

心理與神經資訊學

(Psychoinformatics & Neuroinformatics)

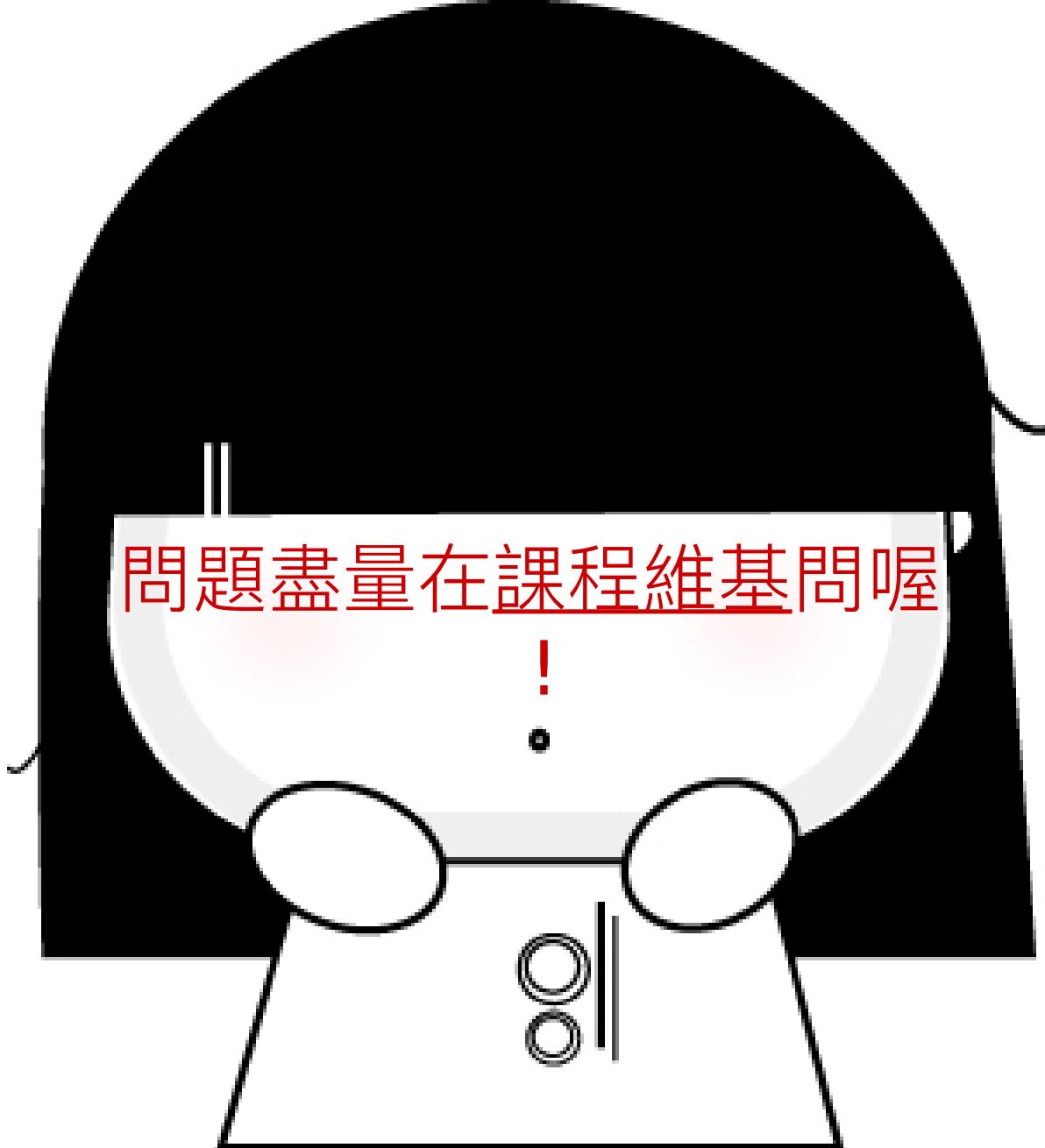
課號：Psy1007

識別碼：20710100

教室：博雅 101

時間：四 234





問題盡量在課程維基問喔

!

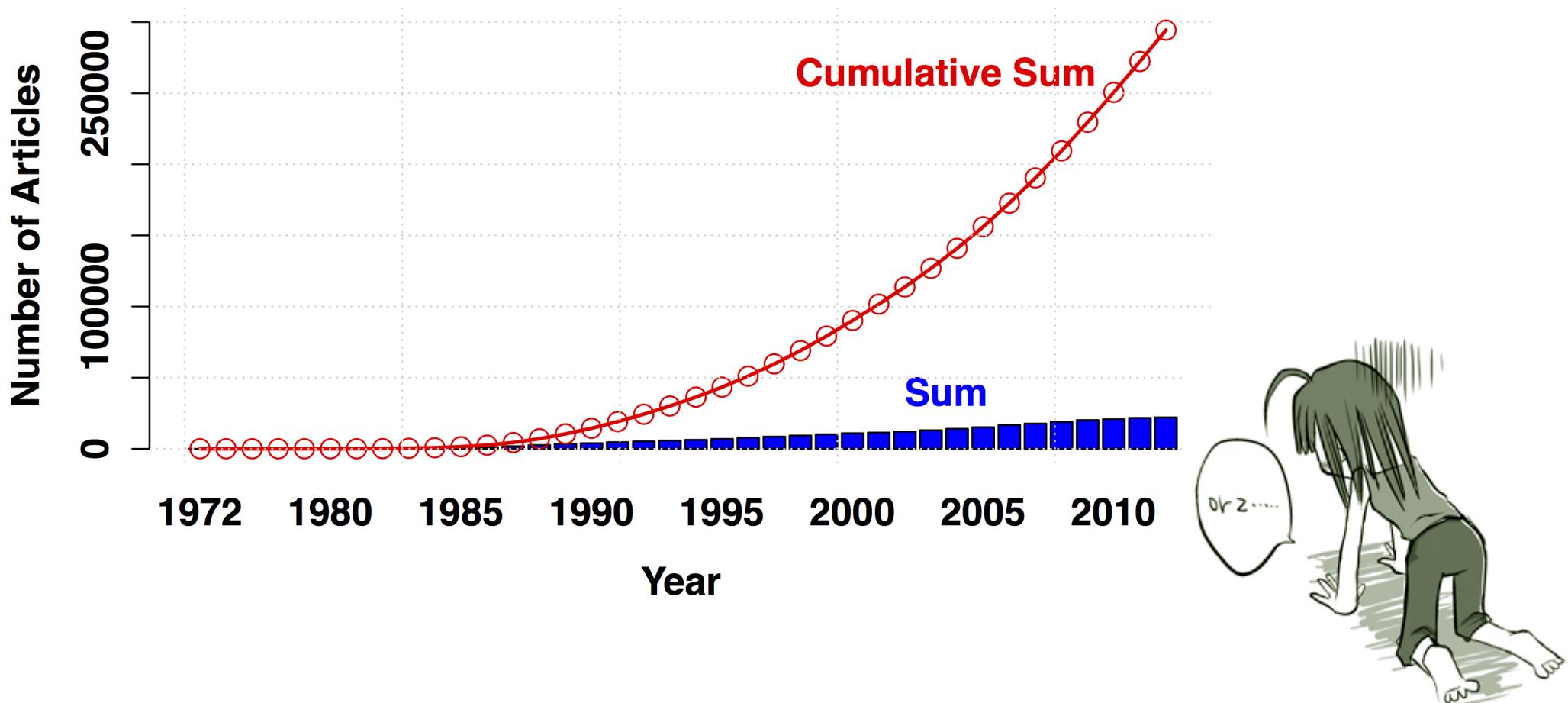
網頁前端技術

(HTML5, CSS, & JavaScript)

