

**COMP-SCI 5540 Principles of Big Data Management**

**Project – 3**

**Twitter Data Analysis**

# Submitted by **Team – 6**

## Rakesh Reddy Pallepati – rpd54

## Uday kiran Dora Sevana – uskc2

## Vijaya Lakshmi Meenakshi Borusu – vbmxb

Onica Sai Prasanna Lakshmi Rayineedi – orr83

**1. Introduction**

The main aim of this project is to develop a system to archive Twitters Data. This project mainly concentrates on collecting tweets from twitter using twitter API, and performing Analytical Analysis on the collected Twitter data using the keyword “Music”. Analytical Analysis is implemented using Analytical tasks, these tasks consists of Queries which are implemented using Scala. We get the results in the form of tables which are stored in MongoDB and then used for visualization. We try to emulate the twitter file system and understand the way it works through this analysis. By doing this we get to understand the handling and analysis of large amounts of data in the real-time environment.

**Tasks Performed:**

* Collection of tweets in Java Script Object Notation (JSON) format about “MUSIC”
* For the collected tweets, Analytical tasks are written in the form of Queries
* These Queries are implemented using Scala
* There are Four Data Frames and one RDD implemented (For, tweets.json file which contains of twitter data on “Music”)
* Also, a RDD is implemented for the “trends.txt” file using Scala.
* Results are obtained in the form of Tables
* These tables are stored in MongoDB which are utilized for Visualization

