Task-4: Stream twitter data into Spark continuously for 2 hours. After 2 hours the streaming should automatically stop. Identify 2 topic keywords. As the data is streaming in, based on the 2 topic keywords, count the number of times the keywords appear in the tweets. Output the count for each keyword. Caution: Do not use common words.

Note: Specify the following in your README file:

- The 2 keywords used
- The date and time when you started collecting data.

(marks will be deducted if the above information is not included in the README file)

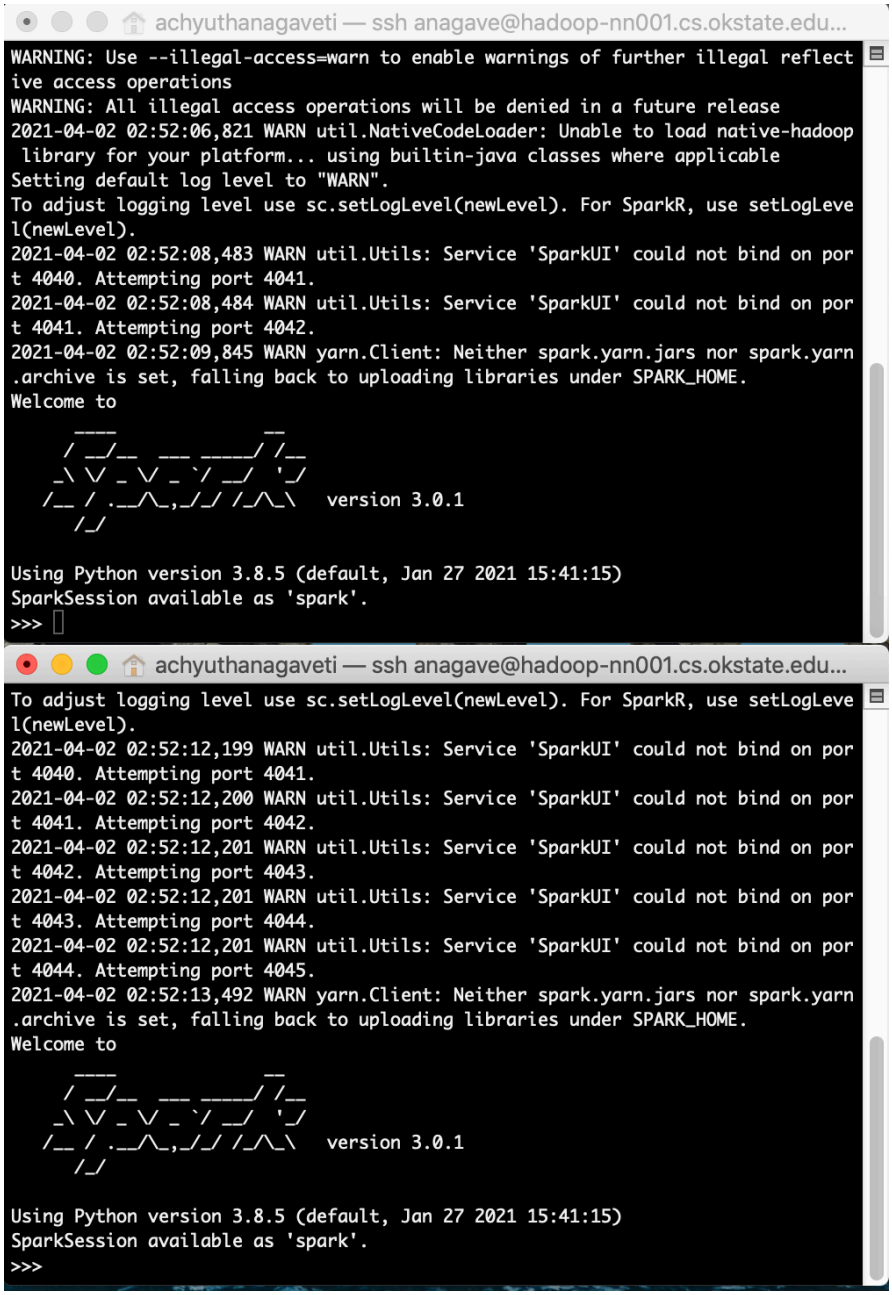
Step 1: Login to HADOOP CLUSTER and Navigate to Pyspark using pyspark command

```
(base) Achyuthas-MacBook-Air:~ achyuthanagaveti$ ssh anagave@hadoop-nn001.cs.okstate.edu
anagave@hadoop-nn001.cs.okstate.edu's password:
Welcome to Ubuntu 20.04.1 LTS (GNU/Linux 5.4.0-62-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

Last login: Fri Apr  2 02:02:28 2021 from 10.200.206.99
```

```
anagave@hadoop-nn001:~$ pyspark
```

```
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 3.0.1
      /_/

Using Python version 3.8.5 (default, Jan 27 2021 15:41:15)
SparkSession available as 'spark'.
```

Step 2: Open two terminals with two spark session. One will act as listener and other will be the data sender.



