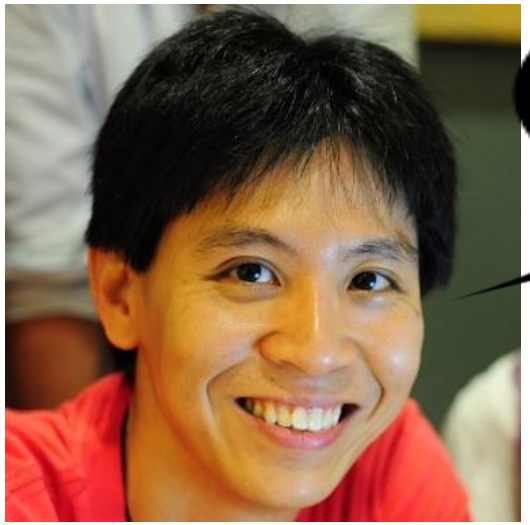
Psychoinformatics & Neuroinformatics

Week 4

Web Scraper

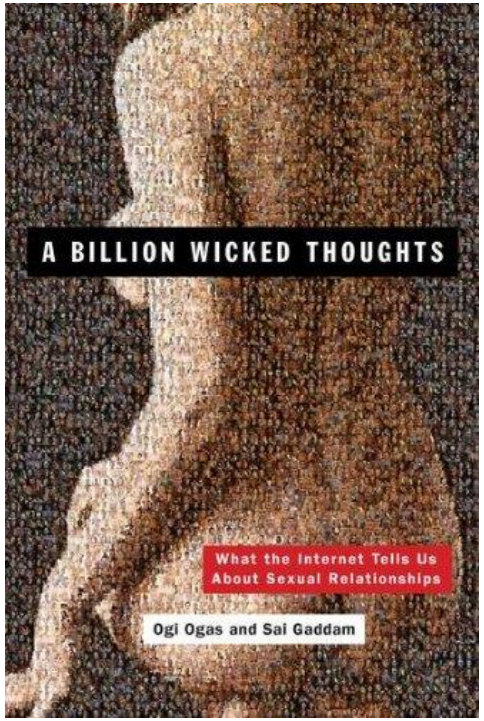
by Tsung-Ren (Tren) Huang 黃從仁

Tren: How to get these data from dogpile?



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

Tren: Thanks for your generous sharing!



Sifu opens the door, but you come & go by yourself!

Relevant Job Demands

9/20 **Python 爬蟲工程師**

橙航有限公司 | 人力仲介代徵

台北市大安區 | 2年以上 | 專科

●專業技能 1. 2年以上爬蟲開發經驗，熟悉瀏覽器原理、前端 JS、AJAX 2. 熟悉一種主流的爬取技術及爬蟲框架工具，如火車採集器/Scrapy/Selenium等 3. 熟悉多線程、網絡編程、網頁抓取原理、正則表達式、HTTP協議 4. 熟

月薪65,000~130,000元

9/22 **Software engineer/軟體工程師(爬蟲)**

香港商香港球鞋團隊有限公司 | 綜合商品批發代理業

台北市中正區 | 3年以上 | 大學

welcome proposals for new tooling stacks. *** 軟體工程師 (爬蟲) Data Pipeline / ETL: from Crawler to DB 1. Familiar with Python 2.

月薪60,000元以上 員工100人 距捷運台北車站340公尺

9/22 **Python 爬蟲工程師**

德采有限公司 | 電腦系統整合服務業

台北市大安區 | 2年以上 | 大學

●專業技能 1. 2年以上爬蟲開發經驗，熟悉瀏覽器原理、前端 JS、AJAX 2. 熟悉一種主流的爬取技術及爬蟲框架工具，如火車採集器/Scrapy/Selenium等 3. 熟悉多線程、網絡編程、網頁抓取原理、正則表達式、HTTP協議 4. 熟

月薪60,000~140,000元 員工30人

glassdoor

All Job Types Posted Any Time $63K-$138K More

Create Job Alert

25 Miles All Cities Company Ratings All Industries

Most Relevant

114 web scraping jobs Jobs

Recorded Future

4.4 ★

Senior Engineer, Open Web Collections - Python and Open Web Scraping

Somerville, MA

$104K - $159K (Glassdoor est.)

Viewed on September 25

Easy Apply 22d

Recorded Future

4.4 ★

Data Engineer II, Dark Web Collections, Python & Web Scraping

Somerville, MA

$73K - $107K (Glassdoor est.)

Viewed on September 25

Easy Apply 22d

Legalist

4.4 ★

Web Scraping Engineer

San Francisco, CA

$79K - $113K (Glassdoor est.)

