**BIG DATA PROGRAMMING**

**PROJECT\_BASED EXAM-2**

**Spark Streaming Task**

**OBJECTIVE:**

Perform Word-Count on Twitter Streaming Data.

**INTRODUCTION:**

* Spark streaming helps to access and process Real-time data with the help of different algorithms like map, Reduce, join etc.
* In this project we use MapReduce algorithm to perform word count on streamed **data.**

**QUESTION 3: (1)**

**IDEA OF THE PROJECT:**

Idea of the project is to apply different concepts learnt in Big-Data-Programming so far. Here we use MapReduce method to calculate word frequency.

**QUESTION 3: (2)**

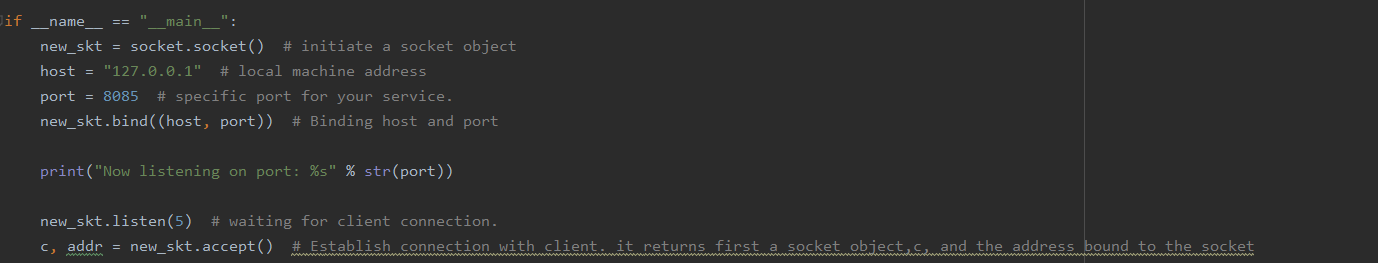
**USAGE OF PROJECT IN REAL-TIME:**

Spark Streaming Context is used for processing the real-time data streams. In real-time, this idea of streaming helps in prediction, analyzing and data processing workloads etc**.**

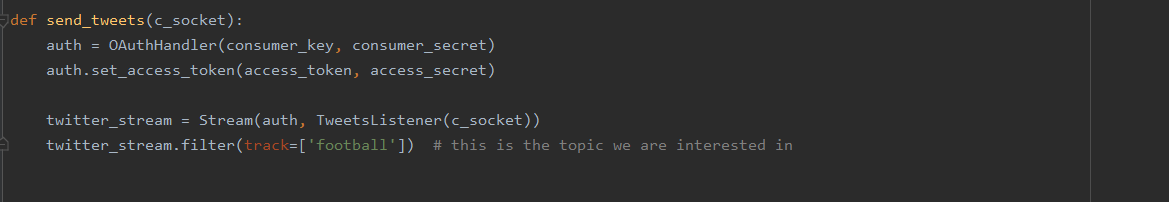
**QUESTION 3: (3)**

**IMPLEMENTATION:**

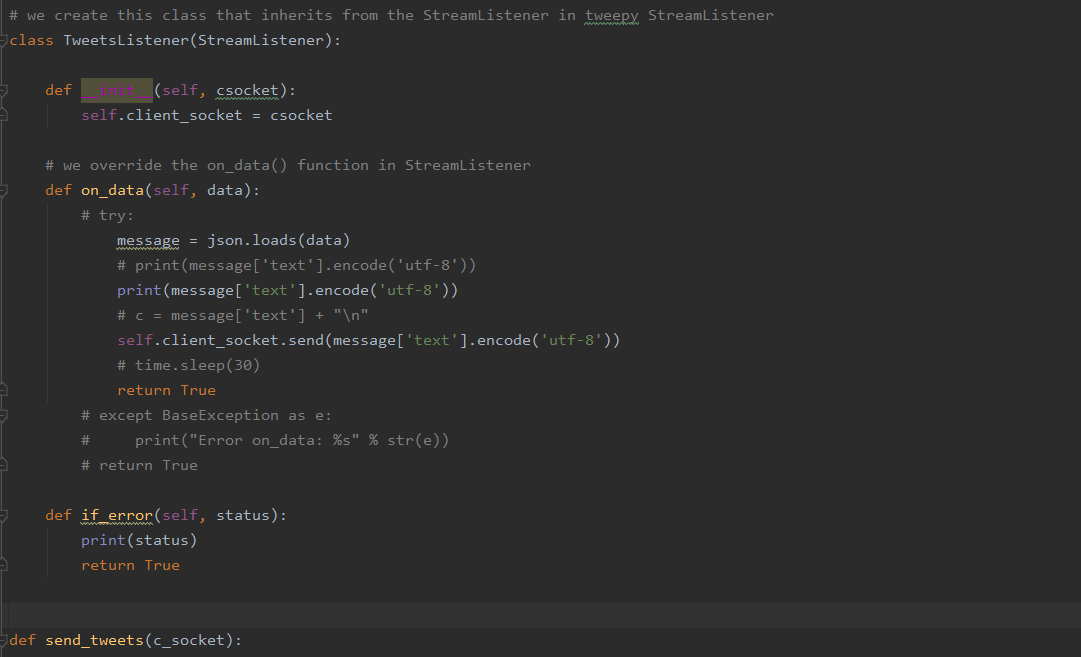
* Initiate a socket object with local machine's IP address and a service specific port number.
* Bind the host and port.
* Make client connection.



