Webscraping using BeautifulSoup in Python

by Hui Xiang Chua

Data Science for Social Good

THE UNIVERSITY OF CHICAGO

DATA DOUBLE CONFIRM

GENERAL ASSEMBLY

essence

What I’ll cover today

Basics of Webscraping in Python

&

3 examples

25 countries where people learn fast, think on their feet, and accomplish a lot at work, ranked

Áine Cain, Business Insider US

October 16, 2018

POPULAR ARTICLES

Take a look at this peculiar blue sand dune that NASA found on Mars

The CEO of Cedele avoids bee hoon for breakfast at all costs – here's what else she does to maintain a healthy lifestyle

Celebrity hairstylist Addy Lee is threatening to sue after a plastic surgeon in Seoul allegedly gave him 'Hush Puppies' eyes

Mars will come closer to Earth tonight than it has been in 15 years — here's how to see it

25. Czech Republic

Kevin Lee / Stringer / Getty Images

Score: 73.7

1. Finland

Bruce Bennett/Getty Images

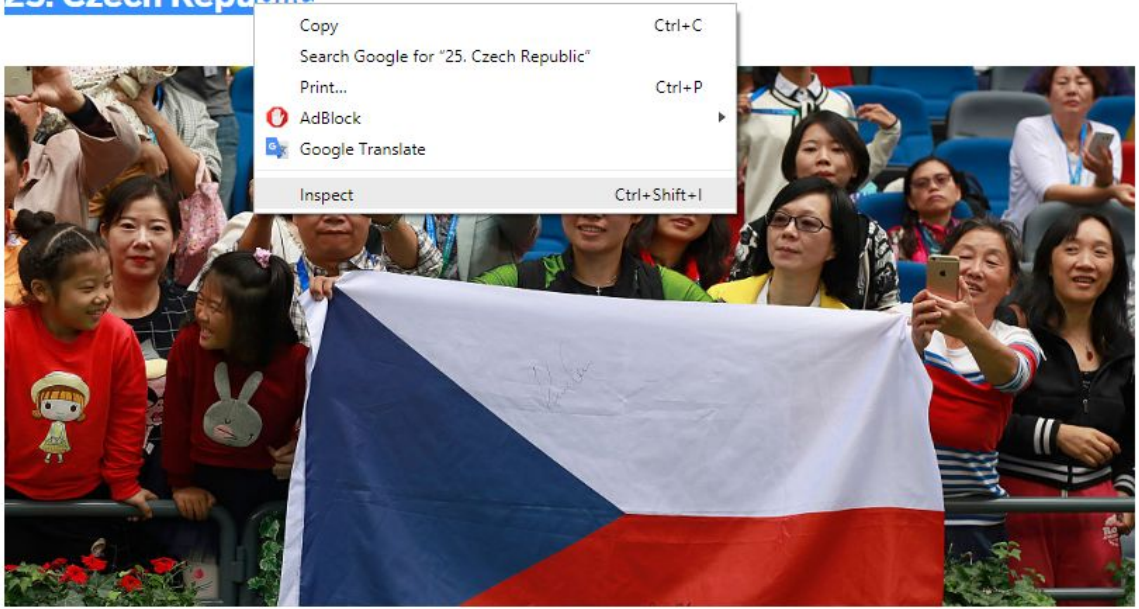
Score: 87.9

	Rank	Country	Score
0	25	Czech Republic	73.7
1	24	Malaysia	74.2
2	23	Latvia	74.5
3	22	Luxembourg	74.7
4	21	Taiwan	75.6
5	20	Singapore	76.0
16	9	Iceland	83.3
17	8	Norway	83.9
18	7	Sweden	84.2
19	6	Netherlands	84.5
20	5	Denmark	84.9
21	4	Germany	85.4
22	3	The United States	86.3
23	2	Switzerland	87.3
24	1	Finland	87.9

Here's a look at the countries with the most skilled employees:

---

25. Czech Republic



ww.businessinsider.sg/highly-skilled-countries-productive-educated-workers-2018-10/?r=UK&IR=T

NEWS TECH INSIDER FINANCE CAREER LIFE VIDEO

ount country-wide
nd institutions.

countries with the most

blic

Images

Elements Console Sources Network 12 6

 Here's a look at the countries with the most skilled employees:
<hr>
<h3>25. Czech Republic</h3> == $0
…
<dl type="image">…</dl>
 Score: 73.7
<hr>

… div div div div div #post-1396192 div #the_bi_content div h3

Styles Event Listeners DOM Breakpoints Properties Accessibility

Filter :hov .cls +

element.style {
}

@media (max-width: 1018px) and (min-width: 768px)
.single .td-post-content div h3 { ?r=UK&IR=T:1255
margin-top: 30px;
}

.single .td-post-content div h3 { ?r=UK&IR=T:805
~~margin-top: 50px;~~
font-size: 25px;
line-height: 32px;
font-family: 'LabGrotesque-Black', Helvetica, Arial, sans-serif;

margin 30
border –
padding –
437.906 × 32
17

Filter Show all

box-sizing border-…
color rgb(1…
display block

Console What's New ×

Highlights from the Chrome 71 update

Hover over a Live Expression to highlight a DOM node
Hover over a result that evaluates to a node to highlight that node in the viewport.

