INFO 6210 - Assignment 3 SQL to NoSQL using MONGODB

Importing all the needed libraries

In [1]:
```
import tweepy
from tweepy import OAuthHandler
import pandas as pd
import json
import os
```

Connecting to twitter API

In [2]:
```
consumer_key = 'XZridMCbX1nP9j2ndIfhXLf0g'
consumer_secret = 'SdRqWfev6Y8lrDjJQ41UM3n0XgnrcgxCfXjLEVNdxOVVjXf6z8'
access_token = '3369008419-dtEhXWLRtxFf7FEFrg8ohwhi38sC2JwIDtHrbzV'
access_secret = '3KtKiLMaaucJMjpv5laRutkV3VAM6Cq3MvCtqc2U6x00R'

auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_secret)
api = tweepy.API(auth, wait_on_rate_limit=True, wait_on_rate_limit_notify=Tru

#Printing the tweepy.api object to validate if we are connected or not
print(api)
if (not api):
    print ("Problem connecting to API")
```

```
<tweepy.api.API object at 0x00000218ECC9E0C8>
```

In [3]:

```python
tweets = []
count = 1
for tweet in tweepy.Cursor(api.search, q="#coronavirus", count=300).items(1):
    print(json.dumps(tweet._json, indent=3))
```

```
{
   "created_at": "Thu Apr 09 03:55:30 +0000 2020",
   "id": 1248097179799093249,
   "id_str": "1248097179799093249",
   "text": "RT @va_shiva: The Deep State #FakeScience Establishment led b
y Fauci cares ZERO about the Immune Health of the American People. To the
m eve\u2026",
   "truncated": false,
   "entities": {
      "hashtags": [
         {
            "text": "FakeScience",
            "indices": [
               29,
               41
            ]
         }
      ],
      "symbols": [],
```

In [ ]:

In [4]:

```
tweets = []
count = 1
for tweet in tweepy.Cursor(api.search, q="#coronavirus", count=450, since='20

    try:
        data = [tweet.id, tweet.user._json['screen_name'], tweet.user._json['
                tweet.entities['hashtags'], tweet.user._json['statuses_count'
                tweet.user._json['followers_count'],
                tweet.user._json['friends_count'], tweet.created_at, tweet.en
        data = tuple(data)
        tweets.append(data)

    except tweepy.TweepError as e:
        print(e.reason)
        continue

    except StopIteration:
        break


df = pd.DataFrame(tweets, columns = ['ID', 'screen_name', 'name', 'text','has
                                     'statuses_count','followers_count','frie
                                     'created_at', 'urls'])

df.to_csv('/Users/srush/Desktop/coronavirus.csv', index=False)
print(df)
```

```
                       ID      screen_name  \
0     1248097183368466433       niiif__we
1     1248097182915260421   FromaHKPeople1
2     1248097182579929098    aquigonzalezp
3     1248097182114353153       rushan614
4     1248097182038675457    gadakh_kishor
...                   ...              ...
2995  1248095669879025664    LizzyBelleFox
2996  1248095669841203200             n69n
2997  1248095669815898112  ValerieLynneCl2
2998  1248095668234838024         madamyez
2999  1248095667664412672    always_angiee

                                      name  \
0                             كلنا مسوؤل 🙏
1                               M. Anthony
2                        AQUILES GONZALEZ P
3                             Rushan Abbas
4                            Kishor Gadakh
...                                    ...
2995                               Liberty
2996  Bake The Hall In The Candle Of Her Brain
2997                Valerie C 🎀 I believe you
2998                     and the livin's easy
2999                                   Angiee

                                                text  \
0     RT @akhbar: "وبحسب وكالة الأنباء السعودية "واس...
1     RT @SenRickScott: This isn’t about politics, @...
2     RT @AlbertoBernalLe: Gracias a Dios se mantien...
```

```
3     RT @Uyghur_American: 🆘Trouble paying bills? La...
4     RT @jogalshailaja: #Kolhapur - \nकोल्हापुरातील...
...                                                 ...
2995  #Bavari #Coronavirus #Covid #CommunistChina Vi...
2996  RT @stevesilberman: Rest in peace Leilani Jord...
2997  RT @bugwannostra: Go back as far as September ...
2998  This happened on March 12👈\n\nIt's now April 8...
2999  Tried beating the #Quarantine blues by making ...

                                               hashtags  statuses_count  \
0                                                    []               4
1                                                    []           25075
2     [{'text': 'Colombia', 'indices': [71, 80]}, {'...            6342
3      [{'text': 'coronavirus', 'indices': [123, 135]}]            4612
4     [{'text': 'Kolhapur', 'indices': [19, 28]}, {'...            1475
...                                                 ...             ...
2995  [{'text': 'Bavari', 'indices': [0, 7]}, {'text...           24530
2996   [{'text': 'coronavirus', 'indices': [120, 132]}]          269269
2997                                                 []          152902
2998                                                 []          227027
2999      [{'text': 'Quarantine', 'indices': [18, 29]}]           11730

      followers_count  friends_count          created_at  \
0                   0             16 2020-04-09 03:55:31
1                3279           4781 2020-04-09 03:55:31
2                 136           1074 2020-04-09 03:55:31
3                5612            499 2020-04-09 03:55:31
4                  62            507 2020-04-09 03:55:31
...               ...            ...                 ...
2995             3819           2822 2020-04-09 03:49:30
2996             2076           4994 2020-04-09 03:49:30
2997             1687           1704 2020-04-09 03:49:30
2998             1636           4732 2020-04-09 03:49:30
2999              268           1325 2020-04-09 03:49:30

                                                   urls
0                                                    []
1                                                    []
2                                                    []
3                                                    []
4                                                    []
...                                                 ...
2995  [{'url': 'https://t.co/v2w297ob5Q', 'expanded_...
2996                                                 []
2997                                                 []
2998  [{'url': 'https://t.co/nykvo0xmEY', 'expanded_...
2999  [{'url': 'https://t.co/tEUghdFv0M', 'expanded_...

[3000 rows x 10 columns]
```

