

## Assignment 22.2

Downloaded the dataset demonetization-tweets.csv and AFFIN.txt and loaded in the local file system `/home/acadgild/Sumona`

1. First we will read the csv file and then split the columns to get and create a DataFrame. Also, we will create a temporary table named **tweets**

```
val tweets = sc.textFile("/home/acadgild/sumona/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").registerTempTable("tweets")
```

```
scala> val tweets = sc.textFile("/home/acadgild/sumona/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").registerTempTable("tweets")
warning: there was one deprecation warning; re-run with -deprecation for details
tweets: Unit = ()

scala> |
```

2. Now from the above temporary table we will select the ID, words and form another temporary table **tweet\_word**

```
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> |
```

3. Here we will read the **AFFIN** file and create a temporary table **affin**

```
val afinn = sc.textFile("/home/acadgild/sumona/AFINN.txt").map(x => x.split("\t")).map(x =>
(x(0),x(1))).toDF("word","rating").registerTempTable("afinn")
```

Then we will join both the tables **tweet\_word** and **affin** and get the views of different people on demonetization

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

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

scala>

scala> val join = spark.sql("select t.id,AVG(a.rating) as rating from tweet_word t join affin 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|
|5029| 3.0|
|1497| 3.0|
|5473| 3.0|
|3494| 3.0|
+-----+
only showing top 20 rows

join: Unit = ()

scala> █
```