Store DOM nodes as global variables

Kevin Lee / Stringer / Getty Images

Score:

24. M

inessinsider.sg/highly-skilled-countries-productive-educated-workers-2018-10/?r=UK&IR=T

WS TECH INSIDER FINANCE CAREER LIFE VIDEO

Elements Console Sources Network

```
<hr>
<h3>25. Czech Republic</h3>
<p>…</p>
<dl type="image">…</dl>
<p> Score: 73.7</p> == $0
<h3>24. Malaysia</h3>
<p>…</p>
<dl type="image">…</dl>
```

... div div div div div #post-1396192 div #the_bi_conten

Styles Event Listeners DOM Breakpoints Properties Accessi

Filter :hov .cls +

```
element.style {
}
.td-post-content p {                    ?r=UK&IR=T:639
  font-family: Georgia,Times,sans-serif;
  font-size: 1.25rem;
  font-family: TiemposTextWeb-Regular,
     Georgia, Times, serif;
  font-weight: 400 !important;
  color: #111516;
  line-height: 1.65;
  margin: ▸ 2.15rem 0 !important;
}
.td-post-content p style.css?ver=7.8:24634
{
  word-wrap: break-word;
```

margin border paddi 437

Filter

▸ box-sizing
▸ color
▸ display

Console What's New ×

Highlights from the Chrome 71 update

Hover over a Live Expression to highlight a DOM node

Hover over a result that evaluates to a node to highlight that node in the viewport.

Store DOM nodes as global variables

Import necessary libraries.

```
from urllib.request import Request, urlopen
import requests
from bs4 import BeautifulSoup
import csv
```

```
import pandas as pd
import numpy as np
```

Parse (get) the html.

```
site = "https://www.businessinsider.sg/highly-skilled-countries-productive-educat
hdr = {'User-Agent': 'Mozilla/5.0'}
bookpage = requests.get(site)
soup = BeautifulSoup(bookpage.text, "lxml")
```

Extract only ‘h3’ tags within the html.

```
site = "https://www.businessinsider.sg/highly-skilled-countries-productive-educat
hdr = {'User-Agent': 'Mozilla/5.0'}
bookpage = requests.get(site)
soup = BeautifulSoup(bookpage.text, "lxml")
```

```
soup.find_all('h3')
```

```
[<h3>25. Czech Republic</h3>,
 <h3>24. Malaysia</h3>,
 <h3>23. Latvia</h3>,
 <h3>22. Luxembourg</h3>,
 <h3>21. Taiwan</h3>,
 <h3>20. Singapore</h3>,
 <h3>19. Hong Kong SAR</h3>,
 <h3>18. Estonia</h3>,
```

We have to split the rank and country into two data variables.

Write a loop to scrape the rank and country.

```
countries = []
for i in range(0,len(soup.find_all('h3'))):
    c = soup.find_all('h3')[i].get_text()
    rank = c.split('.')[0]
    country = c.split('. ')[1]
    countries.append((rank, country))
```

We have to split the rank and country into two data variables.

Gets the first part of text before “.”

Gets the second part of text after “. ”

```
countries
```

```
[('25', 'Czech Republic'),
 ('24', 'Malaysia'),
 ('23', 'Latvia'),
 ('22', 'Luxembourg'),
 ('21', 'Taiwan'),
 ('20', 'Singapore'),
 ('19', 'Hong Kong SAR'),
 ('18', 'Estonia'),
 ('17', 'Austria'),
```

Convert the array into a dataframe.

```
df = pd.DataFrame(np.array(countries))
```

```
df.columns = ['Rank','Country']
```

```
df
```

	Rank	Country
0	25	Czech Republic
1	24	Malaysia
2	23	Latvia
3	22	Luxembourg
4	21	Taiwan
5	20	Singapore

Now, extract only ‘p’ tags.

```
soup.find_all('p')
```

```
[<p>
 </p>,
 <p>  Which country boasts the most highly skilled workers? </p>,
 <p>  Well, according to the World Economic Forum’s 2018 global  competitivene
ss <a href="http://www3.weforum.org/docs/GCR2018/05FullReport/TheGlobalCompeti
tivenessReport2018.pdf">report</a>, you should book a  ticket to Finland if yo
u want to meet some highly skilled  employees. In the category of skills, the
Scandinavian country’s  workforce received a score of 87.9 out of 100.</p>,
 <p>  According to the report, the skill score measured “the general  level of
skills of the workforce and the quantity and quality of  education.” </p>,
 <p>  Specifically, a high quality education featured “developing  digital lit
eracy, interpersonal skills, and the ability to think  critically and creative
ly.” In addition, the report found  highly-educated societies are more product
ive. This year’s global  competitiveness report took into account country-wide
```

Where are the scores? Scroll down

Scroll down further.

```
soup.find_all('p')
```

```
 <p><img bisrc="https://static.businessinsider.com/image/5bc63aefb5d97419bb61b
064.png" class="bi_image size-medium wp-image-1398163" id="image-5bc63aefb5d97
419bb61b064" src="https://static.businessinsider.sg/2018/10/10/5bc63aefb5d9741
9bb61b064.png"/></p>,
 <p>  Score: 73.7</p>,
 <p><img bisrc="https://static.businessinsider.com/image/5bc63ae3dee789198c29f
212.png" class="bi_image size-medium wp-image-1398188" id="image-5bc63ae3dee78
9198c29f212" src="https://static.businessinsider.sg/2018/10/10/5bc63ae3dee7891
98c29f212.png"/></p>,
 <p>  Source: 74.2</p>,
 <p><img bisrc="https://static.businessinsider.com/image/5bc63acec1aa1319a47a2
3f4.png" class="bi_image size-medium wp-image-1398187" id="image-5bc63acec1aa1
319a47a23f4" src="https://static.businessinsider.sg/2018/10/10/5bc63acec1aa131
9a47a23f4.png"/></p>,
```