Easy Apply 30d+

The PRACTIAL uses for web scrapping

Market/marketing research

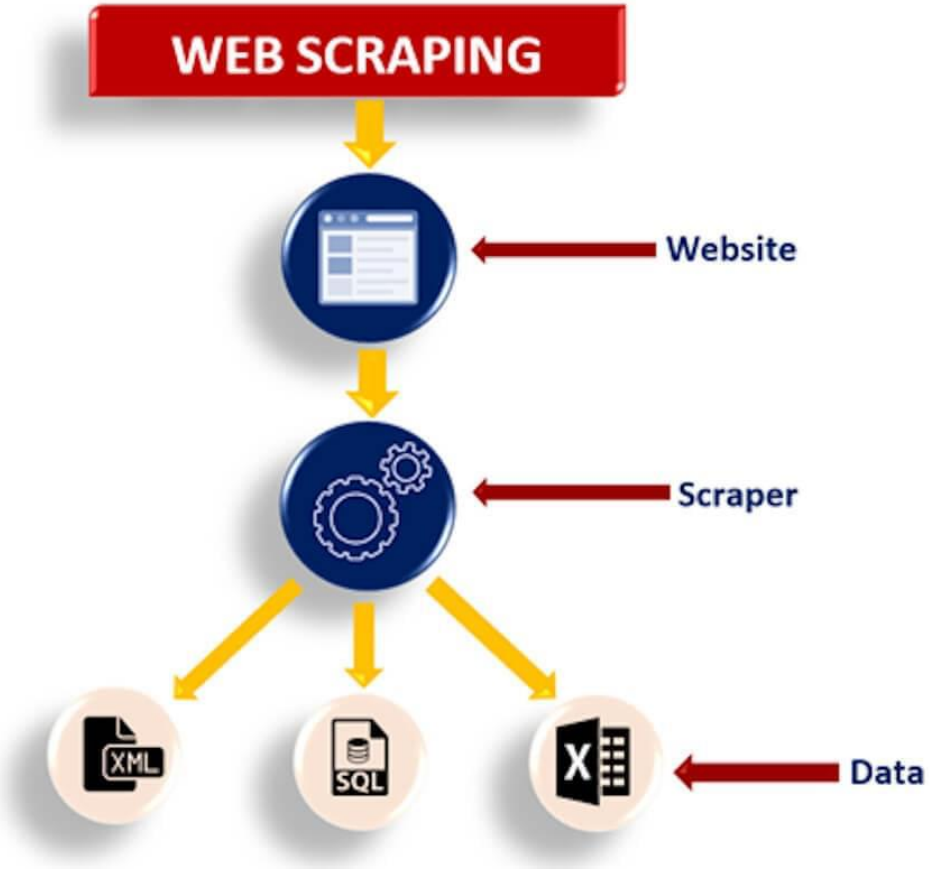
Understand your market/marketing

Building (real-time, big) data products

E.g., Google services & PayPal Honey

Building AI products

Your shiny AI (e.g. LLMs), needs more training data



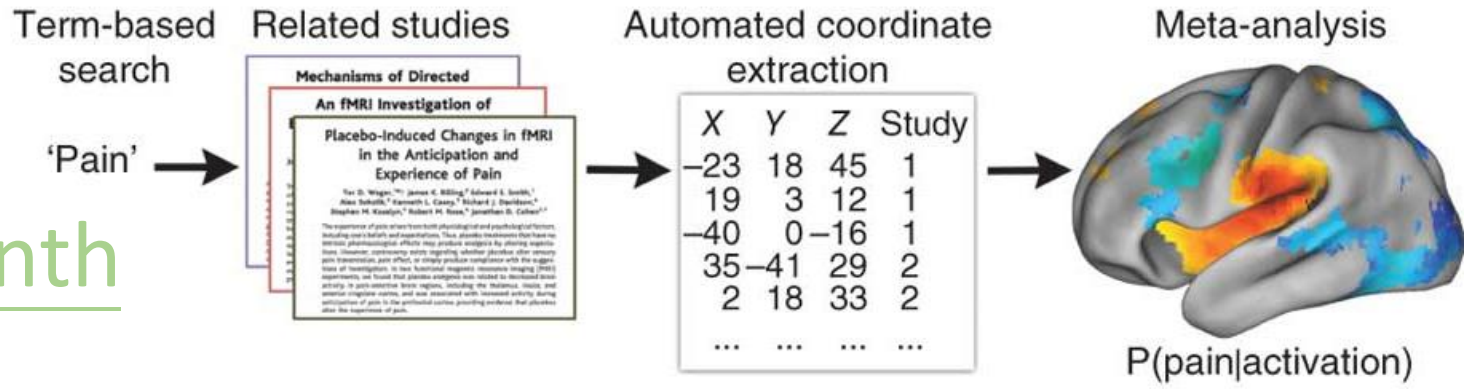
The ACADEMIC uses for web scrapping

(Offline) big data research

For a review, see this article

Large-scale meta-analysis

E.g., metaBUS & NeuroSynth



Building AI products for further analyses

E.g., gender prediction for Twitter data like this & this

Goals for today

Learning web basics

HTML & CSS

Learning basic scrapping skills

Beautiful Soup & Scrapy

Learning advanced scrapping skills

Form, cookie, Selenium, CAPTCHA

Goals for today

Learning web basics

HTML & CSS

Learning basic scrapping skills