In [5]:

```
import pymongo
from pymongo import MongoClient
```

In [6]:

```
# CONNECT TO DATABASE
connection = pymongo.MongoClient("localhost", 27017)

# CREATE DATABASE
database = connection['twitterDB']

# CREATE COLLECTION
collection = database['coronavirus']

print("Database connected")
```

```
Database connected
```

In [7]:

```
def load_csv(csv):
    p=os.path.join("data/", csv)
    print (p)
    data=pd.read_csv(p, encoding = "ISO-8859-1", engine='python')
    return data
tweets_csv=load_csv('C:/Users/srush/Desktop/coronavirus.csv')
tweets_csv.head()
```

```
C:/Users/srush/Desktop/coronavirus.csv
```

Out[7]:

	ID	screen_name	name	text	hashtags	statu
0	1248097183368466433	niiif__we	Ù□Ù□Ù□Ø§ Ù□ Ø³Ù□Ø¤Ù□ ð□□□ð□□¼	RT @akhbar: Ù□Ø¨ØØ³Ø¨ Ù□Ù□Ø§Ù□Ø© Ø§Ù□Ø£Ù□Ø¨Ø§...	[]	
1	1248097182915260421	FromaHKPeople1	M. Anthony	RT @SenRickScott: This isnâ□□t about politics,...	[]	
2	1248097182579929098	aquigonzalezp	AQUILES GONZALEZ P	RT @AlbertoBernalLe: Gracias a Dios se mantien...	[{'text': 'Colombia', 'indices': [71, 80]}, {'...	
3	1248097182114353153	rushan614	Rushan Abbas	RT @Uyghur_American: ð□□□Trouble paying bills?...	[{'text': 'coronavirus', 'indices': [123, 135]}]	
4	1248097182038675457	gadakh_kishor	Kishor Gadakh	RT @jogalshailaja: #Kolhapur - \nà¤□à¥□à¤²à¥□à...	[{'text': 'Kolhapur', 'indices': [19, 28]}, {'...	