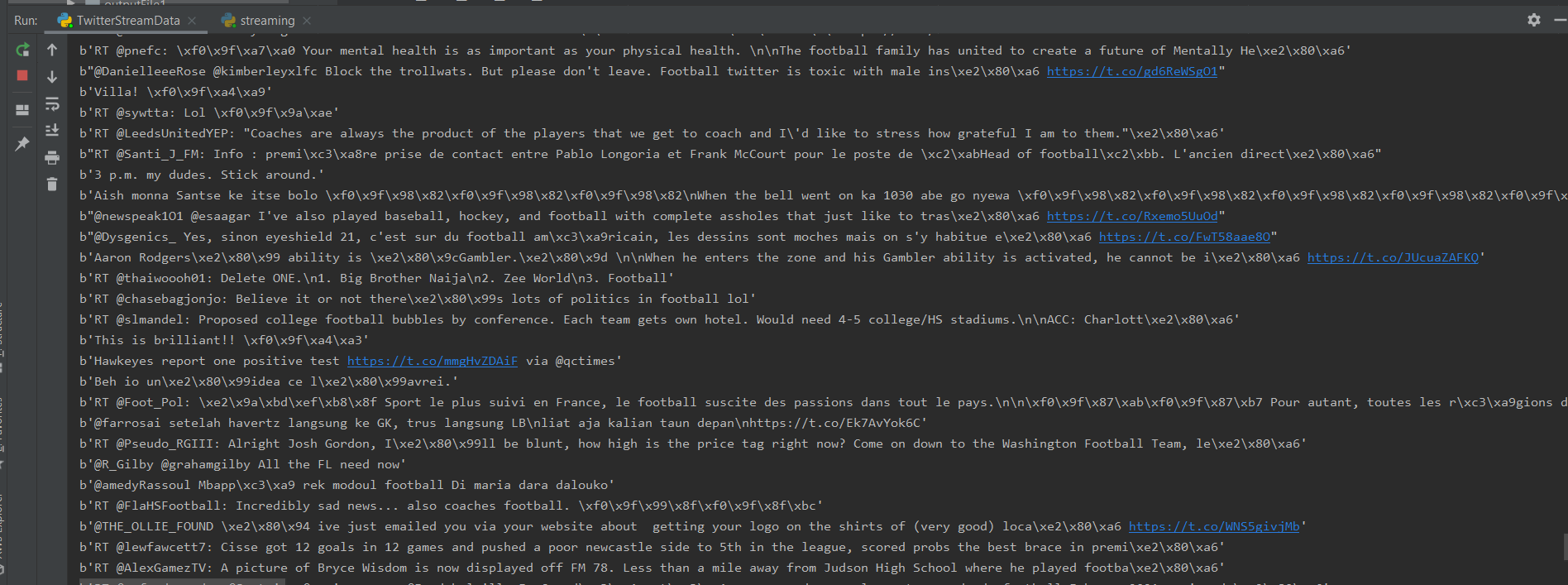
* Create a twitter developer account to get access.
* Get authorization and collect tweets under desired topic.
* Here I have extracted tweets under topic football.



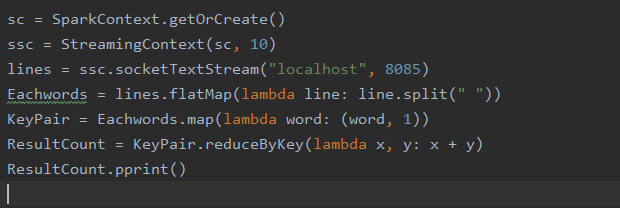
* Create a class and a model where it extracts only text content from the entire data collected.
* Collect the data and read it to self.
* Run the program and the status gets displayed.