腦科學案例研究 (1/3)

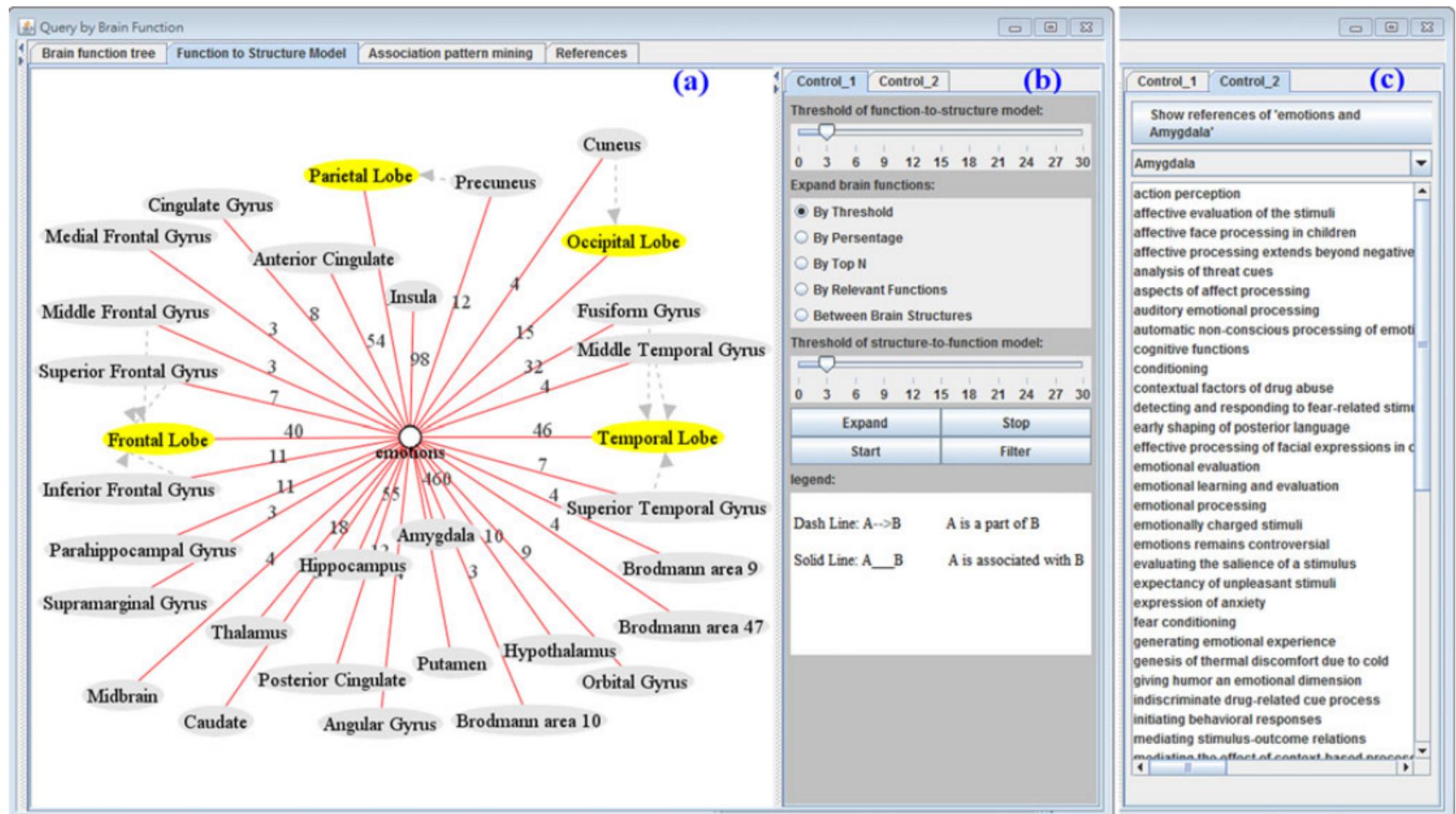
論文怎麼讀都讀不完

Human fMRI Publications



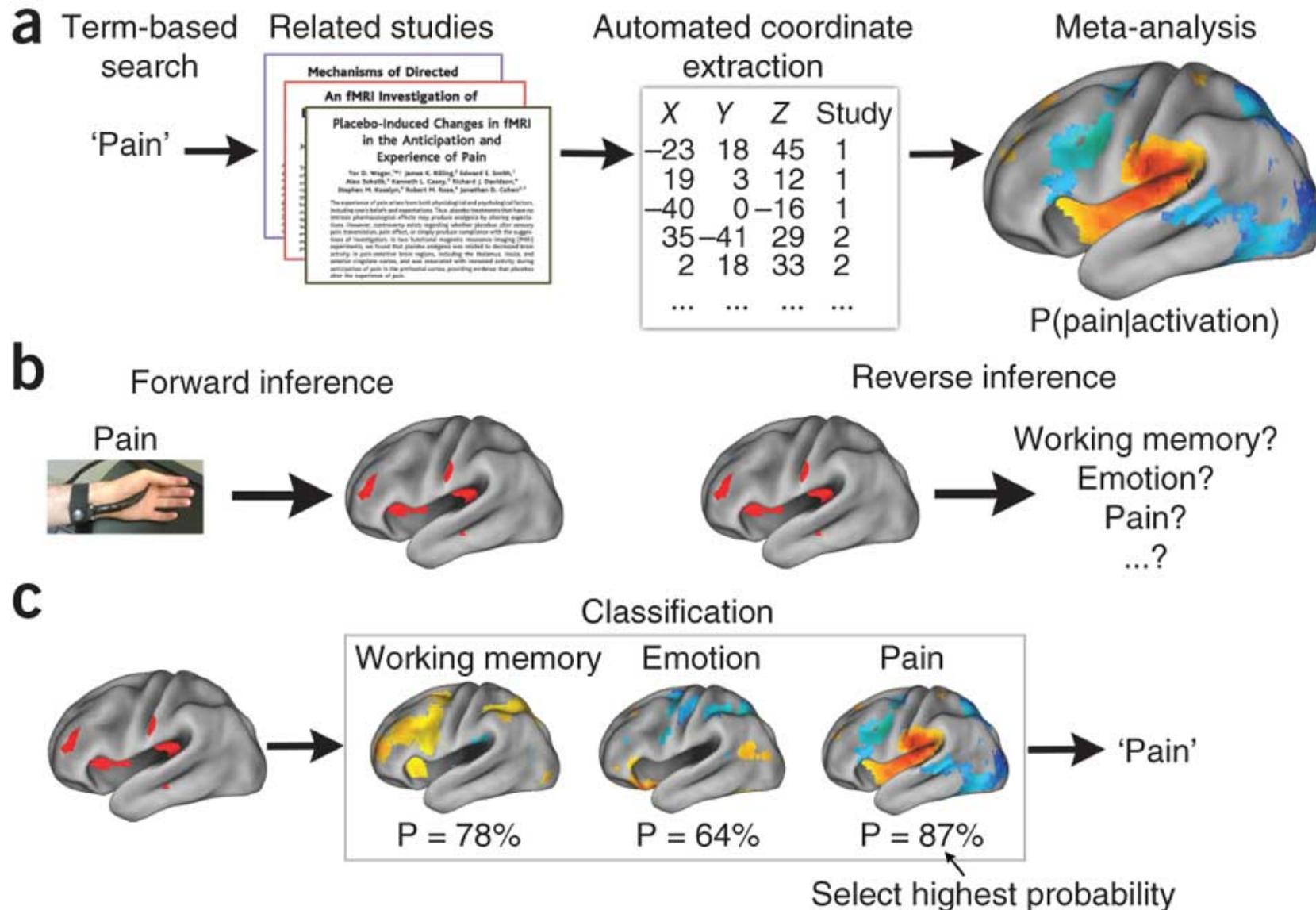
腦科學案例研究 (2/3)