Try to locate the line number where the first score starts. (Trial and error)

```
soup.find_all('p')[11].get_text()
```

```
'  Source: 74.5'
```

```
soup.find_all('p')[9].get_text()
```

```
'  Source: 74.2'
```

```
soup.find_all('p')[7].get_text()
```

```
'  Score: 73.7'
```

First score

- Notice the scores are written every other line.
- Also, we can see some typos. (“Source” instead of “Score”)

Write a loop to extract every other line starting from line 7 of ‘p’ tag.

```
scores = []
for i in range(0,25):
    sc = soup.find_all('p')[7+2*i].get_text()
    score = sc.split(': ')[1]
    scores.append(score)
```

```
df2 = pd.DataFrame(np.array(scores))
df2.columns = ['Score']
```

```
df2
```

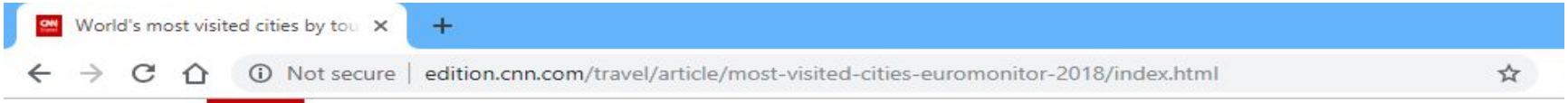
	Score
0	73.7
1	74.2
2	74.5

Merge the two dataframes together with a left join.

```
df.join(df2, how="left")
```

	Rank	Country	Score
0	25	Czech Republic	73.7
1	24	Malaysia	74.2
2	23	Latvia	74.5
3	22	Luxembourg	74.7
4	21	Taiwan	75.6
5	20	Singapore	76.0
6	19	Hong Kong SAR	77.4
7	18	Estonia	78.0
8	17	Austria	78.4

Another example –
where Inspect does not work



Related content | 22 of the best places to visit in the United States

Top 20 cities based on 2017 arrivals and 2018 estimates

1. Hong Kong: 27,880,300 arrivals (2017) / 29,827,200 arrivals (2018)
2. Bangkok, Thailand: 22,453,900 arrivals (2017) / 23,688,800 arrivals (2018)
3. London, England: 19,827,800 arrivals (2017) / 20,715,900 arrivals (2018)
4. Singapore: 17,618,800 arrivals (2017) / 18,551,200 arrivals (2018)
5. Macau: 17,337,200 arrivals (2017) / 18,931,400 arrivals (2018)
6. Paris, France: 15,834,200 arrivals (2017) / 16,863,500 arrivals (2018)
7. Dubai, United Arab Emirates: 15,790,000 arrivals (2017) / 16,658,500 arrivals (2018)
8. New York City, USA: 13,100,000 arrivals (2017) / 13,500,000 arrivals (2018)
9. Kuala Lumpur, Malaysia: 12,843,500 arrivals (2017) / 13,434,000 arrivals (2018)
10. Shenzhen, China: 12,075,100 arrivals (2017) / 12,437,300 arrivals (2018)
11. Phuket, Thailand: 11,613,100 arrivals (2017) / 11,945,500 arrivals (2018)
12. Istanbul, Turkey: 10,730,300 arrivals (2017) / 12,121,100 arrivals (2018)

```
df
```

	rank	city	country	arrivals_2017	arrivals_2018
0	1	Hong Kong	Hong Kong	27880300	29827200
1	2	Bangkok	Thailand	22453900	23688800
2	3	London	England	19827800	20715900
3	4	Singapore	Singapore	17618800	18551200
4	5	Macau	Macau	17337200	18931400
5	6	Paris	France	15834200	16863500
6	7	Dubai	United Arab Emirates	15790000	16658500
7	8	New York City	USA	13100000	13500000
8	9	Kuala Lumpur	Malaysia	12843500	13434000
9	10	Shenzhen	China	12075100	12437300
10	11	Phuket	Thailand	11613100	11945500
11	12	Istanbul	Turkey	10730300	12121100
12	13	Delhi	India	10157000	12505300

⋮

Print to view html.

```
site = "http://edition.cnn.com/travel/article/most-visited-cities-euromonitor-201
hdr = {'User-Agent': 'Mozilla/5.0'}
bookpage = requests.get(site)
soup = BeautifulSoup(bookpage.text, "lxml")
```

```
print(soup.prettify)
```

```
Object.defineProperty(window,"Raven",{value:e,writable:!0,enumerable:!0,config
urable:!0}),e.config(n,t).install(),i.forEach(function(n){var t=n.error,i=n.ex
tra;return e.captureException(t,{extra:i})})},enumerable:!0,configurable:!0}),
window.addEventListener("error",a),window.addEventListener("load",function(){v
ar n=document.createElement("script");n.crossOrigin="anonymous",n.async=!0,n.s
rc=e,document.head.appendChild(n)})}('//cdn.ravenjs.com/3.17.0/raven.min.js',
 'https://0d07ad66266b4248ab931abdcb668326@sentry.io/206797', {"environmen
t":"production","release":"1","ignoreErrors":["404 (Not Found)"],"ignoreUrls":
["//z.cdn.turner.com/ads/adfuel/ais/cnn-ais.min.js","//z.cdn.turner.com/ads/ad
fuel/adfuel-1.2.4.min.js","//tag.bounceexchange.com/340/i.js","//static.chartb
```