Step 3: Type the code as in Achyutha_NagavetiBhavaniSanthoshi_Program_4 file, Type Listener session in one terminal and Type Data Sender session in other terminal.

Step 4: Import all the required packages

```
>>> import tweepy
>>> from tweepy import OAuthHandler
>>> from tweepy import Stream
>>> from tweepy.streaming import StreamListener
>>> import socket
>>> import json
>>> from datetime import datetime
>>> now = datetime.now()
>>> print(now)
2021-04-02 02:53:03.315525
```

Step 5: Give the access tokens and API tokens from twitter developer account

```
>>> access_token = "1163449936140222464-RtMJqcL1kbvtdOm8uAHTF3TkMp3bLP"
>>> access_secret = "9GFCAce7BMWHfPcUjYkdqomZ9Ix1GjLQei4XhpAyvl1Fw"
>>> consumer_key = "U20Ejib46e1uOV2dva3FXkTim"
>>> consumer_secret = "llGqrSuXUNyz3767ZGaQDeJTlr4AcJCZ5lVwUraeDKxiXchxuu"
```

Step 6: Create a class for tweets listener

```
>>> class TweetsListener(StreamListener):
...     def __init__(self, csocket):
...             self.client_socket = csocket
...     def on_data(self, data):
...             try:
...                     msg = json.loads(data)
...                     print(msg['text'].encode('utf-8'))
...                     self.client_socket.send(msg['text'].encode('utf-8'))
...                     return True
...             except BaseException as e:
...                     print("Something is wrong : %s" % str(e))
...                     return True
...     def on_error(self, status):
...             print(status)
...             return True
...
>>> def sendData(c_socket,Keyword):
...     auth = OAuthHandler(consumer_key, consumer_secret)
...     auth.set_access_token(access_token, access_secret)
...     twitter_stream = Stream(auth, TweetsListener(c_socket))
...     twitter_stream.filter(track=Keyword,languages=["en"])
...
```

Step 7: Specify the get the host name through socket connection and Mention the port for listening and give required address.

```
>>> s = socket.socket()
>>> host = socket.gethostname()
```

```
>>> port = 5566
>>> s.bind((host, port))
>>> print("Listening on port: %s" % str(port))
Listening on port: 5566
>>> s.listen(5)
>>> c, addr = s.accept()
>>> print("Received request from: " + str(addr))
Received request from: ('192.168.122.2', 57122)
```

Step 8: Create a sender session and type the code as in Achyutha_NagavetiBhavaniSanthoshi_Program_4 file. Firstly, Import all required packages

```
>>> import pyspark
>>> from pyspark.sql import SparkSession
>>> from pyspark.sql.functions import *
>>> from pyspark.sql.types import *
>>> import pyspark.sql.functions as F
>>> from datetime import datetime
>>> session = SparkSession.builder.appName("Twitter-stream").master("local[*]").
getOrCreate()
>>> now = datetime.now()
>>> print(now)
2021-04-02 02:58:18.963137
```

Step 9: Create a def for Data preprocessing

```
>>> def preprocessing(lines):
...     words = lines.select(explode(split(lines.value, "t_end")).alias("word"))
...     words = words.na.replace('', None)
...     words = words.na.drop()
...     words = words.withColumn('word', F.regexp_replace('word', r'http\S+', ''
))
...     words = words.withColumn('word', F.regexp_replace('word', '@\w+', ''))
...     words = words.withColumn('word', F.regexp_replace('word', '#', ''))
...     words = words.withColumn('word', F.regexp_replace('word', 'RT', ''))
...     words = words.withColumn('word', F.regexp_replace('word', ':', ''))
...     words = words.withColumn('word', F.regexp_replace('word', '[^a-z_A-Z ]',
 ''))
...     words = words.withColumn('word', F.regexp_replace('word', "##$$$123&&'?!
!__!','[^[:alnum:]'' '']", ' '))
...     return words
...
```

Step 10: Create variables for lines and words which will be used for preprocessing

```
>>> lines = session.readStream.format("socket").option("host","hadoop-nn001.cs.o
kstate.edu").option("port", 5566).load()
2021-04-02 03:00:09,239 WARN sources.TextSocketSourceProvider: The socket source
 should not be used for production applications! It does not support recovery.
>>> words  = preprocessing(lines)
```

Step 11: Use filter to search for the keywords NASA and Mars. Using a query the count is displayed.