In [8]:

```
#Converting csv dataset jso format

new =json.loads(tweets_csv.to_json(orient='records'))
new[0]
```

Out[8]:

```
{'ID': 1248097183368466433,
 'screen_name': 'niiif__we',
 'name': 'Ù\x83Ù\x84Ù\x86Ø§ Ù\x85Ø³Ù\x88Ø¤Ù\x84 ð\x9f\x99\x8fð\x9f\x8f¼',
 'text': 'RT @akhbar: Ù\x88Ø¨Ø\xadØ³Ø¨ Ù\x88Ù\x83Ø§Ù\x84Ø© Ø§Ù\x84Ø£Ù\x86Ø¨
Ø§Ø¡ Ø§Ù\x84Ø³Ø¹Ù\x88Ø¯Ù\x8aØ© "Ù\x88Ø§Ø³"Ø\x8c Ù\x81Ø¥Ù\x86 Ø§Ù\x84Ù\x82Ø
±Ø§Ø± Ø§Ù\x84Ù\x85Ù\x84Ù\x83Ù\x8a Ù\x8aÙ\x82Ø¶Ù\x8a Ø¨Ø§Ù\x84Ø¥Ù\x81Ø±Ø§Ø¬
Ø§Ù\x84Ù\x85Ø¤Ù\x82Ø ªØ\x8c Ø¨Ø´Ù\x83Ù\x84 Ù\x81Ù\x88Ø±Ù\x8aØ\x8c Ø¹Ù\x85Ù\x
86 Ø\xadØ¨Ø³ Ø ªÙ\x86Ù\x81Ù\x8aØ°Ø§ Ù\x84Ø ªÙ\x84Ù\x83 Ø§Ù\x84Ø£Ø\xadÙ\x83Ø§Ù
\x85 Ù\x88Ø§Ù\x84Ø£Ù\x88Ø§Ù\x85Ø±.\n\n#â\x80¦',
 'hashtags': '[]',
 'statuses_count': 4,
 'followers_count': 0,
 'friends_count': 16,
 'created_at': '2020-04-09 03:55:31',
 'urls': '[]'}
```

In [9]:

```
#Adding records to the collection

collection.insert(new)
```

```
C:\Users\srush\Anaconda3\lib\site-packages\ipykernel_launcher.py:3: Depre
cationWarning: insert is deprecated. Use insert_one or insert_many instea
d.
  This is separate from the ipykernel package so we can avoid doing impor
ts until
```

In [10]:

```
#Validating if the connection is added to the MongoDB Database

connection.list_database_names()
```

Out[10]:

```
['admin', 'config', 'local', 'twitterDB']
```

In [11]:

```
#Printing first five documents to check the content of the collection

for d in collection.find()[:].limit(5):
    print(d)
```

```
{'_id': ObjectId('5e8e9c9ed3d98fc8dd11a409'), 'ID': 1248096997501841408, 's
creen_name': 'JustJoolz01', 'name': 'Jules', 'text': "RT @bugwannostra: Go
back as far as September 2013, you can see the efforts to scuttle our medic
al care in Australia. Isn't it ironic the sâ\x80¦", 'hashtags': '[]', 'stat
uses_count': 4678, 'followers_count': 80, 'friends_count': 155, 'created_a
t': '2020-04-09 03:54:47', 'urls': '[]'}
{'_id': ObjectId('5e8e9c9ed3d98fc8dd11a40a'), 'ID': 1248096996751114241, 's
creen_name': 'GHQ2000', 'name': 'ã\x81¤ã\x81\x90ã\x81¿', 'text': "RT @MOFA_
Taiwan: These days we've been busy packing face masks for countries hard hi
t by #Coronavirus. Allies &amp; friends, #Taiwan is coming!â\x80¦", 'hashta
gs': "[{'text': 'Coronavirus', 'indices': [89, 101]}, {'text': 'Taiwan', 'i
ndices': [125, 132]}]", 'statuses_count': 168828, 'followers_count': 605,
'friends_count': 434, 'created_at': '2020-04-09 03:54:47', 'urls': '[]'}
{'_id': ObjectId('5e8e9c9ed3d98fc8dd11a40b'), 'ID': 1248096996508008450, 's
creen_name': 'arte_prima', 'name': 'Ivan Santos', 'text': 'RT @fatourgente:
Brasil volta a bater recorde de mortes por #coronavÃ\xadrus em 24 horas: fo
ram 133, totalizando 800 vÃ\xadtimas', 'hashtags': "[{'text': 'coronavÃ\xad
rus', 'indices': [60, 72]}]", 'statuses_count': 479791, 'followers_count':
12995, 'friends_count': 11987, 'created_at': '2020-04-09 03:54:47', 'urls':
'[]'}
{'_id': ObjectId('5e8e9c9ed3d98fc8dd11a40c'), 'ID': 1248096996180795392, 's
creen_name': 'Melonpieri1', 'name': 'Melonpieri', 'text': '#Starwars #Paras
ite #Memes #memesdaily #COVID19 #coronavirus https://t.co/bpYlk7uK4x', (htt
ps://t.co/bpYlk7uK4x',) 'hashtags': "[{'text': 'Starwars', 'indices': [0,
 9]}, {'text': 'Parasite', 'indices': [10, 19]}, {'text': 'Memes', 'indice
s': [20, 26]}, {'text': 'memesdaily', 'indices': [27, 38]}, {'text': 'COVID
19', 'indices': [39, 47]}, {'text': 'coronavirus', 'indices': [48, 60]}]",
 'statuses_count': 60, 'followers_count': 15, 'friends_count': 121, 'create
d_at': '2020-04-09 03:54:47', 'urls': '[]'}
{'_id': ObjectId('5e8e9c9ed3d98fc8dd11a40d'), 'ID': 1248096993949384704, 's
creen_name': 'ArletteBG', 'name': 'Arlette**', 'text': 'RT @Milenio: Estima
n que 107 millones de mujeres en #AmÃ©ricaLatina queden en pobreza tras #co
ronavirus\nhttps://t.co/lvZ079cotc https://t.coâ\x80¦', (https://t.coâ\x80
¦',) 'hashtags': "[{'text': 'AmÃ©ricaLatina', 'indices': [52, 66]}, {'tex
t': 'coronavirus', 'indices': [90, 102]}]", 'statuses_count': 5995, 'follow
ers_count': 147, 'friends_count': 465, 'created_at': '2020-04-09 03:54:46',
'urls': "[{'url': 'https://t.co/lvZ079cotc', 'expanded_url': 'https://mile.
io/2K2Xw1v', 'display_url': 'mile.io/2K2Xw1v', 'indices': [103, 126]}]"}
```

