
}
a: bbs-base.css:125
link {
   color: ■#aaa;
}
user agent stylesh…
a:-webkit-any-link
{
   ~~color: -webkit-link;~~
   cursor: pointer;
   text-decoration:
```

Get Hrefs (3/3)

點進來可以發現每篇文章的網址和剛剛看到的Href一樣：



安全 | https://www.ptt.cc/bbs/TWICE/M.1532233954.A.02D.html

作者 elvissu (won)
看板 TWICE
標題 [影音] 180722 SBS 人氣歌謠
時間 Sun Jul 22 12:32:28 2018

180722 SBS 人氣歌謠

TWICE - Dance The Night Away (一位候補)

https://youtu.be/YSa1B0L6xus 官方水管

這樣我們就知道要抓取每篇文章的href！

```
import requests
from bs4 import BeautifulSoup
article_href =[]
```

1. 建立空list存放所有href

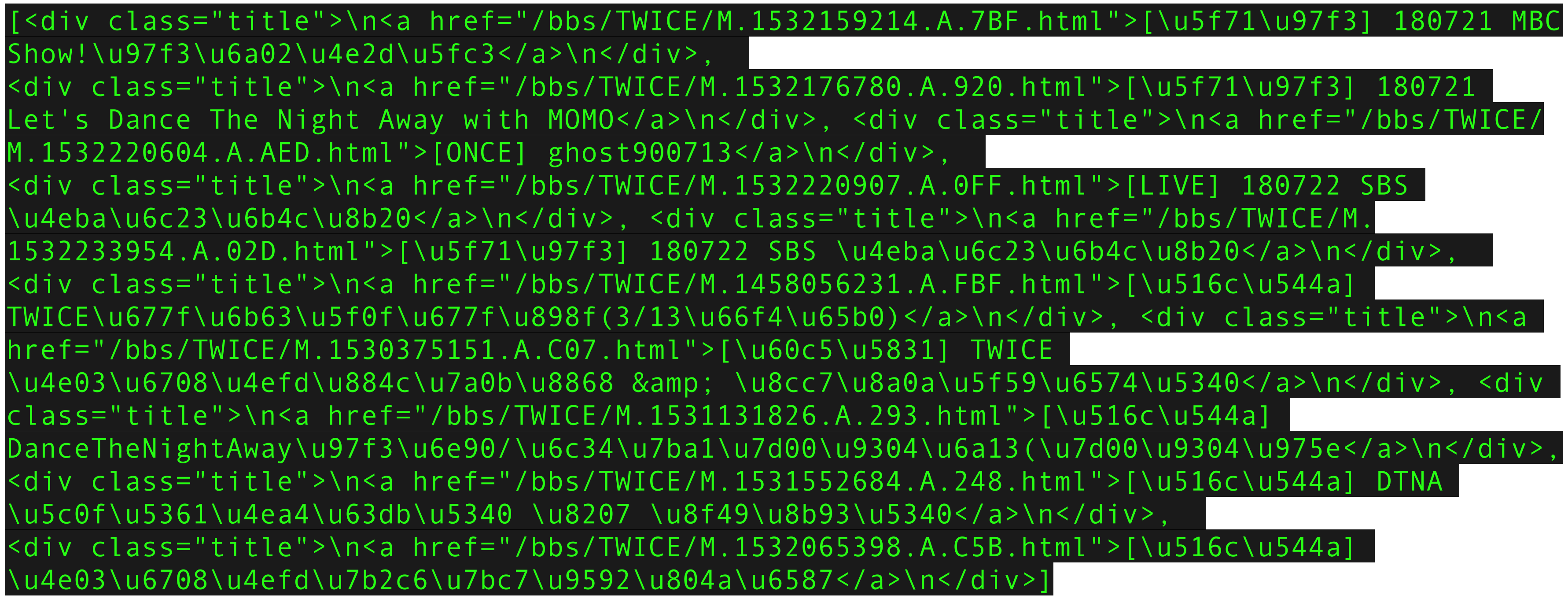
```
r = requests.get("https://www.ptt.cc/bbs/TWICE/index.html")
soup = BeautifulSoup(r.text,"html.parser")
```

2. 對該板的Url發送requests 將回傳物件（頁面）用bs4解析

```
results = soup.findAll("div",{"class":"title"})
```

3. 解析出所有’div’且class值為title的標籤

回傳的物件可以看到是一個list，每個div內包含a的標籤：