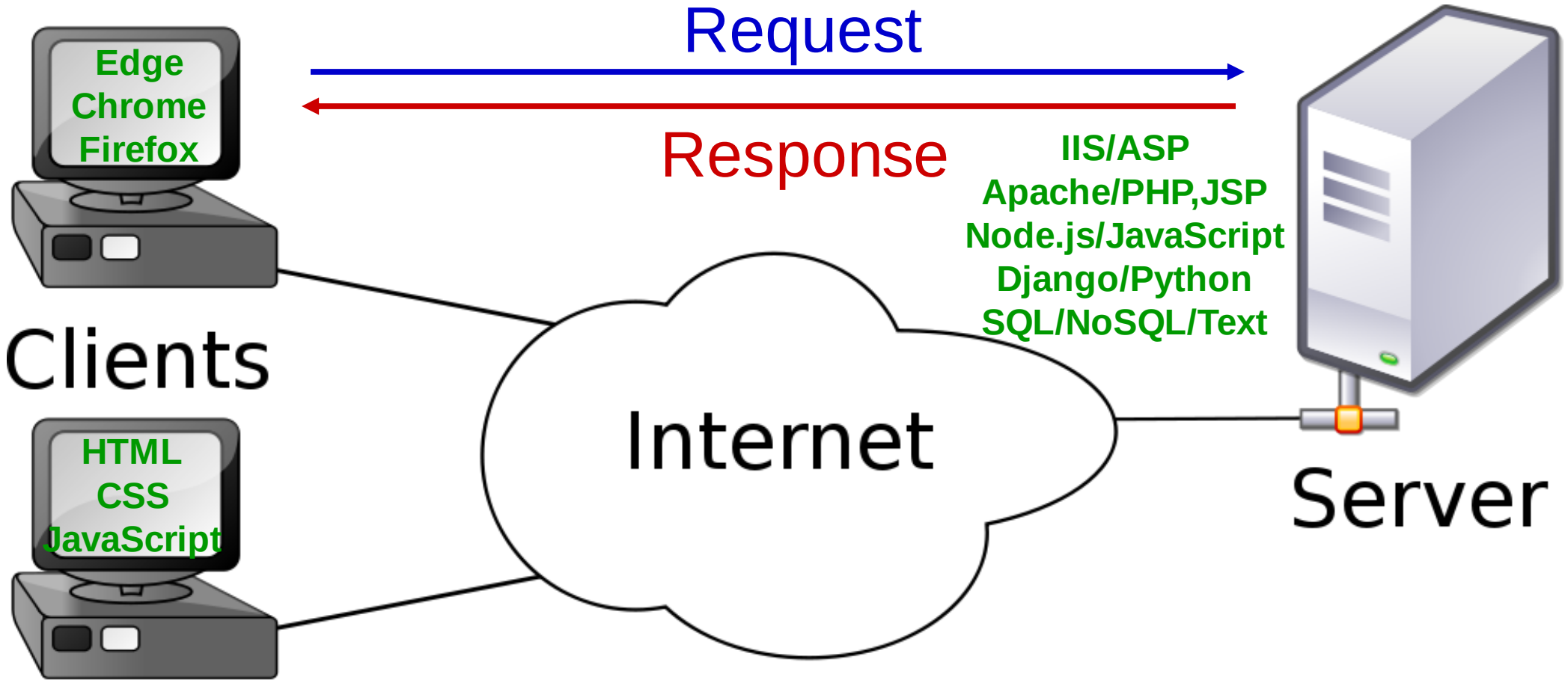
Beautiful Soup & Scrapy

Learning advanced scrapping skills

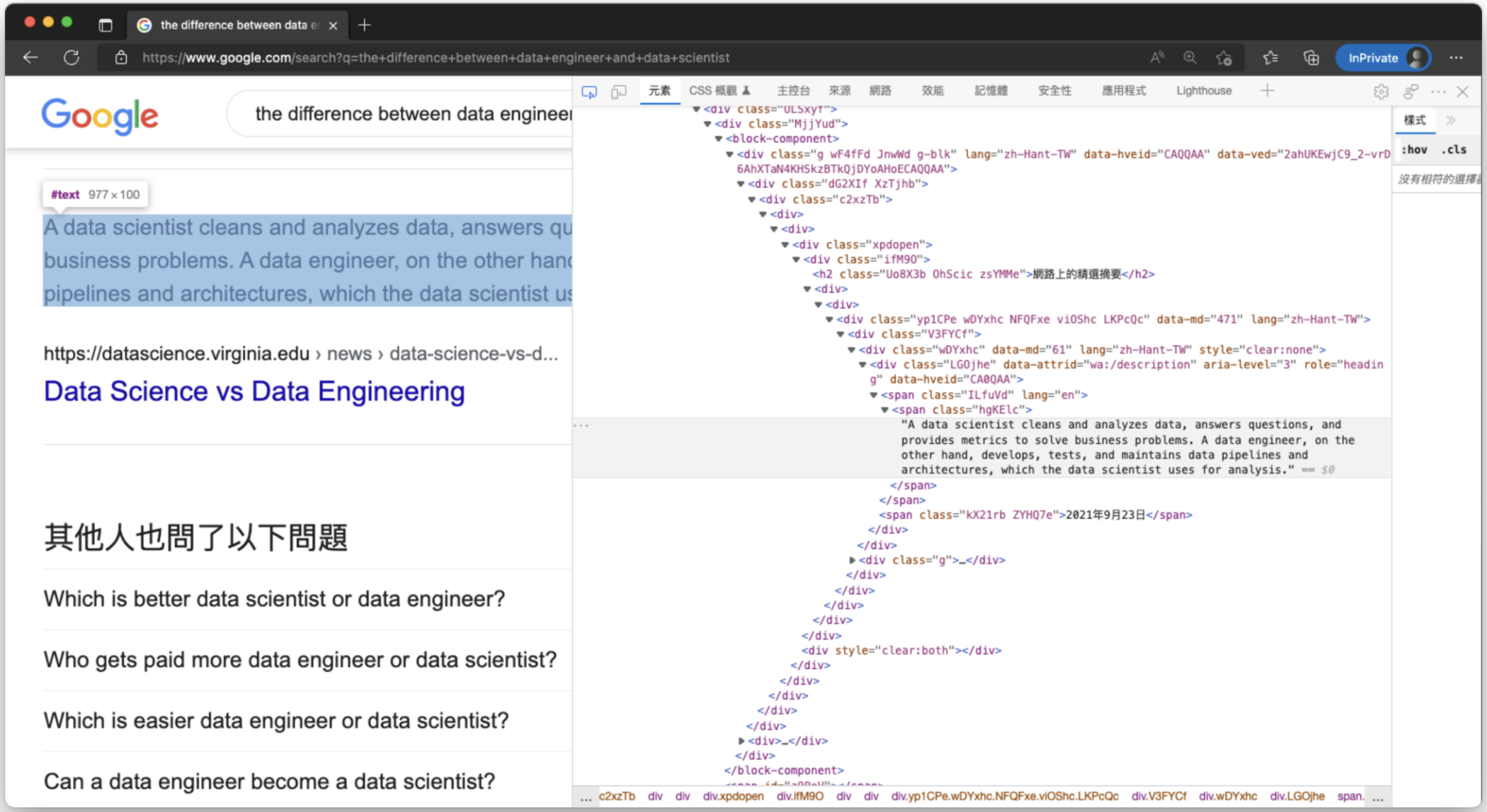
Form, cookie, Selenium, CAPTCHA

Frontend vs. Backend

Client=Frontend for getting data; Server=Backend for giving data



Source codes of a webpage



HyperText Markup Language (HTML)

```
<h1>h1</h1><hr><h2>h2</hr> <a href=http://ntu.edu.tw>link</a>
You<br>can break lines<img src=http://bit.ly/1p2za46>
<center>Center & <font color=red>change colors</font></center>
<ul>
<li><b>Bold</b></li>
<li><i>Italics</i></li>   <li><u>Underscore</u></li>
</ul>
<table border=1>
<tr><td>11</td>   <td>12</td></tr>
<tr><td>21</td>          <td>22</td></tr>
</table>
```

Cascading Style Sheets (CSS)

```
<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>
<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>
```