**2. Requirements of the System:**

**Software Requirements:**

* Python
* JDK
* Scala
* Intellij IDEA
* Apache Spark

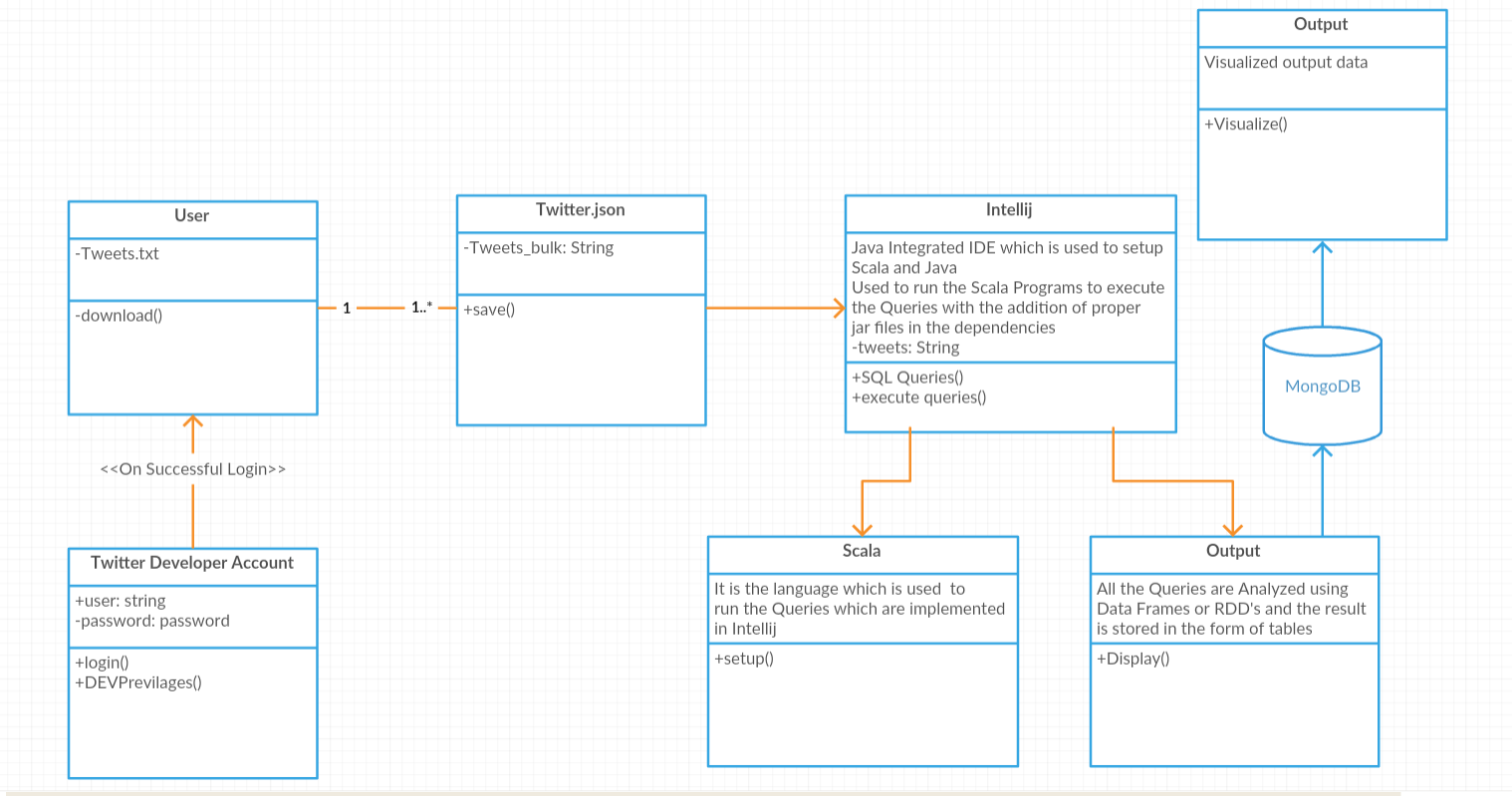
**Language Requirements:**

* Python
* Scala
* SQL

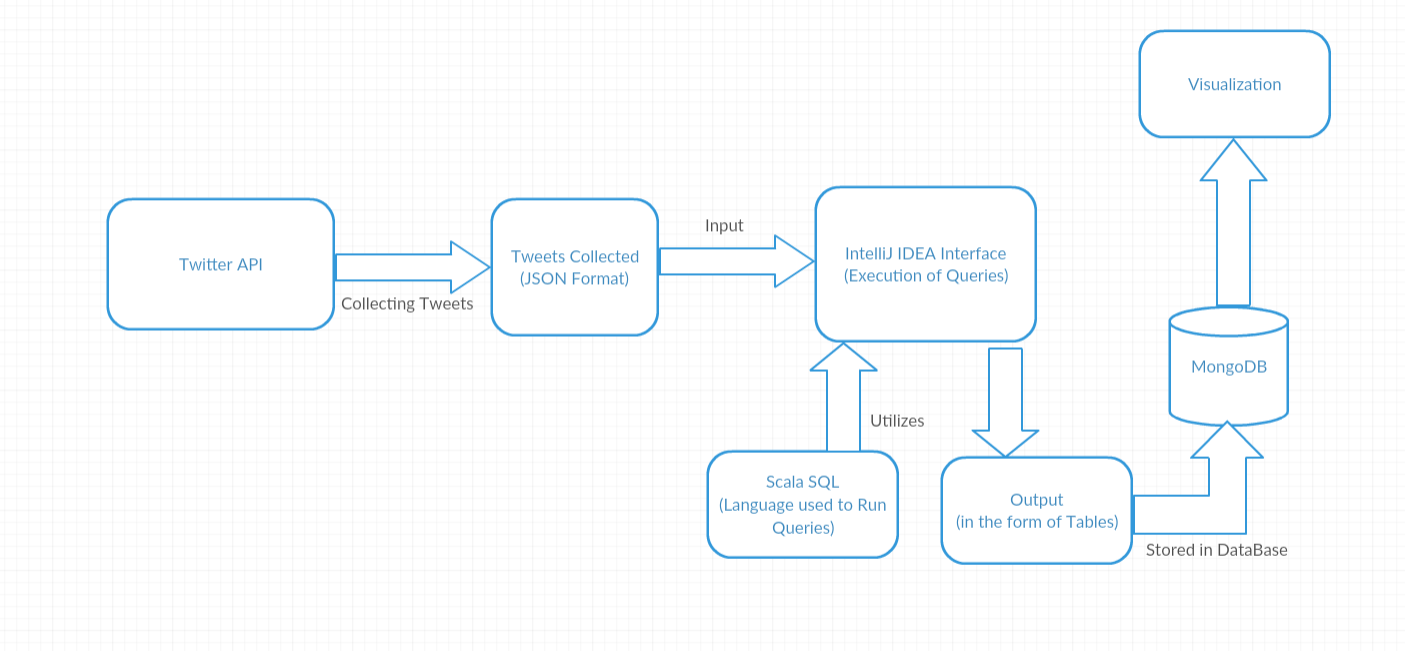
1. **Software Architecture & Design:**

Tweets on “Music” are extracted from twitter using a python code which uses Twitter consumer\_key, consumer\_secret, access\_token and access\_token\_secret which are developed using Twitter API. These tweets are collected in a text file in a JSON format. This JSON file is passed as input to the Scala program which consists of SQL Queries and implemented using Intellij IDEA and we get the output in the form of Tables and these are stored in MongoDB from which Visualization is performed

**Class Diagram** which describes the systems static structure is shown below,



**System Architecture Diagram** which describes the system structure and applications is shown below,

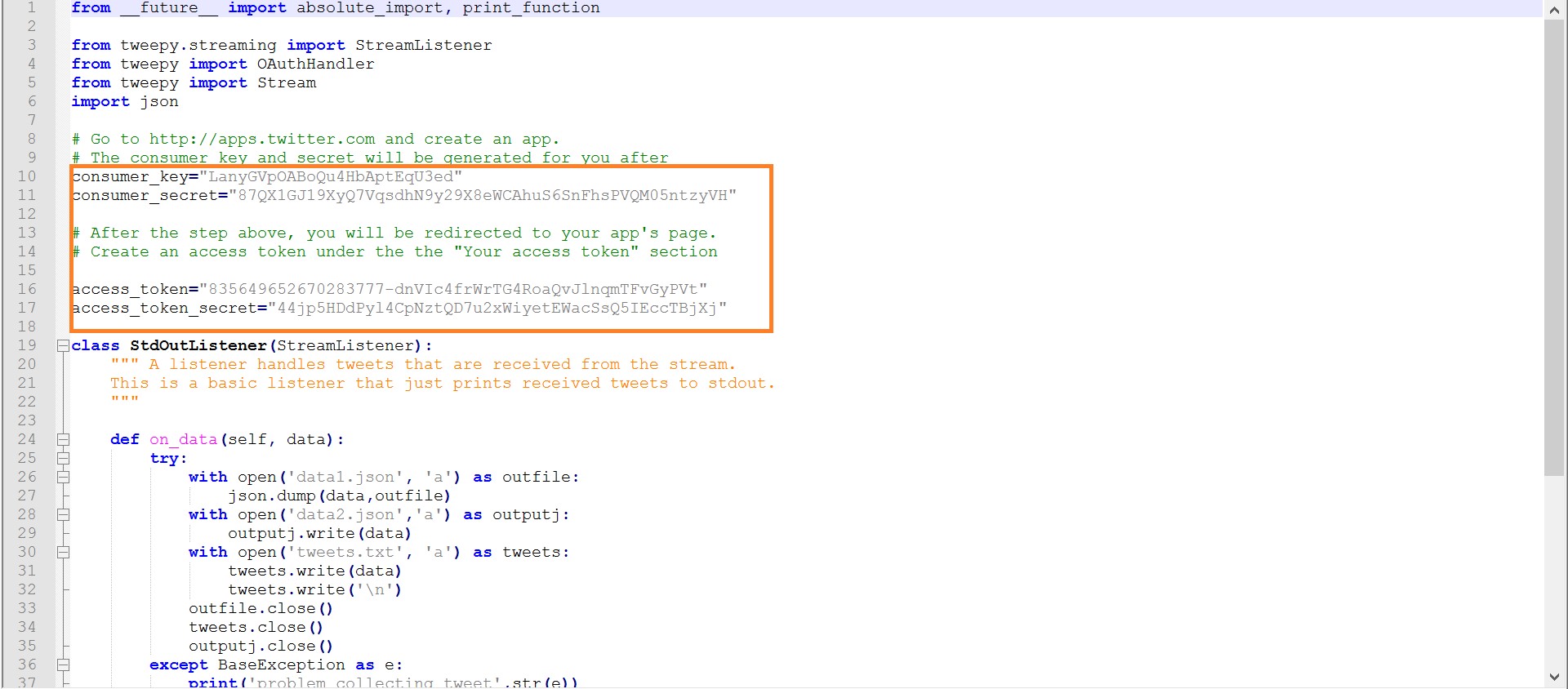


1. **System Description:**

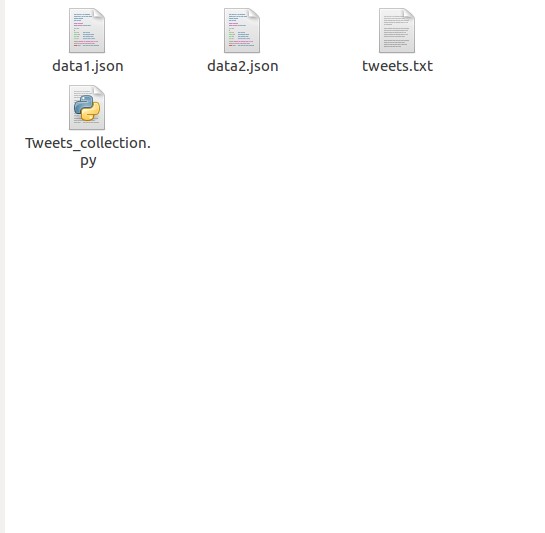
**Tweet Collection:**

* A Developer Account is created, twitter consumer secret, key and tokens are obtained. and the tweets are collected on a topic and the topic we chose is music. Close to 100k tweets are collected.
* The tweets collected are in the format of JSON data and text format. Now the obtained JSON format file is used to analysis.