```
[<div class="title">\n<a href="/bbs/TWICE/M.1532159214.A.7BF.html">[\u5f71\u97f3] 180721 MBC
Show!\u97f3\u6a02\u4e2d\u5fc3</a>\n</div>,
<div class="title">\n<a href="/bbs/TWICE/M.1532176780.A.920.html">[\u5f71\u97f3] 180721
Let's Dance The Night Away with MOMO</a>\n</div>, <div class="title">\n<a href="/bbs/TWICE/
M.1532220604.A.AED.html">[ONCE] ghost900713</a>\n</div>,
<div class="title">\n<a href="/bbs/TWICE/M.1532220907.A.0FF.html">[LIVE] 180722 SBS
\u4eba\u6c23\u6b4c\u8b20</a>\n</div>, <div class="title">\n<a href="/bbs/TWICE/M.
1532233954.A.02D.html">[\u5f71\u97f3] 180722 SBS \u4eba\u6c23\u6b4c\u8b20</a>\n</div>,
<div class="title">\n<a href="/bbs/TWICE/M.1458056231.A.FBF.html">[\u516c\u544a]
TWICE\u677f\u6b63\u5f0f\u677f\u898f(3/13\u66f4\u65b0)</a>\n</div>, <div class="title">\n<a
href="/bbs/TWICE/M.1530375151.A.C07.html">[\u60c5\u5831] TWICE
\u4e03\u6708\u4efd\u884c\u7a0b\u8868 &amp; \u8cc7\u8a0a\u5f59\u6574\u5340</a>\n</div>, <div
class="title">\n<a href="/bbs/TWICE/M.1531131826.A.293.html">[\u516c\u544a]
DanceTheNightAway\u97f3\u6e90/\u6c34\u7ba1\u7d00\u9304\u6a13(\u7d00\u9304\u975e</a>\n</div>,
<div class="title">\n<a href="/bbs/TWICE/M.1531552684.A.248.html">[\u516c\u544a] DTNA
\u5c0f\u5361\u4ea4\u63db\u5340 \u8207 \u8f49\u8b93\u5340</a>\n</div>,
<div class="title">\n<a href="/bbs/TWICE/M.1532065398.A.C5B.html">[\u516c\u544a]
\u4e03\u6708\u4efd\u7b2c6\u7bc7\u9592\u804a\u6587</a>\n</div>]
```

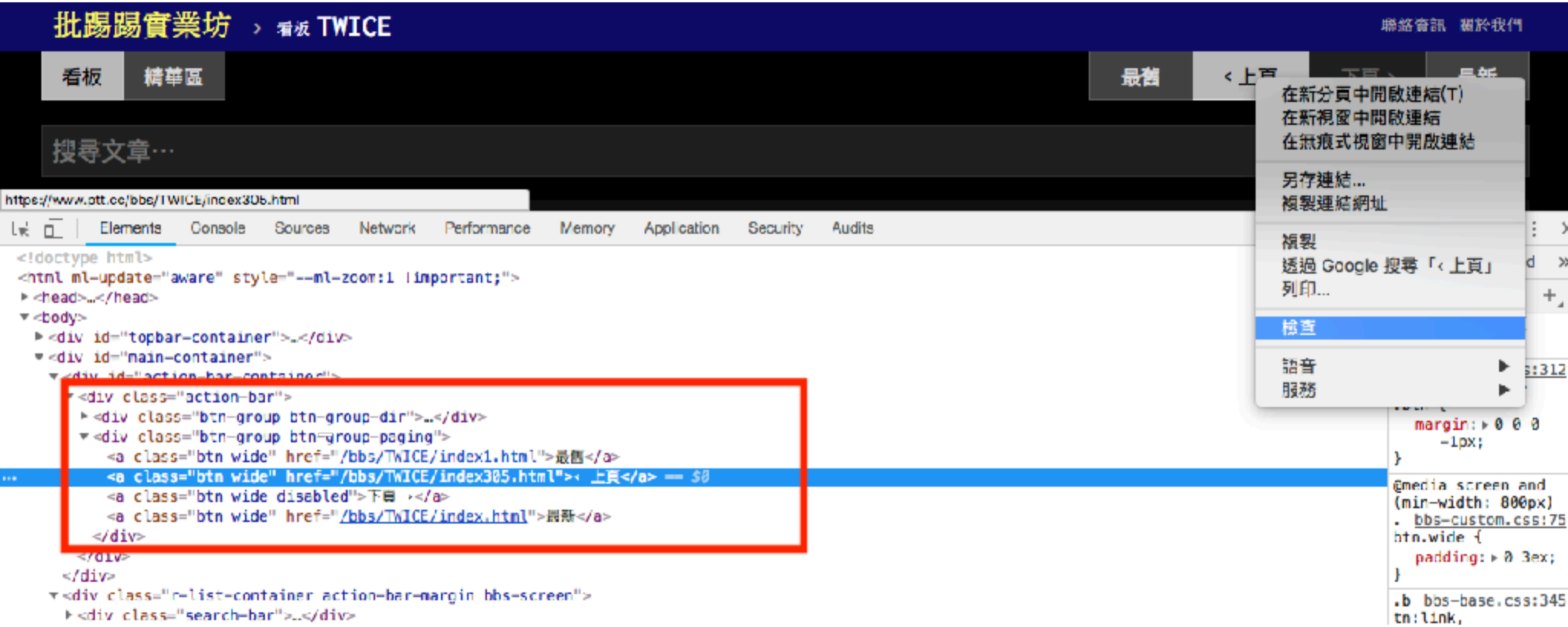
```
import requests
from bs4 import BeautifulSoup

article_href =[]
r = requests.get("https://www.ptt.cc/bbs/TWICE/index.html")
soup = BeautifulSoup(r.text,"html.parser")
results = soup.findAll(“div",{"class":"title"})

for item in results:
    item_href = item.find("a").attrs["href"]
```

4. 用迴圈去把每個div中a標籤的 ‘href’ 找出來
因為每個div只有一個a所以只需要用find(“a”)

```
    article_href.append(item_href)
    print(item_href)
```

5. 將每個解析出來的href 放到list裡面

Changes Pages (1/4)



