**Assignment 17.2 Spark**

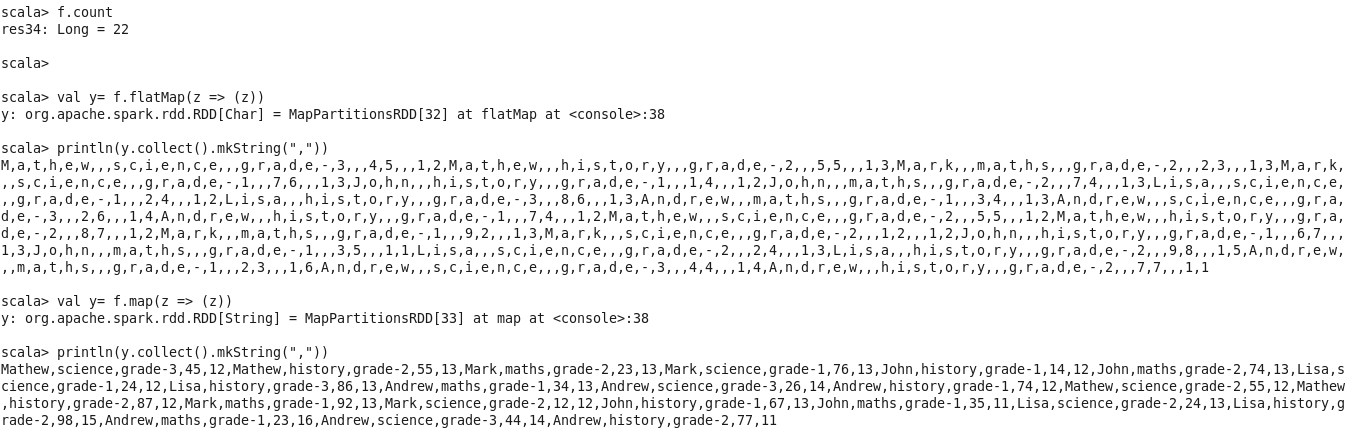
**Dataset :** [**https://drive.google.com/file/d/0B\_P3pWagdIrrQnAxZHZKcmxBQW8/view**](https://drive.google.com/file/d/0B_P3pWagdIrrQnAxZHZKcmxBQW8/view)

**Problem Statement 1:**

1. Read the text file, and create a tuples rdd.
   * val f = sc.textFile("hdfs://quickstart.cloudera:8020/user/cloudera/School\_Dataset.txt")
   * val y= f.flatMap(z => (z))
   * println(y.collect().mkString(","))

or

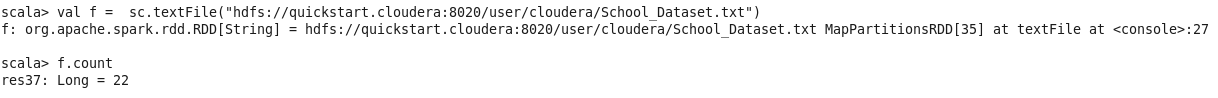
* + val y= f.map(z => (z))
  + println(y.collect(). mkString(","))



* + val w = f.flatMap(l => l.split(" "))
  + val x = w.map(word => (word, 1)).cache()
  + x.count()  
    val z = x.reduceByKey(\_+\_).collect()

2. Find the count of total number of rows present.

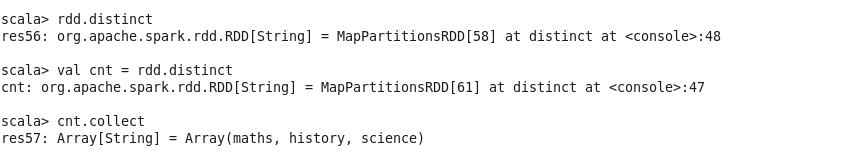
* + val f = sc.textFile("hdfs://quickstart.cloudera:8020/user/cloudera/School\_Dataset.txt")
  + f.count



3.What is the distinct number of subjects present in the entire school?

Subject index in dataset is one

* + val f = sc.textFile("hdfs://quickstart.cloudera:8020/user/cloudera/School\_Dataset.txt")
  + val rdd = f.map(x => { val row = x.split(",").toList ( row.apply(1)) })
  + rdd.countByValue
  + val cnt = rdd.distinct
  + cnt.collect()



4. what is the count of the number of students in the school, whose name is Mathew and marks is 55.

