

---

## Program for Task5:

**Task-5:** Write a Spark trigger based on the processing time. Execute the streaming query of No. 4 above at regular 10 minute intervals. After 3 hours the streaming should automatically stop

### Listener:

*# Importing 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
```

*# Access tokens and API tokens from Twitter developer Credentials to connect to twitter account*

```
>>> access_token = "1163449936140222464-RtMJqcL1kbvtdOm8uAHTF3TkMp3bLP"
>>> access_secret = "9GFCAce7BMWHfPcUjYkdqomZ9Ix1GjLQei4XhpAyl1Fw"
>>> consumer_key = "U20Ejib46e1uOV2dva3FXkTim"
>>> consumer_secret = "lIGqrSuXUNyz3767ZGaQDeJTlr4AcJCZ5IVwUraeDKxiXchxuu"
```

*# Listener class for tweets*

```
>>> class TweetsListener(StreamListener):
    # initialized the constructor
...     def __init__(self, csocket):
...         self.client_socket = csocket
...     def on_data(self, data):
...         try:
...             # Twitter data which comes as a JSON format is read
...             msg = json.loads(data)
...             # the 'text' in the JSON file contains the tweet.
...             print(msg['text'].encode('utf-8'))
...             # the tweet data is sent to the client socket
...             self.client_socket.send(msg['text'].encode('utf-8'))
...             return True
...         except BaseException as e:
...             # Exception handling
...             print("Something is wrong : %s" % str(e))
...             return True
...     def on_error(self, status):
...         print(status)
...         return True
...
```

*# Send the tweets to socket port*

```
>>> def sendData(c_socket, Keyword):
    # passing authentication credentials keys
...     auth = OAuthHandler(consumer_key, consumer_secret)
```

```

... auth.set_access_token(access_token, access_secret)
    # twitter_stream will get the actual live tweet data
... twitter_stream = Stream(auth, TweetsListener(c_socket))
    # filter the tweet feeds related to Keyword and language english
... twitter_stream.filter(track=Keyword, languages=["en"])
...
# create a socket object
>>> s = socket.socket()
# Get local machine name : host and port, Port numbers start from 5000
>>> host = socket.gethostname()
>>> port = 5566
# Bind port and socket
>>> s.bind((host, port))
>>> print("Listening on port: %s" % str(port))
Listening on port: 5566
# Establish the connection with client.
>>> s.listen(5)
>>> c, addr = s.accept()
# Waits till the sender sends the data
>>> print("Received request from: " + str(addr))
Received request from: ('192.168.122.2', 57122)
# Keep the stream data available
>>> sendData(c, Keyword=['NASA', 'Mars'])

```

## Sender:

```

# Importing the 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()
# printing the start time of streaming data
>>> now = datetime.now()
>>> print(now)
2021-04-02 12:43:52.922813
# Defining function for Data pre-processing the tweets. Removing special
characters, RT, hashtags and urls
>>> 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', "###$$$123&&!!__!", '[^[:alnum:]]'
... ']', ''))
... words = words.withColumn('word', F.regexp_replace('word', '^[a-z_A-Z]', ''))
... return words

```

*# load the streaming data from socket connection*

```

>>> lines = session.readStream.format("socket").option("host", "hadoop-
nn001.cs.okstate.edu").option("port", 5599).load()
2021-04-02 12:46:17,767 WARN sources.TextSocketSourceProvider: The socket source should
not be used for production applications! It does not support recovery.

```

*# applying data pre-processing using preprocessing function*

```

>>> 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 mode for triggered processing for the duration of 10 minutes*

```

>>> query_triggered = filtered.writeStream\
...     .outputMode("complete")\
...     .format("memory")\
...     .queryName("count_triggered")\
...     .trigger(processingTime='10 minutes')\
...     .start()

```

2021-04-02 12:46:48,526 WARN streaming.StreamingQueryManager: Temporary checkpoint location created which is deleted normally when the query didn't fail: /tmp/temporary-28534694-4b86-48ca-826d-7eba6e26b0b2. 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.