```
<!doctype html>
<html ml-update="aware" style="--ml-zoom:1 !important;">
 ▶<head>…</head>
 ▼<body>
   ▶<div id="topbar-container">…</div>
   ▼<div id="main-container">
     ▼<div id="action-bar-container">
       ▼<div class="action-bar">
         ▶<div class="btn-group btn-group-dir">…</div>
         ▼<div class="btn-group btn-group-paging">
             <a class="btn wide" href="/bbs/TWICE/index1.html">最舊</a>
...          <a class="btn wide" href="/bbs/TWICE/index305.html">‹ 上頁</a> == $0
             <a class="btn wide disabled">下頁 ›</a>
             <a class="btn wide" href="/bbs/TWICE/index.html">最新</a>
           </div>
         </div>
       </div>
     ▼<div class="r-list-container action-bar-margin bbs-screen">
       ▶<div class="search-bar">…</div>
```

```
  margin: ▶ 0 0 0
      -1px;
}
@media screen and
(min-width: 800px)
. bbs-custom.css:75
btn.wide {
  padding: ▶ 0 3ex;
}
.b bbs-base.css:345
tn:link,
```

Changes Pages (2/4)

```
import requests
from bs4 import BeautifulSoup
```

1. 這次我們用CSS選擇器來解析取得div內class為btn-group下的a標籤

```
btn = soup.select('div.btn-group > a')
```

```
[<a class="btn selected" href="/bbs/TWICE/index.html">看板</a>,
<a class="btn" href="/man/TWICE/index.html">精華區</a>,
<a class="btn wide" href="/bbs/TWICE/index1.html">最舊</a>,
<a class="btn wide" href="/bbs/TWICE/index305.html">‹ 上頁</a>,
<a class="btn wide disabled">下頁 ›</a>, <a class="btn wide" href="/bbs/TWICE/index.html">
最新</a>]
```

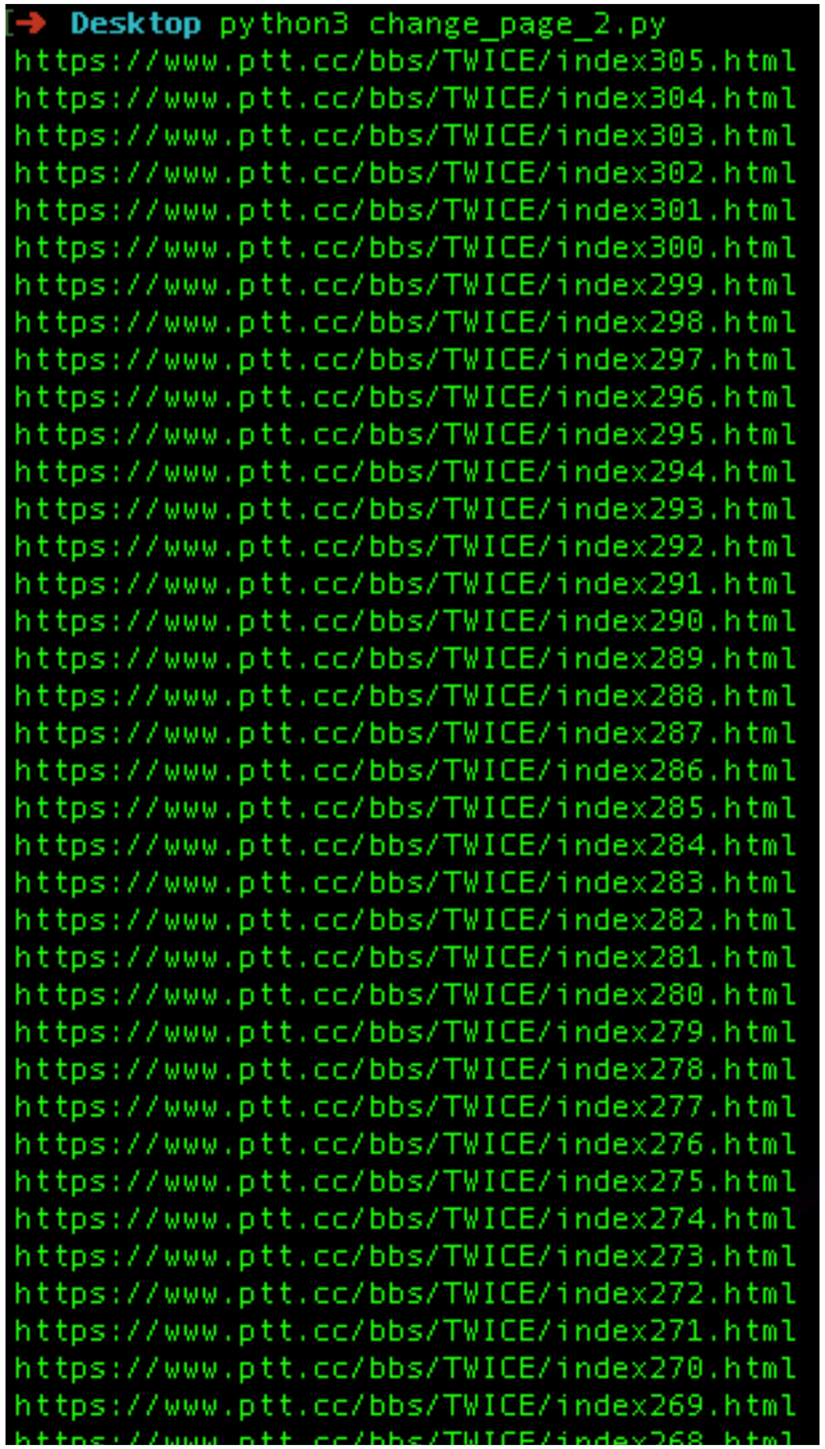
```
up_page_href = btn[3]['href']
```

2. 回傳的結果可以看到要的「上一頁」在第3個Index，用[‘href’]取得它的href