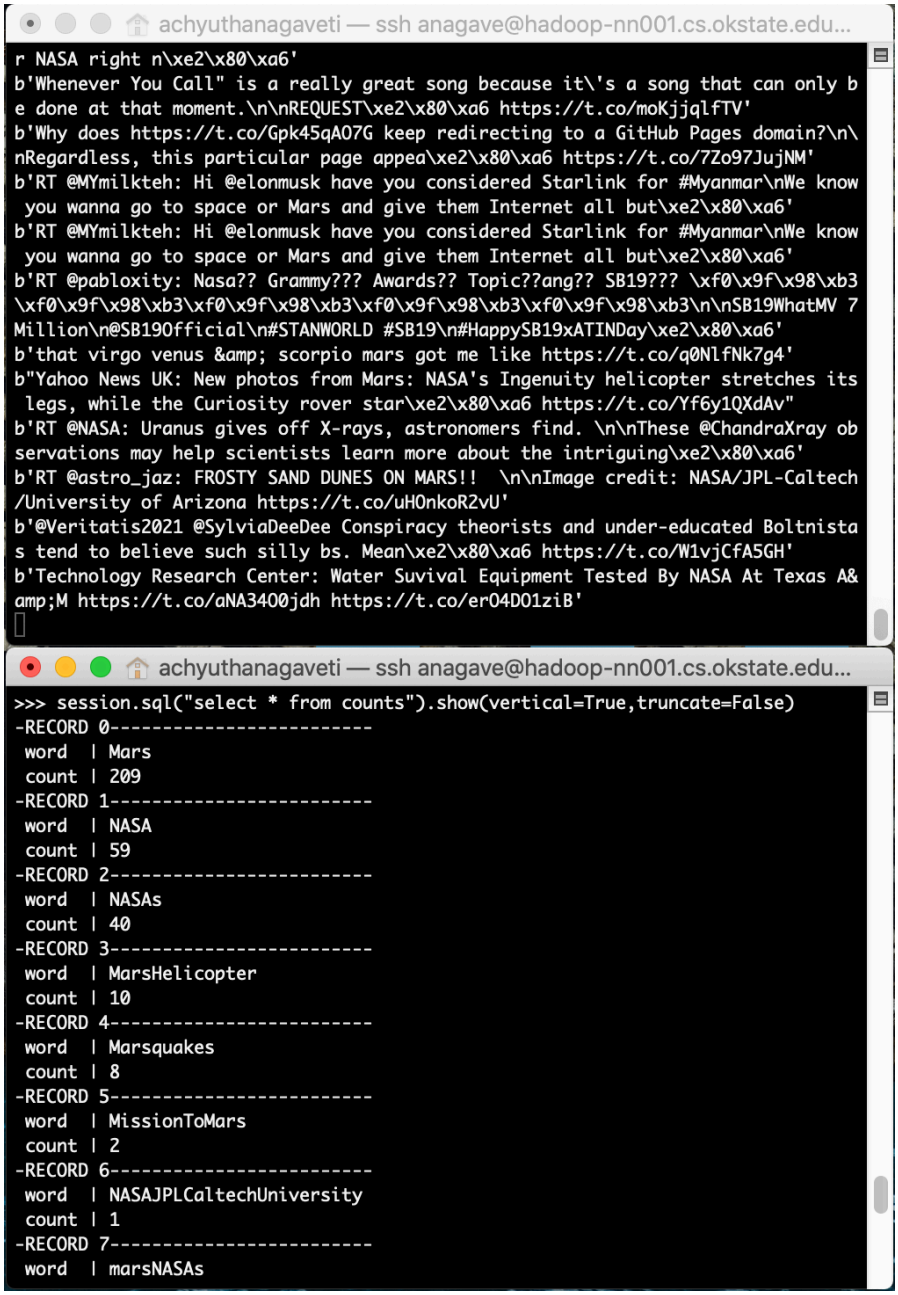
```
>>> filtered = words.withColumn('word', explode(split(col('word'), ' '))) \
...             .groupBy('word') \
...             .count() \
...             .sort('count', ascending=False). \
...             filter((col('word').contains("NASA")) | (col('word').contains('M
ars')))
>>> query = filtered.writeStream\
...                 .outputMode("complete")\
...                 .format("memory")\
...                 .queryName("counts")\
...                 .start()
2021-04-02 03:01:12,840 WARN streaming.StreamingQueryManager: Temporary checkpoi
nt location created which is deleted normally when the query didn't fail: /tmp/t
emporary-f9ecf06e-07ce-4406-8147-6e5d5ee127eb. If it's required to delete it und
er any circumstances, please set spark.sql.streaming.forceDeleteTempCheckpointLo
cation to true. Important to know deleting temp checkpoint folder is best effort
.
[Stage 38:=============================================>        (170 + 2) / 200]
```