This is the code for the tweet collection

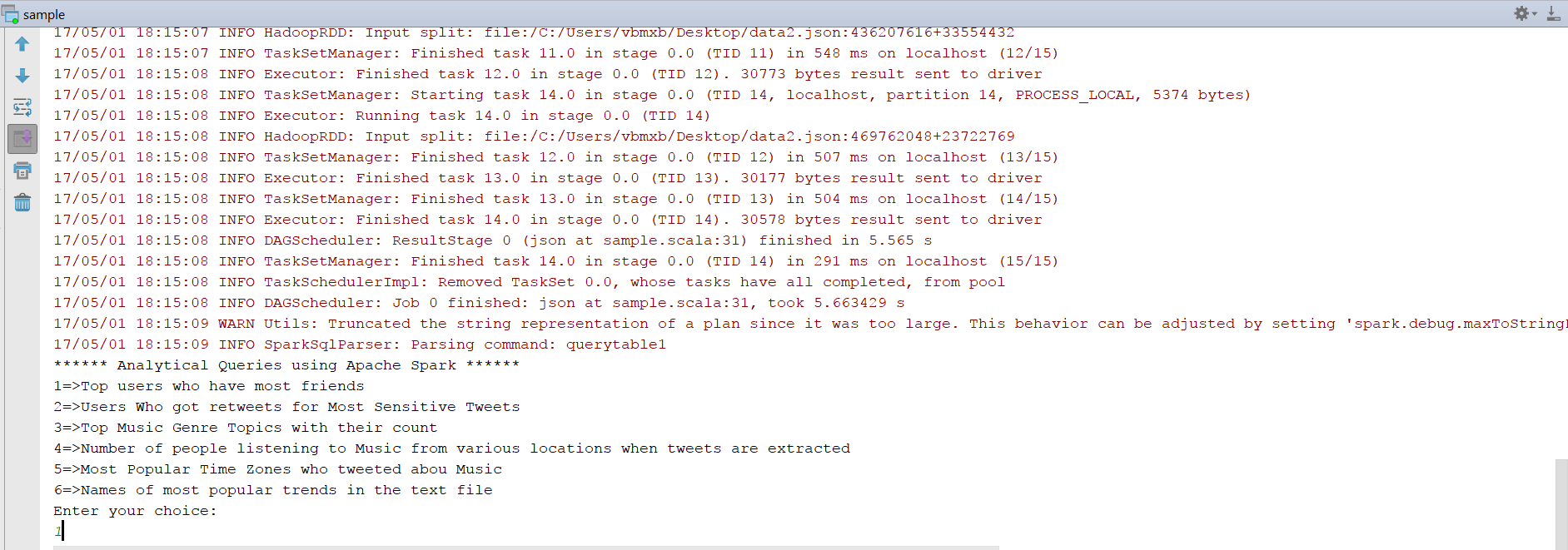


And these are the output files generated.



