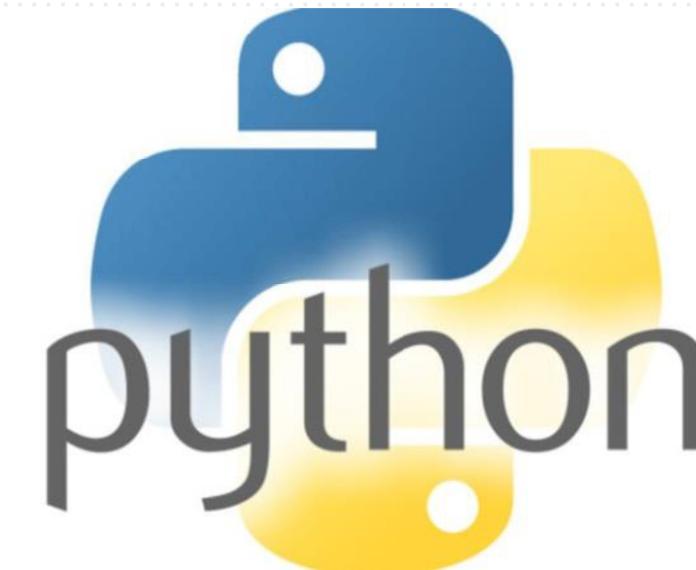


Python Crawler Issues

Python 爬蟲常見問題



Python Web Crawler

Python 爬蟲程式常見問題

- ◆ Python爬蟲程式是向網頁伺服器送出HTTP請求後，從回傳的HTML網頁擷取出內容。
- ◆ Python有很多網路爬蟲函式庫和網頁定位方式。基本上，如果我們只需爬取網頁網站的數頁網頁，請使用Beautiful Soup；如果需要爬取JavaScript產生的動態網頁，或與表單進行互動，請使用Selenium。
- ◆ 但是在目前的網站很多都內建「防爬機制」，連線時可能會遇到一些問題，以下我們來分析/處理常見的問題。



BeautifulSoup



Python 爬蟲程式常見問題 - 被封鎖

- ◆ 所謂被網站封鎖，就是原本爬蟲程式可以運作，但運作一段時間後失效，且得到網站連不上或未經授權等訊息，通常是因為你的爬蟲程式(或你的主機IP)已經被網站列入黑名單，禁止存取。
- ◆ 被網站封鎖的原因不外乎兩個：「**爬蟲不像人**」，或是「**爬蟲沒有禮貌**」，以下分別說明。

爬蟲不像人：

- ◆ 爬蟲不像人，指的是程式的動作太快了，快到超過人類操作的速度，這是最常見被網站封鎖的原因。只要是稍有規模網站，都會有防止分散式攻擊的機制，因此只要同一個大量、快速的向網站索取資料，網站馬上就會把來源判斷為惡意程式並封鎖。因此，設定時間間隔是一件非常重要的事情。
- ◆ 我們可以在爬蟲程式碼中，每幾個步驟插入一個 `time.sleep()` 函數，放慢爬蟲的速度，很多時候這樣一個簡單的動作就可以解決被封鎖的問題。

Python 爬蟲程式常見問題 - 被封鎖

在多次HTTP請求之間加上延遲時間

- ◆ 因為Python爬蟲程式很可能需要在極短的時間內，針對同一網站密集的送出HTTP請求。例如在1內送出超過10次請求，為了避免被駭客攻擊，網站大都有預防密集請求的保護機制。
- ◆ 爬蟲程式應該避免在短時間密集送出HTTP請求，而是在每一次請求之間等待幾秒鐘，如下程式碼匯入time模組，for迴圈一共送出9次HTTP請求，在每一次請求之間呼叫time.sleep(5)函數暫停幾秒鐘，以此例而言參數是5秒，也就是每等5秒鐘才送出一次HTTP請求。

```
1 import time
2 import requests
3 url = "http://www.major-tests.com/word-Lists/word-List-0{0}.html"
4 for i in range(1, 10):
5     url = url.format(i)
6     r = requests.get(url)
7     print(r.status_code)
8     print("等待5秒鐘...")
9     time.sleep(5)
```

Python 爬蟲程式常見問題 - 被封鎖

爬蟲沒有禮貌

- ◆ 第二個被封鎖的原因是你的程式沒有遵守目標網站希望你依循的規則，也就是沒有禮貌。
- ◆ 通常稍具規模的網站，在其主網域下都會有一個`robots.txt`文件，這是一個約定俗成的文件，裡面記載著這網站希望任何瀏覽器及爬蟲程式遵守的規定，例如禁止的 `User-Agent`、不希望你存取的目錄及檔案類型、或向網站索取資料的時間間隔等。
- ◆ 違反了`robots.txt`裡面的規定會怎樣？其實這些都只是約定俗成的規範，違反了很多時候並沒有什麼大礙，就只是你的爬蟲程式會很快的地封鎖而已；但是若你是常態性或大量地爬取資料，或將爬蟲程式用在商業用途，就必須注意你的目標網址希望你遵守的規範，以免有法律上的疑慮。
- ◆ 有關`robots.txt`的更多說明可以參考維基百科：<https://zh.Wikipedia.org/wiki/Robots.txt>

