1. Spark Streaming :

1.1 Objective:

1. The Popular Topics in last 60 seconds.
2. The Popular Topics in last 10 seconds.
3. The Username & his/her Tweets.
4. The Time at which the User Tweeted.
5. The FriendsCount of User.
6. The number of Tweets & Score of User.

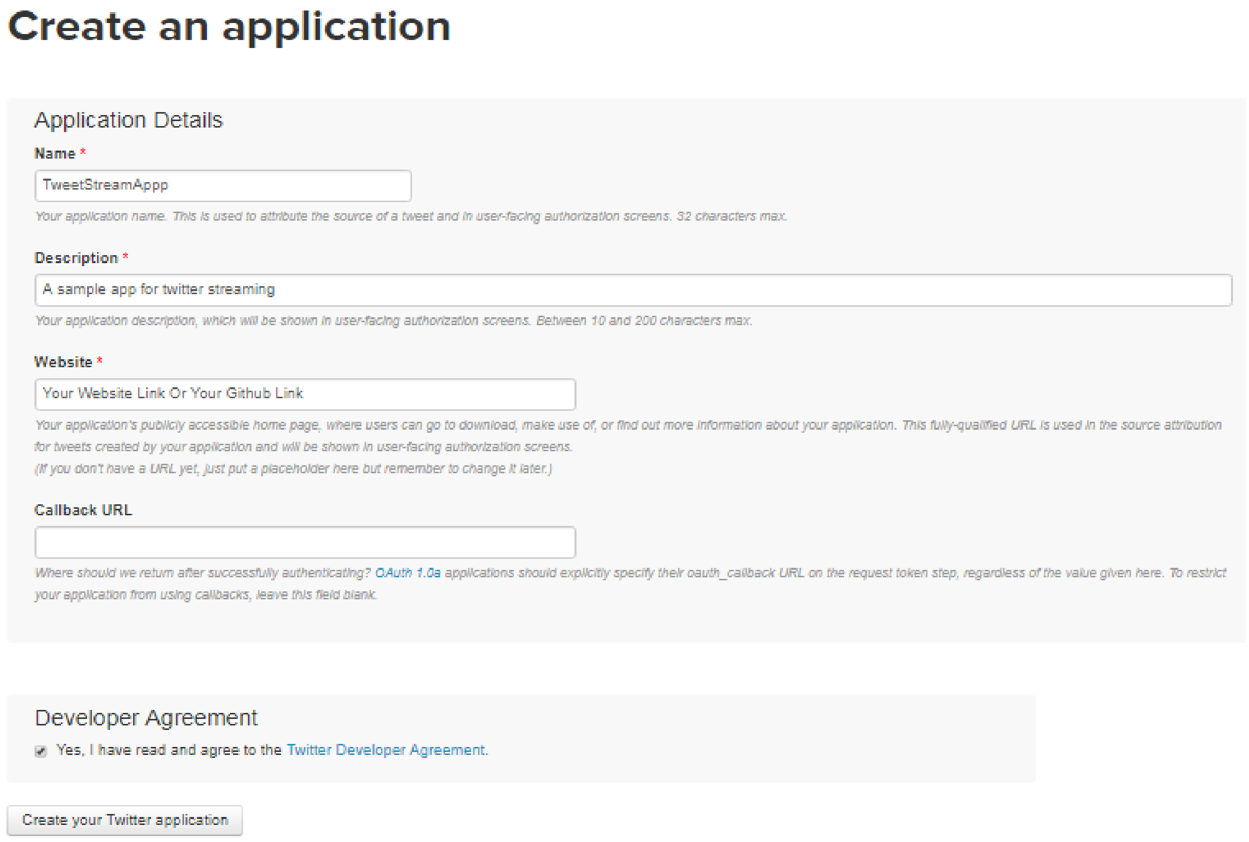
1.2 Procedure:

Stage 1:

For Spark Streaming with Twitter Feed we need to first get our Access Tokens from Twitter.

For Access tokens we need to have a twitter account. So create a twitter account if you don't have one.

Now in order to get tweets from Twitter, Login to https://apps.twitter.com/ with your twitter credentials & "Create new app" and then fill the below form click on "Create your Twitter application."



Create an application

Application Details

Name *

TweetStreamAppp

Your application name. This is used to attribute the source of a tweet and in user-facing authorization screens. 32 characters max.

Description *

A sample app for twitter streaming

Your application description, which will be shown in user-facing authorization screens. Between 10 and 200 characters max.

Website *

Your Website Link Or Your Github Link

Your application's publicly accessible home page, where users can go to download, make use of, or find out more information about your application. This fully-qualified URL is used in the source attribution for tweets created by your application and will be shown in user-facing authorization screens.
(If you don't have a URL yet, just put a placeholder here but remember to change it later.)

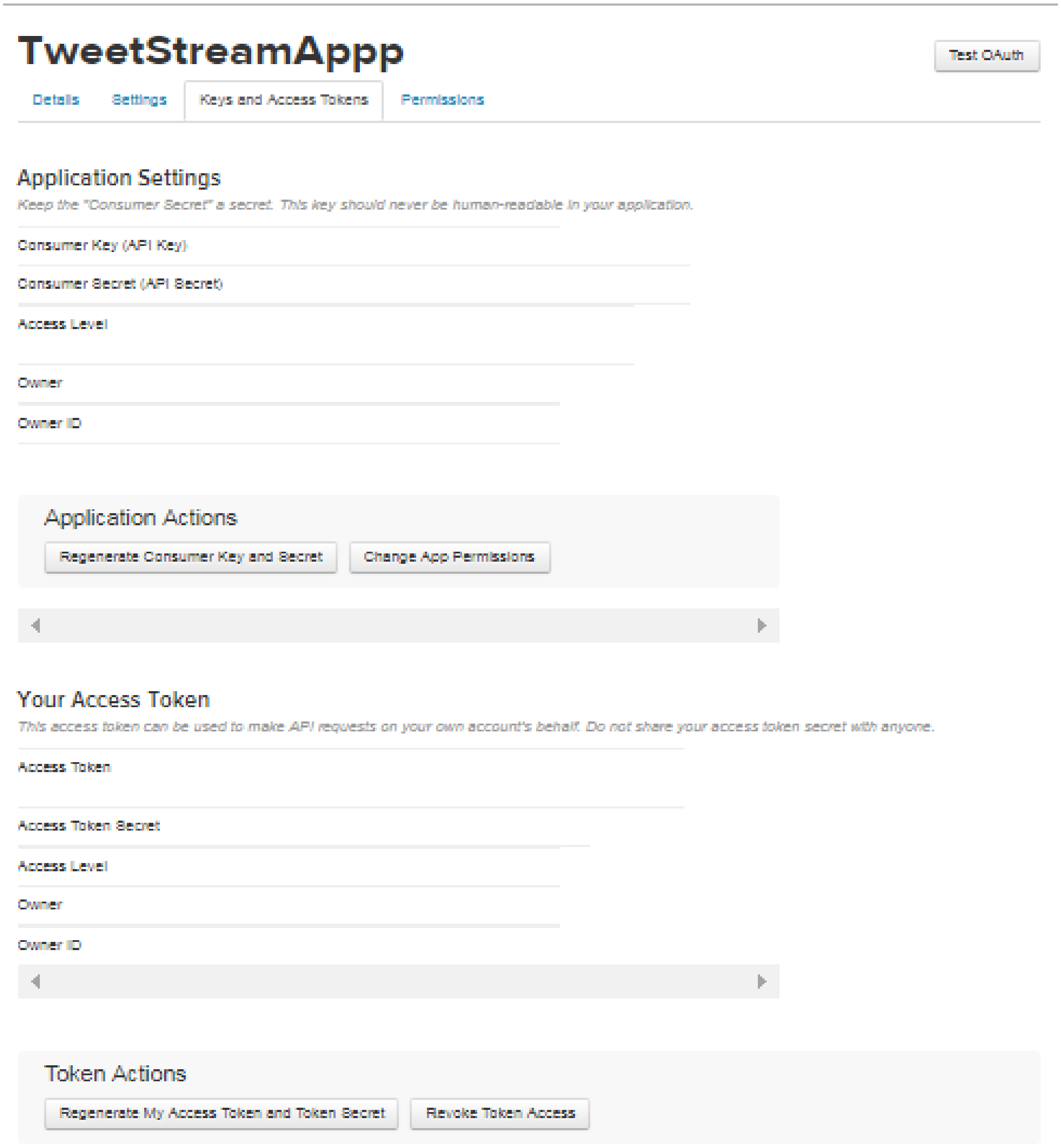
Callback URL

Where should we return after successfully authenticating? OAuth 1.0a applications should explicitly specify their oauth_callback URL on the request token step, regardless of the value given here. To restrict your application from using callbacks, leave this field blank.

Developer Agreement

☑ Yes, I have read and agree to the Twitter Developer Agreement.

The Application Name should be unique. Now go to Key & Access Tokens.



Application Settings

Keep the "Consumer Secret" a secret. This key should never be human-readable in your application.

Consumer Key (API Key)

Consumer Secret (API Secret)

Access Level

Owner

Owner ID

Application Actions

Your Access Token

This access token can be used to make API requests on your own account's behalf. Do not share your access token secret with anyone.