Name index in dataset is zero.

val f = sc.textFile("hdfs://quickstart.cloudera:8020/user/cloudera/School\_Dataset.txt")

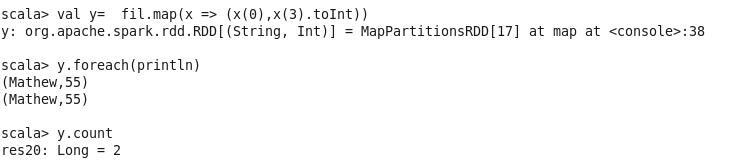
val counts = f.filter { x => {if(x.toString().split(",").length >= 4) true else false} }.map(line=>{line.toString().split(",")})

val fil = counts.filter(x=>{if(x(0).equalsIgnoreCase("Mathew")&&(x(3).matches(("55")))) true else false })

val y= fil.map(x => (x(0),x(3).toInt))

y.foreach(println)

y.count()



**Problem Statement 2:**

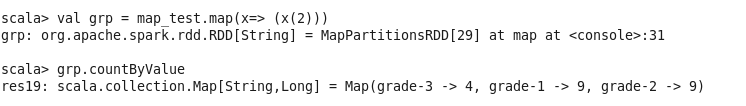
1. What is the count of students per grade in the school?

val f = sc.textFile("hdfs://quickstart.cloudera:8020/user/cloudera/School\_Dataset.txt")

val map\_test= f.map(x => x.split(","))

val grp = map\_test.map(x=> (x(2)))

grp.countByValue



2. Find the average of each student (Note - Mathew is grade-1, is different from Mathew in some other grade!)

import sqlContext.implicits.\_

import org.apache.spark.sql.functions.\_

val sqlcontext = new org.apache.spark.sql.SQLContext(sc)

val f = sc.textFile("hdfs://quickstart.cloudera:8020/user/cloudera/School\_Dataset.txt")

val map\_test= f.map(x => x.split(","))

val cnt1= map\_test.map(y=> (y(0),y(1),y(2),y(3)))

val df = cnt1.toDF()

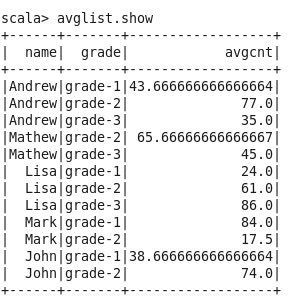
-----

df.registerTempTable("df1");

val stulist = sqlContext.sql("SELECT \* FROM df1")

val avglist = sqlContext.sql("SELECT \_1 as name,\_3 as grade, avg(\_4) as avgcnt FROM df1 group by \_1,\_3")

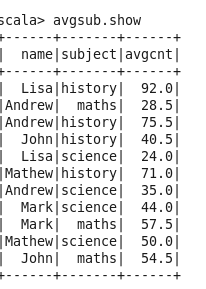
avglist.show()



3. What is the average score of students in each subject across all grades?

val avgsub = sqlContext.sql("SELECT \_1 as name,\_2 as subject, Avg(\_4) as avgcnt FROM df1 group by \_1,\_2")

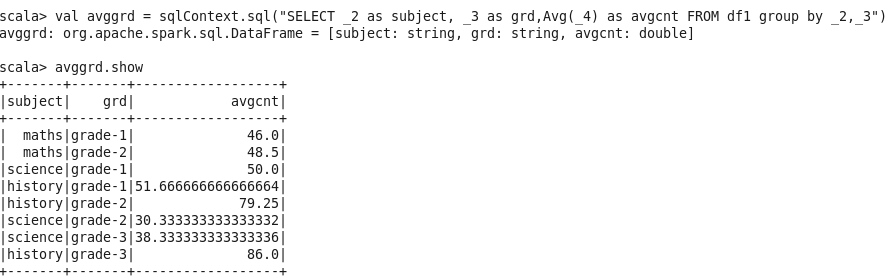
avgsub.show



4. What is the average score of students in each subject per grade?

val avggrd = sqlContext.sql("SELECT \_2 as subject, \_3 as grd,Avg(\_4) as avgcnt FROM df1 group by \_2,\_3")

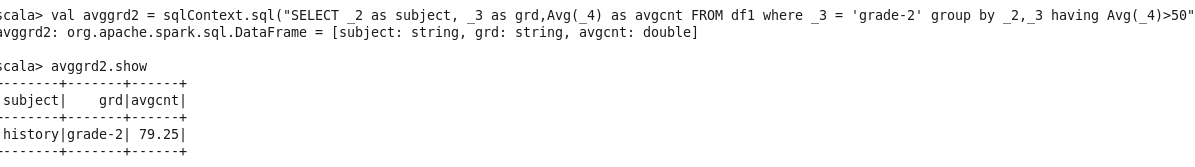
avgsub.show



5. For all students in grade-2, how many have average score greater than 50?

val avggrd2 = sqlContext.sql("SELECT \_2 as subject, \_3 as grd,Avg(\_4) as avgcnt FROM df1 where \_3 = 'grade-2' group by \_2,\_3 having Avg(\_4)>50")

avggrd2.show



**Problem Statement 3:**

Are there any students in the college that satisfy the below criteria:

1. Average score per student\_name across all grades is same as average score per student\_name per grade

Hint - Use Intersection Property.

val avggrd = sqlContext.sql("SELECT \_2 as subject, \_3 as grd,Avg(\_4) as avgcnt FROM df1 group by \_2,\_3")

val avggrd3 = sqlContext.sql("SELECT \_1 as name,\_2 as subject, Avg(\_4) as avgcnt FROM df1 group by \_1,\_2")

val find = avggrd3.unionAll(avggrd).except(avggrd3.intersect(avggrd))