Scroll to locate tag and class for interested data to be scraped.

```
soup.find_all('div', class_="Paragraph__component")
```

```
[<div class="Paragraph__component"><cite class="Paragraph__cite">(CNN) – </cit
e><span>Bright lights, big cities. </span></div>,
 <div class="Paragraph__component"><span>The lure of urban centers has lost no
ne of its appeal in 2018, with tourists expected to make around 1.4 billion tr
ips to cities around the world this year.  </span></div>,
 <div class="Paragraph__component"><span>UK-based market research company Euro
monitor International has just released its <a href="https://go.euromonitor.co
m/white-paper-travel-2018-100-cities" target="_blank">Top 100 City Destination
s 2018</a> report, an annual ranking of the world's most popular cities by int
ernational tourism arrivals.</span></div>,
 <div class="Paragraph__component"><span>Once again, <a href="https://www.cnn.
com/travel/destinations/hong-kong" target="_blank">Hong Kong</a> has come out
on top. Close to 30 million tourists are expected to <a href="https://www.cnn.
com/travel" target="_blank">travel</a> to the region before the year is out --
and more than 50% of them will be from the Chinese mainland. </span></div>,
 <div class="Paragraph__component"><span><h3>Travel is on the rise</h3></span>
```

Again, trial and error to locate first line of code to be scraped.

```
soup.find_all('div', class_="Paragraph__component")[23]
```

```
<div class="Paragraph__component"><span>1. Hong Kong: 27,880,300 arrivals (201
7) / 29,827,200 arrivals (2018)</span></div>
```

```
soup.find_all('div', class_="Paragraph__component")[23].get_text()
```

```
'1. Hong Kong: 27,880,300 arrivals (2017) / 29,827,200 arrivals (2018)'
```

```
state = soup.find_all('div', class_="Paragraph__component")[23].get_text()
rank = state.split('.')[0]
rank
```

```
'1'
```

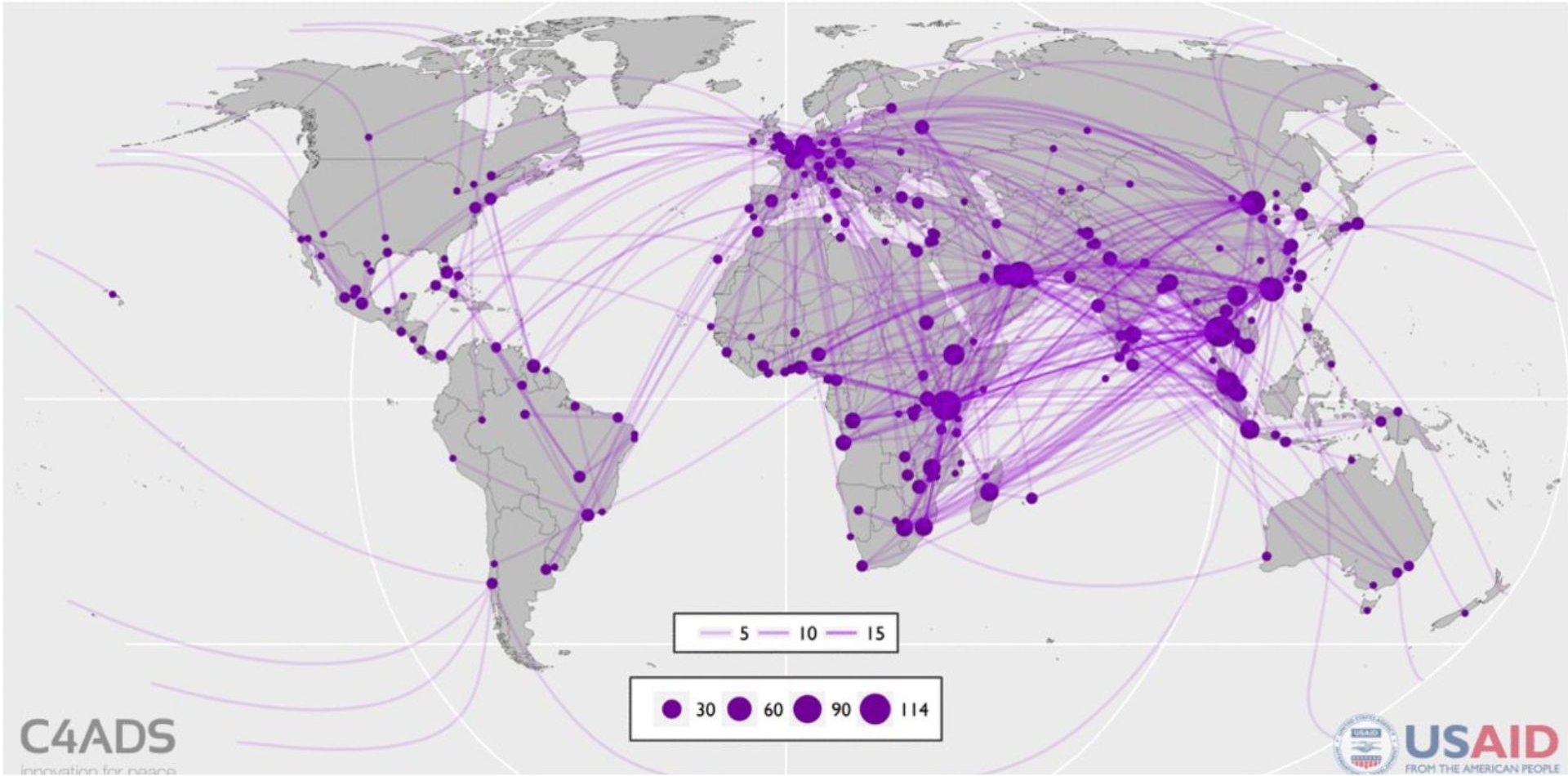
```
city = state.split('. ')[1].split(':')[0]
city
```