In [12]:

```
#As data is converted from csv to json, its format is string in the beginning
#The "created_at" column has date and time information together as one
#Using below datetime function to convert "created_at" into a right format.

def to_datetime(datestring):
    dt = datetime.strptime(datestring.strip(), '%Y-%m-%d %H:%M:%S')
    #dt = dt.utcnow()
    return dt
```

NLTK is a leading platform for building Python programs to work with human language data.

Natural Language Processing with Python provides a practical introduction to programming for language processing. Written by the creators of NLTK, it guides the reader through the fundamentals of writing Python programs, working with corpora, categorizing text, analyzing linguistic structure, and more. The online version of the book has been been updated for Python 3 and NLTK 3.

Learn More: https://www.nltk.org/index.html (https://www.nltk.org/index.html)

In [13]:

```
import nltk

nltk.download('stopwords')
from nltk.corpus import stopwords
stop_words_list = list(stopwords.words('english'))
stop_words={}
for tag in stop_words_list:
    stop_words[tag]=0
print (stop_words.keys())
```

```
dict_keys(['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'yo
u', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yo
urselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'h
erself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their', 'theirs',
'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 't
hese', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'h
ave', 'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'th
e', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'while', 'of', 'a
t', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through',
'during', 'before', 'after', 'above', 'below', 'to', 'from', 'up', 'down',
'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further', 'then', 'onc
e', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both',
'each', 'few', 'more', 'most', 'other', 'some', 'such', 'no', 'nor', 'not',
'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can', 'wil
l', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm',
'o', 're', 've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't", 'did
n', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'have
n', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn', "must
n't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn',
"wasn't", 'weren', "weren't", 'won', "won't", 'wouldn', "wouldn't"])

[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\srush\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
```

re — Regular expression operations

This module provides regular expression matching operations similar to those found in Perl.

Both patterns and strings to be searched can be Unicode strings (str) as well as 8-bit strings (bytes). However, Unicode strings and 8-bit strings cannot be mixed: that is, you cannot match a Unicode string with a byte pattern or vice-versa; similarly, when asking for a substitution, the replacement string must be of the same type as both the pattern and the search string.

