

心理與神經資訊學

(Psychoinformatics & Neuroinformatics)

課號: Psy5261

教室: 綜合302

識別碼: 227U9340

時間: 五234





作業可討論但請不要抄襲喔

!

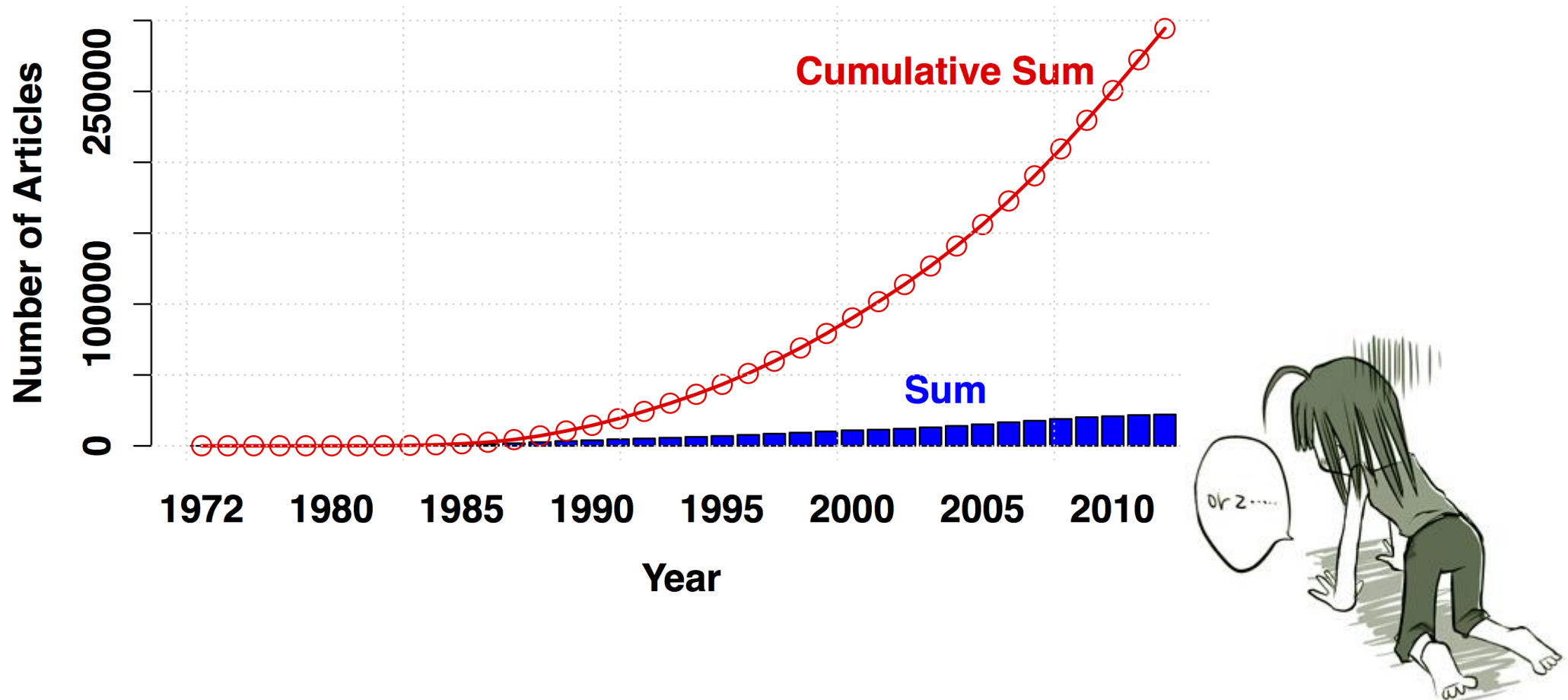
網頁前端技術

(HTML5, CSS, & JavaScript)

腦科學案例研究(1/3)