Example 1: Webscraping e-commerce platforms

Identifying illegal wildlife trading

Our project partner

Total Trafficking Routes



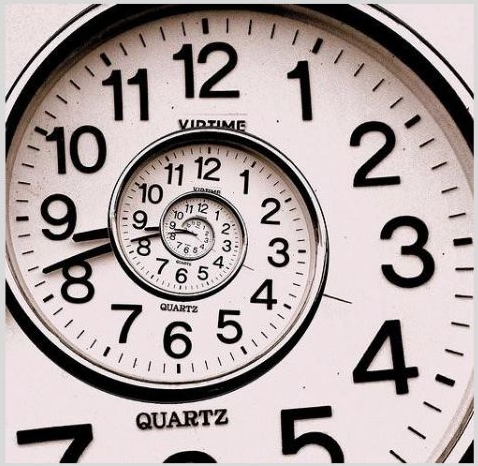
Singapore

Top 10

Major hubs for the illegal wildlife trade

> Budding Data Scientists Hackathon

Trades happen on Carousell



ACRES has to go through the listings, wasting time

Scraping data

```
for i in range(0,n):
    try:
        name = soup.find_all('h4', id="productCardTitle")[i].get_text().strip()
    except:
        name = "NA"
    try:
        username = soup.find_all('h3')[i].get_text().strip()
    except:
        username = "NA"
    try:
        info = soup.find_all('dd')
        price = info[3*i].text.strip()
    except:
        price = "NA"
    try:
        brief = info[3*i+1].text.strip()
    except:
        brief = "NA"
    try:
        time =  soup.find_all('time').get_text().strip()
    except:
        time = "NA"
    try:
        rawlink = soup.find_all('a',class_="G-e")[2*i].get('href')
        link = 'https://sg.carousell.com' + rawlink
    except:
        link = "NA"
    data.append((name, username, price, brief, time, link))
```

Information scraped from each listing:

1. Name of product
2. Username/ID of seller
3. Price
4. Brief description provided by seller
5. Time
6. Link to listing (for easy reference for staff)

Example 2: Webscraping directories

Singapore Government Directory

Directory

Search by Name, Organisation, Job Title, Telephone No. or Email

Search

Listen

The Singapore Government Directory is an online information service to facilitate communication between members of the public and the public service. It includes a listing of ministries, statutory boards, organs of state

Statutory Boards

- ACCOUNTING AND CORPORATE REGULATORY AUTHORITY (ACRA)
- AGENCY FOR SCIENCE, TECHNOLOGY AND RESEARCH (A*STAR)
- AGRI-FOOD & VETERINARY AUTHORITY OF SINGAPORE (AVA)
- BOARD OF ARCHITECTS (BOA)
- BUILDING AND CONSTRUCTION AUTHORITY (BCA)
- CASINO REGULATORY AUTHORITY OF SINGAPORE (CRA)
- CENTRAL PROVIDENT FUND BOARD (CPFB)
- CIVIL AVIATION AUTHORITY OF SINGAPORE (CAAS)
- CIVIL SERVICE COLLEGE (CSC)
- COMPETITION AND CONSUMER COMMISSION OF SINGAPORE (CCCS)
- COUNCIL FOR ESTATE AGENCIES (CEA)
- DEFENCE SCIENCE AND TECHNOLOGY AGENCY (DSTA)

Directory

Search by Name, Organisation, Job Title, Telephone No. or Email

Search

MINISTRY OF FINANCE
ACCOUNTING AND CORPORATE REGULATORY AUTHORITY

Address :

10 Anson Road
#05-01/15 International Plaza
Singapore 079903
CSMailbox:
ACRA GVT-Accounting and Corporate Regulatory Authority/ACRA/SINGOV
Feedback Website: https://www.acra.gov.sg/feedback

View agency's location map

Agency for Science, Technology and Research
SINGAPORE

MINISTRY OF TRADE AND INDUSTRY
AGENCY FOR SCIENCE, TECHNOLOGY AND RESEARCH

Address :

1 Fusionopolis Way #20-10 Connexis North Singapore 138632

View agency's location map

Tel : 68266111

Fax : 67771711

http://www.a-star.edu.sg
contact@a-star.edu.sg

Too many clicks needed

Webscraping

Dashboarding

```
from urllib.request import Request, urlopen
import requests
from bs4 import BeautifulSoup
import csv
import shutil
import re
import time
import pandas as pd
import numpy as np
```

```
['name','address','tel','fax','website','email']
```