Access Token

Access Token Secret

Access Level

Owner

Owner ID

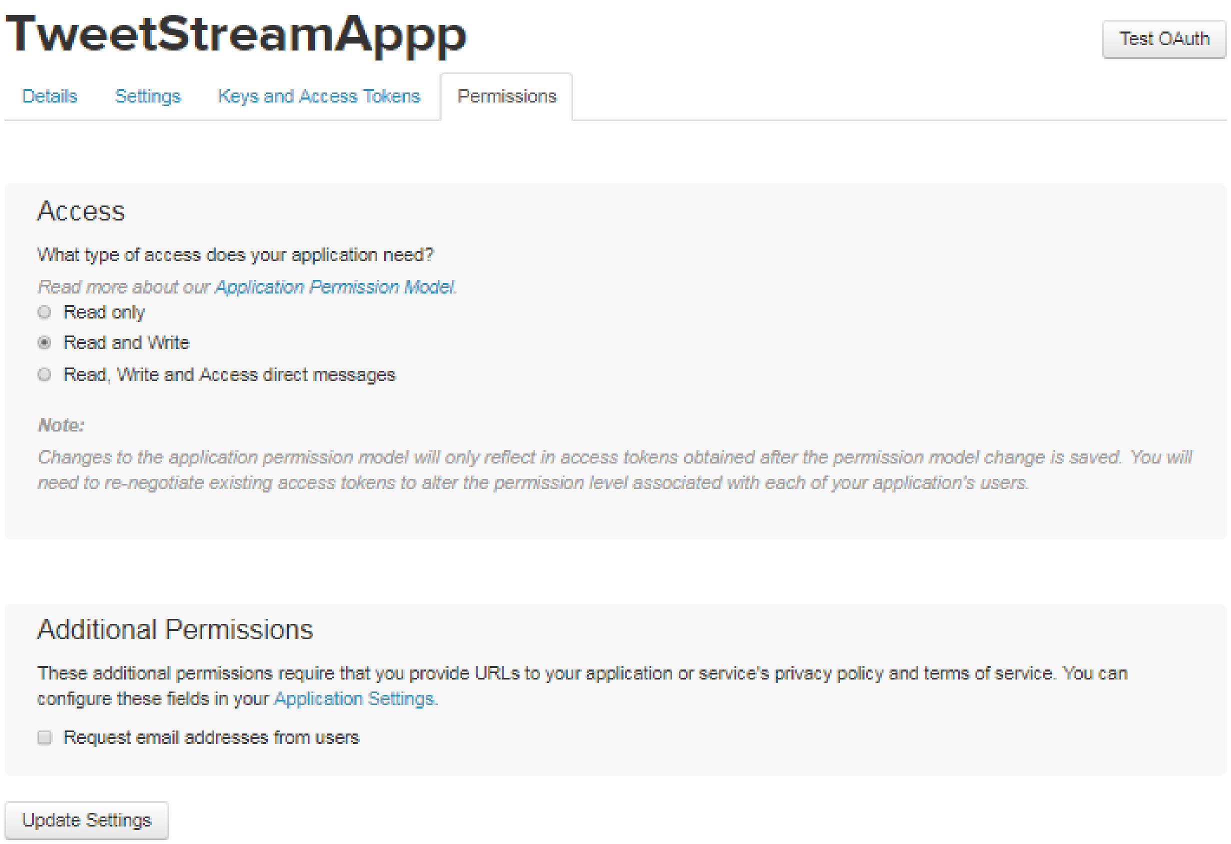
Token Actions

You will get the necessary 4 tokens i.e.:

1. Consumer Key (API Key)
2. Consumer Secret (API Key)
3. Access Token
4. Access Token Secret

If the keys are not generated then you can click on "Regenerate Consumer Key and Secret" and "Regenerate My Access Token and Token Secret"

Also we need to change the permissions. Therefore click on "Change App Permissions".



TweetStreamAppp

Access

What type of access does your application need?

Read more about our Application Permission Model.

○ Read only

● Read and Write

○ Read, Write and Access direct messages

Note:

Changes to the application permission model will only reflect in access tokens obtained after the permission model change is saved. You will need to re-negotiate existing access tokens to alter the permission level associated with each of your application's users.

Additional Permissions

These additional permissions require that you provide URLs to your application or service's privacy policy and terms of service. You can configure these fields in your Application Settings.

□ Request email addresses from users

Select Read & Write (if not selected) & Update Settings.
Your “TweetStreamAppp” is created with the necessary 4 tokens.
Copy Paste those 4 tokens in a notepad file & save it as TwitterTokens.txt

Stage 2:

Now we need to set the HADOOP_HOME environment variable in Windows.

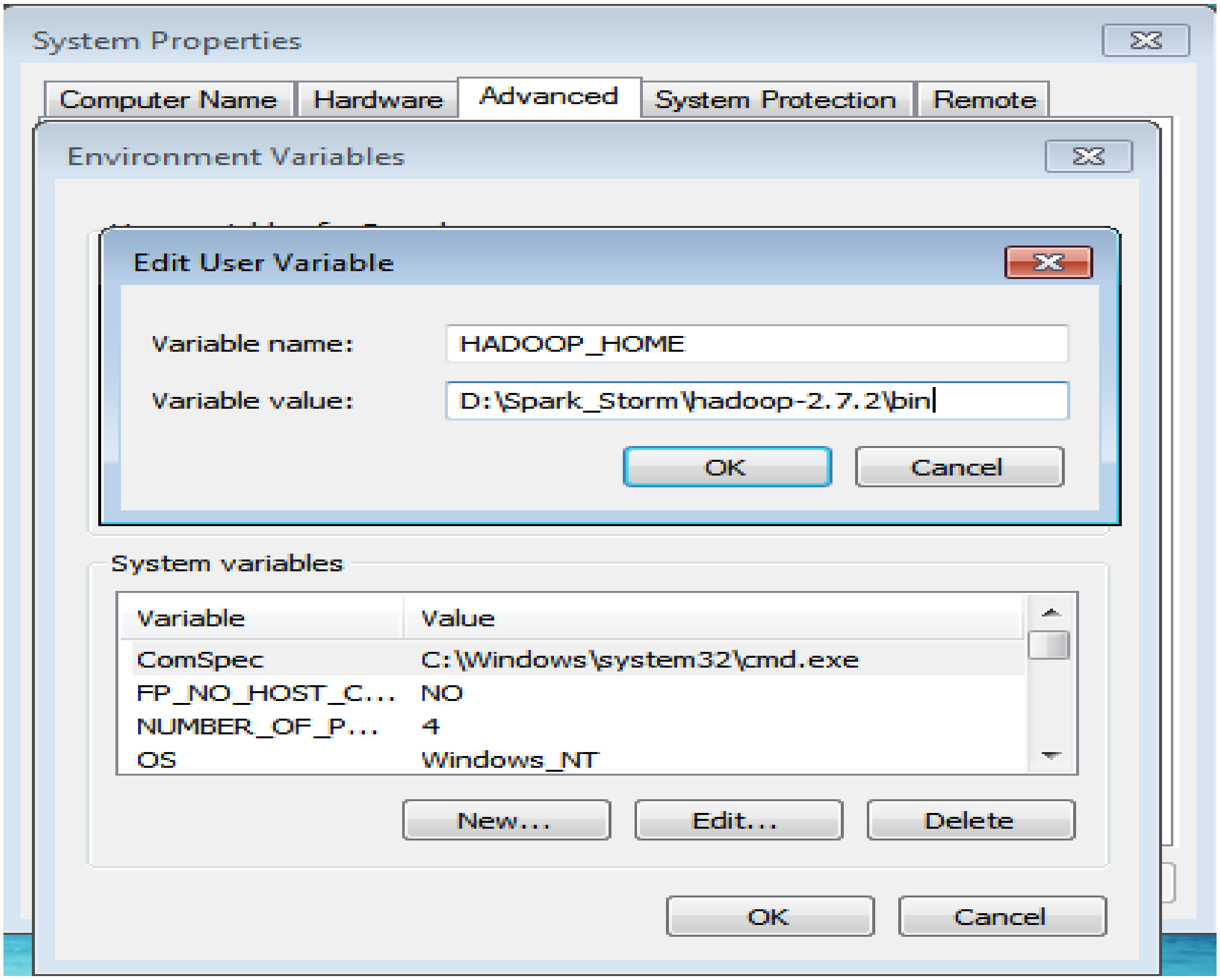
For setting HADOOP_HOME first extract the tar file hadoop-2.7.2.tar.gz to windows location say D:\Spark_Storm

If you are running Spark on Windows with Hadoop, then you need to ensure your windows hadoop installation is properly installed to run spark you need to have winutils.exe in your hadoop home directory bin folder.

You can download the winutils.exe from
https://codeload.github.com/sardetushar/hadooponwindows/zip/master

Extract the hadooponwindows folder.Go to
D:\Spark_Storm\hadooponwindows-master\bin\
& copy winutils.exe to
D:\Spark_Storm\hadoop-2.7.2\bin

Now RightClick on your MyComputer →Properties → Advanced system settings → Environment Variables



Variable name: HADOOP_HOME

Variable value: D:\Spark_Storm\hadoop-2.7.2\bin

System variables

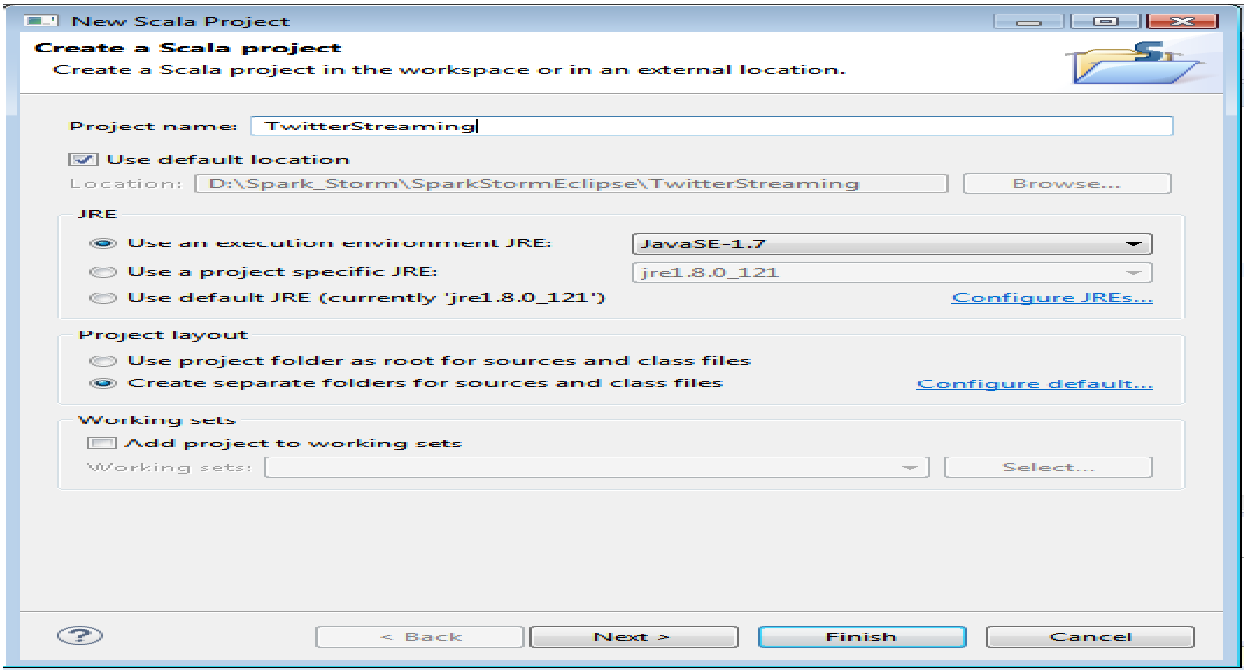
Variable	Value
ComSpec	C:\Windows\system32\cmd.exe
FP_NO_HOST_C...	NO
NUMBER_OF_P...	4
OS	Windows_NT

Click on New of User Variables. Set properties as below & OK.

Similary make sure your JAVA_HOME is also set.

Stage 3:

Now open your Scala-Eclipse IDE. (If you don't have one you can download from http://scala-ide.org/)



New Scala Project

Create a Scala project

Create a Scala project in the workspace or in an external location.