```
next_page_url = 'https://www.ptt.cc' + up_page_href
```

3. 上一頁的Url放到變數nexy_page_url內

Changes Pages (3/4)



```
➜ Desktop python3 change_page_2.py
https://www.ptt.cc/bbs/TWICE/index305.html
https://www.ptt.cc/bbs/TWICE/index304.html
https://www.ptt.cc/bbs/TWICE/index303.html
https://www.ptt.cc/bbs/TWICE/index302.html
https://www.ptt.cc/bbs/TWICE/index301.html
https://www.ptt.cc/bbs/TWICE/index300.html
https://www.ptt.cc/bbs/TWICE/index299.html
https://www.ptt.cc/bbs/TWICE/index298.html
https://www.ptt.cc/bbs/TWICE/index297.html
https://www.ptt.cc/bbs/TWICE/index296.html
https://www.ptt.cc/bbs/TWICE/index295.html
https://www.ptt.cc/bbs/TWICE/index294.html
https://www.ptt.cc/bbs/TWICE/index293.html
https://www.ptt.cc/bbs/TWICE/index292.html
https://www.ptt.cc/bbs/TWICE/index291.html
https://www.ptt.cc/bbs/TWICE/index290.html
https://www.ptt.cc/bbs/TWICE/index289.html
https://www.ptt.cc/bbs/TWICE/index288.html
https://www.ptt.cc/bbs/TWICE/index287.html
https://www.ptt.cc/bbs/TWICE/index286.html
https://www.ptt.cc/bbs/TWICE/index285.html
https://www.ptt.cc/bbs/TWICE/index284.html
https://www.ptt.cc/bbs/TWICE/index283.html
https://www.ptt.cc/bbs/TWICE/index282.html
https://www.ptt.cc/bbs/TWICE/index281.html
https://www.ptt.cc/bbs/TWICE/index280.html
https://www.ptt.cc/bbs/TWICE/index279.html
https://www.ptt.cc/bbs/TWICE/index278.html
https://www.ptt.cc/bbs/TWICE/index277.html
https://www.ptt.cc/bbs/TWICE/index276.html
https://www.ptt.cc/bbs/TWICE/index275.html
https://www.ptt.cc/bbs/TWICE/index274.html
https://www.ptt.cc/bbs/TWICE/index273.html
https://www.ptt.cc/bbs/TWICE/index272.html
https://www.ptt.cc/bbs/TWICE/index271.html
https://www.ptt.cc/bbs/TWICE/index270.html
https://www.ptt.cc/bbs/TWICE/index269.html
https://www.ptt.cc/bbs/TWICE/index268.html
```

2. 第一次呼叫的時候預設用主頁的url

```python
import requests
from bs4 import BeautifulSoup
def main_function(url="https://www.ptt.cc/bbs/TWICE/index.html"):
    r = requests.get(url)
    soup = BeautifulSoup(r.text,"html.parser")
    btn = soup.select('div.btn-group > a')
    up_page_href = btn[3]['href']
    next_page_url = 'https://www.ptt.cc' + up_page_href
```

3. 利用遞迴，每取得上一頁的url 就再丟到function內再去取得更上一頁的url

```python
    main_function(url=next_page_url)