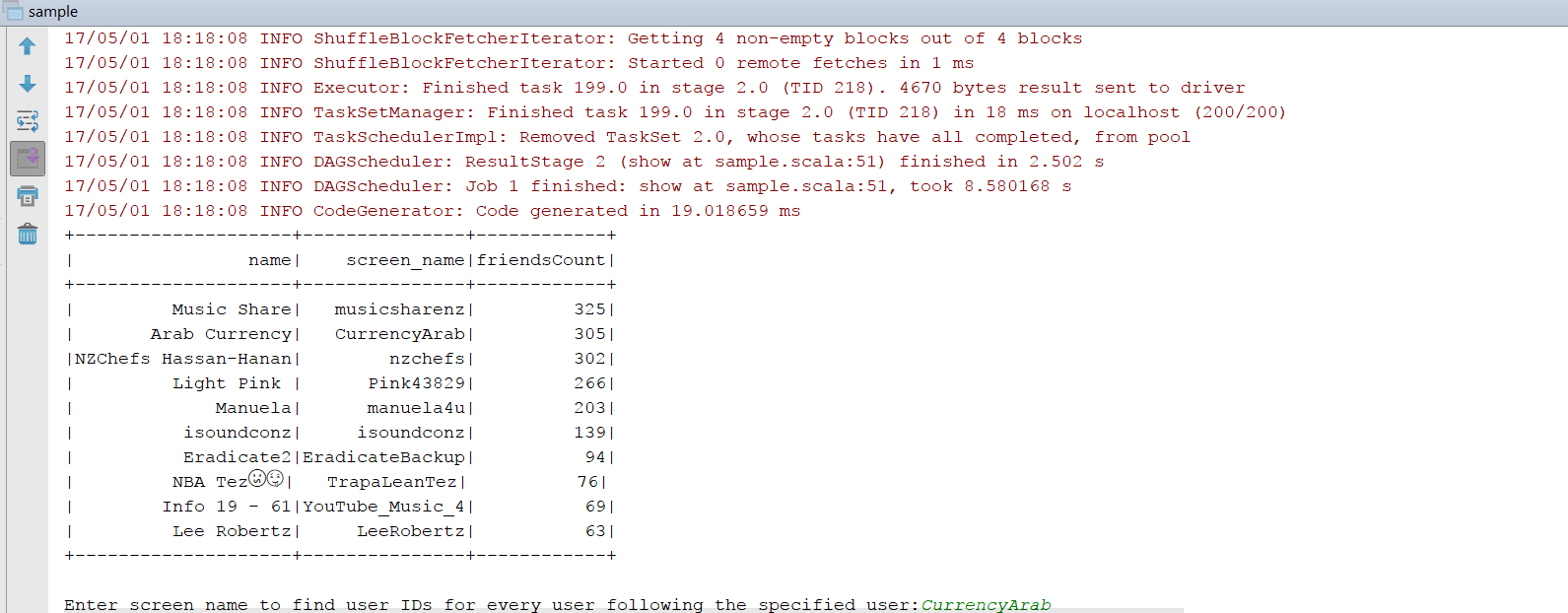
**Analytical Tasks:**

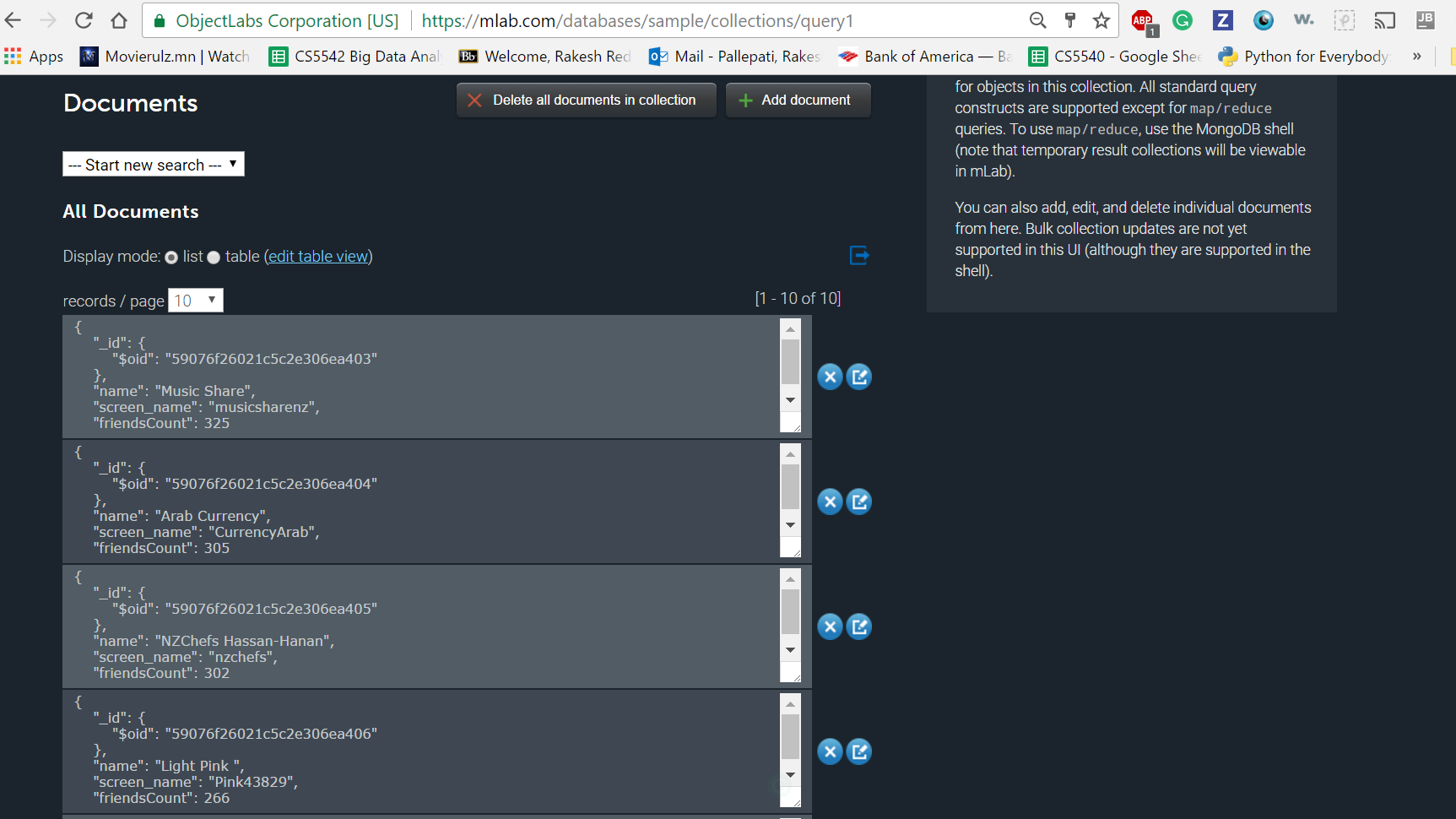
* Json file which has the tweets collected is given as input to the Scala program and implemented using IntelliJ
* Queries are implemented using this file
* Now, a switch case is used for the selection of Queries



1. **Data Frame Query-1:** Top Users who has Most Friends

In this Query We Analyze that, weather the user is Active in twitter or not by fetching the user’s followers count and the screen names of the users. This output is represented in the form of table which consists of “User Name”, “Screen Name” and “Followers Count”.



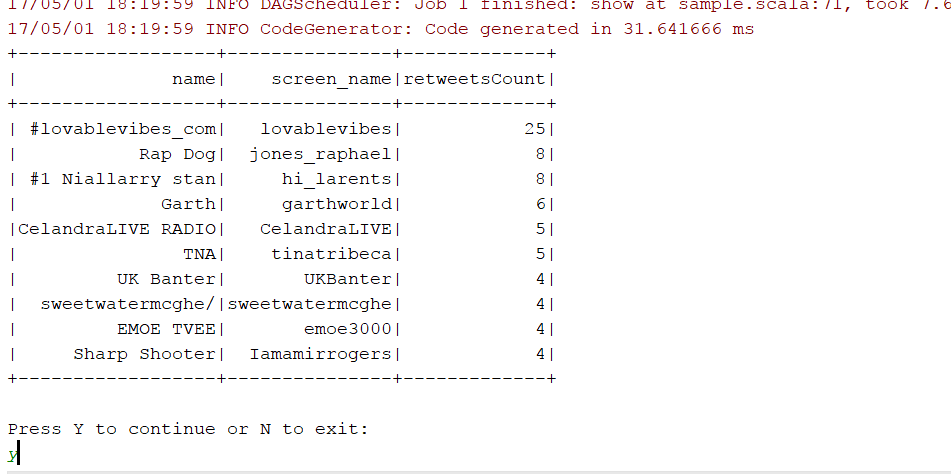
****

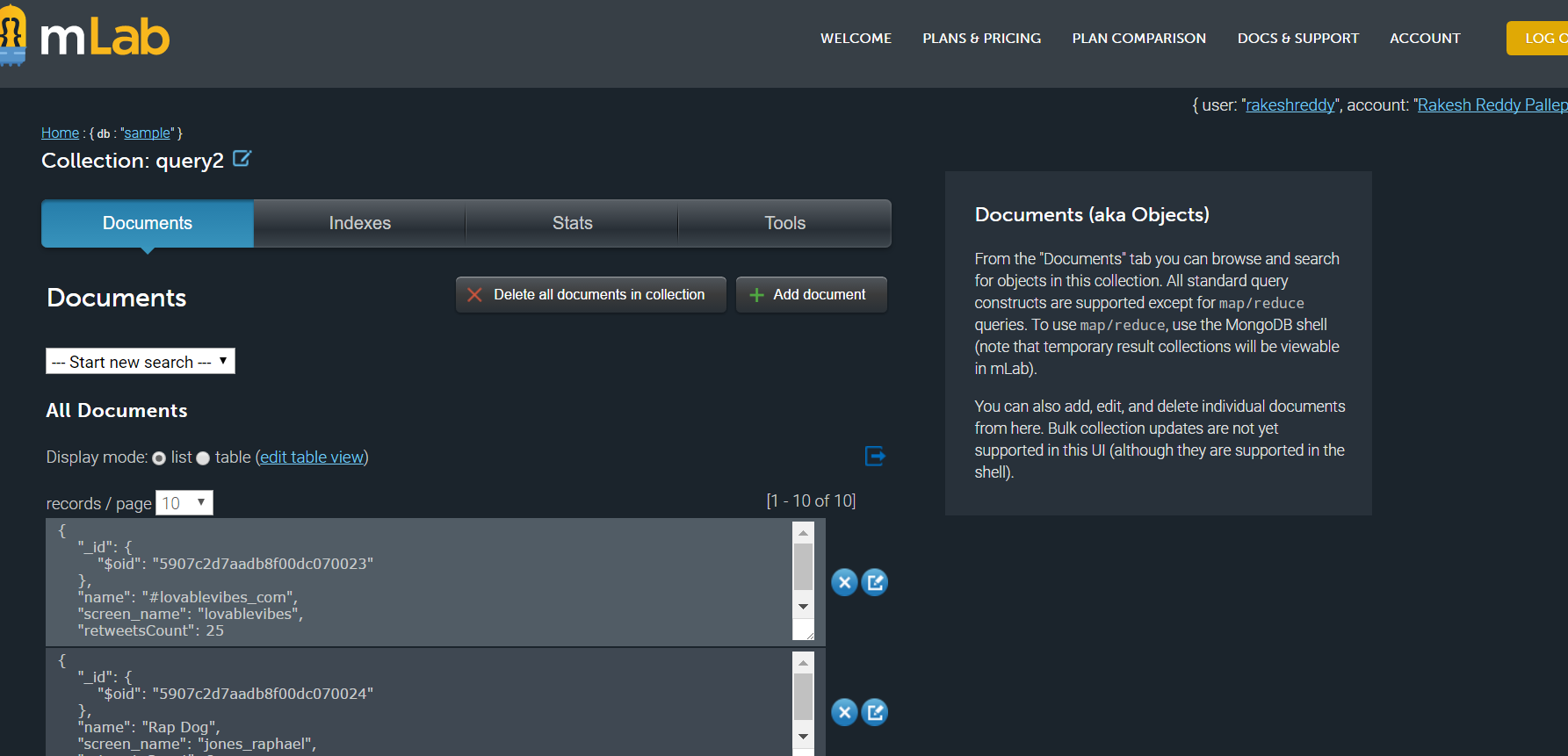
**Visualization Output:**

This is the Visualization Output generated form the output table, which is stored in MongoDB

