# Name: Sandipto Sanyal[¶](#Name:-Sandipto-Sanyal)

# PGID: 12010004[¶](#PGID:-12010004)

# Requirements[¶](#Requirements)

Used Chrome driver for version Chrome version 83.  
For other chrome versions Download the related driver from: <https://chromedriver.chromium.org/downloads>

# Packages required[¶](#Packages-required)

Uncomment and run if not the below packages are installed

In [1]:

#!pip install selenium

#!pip install beautifulsoup4

#!pip install pandas

# Import libraries[¶](#Import-libraries)

In [2]:

import time

from selenium import webdriver

from bs4 import BeautifulSoup

import bs4

import re

import pandas as pd

from selenium.common import exceptions

import warnings

# User configurable variables[¶](#User-configurable-variables)

search\_criteria and pages\_to\_scan are user configurable. They denote the course names and number of pages to scan for each course name.  
To reduce the running time we are keeping number of pages to perform extract data as 25.

In [3]:

search\_criteria = 'Information Technology'

location\_criteria = 'Hyderabad'

pages\_to\_scan = 25

# Launch the chrome browser[¶](#Launch-the-chrome-browser)

In [4]:

options = webdriver.ChromeOptions()

options.add\_argument('--ignore-certificate-errors')

options.add\_argument("--test-type")

locationOfWebdriver = "./chromedriver\_win32/chromedriver.exe"

driver = webdriver.Chrome(locationOfWebdriver)

# Visit Naukri.com[¶](#Visit-Naukri.com)

In [5]:

driver.get('https://www.naukri.com/')

time.sleep(5)

# Search for jobs through the search bar[¶](#Search-for-jobs-through-the-search-bar)

In [6]:

def search\_the\_job(search\_criteria: str,

location\_criteria: str):

'''

Run the website actions like scrolling to load the price tags

'''

def find\_the\_course\_search\_box():

'''

Target the search box using native Javascript and interact with it.

Highlevel selenium API was not able to interact with the searchbox

'''

course\_search\_box = driver.find\_element\_by\_xpath('//input[@name="keyword" \

]')

location\_search\_box = driver.find\_element\_by\_xpath('//input[@name="location" \

]')

# selenium highlevel API was not able to interact with the search box

# so went into javascript level to

js\_actions = "arguments[0].value='{}';arguments[1].value = '{}'".format(search\_criteria, location\_criteria)

driver.execute\_script(js\_actions,

course\_search\_box,

location\_search\_box

)

course\_search\_box.submit()

# Now do a new search for the course in search bar

course\_to\_search = search\_criteria

find\_the\_course\_search\_box()

time.sleep(3)

In [7]:

search\_the\_job(search\_criteria, location\_criteria)

# Go to the next page[¶](#Go-to-the-next-page)

Below we define a method to go to the next page

In [8]:

def go\_to\_next\_page():

print('going to next page...')

next\_button = driver.find\_element\_by\_xpath('//a[@class="fright fs14 btn-secondary br2"]')

js\_actions = "arguments[0].click()"

driver.execute\_script(js\_actions,

next\_button,

)

time.sleep(10)

# Loop with all the search criteria and the individual individual whole page to extract course related information[¶](#Loop-with-all-the-search-criteria-and-t)

Go to the automated chrome browser to view how selenium is putting the search criteria one by one

## Clean the skills required tag[¶](#Clean-the-skills-required-tag)

This is to remove the tag elements of the skills required section of each job to join as strings.

In [9]:

def clean\_skills\_required(skillset\_tag\_parent\_tag: bs4.element.Tag):

'''

This method parses the elements under tag: <ul class="tags has-description">

skillset\_tag\_parent\_tag: bs4 element

'''

skills\_tags = skillset\_tag\_parent\_tag.find\_all('li',class\_='fleft fs12 grey-text lh16 dot')

skills = []

for skill\_tag in skills\_tags:

skills.append(skill\_tag.get\_text())

return ','.join(skills)

## Columns we will be extracting[¶](#Columns-we-will-be-extracting)

Since naukri.com provides the following features of a particular job, we are extracting them as it is:  
'search\_criteria', 'job\_title', 'company\_name', 'experience', 'salary', 'location', 'experience\_description', 'skills\_required', 'days\_posted'

In [10]:

data = {'search\_criteria':[],

'job\_title':[],

'company\_name':[],

'experience':[],

'salary':[],

'location':[],

'experience\_description':[],

'skills\_required':[],

'days\_posted':[],

}

for page in range(1,pages\_to\_scan+1):

print('URL scraping: {}'.format(driver.current\_url))

