Big Data Programming

## CSEE5590/490

**Module 2 Lab 2**

**Report**

Submitted By:

Sneha Mishra:

Class ID: 11

Aditya Soman

Class ID: 19

Team: 12  
Professor: Yugyung Lee

Name: Sneha Mishra  
Class ID: 11  
Email: [smccr@mail.umkc.edu](mailto:smccr@mail.umkc.edu)  
[MyGitHub](https://github.com/SnehaMishra28/BigData_Programming_Summer2018/wiki/Module-2:-Lab-%232)

Technical Partner:  
Name: Aditya Soman  
Class ID: 19  
Email: [as9f3@mail.umkc.edu](mailto:as9f3@mail.umkc.edu)  
[GitHub](https://github.com/adityasomann/Big-Data-Programming)

YouTube Link explaining the Lab work can be found [here](https://www.youtube.com/watch?v=S63bc4O5HhA&feature=youtu.be)  
The report for the Lab work is [here](https://github.com/SnehaMishra28/BigData_Programming_Summer2018/wiki/Module-2:-Lab-%232)  
The source code for this lab work can be found [here](https://github.com/SnehaMishra28/BigData_Programming_Summer2018/tree/master/Lab/Mod2-Lab%232/Source/Mod2_Lab4_Source)  
The available datasets formats can be found [here](https://github.com/SnehaMishra28/BigData_Programming_Summer2018/tree/master/Lab/Mod2-Lab%232/Source/Dataset)

**Objective**

Understanding Spark Classification, Spark Streaming and Spark Graphx Task.

**Features**

1. Use of Classification Algorithms such as Naïve Bayes, Decision Tree, Random Forest for attribute classification.
2. Report the Confusion matrix, Accuracy based on FMeasure, Precision & Recall for all the algorithms.
3. Reason why one of algorithms out performs the rest.
4. Perform Word-Count on Twitter Streaming Data using Spark.
5. Perform Page Rank on given Dataset.
6. State importance of using graphx on the chosen dataset.