1. **Data Frame Query-2:** Who Got Retweets for the most Sensitive Tweets

In this Query, we collect the Retweet count of the tweets which consists of Sensitive Information, that is we count the tweets consisting of “URL”, “Images”, and “Videos” information which are retweeted. This output is represented in the form of table which consists of “User Name”, “Screen Name” and “Retweet Count”.



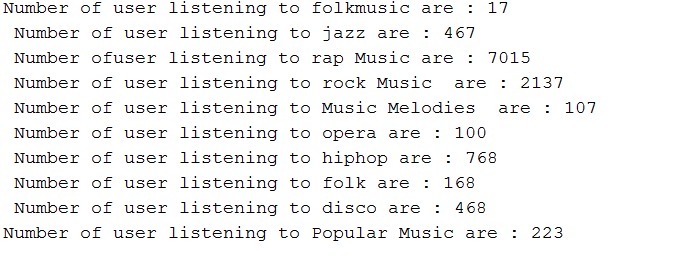


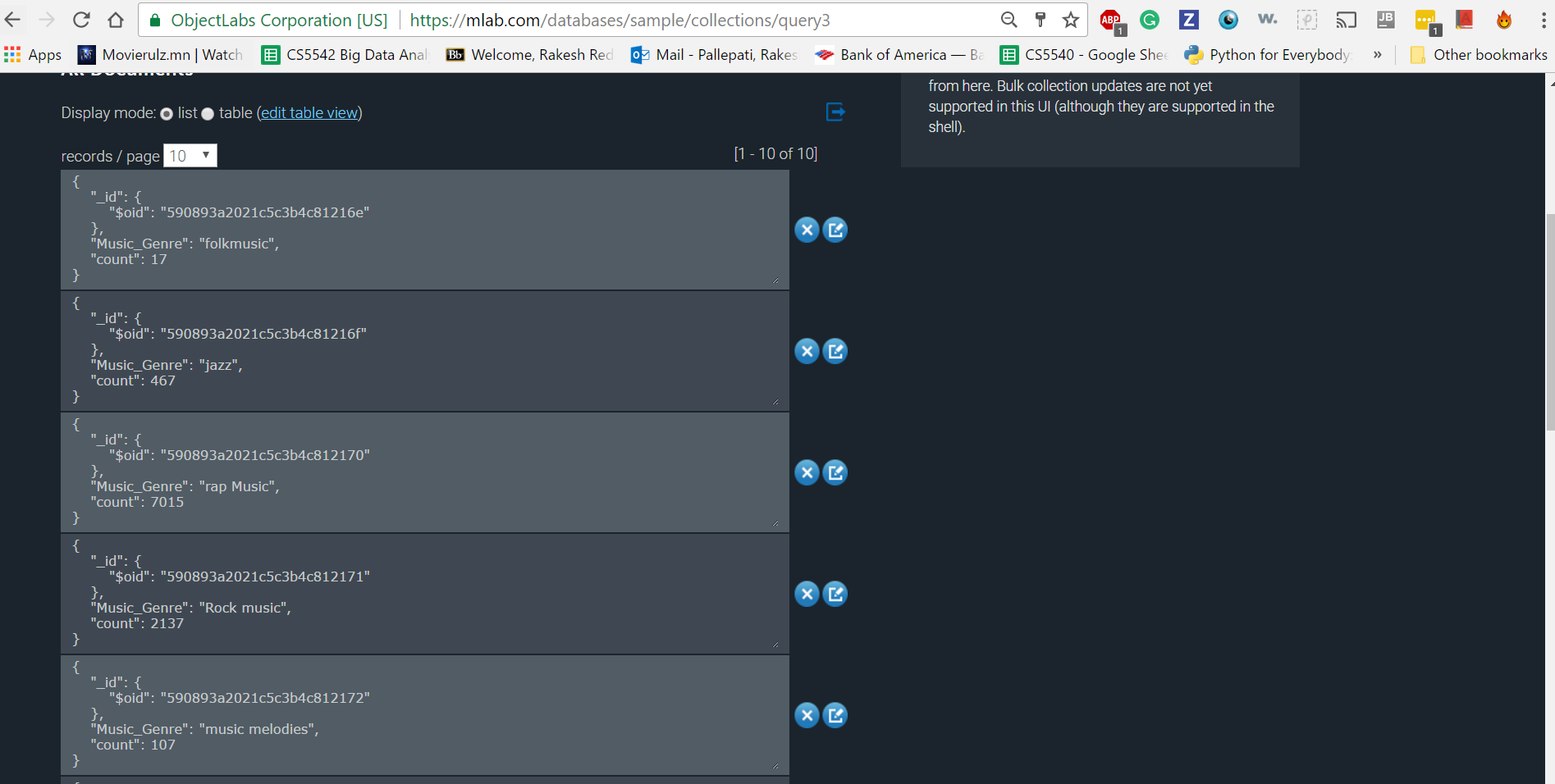
**Visualization Output:**

This is the Visualization Output generated form the output table, which is stored in MongoDB

1. **RDD Query-3:** Top Music Genre topics with their count

In this Query, we analyze the top music genre topics based on their count. The output of this Query is shown below



