

```
import org.apache.spark.streaming._
import org.apache.spark.streaming.twitter._
import org.apache.spark.storage.StorageLevel
import scala.io.Source
import scala.collection.mutable.HashMap
import java.io.File
import org.apache.log4j.Logger
import org.apache.log4j.Level
import sys.process.StringSeqToProcess
```

FINISHED ▶ ⌵ 📖 ⚙

```
import org.apache.spark.streaming._
import org.apache.spark.streaming.twitter._
import org.apache.spark.storage.StorageLevel
import scala.io.Source
import scala.collection.mutable.HashMap
import java.io.File
import org.apache.log4j.Logger
import org.apache.log4j.Level
import sys.process.StringSeqToProcess
```

Took 5 sec. Last updated by anonymous at January 26 2017, 7:37:16 PM.

```
/** Configures the OAuth Credentials for accessing Twitter */
def configureTwitterCredentials(apiKey: String, apiSecret: String, accessToken: String, accessTo
  val configs = new HashMap[String, String] += Seq(
    "apiKey" -> apiKey, "apiSecret" -> apiSecret, "accessToken" -> accessToken, "accessTokenS
  println("Configuring Twitter OAuth")
  configs.foreach{ case(key, value) =>
    if (value.trim.isEmpty) {
      throw new Exception("Error setting authentication - value for " + key + " not set")
    }
    val fullKey = "twitter4j.oauth." + key.replace("api", "consumer")
    System.setProperty(fullKey, value.trim)
    println("\tProperty " + fullKey + " set as [" + value.trim + "]")
  }
  println()
}
```

FINISHED ▶ ⌵ 📖 ⚙

```
configureTwitterCredentials: (apiKey: String, apiSecret: String, accessToken: String, accessTo
kenSecret: String)Unit
```

Took 2 sec. Last updated by anonymous at January 26 2017, 7:37:31 PM.

```
// Configure Twitter credentials
val apiKey = "ExGEfwkCmKTnINTic11v7wJEB"
val apiSecret = "0gOrOedCh0yeunV86bzHi3BuknTMW19Y8IVwtDm4QIM4t3V6Ei"
val accessToken = "824754411880710144-rriTEzEswHX71Yo0bTAPuaRVE51vHhn"
```

FINISHED ▶ ⌵ 📖 ⚙

```
val accessTokenSecret = "gbnq7l0ra8X00nwbXRF4VqTrfQVawe01Yl0Ved6KW9i56"
configureTwitterCredentials(apiKey, apiSecret, accessToken, accessTokenSecret)
```

```
apiKey: String = ExGEfwkCmKTnINTic11v7wJEb
apiSecret: String = Og0rOedCh0yeunV86bzHi3BuknTMW19Y8IVwtDm4QIM4t3V6Ei
accessToken: String = 824754411880710144-rrITeZEsW71YoObTAPuaRVE51vHhn
accessTokenSecret: String = gbnq7l0ra8X00nwbXRF4VqTrfQVawe01Yl0Ved6KW9i56
Configuring Twitter OAuth
  Property twitter4j.oauth.consumerKey set as [ExGEfwkCmKTnINTic11v7wJEb]
  Property twitter4j.oauth.accessToken set as [824754411880710144-rrITeZEsW71YoObTAPuaRVE51vHhn]
  Property twitter4j.oauth.consumerSecret set as [Og0rOedCh0yeunV86bzHi3BuknTMW19Y8IVwtDm4QIM4t3V6Ei]
  Property twitter4j.oauth.accessTokenSecret set as [gbnq7l0ra8X00nwbXRF4VqTrfQVawe01Yl0Ved6KW9i56]
```

Took 3 sec. Last updated by anonymous at January 26 2017, 7:37:37 PM.

FINISHED ▷ ⌘ 📖 ⚙

```
import org.apache.spark.{ SparkConf, SparkContext}
import org.apache.spark.streaming._
import org.apache.spark.streaming.twitter._
import org.apache.spark.SparkContext._
```

```
import org.apache.spark.{SparkConf, SparkContext}
import org.apache.spark.streaming._
import org.apache.spark.streaming.twitter._
import org.apache.spark.SparkContext._
```

Took 2 sec. Last updated by anonymous at January 26 2017, 7:37:41 PM.

FINISHED ▷ ⌘ 📖 ⚙

```
val ssc = new StreamingContext(sc, Seconds(2))

val tweets = TwitterUtils.createStream(ssc, None)
val twt = tweets.window(Seconds(10))
```

```
ssc: org.apache.spark.streaming.StreamingContext = org.apache.spark.streaming.StreamingContext@52681750
tweets: org.apache.spark.streaming.dstream.ReceiverInputDStream[twitter4j.Status] = org.apache.spark.streaming.twitter.TwitterInputDStream@1875effc
twt: org.apache.spark.streaming.dstream.DStream[twitter4j.Status] = org.apache.spark.streaming.dstream.WindowedDStream@25557ea9
```

Took 3 sec. Last updated by anonymous at January 26 2017, 7:37:47 PM.