Learn more: https://docs.python.org/3/library/re.html (https://docs.python.org/3/library/re.html)

In [14]:

```
import re

#This tokenize function extracts words by certain rules.
#For instance blanks or punctuation marks in the text are used to split words
def tokenize(txt):
  txt=re.sub(r'\n', ' ',txt)
  txt=re.compile(r'[\.][ ]+').sub(' ',txt)
  txt=re.compile(r'[\,][ ]+').sub(' ',txt)
  txt=re.compile(r'[_+;=!@$%^&\*\"\?]').sub(' ',txt)
  splitter=re.compile(r'[ ]+')

  # Split the words by non-alpha characters
  words=splitter.split(txt)
  return words

print (tokenize(d['text']))
```

```
['RT', 'Milenio:', 'Estiman', 'que', '107', 'millones', 'de', 'mujeres', 'e
n', '#AmÃ©ricaLatina', 'queden', 'en', 'pobreza', 'tras', '#coronavirus',
'https://t.co/lvZ079cotc', 'https://t.coâ\x80¦']
```

In [15]:

```
def update_urls_tags(url_list,urls,hashtag_list,hashtags,tag_list,tags):
  for url in url_list:
      if url in urls:
        urls[url]=urls[url]+1
      else:
        urls[url]=1
  for tag in tag_list:
      if tag in tags:
        tags[tag]=tags[tag]+1
      else:
        tags[tag]=1
  for hashtag in hashtag_list:
      if hashtag in hashtags:
        hashtags[hashtag]=hashtags[hashtag]+1
      else:
        hashtags[hashtag]=1
  return urls,hashtags,tags
```

In [16]:

```
hashtags={}
urls={}
tags={}

def extract_tags_urls(dct,words,stop):
  i=0
  tags={}
  tokens={}
  urls={}
  size=len(words)
  while i < size:
    ngram = words[i]
    i=i+1
    if len(ngram) < 1: continue
    if len(ngram) > 4:
      if ngram[0:4].lower()=='http':
        if ngram in urls:
          urls[ngram]=urls[ngram]+1
        else:
          urls[ngram]=1
    if ngram[0]=='#':

        #ngram=re.sub(r'\#', '',ngram) <if you want to remove the #>

      tags[ngram]=1
    if ngram.lower() not in stop:
        tokens[ngram]=1
    if ngram in dct:
      tags[ngram]=1
    if i < (size-1):
      ngram = words[i] + ' ' + words[i+1]
      if words[i].lower() not in stop:
        tokens[ngram]=1
      if ngram in dct:
        tags[ngram]=1
    if i < (size-2):
      ngram = words[i] + ' ' + words[i+1] + ' ' + words[i+2]
      if ngram in dct:
        tags[ngram]=1
  return list(tags.keys()),list(urls.keys()),list(tokens.keys())

print (extract_tags_urls(hashtags,(tokenize(d['text'])),stop_words))
```

```
(['#AmÃ©ricaLatina', '#coronavirus'], ['https://t.co/lvZ079cotc', 'https://
t.coâ\x80¦'], ['RT', 'Milenio: Estiman', 'Milenio:', 'Estiman que', 'Estima
n', 'que 107', 'que', '107 millones', '107', 'millones de', 'millones', 'de
mujeres', 'de', 'mujeres en', 'mujeres', 'en #AmÃ©ricaLatina', 'en', '#AmÃ©
ricaLatina queden', '#AmÃ©ricaLatina', 'queden en', 'queden', 'en pobreza',
'pobreza tras', 'pobreza', 'tras #coronavirus', 'tras', '#coronavirus http
s://t.co/lvZ079cotc', (https://t.co/lvZ079cotc',) '#coronavirus', 'https://
t.co/lvZ079cotc https://t.coâ\x80¦', (https://t.coâ\x80¦',) 'https://t.co/l
vZ079cotc', 'https://t.coâ\x80¦'])
```