Python 爬蟲程式常見問題 - 被封鎖

- ◆ amazon.com的robots.txt (網址為<https://amazon.com/robots.txt>) User-Agent : *表示該網站希望所有的程式(不管是瀏覽器或爬蟲程式)都不會去存取(Disallow)圖示中的網站目錄。

```
User-agent: *
Disallow: /exec/obidos/account-access-login
Disallow: /exec/obidos/change-style
Disallow: /exec/obidos/flex-sign-in
Disallow: /exec/obidos/handle-buy-box
Disallow: /exec/obidos/tg/cm/member/
Disallow: /gp/aw/help/id=sss
Disallow: /gp/cart
Disallow: /gp/flex
Disallow: /gp/product/e-mail-friend
Disallow: /gp/product/product-availability
Disallow: /gp/product/rate-this-item
Disallow: /gp/sign-in
Disallow: /gp	reader
Disallow: /gp/sitbv3/reader
Disallow: /gp/richpub/syltguides/create
Disallow: /gp/gfix
Disallow: /gp/associations/wizard.html
Disallow: /gp/dmusic/order
Disallow: /gp/legacy-handle-buy-box.html
Disallow: /gp/aws/ssop
```

```
Disallow: /gp/yourstore
Disallow: /gp/gift-central/organizer/add-wishlist
Disallow: /gp/vote
Disallow: /gp/voting/
Disallow: /gp/music/wma-pop-up
Disallow: /gp/customer-images
Disallow: /gp/richpub/listmania/createline
Disallow: /gp/content-form
Disallow: /gp/pdp/invitation/invite
Disallow: /gp/customer-reviews/common/du
Disallow: /gp/customer-reviews/write-a-review.html
Disallow: /gp/associations/wizard.html
Disallow: /gp/music/clipserv
Disallow: /gp/customer-media/upload
Disallow: /gp/history
Disallow: /gp/item-dispatch
Disallow: /gp/dmusic/order/handle-buy-box.html
Disallow: /gp/recsradio
Disallow: /gp/slredirect
Disallow: /dp/shipping/
```

```
Disallow: /dp/twister-update/
Disallow: /dp/manual-submit/
Disallow: /dp/e-mail-friend/
Disallow: /dp/product-availability/
Disallow: /dp/rate-this-item/
Disallow: /gp/registry/wishlist/*/reserve
Disallow: /gp/structured-ratings/actions/get-experience.html
Disallow: /gp/twitter/
Disallow: /ap/signin
Disallow: /gp/registry/wishlist/
Disallow: /wishlist/
Allow: /wishlist/universal*
Allow: /wishlist/vendor-button*
Allow: /wishlist/get-button*
Disallow: /gp/wishlist/
Allow: /gp/wishlist/universal*
Allow: /gp/wishlist/vendor-button*
```

Python 爬蟲程式常見問題 - 被封鎖

- ◆ 再看一個facebook的例子(網址為<https://www.facebook.com/robots.txt>)。基本上此檔案的前兩行就告訴你 facebook 不允許未授權的爬蟲行為，並導引你到相關網頁說明。
- ◆ 右方User-Agent的部份你會看到Applebot、badiuspier等值，這是給其他大公司的爬蟲程式看的，因為大公司的爬蟲程式(如Google、百度或Bing等)通常會表明自己的身分，也會試著遵守各網站robots.txt內的規定。

```
# Notice: Crawling Facebook is prohibited unless you have express written  
# permission. See: http://www.facebook.com/apps/site_scraping_tos_terms.php  
  
User-agent: Applebot  
Disallow: /ajax/  
Disallow: /album.php  
Disallow: /checkpoint/  
Disallow: /contact_importer/  
Disallow: /dialog/  
Disallow: /fbml/ajax/dialog/  
Disallow: /feeds/  
Disallow: /file_download.php  
Disallow: /hashtag/  
Disallow: /l.php  
Disallow: /live/  
Disallow: /moments_app/  
Disallow: /p.php  
Disallow: /photo.php  
Disallow: /photos.php  
Disallow: /share.php  
Disallow: /share/  
Disallow: /sharer.php  
Disallow: /sharer/  
  
User-agent: Applebot  
Allow: /ajax/bootloader-endpoint/  
Allow: /ajax/pagelet/generic.php/PagePostsSectionPagelet  
Allow: /safetycheck/  
  
User-agent: baiduspider  
Allow: /ajax/bootloader-endpoint/  
Allow: /ajax/pagelet/generic.php/PagePostsSectionPagelet  
Allow: /safetycheck/  
  
User-agent: Bingbot  
Allow: /ajax/bootloader-endpoint/  
Allow: /ajax/pagelet/generic.php/PagePostsSectionPagelet  
Allow: /safetycheck/  
  
User-agent: Googlebot  
Allow: /ajax/bootloader-endpoint/  
Allow: /ajax/pagelet/generic.php/PagePostsSectionPagelet  
Allow: /safetycheck/
```