Project name: TwitterStreaming

☑ Use default location

Location: D:\Spark_Storm\SparkStormEclipse\TwitterStreaming

JRE

● Use an execution environment JRE: JavaSE-1.7

○ Use a project specific JRE: jre1.8.0_121

○ Use default JRE (currently 'jre1.8.0_121')

Project layout

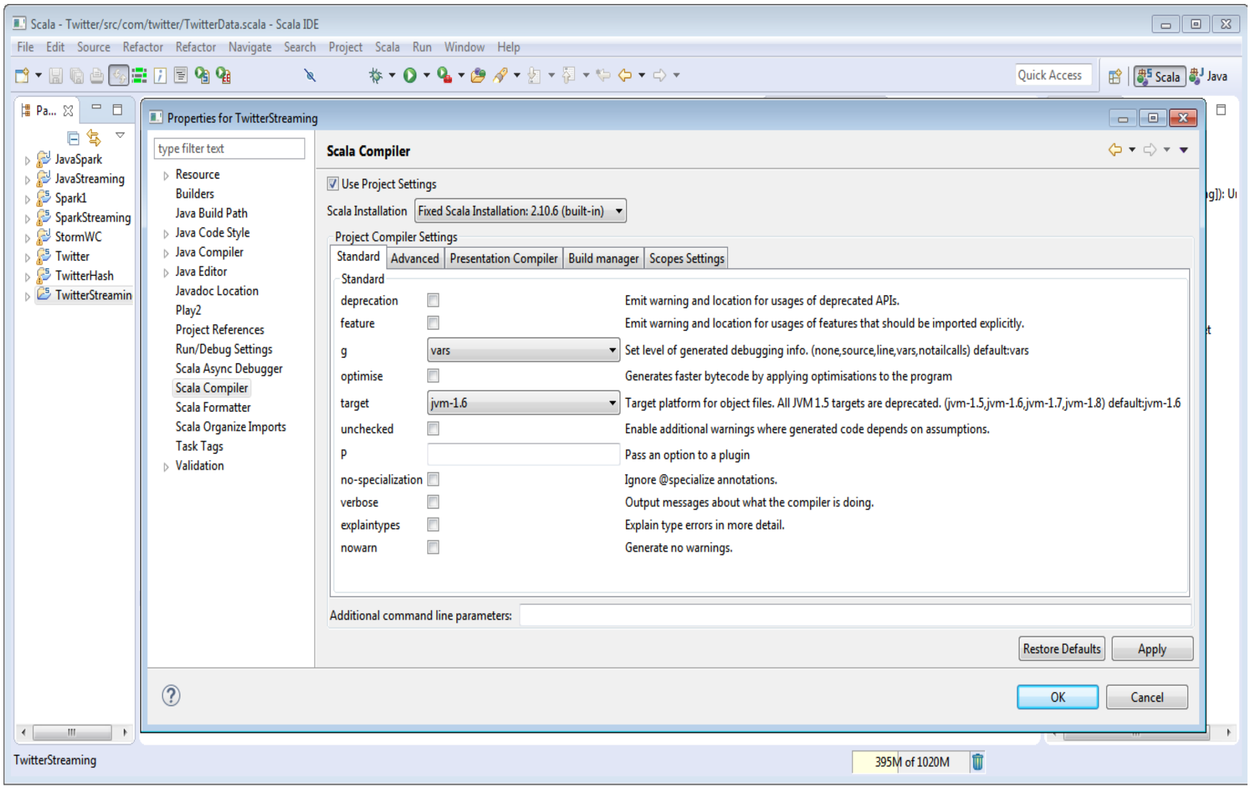
○ Use project folder as root for sources and class files

● Create separate folders for sources and class files

Working sets

□ Add project to working sets

Select File → New → Scala Project And Give a project name say “TwitterStreaming” and Finish.



Scala - Twitter/src/com/twitter/TwitterData.scala - Scala IDE

Properties for TwitterStreaming

Scala Compiler

☑ Use Project Settings

Scala Installation: Fixed Scala Installation: 2.10.6 (built-in)

Project Compiler Settings

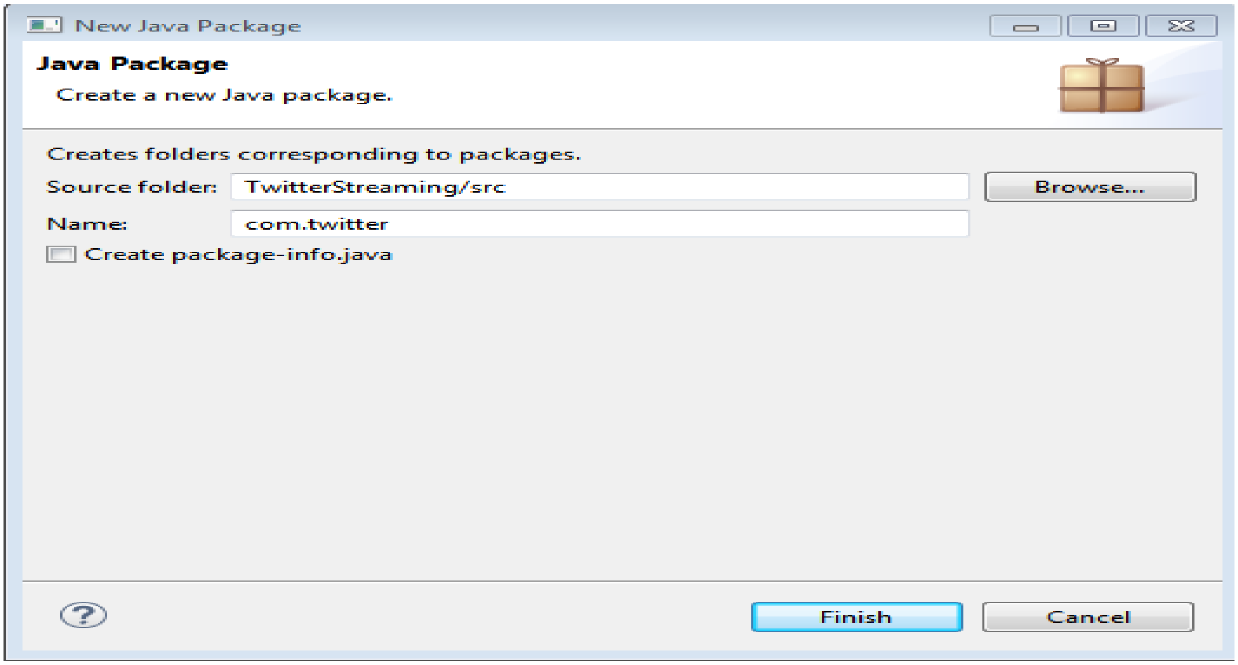
Standard | Advanced | Presentation Compiler | Build manager | Scopes Settings

Standard

deprecation	□	Emit warning and location for usages of deprecated APIs.
feature	□	Emit warning and location for usages of features that should be imported explicitly.
g	vars	Set level of generated debugging info. (none,source,line,vars,notailcalls) default:vars
optimise	□	Generates faster bytecode by applying optimisations to the program
target	jvm-1.6	Target platform for object files. All JVM 1.5 targets are deprecated. (jvm-1.5,jvm-1.6,jvm-1.7,jvm-1.8) default:jvm-1.6
unchecked	□	Enable additional warnings where generated code depends on assumptions.
P		Pass an option to a plugin
no-specialization	□	Ignore @specialize annotations.
verbose	□	Output messages about what the compiler is doing.
explaintypes	□	Explain type errors in more detail.
nowarn	□	Generate no warnings.

Additional command line parameters:

RightClick on TwitterStreaming →Properties →Scala Compiler → Use Project Settings
Select Fixed Scala Installation: 2.10.6(built-in) → OK



Java Package

Create a new Java package.

Creates folders corresponding to packages.

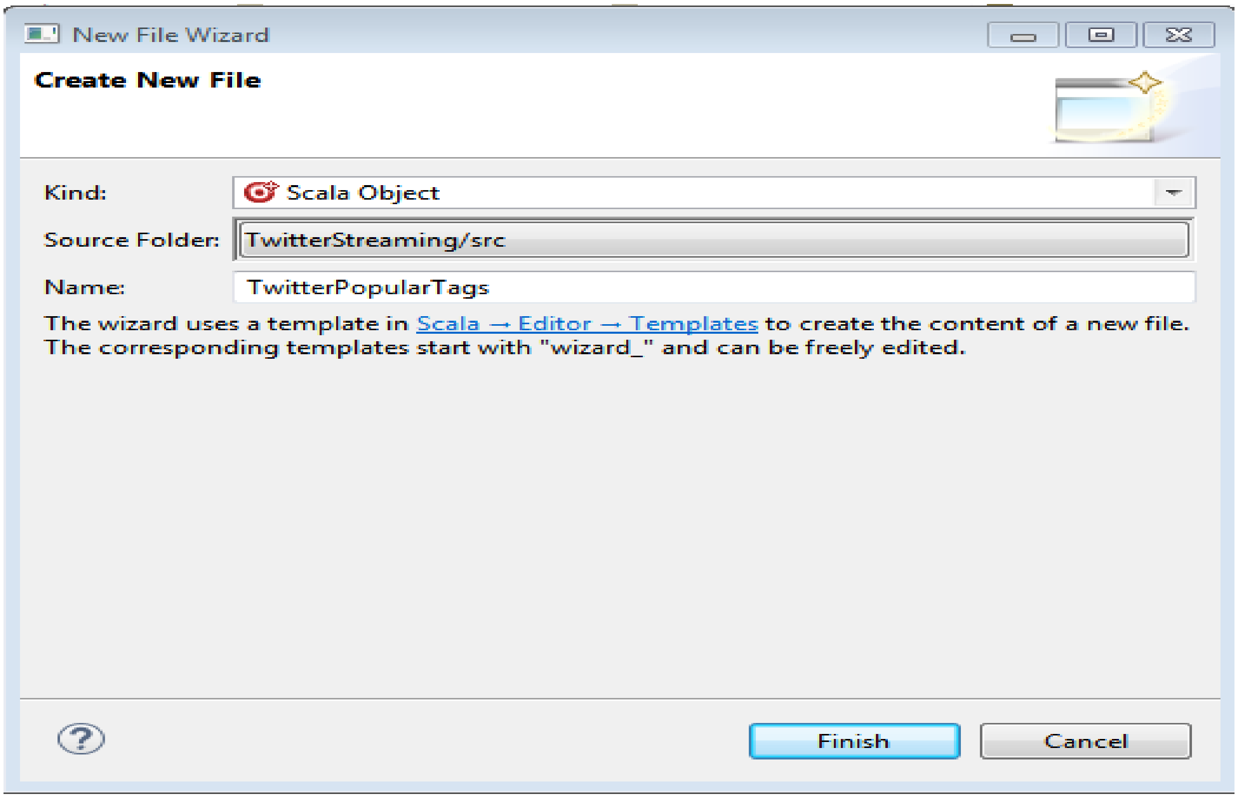
Source folder: TwitterStreaming/src

Name: com.twitter

□ Create package-info.java

RightClick on TwitterStreaming → New → package

Give Name as “com.twitter” → Finish



Create New File

Kind: Scala Object

Source Folder: TwitterStreaming/src

Name: TwitterPopularTags

The wizard uses a template in Scala → Editor → Templates to create the content of a new file. The corresponding templates start with "wizard_" and can be freely edited.