main_function()
```

1. 第一次呼叫function

Changes Pages (4/4)

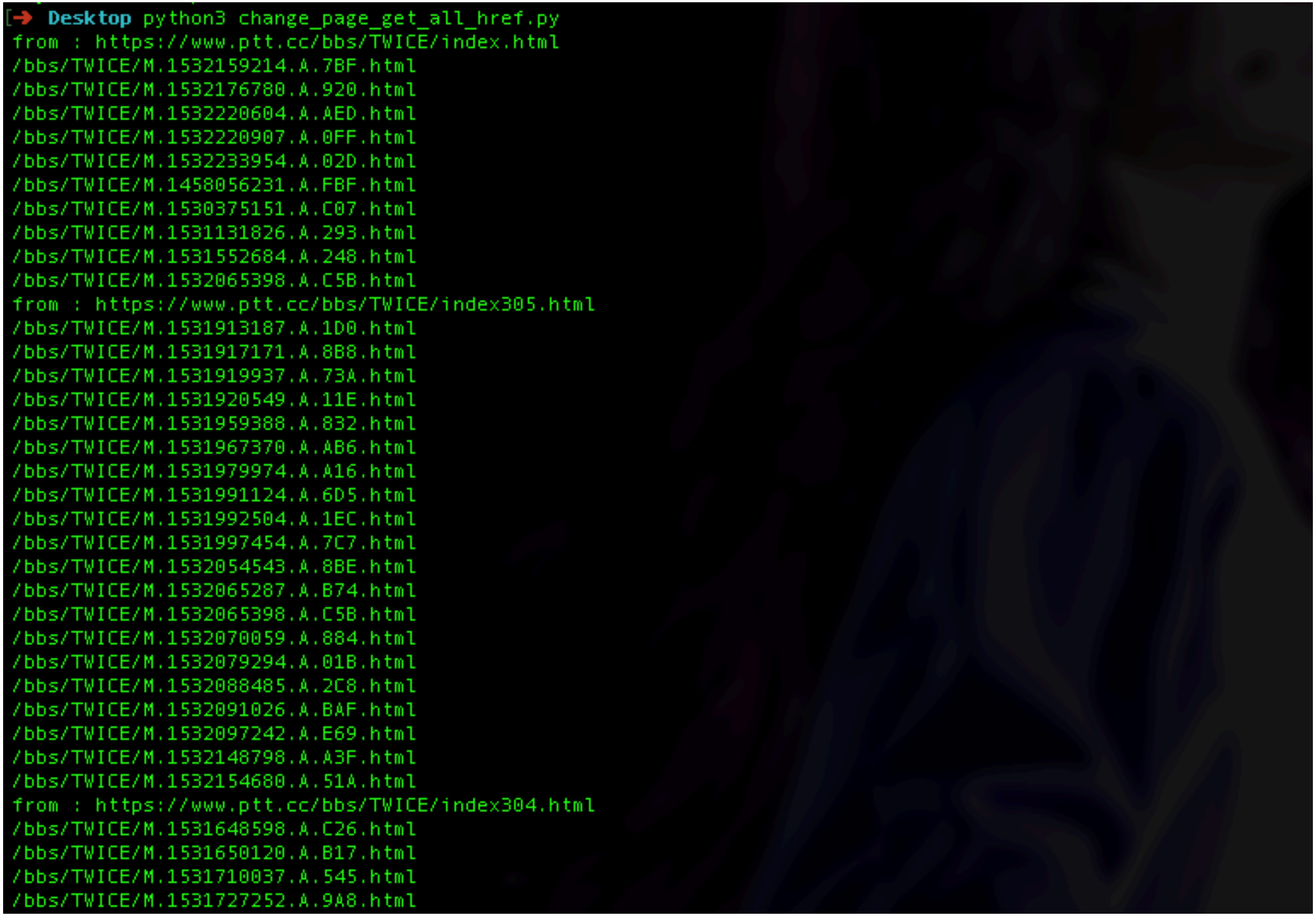
```
def get_all_articles_href(page_url="https://www.ptt.cc/bbs/TWICE/index.html"):
    article_href =[]
    r = requests.get(page_url)
    soup = BeautifulSoup(r.text,"html.parser")
    results = soup.findAll("div",{"class":"title"})
    for item in results:
        try:
            item_href = item.find("a").attrs["href"]
            article_href.append(item_href)
        except:
            pass
    # print(article_href)
    return article_href
```

1. 將前面所介紹的取得每個文章的href 寫成一個function

2. 收集該頁的所有href後放到list內 最後將 list 回傳

```
def main_function(url="https://www.ptt.cc/bbs/TWICE/index.html"):
    r = requests.get(url)
    soup = BeautifulSoup(r.text,"html.parser")

    this_page_article_href = get_all_articles_href(page_url=url)
    btn = soup.select('div.btn-group > a')
    up_page_href = btn[3][‘href']
    next_page_url = 'https://www.ptt.cc' + up_page_href
    main_function(url=next_page_url)
main_function()
```

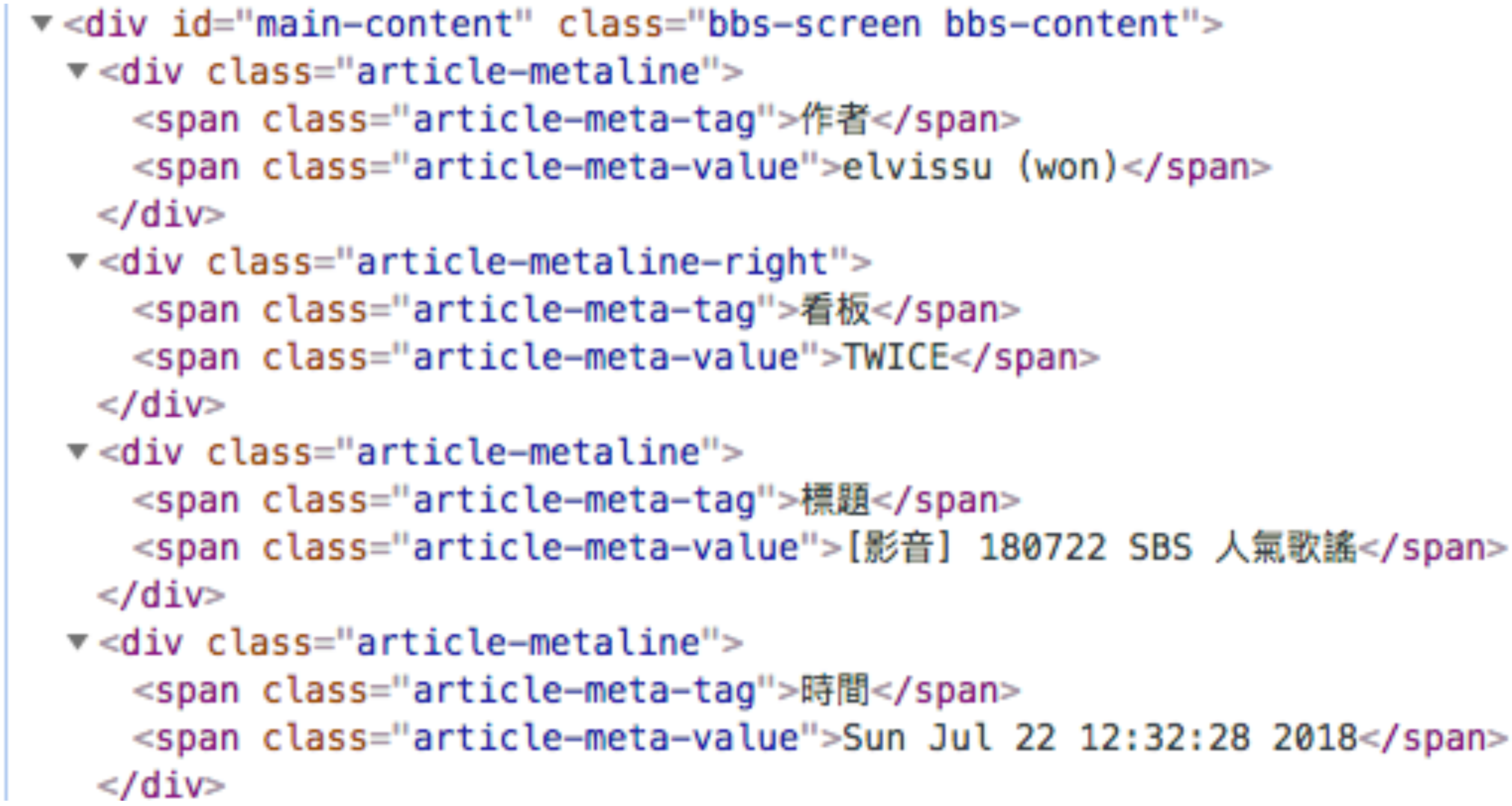
3. 用this_page_article_href變數接收function回傳值