Python 爬蟲程式常見問題 - 被封鎖

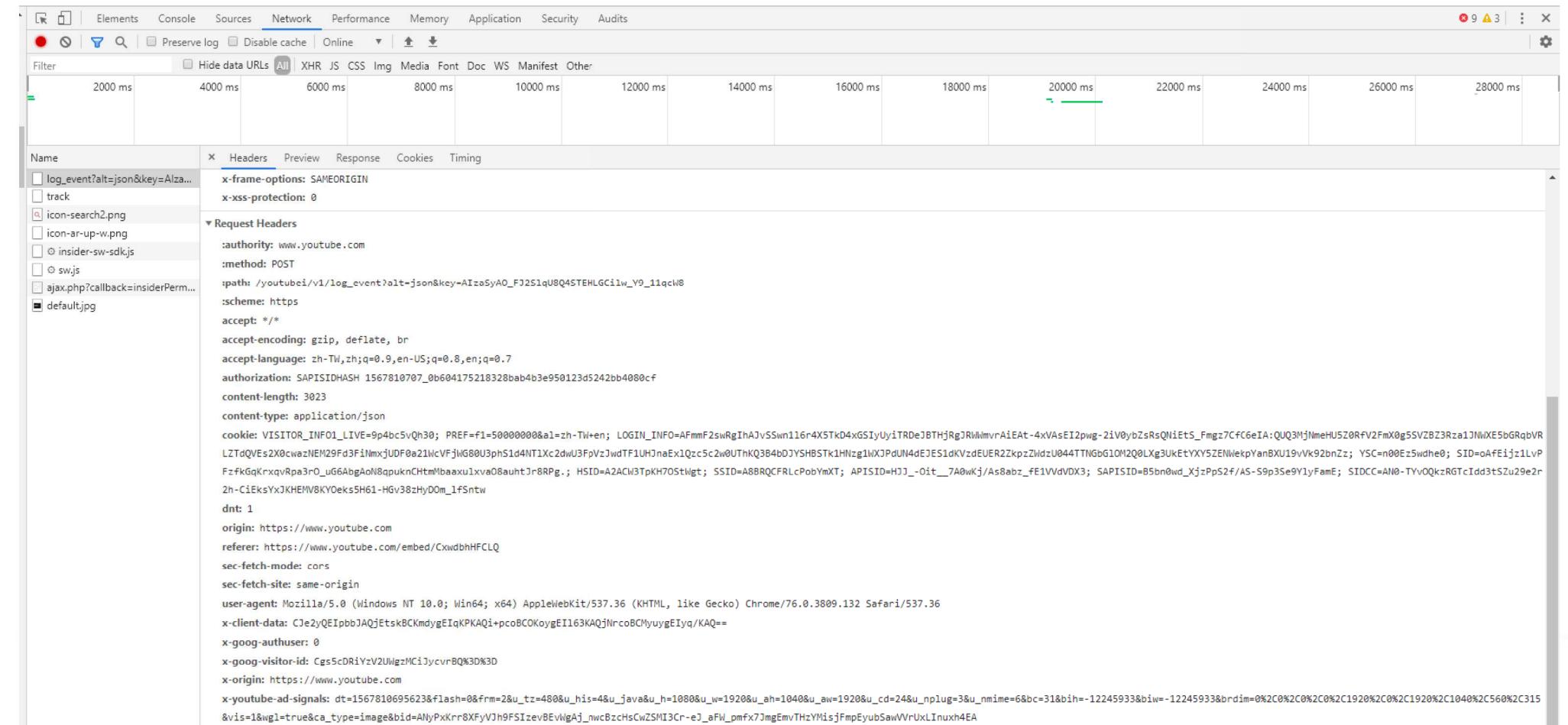
- ◆ 最後是wordpress.com(<https://wordpress.com/robots.txt>)的規定。在這個網站中，它首先提示了若要常態性收集資料，可以使用他們建議的方式；

```
# If you are regularly crawling WordPress.com sites, please use our firehose to receive real-time push updates instead.  
# Please see https://developer.wordpress.com/docs/firehose/ for more details.
```

```
Sitemap: https://wordpress.com/sitemap.xml  
Sitemap: https://wordpress.com/news-sitemap.xml  
  
User-agent: *  
Disallow: /wp-admin/  
Allow: /wp-admin/admin-ajax.php  
Disallow: /wp-signup.php  
Disallow: /press-this.php  
Disallow: /remote-login.php  
Disallow: /activate/  
Disallow: /cgi-bin/  
Disallow: /mshots/v1/  
Disallow: /next/  
Disallow: /public.api/
```