In [17]:

```
#Following code keeps records of used hashtags and how many times they are us

cnt=0
for tweet in new:
   #
    retweet_count=0
    try:
        retweet_count=int(tweet['hashtags_count'])
    except:
        pass
    tweet_tags,tweet_urls,tweet_ngrams=extract_tags_urls(hashtags,(tokenize(t
    print (tweet_tags)
    urls,hashtags,tags=update_urls_tags(tweet_urls,urls,tweet_tags,hashtags,t
    try:

        #j=tweet_json(tweet['ID'], tweet['screen_name'], tweet['name'],tweet[
              #tweet['statuses_count'], tweet['followers_count'],tweet['frier
              #tweet['created_at'],tweet['urls'],hashtags_count,tweet_tags)

        j=new(tweet['ID'], tweet['screen_name'], tweet['name'],tweet['text'],
              tweet['statuses_count'], tweet['followers_count'],tweet['frienc
              tweet['created_at'],tweet['urls'],hashtags_count,tweet_tags)
        result = collection.insert_one(j)
        cnt+=1
    except:
        pass
print ("%d tweets inserted."%cnt)
```

```
['#â\x80¦']
[]
['#Colombia', '#Suecia', '#RepublicaCheca']
['#coronavirus']
['#Kolhapur', '#coronavirus']
['#coronavirus.â\x80\x9d']
['#Elektra', '#RicardoSalinasPliego', '#ATodaMadre', '#Cuarentena', '#Abo
nosChiquitos,a']
['#FakeScience']
[]
['#ShabEBarat', '#CoronaVirus']
['#uci', '#coronavirus', '#hilo']
['#coronavirusâ\x80¦']
[]
['#Coronavirus']
['#TablighiJamaat', '#coronavirus']
['#coronavirus']
['#Coâ\x80¦']
['#Ramayana', '#Hydroxychloroquine']
```

```
In [18]:  for key, value in hashtags.items():
              print ("%s count %d"%(key, value))
```

```
#nilo count 7
#coronavirusâ□¦ count 31
#Coronavirus count 341
#TablighiJamaat count 22
#Coâ□¦ count 1
#Ramayana count 10
#Hydroxychloroquine count 13
#Covid19 count 25
#ThanksObama count 1
#coâ□¦ count 9
#Coronavâ□¦ count 17
#StayAtHomeAndStaySafe count 9
#Jesucristo count 2
#karnatâ□¦ count 1
#Briones count 3
#DevolucionCompleta count 3
#infowars count 1
#Truth count 1
#Wuhan count 31
#BeatTheVirâ□¦ count 20
```

Q1 - What are the tags associated with a Person, Place or Thing?

```
In [19]:  from collections import Counter
          for user1, count in Counter(hashtags).most_common(10):
                print(user1 + "\t" + str(count))
```

```
#coronavirus    1095
#Coronavirus    341
#COVID19        182
#CoronaVirus    79
#COVIDã□¼19     39
#coronavirusâ□¦ 31
#Wuhan  31
#COâ□¦  30
#Covid19        25
#Taiwan 23
```

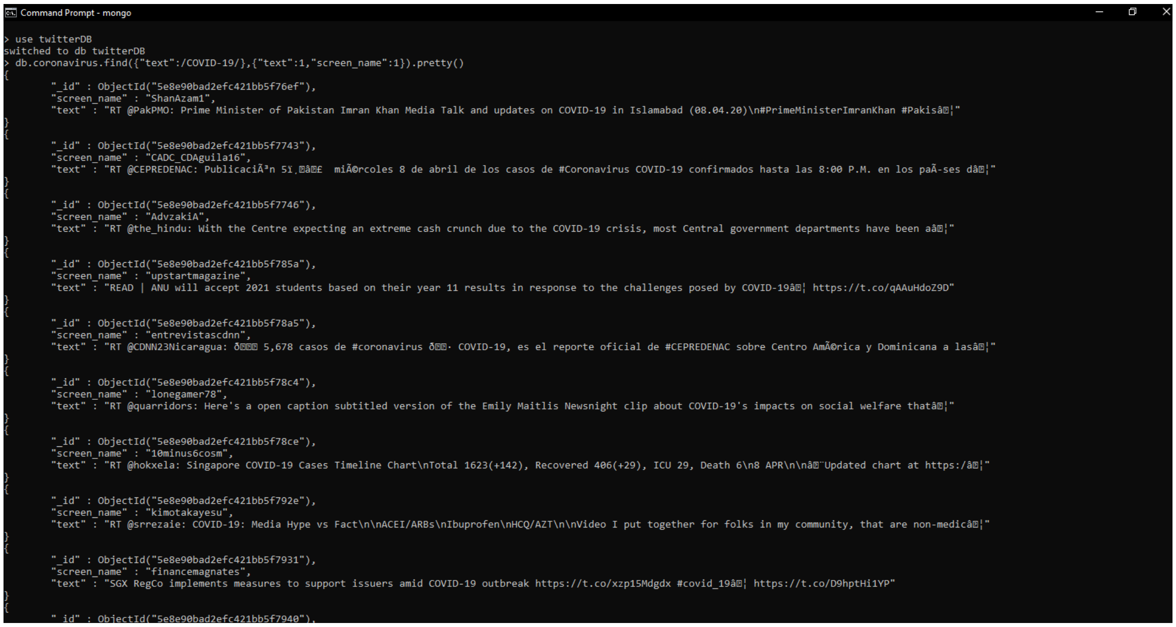
Q2 - What social media users are like other social media users in your domain?