```
[➜ Desktop python3 change_page_get_all_href.py
 from : https://www.ptt.cc/bbs/TWICE/index.html
 /bbs/TWICE/M.1532159214.A.7BF.html
 /bbs/TWICE/M.1532176780.A.920.html
 /bbs/TWICE/M.1532220604.A.AED.html
 /bbs/TWICE/M.1532220907.A.0FF.html
 /bbs/TWICE/M.1532233954.A.02D.html
 /bbs/TWICE/M.1458056231.A.FBF.html
 /bbs/TWICE/M.1530375151.A.C07.html
 /bbs/TWICE/M.1531131826.A.293.html
 /bbs/TWICE/M.1531552684.A.248.html
 /bbs/TWICE/M.1532065398.A.C5B.html
 from : https://www.ptt.cc/bbs/TWICE/index305.html
 /bbs/TWICE/M.1531913187.A.1D0.html
 /bbs/TWICE/M.1531917171.A.8B8.html
 /bbs/TWICE/M.1531919937.A.73A.html
 /bbs/TWICE/M.1531920549.A.11E.html
 /bbs/TWICE/M.1531959388.A.832.html
 /bbs/TWICE/M.1531967370.A.AB6.html
 /bbs/TWICE/M.1531979974.A.A16.html
 /bbs/TWICE/M.1531991124.A.6D5.html
 /bbs/TWICE/M.1531992504.A.1EC.html
 /bbs/TWICE/M.1531997454.A.7C7.html
 /bbs/TWICE/M.1532054543.A.8BE.html
 /bbs/TWICE/M.1532065287.A.B74.html
 /bbs/TWICE/M.1532065398.A.C5B.html
 /bbs/TWICE/M.1532070059.A.884.html
 /bbs/TWICE/M.1532079294.A.01B.html
 /bbs/TWICE/M.1532088485.A.2C8.html
 /bbs/TWICE/M.1532091026.A.BAF.html
 /bbs/TWICE/M.1532097242.A.E69.html
 /bbs/TWICE/M.1532148798.A.A3F.html
 /bbs/TWICE/M.1532154680.A.51A.html
 from : https://www.ptt.cc/bbs/TWICE/index304.html
 /bbs/TWICE/M.1531648598.A.C26.html
 /bbs/TWICE/M.1531650120.A.B17.html
 /bbs/TWICE/M.1531710037.A.545.html
 /bbs/TWICE/M.1531727252.A.9A8.html
```

Parser Article (1/3)

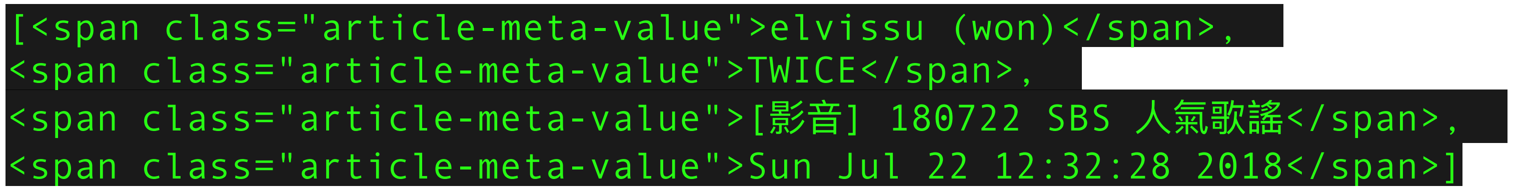
可以看到「作者」、「看板」、「標題」、「時間」資料
都放在 **<span class="article-meta-value"> </span>**