Python 爬蟲程式常見問題 - 表頭偽裝

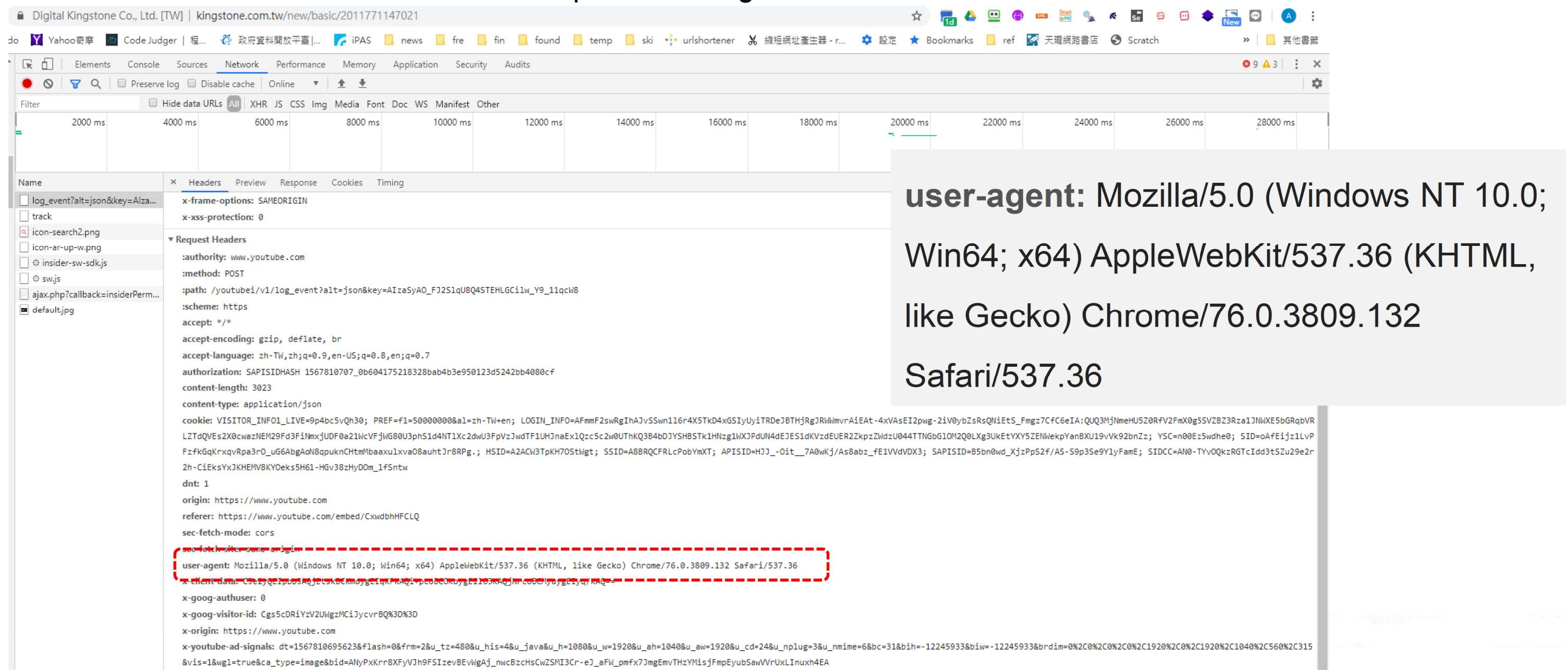
- ◆ 所謂的表頭 (headers) 指的是傳送訊息給網站主機時，一併傳送的訊息描述資料，可以在開發者工具中的 **Requests Headers** 區塊中看到。有時候網站主機不僅會檢查程式傳送給它的資料，還會一併檢查某些表頭欄位，檢查的目的通常是為了確認訊息的確是從使用者端(亦即瀏覽器)所發出，或是其他資安相關理由等。
- ◆ 一般來說，在 **requests** 請求中附加 **User-Agent** 表頭是一個好習慣，可以幫你避免不少錯誤訊息。其他常見的 **header** 檢查位還有 **Referer** 或 **Cookie** 等。



Python 爬蟲程式常見問題 - 表頭偽裝

- ◆ 網站最常檢查的表頭欄位是User-Agent，這也是瀏覽器發出的請求，或各家公司的自動化爬蟲用以表明自己身份的欄位。

以下為金石堂網路站某本書的網址如右：<https://www.kingstone.com.tw/new/basic/2011771147021>



The screenshot shows the Network tab in the Chrome DevTools developer tools. A specific request to 'log_event' is selected. The Headers section displays the following User-Agent header:

```
user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/76.0.3809.132 Safari/537.36
```

The entire User-Agent string is highlighted with a red dashed rectangle.