MongoDB Shell Script: db.coronavirus.find({"text":/COVID-19/},{"text":1,"screen_name":1}).pretty()

In [20]:

```
#Output Screensot of the script
from IPython.display import Image
Image("Q2_Answer.png")
```

Out[20]:



```
> use twitterDB
switched to db twitterDB
> db.coronavirus.find({"text":/COVID-19/},{"text":1,"screen_name":1}).pretty()
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f76ef"),
        "screen_name" : "ShanAzam1",
        "text" : "RT @PakPMO: Prime Minister of Pakistan Imran Khan Media Talk and updates on COVID-19 in Islamabad (08.04.20)\n#PrimeMinisterImranKhan #Pakisâ¦"
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7743"),
        "screen_name" : "CADC_CDAguila16",
        "text" : "RT @CEPREDENAC: PublicaciÃ³n 5ï¸â£  miÃ©rcoles 8 de abril de los casos de #Coronavirus COVID-19 confirmados hasta las 8:00 P.M. en los paÃses dâ¦"
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7746"),
        "screen_name" : "AdvzakiA",
        "text" : "RT @the_hindu: With the Centre expecting an extreme cash crunch due to the COVID-19 crisis, most Central government departments have been aâ¦"
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f785a"),
        "screen_name" : "upstartmagazine",
        "text" : "READ | ANU will accept 2021 students based on their year 11 results in response to the challenges posed by COVID-19â¦ https://t.co/qAAuHdoZ9D"
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f78a5"),
        "screen_name" : "entrevistascdnn",
        "text" : "RT @CDNN23Nicaragua: ð 5,678 casos de #coronavirus ð· COVID-19, es el reporte oficial de #CEPREDENAC sobre Centro AmÃ©rica y Dominicana a lasâ¦"
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f78c4"),
        "screen_name" : "lonegamer78",
        "text" : "RT @quarridors: Here's a open caption subtitled version of the Emily Maitlis Newsnight clip about COVID-19's impacts on social welfare thatâ¦"
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f78ce"),
        "screen_name" : "10minus6cosm",
        "text" : "RT @hokxela: Singapore COVID-19 Cases Timeline Chart\nTotal 1623(+142), Recovered 406(+29), ICU 29, Death 6\n8 APR\n\nâ¨Updated chart at https:/â¦"
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f792e"),
        "screen_name" : "kimotakayesu",
        "text" : "RT @srrezaie: COVID-19: Media Hype vs Fact\n\nACEI/ARBs\nIbuprofen\nHCQ/AZT\n\nVideo I put together for folks in my community, that are non-medicâ¦"
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7931"),
        "screen_name" : "financemagnates",
        "text" : "SGX RegCo implements measures to support issuers amid COVID-19 outbreak https://t.co/xzp15Mdgdx #covid_19â¦ https://t.co/D9hptHi1YP"
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7940"),
```

