**Set up twitter API access:**  <https://apps.twitter.com/app>

In the above specified link, after logging in, we need to set up the twitter API access by providing the application name, description and website. These are provided under the settings tab. We are then provided with our required Consumer key (API Key), Consumer token (API Token), Access Token and Access Token Secret.

**Python program (sreaming\_API.py):**

After creating a twitter application, we then write this python program to connect the application with the mongodb localhost.

1. Import the libraries and required function in the first few lines of code.
2. The mongodb path has been setup:

MONGODB\_HOST= **'mongodb://localhost/your\_database\_name'**

1. Words that we want to listen from twitter are setup.

**WORDS = ['#country', '#games', '#2017', '#football']**

Connection to the twitter API is made as shown below

CONSUMER\_KEY= “your\_api\_key”

CONSUMER\_SECRET= “your\_api\_secret”

ACCESS\_TOKEN= “your\_token”

ACCESS\_TOKEN\_SECRET= “your\_token\_secret”

1. Listener class is build

class StreamListener(tweepy.StreamListener):

#This is a class provided by tweepy to access the Twitter Streaming API.

def on\_connect(self):

# Called initially to connect to the Streaming API

print("You are now connected to the streaming API.")

def on\_error(self, status\_code):

# On error - if an error occurs, display the error / status code

print('An Error has occured: ' + repr(status\_code))

return False

def on\_data(self, data):

#This is the meat of the script...it connects to your mongoDB and stores the tweet

try:

client = MongoClient(MONGO\_HOST)

# Use twitterdb database. If it doesn't exist, it will be created.

db = client.twitterdb

# Decode the JSON from Twitter

datajson = json.loads(data)

#grab the 'created\_at' data from the Tweet to use for display

created\_at = datajson['created\_at']

#print out a message to the screen that we have collected a tweet

print("Tweet collected at " + str(created\_at))

#insert the data into the mongoDB into a collection called twitter\_search

#if twitter\_search doesn't exist, it will be created.

db.twitter\_search.insert(datajson)

except Exception as e:

print(e)

1. Set up to start the listener class is made

auth = tweepy.OAuthHandler(CONSUMER\_KEY, CONSUMER\_SECRET)

auth.set\_access\_token(ACCESS\_TOKEN, ACCESS\_TOKEN\_SECRET)

#Set up the listener. The 'wait\_on\_rate\_limit=True' is needed to help with Twitter API rate limiting.

listener = StreamListener(api=tweepy.API(wait\_on\_rate\_limit=True))

streamer = tweepy.Stream(auth=auth, listener=listener)

print("Tracking: " + str(WORDS))

streamer.filter(track=WORDS)

**MongoDB:**

1. Connect to the local host from command prompt:

Open the command and change directory to the directory at which mongodb’s bin folder is present. Make sure that you already have a data folder in the parent directory. Now executethe command ‘mongod’ which starts the mongodb local host.

Open and another command prompt, change directory to your bin folder and execute command ‘mongo’ which starts our mongodb server where you can execute your mongodb commands

1. Create database twitterdb

Mongodb Enterprise> use twitterdb

This creates and switches database to twitterdb database

1. We can also view and manage data using mongodb compass

Connect to the localhost by giving the hostname as localhost and port number as 27017 (These values will also be displayed in fields as default)

1. Export data from mongodb to csv

**mongoexport --host=localhost:27017 --db=twitterdb --collection=twitter\_search --type=csv --fields=id -o C:\Users\chala\Desktop\twData.csv**