論文怎麼讀都讀不完

Human fMRI Publications

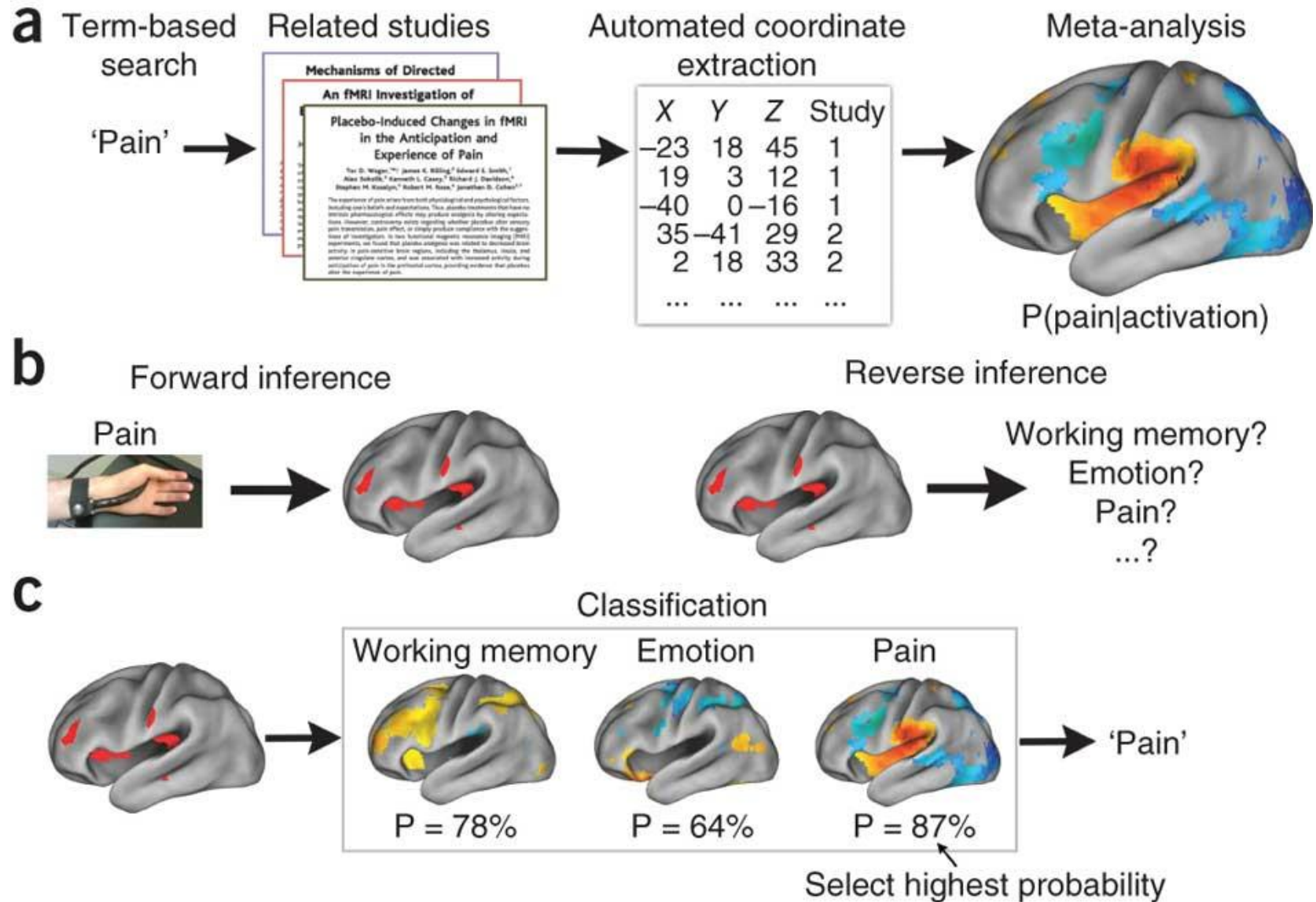


今天學完就可以讓程式可以幫我們讀論文



腦科學案例研究(3/3)

用腦座標取代腦名稱可以更精確



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 ...