RightClick on com.twitter package in TwitterStreaming → New → Scala Object

Give Name as “TwitterPopularTags”

Now create a lib folder in TwitterStreaming(RightClick → TwitterStreaming → New → Folder)

Give Folder Name as “lib”

Add the following jar files in the lib folder:

1. spark-assembly-1.6.0-hadoop2.6.0.jar
2. spark-streaming-twitter_2.10-1.6.1.jar
3. twitter4j-core-4.0.4.jar
4. twitter4j-stream-4.0.4.jar

After jar files are added to the lib folder then RightClick on lib folder → Select all the jar files → Configure Build-Path → Add to Build-Path

Now you are ready for coding.

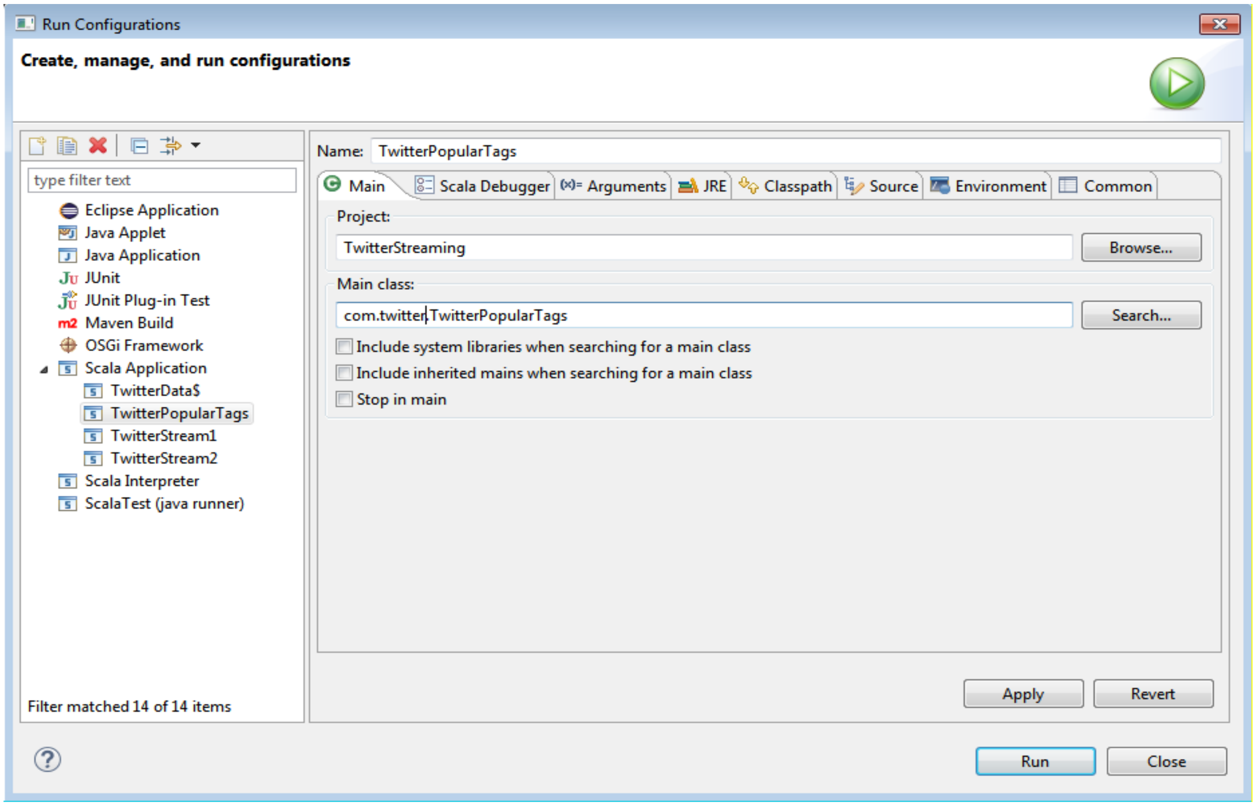
Similarly Create StreamingExamples Scala Object ,TwitterStream1 Scala Object, TwitterStream2 Scala Object in TwitterStreaming Source folder.

StreamingExamples.scala is common for all other scala files.

Now after coding we have to run our TwitterPopularTags.scala as “ScalaApplication”.

So RightClick on TwitterPopularTags.scala in TwitterStreaming folder.

RightClick on TwitterStreaming.scala → RunAs → Run Configurations



Name: TwitterPopularTags

Project: TwitterStreaming

Main class: com.twitter.TwitterPopularTags

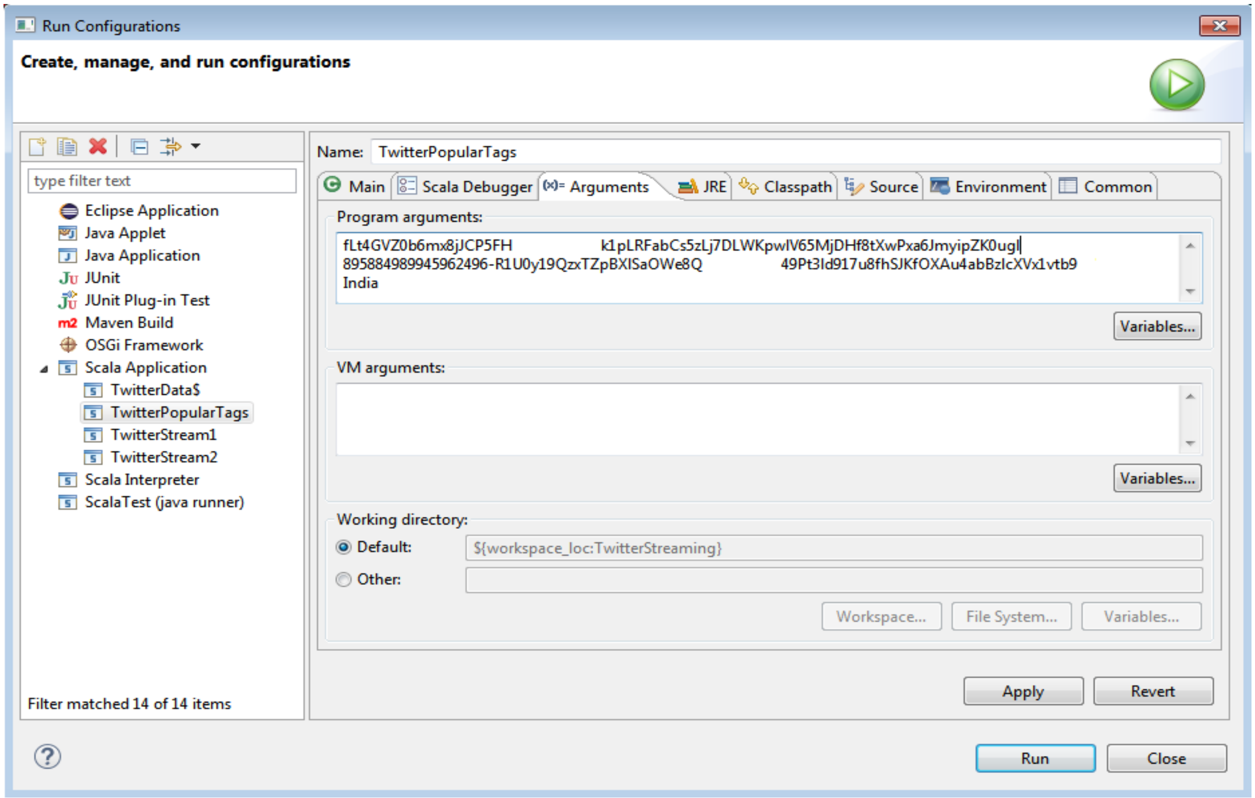
□ Include system libraries when searching for a main class

□ Include inherited mains when searching for a main class

□ Stop in main

Add your TwitterTokens in Arguments and after 4 tokens add any name you want tweets of say e.g. India , NarendraModi, DonaldTrump etc and Apply → Run

You will get the output on console.



Name: TwitterPopularTags

Program arguments:

```
fLt4GVZ0b6mx8jJCP5FH          k1pLRFabCs5zLj7DLWKpwlV65MjDHf8tXwPxa6JmyipZK0ugl
895884989945962496-R1U0y19QzxTZpBXlSaOWe8Q          49Pt3Id917u8fhSJKfOXAu4abBzIcXVx1vtb9
India
```

VM arguments:

Working directory:

● Default: ${workspace_loc:TwitterStreaming}

○ Other:

StreamingExamples.scala

```scala
package com.twitter

import org.apache.spark.Logging

import org.apache.log4j.{Level, Logger}

/** Utility functions for Spark Streaming examples. */
object StreamingExamples extends Logging {

  /** Set reasonable logging levels for streaming if the user has not configured log4j. */
  def setStreamingLogLevels() {
    val log4jInitialized = Logger.getRootLogger.getAllAppenders.hasMoreElements
    if (!log4jInitialized) {
      // We first log something to initialize Spark's default logging, then we override the
      // logging level.
      logInfo("Setting log level to [WARN] for streaming example." +
        " To override add a custom log4j.properties to the classpath.")
      Logger.getRootLogger.setLevel(Level.WARN)
    }
  }
}
```

TwitterPopularTags.scala

```scala
package com.twitter

import org.apache.spark.streaming.{Seconds, StreamingContext}
import org.apache.spark.SparkContext._
import org.apache.spark.streaming.twitter._
import org.apache.spark.SparkConf
import org.apache.spark.streaming.twitter.TwitterUtils

object TwitterPopularTags {
  def main(args: Array[String]) {
    if (args.length < 4) {
      System.err.println("Usage: TwitterPopularTags <consumer key> <consumer secret> " +
        "<access token> <access token secret> [<filters>]")
      System.exit(1)
    }

    StreamingExamples.setStreamingLogLevels()

    val Array(consumerKey, consumerSecret, accessToken, accessTokenSecret) = args.take(4)
    val filters = args.takeRight(args.length - 4)

    // Set the system properties so that Twitter4j library used by twitter stream
    // can use them to generat OAuth credentials
    System.setProperty("hadoop.home.dir", "D:\\Spark_Storm\\hadoop-2.7.2\\bin");
    System.setProperty("twitter4j.oauth.consumerKey", consumerKey)
    System.setProperty("twitter4j.oauth.consumerSecret", consumerSecret)
    System.setProperty("twitter4j.oauth.accessToken", accessToken)
    System.setProperty("twitter4j.oauth.accessTokenSecret", accessTokenSecret)

    val sparkConf = new SparkConf().setAppName("TwitterPopularTags").setMaster("local[2]")
    val ssc = new StreamingContext(sparkConf, Seconds(2))
    val stream = TwitterUtils.createStream(ssc, None, filters)

    val hashTags = stream.flatMap(status => status.getText.split(" ").filter(_.startsWith("#")))

    val topCounts60 = hashTags.map((_, 1)).reduceByKeyAndWindow(_ + _, Seconds(60))
                     .map{case (topic, count) => (count, topic)}
                     .transform(_.sortByKey(false))

    val topCounts10 = hashTags.map((_, 1)).reduceByKeyAndWindow(_ + _, Seconds(10))
                     .map{case (topic, count) => (count, topic)}
                     .transform(_.sortByKey(false))

    // Print popular hashtags
    topCounts60.foreachRDD(rdd => {
      val topList = rdd.take(10)
      println("\nPopular topics in last 60 seconds (%s total):".format(rdd.count()))
      topList.foreach{case (count, tag) => println("%s (%s tweets)".format(tag, count))}
    })

    topCounts10.foreachRDD(rdd => {
      val topList = rdd.take(10)
      println("\nPopular topics in last 10 seconds (%s total):".format(rdd.count()))
      topList.foreach{case (count, tag) => println("%s (%s tweets)".format(tag, count))}
    })

    ssc.start()
    //ssc.awaitTermination()
    ssc.awaitTermination(15000)
  }
}
```