Step 12: The query will be listened by the listener session

Step 13: wait till all the stages are completed(stages are highlighted in the screenshots for reference)


achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
ind and help me sell my first one\xe2\x80\xa6 https://t.co/0uQqTXKRrf'
b'\xf0\x9f\x83\x8fNFT integration coming soon (dev said 3 weeks or so).\n\xf0\x9f\x92\xb0 CHAI will integrate Anchor fixed 20% APY for millions of u\xe2\x80\xa6 https://t.co/tORoxRIs3W'
b'RT @Mrbankstips: All I\xe2\x80\x99m saying is, if we don\xe2\x80\x99t checkmate climate change and global warming, 90% of crops will have to be grown in green ho\xe2\x80\xa6'
b'@NftShilling @berni_omar Please check out my amazing artworks under the username quantummind and help me sell my fi\xe2\x80\xa6 https://t.co/uAqpuxZalk'
b"RT @konstructivizm: M87's Central Black Hole in Polarized Light by the EHT\nNASA https://t.co/qmNu75vQAd"
b'RT @VirtualAstro: BREAKING NEWS!!!\n\nFOSSILS FOUND ON MARS!!!\n\nhttps://t.co/pjc5vtSseA'
b'RT @NASA_Marshall: As @NASAArtemis builds a sustained human presence on the Moon, @NASASCaN technologies will help us get our way around >>\xe2\x80\xa6'
b'RT @pabloxity: Nasa?? Grammy??? Awards?? Topic??ang?? SB19??? \xf0\x9f\x98\xb3\xf0\x9f\x98\xb3\xf0\x9f\x98\xb3\xf0\x9f\x98\xb3\xf0\x9f\x98\xb3\n\nSB19WhatMV 7 Million\n@SB19Official\n#STANWORLD #SB19\n#HappySB19xATINDay\xe2\x80\xa6'
b'RT @himbo_anonymous: NASA after getting its first images of a Black Hole'
b'Why #China\xe2\x80\x99 space program could overtake \xe2\x81\xa6@NASA\xe2\x81\xa9 ? https://t.co/3DSzJahRFw'
b'@InfographicTony @MarcusHouse @Erdayastronaut @FelixSchlang @torybruno @elonmusk @Peter_J_Beck @TJ_Cooney @NASA Wha\xe2\x80\xa6 https://t.co/ZOcSc3IMNQ'
achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
[...
[>>> lines = session.readStream.format("socket").option("host","hadoop-nn001.cs.okstate.edu").option("port", 5566).load()
2021-04-02 03:00:09,239 WARN sources.TextSocketSourceProvider: The socket source should not be used for production applications! It does not support recovery.
[>>> words = preprocessing(lines)
>>> filtered = words.withColumn('word', explode(split(col('word'), ' '))) \
...   .groupBy('word') \
...   .count() \
...   .sort('count', ascending=False). \
[...   filter((col('word').contains("NASA")) | (col('word').contains('Mars')))
>>> query = filtered.writeStream\
...   .outputMode("complete")\
...   .format("memory")\
...   .queryName("counts")\
[...   .start()
2021-04-02 03:01:12,840 WARN streaming.StreamingQueryManager: Temporary checkpoint location created which is deleted normally when the query didn't fail: /tmp/temporary-f9ecf06e-07ce-4406-8147-6e5d5ee127eb. If it's required to delete it under any circumstances, please set spark.sql.streaming.forceDeleteTempCheckpointLocation to true. Important to know deleting temp checkpoint folder is best effort.
[Stage 331:========>   (32 + 2) / 200]

