Bigdata Assignment 6.4

Problem Statement 1:
1. Read the text file, and create a tupled rdd.
2. Find the count of total number of rows present.
3. What is the distinct number of subjects present in the entire school
4. What is the count of the number of students in the school, whose name is Mathew and marks is 55

Solution -

1. rdd is created from the the text file and then tuple is created using map function.
val rdd = sc.textFile("file:///home/acadgild/RITESH/6.2.Assignment/17.2_Dataset.txt")

val tuplerdd = rdd.map(x=>x.split(",")).map(array=>array(0),array(1),array(2),array(3)))

```
scala> val rdd = sc.textFile("file:///home/acadgild/RITESH/6.2Assignment/17.2_Dataset.txt")
rdd: org.apache.spark.rdd.RDD[String] = file:///home/acadgild/RITESH/6.2Assignment/17.2_Dataset.txt MapPartitionsRDD[1] at te
xtFile at <console>:24

scala> val tuplerdd = rdd.map(x=>x.split(",")).map(array=>(array(0),array(1),array(2),array(3)))
tuplerdd: org.apache.spark.rdd.RDD[(String, String, String, String)] = MapPartitionsRDD[3] at map at <console>:26

scala> tuplerdd.foreach(println)
(Mathew,science,grade-3,45)
(Mathew,history,grade-2,55)
(Mark,maths,grade-2,23)
(Mark,science,grade-1,76)
(John,history,grade-1,14)
(John,maths,grade-2,74)
(Lisa,science,grade-1,24)
(Lisa,history,grade-3,86)
(Andrew,maths,grade-1,34)
(Andrew,science,grade-3,26)
(Andrew,history,grade-1,74)
(Mathew,science,grade-2,55)
(Mathew,history,grade-2,87)
(Mark,maths,grade-1,92)
(Mark,science,grade-2,12)
(John,history,grade-1,67)
(John,maths,grade-1,35)
(Lisa,science,grade-2,24)
(Lisa,history,grade-2,98)
(Andrew,maths,grade-1,23)
(Andrew,science,grade-3,44)
(Andrew,history,grade-2,77)
```

2. Count function is used to count no of rows present.
tuplerdd.count
Output - 22

```
scala> tuplerdd.count
res1: Long = 22
```

3. From the tuple rdd , we selected , the distinct subjects using map ,distinct and count

val distinctSubject = tuplerdd.map(x=>x._2).distinct.count

Output - 3

```
scala> val distinctSubject = tuplerdd.map(x=>x._2).distinct.count
distinctSubject: Long = 3
```

4. From tuplerdd we filtered where name is Mathew and marks is 55 and counted using count function

val No_stud =
tuplerdd.filter(x=>(x._1==”Mathew”)&&(x._4==”55”)))
No_stud.collect
No_stud.count

Output - 2

```
scala> val No_stud = tuplerdd.filter(x=>((x._1=="Mathew")&&(x._4=="55")))
No_stud: org.apache.spark.rdd.RDD[(String, String, String, String)] = MapPartitionsRDD[12] at filter at <console>:28

scala> No_stud.collect
res4: Array[(String, String, String, String)] = Array((Mathew,history,grade-2,55), (Mathew,science,grade-2,55))

scala> No_stud.count
res5: Long = 2
```

Problem Statement 2:
1. What is the count of students per grade in the school?
2. Find the average of each student (Note - Mathew is grade-1, is different from Mathew in some other grade!)
3. What is the average score of students in each subject across all grades?
4. What is the average score of students in each subject per grade?
5. For all students in grade-2, how many have average score greater than 50?

Solution -

1. From tuple rdd we grouped by grade , and then we mapped for each grade and its corresponding no of students.

Val stud_grade = tuplerdd.groupBy(x=>x._3)
val gradeCounts = stud_grade.map(x=>(x._1,x._2.size)
gradeCounts.collect

```
scala> val stud_grade = tuplerdd.groupBy(x=>x._3)
stud_grade: org.apache.spark.rdd.RDD[(String, Iterable[(String, String, String, String)])] = ShuffledRDD[14] at groupBy at <c
onsole>:28

scala> val gradeCounts = stud_grade.map(x=>(x._1,x._2.size))
gradeCounts: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[15] at map at <console>:30

scala> gradeCounts.collect
res6: Array[(String, Int)] = Array((grade-3,4), (grade-1,9), (grade-2,9))
```

2. From rdd tuple , it is grouped by name and grade

val students = tuplerdd.groupBy(x==>(x._1,x._3))
val stud_avg =
students.map(x=>(x._1._1,x._1._2,x._2.map(_._4.toInt).sum/x._2.size))

```
scala> val students = tuplerdd.groupBy(x=>(x._1,x._3))
students: org.apache.spark.rdd.RDD[((String, String), Iterable[(String, String, String, String)])] = ShuffledRDD[9] at groupB
y at <console>:31

scala> val stud_avg = students.map(x=>(x._1._1,x._1._2,x._2.map(_._4.toInt).sum/x._2.size))
stud_avg: org.apache.spark.rdd.RDD[(String, String, Int)] = MapPartitionsRDD[10] at map at <console>:33

scala> stud_avg.collect
res1: Array[(String, String, Int)] = Array((Lisa,grade-1,24), (Mark,grade-2,17), (Lisa,grade-2,61), (Mathew,grade-3,45), (And
rew,grade-2,77), (Andrew,grade-1,43), (Lisa,grade-3,86), (John,grade-1,38), (John,grade-2,74), (Mark,grade-1,84), (Andrew,gra
de-3,35), (Mathew,grade-2,65))
```