Goals for today

Learning web basics

HTML & CSS

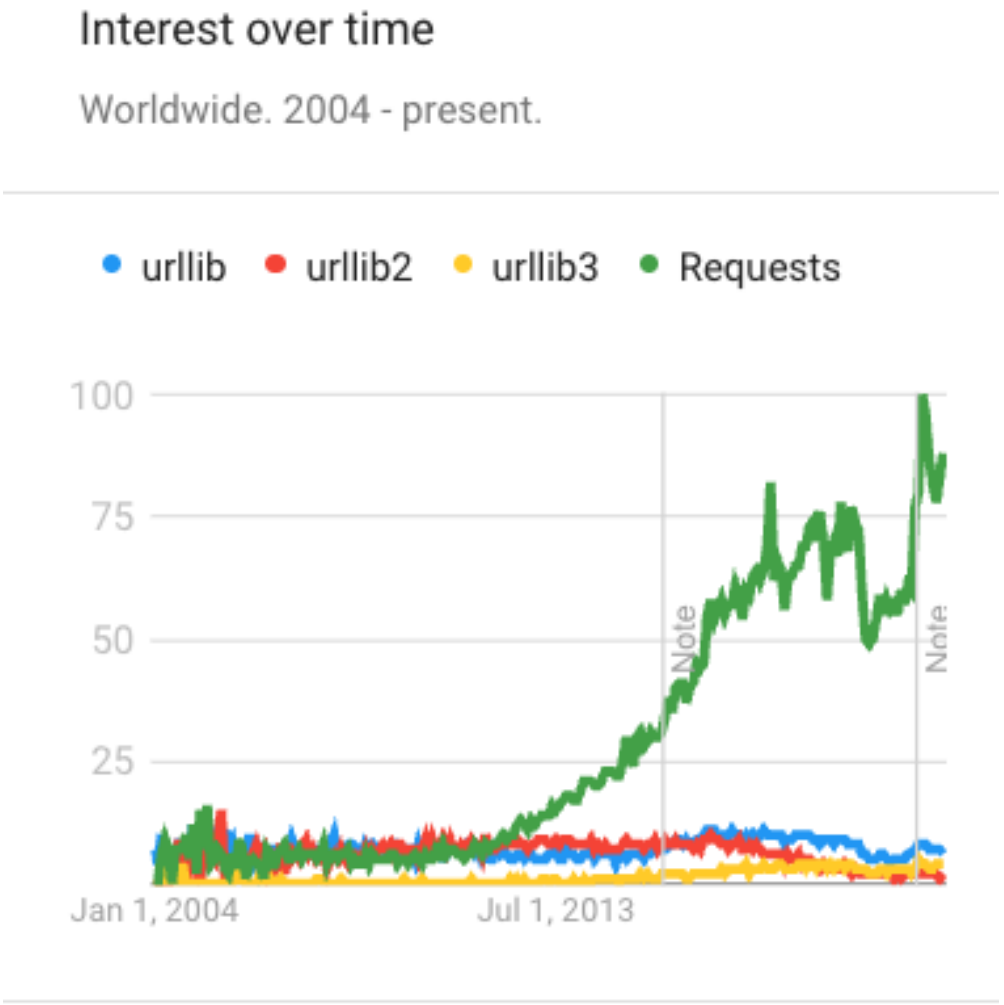
Learning basic scrapping skills

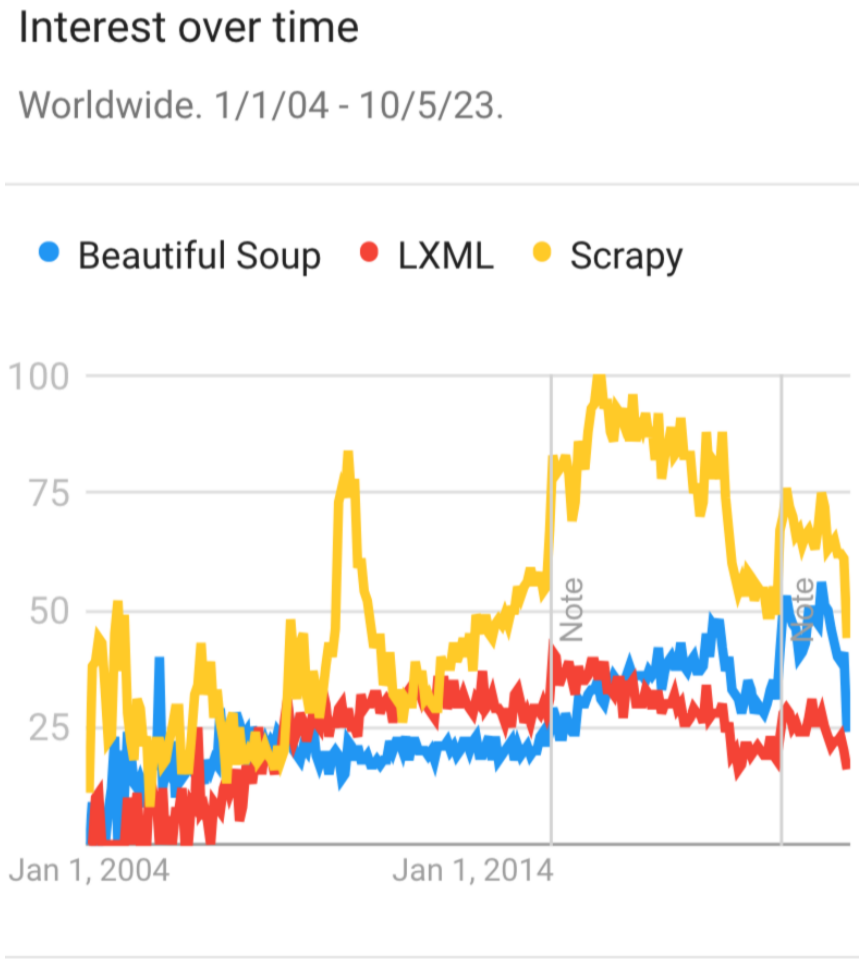
Beautiful Soup & Scrapy

Learning advanced scrapping skills

Form, cookie, Selenium, CAPTCHA

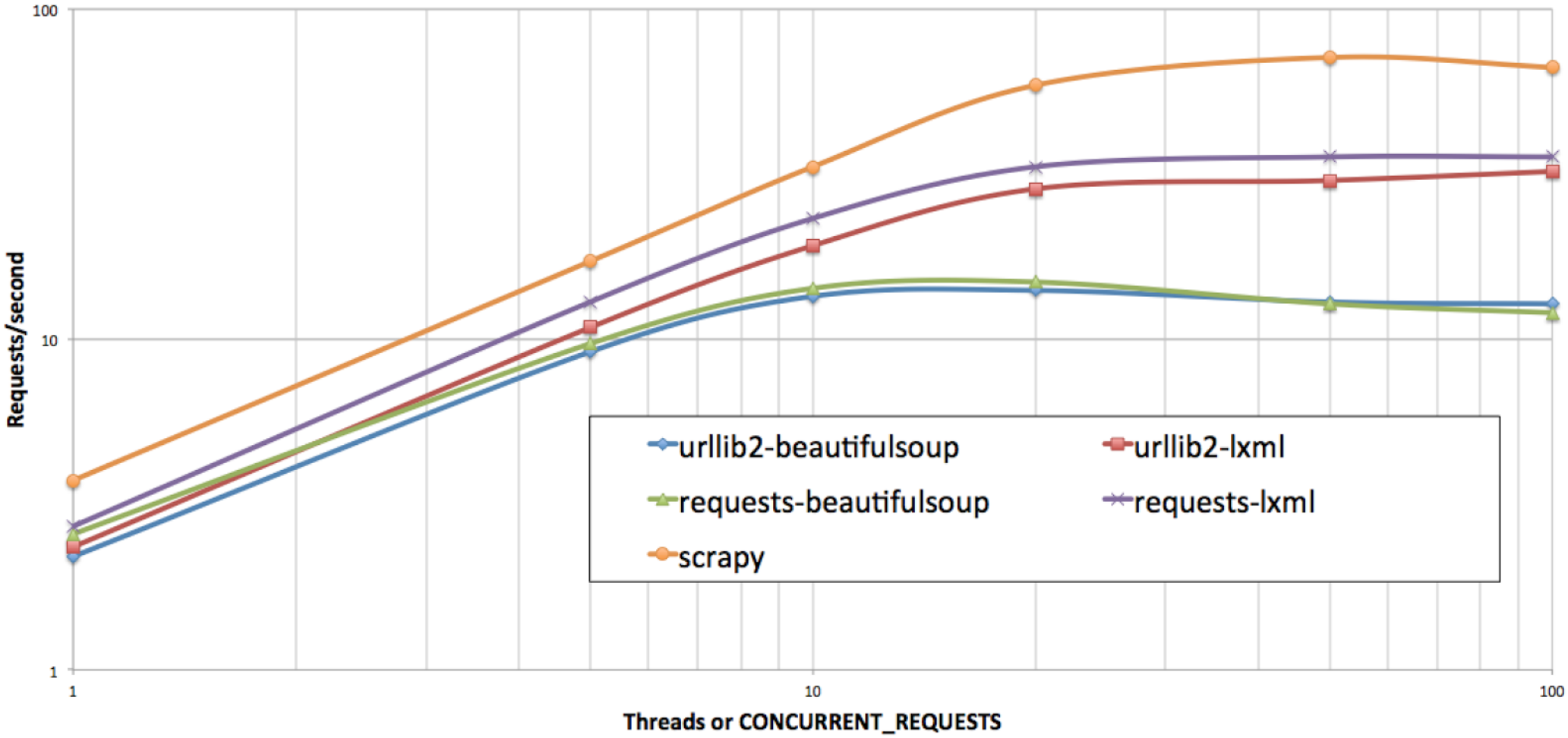
Tools of the trade: Popularity





Tools of the trade: Performance (1/2)

