In [ ]:

```
#You need to create a twitter developer account
```

In [1]:

```
#Example from Just added more notes
#https://www.earthdatascience.org/courses/use-data-open-source-python/intro-to
-apis/twitter-data-in-python/
```

Authorizing an application to access Twitter account data

- To access the Twitter API, you will need 4 things from the your Twitter App page.
- These keys are located in your Twitter app settings in the Keys and Access Tokens tab.

In [6]:

```
#import libraries that you need
import tweepy
from tweepy import OAuthHandler, Stream

#import other required libraries
import os
import pandas as pd
```

- Go to https://developer.twitter.com/en/apps (https://developer.twitter.com/en/apps) to create an app and get values
- for these credentials, which you'll need to provide in place of these
- empty string values that are defined as placeholders.
- See https://developer.twitter.com/en/docs/basics/authentication/overview/oauth (https://developer.twitter.com/en/docs/basics/authentication/overview/oauth)
- for more information on Twitter's OAuth implementation.

In [7]:

```
#required keys and tokens

access_token = '1219125691028930560-CZyXhFlgCpMM8rG11KwuYJaMoX7uNa'
access_secret = '0qSAxEoLmH9pKDPVFy2pQzbloaGxRcbb0JJgnkhq5F2d4'
consumer_key = 'HpU6B5BVTuwAfa5nYX1vAVxgD'
consumer_secret = 'Cgcs5YIHpIp5Pu5US3N0XAX8N4jlJgPQrE4aK8LYkM89nqeTQa'


auth = tweepy.OAuthHandler(consumer_key, consumer_secret)
auth.set_access_token(access_token, access_secret)

twitter_api = tweepy.API(auth)



# Nothing to see by displaying twitter_api except that it's now a
# defined variable

print(twitter_api)
```

```
<tweepy.api.API object at 0x000001FAEF7B8F08>
```

Search Twitter for Tweets

- Now you are ready to search Twitter for recent tweets!
- Start by finding recent tweets that use the #wildfires hashtag.
- You will use the .Cursor method to get an object containing tweets containing the hashtag #wildfires.
- To create this query, you will define the:
  - Search term - in this case #wildfires
  - the start date of your search
  - Remember that the Twitter API only allows you to access the past few weeks of tweets, so you cannot dig into the history too far.

In [9]:

```
# Define the search term and the date_since date as variables
search_words = "#wildfires"
date_since = "2018-11-16"
```

In [10]:

```
#Below you use .Cursor() to search twitter for tweets containing the search te
rm #wildfires.
#You can restrict the number of tweets returned by specifying a number in the
 .items() method.
#.items(5) will return 5 of the most recent tweets.

# Collect tweets
tweets = tweepy.Cursor(twitter_api.search,
              q=search_words,
              lang="en",
              since=date_since).items(5)
tweets
```

Out[10]:

```
<tweepy.cursor.ItemIterator at 0x1faec837308>
```

- .Cursor() returns an object that you can iterate or loop over to access the data collected.
- Each item in the iterator has various attributes that you can access to get information about each tweet including:
  - the text of the tweet
  - who sent the tweet
  - the date the tweet was sent and more
- The code below loops through the object and prints the text associated with each tweet.

In [11]:

```
# Collect tweets
tweets = tweepy.Cursor(twitter_api.search,
              q=search_words,
              lang="en",
              since=date_since).items(5)

# Iterate and print tweets
for tweet in tweets:
    print(tweet.text)
```

```
1/2 Local officials in #Russia's remote #Irkutsk province are caught on camer
a setting fire to the forests. The ori… https://t.co/zGjOTr2iUj
RT @m_parrington: Save the dates for some exciting presentations on global #w
ildfires at #EGU20 next week!

Session NH7.2 on spatial &amp; temp…
RT @sfe2016dublin: #GlobalWarning !!

#FireLosses In #Australia &gt; #Bushfires Leave 470 Plants &amp; Nearly 200 A
nimals In Extreme Stress ~ Gove…
RT @Grantham_IC: "In a matter of years the UK will be ill prepared to handle
#wildfires. It must consider what it might need in the future…
RT @m_parrington: Save the dates for some exciting presentations on global #w
ildfires at #EGU20 next week!

Session NH7.2 on spatial &amp; temp…
```

- The above approach uses a standard for loop.
- However, this is an excellent place to use a Python list comprehension.
- A list comprehension provides an efficient way to collect object elements contained within an iterator as a list.