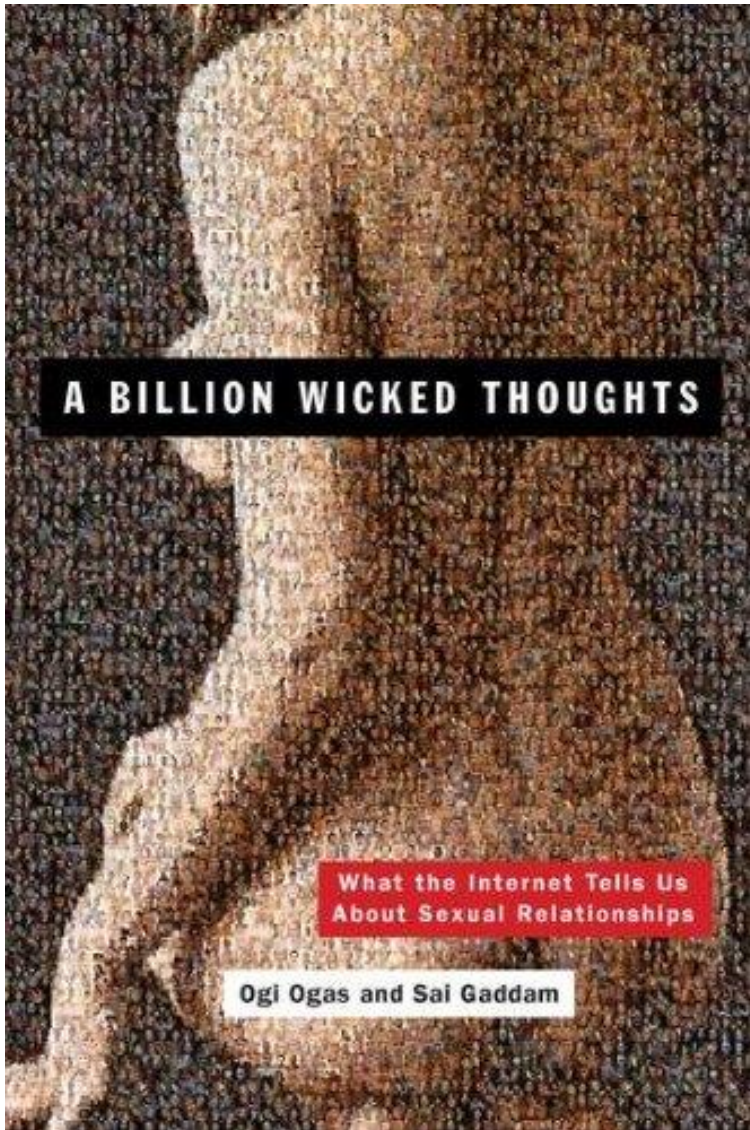


感謝大大無私分享
Orz



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.request
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'))
```

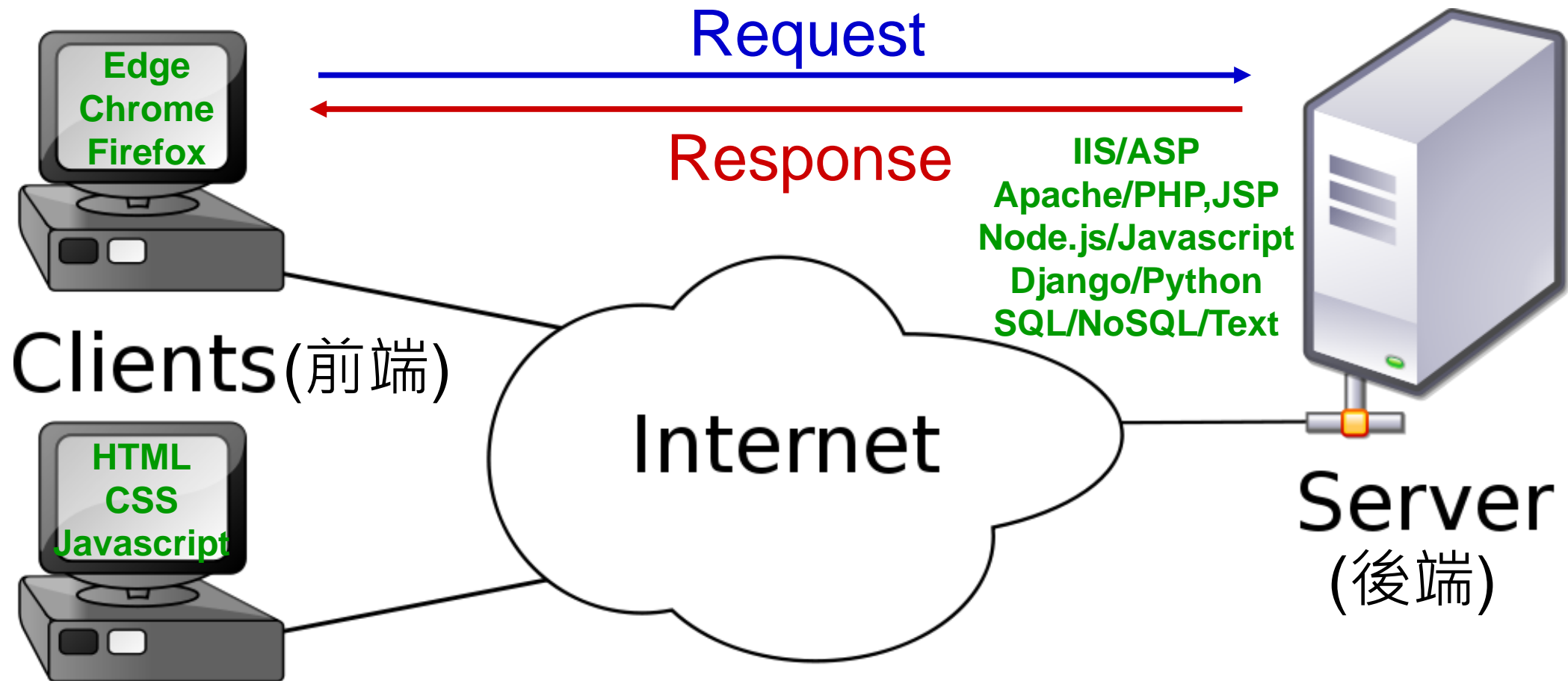
這些是什麼東東？

```
<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`</h2>```連接``
這樣`
`可以斷行 ``
`<center>`這樣可以置中和``換色```</center>`
``
````粗體````
```<i>`斜體`</i>``` ```<u>`底線`</u>```
``
`<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](#)學

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

```
This is <span class=yy>test1</span><hr>
```

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

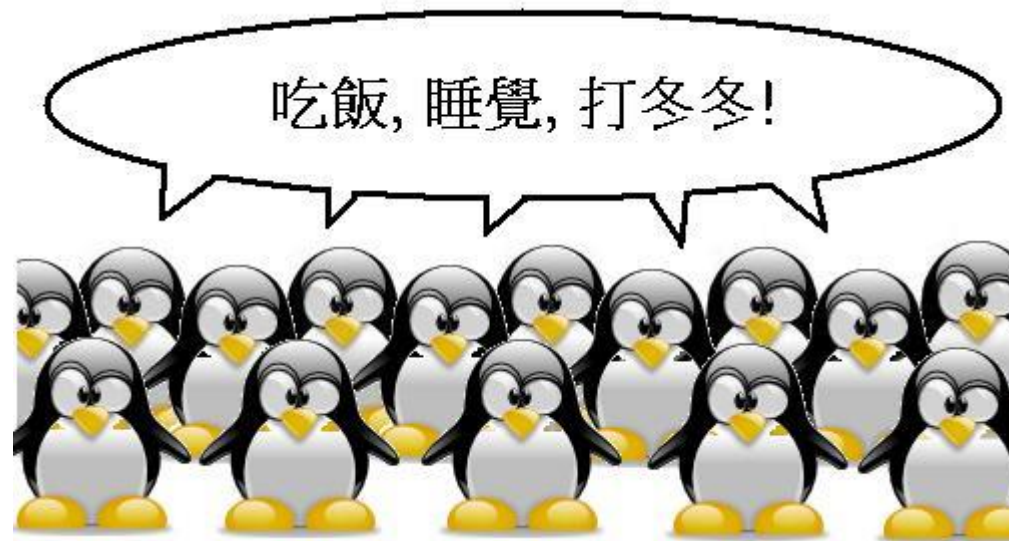
```
This is <span id=gg>test3</span><hr>
```

```
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 web browser window with the address bar displaying `www.ptt.cc/bbs/Boy-Girl/index1393.html`. The browser's toolbar includes navigation buttons (back, forward, refresh, home) and a search icon. Below the address bar, there are several tabs: Apps, Weather, Calendar, Inbox (29,791), TiddlyWiki, NotePad, Matlab, Radio, NTU, and Misc. The main content area of the browser shows a forum page titled "批踢踢實業坊" with a sub-header "看板 Boy-Girl". The page lists several forum posts. The first post is titled "Re: [心情] 我要怎麼原諒那個人" and is by user "wpson". The second post is also titled "Re: [心情] 我要怎麼原諒那個人" and is by user "mileslo". The third post is titled "Re: [求助] 約好單獨出遊卻變成三人出遊，什麼意思？" and is by user "Azabulu". The browser's developer tools are open at the bottom, showing the "Elements" panel on the left and the "Styles" panel on the right. The "Elements" panel shows the HTML structure of the page, with the following code highlighted:

```
<!DOCTYPE html>
<html>
  <head>...</head>
  <body>
    <div id="topbar-container">...</div>
    <div id="main-container">
      <div id="action-bar-container">...</div>
      <div class="r-list-container bbs-screen">
        <div class="r-ent">
          <div class="nrec">...</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>
        </div>
      </div>
    </div>
  </body>
</html>
```

The "Styles" panel on the right shows the CSS rules for the selected element. The rules are:

- `element.style { }`
- `media="screen" a:visited { color: #888; }`
- `media="screen" a:link { color: #aaa; }`
- `a:-webkit-any-link { user agent stylesheet color: -webkit-link; text-decoration: underline; }`

The browser's status bar at the bottom shows the current page is `html body div#main-container div.r-list-container.bbs-screen div.r-ent div.title a`.

工欲善其事必先利其器(2/2)

Firefox也有對應的Developer Tools (原Firebug)

The screenshot shows the Mozilla Firefox browser window displaying the Ptt forum page '批踢踢實業坊' (Ptt Forum). The address bar shows the URL 'www.ptt.cc/bbs/Boy-Girl/index1393.html'. The page title is '看板 Boy-Girl 文章列表 - 批踢踢實業坊 - Mozilla Firefox'. The page content shows a list of forum posts. The first post is 'Re: [心情] 我要怎麼原諒那個人' by 'wbson' on 3/16. The second post is 'Re: [心情] 我要怎麼原諒那個人' by 'mileslo' on 3/16, with a note '(本文已被刪除) [Azabulu]'. The third post is 'Re: [求助] 約好單獨出遊卻變成三人出遊, 什麼意思?'.

The Firefox Developer Tools interface is open at the bottom. The 'HTML' tab is selected, showing the DOM tree. The selected element is a link with the text 'Re: [心情] 我要怎麼原諒那個人'. The 'Style' tab is also open, showing the CSS rules for the selected element. The rules include 'a:hover' and 'a:link' from 'bbs.css'. The 'a:link' rule has a color of 'AAAAAA'.

DOM Tree:

```
<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>
    <div class="r-ent">
    </div>
    <div class="r-ent">
    </div>
  </div>
</div>
</div>
```

Style Rules:

```
a:hover bbs.css (line 118)
{
  background-color: #CCCCC;
  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)

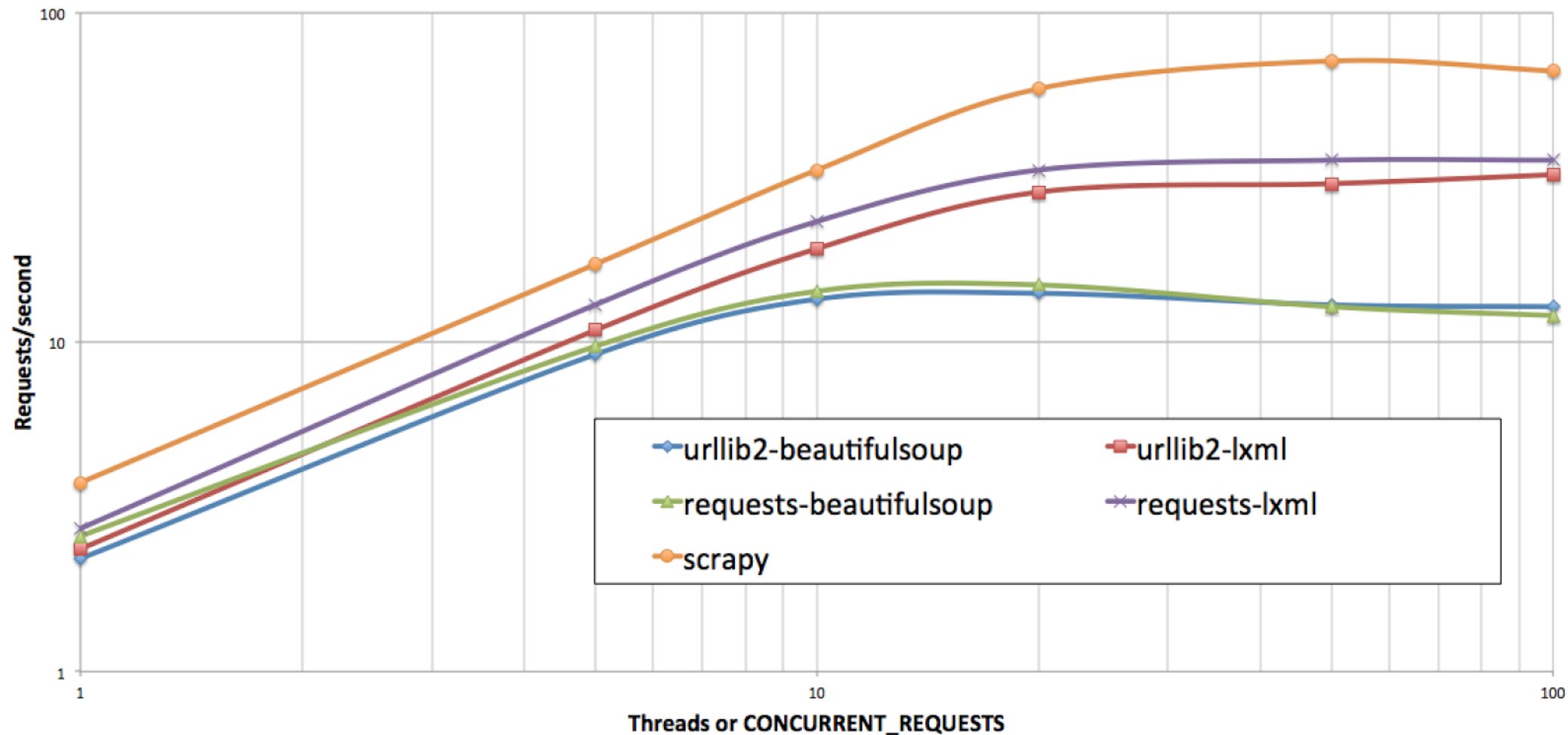
不同"爬蟲"的比較(1/3)