The feature-richer, the slower

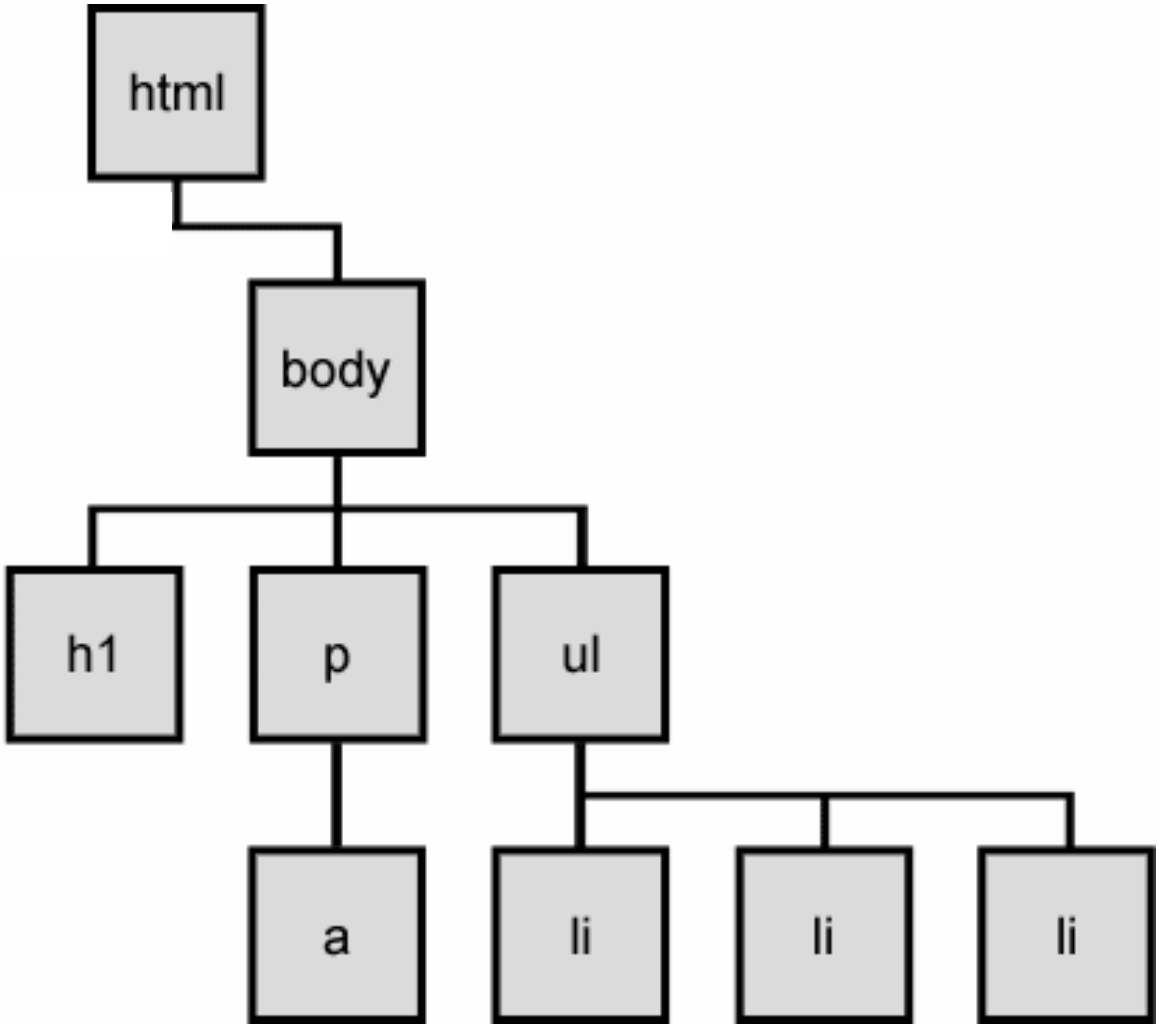


Tools of the trade: Performance (2/2)

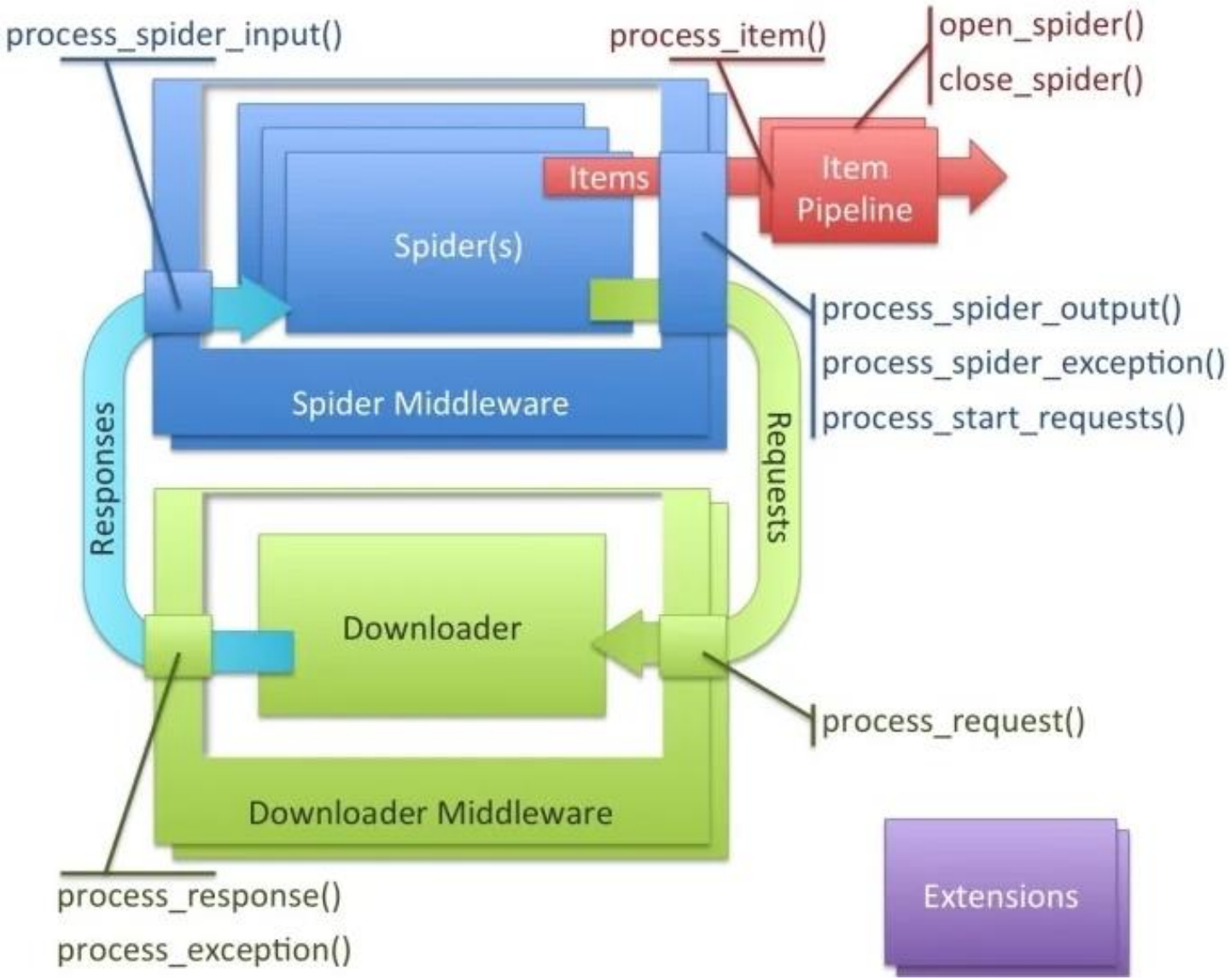
The backend parser can be switched in Beautiful Soup 4

Parser	Typical usage	Advantages	Disadvantages
Python's html.parser	`BeautifulSoup(markup, "html.parser")`	• Batteries included • Decent speed • Lenient (As of Python 3.2)	• Not as fast as lxml, less lenient than html5lib.
lxml's HTML parser	`BeautifulSoup(markup, "lxml")`	• Very fast • Lenient	• External C dependency
lxml's XML parser	`BeautifulSoup(markup, "lxml-xml")` `BeautifulSoup(markup, "xml")`	• Very fast • The only currently supported XML parser	• External C dependency
html5lib	`BeautifulSoup(markup, "html5lib")`	• Extremely lenient • Parses pages the same way a web browser does • Creates valid HTML5	• Very slow • External Python dependency

B-Soup: A HTML parser

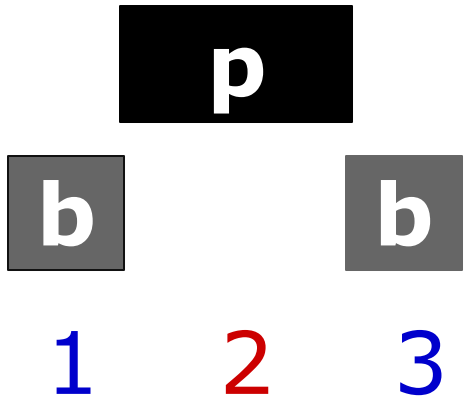


Scrapy: A heavy/feature-rich scrapper



Beautiful Soup 101 (1/2)