# get the page xml

soup = BeautifulSoup(driver.page\_source, 'lxml')

# list the jobs

courses = soup.find\_all('article',class\_='jobTuple bgWhite br4 mb-8')

print('Number of jobs found on page {}: {}'.format(page, len(courses)))

for course in courses:

try:

data['search\_criteria'].append(search\_criteria)

except:

data['search\_criteria'].append(None)

try:

data['job\_title'].append(course.div.div.a.get\_text());

except:

data['job\_title'].append(None)

try:

data['company\_name'].append(course.find('div',class\_='mt-7 companyInfo subheading lh16').a.get\_text());

except:

data['company\_name'].append(None)

try:

data['experience'].append(course.find('ul',class\_='mt-7').find('li', class\_='fleft grey-text br2 placeHolderLi experience').span.get\_text());

except:

data['experience'].append(None)

try:

data['salary'].append(course.find('ul',class\_='mt-7').find('li', class\_='fleft grey-text br2 placeHolderLi salary').span.get\_text());

except:

data['salary'].append(None)

try:

data['location'].append(course.find('ul',class\_='mt-7').find('li', class\_='fleft grey-text br2 placeHolderLi location').span.get\_text());

except:

data['location'].append(None)

try:

data['experience\_description'].append(course.find('div',class\_='job-description fs12 grey-text').get\_text());

except:

data['experience\_description'].append(None)

try:

skills = clean\_skills\_required(course.find('ul',class\_='tags has-description'))

data['skills\_required'].append(skills);

except:

data['skills\_required'].append(None)

try:

data['days\_posted'].append(course.find('div', class\_='jobTupleFooter mt-20').div.span.get\_text());

except:

data['days\_posted'].append(None)

try:

# visit next page

go\_to\_next\_page()

except:

try:

go\_to\_next\_page()

except exceptions.ElementClickInterceptedException:

try:

go\_to\_next\_page()

except Exception as e:

warnings.warn("Warning........... Couldn\'t navigate to next page: Exception occured: {}".format(e))

pass

print('Search process ended')

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad

Number of jobs found on page 1: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-2

Number of jobs found on page 2: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-3

Number of jobs found on page 3: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-4

Number of jobs found on page 4: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-5

Number of jobs found on page 5: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-6

Number of jobs found on page 6: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-7

Number of jobs found on page 7: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-8

Number of jobs found on page 8: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-9

Number of jobs found on page 9: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-10

Number of jobs found on page 10: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-11

Number of jobs found on page 11: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-12

Number of jobs found on page 12: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-13

Number of jobs found on page 13: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-14

Number of jobs found on page 14: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-15

Number of jobs found on page 15: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-16

Number of jobs found on page 16: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-17

Number of jobs found on page 17: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-18

Number of jobs found on page 18: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-19

Number of jobs found on page 19: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-20

Number of jobs found on page 20: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-21

Number of jobs found on page 21: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-22

Number of jobs found on page 22: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-23

Number of jobs found on page 23: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-24

Number of jobs found on page 24: 20

going to next page...

URL scraping: https://www.naukri.com/information-technology-jobs-in-hyderabad-secunderabad-25

Number of jobs found on page 25: 20

going to next page...

Search process ended

In [11]:

df = pd.DataFrame(data=data)

display(df.head())

df.to\_csv('Hyderabad\_Sandipto\_Sanyal.csv',index=False)

|  | **search\_criteria** | **job\_title** | **company\_name** | **experience** | **salary** | **location** | **experience\_description** | **skills\_required** | **days\_posted** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **0** | Information Technology | Information Technology (.Net) | Cognia Technologies | 2-4 Yrs | Not disclosed | Hyderabad | Information Technology (.Net) B.Tech , BBA , ... | assembly language,Six Sigma,JavaScript,Real Ti... | 30+ Days Ago |
| **1** | Information Technology | GENERAL MANAGER (INFORMATION TECHNOLOGY) | Hyderabad Race Club | 10-15 Yrs | Not disclosed | Hyderabad | # Job Code : JOB3090 Job Description Hyderaba... | INFORMATION TECHNOLOGY,General Manager IT,Debu... | 30+ Days Ago |
| **2** | Information Technology | GENERAL MANAGER (INFORMATION TECHNOLOGY) | Hyderabad Race Club | 10-15 Yrs | Not disclosed | Hyderabad | GENERAL MANAGER (INFORMATION TECHNOLOGY) Job ... | Computer science,management information system... | 30+ Days Ago |
| **3** | Information Technology | Information Technology | TERRA TECHNOLOGY | 1-3 Yrs | Not disclosed | Hyderabad | Knowledge and experience with ticketing tools... | Unix,Service management,global operations,Linu... | 30+ Days Ago |
| **4** | Information Technology | Information Technology | TERRA TECHNOLOGY | 1-3 Yrs | Not disclosed | Hyderabad | •Knowledge and experience with ticketing tool... | Unix,Service management,global operations,Data... | 30+ Days Ago |