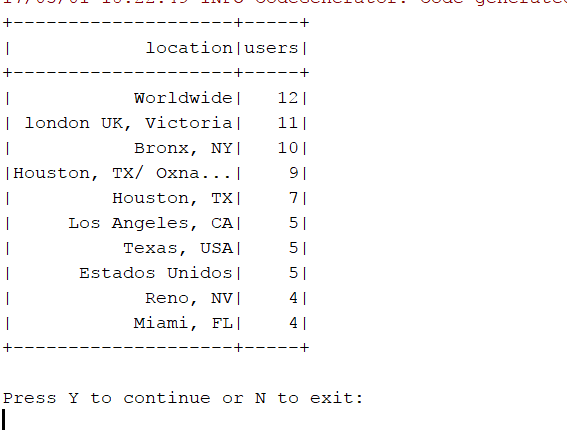


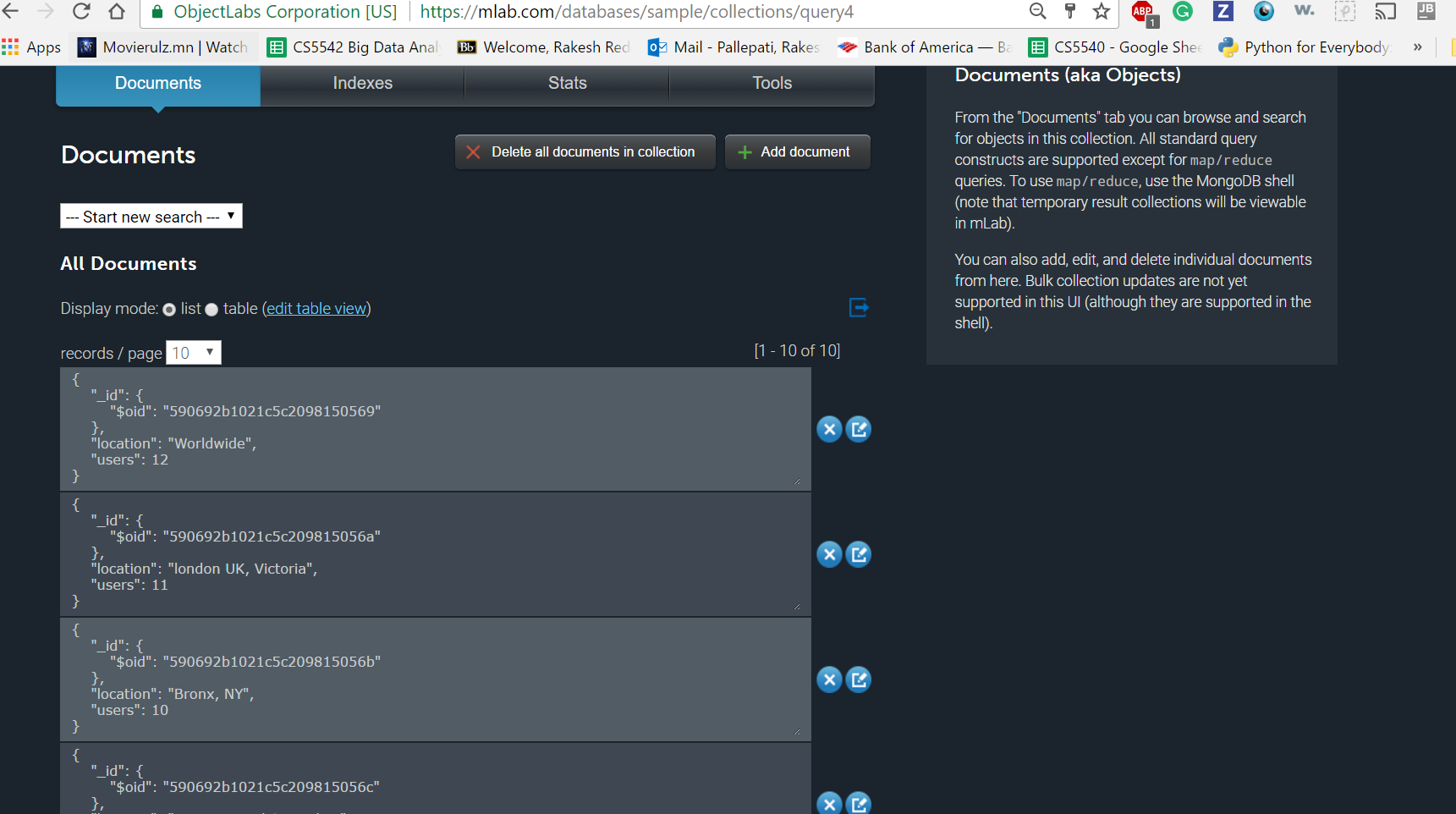
**Visualization Output:**

This is the Visualization Output generated form the output table, which is stored in MongoDB

1. **Data Frame Query-4:** Number of people listening to music from various locations when tweets are extracted

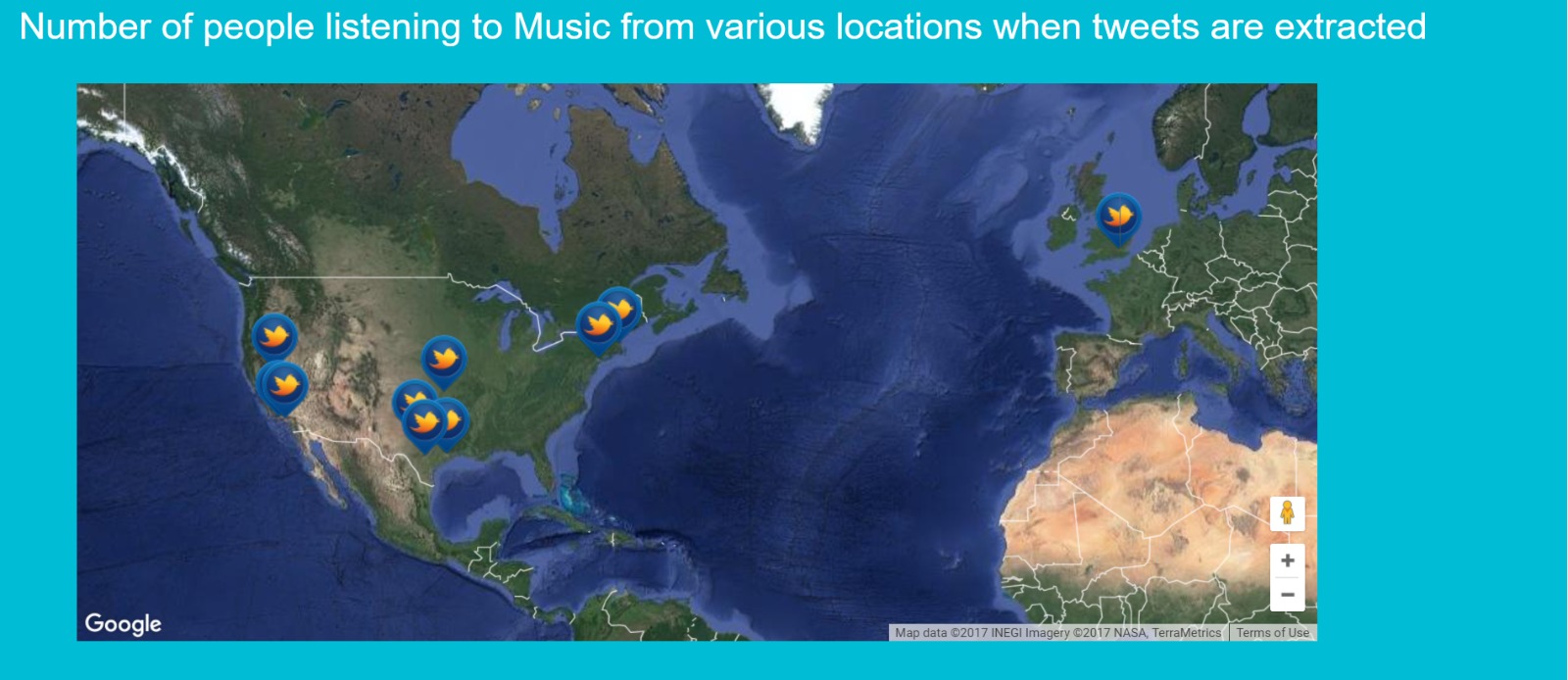
In this Query, we analyze the count if people listening to music from different locations when the tweets are extracted. The result of this query consists of “Location” and “Number of Users” when the tweets are generated.