Output :


achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
r NASA right n\xe2\x80\xa6'
b'Whenever You Call" is a really great song because it\'s a song that can only b
e done at that moment.\n\nREQUEST\xe2\x80\xa6 https://t.co/moKjjqlfTV'
b'Why does https://t.co/Gpk45qAO7G keep redirecting to a GitHub Pages domain?\n\
nRegardless, this particular page appea\xe2\x80\xa6 https://t.co/7Zo97JujNM'
b'RT @MYmilkteh: Hi @elonmusk have you considered Starlink for #Myanmar\nWe know
you wanna go to space or Mars and give them Internet all but\xe2\x80\xa6'
b'RT @MYmilkteh: Hi @elonmusk have you considered Starlink for #Myanmar\nWe know
you wanna go to space or Mars and give them Internet all but\xe2\x80\xa6'
b'RT @pabloxity: Nasa?? Grammy??? Awards?? Topic??ang?? SB19??? \xf0\x9f\x98\xb3
\xf0\x9f\x98\xb3\xf0\x9f\x98\xb3\xf0\x9f\x98\xb3\xf0\x9f\x98\xb3\n\nSB19WhatMV 7
Million\n@SB19Official\n#STANWORLD #SB19\n#HappySB19xATINDay\xe2\x80\xa6'
b'that virgo venus & scorpio mars got me like https://t.co/q0NlfNk7g4'
b"Yahoo News UK: New photos from Mars: NASA's Ingenuity helicopter stretches its
legs, while the Curiosity rover star\xe2\x80\xa6 https://t.co/Yf6y1QXdAv"
b'RT @NASA: Uranus gives off X-rays, astronomers find. \n\nThese @ChandraXray ob
servations may help scientists learn more about the intriguing\xe2\x80\xa6'
b'RT @astro_jaz: FROSTY SAND DUNES ON MARS!! \n\nImage credit: NASA/JPL-Caltech
/University of Arizona https://t.co/uHOnkoR2vU'
b'@Veritatis2021 @SylviaDeeDee Conspiracy theorists and under-educated Boltnista
s tend to believe such silly bs. Mean\xe2\x80\xa6 https://t.co/W1vjCfA5GH'
b'Technology Research Center: Water Suvival Equipment Tested By NASA At Texas A&
amp;amp;M https://t.co/aNA3400jdh https://t.co/erO4DO1ziB'
achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
>>> session.sql("select * from counts").show(vertical=True,truncate=False)
-RECORD 0------------------------
word | Mars
count | 209
-RECORD 1------------------------
word | NASA
count | 59
-RECORD 2------------------------
word | NASAs
count | 40
-RECORD 3------------------------
word | MarsHelicopter
count | 10
-RECORD 4------------------------
word | Marsquakes
count | 8
-RECORD 5------------------------
word | MissionToMars
count | 2
-RECORD 6------------------------
word | NASAJPLCaltechUniversity
count | 1
-RECORD 7------------------------
word | marsNASAs

```
 word  | marsNASAs
 count | 1
-RECORD 8-----------------------
 word  | MarsWhat
 count | 1
-RECORD 9-----------------------
 word  | DayNASA
 count | 1
-RECORD 10----------------------
 word  | HappySBxATINDayNASAs
 count | 1
-RECORD 11----------------------
 word  | imagesMars
 count | 1
-RECORD 12----------------------
 word  | CountdownToMars
 count | 1
-RECORD 13----------------------
 word  | HappySBxATINDayNASA
 count | 1
-RECORD 14----------------------
 word  | MarsThe
 count | 1
-RECORD 15----------------------
 word  | MarsId
 count | 1
-RECORD 16----------------------
 word  | OccupyMars
 count | 1
-RECORD 17----------------------
 word  | NASAJPLCaltechASU
 count | 1
-RECORD 18----------------------
 word  | meNASA
```

```
-RECORD 18----------------------
 word  | meNASA
 count | 1
-RECORD 19----------------------
 word  | NASAFellows
 count | 1
only showing top 20 rows

[Stage 372:=============================================>       (173 + 1) / 200]
```

To stop the query use below command

```
>>> query.stop()
>>>
```