今天學完就可以讓程式可以幫我們讀論文



腦科學案例研究 (3/3)

用腦座標取代腦名稱可以更精確



Yarkoni et al., 2011, *Nature Methods*

[疑問] 網路上的自己才是真正的自己嗎

Fri Sep 25 17:22:05 2015

最近有點好奇，很多人在匿名的環境下發表的言論跟他們在現實上所表現的並不一樣
就好像以前用西斯匿名帳號發文的人，現實生活中可能是很避諱談到性這個方面的
那在網路上的時候才是真正表現自己的個性嗎？

--
Sent from my Android

-
- ※ 發信站: 批踢踢實業坊(ptt.cc), 來自: 223.136.95.38
 - ※ 文章網址: <https://www.ptt.cc/bbs/PSY/M.1443172928.A.CEC.html>
 - maoapple: 每個人的自我都有很多不同的面向，你可以理解成這是其 09/25 18:39
 - maoapple: 中一個面向。 09/25 18:39
 - 推 twcandyman: 推樓上 也有人是盡可能表現一樣的 每個人的方式不同 09/25 20:29
 - 推 winken2004: 現實中的你加網路中的你才是真正的你 09/27 01:52
 - lunenoir: 每個面向的自己都是你，看你願不願意接納而已 10/02 14:51

心理學案例研究 (1/2)

當本門課魯師還在唸博士班時，
有一天系上的印度學長跑來問網路擷取資料的問題。

Tren: How to get these
data from dogpile?

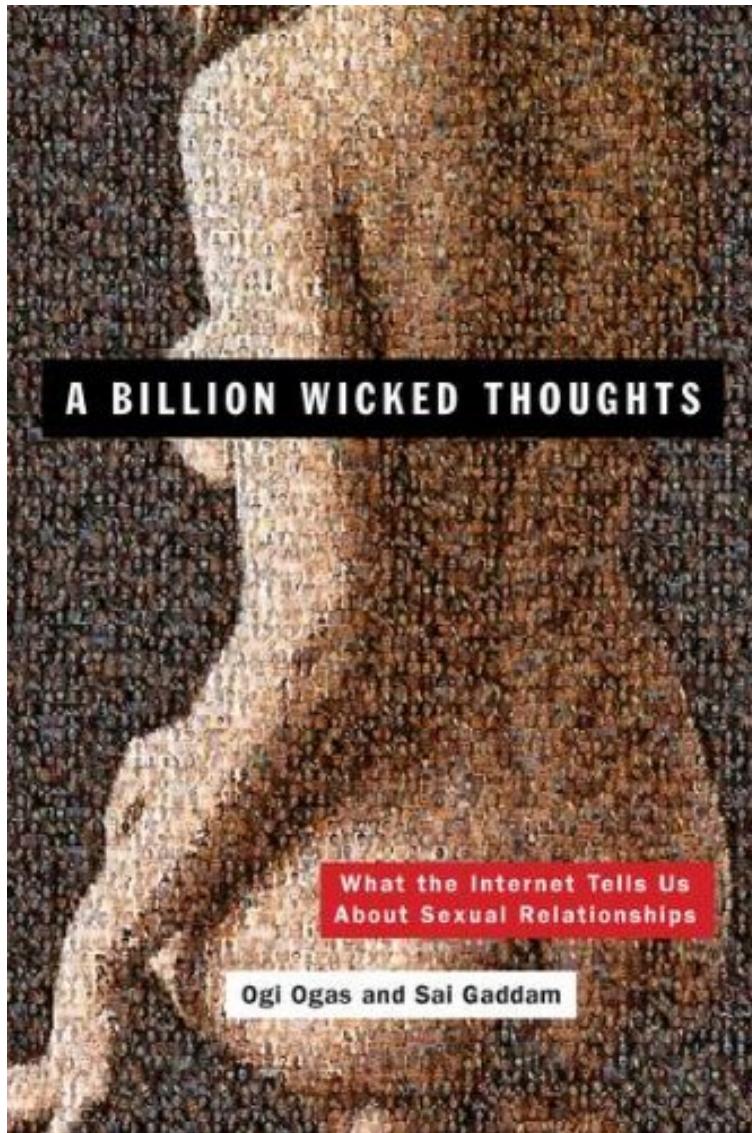
Sai: That's easy!
You can ...

感謝大大無私分享
Orz



心理學案例研究 (2/2)

結果印度學長後來和另一位學長出了這本書：



小故事大啟示：

師父領進門
修行在個人

今天實作範例：愛情心理學研究

假設我們要到PTT網頁版的Boy-Girl版搜集資料



問世堅情為何物
直叫人生死相許



```
import urllib
u='http://www.ptt.cc/bbs/Boy-Girl/'
r=urllib.request.Request(u,headers={'User-Agent':''})
data=urllib.request.urlopen(r).read()
print(data.decode('utf-8'))
```

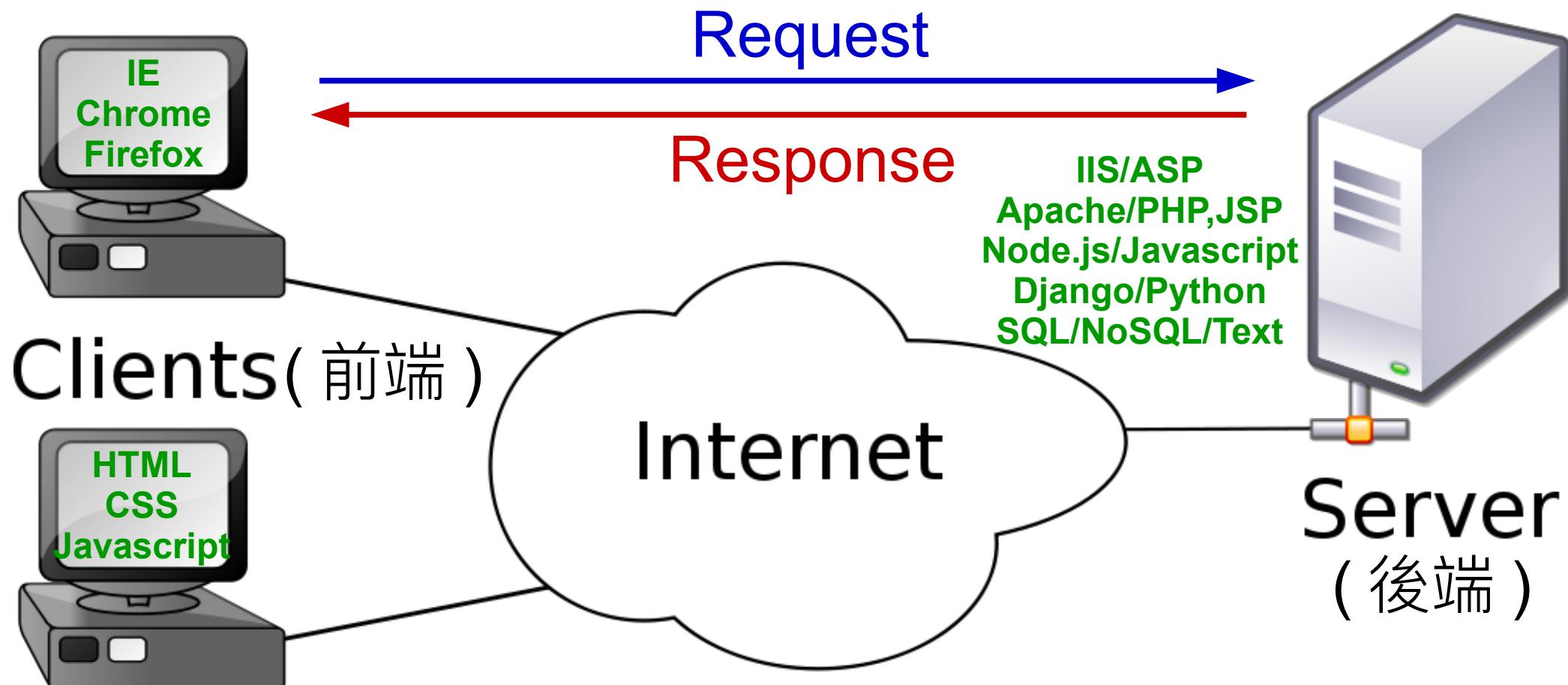
這些是什麼東東？

```
<div class="r-ent">  
<div class="nrec"><span class="hl f2">1</span></div>  
<div class="mark"></div>  
<div class="title">  
<a href="/bbs/Boy-Girl/M.1394964006.A.850.html">Re: [心情] 我  
要怎麼原諒那個人 </a>  
</div>  
<div class="meta">  
<div class="date">3/16</div>  
<div class="author">wbson</div>  
</div>  
</div>
```