Helping us to navigate in a HTML document

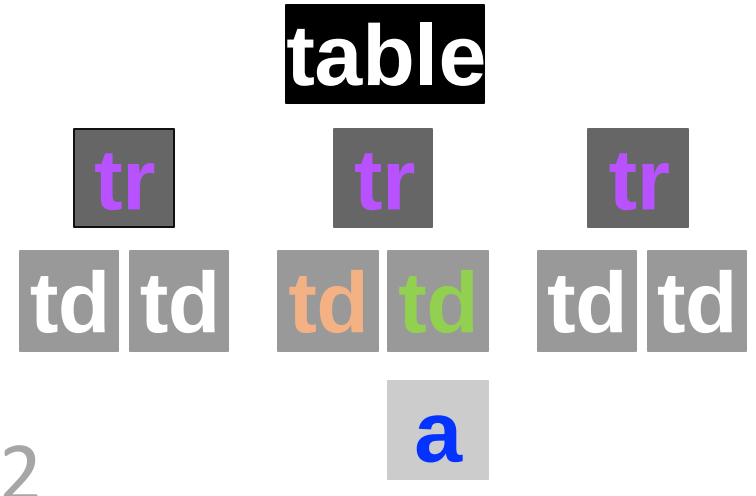



```
from bs4 import BeautifulSoup as BS
data='<p><b>1</b>2<b>3</b></p>'
s=BS(data,'lxml')
print(s) #<html><body><p><b>1</b>2<b>3</b></p></body></html>
print(s.find('2')) #2
print(s.find('p')) #<p><b>1</b>2<b>3</b></p>
print(s.b.string) #1
print(s.find_all('b')) [<b>1</b>, <b>3</b>]
```

Beautiful Soup 101 (2/2)

```
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>'''

from bs4 import BeautifulSoup as BS
s=BS(data)
print(s.a,'\n',s.a.string) # <a href=...>22</a> &  22
print(s.a.parent.previous_sibling) #<td>21</td>
for child in s.tr.children:
    print(child) #<td>11</td> & <td>12</td>
```