*#the result is displayed through sql from data stream*

## Output 1:

```
achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
b'RT @FitzyLeakz: BRUNO MARS EMOTE! https://t.co/LpovzUNFPz'
b'RT @ray8fisher: We certainly do.\n\n#SnyderCut'
b'RT @ZackSnyder: We live in a society... where you can watch #ZackSnydersJustic
eLeague on @HBOMax. https://t.co/8l3CwvynZq #SnyderCut #UsUn\xe2\x80\xa6'
b'LETS FUCKING GOOOOOOOO LETS FUCKING GOOOOOOO'
b'Easy, Mars can cut. https://t.co/DsnEbtMiDA'
b'RT @JacyKhan: Feels like Geeta from Swades in a simple Pink cotton saree &
calm village like surrounding \xf0\x9f\xb8 \n\nWaiting for my Handsome Moha
n\xe2\x80\xa6'
b'Hustler Music\nRight above it'
b'RT @NASAEarth: The coastal state of Sinaloa is one of the largest shrimp farmi
ng regions in #Mexico. Vast areas of wetland (blue-green) and\xe2\x80\xa6'
b'RT @kidmingyu: who expected bruno mars and hoshi's first meeting will be on th
e charts \xf0\x9f\xa4\xa3 https://t.co/ntNzfgKKjH"
b'RT @ray8fisher: We certainly do.\n\n#SnyderCut'
b'wait no this is groovy asf. cop!'
b'RT @FitzyLeakz: BRUNO MARS EMOTE! https://t.co/LpovzUNFPz'
b'mars. disgusting chocolate'
b'RT @CaseyDreier: A quick recap of our Day of Action:\n\n* 167 congressional me
etings\n* 145 participants from 30 different states\n* More than\xe2\x80\xa6'
b'@boonecutler ""In reaching this conclusion, the Board concluded that there was
no evidence supporting Santos\xe2\x80\x99s clai\xe2\x80\xa6 https://t.co/NSGXFh
1uhY'

achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
>>> print(datetime.now())
2021-04-02 12:55:37.754789
>>> session.sql("select * from count_triggered").show(vertical=True,truncate=Fal
se)
-RECORD 0-----
word | Mars
count | 43
-RECORD 1-----
word | NASA
count | 9
-RECORD 2-----
word | NASAs
count | 2
-RECORD 3-----
word | UsUnNASA
count | 2
-RECORD 4-----
word | UsUnNASArecently
count | 1
-RECORD 5-----
word | MarsBounty
count | 1
-RECORD 6-----
word | SailorMars
count | 1
-RECORD 7-----
word | RestoreTheSnyderVerseMars
count | 1
>>>
```

## Output 2:

```
achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
b'RT @Hauwa_L: Me, as Elon Musk\'s wife.\n\n"Baby, I\'ve sold Mars ticket to 2,000 people o"\n\nElon: What?! The space ship is not working yet.\n\nMe\xe2\x80\xa6'
b'RT @ray8fisher: We certainly do.\n\n#SnyderCut'
b'RT @NASA_Marshall: Since its 2018 landing, @NASAIoSight has detected more than 500 'marsquakes' on the Red Planet\xe2\x80\x94with two of the strongest\xe2\x80\xa6'
b'RT @ray8fisher: We certainly do.\n\n#SnyderCut'
b'RT @ZackSnyder: We live in a society... where you can watch #ZackSnydersJusticeLeague on @HBOMax. https://t.co/8l3CwvynZq #SnyderCut #UsUn\xe2\x80\xa6'
b'RT @Hauwa_L: Me, as Elon Musk\'s wife.\n\n"Baby, I\'ve sold Mars ticket to 2,000 people o"\n\nElon: What?! The space ship is not working yet.\n\nMe\xe2\x80\xa6'
b'And because that was too much earnestness for my irony-poisoned brain, I will leave you with this Russian meme from\xe2\x80\xa6 https://t.co/PiK22wWMBg'
b'RT @ShinaBR: UPCOMING BRUNO MARS EMOTE\n\n(Thanks to @FrenzyLeaks for sending me the post!) https://t.co/TCwjUnowxS'
b'@databourg April 3, 1926: Birth of US astronaut \xe2\x80\x94\xe2\x80\x94Gus\xe2\x80\x94Grisson (\xe2\x80\x94 January 27, 1967). He was crew member on space mi\xe2\x80\x94\xe2\x80\x94 https://t.co/fATSa6LB46'
b'RT @space360: NASA\xe2\x80\x94s Perseverance rover\xe2\x80\x94s latest images of the surface of Mars. One day humans will step foot on this surface. https://t.c\xe2\x80\x94\xe2\x80\x94'

achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
>>> print(datetime.now())
2021-04-02 13:08:22.010169
>>> session.sql("select * from count_triggered").show(vertical=True,truncate=False)
-RECORD 0-----
word | Mars
count | 322
-RECORD 1-----
word | NASA
count | 57
-RECORD 2-----
word | NASAs
count | 19
-RECORD 3-----
word | SailorMars
count | 4
-RECORD 4-----
word | UsUnNASA
count | 4
-RECORD 5-----
word | VenusMars
count | 3
-RECORD 6-----
word | MarsHelicopter
count | 2
-RECORD 7-----
word | SnyderCutMarsIts
count | 1
-RECORD 8-----
word | UsUnNASArecently
count | 1
```

```
achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
-RECORD 9-----
-RECORD 19-----
word | xdddMars
count | 1
only showing top 20 rows
>>>
word | Mars
count | 1
-RECORD 12-----
word | SnyderCutMars
count | 1
-RECORD 13-----
word | MarsHoly
count | 1
-RECORD 14-----
word | MarsOdyssey
count | 1
-RECORD 15-----
word | MarsBounty
count | 1
-RECORD 16-----
word | NASAMODIS
count | 1
-RECORD 17-----
word | SacchariNASAs
count | 1
-RECORD 18-----
word | Marss
count | 1
-RECORD 19-----
```