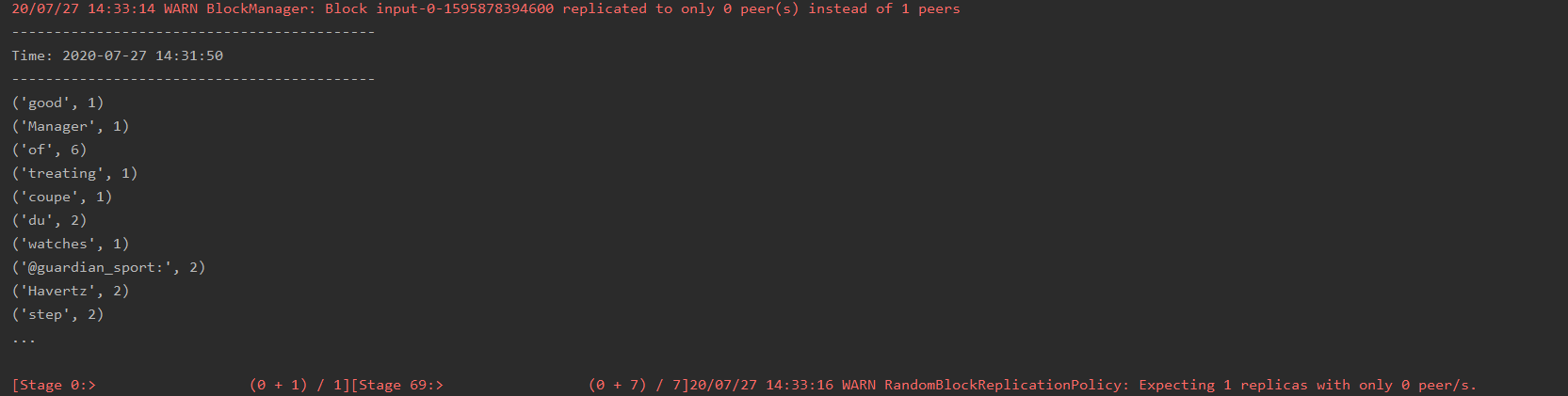
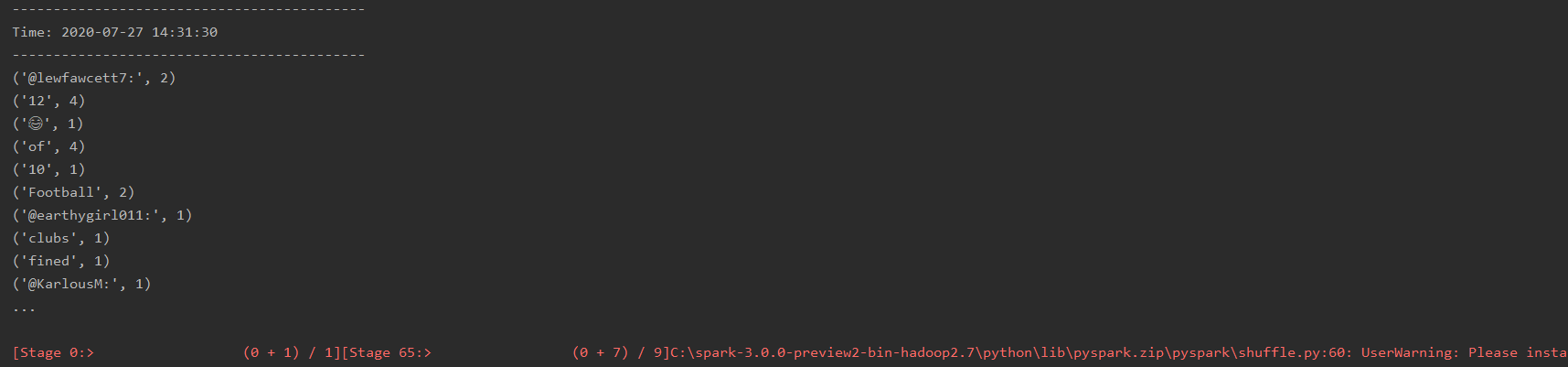


OUTPUT:



* Later initialize the streaming part.
* Give the same port number as given earlier with which client receives the data from server.
* Use flatMap, where it helps in splitting up a string or a sentence Separated and terminated by a delimiter space as shown in the code below.
* Using Map and reduce methods, split up the words and count their repetition.
* Store the resultant word frequency and print them. Thus, output is obtained.



OUTPUT:****

**QUESTION 3: (4)**

**CHALLENGES FACED:**

Had minor issues while writing tweets into socket as it consumes little more time comparatively. And also getting access from twitter also takes time. Irrespective of these two, entire execution procedure went very effectively without any obstacles.

**QUESTION 3: (5)**

**MILESTONES AND INTEGRATION OF THE PROJECT**:

Work went very effective and smooth without any obstacles. Team split-up made it easy to perform tasks. As there is no dependency for one task to other, individuals performed one task each and accomplished the tasks given.

**TEAM MEMBERS AND CONTRIBUTION:**

Roshini varada -- Hadoop MapReduce Algorithm

Link -- https://github.com/RoshiniVarada/BDP\_Project2/wiki/CASE1

Sarika Reddy Kota -- Spark Data Frames

Link -- https://github.com/RoshiniVarada/BDP\_Project2/wiki/CASE2

Pallavi Arikatla -- Spark streaming

Link -- https://github.com/RoshiniVarada/BDP\_Project2/wiki/CASE3

Zakari, Abdulmuhaymin -- Spark Graphx

Link -- https://github.com/RoshiniVarada/BDP\_Project2/wiki/CASE4

**Video Link:**

https://youtu.be/UtsiVZaijyg

**PALLAVI ARIKATLA**

**16301244**