Output :

```
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties

17/08/19 10:19:46 INFO StreamingExamples: Setting log level to [WARN] for streaming example.
To override add a custom log4j.properties to the classpath.

17/08/19 10:19:47 WARN NativeCodeLoader: Unable to load native-hadoop library for your
platform... using builtin-java classes where applicable



[Stage 2:>                                                          (0 + 0) / 2]





Popular topics in last 60 seconds (0 total):



Popular topics in last 10 seconds (0 total):

17/08/19 10:19:54 WARN BlockManager: Block input-0-1503118193800 replicated to only 0 peer(s)
instead of 1 peers



Popular topics in last 60 seconds (0 total):

17/08/19 10:19:54 WARN BlockManager: Block input-0-1503118194600 replicated to only 0 peer(s)
instead of 1 peers



Popular topics in last 10 seconds (0 total):

17/08/19 10:19:55 WARN BlockManager: Block input-0-1503118194800 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:19:55 WARN BlockManager: Block input-0-1503118195000 replicated to only 0 peer(s)
```

```
instead of 1 peers

17/08/19 10:19:56 WARN BlockManager: Block input-0-1503118196200 replicated to only 0 peer(s) instead of 1 peers

17/08/19 10:19:56 WARN BlockManager: Block input-0-1503118196400 replicated to only 0 peer(s) instead of 1 peers

17/08/19 10:19:57 WARN BlockManager: Block input-0-1503118197000 replicated to only 0 peer(s) instead of 1 peers



Popular topics in last 60 seconds (11 total):

#Samantha, (1 tweets)

#TwitterLove (1 tweets)

#Vijay, (1 tweets)

#Neethanae" (1 tweets)

#India (1 tweets)

#Mersal" (1 tweets)

#Prophecies_Of_Nostradamus (1 tweets)

#ShreyaGhoshal, (1 tweets)

#KajalAgarwal (1 tweets)

#ARRahman, (1 tweets)

17/08/19 10:19:57 WARN BlockManager: Block input-0-1503118197200 replicated to only 0 peer(s) instead of 1 peers



Popular topics in last 10 seconds (11 total):

#Samantha, (1 tweets)

#TwitterLove (1 tweets)

#Vijay, (1 tweets)

#Neethanae" (1 tweets)

#India (1 tweets)

#Mersal" (1 tweets)

#Prophecies_Of_Nostradamus (1 tweets)

#ShreyaGhoshal, (1 tweets)
```

```
#KajalAgarwal (1 tweets)

#ARRahman, (1 tweets)

17/08/19 10:19:57 WARN BlockManager: Block input-0-1503118197600 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:19:58 WARN BlockManager: Block input-0-1503118198200 replicated to only 0 peer(s)
instead of 1 peers



Popular topics in last 60 seconds (13 total):

#Samantha, (1 tweets)

#TwitterLove (1 tweets)

#Vijay, (1 tweets)

#Neethanae" (1 tweets)

#India (1 tweets)

#Mersal" (1 tweets)

#Saudi (1 tweets)

#Prophecies_Of_Nostradamus (1 tweets)

#ShreyaGhoshal, (1 tweets)

#ShahInMP (1 tweets)



Popular topics in last 10 seconds (13 total):

#Samantha, (1 tweets)

#TwitterLove (1 tweets)

#Vijay, (1 tweets)

#Neethanae" (1 tweets)

#India (1 tweets)

#Mersal" (1 tweets)

#Saudi (1 tweets)

#Prophecies_Of_Nostradamus (1 tweets)

#ShreyaGhoshal, (1 tweets)

#ShahInMP (1 tweets)
```

```
17/08/19 10:19:59 WARN BlockManager: Block input-0-1503118199200 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:19:59 WARN BlockManager: Block input-0-1503118199400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:19:59 WARN BlockManager: Block input-0-1503118199600 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:00 WARN BlockManager: Block input-0-1503118200000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:00 WARN BlockManager: Block input-0-1503118200200 replicated to only 0 peer(s)
instead of 1 peers


Popular topics in last 60 seconds (14 total):

#Samantha, (1 tweets)

#Bar… (1 tweets)

#TwitterLove (1 tweets)

#Vijay, (1 tweets)

#Neethanae" (1 tweets)

#India (1 tweets)

#Mersal" (1 tweets)

#Saudi (1 tweets)

#Prophecies_Of_Nostradamus (1 tweets)

#ShreyaGhoshal, (1 tweets)


Popular topics in last 10 seconds (14 total):

#Samantha, (1 tweets)

#Bar… (1 tweets)

#TwitterLove (1 tweets)

#Vijay, (1 tweets)

#Neethanae" (1 tweets)

#India (1 tweets)

#Mersal" (1 tweets)
```

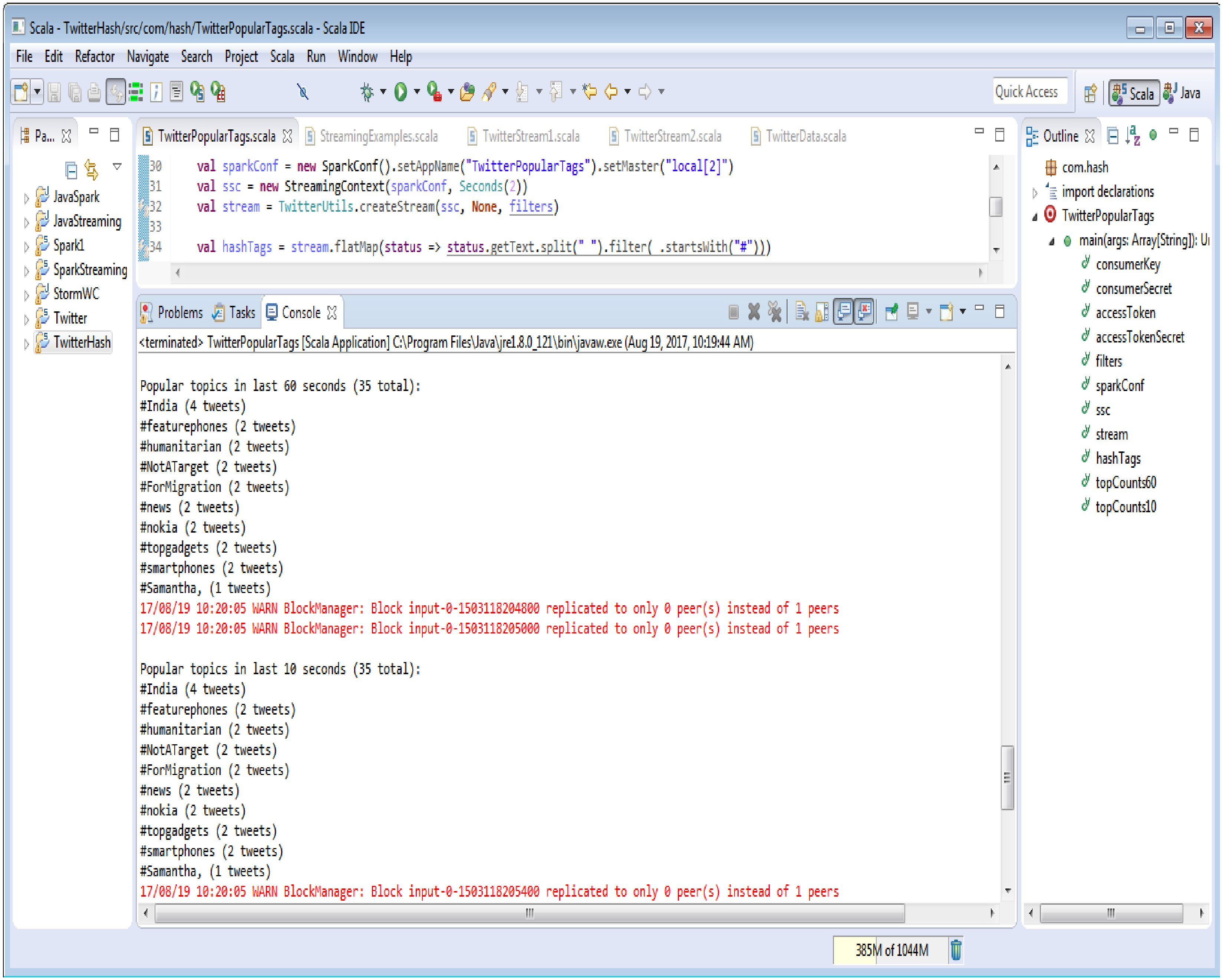
```
#Saudi (1 tweets)

#Prophecies_Of_Nostradamus (1 tweets)

#ShreyaGhoshal, (1 tweets)

17/08/19 10:20:01 WARN BlockManager: Block input-0-1503118201400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:01 WARN BlockManager: Block input-0-1503118201600 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:02 WARN BlockManager: Block input-0-1503118202000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:02 WARN BlockManager: Block input-0-1503118202200 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:02 WARN BlockManager: Block input-0-1503118202400 replicated to only 0 peer(s)
instead of 1 peers



Popular topics in last 60 seconds (23 total):

#Samantha, (1 tweets)

#Bar… (1 tweets)

#featurephones (1 tweets)

#TwitterLove (1 tweets)

#Vijay, (1 tweets)

#Neethanae" (1 tweets)

#1921 (1 tweets)

#EscortNewDelhi

https://t.co/RQ0jevGNGq (1 tweets)

#India (1 tweets)

#news (1 tweets)

17/08/19 10:20:03 WARN BlockManager: Block input-0-1503118202800 replicated to only 0 peer(s)
instead of 1 peers



Popular topics in last 10 seconds (23 total):

#Samantha, (1 tweets)

#Bar… (1 tweets)
```

```
#featurephones (1 tweets)

#TwitterLove (1 tweets)

#Vijay, (1 tweets)

#Neethanae" (1 tweets)

#1921 (1 tweets)

#EscortNewDelhi

https://t.co/RQ0jevGNGq (1 tweets)

#India (1 tweets)

#news (1 tweets)

17/08/19 10:20:03 WARN BlockManager: Block input-0-1503118203000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:03 WARN BlockManager: Block input-0-1503118203200 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:03 WARN BlockManager: Block input-0-1503118203400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:03 WARN BlockManager: Block input-0-1503118203600 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:04 WARN BlockManager: Block input-0-1503118203800 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:04 WARN BlockManager: Block input-0-1503118204000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:04 WARN BlockManager: Block input-0-1503118204400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:04 WARN BlockManager: Block input-0-1503118204600 replicated to only 0 peer(s)
instead of 1 peers



Popular topics in last 60 seconds (35 total):

#India (4 tweets)

#featurephones (2 tweets)

#humanitarian (2 tweets)

#NotATarget (2 tweets)

#ForMigration (2 tweets)
```

```
#news (2 tweets)

#nokia (2 tweets)

#topgadgets (2 tweets)

#smartphones (2 tweets)

#Samantha, (1 tweets)

17/08/19 10:20:05 WARN BlockManager: Block input-0-1503118204800 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:05 WARN BlockManager: Block input-0-1503118205000 replicated to only 0 peer(s)
instead of 1 peers



Popular topics in last 10 seconds (35 total):

#India (4 tweets)

#featurephones (2 tweets)

#humanitarian (2 tweets)

#NotATarget (2 tweets)

#ForMigration (2 tweets)

#news (2 tweets)

#nokia (2 tweets)

#topgadgets (2 tweets)

#smartphones (2 tweets)

#Samantha, (1 tweets)

17/08/19 10:20:05 WARN BlockManager: Block input-0-1503118205400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:20:05 ERROR ReceiverTracker: Deregistered receiver for stream 0: Stopped by driver

17/08/19 10:20:05 WARN BlockManager: Block input-0-1503118205600 replicated to only 0 peer(s)
instead of 1 peers



Popular topics in last 60 seconds (48 total):

#India (7 tweets)

#humanitarian (3 tweets)

#DeineWahl (3 tweets)
```

```
#NotATarget (3 tweets)

#ForMigration (3 tweets)

#featurephones (2 tweets)

#ICRC (2 tweets)

#news (2 tweets)

#nokia (2 tweets)

#2) يوم_الجمعه tweets)


Popular topics in last 10 seconds (38 total):

#India (6 tweets)

#humanitarian (3 tweets)

#DeineWahl (3 tweets)

#NotATarget (3 tweets)

#ForMigration (3 tweets)

#featurephones (2 tweets)

#ICRC (2 tweets)

#news (2 tweets)

#nokia (2 tweets)

#2) يوم_الجمعه tweets)
```