網頁前端 (frontend) 和後端 (backend)

前端資料由瀏覽器來處理，
主要是頁面的顯示和小資料的儲存 / 查詢



後端資料由伺服器來處理，主要是大資料的儲存 / 查詢

基本的 HTML 語法

<h1> 標 1</h1><hr> <h2> 標 2</hr> 連接

這樣 `
` 可以斷行 ``

<center> 這樣可以置中和 換色 </center>

< />

<|i> 粗體 </|i>

<|i><i> 斜體 </i></i> <|i><u> 底線 </u></i>


```
<table border=1>
```

```
<tr><td>11</td>    <td>12</td></tr>
```

```
<tr><td>21</td> <td>22</td></tr>
```

</table>



更多的 HTML 語法可來這裡學

CSS: One style fits all

```
<style>
```

```
body {color:white; background-color:black;}
```

```
h1 {color:red; font-size:20pt}
```

```
.yy {color:yellow}
```

```
span#gg {color:green}
```

```
div#bb {color:blue}
```

```
</style>
```

更多的 CSS 語法可來這裡或這裡學

```
<h1>Hi!</h1>
```

This is test1<hr>

This is <div class=yy>test2</div><hr>

This is test3<hr>

This is <div id=gg>test4</div><hr>

This is <div id=bb>test5</div><hr>

這些是什麼東東？

```
<div class="r-ent">  
<div class="nrec"><span class="hl f2">1</span></div>  
<div class="mark"></div>  
<div class="title">  
  <a href="/bbs/Boy-Girl/M.1394964006.A.850.html">Re: [心情]  
我要怎麼原諒那個人 </a>  
</div>  
<div class="meta">  
  <div class="date"> 3/16</div>  
  <div class="author">wbson</div>  
</div>  
</div>
```



工欲善其事必先利其器 (1/2)

使用 Chrome 的 Developer Tools 幫忙理解

The screenshot shows a PTT thread titled "Re: [心情] 我要怎麼原諒那個人" (Index 1393) in a browser window. The browser's address bar shows the URL www.ptt.cc/bbs/Boy-Girl/index1393.html. The page content includes several posts and navigation links like "最舊", "<上頁", "下頁>", and "最新". Below the main content, there is a link to another thread: "Re: [求助] 約好單獨出遊卻變成三人出遊，什麼意思？" (Index 1394964006). The bottom half of the image displays the Chrome Developer Tools. The "Elements" tab is active, showing the DOM tree with nodes like <html>, <body>, and various container and list items. The "Styles" tab is also visible, showing CSS rules for elements like "a:visited" and "a:link" from the file "bbs.css". The "Computed" tab shows the final styles applied to specific elements based on the current browser settings.

工欲善其事必先利其器 (2/2)

使用 Firebug for FireFox 幫忙理解

看板 Boy-Girl 文章列表 - 批踢踢實業坊 - Mozilla Firefox

File Edit View Bookmarks Tools Help

看板 Boy-Girl 文章列表 - 批踢踢實業坊 +

www.ptt.cc/bbs/Boy-Girl/index1393.html

批踢踢實業坊 > 看板 Boy-Girl

聯絡資訊 關於我們

看板 精華區

最舊 < 上頁 下頁 > 最新

1 Re: [心情] 我要怎麼原諒那個人
3/16 wbson

2 Re: [心情] 我要怎麼原諒那個人
3/16 mileslo
(本文已被刪除) [Azabulu]
3/16 -

2 Re: [求助] 約好單獨出遊卻變成三人出遊，什麼意思？

www.ptt.cc/bbs/Boy-Girl/M.1394964006.A.850.html

Console HTML CSS Script DOM Net Cookies

Style Computed Layout DOM

```
+ <div id="action-bar-container">
- <div class="r-list-container bbs-screen">
  <div class="r-ent">
    <div class="nrec">
      <div class="mark"></div>
    <div class="title">
      <a href="/bbs/Boy-Girl/M.1394964006.A.850.html">Re: [心情] 我要怎麼原諒那個人</a>
    </div>
    <div class="meta">
    </div>
  </div>
  <div class="r-ent">
  <div class="r-ent">
    <div class="r-ent">
```

```
a:hover      bbs.css (line 118)
{
  background-color: #CCCCCC;
  color: #333333;
}
a:link       bbs.css (line 115)
{
  color: #AAAAAA;
}
Inherited from div.title
.r-ent        bbs.css (line 346)
> * {
```

網頁資料的搜集 (LXML, Scrapy, & Selenium)