In [3]:

```
site = "https://www.gov.sg/sgdi/statutory-boards"
hdr = {'User-Agent': 'Mozilla/5.0'}
bookpage = requests.get(site)
soup = BeautifulSoup(bookpage.text, "html.parser")
titles = soup.find_all('li')
title_db = []
url_db = []
data = []

for i in range(45,108):
    name = titles[i].text.strip()
    url = titles[i].a['href']
    title_db.append(name)
    url_db.append(url)

for i in range(0,len(title_db)):
#for i in range(0,3):
    name = title_db[i]
    nextsite = "https://www.gov.sg"+url_db[i]
    hdr = {'User-Agent': 'Mozilla/5.0'}
    onebookpage = requests.get(nextsite)
    onesoup = BeautifulSoup(onebookpage.text, "html.parser")
    #print(onesoup.prettify())
    try:
        address = onesoup.find('p', class_ = "street-address").text.strip()
    except:
        address = "NA"
    try:
        tel = onesoup.find('p', class_ = "tel-info").get_text().strip()
    except:
        tel = "NA"
    try:
```

	name	address	tel	fax	website	email
0	ACCOUNTING AND CORPORATE REGULATORY AUTHORITY ...	10 Anson Road \r\n#05-01/15 International Plaz...	62486028 (ACRA Helpdesk)	NA	http://www.acra.gov.sg	http://ww (Enquiry '
1	AGENCY FOR SCIENCE, TECHNOLOGY AND RESEARCH (A...	1 Fusionopolis Way\r\n#20-10 Connexis North\r\...	68266111	67771711	http://www.a-star.edu.sg	contact@
2	AGRI-FOOD & VETERINARY AUTHORITY OF SINGAPORE ...	52, Jurong Gateway Road #14-01\r\nSingapore 60...	68052992 Lines For Public Enquiries/Feedback:	63341831	http://www.ava.gov.sgwww.ava.gov.sg/contactus	www.ava.
3	BOARD OF ARCHITECTS (BOA)	5 Maxwell Road\r\n#01-03, Tower Block, MND Com...	62225295	62224452	http://www.boa.gov.sg	boarch@
4	BUILDING AND CONSTRUCTION AUTHORITY (BCA)	52 Jurong Gateway Road\r\n#11-01 Singapore 608550	1800-3425222	63344287	http://www.bca.gov.sgbca_enquiry@bca.gov.sg	bca_enqu

```
    try:
        logourl = onesoup.find_all('img')[-1]['src']
        logoname = onesoup.find_all('img')[-1]['alt']
        r = requests.get("https://www.gov.sg"+logourl,
         stream=True, headers={'User-agent': 'Mozilla/5.0'})
        if r.status_code == 200:
            with open(logoname+logourl[-4:], 'wb') as f:
                r.raw.decode_content = True
                shutil.copyfileobj(r.raw, f)
    except:
        pass
```