Beautiful Soup是最好學/寫的

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

不同"爬蟲"的比較(2/3)

Beautiful Soup卻是處理速度最慢的

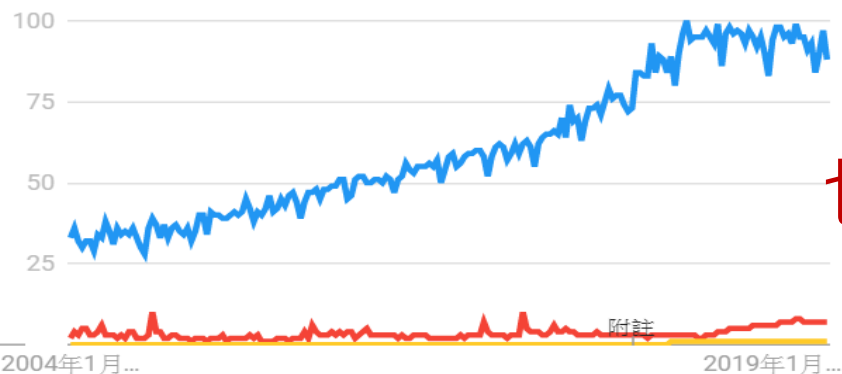


不同"爬蟲"的比較(3/3)