Goals for today

Learning web basics

HTML & CSS

Learning basic scrapping skills

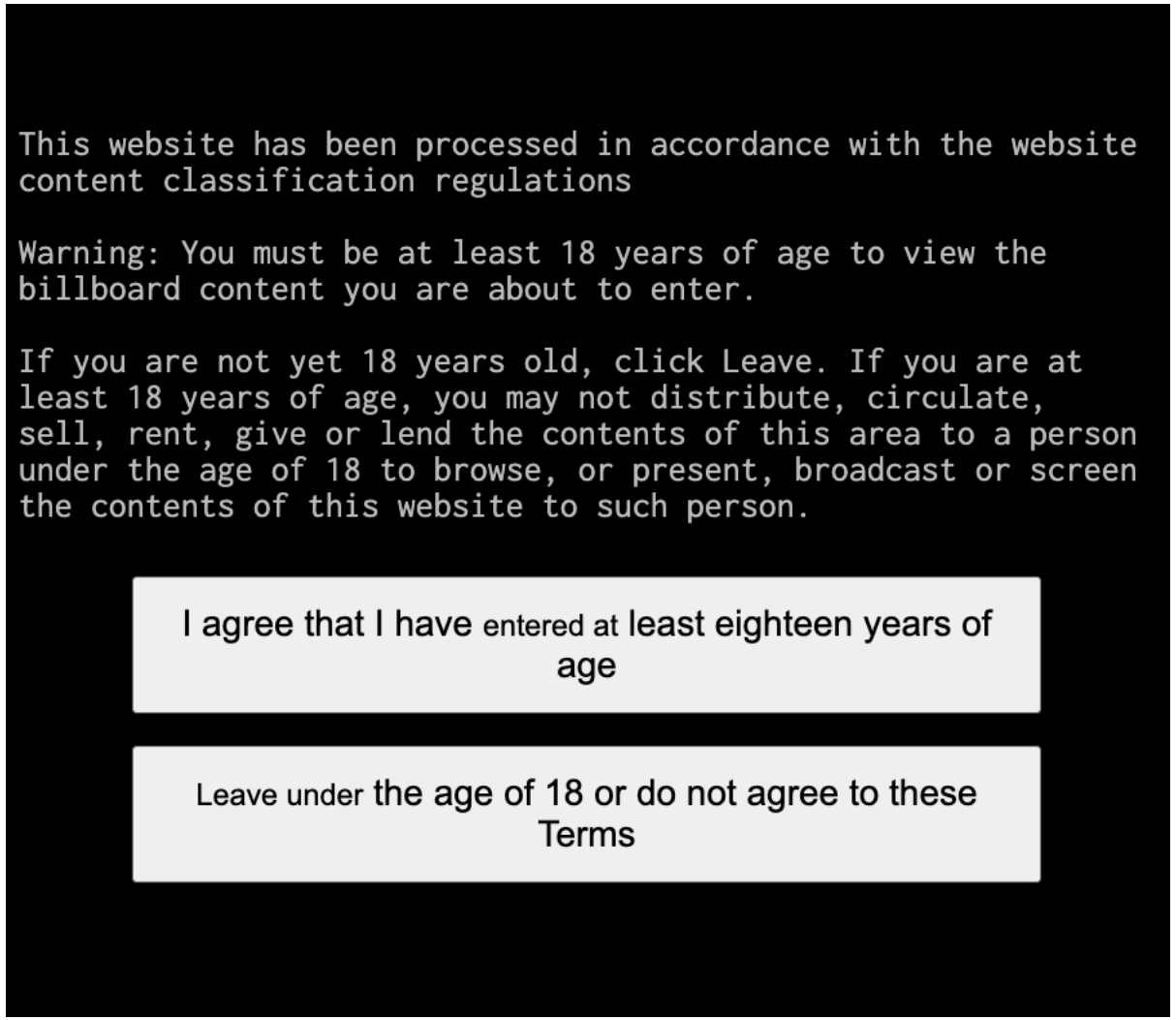
Beautiful Soup & Scrapy

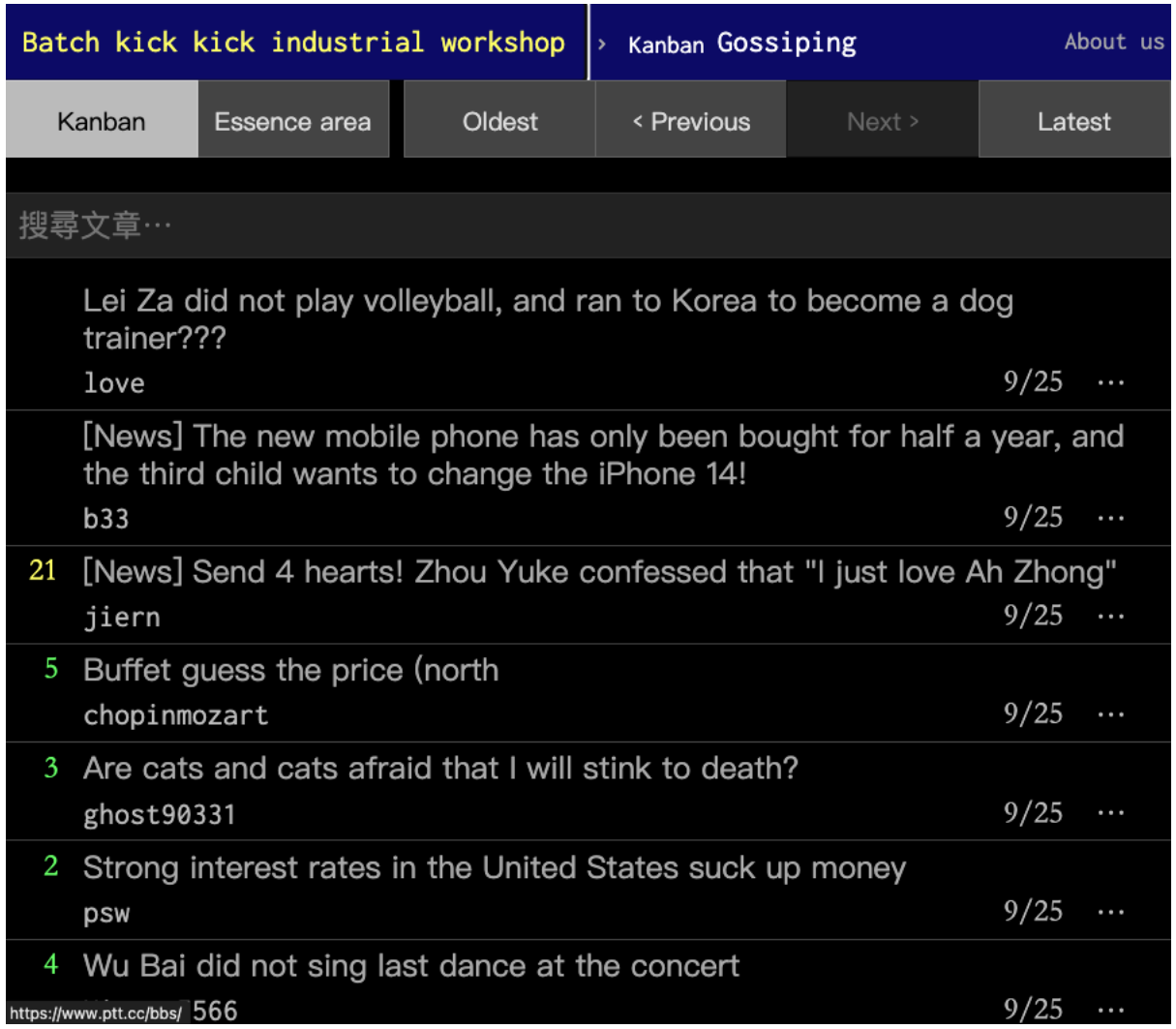
Learning advanced scrapping skills

Form, cookie, Selenium, CAPTCHA

PTT: Age-restricted boards

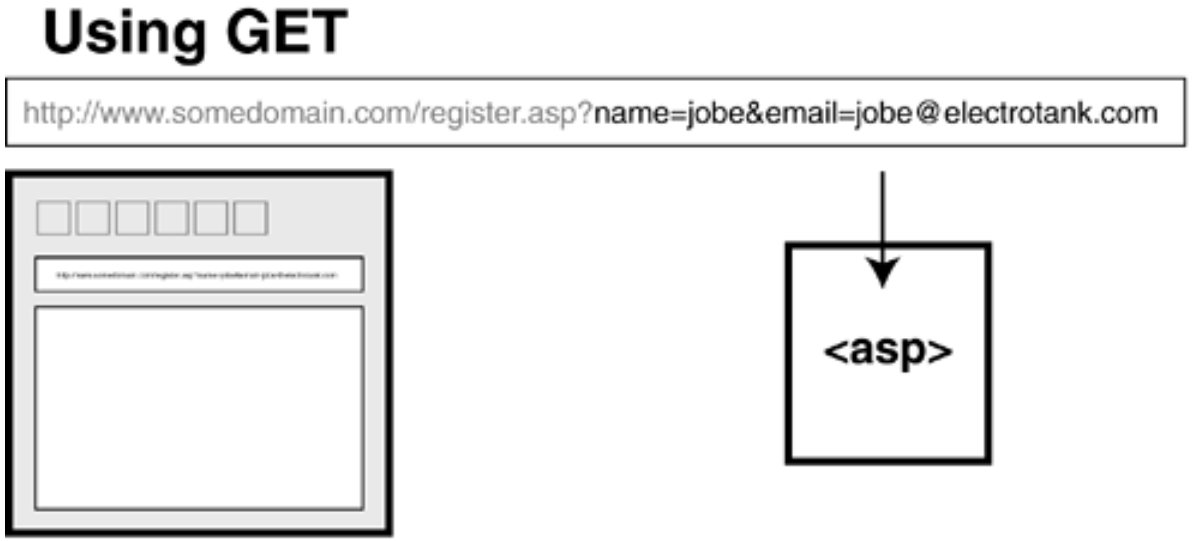
http://ptt.cc/bbs/Gossiping



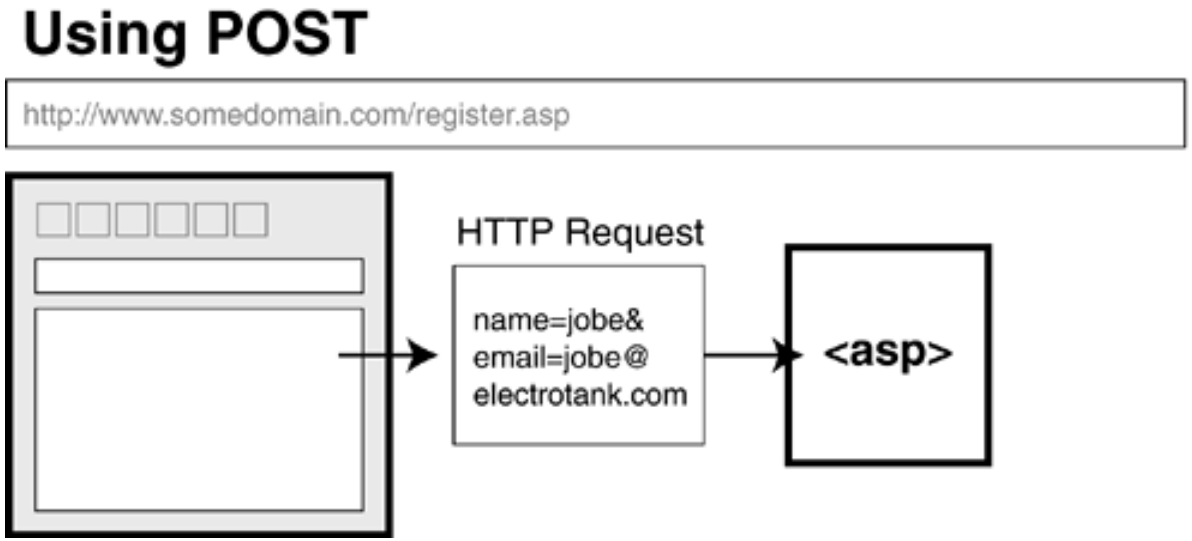


Sending requests via POST vs. GET

More convenient:



More secure:



Requests & Responses

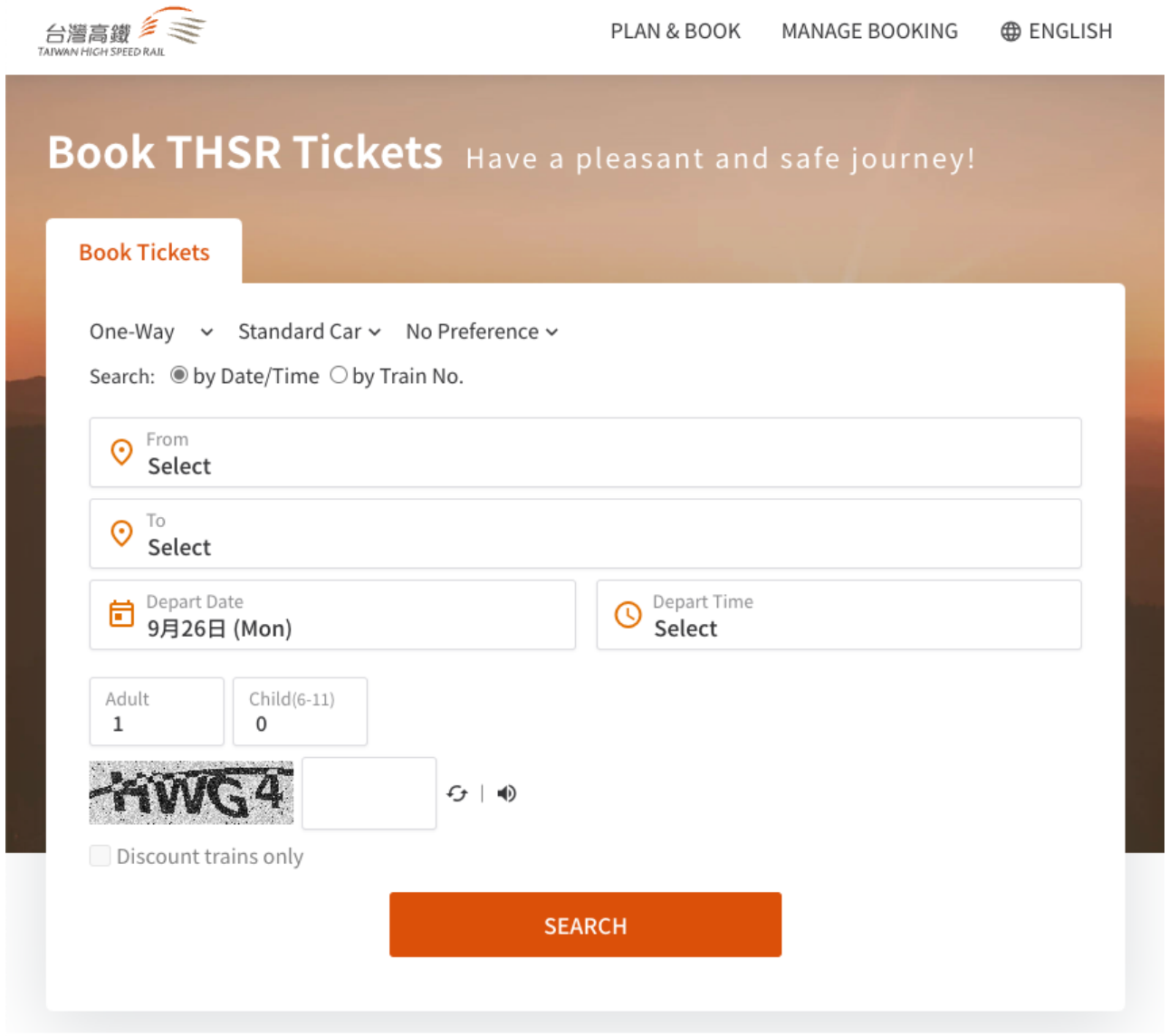
```
<div class="bbs-screen bbs-content center clear">
    <form action="/ask/over18" method="post">
        <input type="hidden" name="from" value="/bbs/Gossiping/index.html">
        <button class="btn-big" type="submit" name="yes" value="yes">我同意，我已年滿十八歲<br><small>進入</small></button>
        <button 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>
```

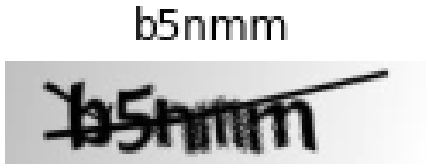
神 Selenium 神

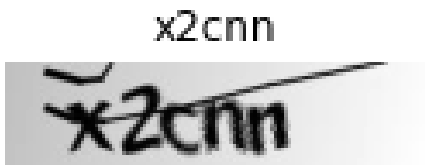
Selenium can automate all browsing actions

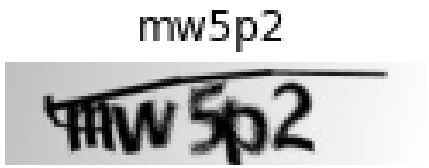
```
from selenium import webdriver
URI='https://www.ptt.cc/bbs/Gossiping/'
driver=webdriver.Chrome() # try Firefox() or Edge()
driver.get(URI)
btn=driver.find_element_by_name('yes')
driver.save_screenshot('before_click.png')
btn.click()
driver.save_screenshot('after_click.png')
print(driver.page_source)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

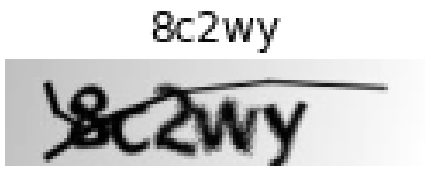
Solving CAPTCHA by OCR

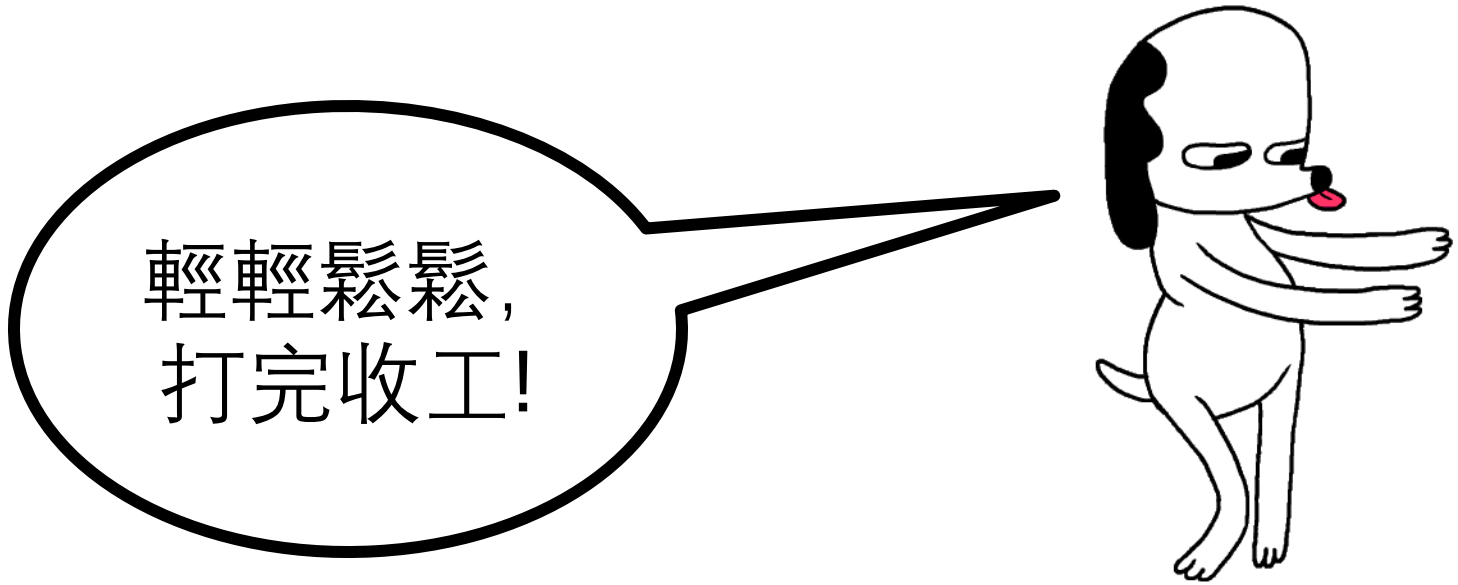











輕輕鬆鬆,
打完收工!

Game Over