```
▼ <div id="main-content" class="bbs-screen bbs-content">
  ▼ <div class="article-metaline">
      <span class="article-meta-tag">作者</span>
      <span class="article-meta-value">elvissu (won)</span>
    </div>
  ▼ <div class="article-metaline-right">
      <span class="article-meta-tag">看板</span>
      <span class="article-meta-value">TWICE</span>
    </div>
  ▼ <div class="article-metaline">
      <span class="article-meta-tag">標題</span>
      <span class="article-meta-value">[影音] 180722 SBS 人氣歌謠</span>
    </div>
  ▼ <div class="article-metaline">
      <span class="article-meta-tag">時間</span>
      <span class="article-meta-value">Sun Jul 22 12:32:28 2018</span>
    </div>
```

Parser Article (2/3)

```
import requests
from bs4 import BeautifulSoup
r = requests.get("https://www.ptt.cc/bbs/TWICE/M.1532233954.A.02D.html")
soup = BeautifulSoup(r.text,”html.parser")
```



```
[<span class="article-meta-value">elvissu (won)</span>,
<span class="article-meta-value">TWICE</span>,
<span class="article-meta-value">[影音] 180722 SBS 人氣歌謠</span>,
<span class="article-meta-value">Sun Jul 22 12:32:28 2018</span>]
```

1. soup.select(‘span.article-meta-value’) 解析出來的內容

```
author = soup.select('span.article-meta-value')[0].text
board = soup.select('span.article-meta-value')[1].text
title = soup.select('span.article-meta-value')[2].text
time = soup.select('span.article-meta-value')[3].text
print('作者:', author)
print(board,' 看版')
print('標題:', title)
print('時間:', time)
```

2. 第0, 1, 2, 3 index 分別是作者、看板、標題、時間

Parser Article (3/3)

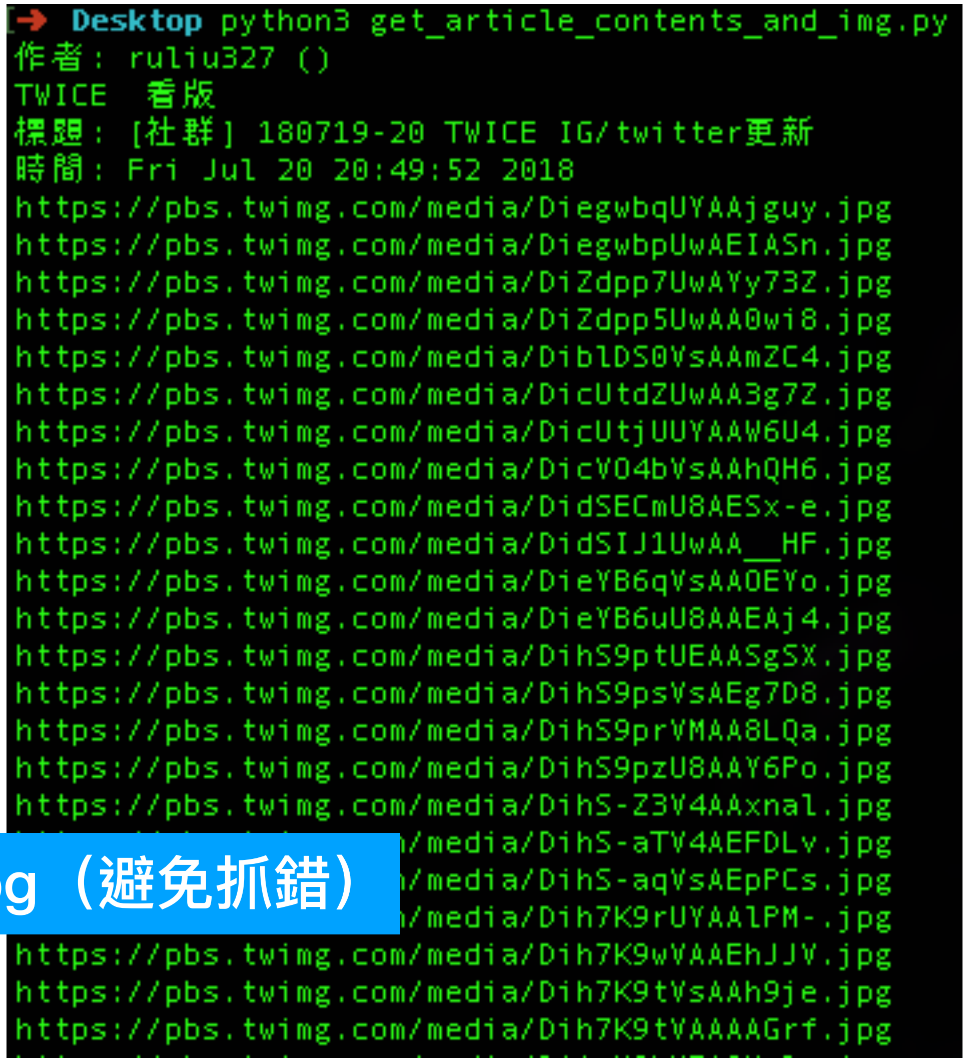
```python
import requests
from bs4 import BeautifulSoup
def get_articles_content(this_page_article_href="/bbs/TWICE/M.1532091026.A.BAF.html"):
    r = requests.get("https://www.ptt.cc" + this_page_article_href )
    soup = BeautifulSoup(r.text,"html.parser")
    try:
        author = soup.select('span.article-meta-value')[0].text
        board = soup.select('span.article-meta-value')[1].text
        title = soup.select('span.article-meta-value')[2].text
        time = soup.select('span.article-meta-value')[3].text
        print('作者:', author)
        print(board,' 看版')
        print('標題:', title)
        print('時間:', time)
    except:
        pass
    imgs = soup.findAll('a')
    for img in imgs:
        if '.jpg' in img['href']:
            print(img['href'])
get_articles_content()
```

1. 在function內丟入一個文章的href

2. 爬取看板內容（同上頁）

3. 找出所有a標籤（圖片）

4. 判斷圖片網址是否包含jpg（避免抓錯）