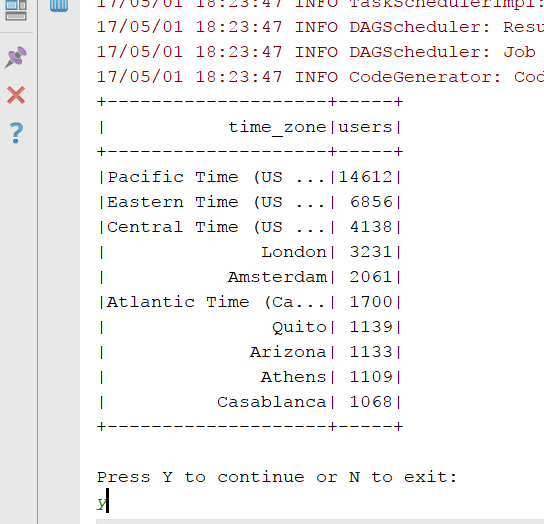
**Visualization Output:**

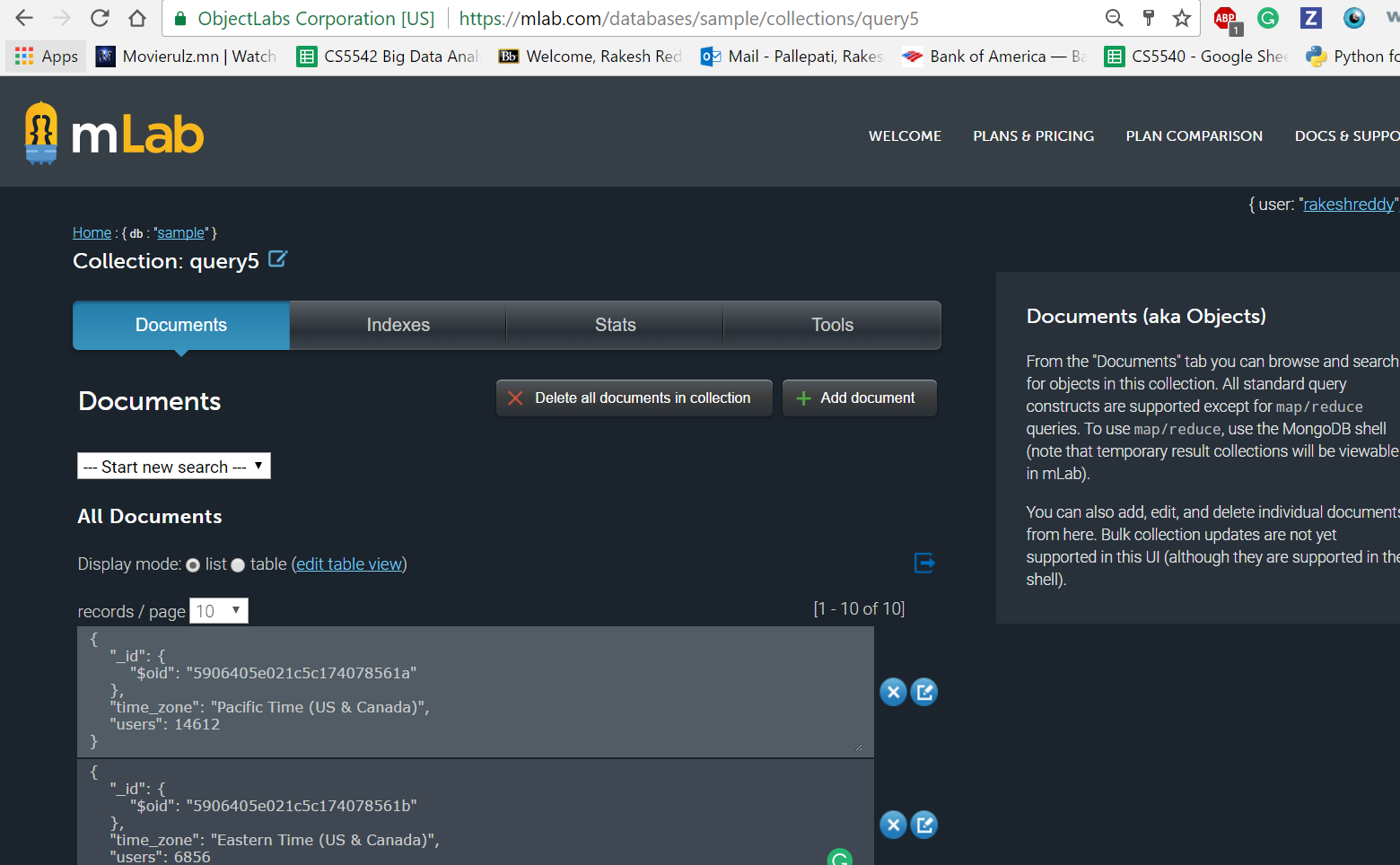
This is the Visualization Output generated form the output table, which is stored in MongoDB



1. **Data Frame Query-5:** Most popular time zones who tweeted about music

In this Query we analyze the top time zones who tweeted about music based on the tweets.json file. The result of this Query consists of “time zone” and “number of users”