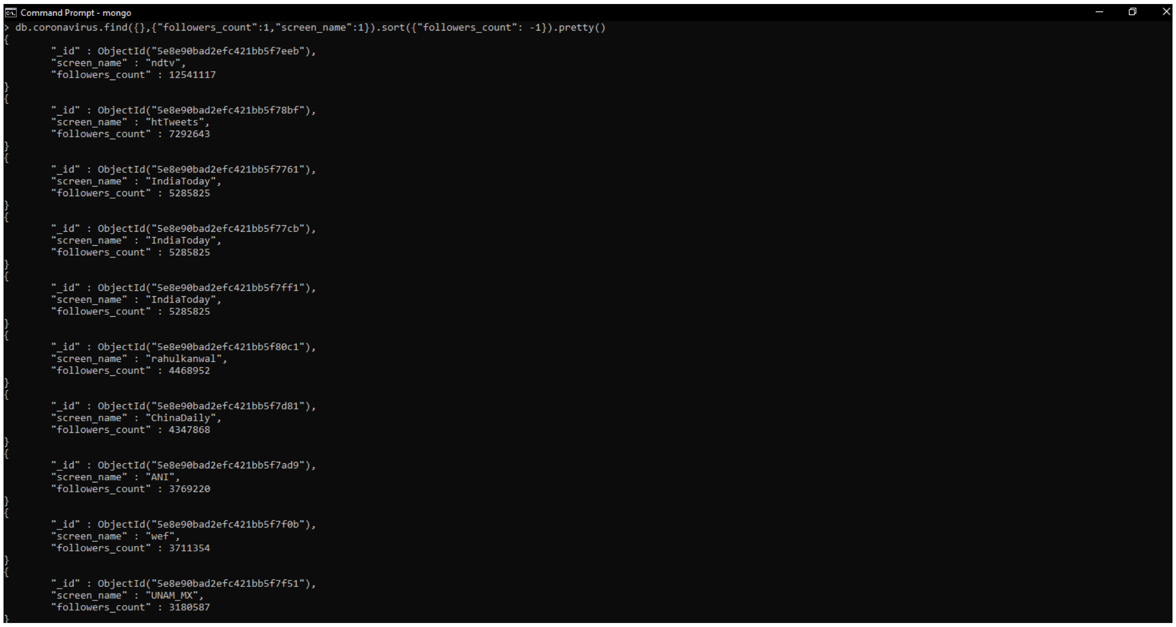
Q3 - What People, Places or Things are popular in your domain?

MongoDB Shell Script: db.coronavirus.find({}, {"followers_count":1,"screen_name":1}).sort({"followers_count": -1}).pretty()

In [21]:

```
#Output Screensot of the script
from IPython.display import Image
Image("Q3_Answer.png")
```

Out[21]:



```
> db.coronavirus.find({},{"followers_count":1,"screen_name":1}).sort({"followers_count": -1}).pretty()
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7eeb"),
        "screen_name" : "ndtv",
        "followers_count" : 12541117
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f78bf"),
        "screen_name" : "htTweets",
        "followers_count" : 7292643
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7761"),
        "screen_name" : "IndiaToday",
        "followers_count" : 5285825
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f77cb"),
        "screen_name" : "IndiaToday",
        "followers_count" : 5285825
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7ff1"),
        "screen_name" : "IndiaToday",
        "followers_count" : 5285825
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f80c1"),
        "screen_name" : "rahulkanwal",
        "followers_count" : 4468952
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7d81"),
        "screen_name" : "ChinaDaily",
        "followers_count" : 4347868
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7ad9"),
        "screen_name" : "ANI",
        "followers_count" : 3769220
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7f0b"),
        "screen_name" : "wef",
        "followers_count" : 3711354
}
{
        "_id" : ObjectId("5e8e90bad2efc421bb5f7f51"),
        "screen_name" : "UNAM_MX",
        "followers_count" : 3180587
}
```

Q4 - What People, Places or Things are trending in your domain?

In [22]:

```
import pprint
pprint.pprint(collection.find_one({"text": "Brazil"},{"_id":1}))
```

```
None
```

In [23]:

```
list(collection.find({
"created_at": {
        "$gte":"2020-04-08 22:40:57",
        "$lte": "2020-04-09 01:40:57"
    }
}))
```

Out[23]:

```
[]
```

AUDIT VALIDITY/ACCURACY: With the above data we extrated the twitter data into a csv file and loaded into mongoDB server. We used different pakages in cleaning the date into user readable format and ran mongo quries to get the intended output.

AUDIT COMPLETNESS: All the questions are answered with respect to real implementation in real world and the assignment ask.

AUDIT CONSISTENCY/UNIFORMITY: The datasets which we used in this assignment all the needed fields for proper extration of needed data.

LICENSE:

Copyright 2020 Srushti Dhamangaonkar

Copyright 2020 Srushti Dhamangaonkar

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.