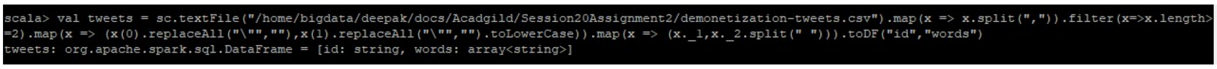
Session 20 Assignment 2

Twitter Sentiment Analysis Using Spark

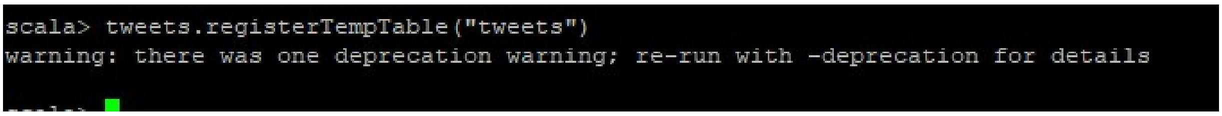
//First read the input data file.

val tweets = sc.textFile("/home/bigdata/deepak/docs/Acadgild/Session20Assignment2/demonetization-tweets.csv").map(x => x.split(",")).filter(x=>x.length>=2).map(x => (x(0).replaceAll("\"",""),x(1).replaceAll("\"","").toLowerCase)).map(x => (x._1,x._2.split(" "))).toDF("id","words")



```
scala> val tweets = sc.textFile("/home/bigdata/deepak/docs/Acadgild/Session20Assignment2/demonetization-tweets.csv").map(x => x.split(",")).filter(x=>x.length>
=2).map(x => (x(0).replaceAll("\"",""),x(1).replaceAll("\"","").toLowerCase)).map(x => (x._1,x._2.split(" "))).toDF("id","words")
tweets: org.apache.spark.sql.DataFrame = [id: string, words: array<string>]
```

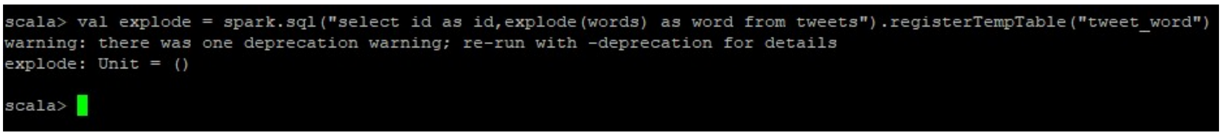
tweets.registerTempTable("tweets")



```
scala> tweets.registerTempTable("tweets")
warning: there was one deprecation warning; re-run with -deprecation for details

scala>
```

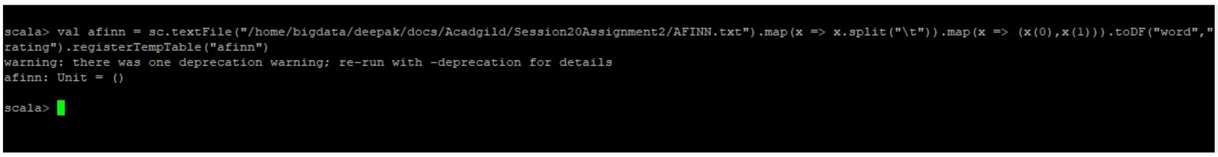
val explode = spark.sql("select id as id,explode(words) as word from tweets").registerTempTable("tweet_word")



```
scala> val explode = spark.sql("select id as id,explode(words) as word from tweets").registerTempTable("tweet_word")
warning: there was one deprecation warning; re-run with -deprecation for details
explode: Unit = ()

scala>
```

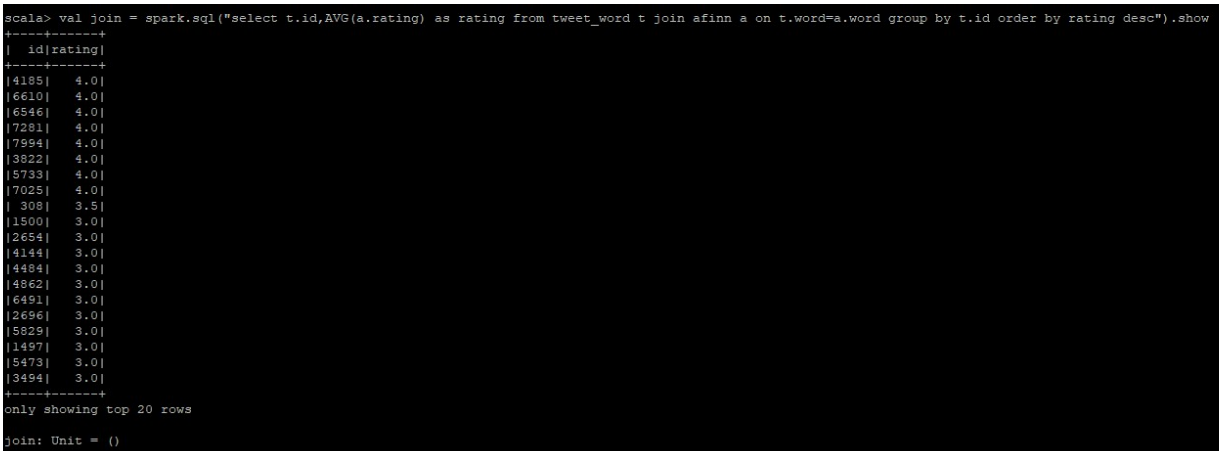
val afinn = sc.textFile("/home/bigdata/deepak/docs/Acadgild/Session20Assignment2/AFINN.txt").map(x => x.split("\t")).map(x => (x(0),x(1))).toDF("word","rating").registerTempTable("afinn")



```
scala> val afinn = sc.textFile("/home/bigdata/deepak/docs/Acadgild/Session20Assignment2/AFINN.txt").map(x => x.split("\t")).map(x => (x(0),x(1))).toDF("word","
rating").registerTempTable("afinn")
warning: there was one deprecation warning; re-run with -deprecation for details
afinn: Unit = ()

scala>
```

val join = spark.sql("select t.id,AVG(a.rating) as rating from tweet_word t join afinn a on t.word=a.word group by t.id order by rating desc").show



```
scala> val join = spark.sql("select t.id,AVG(a.rating) as rating from tweet_word t join afinn a on t.word=a.word group by t.id order by rating desc").show
+----+------+
|  id|rating|
+----+------+
|4185|   4.0|
|6610|   4.0|
|6546|   4.0|
|7281|   4.0|
|7994|   4.0|
|3822|   4.0|
|5733|   4.0|
|7025|   4.0|
| 308|   3.5|
|1500|   3.0|
|2654|   3.0|
|4144|   3.0|
|4484|   3.0|
|4862|   3.0|
|6491|   3.0|
|2696|   3.0|
|5829|   3.0|
|1497|   3.0|
|5473|   3.0|
|3494|   3.0|
+----+------+
only showing top 20 rows

join: Unit = ()
```