搜尋熱度的趨勢變化

Google Trends

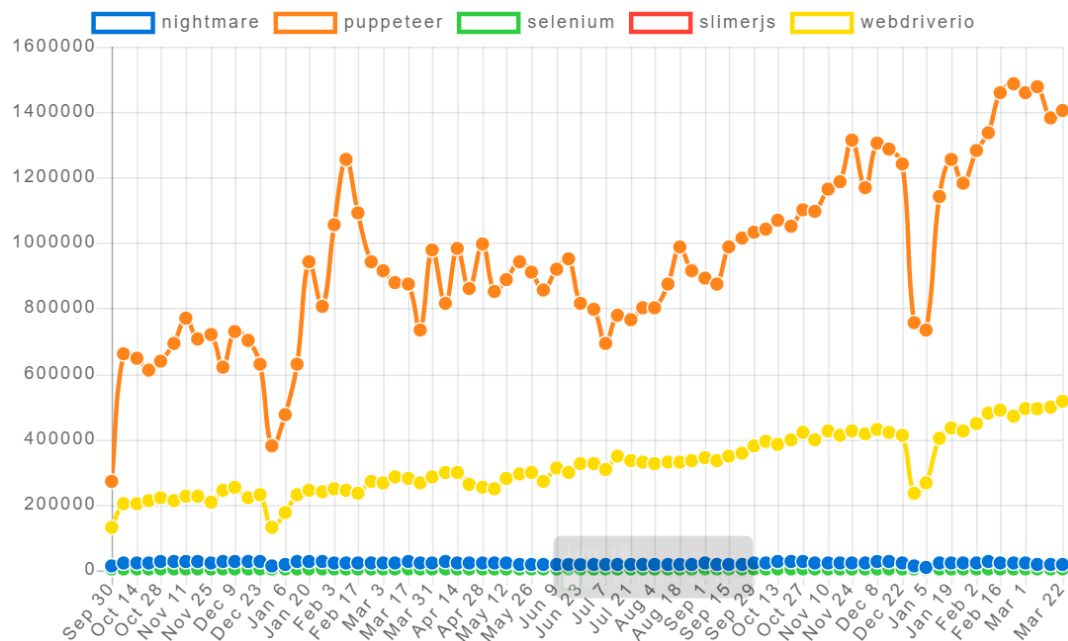
● Selenium ● puppeteer ● webdriverio



也有用JS寫的瀏覽器控制器

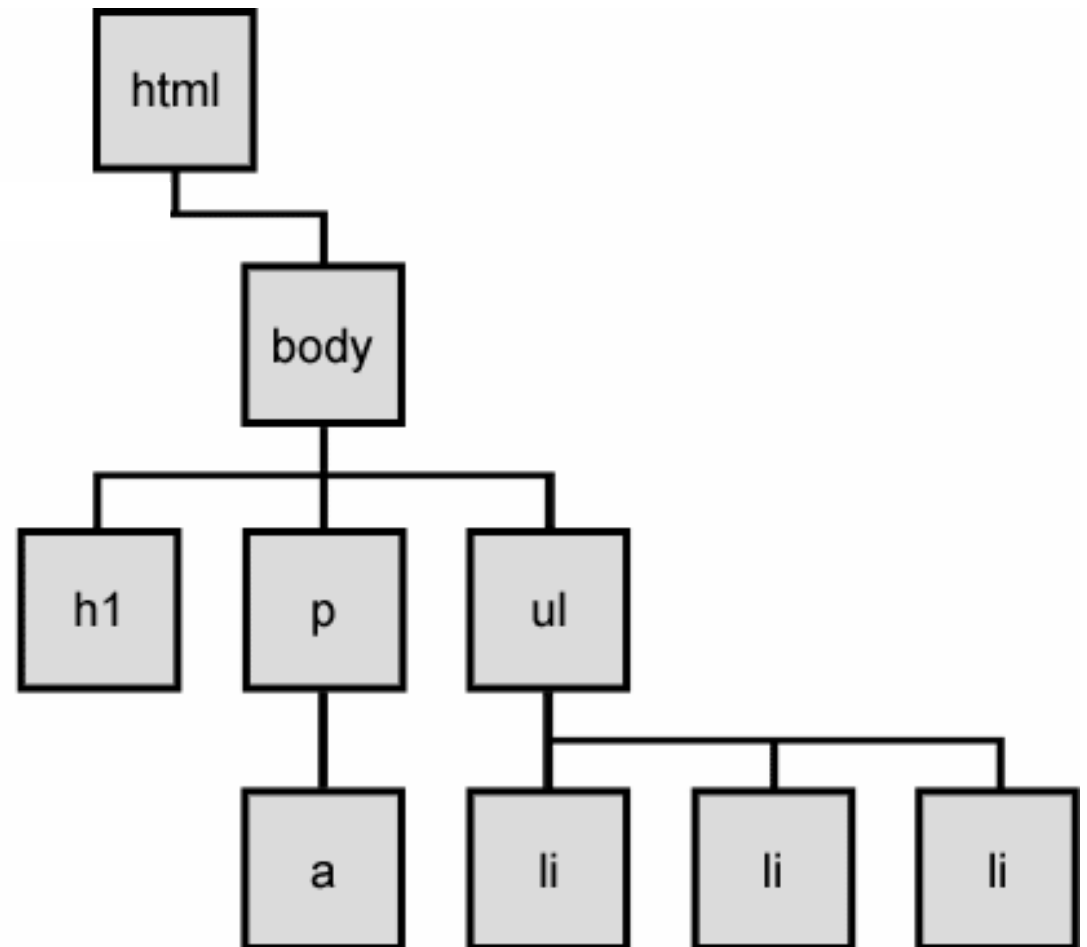
最夯的是

Downloads in past 2 Years

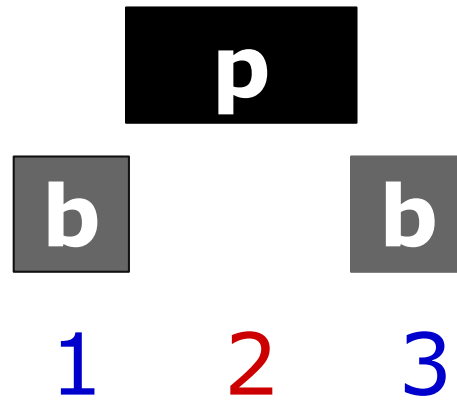


LXML: HTML Parser

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>
```

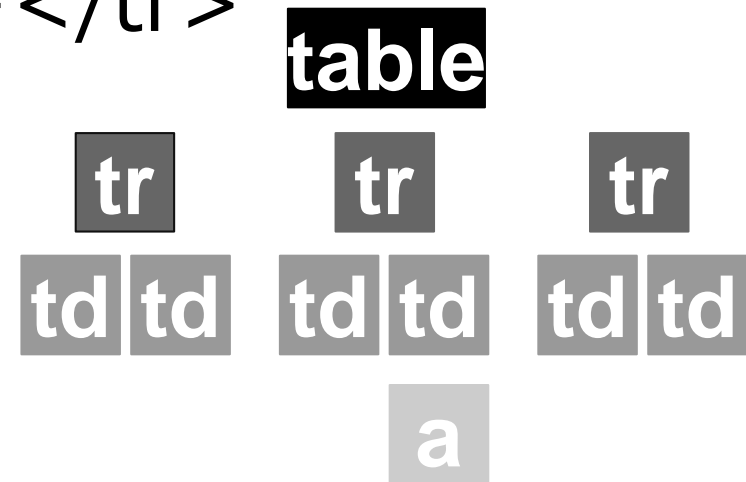
```
<tr><td>11</td><td>12</td></tr>
```