# Steps:

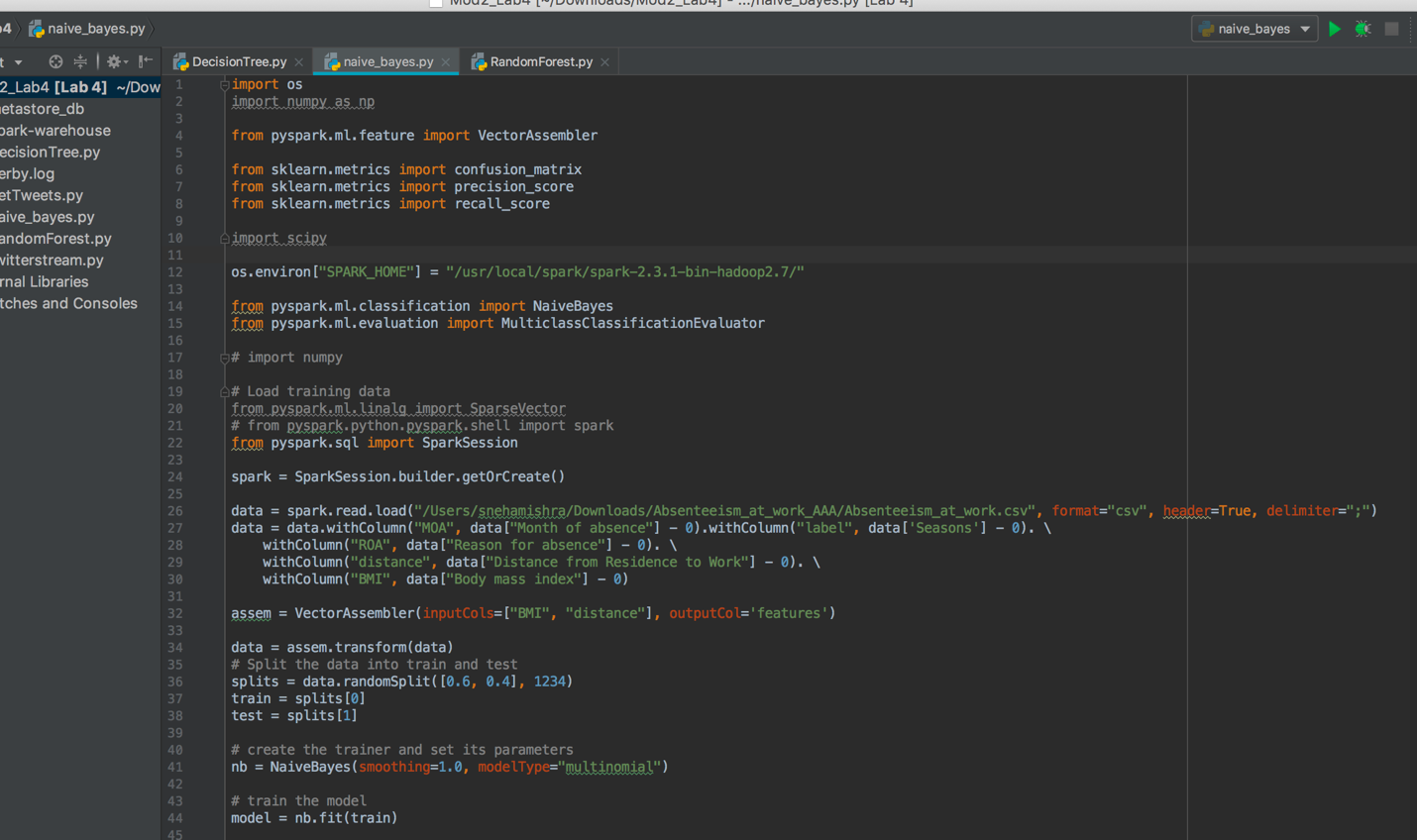
## Part 1: Spark Classification Task

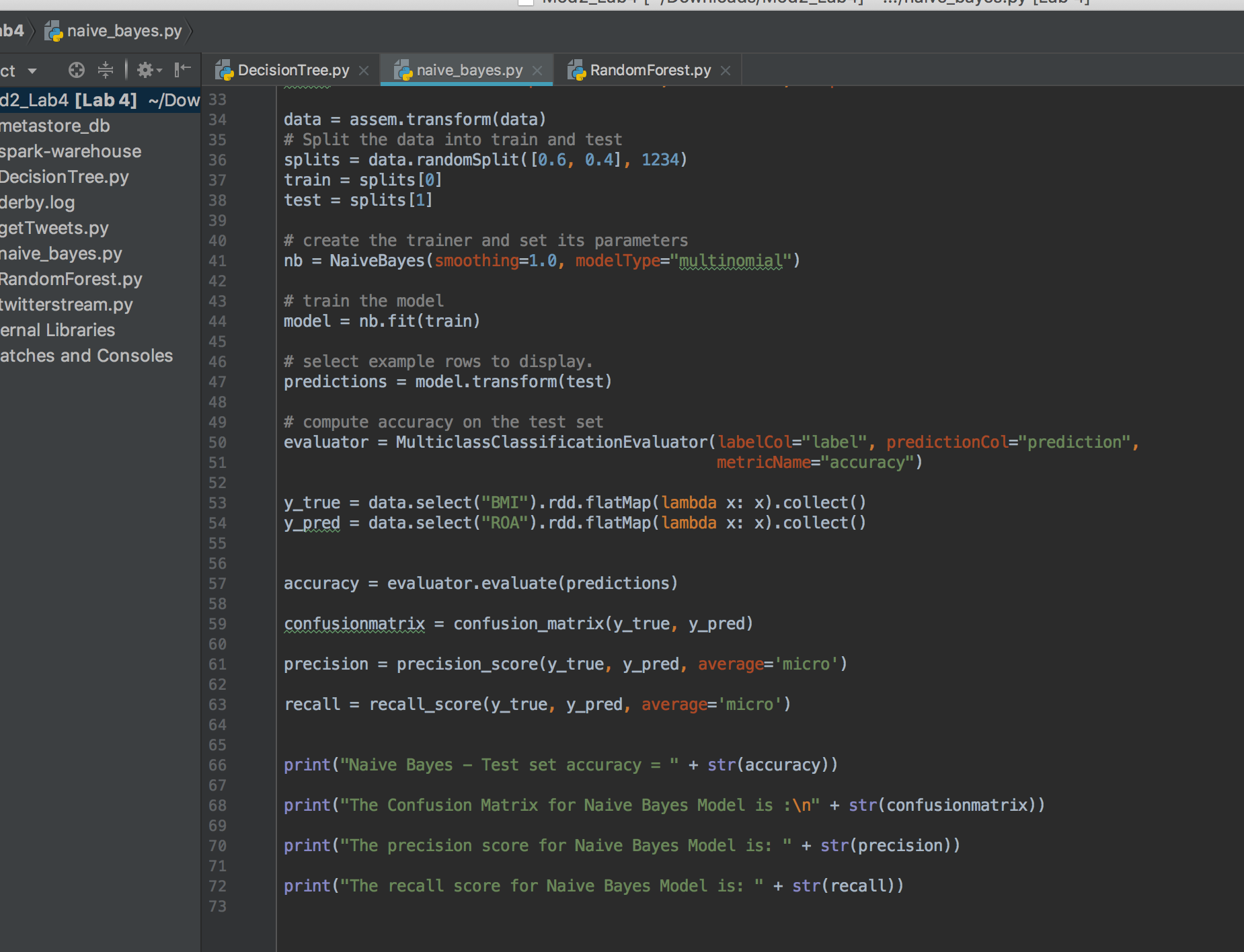
This task contains working on 3 algorithms namely:

### 1. Naïve Bayes:

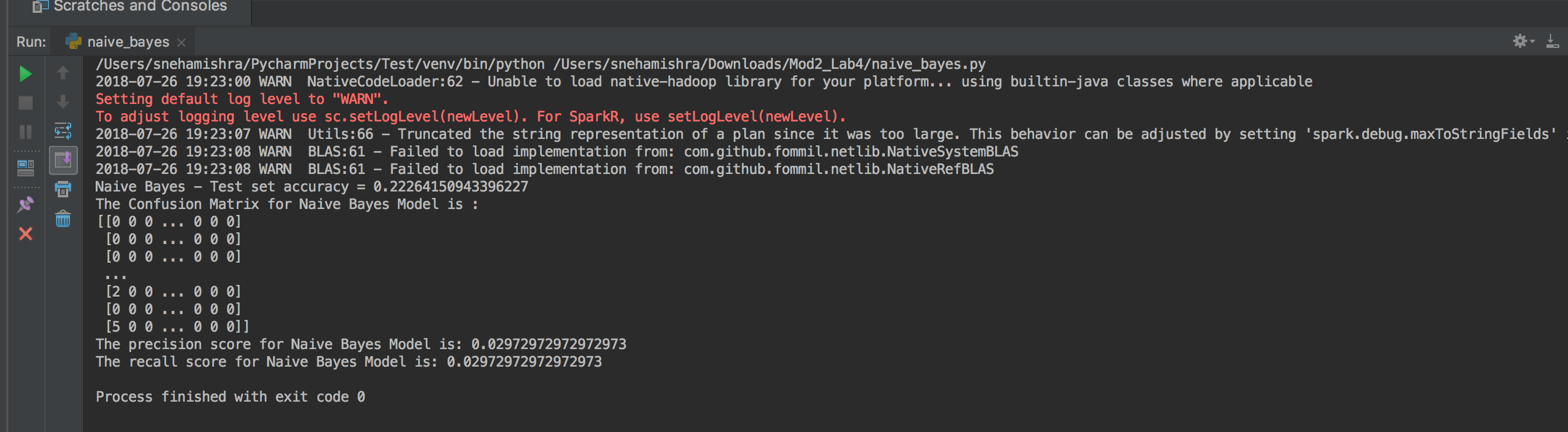
It is a classification technique based on Bayes’ theorem. Naïve Bayes classifier assumes that the presence of a particular feature in a class is unrelated to the presence of any other feature.