acra_logo.png

ASTAR.jpg

ava.gif

BCA.gif


BOA.gif

CAAS.jpg

CCCS.jpg


CPFB.png

CRA.jpg

EDB.jpg

EMA.gif

HDB.gif

HLB.gif

HPB.jpg

HSA.jpg

IMDA.jpg

IPOS.png

iras.gif

ISEAS.jpg

ite.gif

JTC.png

LTA.gif

MCCY.png

MCI-logo.png

MEWR-Logo.jpg

MFAlogo.png

MHA.gif

MINDEF.jpg

MINLAW-1.jpg

MND.gif

MOE.gif

MOF.jpg

MOH.gif

MOM.gif

MOT.jpg

MPA.png

MSF.jpg

MTI.gif


MUIS.jpg

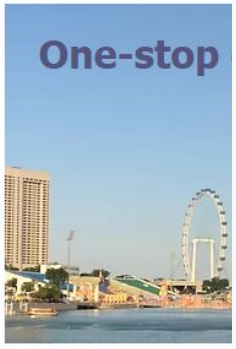
NAC.jpg

NCSS.gif

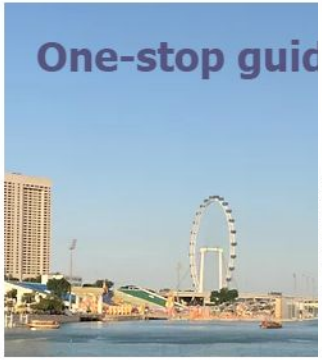
NEA.png

NHB.jpg

NLB.gif



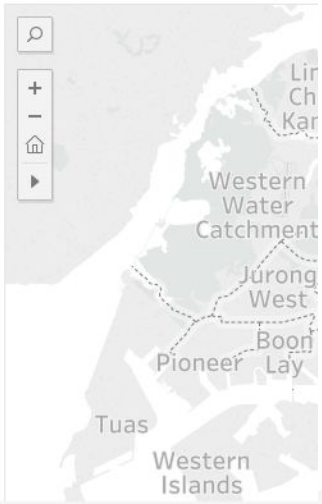
Select Organisation(s) of interes
All

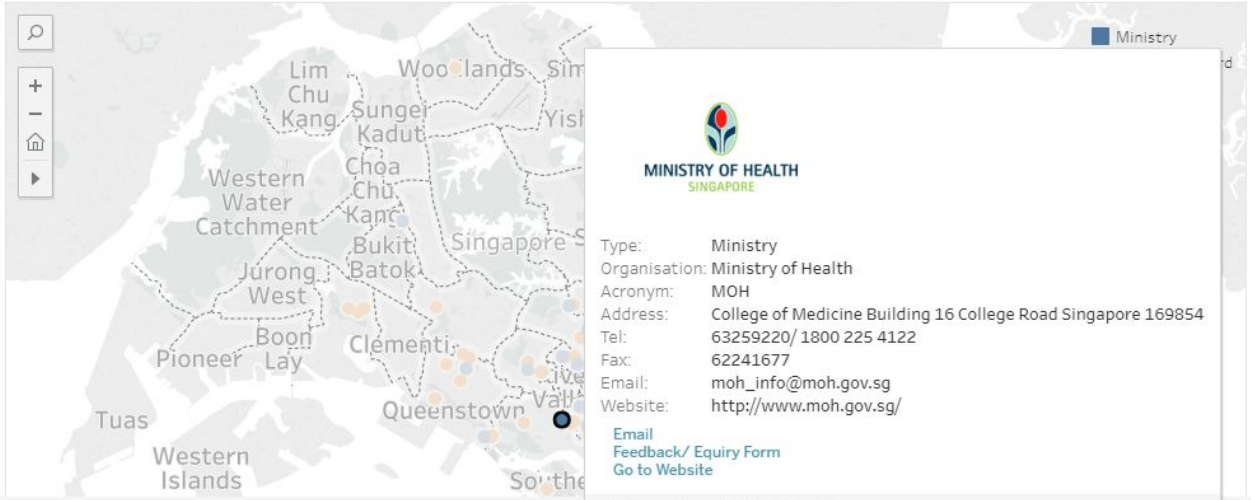


Select Organisation(s) of interest
(All)

Select Organisation(s) of interest
(All)

Filter to see Statutory Boards under a particular Ministry
(All)





MINISTRY OF HEALTH
SINGAPORE

Type:	Ministry
Organisation:	Ministry of Health
Acronym:	MOH
Address:	College of Medicine Building 16 College Road Singapore 169854
Tel:	63259220/ 1800 225 4122
Fax:	62241677
Email:	moh_info@moh.gov.sg
Website:	http://www.moh.gov.sg/

Email
Feedback/ Equiry Form
Go to Website

✓ Keep Only ✕ Exclude

Example 3: Webscraping daily updates

Updates

Date	Title
25 Apr 2020	46 More Cases Discharged; 618 New Cases of COVID-19 Infection Confirmed
25 Apr 2020	National Flag may now be Displayed with Immediate Effect till End of National Day Celebrations Period - Ministry of Culture, Community and Youth (MCCY)
25 Apr 2020	618 New Cases of COVID-19 Infection
24 Apr 2020	38 More Cases Discharged 897 New Cases of COVID-19 Infection Confirmed
24 Apr 2020	Inter-agency Advisory on Supporting Mental Well-being of Workers under COVID-19 work arrangements - Ministry of Manpower (MOM), Ministry of Social and Family Development (MSF), Agency for Integrated Care (AIC), Institute of Mental Health (IMH) and National Council of Social Services (NCSS)

This notebook explores the Number of updates on COVID-19 local situation (Singapore) on Ministry of Health's website since January.

Data is first scraped off the website using BeautifulSoup and then Plotly is used to build an interactive visualization. Some static charts are also created using Seaborn. The data contains updates between 2 Jan 2020 and 25 March 2020.

In [1]:
```
from urllib.request import Request, urlopen
import requests
from bs4 import BeautifulSoup
import csv
```

In [2]:
```
import pandas as pd
import numpy as np
```

In [3]:
```
import re
```

In [4]:
```
site = "https://www.moh.gov.sg/covid-19/past-updates"
hdr = {'User-Agent': 'Mozilla/5.0'}
bookpage = requests.get(site)
soup = BeautifulSoup(bookpage.text, "html.parser")
#print(soup.prettify())
```

In [5]:
```
soup.find_all('span')
```

Out[5]:
```
[<span class="sgds-icon sgds-icon-sg-crest"></span>,
 <span>A Singapore Government Agency Website</span>,
 <span class="sr-only">Toggle navigation</span>,
 <span class="sgds-icon sgds-icon-menu"></span>,
 <span class="sgds-icon sgds-icon-search"></span>,
 <span class="caret"></span>,
 <span class="caret"></span>,
 <span class="caret"></span>,
 <span class="sgds-icon sgds-icon-search"></span>,
 <span class="sgds-icon sgds-icon-search"></span>,
 <span class="last">26 Mar 2020</span>,
 <span style="font-family: Arial; font-size: 16px;">Find past updates on the COVID-19 (Coronavirus Disease
2019) situation in Singapore below. <em>For the latest updates, please click </em><strong><a href="/covid-
19" target="_blank"><em>here</em></a></strong>.</span>,
 <span style="font-size: 16px;"><strong><span style="font-family: Arial;"> Updates</span></strong></span>,
 <span style="font-family: Arial;"> Updates</span>,
 <span style="font-size: 16px; font-family: Arial;"><strong>Date</strong><br/></span>,
 <span style="font-family: Arial; font-size: 16px;"><strong>Title</strong><br/></span>,
 <span style="font-family: Arial; font-size: 16px;">25 Mar 2020</span>,
 <span style="font-family: Arial; ><span style="font-size: 16px; ><a class="" href="https://www.ica.gov.s
g/careers/news-and-publications/media-releases/media-release/travellers-arriving-in-singapore-will-receive
-advance-notification-of-stay-home-notice-requirements" target="_blank" title="">Travellers Arriving In Si
ngapore Will Receive Advance Notification Of Stay-Home Notice Requirements</a> - Immigration &amp;amp; Checkpo
ints Authority (ICA) </span></span>,
 <span style="font-size: 16px;"><a class="" href="https://www.ica.gov.sg/careers/news-and-publications/med
ia-releases/media-release/travellers-arriving-in-singapore-will-receive-advance-notification-of-stay-home-
notice-requirements" target="_blank" title="">Travellers Arriving In Singapore Will Receive Advance Notifi
cation Of Stay-Home Notice Requirements</a> - Immigration &amp;amp; Checkpoints Authority (ICA) </span>,
 <span style="font-family: Arial; font-size: 16px;">25 Mar 2020</span>,
 <span style="font-family: Arial; font-size: 16px;"><a href="https://www.moh.gov.sg/news-highlights/detail
```