# Stop the driver[¶](#Stop-the-driver)

In [12]:

driver.quit()

In [ ]:

# Name: Sandipto Sanyal[¶](#Name:-Sandipto-Sanyal)

# PGID: 12010004[¶](#PGID:-12010004)

# Import the required libraries[¶](#Import-the-required-libraries)

In [1]:

import requests

import json

import pandas as pd

# Hit the URL[¶](#Hit-the-URL)

As per the documentation the URL has to be in the below format to extract out the necessary details

In [2]:

url = ('http://newsapi.org/v2/everything?'

'q=Corona Virus&'

'language=en&'

'from=2020-08-01&'

'sortBy=publishedAt&'

'pageSize=100&'

'apiKey=98af07d5f1054ba9b1589cf5da1b6f59'

)

response = requests.get(url)

# Dump the response to text file[¶](#Dump-the-response-to-text-file)

In [3]:

with open('temp.txt','w') as f:

json.dump(response.json(), f)

# Create the dataframe from the JSON[¶](#Create-the-dataframe-from-the-JSON)

## Declare the dataframe schema[¶](#Declare-the-dataframe-schema)

In [4]:

data = {'source\_id':[],

'source\_name':[],

'author':[],

'title':[],

'description':[],

'content':[]

}

response\_json = response.json()

## Loop all the articles returned in our response to load them[¶](#Loop-all-the-articles-returned-in-our-r)

In [5]:

for article in response\_json['articles']:

data['source\_id'].append(article.get('source').get('id'))

data['source\_name'].append(article.get('source').get('name'))

data['author'].append(article.get('author'))

data['title'].append(article.get('title'))

data['description'].append(article.get('description'))

data['content'].append(article.get('content'))

In [6]:

df = pd.DataFrame(data=data)

df

Out[6]:

|  | **source\_id** | **source\_name** | **author** | **title** | **description** | **content** |
| --- | --- | --- | --- | --- | --- | --- |
| **0** | None | Belloflostsouls.net | Thomas aka Goatboy | Goatboy’s Warhammer 40k : FAQtastic Loose Ends | Goatboy here with some unanswered questions fr... | Goatboy here with some unanswered questions fr... |
| **1** | None | Daily Mail | By Levi Parsons For Daily Mail Australia | Teachers are the most likely profession to see... | Teachers in coronavirus-hit Victoria have lodg... | Teachers in coronavirus-hit Victoria have lodg... |
| **2** | the-hindu | The Hindu | Andhra Pradesh Bureau | Andhra Pradesh steps up its fight against coro... | Here is a look at what the government of Andhr... | Here is a look at what the government of Andhr... |
| **3** | None | Moneycontrol | None | When COVID-19 vaccines become available, will ... | Since health policies cover hospitalisation, o... | The world is waiting with bated breath for a v... |
| **4** | None | All India Radio | None | News Services Division, All India Radio News -... | News Services Division, All India Radio News  ... | THE HEADLINES\r\n:-\r\n<ul><li> Prime Minister... |
| **...** | ... | ... | ... | ... | ... | ... |
| **95** | None | Livejournal.com | toxic\_illusion | ONTD Roundup | For Thursday, August 6, 2020:<ul><li>SuperM an... | i'm 5 months sober from alcohol today!!!about ... |
| **96** | spiegel-online | Spiegel Online | Alexander Chernyshev, Wu Dandan, Georg Fahrion... | Corona: Cutting Corners in the Race for a Vaccine | Russia, China and India are racing to find a c... | When Alexander Ginzburg injected himself with ... |
| **97** | None | Slashdot.org | Anonymous Coward | Personal Care gifts for Staff | Even though we are right in the middle of the ... | The Fine Print: The following comments are own... |
| **98** | None | Slashdot.org | Anonymous Coward | A Favorite Corporate Gifting Ideas in 2020 | Even though we are right in the middle of the ... | The Fine Print: The following comments are own... |
| **99** | None | Annvoskamp.com | Ann Voskamp | How to Show Up in Beirut as the Hands of Jesus... | When it feels like the end of the world — we c... | It feels like it’s a physical manifestation of... |

100 rows × 6 columns

# Export the dataframe[¶](#Export-the-dataframe)

In [7]:

df.to\_csv('News\_api\_Sandipto\_Sanyal.csv', index=False)