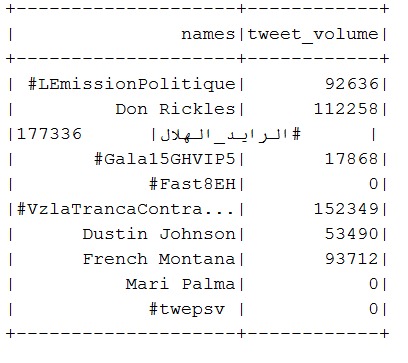


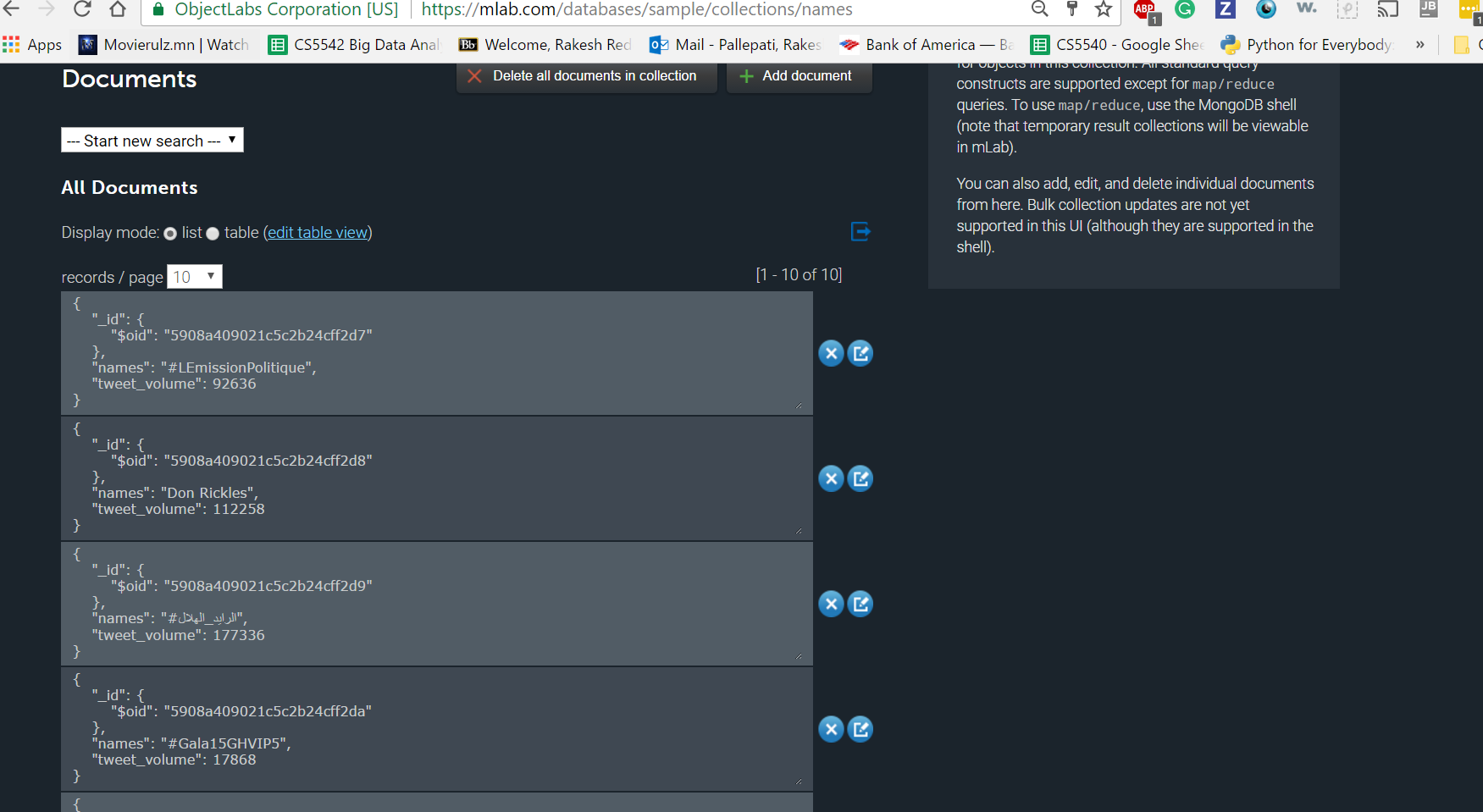
****

**Visualization Output:**

This is the Visualization Output generated form the output table, which is stored in MongoDB

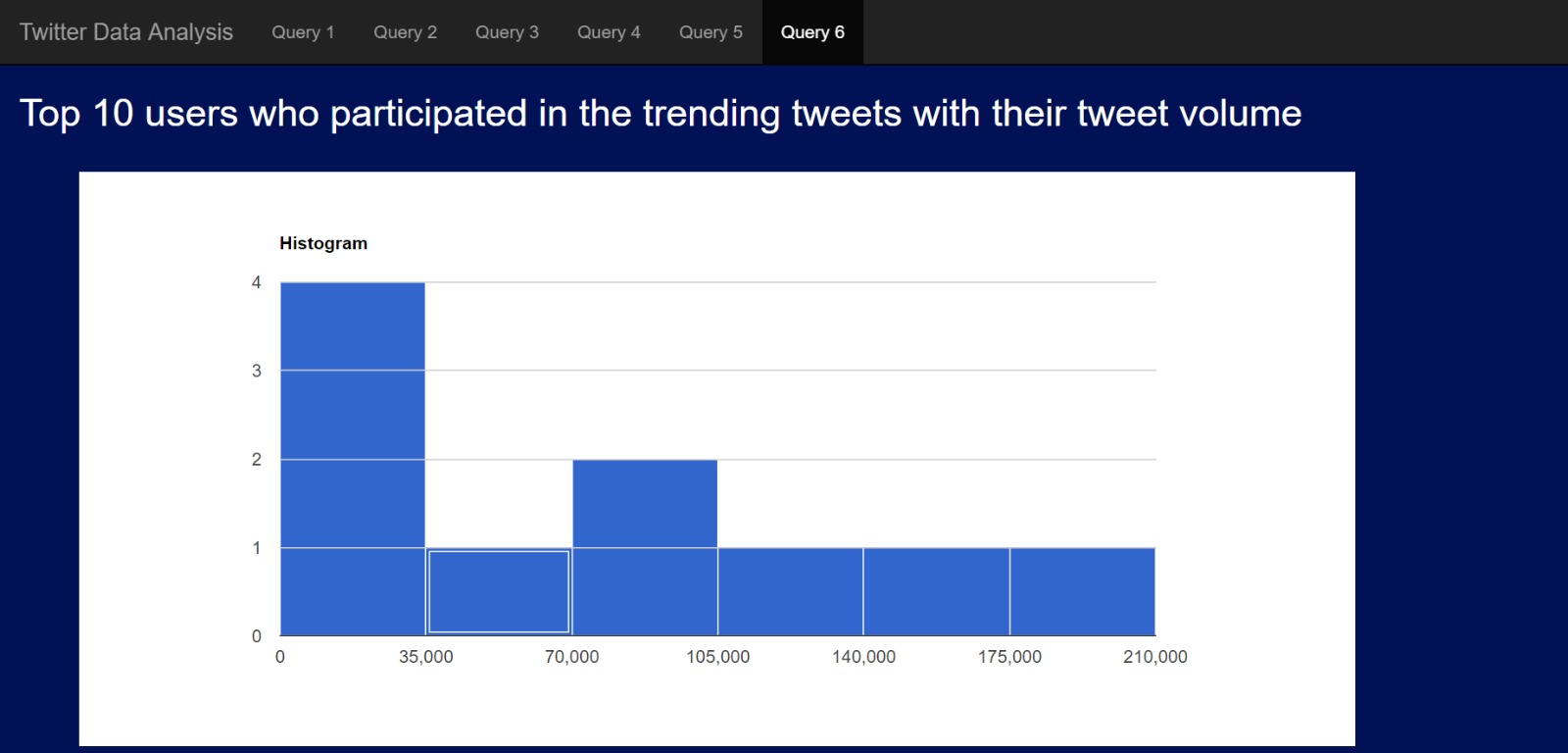
1. **RDD Query-6:** Names of most popular trends in the text file

In this Query, we analyze the popular trends in the “trends.txt” file. This is a RDD Query for which result is displayed as below



**Visualization Output:**

This is the Visualization Output generated form the output table, which is stored in MongoDB



**References:**

* [https://apps.twitter.com](https://apps.twitter.com/)
* [http://www.tutorialspoint.com](http://www.tutorialspoint.com/)
* https://www.jetbrains.com/idea/