#### Code for the Algorithm:



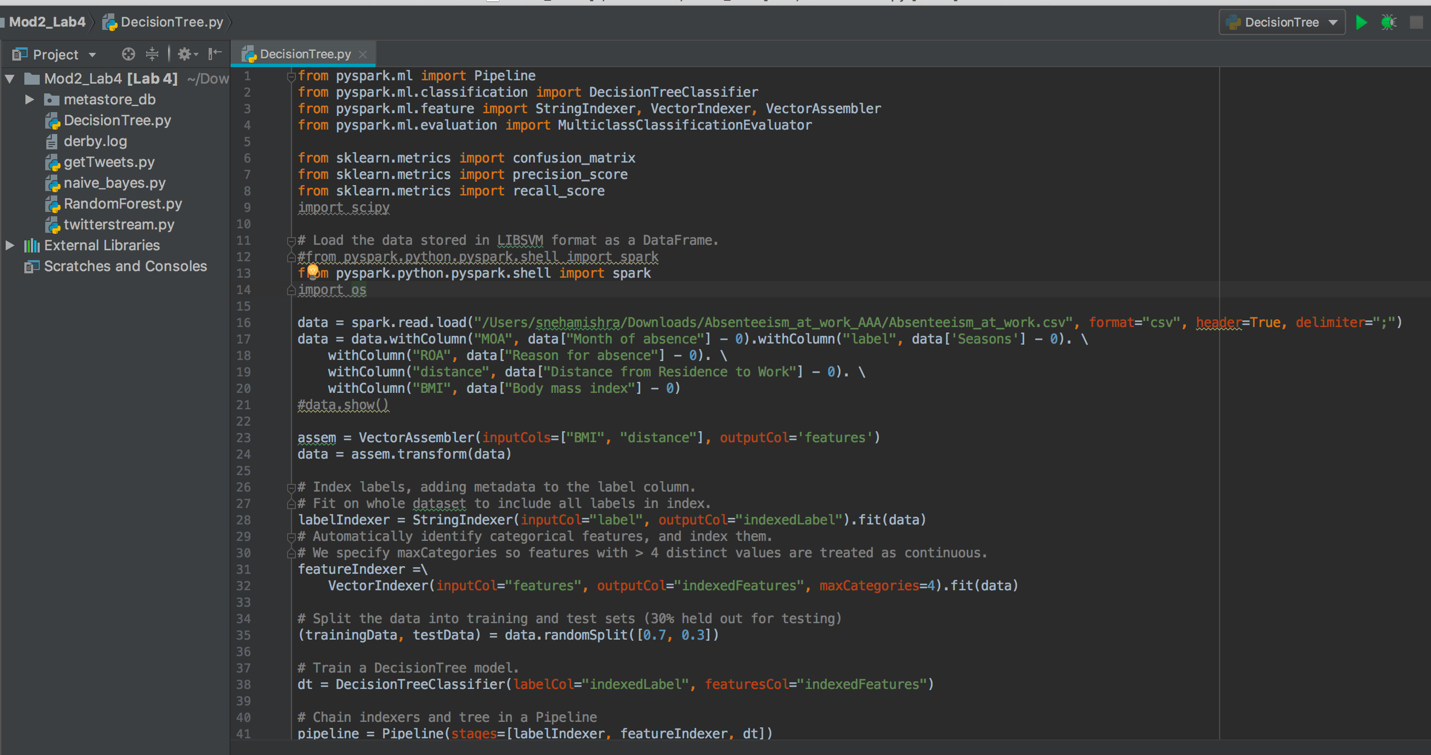


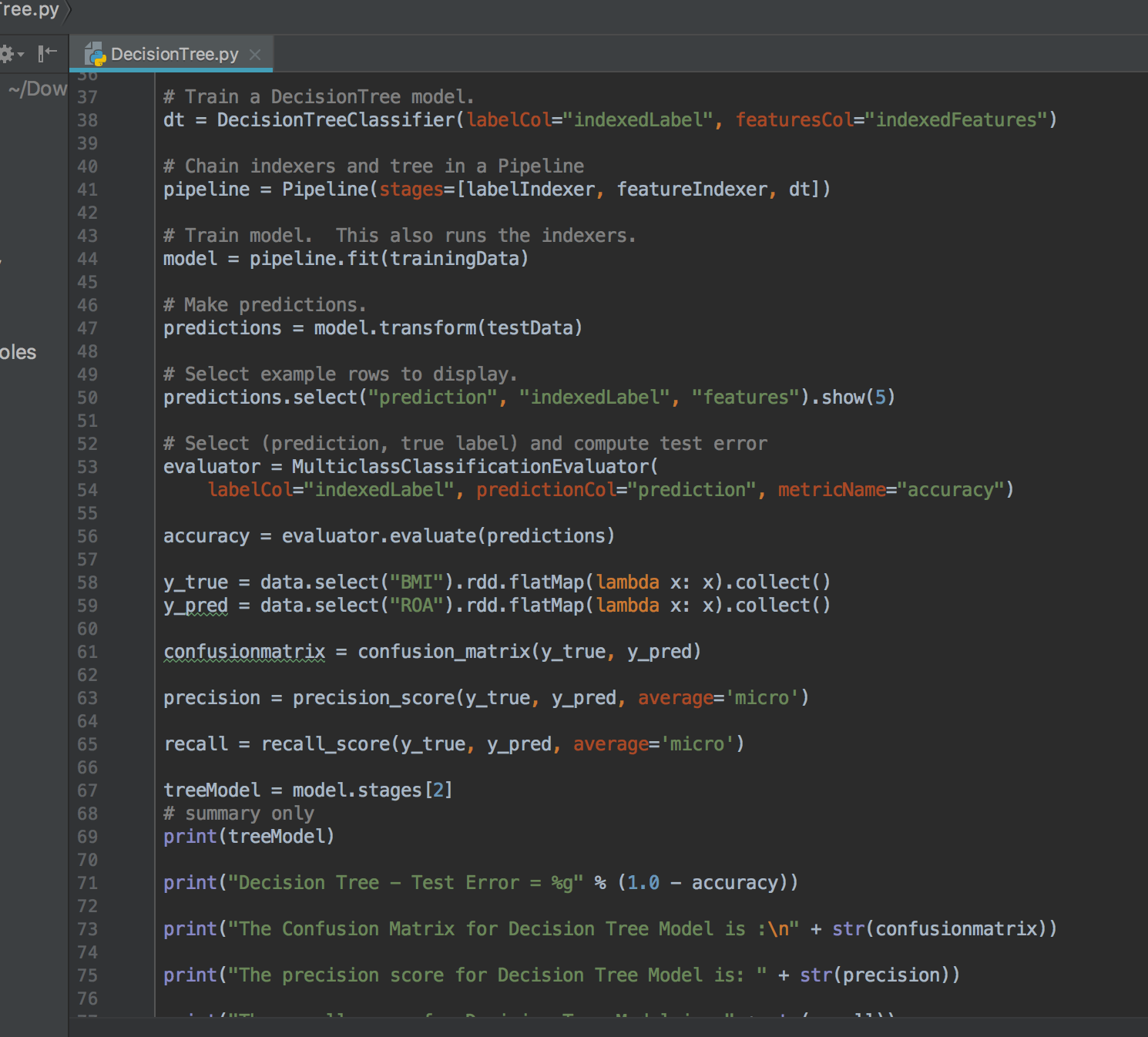
#### Output after running the Algorithm:



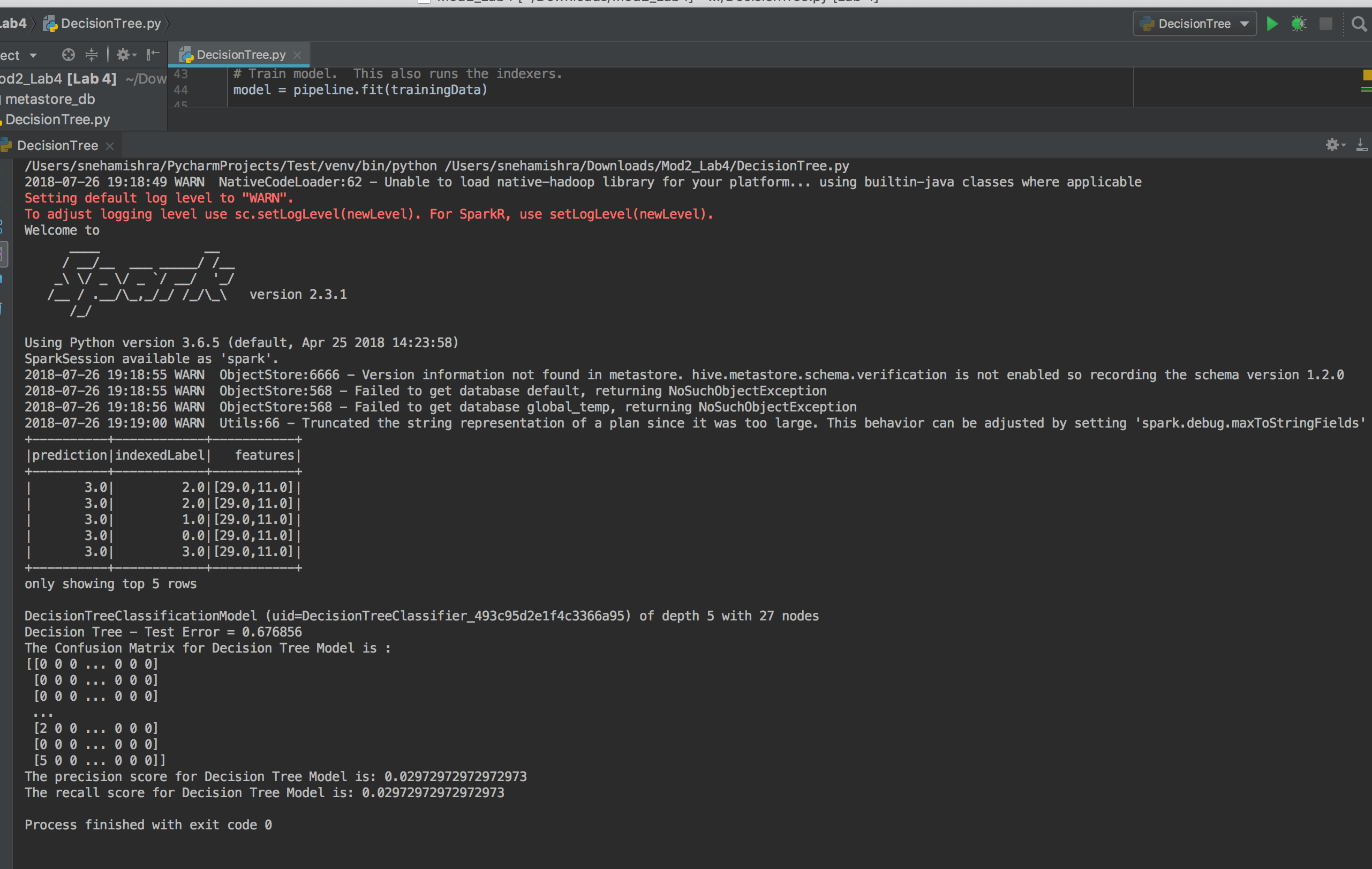
### 2. Decision Tree:

#### Code for the Algorithm:



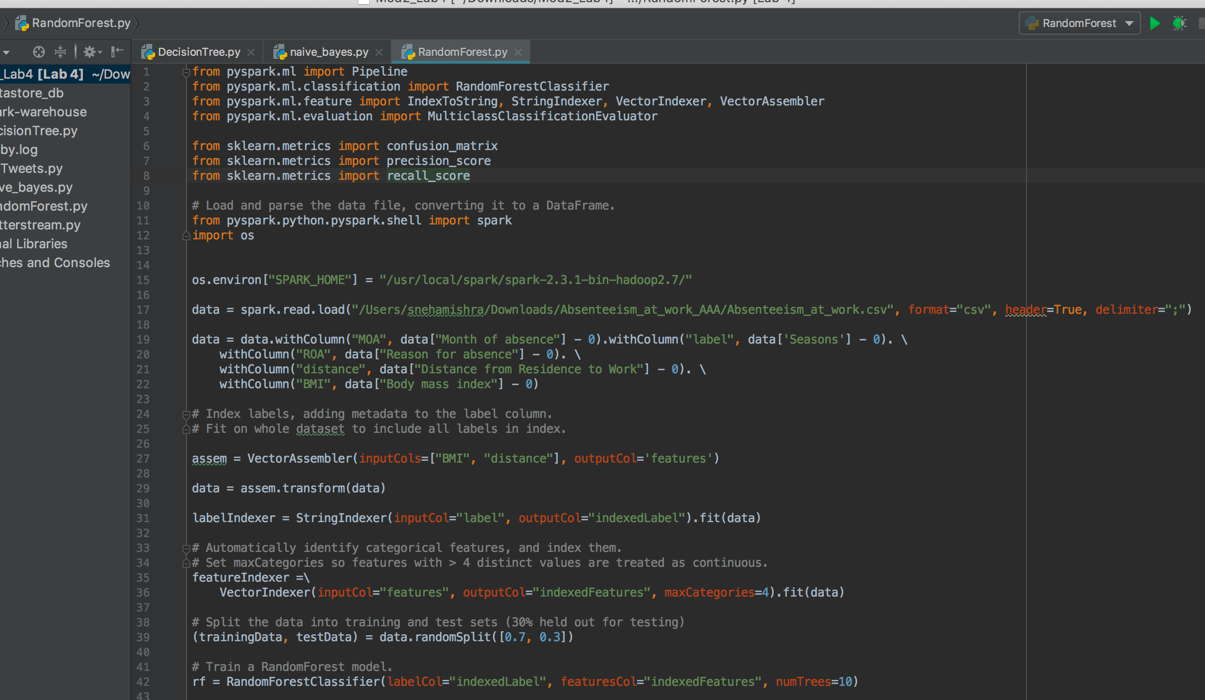


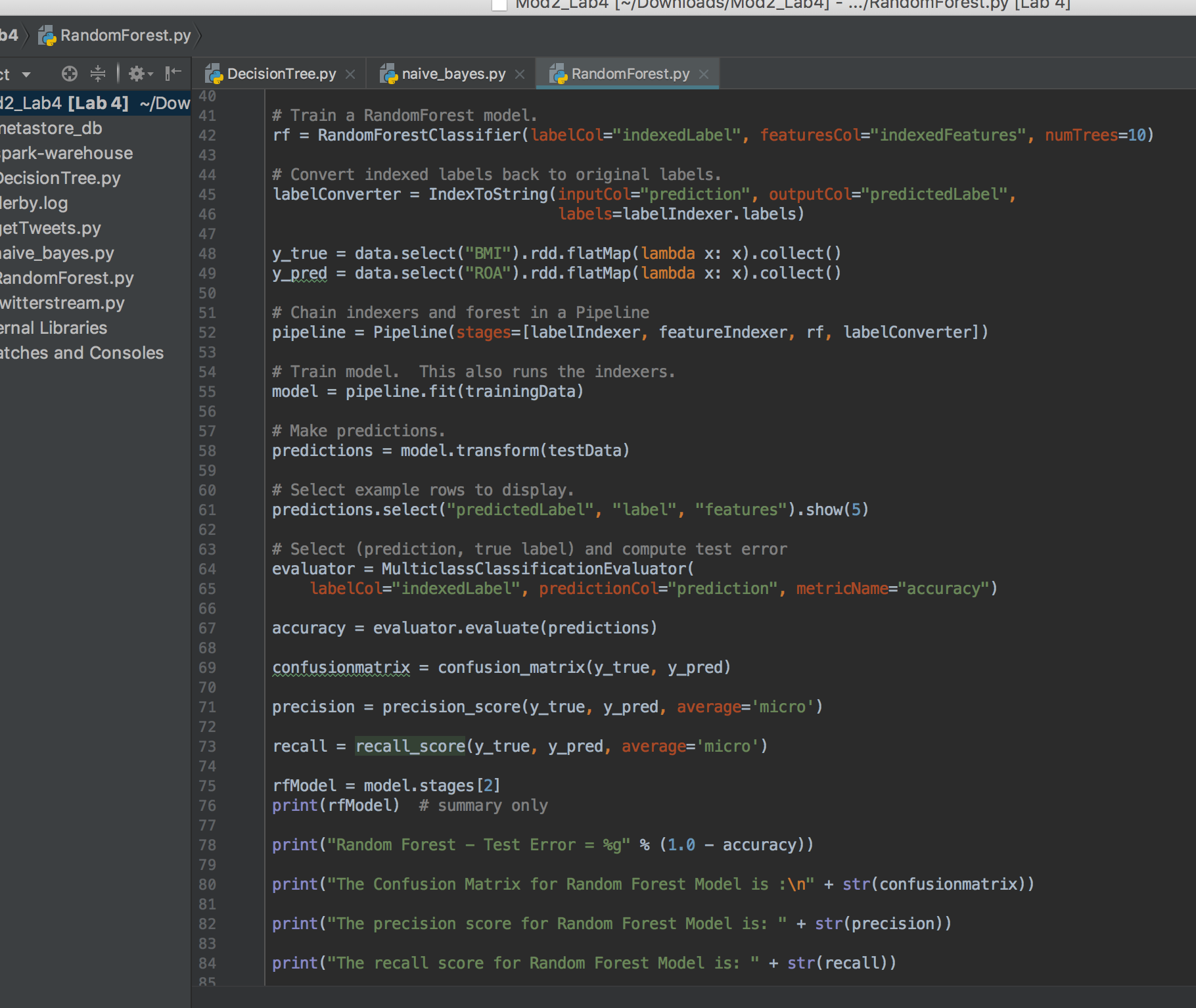
#### Output after running the Algorithm:



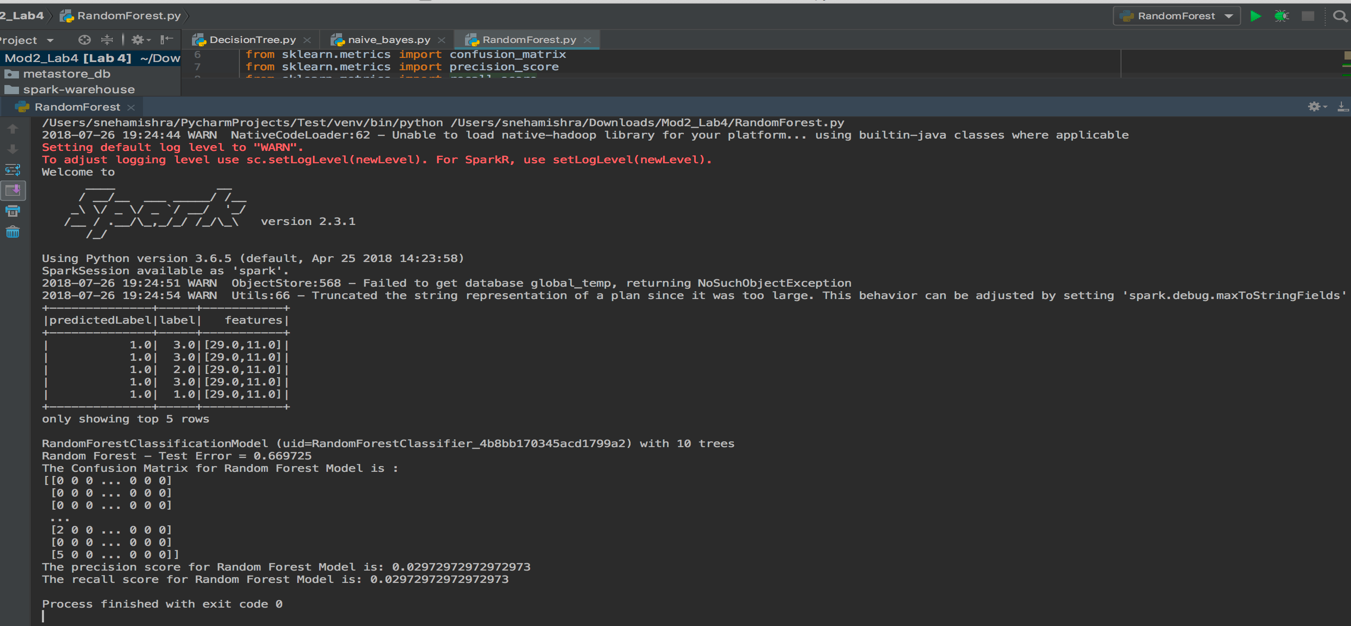
### 3. Random Tree:

#### Code for the Algorithm:





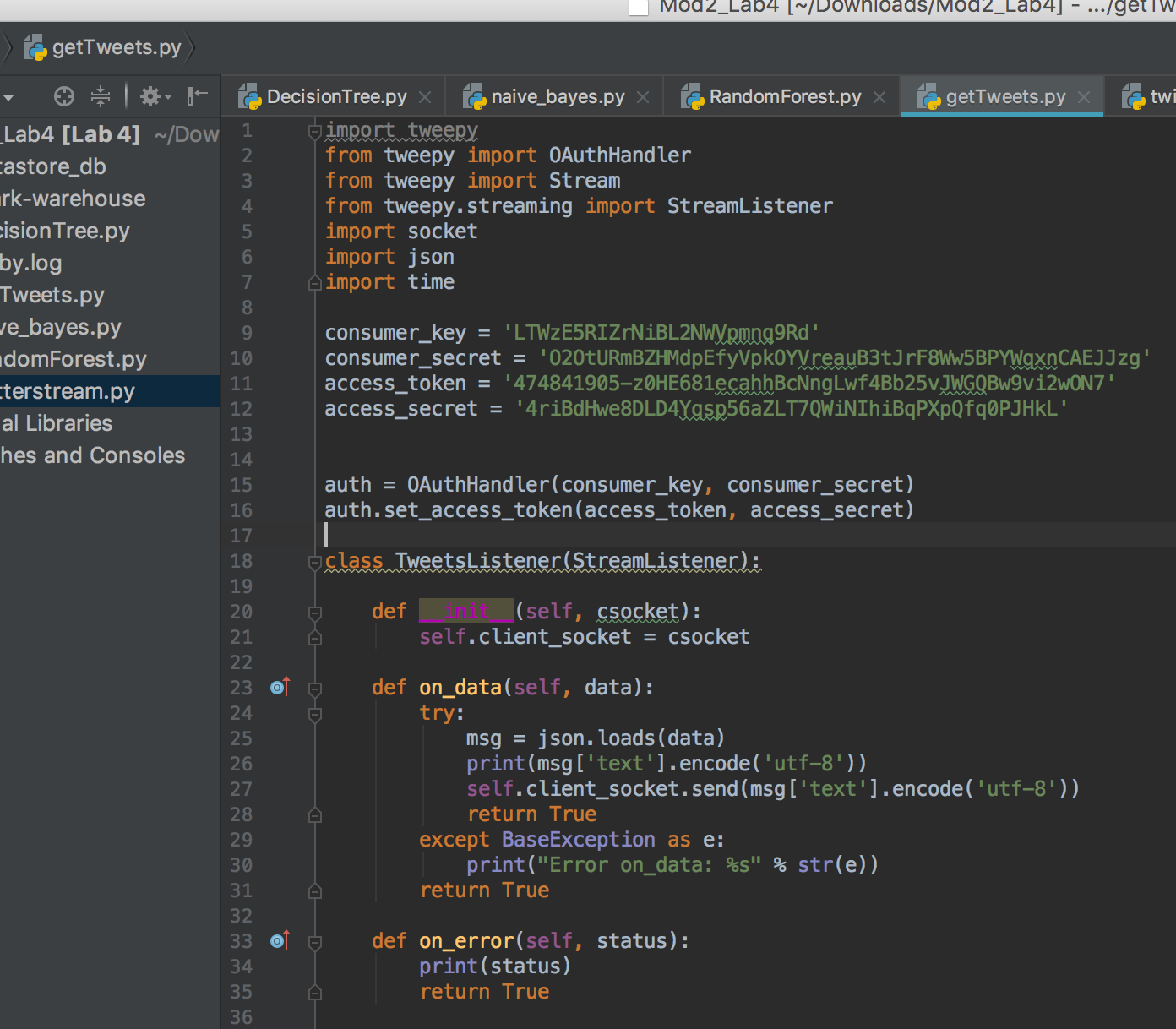
#### Output after running the Algorithm:

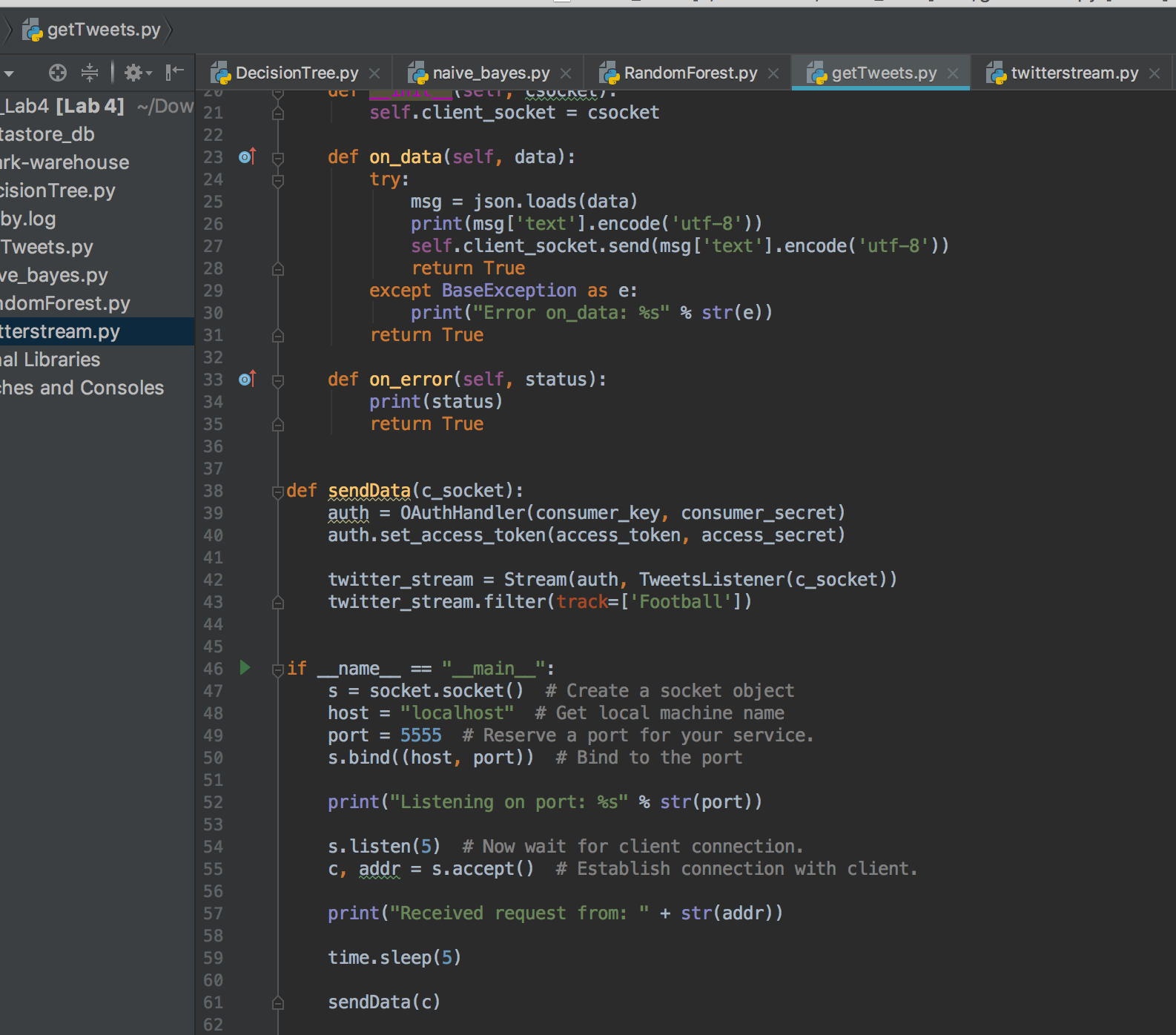


## Part 2: Spark Streaming Task

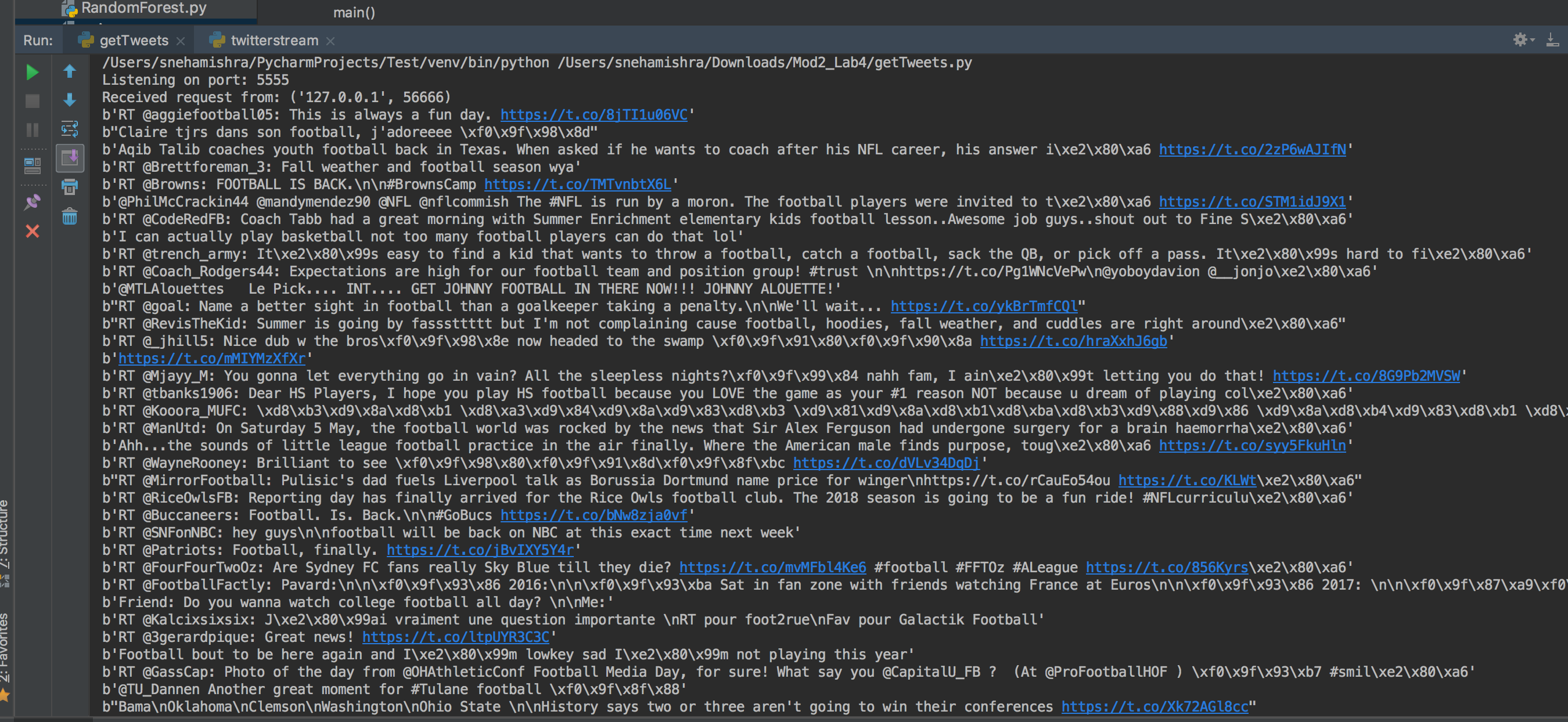
In this task we perform Word-Count on Twitter Streaming Data using Spark. First we get the Twitter data and then we stream it.