```
→ Desktop python3 get_article_contents_and_img.py
作者: ruliu327 ()
TWICE  看版
標題: [社群] 180719-20 TWICE IG/twitter更新
時間: Fri Jul 20 20:49:52 2018
https://pbs.twimg.com/media/DiegwbqUYAAjguy.jpg
https://pbs.twimg.com/media/DiegwbpUwAEIASn.jpg
https://pbs.twimg.com/media/DiZdpp7UwAYy73Z.jpg
https://pbs.twimg.com/media/DiZdpp5UwAA0wi8.jpg
https://pbs.twimg.com/media/DiblDS0VsAAmZC4.jpg
https://pbs.twimg.com/media/DicUtdZUwAA3g7Z.jpg
https://pbs.twimg.com/media/DicUtjUUYAAW6U4.jpg
https://pbs.twimg.com/media/DicVO4bVsAAhQH6.jpg
https://pbs.twimg.com/media/DidSECmU8AESx-e.jpg
https://pbs.twimg.com/media/DidSIJ1UwAA__HF.jpg
https://pbs.twimg.com/media/DieYB6qVsAAOEYo.jpg
https://pbs.twimg.com/media/DieYB6uU8AAEAj4.jpg
https://pbs.twimg.com/media/DihS9ptUEAASgSX.jpg
https://pbs.twimg.com/media/DihS9psVsAEg7D8.jpg
https://pbs.twimg.com/media/DihS9prVMAA8LQa.jpg
https://pbs.twimg.com/media/DihS9pzU8AAY6Po.jpg
https://pbs.twimg.com/media/DihS-Z3V4AAxnal.jpg
                 /media/DihS-aTV4AEFDLv.jpg
                 /media/DihS-aqVsAEpPCs.jpg
                 /media/Dih7K9rUYAAlPM-.jpg
https://pbs.twimg.com/media/Dih7K9wVAAEhJJV.jpg
https://pbs.twimg.com/media/Dih7K9tVsAAh9je.jpg
https://pbs.twimg.com/media/Dih7K9tVAAAAGrf.jpg
```

Download Images

```
from bs4 import BeautifulSoup
import requests
import shutil

img_url = 'https://pbs.twimg.com/media/DiZdpp5UwAA0wi8.jpg'
img_name = 'twice'


r = requests.get(img_url, stream=True)
file_name = img_name

print( 'save img to  ./image/'+ file_name + ‘.jpg')

with open('./image/' + file_name + '.jpg', 'wb+') as out_file:
    shutil.copyfileobj(r.raw, out_file)
```

1. 圖片連結

2. 圖片名稱

3. 用shutil.copyfileobj 方法存圖片（較快吧？）

2. 圖片名稱用數字（不重複）

```
def get_articles_content(this_page_article_href):
    image_count = 0
    for url in this_page_article_href:
        r = requests.get("https://www.ptt.cc" + url )
        soup = BeautifulSoup(r.text,"html.parser")
        try:
            author = soup.select('span.article-meta-value')[0].text
            board = soup.select('span.article-meta-value')[1].text
            title = soup.select('span.article-meta-value')[2].text
            time = soup.select('span.article-meta-value')[3].text
            print('作者:', author)
            print(board,' 看版')
            print('標題:', title)
            print('時間:', time)
        except:
            pass
        imgs = soup.findAll('a')
        for img in imgs:
            if '.jpg' in img['href']:
                download_img_from_article(img_url= img['href'], img_name = image_count)
                image_count += 1
```

3. 得到圖片連結後丟入download_img_from_article

1. 建立download_img_from_artiacle() function

```
def download_img_from_article(img_url, img_name):
    r = requests.get(img_url, stream = True)
    file_name = str(img_name + 1)
    print( 'save img to  ./image/'+ file_name + '.jpg')
    try:
        with open('./image/' + file_name + '.jpg', 'wb') as out_file:
            shutil.copyfileobj(r.raw, out_file)
    except:
        print('can not save img', img_url)
```

```
def main_function(url="https://www.ptt.cc/bbs/TWICE/index.html"):
    r = requests.get(url)
    soup = BeautifulSoup(r.text,"html.parser")

    this_page_article_href = get_all_articles_href(page_url=url)
```

1. 取得當頁的每個文章的href連結回傳值存到變數內

```
    get_articles_content(this_page_article_href=this_page_article_href)
```

2. 將所有的文章連結丟入解析文章內的資料

```
    btn = soup.select('div.btn-group > a')
    up_page_href = btn[3]['href']
    next_page_url = 'https://www.ptt.cc' + up_page_href

    main_function(url=next_page_url)
```

3. 切到下一頁

```
main_function()
```

程式碼：

https://github.com/plusoneee/python-learning/blob/master/python-requests/ptt-scrapy/ptt_save_img.py