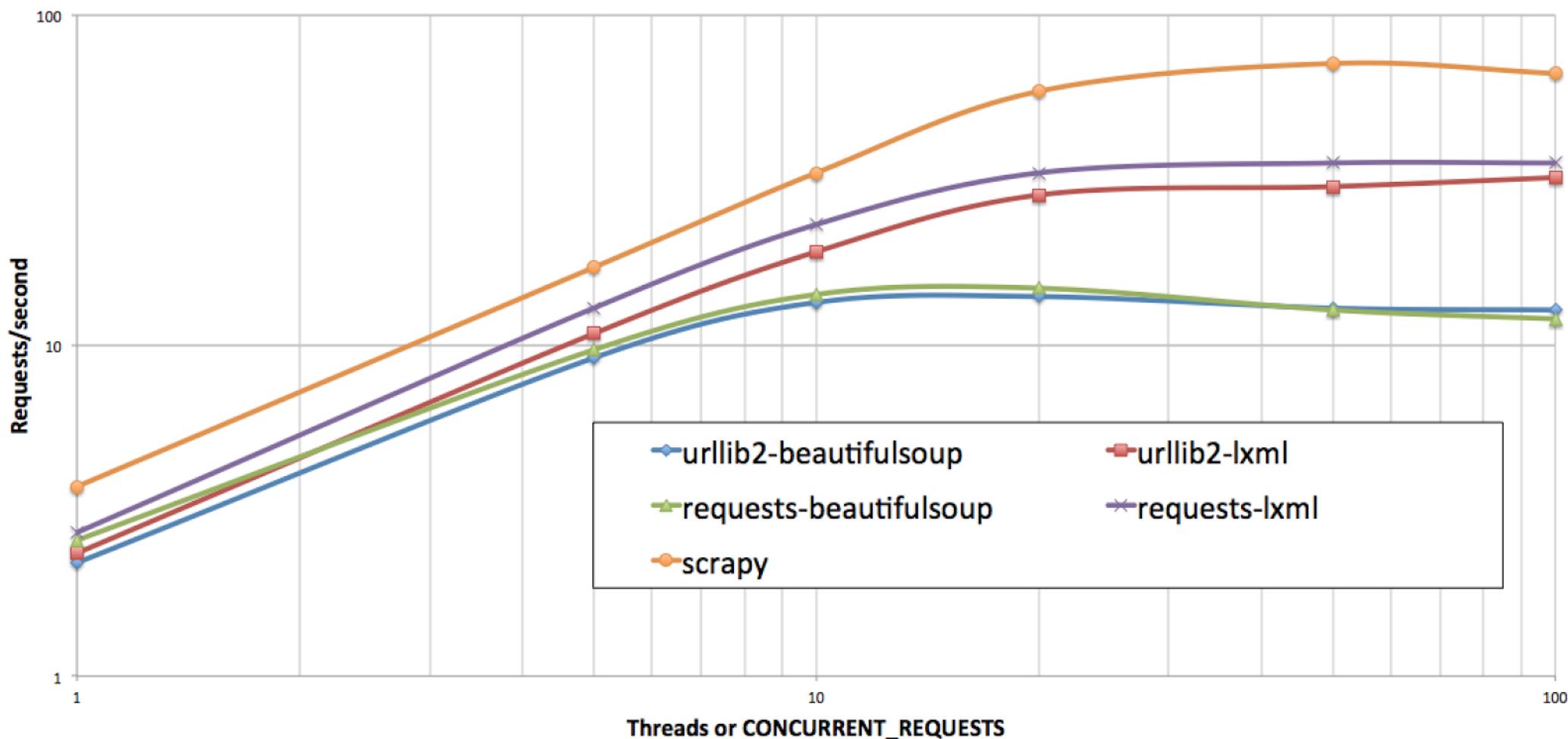
不同爬蟲的比較 (1/2)

Beautiful Soup 是最好學 / 寫的

Aspect	requests + lxml	requests + Beautiful Soup	Scrapy
Performance	✓	✗	✓✓
Ease of installation	✓	✓	✓
Development experience	✗	✓✓	✓
Memory usage	✓	✓	✓
Output files and formats	✗	✗	✓
Javascript support	✗	✗	✗

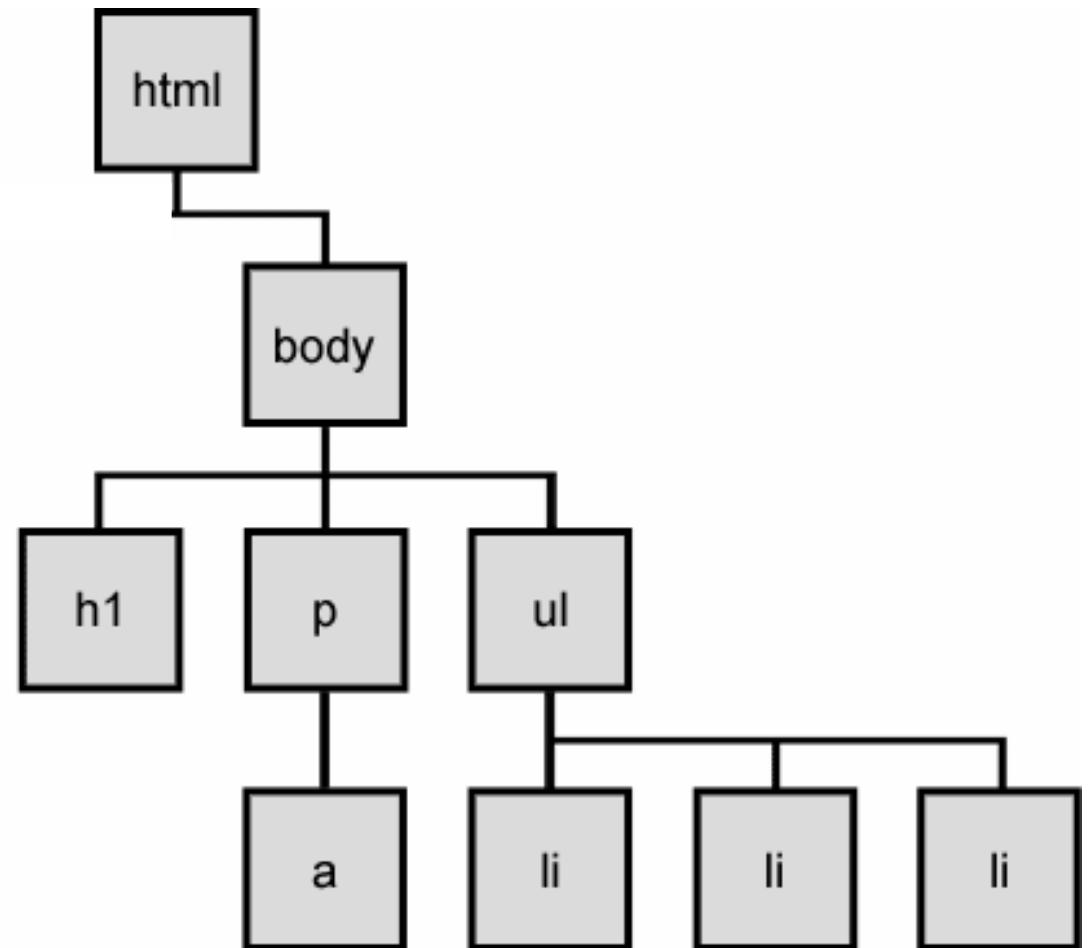
不同爬蟲的比較 (2/2)