```
<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

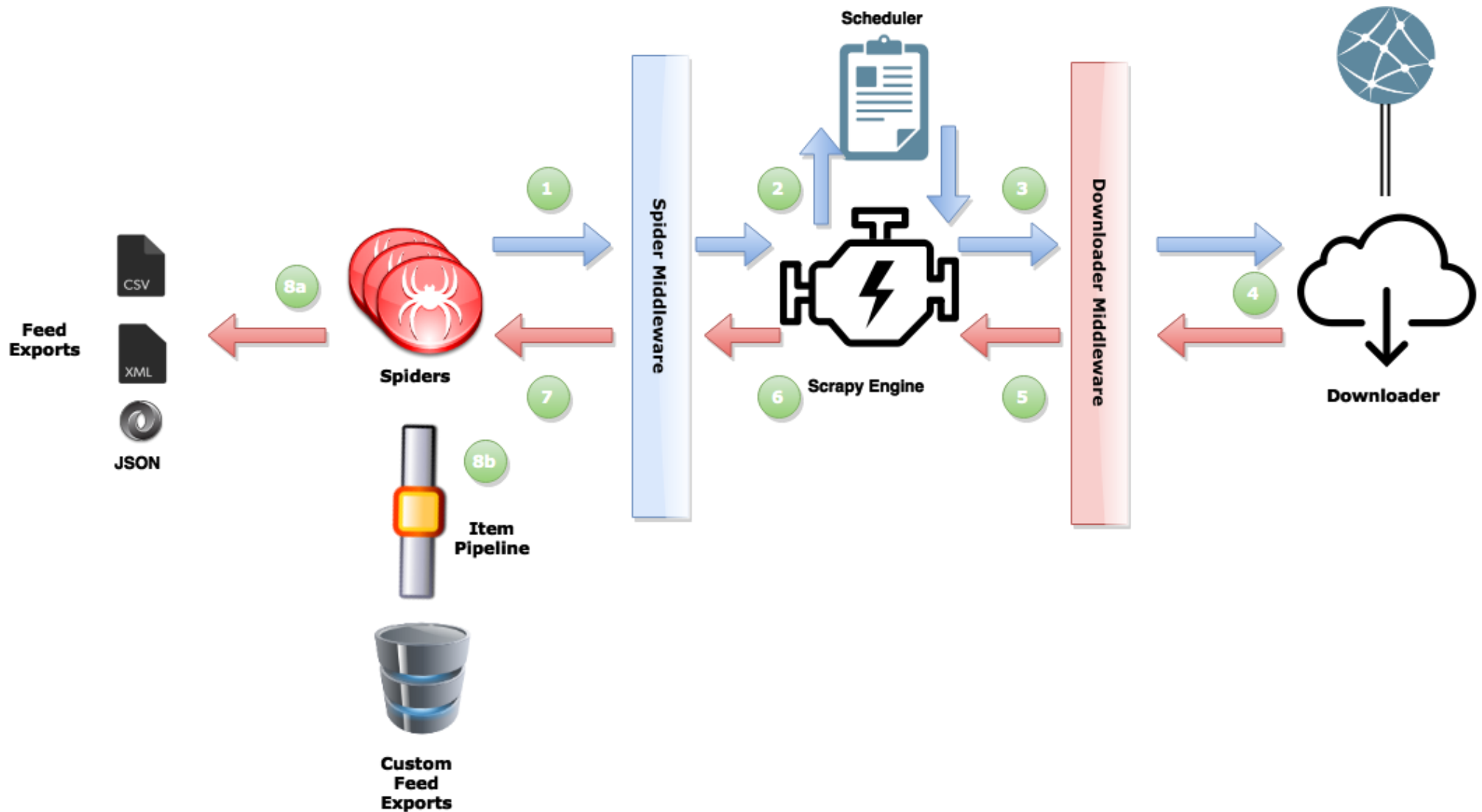
(Scrapy裡其實有個LinkExtractor)

```
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())
```

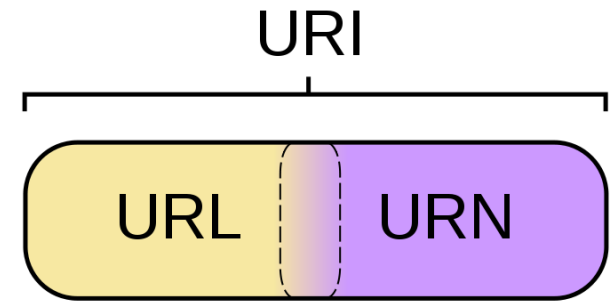


Scrapy架構

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("".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