## Output 3:

```
achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...
b'RT @HYPEX: Bruno Mars "Leave The Door Open" Emote InGame! (Muted to avoid copy
right) https://t.co/ct1HqLbA26'
b'RT @GermanCoin_GCX: Hi #Cryptonians, \nwe #giveaway for the #holidays 300$ in
$GCX to two of you each!\n\x0f\x9f\x95\x8a\xef\xb8\x8f\x0f\x9f\x95\x8a\xef\xb8\
x8f\x0f\x9f\x95\x8a\xef\xb8\x8f\nJust do this:\n- Followus\n- Re\xe2\x80\xa6'
b'@Erdayastronaut @considercosmos @SpacePadreIsle It looks a like Mars!'
b'RT @meiklwagner: Built: 1440. Rebuilt: 2013 - 2021 #BerlinerStadtschloss #Frid
aysForFuture #BerlinerDom_1905 #Berlin\x0f\x9f\x91\x91 #BMW #ClimateChange #\xe2
\x80\xa6'
b'RT @NASASolarSystem: It\xe2\x80\x99s a new month, stargazers. \xe2\x9c\xa8 \xf
0\x9f\x94\xad In April, look for the constellation Leo, Jupiter and Saturn with
the Moon, and the\xe2\x80\xa6'
b'RT @ray8fisher: We certainly do.\n\n#SnyderCut'
b'@FitzLeakz So possible bruno Mars skin'
b'@Fleiers Exactly mars...'
b'RT @NASASolarSystem: It\xe2\x80\x99s a new month, stargazers. \xe2\x9c\xa8 \xf
0\x9f\x94\xad In April, look for the constellation Leo, Jupiter and Saturn with
the Moon, and the\xe2\x80\xa6'
b'RT @latestinspace: Ingenuity helicopter's first flight has been delayed to Apr
il 11th.\n\nvia @NASA https://t.co/dIBTedb5Wm"
b'RT @ray8fisher: We certainly do.\n\n#SnyderCut'
b'RT @rexglacer: Tired & frustrated says a guy who has done nothing in over
a year while collecting his $180++K salary for tweeting!\nYou peop\xe2\x80\xa6'

2021-04-02 13:18:25.481740
>>> session.sql( select * from count_triggered").show(vertical=True,truncate=Fal
se)
-RECORD 0-----
word | Mars
count | 873
-RECORD 1-----
word | NASA
count | 119
-RECORD 2-----
word | NASAs
count | 39
-RECORD 3-----
word | SailorMars
count | 6
-RECORD 4-----
word | MarsHelicopter
count | 4
-RECORD 5-----
word | UsUnNASA
count | 4
-RECORD 6-----
word | VenusMars
count | 3
-RECORD 7-----
word | Marsemote
count | 2
-RECORD 8-----
word | SnyderCutMars
count | 2
-RECORD 9-----
```

```
-RECORD 19-----  
word | MarSHoly  
count | 1  
only showing top 20 rows  
>>> |
```

achyuthanagaveti — ssh anagave@hadoop-nn001.cs.okstate.edu...

```
-RECORD 9-----  
word | Marss  
count | 2  
-RECORD 10-----  
word | Marsquakes  
count | 2  
-RECORD 11-----  
word | SnyderCutMarsIts  
count | 1  
-RECORD 12-----  
word | NASAJPLCaltechUniversity  
>>> query_triggered.stop()  
>>> triggered_then_datetime.now()  
>>> print(triggered_then)  
2021-04-02 13:22:34.379944  
-RECORD 14-----  
word | AscenNASAs  
count | 1  
-RECORD 15-----  
word | beaNASA  
count | 1  
-RECORD 16-----  
word | asMars  
count | 1  
-RECORD 17-----  
word | NASAJPLCal  
count | 1  
-RECORD 18-----  
word | MarsThis  
count | 1  
-RECORD 19-----
```

To stop the query use  
below command