Beautiful Soup 却是處理速度最慢的

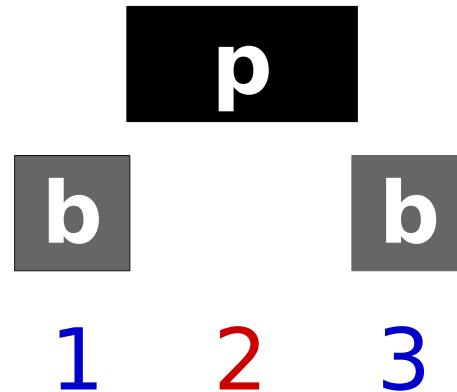


LXML: HTML Parser

讓我們可悠遊在 HTML 文件的樹狀結構中



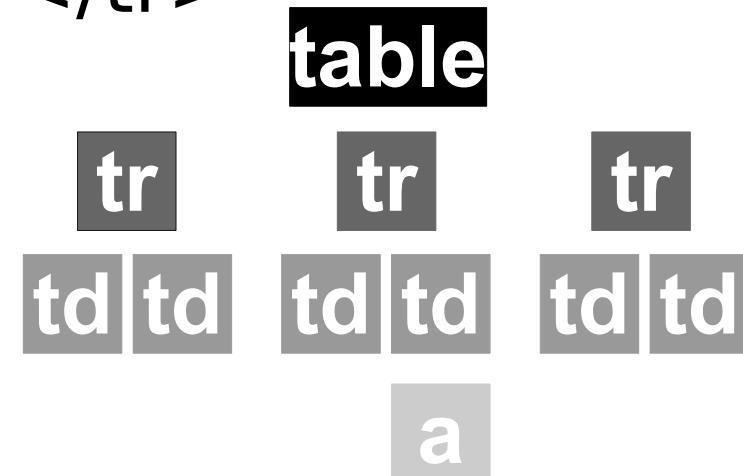
Scrapy Selector 基本語法 (1/2)



```
from scrapy.selector import Selector
data='<p><b>1</b>2<b>3</b></p>'
t=Selector(text=data)
print(t.xpath('//p').extract())
print(t.xpath('//p/text()').extract()) #2
print(t.xpath('//p/*').extract()) #1,3
print(t.xpath('//b/text()')) #1,3
print(t.xpath('//p').re('\d')) #2
```

Scrapy Selector 基本語法 (2/2)

```
from scrapy.selector import Selector  
data='''<table>  
<tr><td>11</td><td>12</td></tr>  
<tr><td>21</td><td><a href="http://ptt.cc">22</a></td></tr>  
<tr><td>31</td><td>32</td></tr>  
</table>'''  
  
t=Selector(text=data)  
print(t.xpath('//td'))  
print(t.xpath('//td/..')[1].xpath('*//text()'))  
print(t.xpath('//a/@href'))  
print(t.xpath('//a/text()'))
```



搜集連結 (links)

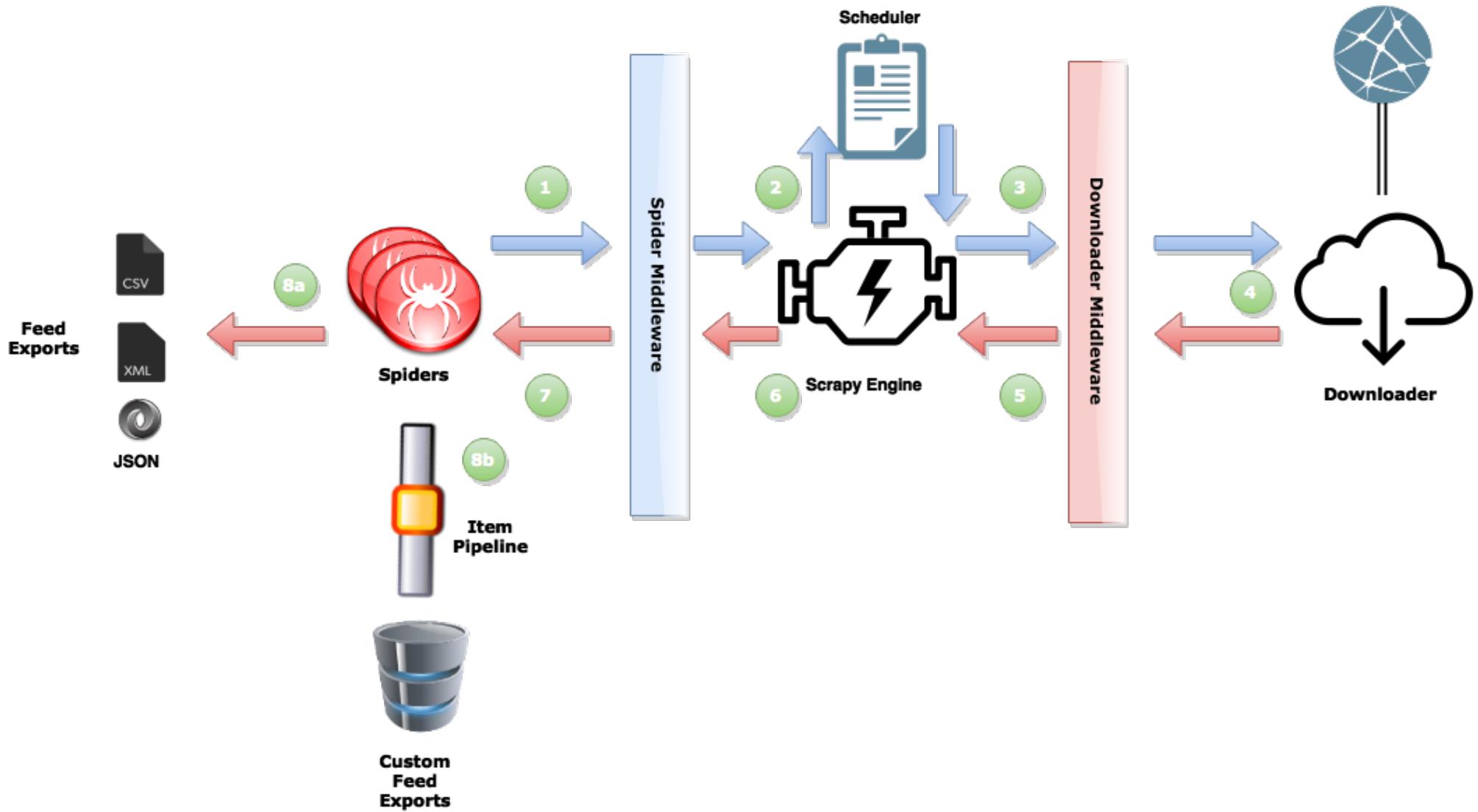
可在 Terminal 裡或是 Jupyter Notebook 跑
(Scrapy 裡有個 LinkExtractor)

```
import scrapy
class Spider(scrapy.Spider):
    name="ptt"
    start_urls=["http://www.ptt.cc/bbs/Boy-Girl/"]
    def parse(self, response):
        for link in response.xpath('//a'):
            print(link.xpath('text()').extract())
            print(link.xpath('@href').extract())
```

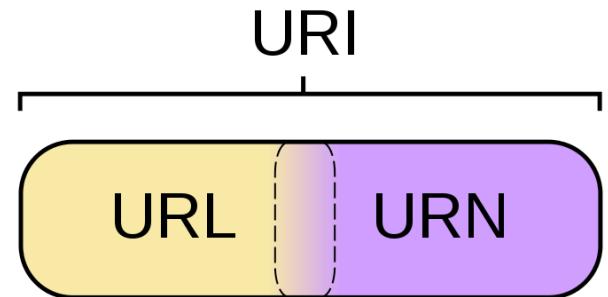


Scrapy 架構

殺雞用牛刀？



LXML 搜集文章資訊 (1/2)



```
import urllib, lxml.html
```

```
URL='http://www.ptt.cc'
```

```
URN='/bbs/Boy-Girl/M.1394964006.A.850.html'
```

```
h={'User-Agent':'Mozilla/5.0'}
```

```
r=urllib.request.Request(URL+URN,headers=h)
```

```
data=urllib.request.urlopen(r).read()
```

```
t=lxml.html.fromstring(data.decode('utf-8'))
```

```
print(t.text_content()) # 整頁
```