TwitterPopularTags Screenshot:



```
    val sparkConf = new SparkConf().setAppName("TwitterPopularTags").setMaster("local[2]")
    val ssc = new StreamingContext(sparkConf, Seconds(2))
    val stream = TwitterUtils.createStream(ssc, None, filters)

    val hashTags = stream.flatMap(status => status.getText.split(" ").filter(_.startsWith("#")))
```

```
<terminated> TwitterPopularTags [Scala Application] C:\Program Files\Java\jre1.8.0_121\bin\javaw.exe (Aug 19, 2017, 10:19:44 AM)

Popular topics in last 60 seconds (35 total):
#India (4 tweets)
#featurephones (2 tweets)
#humanitarian (2 tweets)
#NotATarget (2 tweets)
#ForMigration (2 tweets)
#news (2 tweets)
#nokia (2 tweets)
#topgadgets (2 tweets)
#smartphones (2 tweets)
#Samantha, (1 tweets)
17/08/19 10:20:05 WARN BlockManager: Block input-0-1503118204800 replicated to only 0 peer(s) instead of 1 peers
17/08/19 10:20:05 WARN BlockManager: Block input-0-1503118205000 replicated to only 0 peer(s) instead of 1 peers

Popular topics in last 10 seconds (35 total):
#India (4 tweets)
#featurephones (2 tweets)
#humanitarian (2 tweets)
#NotATarget (2 tweets)
#ForMigration (2 tweets)
#news (2 tweets)
#nokia (2 tweets)
#topgadgets (2 tweets)
#smartphones (2 tweets)
#Samantha, (1 tweets)
17/08/19 10:20:05 WARN BlockManager: Block input-0-1503118205400 replicated to only 0 peer(s) instead of 1 peers
```

TwitterStream1.scala

```scala
package com.twitter

import org.apache.spark.streaming.{Seconds, StreamingContext}
import org.apache.spark.SparkContext._
import org.apache.spark.streaming.twitter._
import org.apache.spark.SparkConf
import org.apache.spark.streaming.twitter.TwitterUtils

object TwitterStream1 {
  def main(args: Array[String]) {
    if (args.length < 4) {
      System.err.println("Usage: TwitterPopularTags <consumer key> <consumer secret> " +
        "<access token> <access token secret> [<filters>]")
      System.exit(1)
    }

    StreamingExamples.setStreamingLogLevels()

    val Array(consumerKey, consumerSecret, accessToken, accessTokenSecret) = args.take(4)
    val filters = args.takeRight(args.length - 4)

    // Set the system properties so that Twitter4j library used by twitter stream
    // can use them to generat OAuth credentials
    System.setProperty("hadoop.home.dir", "D:\\Spark_Storm\\hadoop-2.7.2\\bin");
    System.setProperty("twitter4j.oauth.consumerKey", consumerKey)
    System.setProperty("twitter4j.oauth.consumerSecret", consumerSecret)
    System.setProperty("twitter4j.oauth.accessToken", accessToken)
    System.setProperty("twitter4j.oauth.accessTokenSecret", accessTokenSecret)

    val sparkConf = new SparkConf().setAppName("TwitterStream1").setMaster("local[2]")

    val ssc = new StreamingContext(sparkConf, Seconds(2))
    val stream = TwitterUtils.createStream(ssc, None, filters)

      stream.foreachRDD((rdd, time) => {
      //group tweets by who tweeted it.
      val groupedByUser = rdd.groupBy(f => f.getUser)
      groupedByUser.foreach(f => {
         System.out.println("User: " + f._1.getScreenName)
         System.out.println("Tweet: "+ f._1.getDescription)
         System.out.println("Time:" + f._1.getCreatedAt+"\n")
   })
})
    ssc.start()
    //ssc.awaitTermination()
    ssc.awaitTermination(20000)
  }
}
```

Output :

```
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties

17/08/19 10:09:51 INFO StreamingExamples: Setting log level to [WARN] for streaming example.
To override add a custom log4j.properties to the classpath.
```

```
17/08/19 10:09:52 WARN NativeCodeLoader: Unable to load native-hadoop library for your
platform... using builtin-java classes where applicable

17/08/19 10:10:11 WARN BlockManager: Block input-0-1503117611200 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:11 WARN BlockManager: Block input-0-1503117611600 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:12 WARN BlockManager: Block input-0-1503117611800 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:12 WARN BlockManager: Block input-0-1503117612000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:12 WARN BlockManager: Block input-0-1503117612200 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:12 WARN BlockManager: Block input-0-1503117612400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:12 WARN BlockManager: Block input-0-1503117612600 replicated to only 0 peer(s)
instead of 1 peers

User: veer_h27

Tweet: null

Time:Tue Apr 28 18:18:25 IST 2015



User: wallytronic

Tweet: Celebrity

Time:Sun Aug 09 06:39:01 IST 2009



User: wohnnezoca1979

Tweet: null

Time:Thu Aug 17 08:43:22 IST 2017



17/08/19 10:10:13 WARN BlockManager: Block input-0-1503117613200 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:14 WARN BlockManager: Block input-0-1503117614000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:14 WARN BlockManager: Block input-0-1503117614200 replicated to only 0 peer(s)
instead of 1 peers
```

```
[Stage 4:>                                                        (0 + 0) / 6]User:
markgardner2991

Tweet: null

Time:Sat May 13 13:30:33 IST 2017



User: alokdubey1408

Tweet: Reactive citizen4 necessary Reaction

Like genuine personalities

Hate Double standard of media &

2know me U must hv time.

Served nation as a proud soldierðŸ‡®ðŸ‡³âœˆðŸš⍰

Time:Sat Apr 27 20:20:29 IST 2013



User: RiaCreations

Tweet: In love with ViTharvâ⍰¤ï¸⍰ShiVik

Passionate VM Maker and Editor!

Jaana Na Dil Se Door Is Lifeâ⍰ ¤ï¸⍰ ðŸ’žðŸ’žðŸ’ž

Time:Fri Dec 30 14:21:11 IST 2016



User: har_dik_

Tweet: executive member #BJYM karnavati mahanagar #BJP General Secretary at Krishnanagarward
Joined #BJYM Since 2009 pursuing LLB

Time:Sat Sep 28 11:50:58 IST 2013



17/08/19 10:10:14 WARN BlockManager: Block input-0-1503117614400 replicated to only 0 peer(s)
instead of 1 peers

User: negi_surjeet

Tweet: null

Time:Fri May 20 16:49:03 IST 2011
```

