**import tweepy**

**import codecs**

**import my\_keys**

**# Get the authentication keys from my\_keys.py**

**CONSUMER\_KEY = my\_keys.CONSUMER\_KEY**

**CONSUMER\_SECRET = my\_keys.CONSUMER\_SECRET**

**ACCESS\_KEY = my\_keys.ACCESS\_KEY**

**ACCESS\_SECRET = my\_keys.ACCESS\_SECRET**

**# Authenticate to Twitter with my keys**

**auth1 = tweepy.OAuthHandler(CONSUMER\_KEY, CONSUMER\_SECRET)**

**auth1.set\_access\_token(ACCESS\_KEY, ACCESS\_SECRET)**

**api = tweepy.API(auth1)**

**# create a python class that tells Tweepy what to do when a new status is detected**

**class StreamListener(tweepy.StreamListener):**

**def on\_status(self, status):**

**tweet = status.text**

**user = status.author**

**userid = status.author.id**

**time = status.created\_at**

**source = status.source**

**tweetid = status.id**

**if not ('RT @' in tweet) : print("\"%s\",\"%s\",\"%s\",\"%s\",\"%s\"" % (tweet,user,userid,time,source))**

**# a method to catch error 420 (exceed max number of attempts), and disconnect my stream as protection**

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

**if status\_code == 420:**

**return False**

**# create an instance of the StreamListener object**

**StreamListener = StreamListener()**

**# Connect to the streaming API and save tweets matching my keywords**

**myStream = tweepy.Stream(auth = api.auth, listener=StreamListener)**

**myStream.filter(track=['cost of living',’expensive’,’slowdown’])**