FINISHED ▷ ⌘ 📖 ⚙

```
val sqlContext= new org.apache.spark.sql.SQLContext(sc)
import sqlContext.implicits._
```

```
warning: there was one deprecation warning; re-run with -deprecation for details
sqlContext: org.apache.spark.sql.SQLContext = org.apache.spark.sql.SQLContext@654acb31
import sqlContext.implicits._
```

Took 3 sec. Last updated by anonymous at January 26 2017, 7:37:52 PM.

FINISHED ▶ ⌵ 📖 ⚙

```
case class Tweet(createdAt:Long, text:String)

val tweet = twt.map(status=>
  Tweet(status.getCreatedAt().getTime()/1000, status.getText())
)
```

defined class Tweet

```
tweet: org.apache.spark.streaming.dstream.DStream[Tweet] = org.apache.spark.streaming.dstream.
MappedDStream@2964596f
```

Took 3 sec. Last updated by anonymous at January 26 2017, 7:37:58 PM.

FINISHED ▶ ⌵ 📖 ⚙

```
tweet.foreachRDD(rdd=>
  // Below line works only in spark 1.3.0.
  // For spark 1.1.x and spark 1.2.x,
  // use rdd.registerTempTable("tweets") instead.
  rdd.toDF().registerTempTable("tweets")
)
```

warning: there was one deprecation warning; re-run with -deprecation for details

Took 1 sec. Last updated by anonymous at January 26 2017, 7:38:03 PM.

# Jan26-DataLoad

## Zeppelin

# Jan26-DataLoad

FINISHED ▶ ⌵ 📖 ⚙

keyboard ⚙ 🔒 default ▼

Took 2 sec. Last updated by anonymous at January 26 2017, 7:38:08 PM.

FINISHED ▶ ⌵ 📖 ⚙

```
ssc.start()
```

```
-----
Time: 1485477830000 ms
-----
```

```
StatusJSONImpl{createdAt=Thu Jan 26 19:43:39 EST 2017, id=824779871373008896, text='If you wan
t to be rich yourself, you dont need more scholastic education, you need a real-world financia
l education', source='<a href="http://twitter.com/download/android" rel="nofollow">Twitter for
Android</a>', isTruncated=false, inReplyToStatusId=-1, inReplyToUserId=-1, isFavorited=false,
isRetweeted=false, favoriteCount=0, inReplyToScreenName='null', geoLocation=null, place=null,
retweetCount=0, isPossiblySensitive=false, lang='en', contributorsIDs=[], retweetedStatus=nul
l, userMentionEntities=[], urlEntities=[], hashtagEntities=[], mediaEntities=[], symbolEntitie
s=[], currentUserRetweetId=-1, user=UserJSONImpl{id=1912762165, name='ismael', screenName='_IA
LPZ', location='null', description='null', isContributorsEnabled=false, profileImageUrl='htt
p://pbs.twimg.com/profile_images/744222742187106304/GvMBvW4s_normal.jpg', profileImageUrlHttps
='https://pbs.twimg.com/profile_images/744222742187106304/GvMBvW4s_normal.jpg', isDefaultProfi
leImage=false, url='null', isProtected=false, followersCount=496, status=null, profileBackgrou
ndColor='C0DEED', profileTextColor='333333', profileLinkColor='1DA1F2', profileSidebarFillColo
r='DDEEF6', profileSidebarBorderColor='C0DEED', profileUseBackgroundImage=true, isDefaultProfi
le=true, showAllInlineMedia=false, friendsCount=334, createdAt=Fri Sep 27 21:05:00 EDT 2013, f
```

Took 1 sec. Last updated by anonymous at January 26 2017, 7:40:07 PM.

FINISHED ▶ ⌵ 📖 ⚙

```
%sql select * from tweets where text like '%girl%' limit 10
```



Took 0 sec. Last updated by anonymous at January 26 2017, 7:44:04 PM.

# Zeppelin

# Jan26-DataLoad

Took 2 sec. Last updated by anonymous at January 26 2017, 7:41:06 PM.

FINISHED    

```
)

negative.foreach(p=>
  words.foreach(w=>
    if(p==w) st = st-1
  )
)
if(st>0)
  "positivie"
else if(st<0)
  "negative"
else
  "neutral"
}

// Below line works only in spark 1.3.0.
// For spark 1.1.x and spark 1.2.x,
// use sqlc.registerFunction("sentiment", sentiment _) instead.
salc.udf.reaister("sentiment". sentiment )
```

sentiment: (s: String)String  
res20: org.apache.spark.sql.expressions.UserDefinedFunction = UserDefinedFunction(<function1>,  
StringType,Some(List(StringType)))

Took 3 sec. Last updated by anonymous at January 26 2017, 7:44:46 PM.

%sql select sentiment(text), count(1) from tweets where text like '%girl%' group by sentiment



UDF(text)	coun
neutral	4
positivie	1

Took 1 sec. Last updated by anonymous at January 26 2017, 7:45:27 PM.