#### Collecting Tweets code:

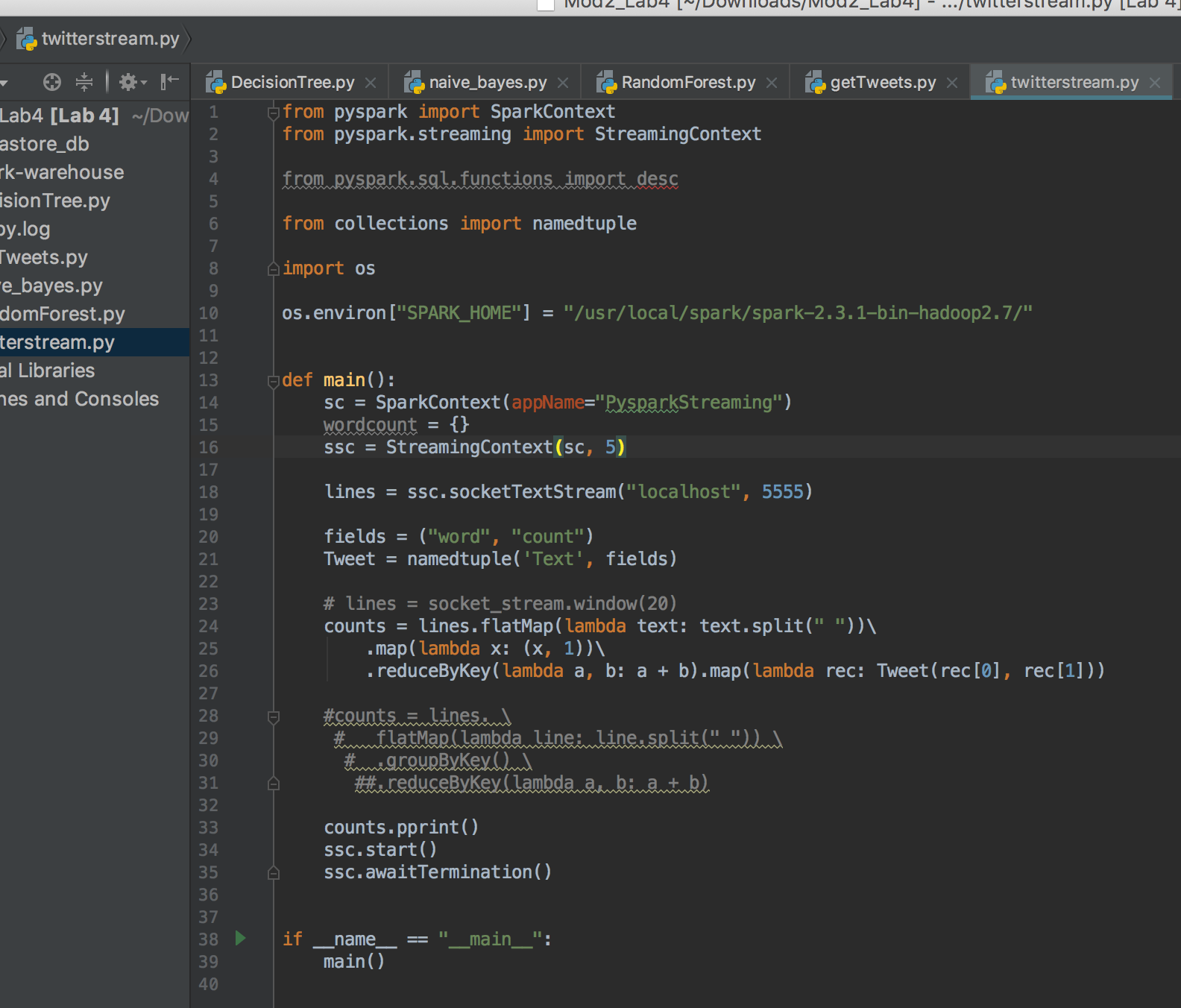




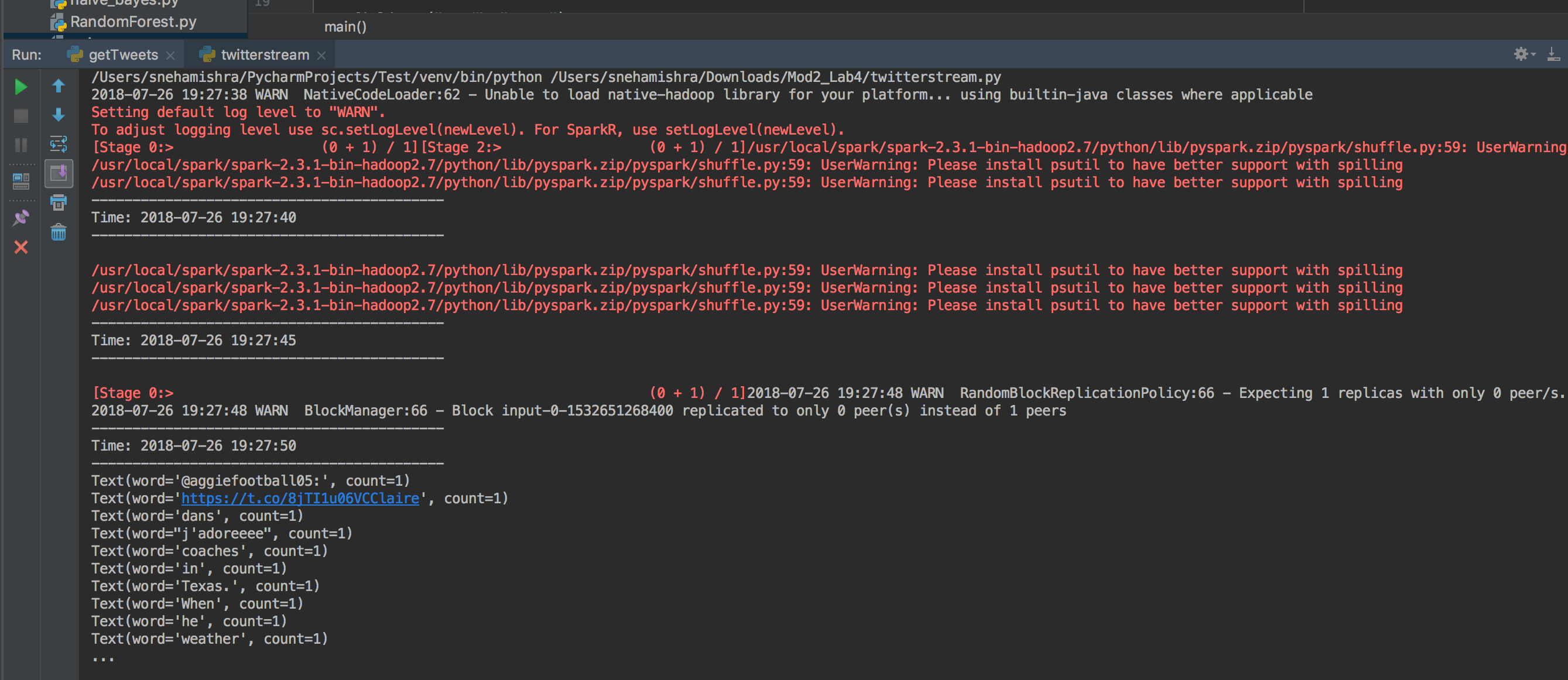
#### Output of Collecting Tweets code:

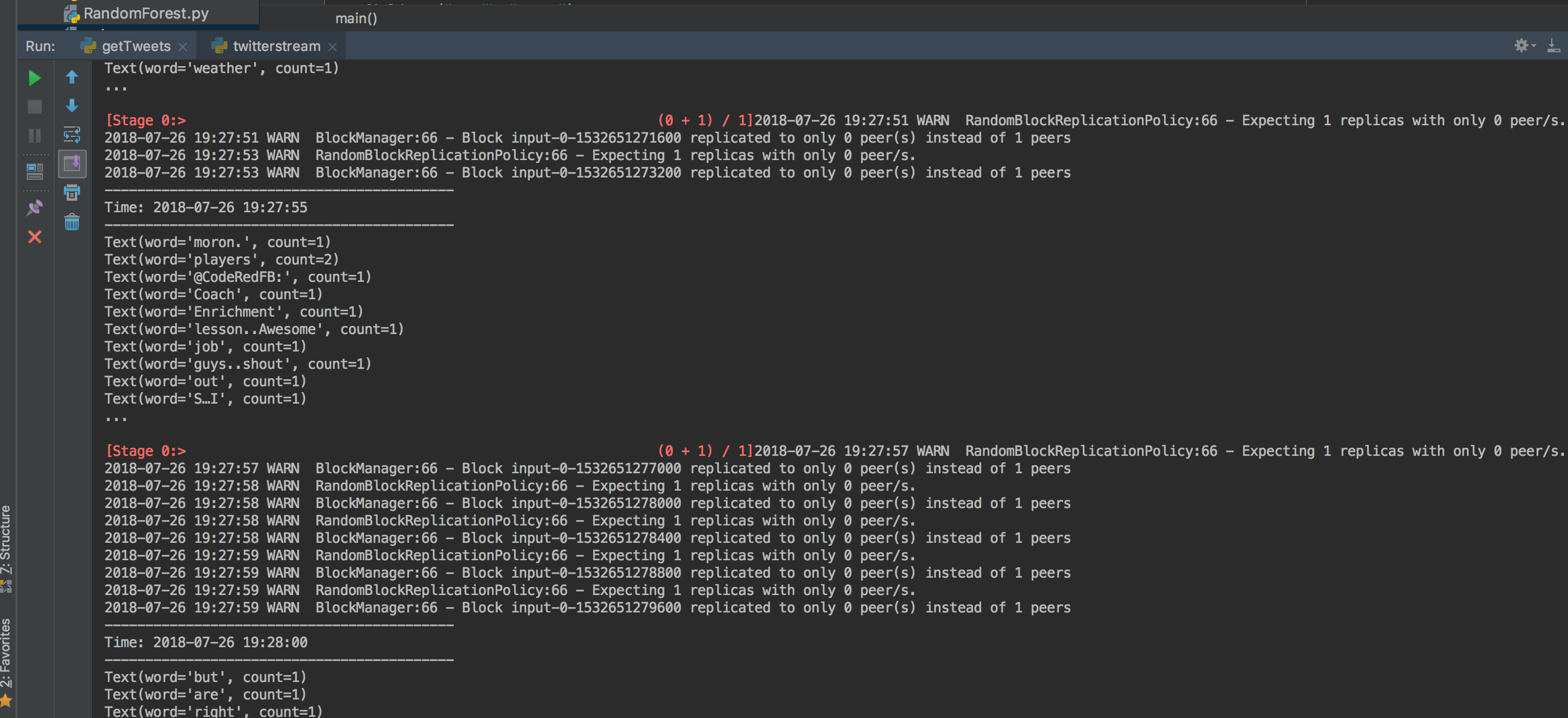


#### Stream Twitter data code:



#### Output of the Twitter Streaming data:





# References:

1. <https://www.linkedin.com/pulse/apache-spark-streaming-twitter-python-laurent-weichberger/>
2. <https://github.com/stefanobaghino/spark-twitter-stream-example>

## Data-sets provided:

1. Absenteeism at work:  
   <https://archive.ics.uci.edu/ml/datasets/Absenteeism+at+work>
2. Immunotherapy Dataset:  
   <https://archive.ics.uci.edu/ml/datasets/Immunotherapy+Dataset>
3. Nashville-meetup Dataset:  
   <https://www.kaggle.com/stkbailey/nashville-meetup>
4. Word Game Dataset:  
   <https://www.kaggle.com/anneloes/wordgame>
5. Cyber Crime Motive:  
   <https://www.kaggle.com/sunilkumarsv/indiacybercrimestats2013>