3. From tuple rdd , we grouped by subject and sum up the marks and find its avg.

Val subjects = tuplerdd.groupBy(x=>(x._2))
val subject_avg =
subjects.map(x=>(x._1,x._2.map(_._4.toInt).sum/x._2.size))
subject_avg.collect

```
scala> val subjects = tuplerdd.groupBy(x=>(x._2))
subjects: org.apache.spark.rdd.RDD[(String, Iterable[(String, String, String, String)])] = ShuffledRDD[20] at groupBy at <co
sole>:28

scala> val subject_avg = subjects.map(x=>(x._1,x._2.map(_._4.toInt).sum/x._2.size))
subject_avg: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[21] at map at <console>:30

scala> subject_avg.collect
res10: Array[(String, Int)] = Array((maths,46), (history,69), (science,38))
```

4. Rdd tuple is grouped by subject and grade, then marks are summed up and its average is calculated

val subjects = tuplerdd.groupBy(x=>(x._2,x._3))
val subject_avg =
subjects.map(x=>(x._1._1,x._1._2,x._2.map(_._4.toInt).sum/x._2.size))

subject_avg.collect

```
scala> val subjects = tuplerdd.groupBy(x=>(x._2,x._3))
subjects: org.apache.spark.rdd.RDD[((String, String), Iterable[(String, String, String, String)])] = ShuffledRDD[23] at group
By at <console>:28

scala> val subject_avg = subjects.map(x=>(x._1._1,x._1._2,x._2.map(_._4.toInt).sum/x._2.size))
subject_avg: org.apache.spark.rdd.RDD[(String, String, Int)] = MapPartitionsRDD[24] at map at <console>:30

scala> subject_avg.collect
res11: Array[(String, String, Int)] = Array((history,grade-2,79), (history,grade-3,86), (maths,grade-1,46), (science,grade-3,
38), (science,grade-1,50), (science,grade-2,30), (history,grade-1,51), (maths,grade-2,48))
```

5. From tuple rdd , we grouped by subject and grade .Then summed the marks and find its avg , then checking whose is greater than 50

val subjects = tuplerdd.groupBy(x=>(x._1,x._3))
val subject_avg =
subjects.map(x=>(x._1._1,x._1._2,x._2.map(_._4.toInt).sum/x._2.size))
subject_avg.collect
subject_avg.count()

```
scala> val subjects = tuplerdd.groupBy(x=>(x._1,x._3))
subjects: org.apache.spark.rdd.RDD[((String, String), Iterable[(String, String, String, String)])] = ShuffledRDD[26] at group
By at <console>:28

scala> val subject_avg = subjects.map(x=>(x._1._1,x._1._2,x._2.map(_._4.toInt).sum/x._2.size))
subject_avg: org.apache.spark.rdd.RDD[(String, String, Int)] = MapPartitionsRDD[27] at map at <console>:30

scala> subject_avg.collect
res12: Array[(String, String, Int)] = Array((Lisa,grade-1,24), (Mark,grade-2,17), (Lisa,grade-2,61), (Mathew,grade-3,45), (An
drew,grade-2,77), (Andrew,grade-1,43), (Lisa,grade-3,86), (John,grade-1,38), (John,grade-2,74), (Mark,grade-1,84), (Andrew,gr
ade-3,35), (Mathew,grade-2,65))
```

```
scala> subject_avg.count()
res16: Long = 12
```

Output - 12

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

Solution-

From rdd tuple , we grouped the data by student name then we summed up its score and calculated its average.

val student_group = tuplerdd.groupBy(x=>(x._1))
val student_avg =
student_group.map(x=>(x._1,x._2.map(_._4.toInt).sum/x._2.size))
student_avg.collect

```
scala> val student_group = tuplerdd.groupBy(x=>(x._1))
student_group: org.apache.spark.rdd.RDD[(String, Iterable[(String, String, String, String)])] = ShuffledRDD[32] at groupBy at
 <console>:28

scala> val student_avg = student_group.map(x=>(x._1,x._2.map(_._4.toInt).sum/x._2.size))
student_avg: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[33] at map at <console>:30

scala> student_avg.collect
res18: Array[(String, Int)] = Array((Mark,50), (Andrew,46), (Mathew,60), (John,47), (Lisa,58))
```

From rdd tuple , we grouped the data by student name and gradethen we summed up its score and calculated its average.

val group = tuplerdd.groupBy(x=>(x._1,x._3))
val avg = group.map(x=>(x._1._1,x._2.map(_._4.toInt).sum/x._2.size))
avg.collect

```
scala> val group = tuplerdd.groupBy(x=>(x._1,x._3))
group: org.apache.spark.rdd.RDD[((String, String), Iterable[(String, String, String, String)])] = ShuffledRDD[35] at groupBy
at <console>:28

scala> val avg = group.map(x=>(x._1._1,x._2.map(_._4.toInt).sum/x._2.size))
avg: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[36] at map at <console>:30

scala> avg.collect
res19: Array[(String, Int)] = Array((Lisa,24), (Mark,17), (Lisa,61), (Mathew,45), (Andrew,77), (Andrew,43), (Lisa,86), (John,
38), (John,74), (Mark,84), (Andrew,35), (Mathew,65))
```

Then we intersected both the data

val result = avg.intersect(student_avg)
result.count

```
scala> val result = avg.intersection(student_avg)
result: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[48] at intersection at <console>:36

scala> result.count
res21: Long = 0
```

Output - 0