In [12]:

```
# Collect tweets
tweets = tweepy.Cursor(twitter_api.search,
                       q=search_words,
                       lang="en",
                       since=date_since).items(5)

# Collect a list of tweets
[tweet.text for tweet in tweets]
```

Out[12]:

```
["1/2 Local officials in #Russia's remote #Irkutsk province are caught on cam
era setting fire to the forests. The ori… https://t.co/zGjOTr2iUj",
 'RT @m_parrington: Save the dates for some exciting presentations on global
#wildfires at #EGU20 next week!\n\nSession NH7.2 on spatial &amp; temp…',
 'RT @sfe2016dublin: #GlobalWarning !!\n\n#FireLosses In #Australia &gt; #Bus
hfires Leave 470 Plants &amp; Nearly 200 Animals In Extreme Stress ~ Gove…',
 'RT @Grantham_IC: "In a matter of years the UK will be ill prepared to handl
e #wildfires. It must consider what it might need in the future…',
 'RT @m_parrington: Save the dates for some exciting presentations on global
#wildfires at #EGU20 next week!\n\nSession NH7.2 on spatial &amp; temp…']
```

To Keep or Remove Retweets

- A retweet is when someone shares someone else’s tweet.
- It is similar to sharing in Facebook.
- Sometimes you may want to remove retweets as they contain duplicate content that might skew your analysis if you are only looking at word frequency.
- Other times, you may want to keep retweets.
- Below you ignore all retweets by adding -filter:retweets to your query.
- The Twitter API documentation has information on other ways to customize your queries.

```
In [13]: new_search = search_words + " -filter:retweets"
         new_search
```

```
Out[13]: '#wildfires -filter:retweets'
```

```
In [14]: tweets = tweepy.Cursor(twitter_api.search,
                                q=new_search,
                                lang="en",
                                since=date_since).items(5)

         [tweet.text for tweet in tweets]
```

```
Out[14]: ["1/2 Local officials in #Russia's remote #Irkutsk province are caught on cam
         era setting fire to the forests. The ori… https://t.co/zGjOTr2iUj",
          'Save the dates for some exciting presentations on global #wildfires at #EGU
         20 next week!\n\nSession NH7.2 on spatial… https://t.co/BaV9Fc8yLX',
          '"In a matter of years the UK will be ill prepared to handle #wildfires. It
         must consider what it might need in the… https://t.co/j3BlY3SFZg',
          '✔87% of #wildfires in #Siberia man-made \n✔ Many Russian cities cloaked
         spring fire smog \n\nNews now: "Forest arson… https://t.co/VhEb8OYo9X',
          'Got them...\n\nRECYCLED #QuickDraw from the #AbLeg public (not empty press)
         gallery #Abpoli #NDP #UCP #Abpoli… https://t.co/E7Y2ZzFyQ5']
```

Who is Tweeting About Wildfires?

- You can access a wealth of information associated with each tweet. Below is an example of accessing the users who are sending the tweets related to #wildfires and their locations.
- Note that user locations are manually entered into Twitter by the user.
- Thus, you will see a lot of variation in the format of this value.
    - tweet.user.screen_name provides the user’s twitter handle associated with each tweet.
    - tweet.user.location provides the user’s provided location.
- You can experiment with other items available within each tweet by typing tweet. and using the tab button to see all of the available attributes stored.

In [15]:

```
tweets = tweepy.Cursor(twitter_api.search,
                           q=new_search,
                           lang="en",
                           since=date_since).items(5)

users_locs = [[tweet.user.screen_name, tweet.user.location] for tweet in tweet
s]
users_locs
```

Out[15]:

```
[['A_Melikishvili', 'Paris, France'],
 ['m_parrington', 'UK'],
 ['Grantham_IC', 'Imperial College London'],
 ['changeobserved', ''],
 ['DougBrinkman', '']]
```

Create a Pandas Dataframe From A List of Tweet Data

One you have a list of items that you wish to work with, you can create a pandas dataframe that contains that data.

In [20]:

```
tweet_text = pd.DataFrame(data=users_locs,
                    columns=['user', "location"])
tweet_text
```

Out[20]:

	user	location
0	A_Melikishvili	Paris, France
1	m_parrington	UK
2	Grantham_IC	Imperial College London
3	changeobserved	
4	DougBrinkman	

Customizing Twitter Queries

- For instance, if you search for climate+change, Twitter will return all tweets that contain both of those words (in a row) in each tweet.

In [19]:

```
#Note that the code below creates a list that can be queried
#using Python indexing to return the first five tweets.

new_search = "climate+change -filter:retweets"

tweets = tweepy.Cursor(twitter_api.search,
                   q=new_search,
                   lang="en",
                   since='2018-04-23').items(1000)

all_tweets = [tweet.text for tweet in tweets]
all_tweets[:5]
```

Out[19]:

```
['Climate Change Threatens Drinking Water Across Great Lakes https://t.co/bKI
H1SNfHa',
 "@Reuters One thinks climate change isn't happening at all (which is wrong)
and the other thinks climate change is a… https://t.co/y4MXM1Adq7",
 'My generation blew it. Fortunately, youth are stepping up. https://t.co/2Yk
l091N0X',
 "Guidelines issued for handling of waste generated during COVID-19 patient's
treatment \nCentral Pollution Control Bo… https://t.co/ROuFzidv5e",
 'Mutual aid groups respond to double threat of coronavirus and climate chang
e https://t.co/CRjnTLUZzh https://t.co/8NAecFp1tz']
```