In [15]:
```
chunk = []
for i in range(0,715-16):
    c = soup.find_all('span')[i+16].get_text().replace(u'\xa0', u'')
    chunk.append((c))
```

In [16]:
```
chunk
```

Out[16]:
```
['25 Mar 2020',
 'Travellers Arriving In Singapore Will Receive Advance Notification Of Stay-Home Notice Requirements - Im
migration & Checkpoints Authority (ICA)',
 'Travellers Arriving In Singapore Will Receive Advance Notification Of Stay-Home Notice Requirements - Im
migration & Checkpoints Authority (ICA)',
 '25 Mar 2020',
 'Five More Cases Discharged; 73 News Cases of COVID-19 Infection Confirmed',
 '25 Mar 2020',
 '[Updated] Advisory for Individuals Sharing Residential Spaces with Persons issued Stay-Home Notice',
 '[Updated] Advisory for Individuals Sharing Residential Spaces with Persons issued Stay-Home Notice',
 '25 Mar 2020',
 '[Updated] Health Advisory for Persons issued Stay-Home Notice (SHN)',
 '25 Mar 2020',
 '[Updated]Advisory for Individuals Sharing Residential Spaces with Persons issued Stay-Home Notice (SHN)
(25 Mar 2020)',
 '[Updated]Advisory for Individuals Sharing Residential Spaces with Persons issued Stay-Home Notice (SHN)
(25 Mar 2020)',
 '25 Mar 2020',
 'Measures to Prevent Congregations of Foreign Workers and Foreign Domestic Workers - Ministry of Manpower
(MOM)',
 '25 Mar 2020',
 'Employers Advised To Plan For More Sustainable Housing Options For Their Workers Usually Housed In Malay
sia - Ministry of Manpower (MOM)',
 '25 Mar 2020',
```

Number of updates on COVID-19 local situation on Ministry of Health's website

DATA DOUBLE CONFIRM

DATA DIVE DAYS

Process and product of various data science tasks—from data collection, data preparation, data visualization, to basic statistical analysis and modelling. Datasets for practice available.

Selected as Top 100 Data Science Resources for 2018 on MastersInDataScience.com

Search or jump to... / Pull requests Issues Marketplace Explore

Set status

Hui Xiang Chua

hxchua

Edit profile

Overview Repositories 5 Projects 0 Packages 0 Stars 2 Followers 6 Following 0

Popular repositories

Customize your pins

datadoubleconfirm

Simple datasets and notebooks for data visualization, statistical analysis and modelling - with write-ups here: http://projectosyo.wix.com/datadoubleconfirm.

Jupyter Notebook ★ 15 15

Notebook: lta_tweets.ipynb
Description: Python code for working with text and datetime data from LTATrafficNews tweets (between 12-Dec-2019 and 29-Dec-2019)

Notebook: mohcovid.ipynb
Description: Python code for scraping updates on COVID-19 local situation (Singapore) on Ministry of Health's website since January and an interactive visualization created using Plotly (download/run the notebook locally to view it) with some static charts created using Seaborn

Notebook: Monte_Carlo_COVID.ipynb
Description: Python code for Monte Carlo simulations on number of COVID-19 cases in Singapore using 20 days of historical data

Notebook: QOTD.ipynb
Description: Python code for scraping quotes off "Inhale, Exhale and Repeat After Me! 150 Best Quotes About Life" by Parade.com

Notebook: seleniummrt.ipynb
Description: Python code for scraping time and fare information between train stations in Singapore from TransitLink Electronic Guide

Notebook: SingStat API.ipynb
Description: Python code for calling data from SingStat Table Builder/ Singapore Department of Statistics using the API function

Notebook: SingStat API.ipynb
Description: Python code for calling data from SingStat Table Builder/ Singapore Department of Statistics using the API function

Notebook: Statistical tests.ipynb
Description: R code for performing various types of two-sample tests and correlation checks

Notebook: StatutoryBoardSG.ipynb
Description: Python code for scraping addresses/ contact information of statutory boards in Singapore from Singapore Government Directory and automating download of organisation logo images

Notebook: Text Frequency Analysis.ipynb
Description: R code for getting frequency distribution of words in a chunk of text

Notebook: textgen-covid_resilience.ipynb
Description: Python code for text-generating neural network trained using text from Singapore's Budget 2020 - Resilience Budget/ Supplementary Budget Statement with textgenrnn

Notebook: unesco.ipynb
Description: Python code for scraping UNESCO World Heritage countries and sites from UNESCO World Heritage Centre - World Heritage List

Notebook: WiDS.ipynb
Description: R code for predicting gender of survey respondents as part of the WiDS 2018 Datathlon

Notebook: youtube_data_analysis-need_key.ipynb
Description: Python code for getting YouTube data for analysis based on a list of search terms using YouTube Data API v3

Questions?

linkedin.com/in/hui-xiang-chua/

@hxchuaruns

projectosyo.wixsite.com/datadoubleconfirm