Other visible headers include:

- x-frame-options: SAMEORIGIN
- x-xss-protection: 0
- Request Headers:
 - :authority: www.youtube.com
 - :method: POST
 - :path: /youtuibe/v1/log_event?alt=json&key=AIzaSyAO_FJ2SlqU8Q4STEHLGCilw_Y9_11qcW8
 - :scheme: https
 - accept: */*
 - accept-encoding: gzip, deflate, br
 - accept-language: zh-TW,zh;q=0.9,en-US;q=0.8,en;q=0.7
 - authorization: SAPISIDHASH 1567810707_0b604175218328bab4b3e950123d5242bb4080cf
 - content-length: 3023
 - content-type: application/json
 - cookie: VISITOR_INFO1_LIVE=9p4bc5vQh30; PREF=f1=50000000&al=zh-TW+en; LOGIN_INFO=AFmmF2swRgIhAJvSSwn116r4X5TkD4xGSIyUyiTRDeJBTHjRgJRWWmvrAiEAt-4xVAsEI2pwg-2iV0ybZsRsQNiEtS_Fmgz7Cfc6eIA:QUQ3MjNmeHU5Z0RfV2FmX0g5SVBZ3Rza1JNWxE5bGRqbVR LzTdqVeS2X0cwazNEM29Fd3Fi1NmxDUDf0a21WcVFjWG80U3phS1d4NT1Xc2dwU3FpVzJwdTF1UHJnaEx1Qzc5c2w0UThKQ3B4bDJYSHBStk1HNzg1WXJPdUN4dEJES1dKVzdEUE2ZkpzWldzU044TTNGbG10M2Q0LXg3UkEtYXY5ZENWekpYanBXU19vVk92bnZz; YSC=n00Ez5wdhe0; SID=oAfEijz1lVp FzfkGqKrxqvRpa3r0_uG6AbgAoN8qupknCHtmBaaxulvxa08auhtJr8RPg.; HSID=A2ACW3TpKH70StWgt; SSID=A8BRQCFLcPobYmXT; APISID=HJJ_-Oit__7A0wKj/As8abz_fE1VVdVDX3; SAPISID=B5bn0wd_XjzPpS2f/AS-S9p3Se9YlyFamE; SIDCC=AN0-TYvOQkzRGtCId3tSzU29e2r 2h-CiEksYxJKHEMV8KY0eks5H61-HGv38zHyD0m_1fSntw
 - dnt: 1
 - origin: https://www.youtube.com
 - referer: https://www.youtube.com/embed/CxwdbhHFCLQ
 - sec-fetch-mode: cors
 - sec-fetch-site: same-origin
 - user-agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/76.0.3809.132 Safari/537.36
 - x-goog-authuser: 0
 - x-googVisitor-id: Cgs5cDRiYzV2UWgzMCiJycvr8Q%3D%3D
 - x-origin: https://www.youtube.com
 - x-youtube-ad-signals: dt=1567810695623&flash=0&frm=2&u_tz=480&u_his=4&u_java&u_h=1080&u_w=1920&u_ah=1040&u_aw=1920&u_cd=24&u_nplug=3&u_nmime=6&bc=31&bih=-12245933&brdim=0%20%20%20%2C1920%20%2C1040%2C560%2C315 &vis=1&wgl=true&ca_type=image&bid=ANyPxKrr8XFyVJh9FSIzevBEvWgAj_nwcBzcHsCwZSMI3Cr-eJ_aFW_pmfx7JmgEmvThzYmisjFmpEyubSawVvruXLInuxh4EA

Python 爬蟲程式常見問題 - 表頭偽裝

更改HTTP表頭偽裝成瀏覽器送出請求

- ◆ 我們如果使用Requests物件送出HTTP請求，網頁網站可以知道是Python程式送出的請求，並不是瀏覽器。
- ◆ 例如以下例子為送出HTTP請求至momo購物網，是使用requests送HTTP請求，執行結果會看到連線錯誤：

```
1 import requests  
2  
3 URL = "https://www.momoshop.com.tw/search/"  
4  
5 r = requests.get(URL+"searchShop.jsp?keyword=HTC")  
6 if r.status_code == requests.codes.ok:  
7     r.encoding = "big5"  
8     print(r.text)  
9 else:  
10    print("HTTP請求錯誤..." + url)
```

```
ConnectionError: ('Connection aborted.', OSError("(10054, 'WSAECONNRESET')"))
```