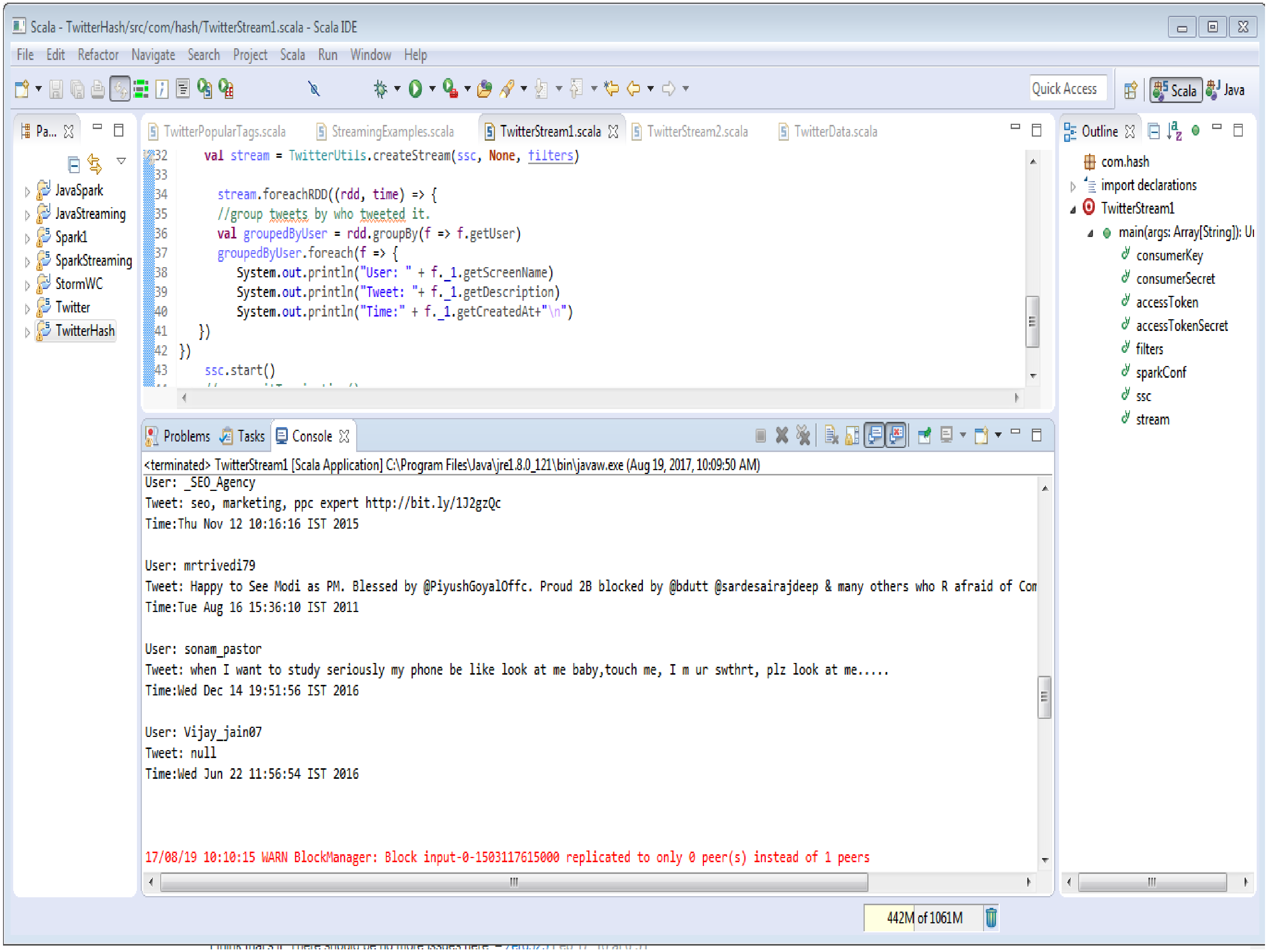
```
User: RakeshBcpsgwl

Tweet: null

Time:Fri May 08 21:52:40 IST 2015



User: veer_h27

Tweet: null

Time:Tue Apr 28 18:18:25 IST 2015



User: MohanChamathil

Tweet: null

Time:Tue Sep 15 23:01:08 IST 2015



User: Debasis_DB

Tweet: IT industry veteran.IIT alumnus. Passionate about India's socio-political system and
how we can collectively aspire to be a great nation.Views are personal.

Time:Thu Jul 11 15:59:46 IST 2013



User: EloiseS23110912

Tweet: null

Time:Thu Aug 17 11:26:54 IST 2017



User: _SEO_Agency

Tweet: seo, marketing, ppc expert http://bit.ly/1J2gzQc

Time:Thu Nov 12 10:16:16 IST 2015



User: mrtrivedi79

Tweet: Happy to See Modi as PM. Blessed by @PiyushGoyalOffc. Proud 2B blocked by @bdutt
@sardesairajdeep & many others who R afraid of Common Indian's Fos/FoE.NO ABUSE

Time:Tue Aug 16 15:36:10 IST 2011
```

```
User: sonam_pastor

Tweet: when I want to study seriously my phone be like look at me baby,touch me, I m ur
swthrt, plz look at me.....

Time:Wed Dec 14 19:51:56 IST 2016



User: Vijay_jain07

Tweet: null

Time:Wed Jun 22 11:56:54 IST 2016





17/08/19 10:10:15 WARN BlockManager: Block input-0-1503117615000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:15 WARN BlockManager: Block input-0-1503117615400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:15 WARN BlockManager: Block input-0-1503117615600 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:10:16 WARN BlockManager: Block input-0-1503117615800 replicated to only 0 peer(s)
instead of 1 peers

User: Parbatrabari3

Tweet: null

Time:Sat May 27 23:28:39 IST 2017



User: markgardner2991

Tweet: null

Time:Sat May 13 13:30:33 IST 2017



User: AnilKum20038432

Tweet: Hum sirf bhonkte hi nahi katte bhi hai...

Time:Sat Apr 08 13:11:56 IST 2017
```

```
User: iopined

Tweet: Have Your Say !

Time:Tue Jul 25 14:39:10 IST 2017



User: nmfansm

Tweet: RTs not endorsement.

Time:Tue Mar 16 07:44:35 IST 2010



17/08/19 10:10:16 WARN BlockManager: Block input-0-1503117616200 replicated to only 0 peer(s)
instead of 1 peers

User: veer_h27

Tweet: null

Time:Tue Apr 28 18:18:25 IST 2015



User: Sidbee17

Tweet: null

Time:Tue Jul 11 15:02:52 IST 2017



User: Keerthy50434546

Tweet: null

Time:Mon Jul 10 09:00:11 IST 2017



User: DailyO_

Tweet: Online opinion, analysis and blog platform from the India Today Group.

Time:Thu Jul 31 12:49:34 IST 2014



User: IslandsGarden

Tweet: Kids, Family and Maternity Photographer in Hyderabad
```

```
Time:Fri Feb 26 15:44:59 IST 2010



User: njhgpriority

Tweet: â˜†Niall Horan gives me a reason to liveâ˜†

Time:Tue Nov 29 03:02:27 IST 2011



17/08/19 10:10:16 WARN BlockManager: Block input-0-1503117616600 replicated to only 0 peer(s) instead of 1 peers

17/08/19 10:10:17 WARN BlockManager: Block input-0-1503117616800 replicated to only 0 peer(s) instead of 1 peers

17/08/19 10:10:17 WARN BlockManager: Block input-0-1503117617000 replicated to only 0 peer(s) instead of 1 peers

17/08/19 10:10:17 WARN BlockManager: Block input-0-1503117617200 replicated to only 0 peer(s) instead of 1 peers

17/08/19 10:10:17 ERROR ReceiverTracker: Deregistered receiver for stream 0: Stopped by driver
```

TwitterStream1 Screenshot:



```
    val stream = TwitterUtils.createStream(ssc, None, filters)

     stream.foreachRDD((rdd, time) => {
     //group tweets by who tweeted it.
     val groupedByUser = rdd.groupBy(f => f.getUser)
     groupedByUser.foreach(f => {
        System.out.println("User: " + f._1.getScreenName)
        System.out.println("Tweet: "+ f._1.getDescription)
        System.out.println("Time:" + f._1.getCreatedAt+"\n")
   })
 })
    ssc.start()
```

```
<terminated> TwitterStream1 [Scala Application] C:\Program Files\Java\jre1.8.0_121\bin\javaw.exe (Aug 19, 2017, 10:09:50 AM)
User: _SEO_Agency
Tweet: seo, marketing, ppc expert http://bit.ly/1J2gzQc
Time:Thu Nov 12 10:16:16 IST 2015

User: mrtrivedi79
Tweet: Happy to See Modi as PM. Blessed by @PiyushGoyalOffc. Proud 2B blocked by @bdutt @sardesairajdeep & many others who R afraid of Com
Time:Tue Aug 16 15:36:10 IST 2011

User: sonam_pastor
Tweet: when I want to study seriously my phone be like look at me baby,touch me, I m ur swthrt, plz look at me.....
Time:Wed Dec 14 19:51:56 IST 2016

User: Vijay_jain07
Tweet: null
Time:Wed Jun 22 11:56:54 IST 2016



17/08/19 10:10:15 WARN BlockManager: Block input-0-1503117615000 replicated to only 0 peer(s) instead of 1 peers
```

TwitterStream2.scala

```scala
package com.twitter

import org.apache.spark.streaming.{Seconds, StreamingContext}
import org.apache.spark.SparkContext._
import org.apache.spark.streaming.twitter._
import org.apache.spark.SparkConf
import org.apache.spark.streaming.twitter.TwitterUtils

object TwitterStream2 {
  def main(args: Array[String]) {
    if (args.length < 4) {
      System.err.println("Usage: TwitterPopularTags <consumer key> <consumer secret> " +
        "<access token> <access token secret> [<filters>]")
      System.exit(1)
    }

    StreamingExamples.setStreamingLogLevels()

    val Array(consumerKey, consumerSecret, accessToken, accessTokenSecret) = args.take(4)
    val filters = args.takeRight(args.length - 4)

    // Set the system properties so that Twitter4j library used by twitter stream
    // can use them to generat OAuth credentials
    System.setProperty("hadoop.home.dir", "D:\\Spark_Storm\\hadoop-2.7.2\\bin");
    System.setProperty("twitter4j.oauth.consumerKey", consumerKey)
    System.setProperty("twitter4j.oauth.consumerSecret", consumerSecret)
    System.setProperty("twitter4j.oauth.accessToken", accessToken)
    System.setProperty("twitter4j.oauth.accessTokenSecret", accessTokenSecret)

    val sparkConf = new SparkConf().setAppName("TwitterStream2").setMaster("local[2]")
    val ssc = new StreamingContext(sparkConf, Seconds(2))
    val stream = TwitterUtils.createStream(ssc, None, filters)

    stream.foreachRDD((rdd, time) => {
    //group tweets by who tweeted it.
    val groupedByUser = rdd.groupBy(f => f.getUser)
      groupedByUser.foreach(f => {
      //how many tweets they sent
        val total = f._2.size
     //how much positive feedback they got, as a double to prevent truncation in score.
          var feedback = 0d
             System.out.println("User: " + f._1.getScreenName)
             System.out.println("Tweet: " + f._1.getDescription)
             System.out.println("Friends Count: " + f._1.getFriendsCount)
             System.out.println("_Tweets: " + f._2.size)
             f._2.foreach(e => feedback += e.getFavoriteCount)
             System.out.println("_Score: " + feedback / total +"\n")
         })
    })
    ssc.start()
    //ssc.awaitTermination()
    ssc.awaitTermination(10000)

}
}
```