LXML 搜集文章資訊 (2/2)

```
x=t.xpath('//div[@id="main-content"]')[0]
print(x.text_content()) # 主文
y=t.xpath('//div[@id="main-content"]/text()')
print("\n".join(y)) # 新文
z=t.xpath('//span[@class="f6"]')
for i in z:
    print(i.text) # 引言
```



```
H=t.xpath('//*[@contains(text(),"恨")])[0]
print(H.text) # 含恨
```

18 禁的八卦版

```
import urllib.request  
u='http://www.ptt.cc/bbs/Gossiping/'  
h={'User-Agent':'Mozilla/5.0'}  
r=urllib.request.Request(u,headers=h)  
data=urllib.request.urlopen(r).read()  
print(data.decode('utf-8'))
```

SORRY
BUT
I'M
HACKER



本網站已依網站內容分級規定處理

警告：您即將進入之看板內容需滿十八歲方可瀏覽。

根據「電腦網路內容分級處理辦法」第六條第三款規定，本網站已於各限制級網頁依照台灣網站分級推廣基金會之規定標示。若您尚未年滿十八歲，請點選離開。若您已滿十八歲，亦不可將本區之內容派發、傳閱、出售、出租、交給或借予年齡未滿18歲的人士瀏覽，或將本網站內容向該人士出示、播放或放映。

我同意，我已年滿十八歲
進入

未滿十八歲或不同意本條款
離開

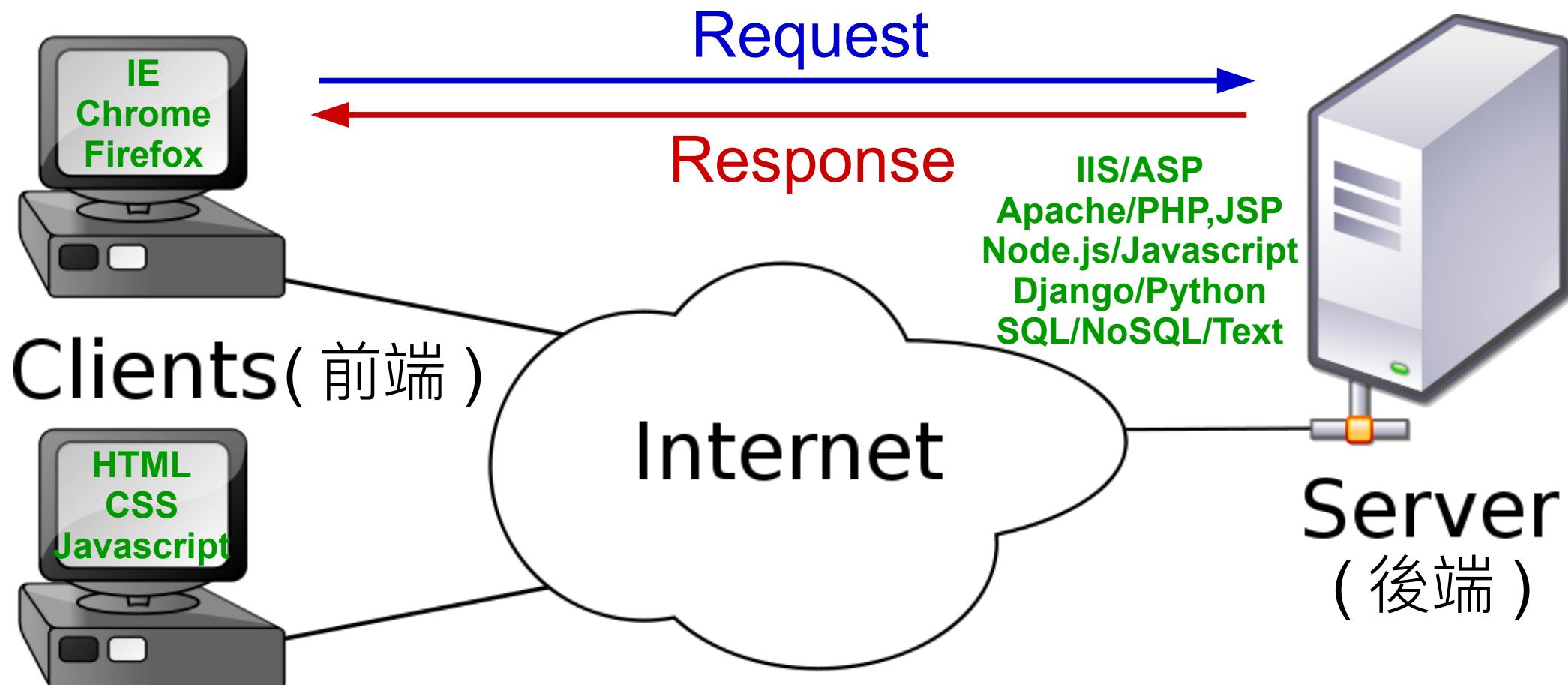
花生鼠模式？

藍字為前端請求 紅字為後端回應者

```
<div class="bbs-screen bbs-content center clear">  
  <form action="/ask/over18" method="post">  
    <input type="hidden" name="from"  
           value="/bbs/Gossiping/index.html">  
    <button class="btn-big" type="submit" name="yes"  
           value="yes">我同意，我已年滿十八歲 <br><small>進入 </sm  
           all></button>  
    <button class="btn-big" type="submit" name="no"  
           value="no">未滿十八歲或不同意本條款 <br><small>離開 </s  
           all></button>  
  </form>  
</div>
```

網頁前端 (frontend) 和後端 (backend)

前端資料由瀏覽器來處理，
主要是頁面的顯示和小資料的儲存 / 查詢



後端資料由伺服器來處理，主要是大資料的儲存 / 查詢

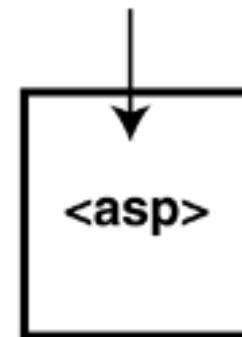
Request 傳送資料方式：Get vs. Post

Using GET

http://www.somedomain.com/register.asp?name=jobe&email=jobe@electrotank.com



比較方便



Using POST

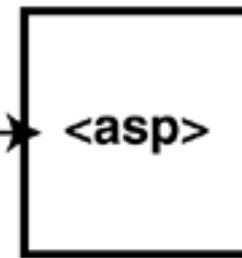
比較安全

http://www.somedomain.com/register.asp



HTTP Request

name=jobe&
email=jobe@
electrotank.com



破解 18 禁的八卦版 (1/3)

```
from urllib import parse,request
```

```
URL='https://www.ptt.cc'
```

```
URN='/ask/over18'
```

```
q=parse.urlencode({'yes':'yes','from':'/bbs/Gossiping/'})
```

```
q=q.encode('utf-8')
```

```
h={'User-Agent':'Mozilla/5.0'}
```

```
req=request.Request(URL+URN,q,h)
```

```
response=request.urlopen(req)
```

```
data=response.read()
```