Python 爬蟲程式常見問題 - 表頭偽裝

更改HTTP表頭偽裝成瀏覽器送出請求

- ◆ 要避免此狀況，我們可以更改表頭資訊方式，假裝從瀏覽器送出HTTP請求。程式碼因為更改HTTP請求的標頭資訊，所以執行結果可以看到成功取回HTML網頁。

```
1 import requests
2 URL="https://www.momoshop.com.tw/search/"
3 headers = { 'user-agent': 'Mozilla/5.0 (Windows NT 10.0; Win64; x64)'
4             'AppleWebKit/537.36 (KHTML, Like Gecko)'
5             'Chrome/63.0.3239.132 Safari/537.36' }
6 r = requests.get(URL+"searchShop.jsp?keyword=HTC", headers=headers)
7 if r.status_code == requests.codes.ok:
8     r.encoding = "big5"
9     print(r.text)
10 else:
11     print("HTTP請求錯誤..." + url)
```

Python 爬蟲程式常見問題 - 例外的HTML標籤

處理例外的HTML標籤

- ◆ 當分析HTML網頁找到目標HTML標籤後，撰寫Python爬蟲程式時需要注意一些例外情況來進行特別處理，否則在爬蟲時可能中斷在這些例外情況。如：PTT BBS的NBA版的HTML標籤，發文的標題文字是

下的標籤。

49

[\[花邊\] Enes Kanter贏得WWE 24/7冠軍腰帶](/bbs/NBA/M.1568107918.A.ED9.html)

```
<div class="r-ent">
    <div class="nrec"><span class="h1 f3">49</span></div>
    <div class="title">
        <a href="/bbs/NBA/M.1568107918.A.ED9.html">[花邊] Enes Kanter贏得WWE 24/7冠軍腰帶</a>
    </div>
```

Python 爬蟲程式常見問題 - 例外的HTML標籤

處理例外的HTML標籤

◆ 如果是刪除的發文則如下。<div class ="title">標籤只有文字內容，沒有<a>標籤，這是發文HTML標籤的例外情況，當發生時，有兩種處理方式：

- 方法一：使用if條件判斷<div class="title">下是否有<a>標籤，沒有<a>標籤就跳過不處理。
- 方法二：使用Beautiful Soup物件建立替代<a>標籤，如果沒有就使用替代標籤來代替。我們接下來就是使用此方法

```
<div class="r-ent">
    <div class="nrec"><span class="hl f0">X2</span></div>
    <div class="title">
        (已被Vedan刪除) &lt;ninebridge&gt;1-1 1-2
    </div>
```

Python 爬蟲程式常見問題 - 例外的HTML標籤

處理例外的HTML標籤

- ◆ 底下的程式範例如以爬取PTT NBA的發文，首先建立DELETED變數，使用Beautiful物件建立標籤籤，程式碼建立標的BeautifulSoup物件，最後的.a是取得此標籤物件，然後送出HTTP請求。

```
1 import requests
2 from bs4 import BeautifulSoup
3
4 URL = "https://www.ptt.cc/bbs/NBA/index.html"
5 DELETED = BeautifulSoup("<a href='Deleted'>本文已刪除</a>", "Lxml").a
6
7 r = requests.get(URL)
8 if r.status_code == requests.codes.ok:
9     r.encoding = "utf8"
10    soup = BeautifulSoup(r.text, "Lxml")
11    tag_divs = soup.find_all("div", class_="r-ent")
12    for tag in tag_divs:
13        tag_a = tag.find("a") or DELETED
14        print(tag_a["href"])
15        print(tag_a.text)
16        print(tag.find("div", class_="author").string)
17 else:
18     print("HTTP請求錯誤..." + url)
```