Output :

```
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties

17/08/19 10:16:43 INFO StreamingExamples: Setting log level to [WARN] for streaming example.
To override add a custom log4j.properties to the classpath.

17/08/19 10:16:44 WARN NativeCodeLoader: Unable to load native-hadoop library for your
platform... using builtin-java classes where applicable

17/08/19 10:16:53 WARN BlockManager: Block input-0-1503118013000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:53 WARN BlockManager: Block input-0-1503118013200 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:53 WARN BlockManager: Block input-0-1503118013400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:53 WARN BlockManager: Block input-0-1503118013600 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:54 WARN BlockManager: Block input-0-1503118013800 replicated to only 0 peer(s)
instead of 1 peers

User: Nuriya4N

Tweet: null

Friends Count: 975

_Tweets: 1

_Score: 0.0



User: sandeep21031983

Tweet: Hindu Nationalist / Proud Indian RTs are not endorsed.

Friends Count: 225

_Tweets: 1

_Score: 0.0



User: ghirardellyy

Tweet: the megalodon is still alive
```

```
Friends Count: 570

_Tweets: 1

_Score: 0.0



User: ItsMandyyyBitch

Tweet: THE ONLY STRAIGHT I AM IS STRAIGHT UP BITCH.

Friends Count: 288

_Tweets: 1

_Score: 0.0



User: sathchand

Tweet: null

Friends Count: 66

_Tweets: 1

_Score: 0.0



17/08/19 10:16:54 WARN BlockManager: Block input-0-1503118014600 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:55 WARN BlockManager: Block input-0-1503118015000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:55 WARN BlockManager: Block input-0-1503118015400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:56 WARN BlockManager: Block input-0-1503118015800 replicated to only 0 peer(s)
instead of 1 peers

User: kabirazad2017

Tweet: Having a brain is no achievement, using it is. Corruption free bharat hi Azad bharat
hai

Friends Count: 256

_Tweets: 1

_Score: 0.0
```

```
User: hayconcagar1973

Tweet: null

Friends Count: 89

_Tweets: 1

_Score: 0.0


User: falserocks

Tweet: where did all the time go?

Friends Count: 574

_Tweets: 1

_Score: 0.0


User: rohitkr277

Tweet: Social Entrepreneur, Law Student, My tweets are my views...RT isnt endorsement. new
here

Friends Count: 224

_Tweets: 1

_Score: 0.0


User: AshishSinghAME

Tweet: Born Place #Amethi

IT-Professional #NationFirst Pro #RSS #BJP✌Followed by Shri @vinitgoenka Ji, @OmMathur_bjp Ji
And @PiyushGoyalOffc  ❤

Friends Count: 205

_Tweets: 1

_Score: 0.0


17/08/19 10:16:56 WARN BlockManager: Block input-0-1503118016400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:56 WARN BlockManager: Block input-0-1503118016600 replicated to only 0 peer(s)
```

```
instead of 1 peers

17/08/19 10:16:57 WARN BlockManager: Block input-0-1503118017000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:57 WARN BlockManager: Block input-0-1503118017200 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:57 WARN BlockManager: Block input-0-1503118017400 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:58 WARN BlockManager: Block input-0-1503118017800 replicated to only 0 peer(s)
instead of 1 peers


[Stage 6:>                                                          (0 + 0) / 6]

User: GmGmsaini44

Tweet: null

Friends Count: 605

_Tweets: 1

_Score: 0.0


User: lauren_brownie7

Tweet:  finding my happily ever after

Friends Count: 162

_Tweets: 1

_Score: 0.0


User: Charlen01530317

Tweet: null

Friends Count: 93

_Tweets: 1

_Score: 0.0


17/08/19 10:16:59 WARN BlockManager: Block input-0-1503118018800 replicated to only 0 peer(s)
instead of 1 peers
```

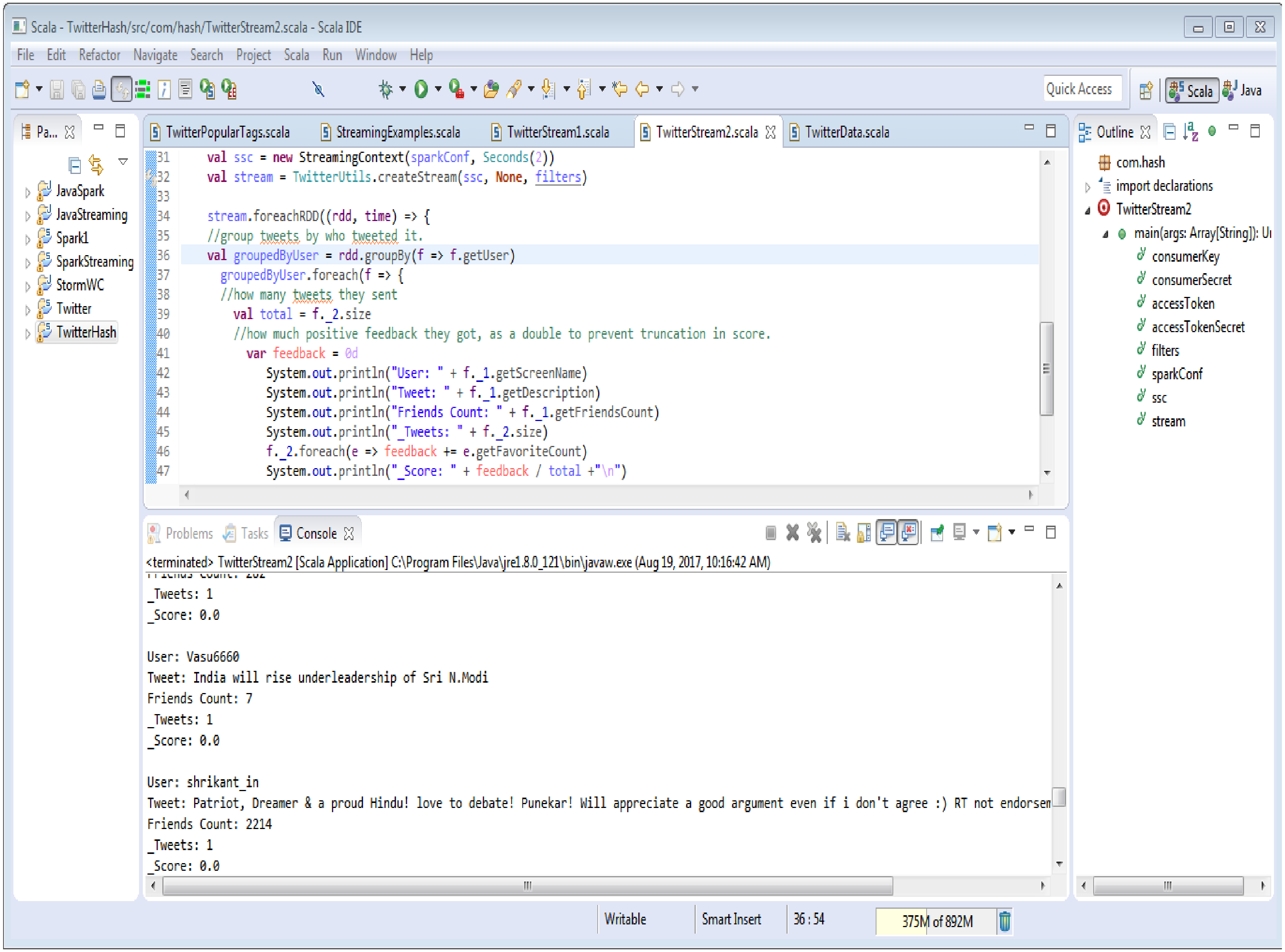
```
User: veekay122002

Tweet: Ex- Fauji.. Proud Indian..

Friends Count: 355

_Tweets: 1

_Score: 0.0



User: PebbleInWaters

Tweet: Regional Director (North) ApON Innovative Solutions http://aponindia.com. Tweets & RTs
are personal. #Traveler #Foodie #Books #Speaker #TechReviewer #Blogger

Friends Count: 5380

_Tweets: 1

_Score: 0.0



User: Ranbir_Kaur

Tweet: A cheated homebuyers at JAYPEE WISHTOWN NOIDA. Trying to wake up government and system.
100% follow back.

Friends Count: 80

_Tweets: 1

_Score: 0.0



User: 3l_4ld0

Tweet: there must be fairness at the end of the day. Porque no somos tan idiotas. Damn! Chin!

Friends Count: 253

_Tweets: 1

_Score: 0.0



User: pillaikala90

Tweet: Community Volunteer, Hiker, Sunshine Addict, Hopeless Romantic, Terminator. I probably
still live in my mom's basement.

Friends Count: 114
```

```
_Tweets: 1

_Score: 0.0





17/08/19 10:16:59 WARN BlockManager: Block input-0-1503118019000 replicated to only 0 peer(s)
instead of 1 peers

17/08/19 10:16:59 WARN BlockManager: Block input-0-1503118019400 replicated to only 0 peer(s)
instead of 1 peers

User: INDBureaucracy

Tweet: We bring you the latest from the Indian Bureaucracy Fraternity

Friends Count: 604

_Tweets: 1

_Score: 0.0


User: Shvnv

Tweet: INSTAGRAM, Shvnv.

Friends Count: 282

_Tweets: 1

_Score: 0.0


User: Vasu6660

Tweet: India will rise underleadership of Sri N.Modi

Friends Count: 7

_Tweets: 1

_Score: 0.0


User: shrikant_in

Tweet: Patriot, Dreamer & a proud Hindu! love to debate! Punekar! Will appreciate a good
argument even if i don't agree :) RT not endorsement
```

```
Friends Count: 2214

_Tweets: 1

_Score: 0.0


User: ghibkeyrapor191

Tweet: null

Friends Count: 33

_Tweets: 1

_Score: 0.0


User: yogesh2011985

Tweet: Yogesh kumar das

Friends Count: 19

_Tweets: 1

_Score: 0.0


User: Vikashch007

Tweet: null

Friends Count: 181

_Tweets: 1

_Score: 0.0


User: meena7099

Tweet: respect fr our  Bharat varsh culture civilization

Friends Count: 4593

_Tweets: 1

_Score: 0.0


User: ivntna
```

```
Tweet: Peace can't be kept by force. It can only achieved by understanding.

Friends Count: 161

_Tweets: 1

_Score: 0.0


17/08/19 10:17:00 ERROR ReceiverTracker: Deregistered receiver for stream 0: Stopped by driver
```

TwitterStream2 Screenshot:



```
    val ssc = new StreamingContext(sparkConf, Seconds(2))
    val stream = TwitterUtils.createStream(ssc, None, filters)

    stream.foreachRDD((rdd, time) => {
    //group tweets by who tweeted it.
    val groupedByUser = rdd.groupBy(f => f.getUser)
      groupedByUser.foreach(f => {
      //how many tweets they sent
        val total = f._2.size
        //how much positive feedback they got, as a double to prevent truncation in score.
          var feedback = 0d
            System.out.println("User: " + f._1.getScreenName)
            System.out.println("Tweet: " + f._1.getDescription)
            System.out.println("Friends Count: " + f._1.getFriendsCount)
            System.out.println("_Tweets: " + f._2.size)
            f._2.foreach(e => feedback += e.getFavoriteCount)
            System.out.println("_Score: " + feedback / total +"\n")
```

```
<terminated> TwitterStream2 [Scala Application] C:\Program Files\Java\jre1.8.0_121\bin\javaw.exe (Aug 19, 2017, 10:16:42 AM)
_Tweets: 1
_Score: 0.0

User: Vasu6660
Tweet: India will rise underleadership of Sri N.Modi
Friends Count: 7
_Tweets: 1
_Score: 0.0

User: shrikant_in
Tweet: Patriot, Dreamer & a proud Hindu! love to debate! Punekar! Will appreciate a good argument even if i don't agree :) RT not endorsem
Friends Count: 2214
_Tweets: 1
_Score: 0.0
```