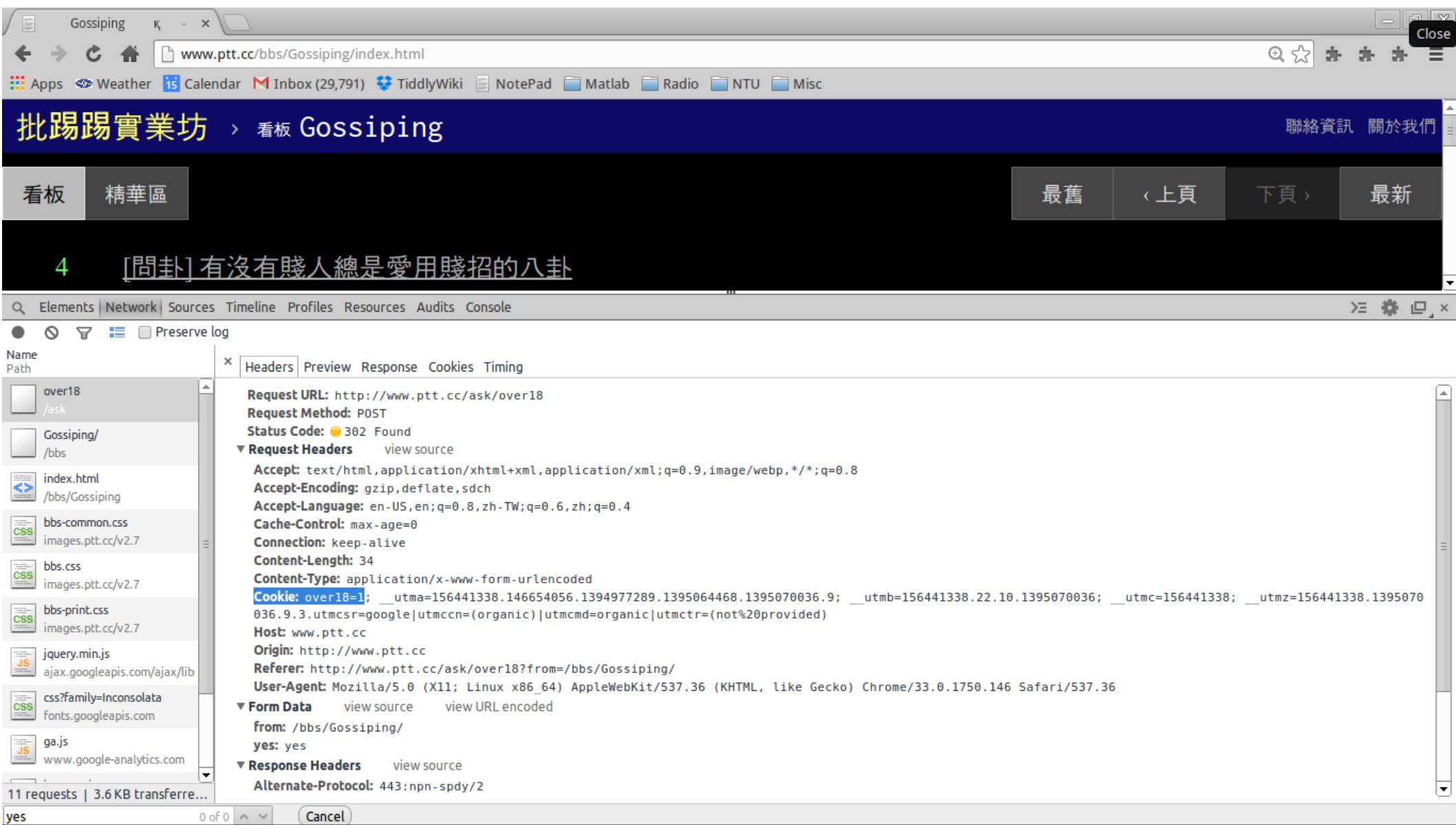
```
print(data.decode('utf-8'))
```

為何還是不行？



破解 18 禁的八卦版 (2/3)

看看瀏覽器到底做了什麼？



The screenshot shows a browser window displaying a forum page from PTT. The URL in the address bar is `www.ptt.cc/bbs/Gossiping/index.html`. The page title is "批踢踢實業坊 > 看板 Gossiping". Below the title, there are navigation buttons for "最舊" (Oldest), "< 上頁" (Previous page), "下頁 >" (Next page), and "最新" (Latest). The main content area shows a thread titled "4 [問卦] 有沒有賤人總是愛用賤招的八卦". At the bottom of the screen, the browser's developer tools Network tab is open, showing a list of requests. The selected request is a POST to `/ask/over18`. The Headers section shows the following details:

- Request URL: `http://www.ptt.cc/ask/over18`
- Request Method: POST
- Status Code: 302 Found
- Request 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-US,en;q=0.8,zh-TW;q=0.6,zh;q=0.4
 - Cache-Control: max-age=0
 - Connection: keep-alive
 - Content-Length: 34
 - Content-Type: application/x-www-form-urlencoded
- Cookie: `over18=1; __utma=156441338.146654056.1394977289.1395064468.1395070036.9; __utmb=156441338.22.10.1395070036; __utmc=156441338; __utmz=156441338.1395070036.9.3.utmcsr=google|utmccn=(organic)|utmcmd=organic|utmctr=(not%20provided)`
- Host: `www.ptt.cc`
- Origin: `http://www.ptt.cc`
- Referer: `http://www.ptt.cc/ask/over18?from=/bbs/Gossiping/`
- User-Agent: `Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/33.0.1750.146 Safari/537.36`

Below the Headers, there are sections for Form Data, Response Headers, and Response Body.

破解 18 禁的八卦版 (3/3)

```
from urllib import parse,request  
URL='https://www.ptt.cc'  
URN='/ask/over18'  
q=parse.urlencode({'yes':'yes','from':'/bbs/Gossiping/'})  
q=q.encode('utf-8')  
h={'Cookie':'over18=1','User-Agent':'Mozilla/5.0'}  
req=request.Request(URL+URN,q,h)  
response=request.urlopen(req)  
data=response.read()  
print(data.decode('utf-8'))
```



更進階的爬蟲議題

師父領進門，修行在個人

驗證碼：[tesserocr/pytesseract](#), [selenium](#), [touclick](#)

JS 產生的動態資料：[scrapy-splash](#)

分散式爬取：[scrapy-redis](#)



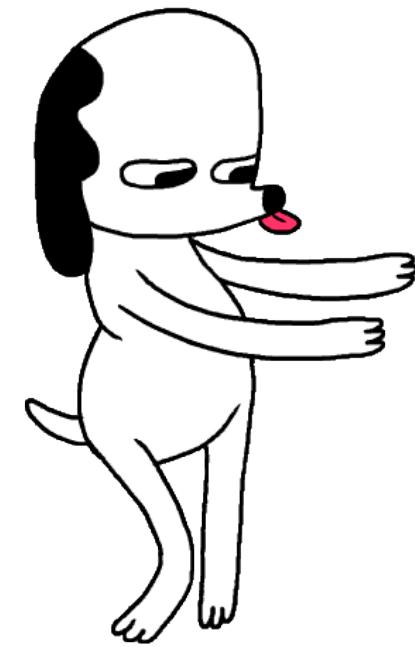
神 Selenium 神

Selenium 可自動化一切瀏覽動作

```
from selenium import webdriver  
URI='https://www.ptt.cc/bbs/Gossiping/'  
driver=webdriver.Chrome() # try Firefox()  
driver.get(URI)  
  
btn=driver.find_element_by_name('yes')  
driver.save_screenshot('before_click.png')  
btn.click()  
  
driver.save_screenshot('after_click.png')  
print(driver.page_source)
```



輕輕鬆鬆，
打完收工！



本週作業

進一步搜尋 Boy-Girl 版資訊

1. index.html 右上角 [[< 上頁](#)] 中包含了總頁數資訊，請用 LXML 抓出此經常變動的數字。
2. 請用 LXML 找出距離現在時間最近的一篇 [爆] 文標題與 URN 。
3. 請用 Selenium 在 index.html 往前翻三頁並拍照。

Game Over