Python 爬蟲程式常見問題 - 例外的HTML標籤

處理例外的HTML標籤

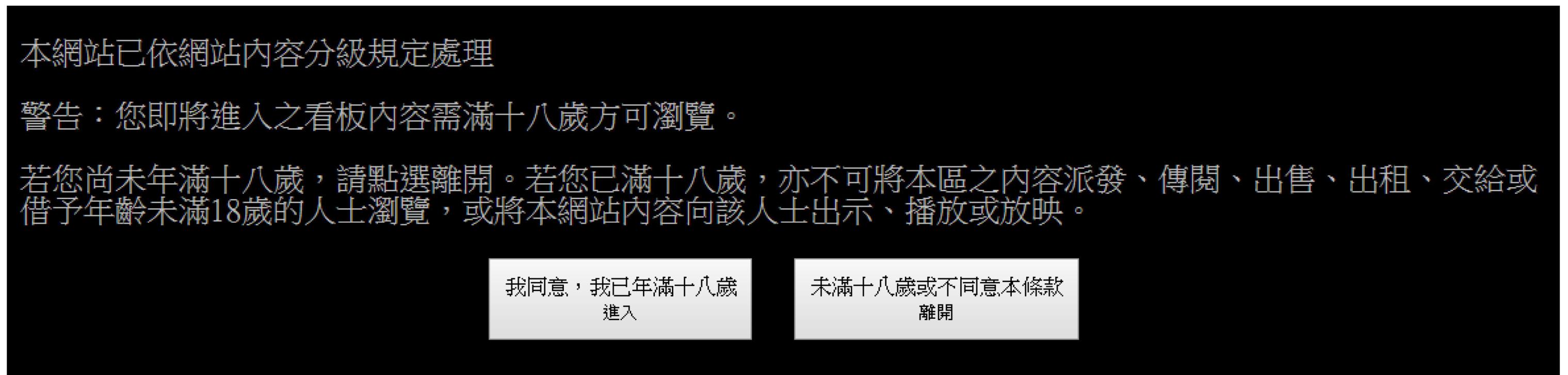
- ◆ 程式碼使用`find_all()`函數找出所有發文的`<div>`標籤後，使用`for/in` 迴圈取出每一篇發文的標題文字，即`<a>`標籤，程式碼使用`find()`函數搜尋`<a>`標籤，沒有找到就指定成`DELETED`變數的`<a>`標籤物件，執行結果可以看到顯示「本文已刪除」(請注意，不是每次都有刪除文章)，這是已刪除的發文，如下所示。

```
1 import requests
2 from bs4 import BeautifulSoup
3
4 URL = "https://www.ptt.cc/bbs/NBA/index.html"
5 DELETED = BeautifulSoup("<a href='Deleted'>本文已刪除</a>", "Lxml").a
6
7 r = requests.get(URL)
8 if r.status_code == requests.codes.ok:
9     r.encoding = "utf8"
10    soup = BeautifulSoup(r.text, "Lxml")
11    tag_divs = soup.find_all("div", class_="r-ent")
12    for tag in tag_divs:
13        tag_a = tag.find("a") or DELETED
14        print(tag_a["href"])
15        print(tag_a.text)
16        print(tag.find("div", class_="author").string)
17 else:
18     print("HTTP請求錯誤..." + url)
```

Python 爬蟲程式常見問題 - 網站內容分級

網站內容分級規定

- ◆ 因為很多網站內容都有分級規定，有些網站在進入前會詢問是否年滿18歲，例如PTT BBS的Gossiping版。因為需按下「我同意，我已年滿十八歲 進行」鈕才能進入網頁。因為 PTT BBS是使用Cookie儲存是否年滿十八歲，因此我們可以在 requests請求指定Cookies來跳過分級規定的畫面。



Python 爬蟲程式常見問題 - 網站內容分級

網站內容分級規定

- ◆ 底下`request.get()`函數的第2個參數然定`cookies`來跳過網站內容分級規定，`for/in` 迴圈是使用`if`條件判斷是否找到`<a>`標籤，而不是使用自訂標來處理例外情況。

```
1 import requests
2 from bs4 import BeautifulSoup
3
4 URL = "https://www.ptt.cc/bbs/Gossiping/index.html"
5
6 r = requests.get(URL, cookies={"over18": "1"})
7 if r.status_code == requests.codes.ok:
8     r.encoding = "utf8"
9     soup = BeautifulSoup(r.text, "Lxml")
10    tag_divs = soup.find_all("div", class_="r-ent")
11    for tag in tag_divs:
12        if tag.find('a'): # 是否有<a>標籤
13            tag_a = tag.find("a")
14            print(tag_a["href"])
15            print(tag_a.text)
16            print(tag.find("div", class_="author").string)
17    else:
18        print("HTTP請求錯誤..." + url)
```