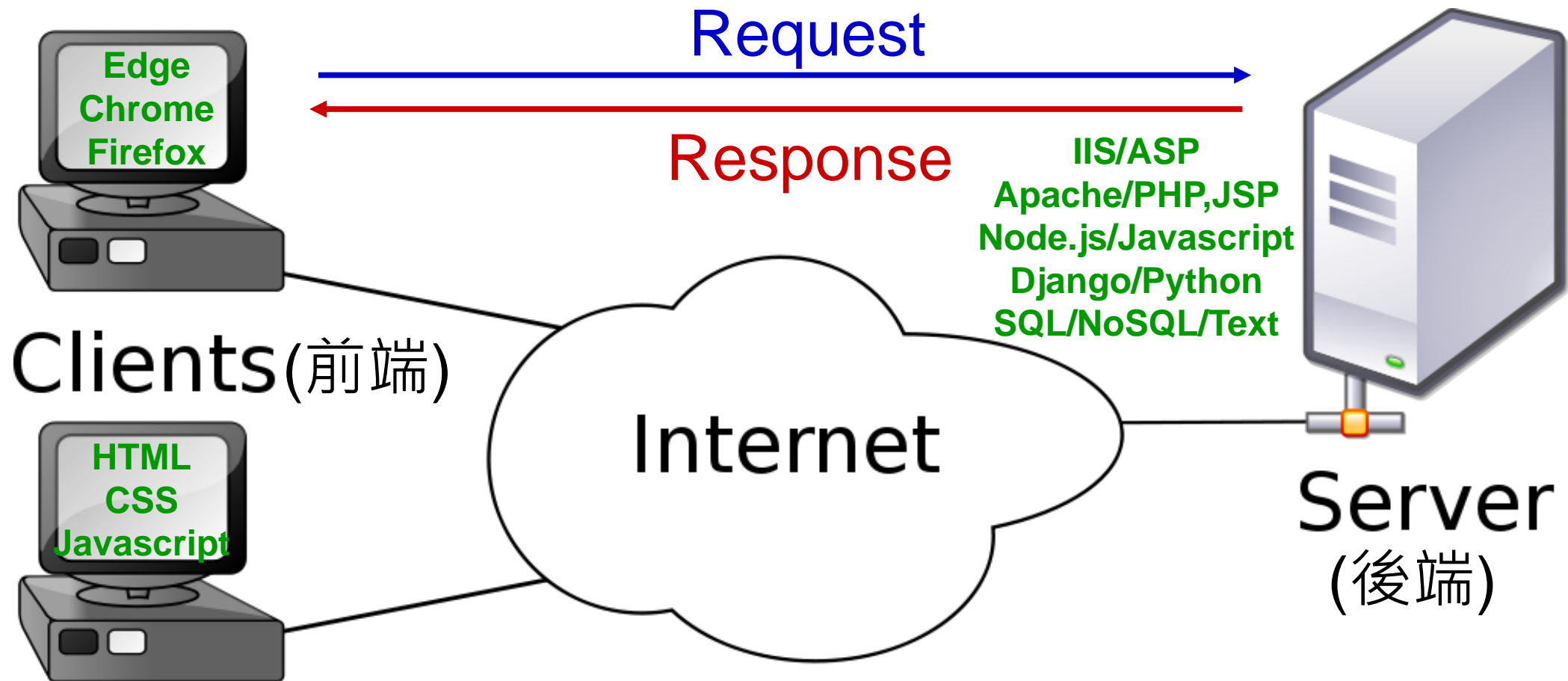
花生鼠模式？

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
</small></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
</small></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

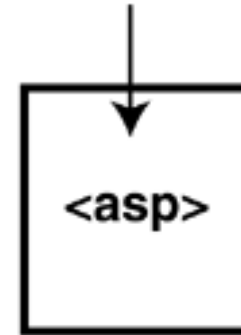
Request傳送資料方式: Get vs. Post

Using GET

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



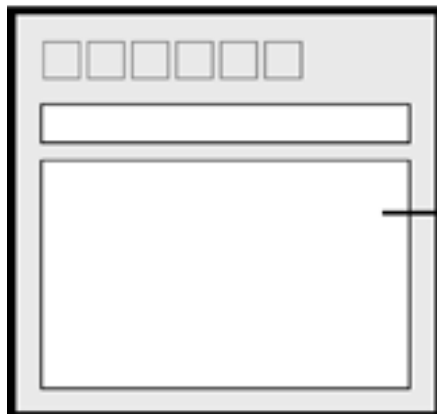
比較方便



Using POST

比較安全

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



HTTP Request

`name=jobe&
email=jobe@
electrotank.com`

`<asp>`

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 web browser window with the address bar displaying `www.ptt.cc/bbs/Gossiping/index.html`. The page title is "批踢踢實業坊 > 看板 Gossiping". The main content area shows a forum post titled "[問卦] 有沒有賤人總是愛用賤招的八卦". The browser's developer tools are open, showing the "Network" tab with a list of requests. The selected request is for `http://www.ptt.cc/ask/over18`, which is a POST request. The "Headers" tab is active, showing 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`
- Form Data:**
 - from:** `/bbs/Gossiping/`
 - yes:** `yes`
- Response Headers:**
 - Alternate-Protocol:** `443:npn-spdy/2`

The bottom of the browser window shows a status bar with "11 requests | 3.6 KB transfered" and a "Cancel" button.

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: [pytesseract](#), [selenium](#), [touchclick](#)

JS產生的動態資料: [scrapy-splash](#)

分散式爬取: [scrapy-redis](#)



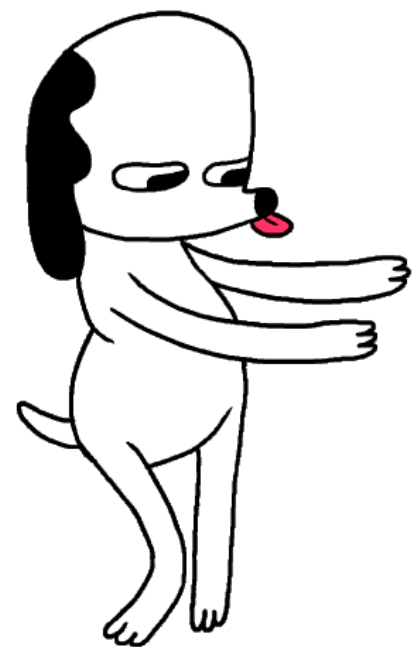
神 Selenium 神

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