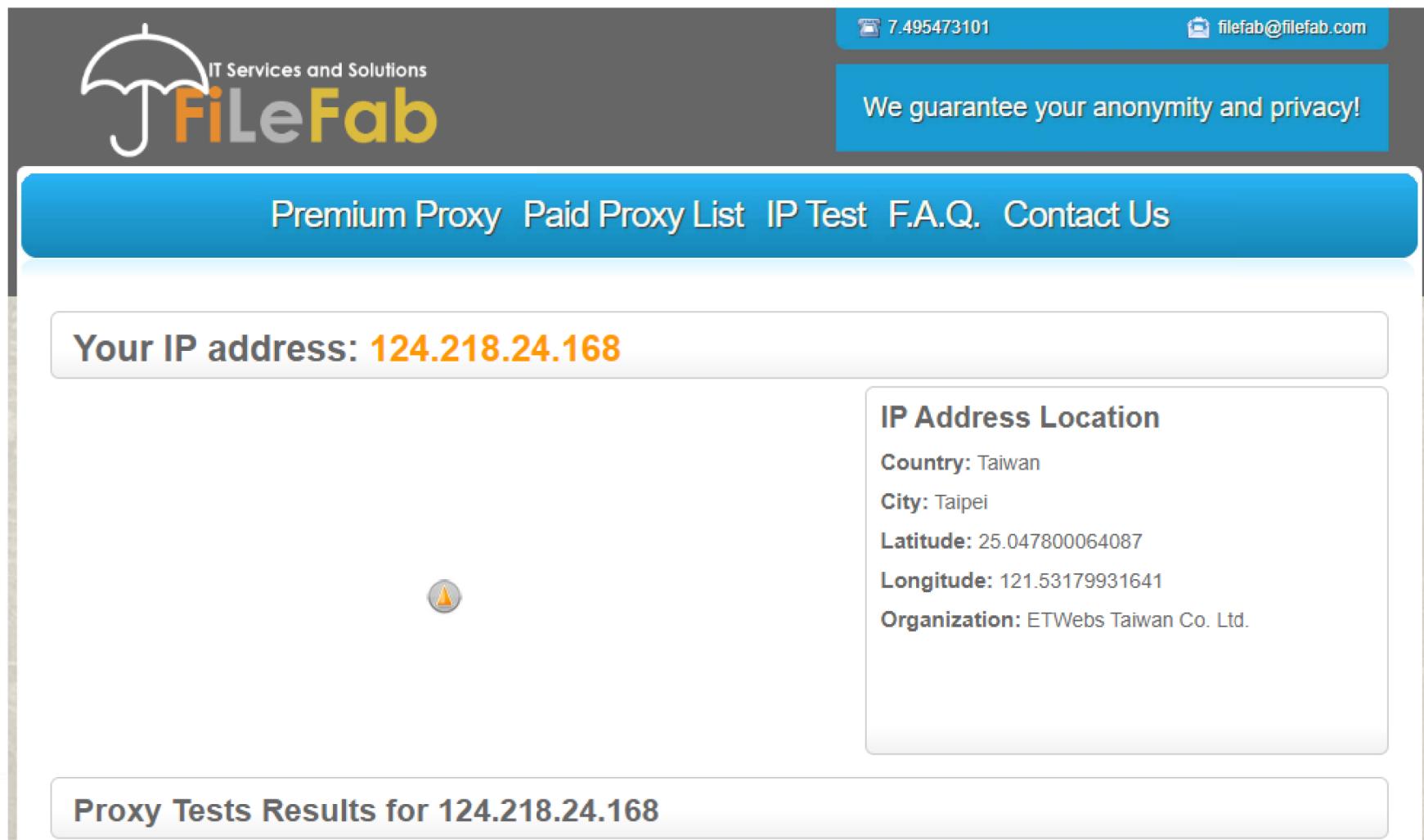
Python 爬蟲程式常見問題 - 使用代理伺服器

◆ 另一個爬蟲程式常見的問題是：如何使用不同的代理伺服器(Proxy Server)，以免多次使用同一個來源IP後被目標網站封？

使用requests套件的話可以很輕易的更換代理IP，重點反而是要能夠找到當下可用的代理伺服器位址。

◆ 此處用以下網頁作為示範：這個網頁可以顯示連線來源IP；

<http://ip.filefab.com/index.php>



Python 爬蟲程式常見問題 - 使用代理伺服器

- ◆ 因此我們在程式碼中，先建立兩個可用的代理IP，並隨機選擇一個用來連線以上網頁，最後再把網頁顯示來源IP的文字部份擷取出來。
- ◆ 代理伺服器查詢：<http://cn-proxy.com/>
- ◆ 可以看到該網頁確實認為請求是從我所使用的Proxy IP 所發出。要注意的是使用Proxy IP並不保證有用，許多網站能夠進一步查詢代理伺服器並取得背後真正的連線來源IP，因此在爬蟲程式的使用上還是要注意道德性原則。

```
1 from bs4 import BeautifulSoup
2 import requests
3 import random
4
5 if __name__ == '__main__':
6     # 代理伺服器查詢: http://cn-proxy.com/
7     proxy_ips = ['81.162.56.154:8081', '51.15.227.220:3128']
8     ip = random.choice(proxy_ips)
9     print('Use', ip)
10    resp = requests.get('http://ip.filefab.com/index.php',
11                          proxies={'http': 'http://' + ip})
12    soup = BeautifulSoup(resp.text, 'html5lib')
13    print(soup.find('h1', id='ipd').text.strip())
```

```
Use 81.162.56.154:8081
Your IP address: 81.162.56.154
```

Q & A