Assignment 19.1
RDD Deep Dive

Task 1

1.Write a program to read a text file and print the number of rows of data in the document.

2. Write a program to read a text file and print the number of words in the document.

3. We have a document where the word separator is -, instead of space. Write a spark

code, to obtain the count of the total number of words present in the document.

1.Write a program to read a text file and print the number of rows of data in the document.

In below program, we have created Spark object initially and then loaded data from text file. Then by using count() function we have printed the number of rows of data in the document.

Code :

```
package Core

import org.apache.spark.sql.SparkSession

object RDD_practice {

  def main(args : Array[String]) = {

    val spark = SparkSession
      .builder()
      .master("local")
      .appName("RDDs")
      .config("spark.some.config.option", "some-value")
      .getOrCreate()

    spark.sparkContext.setLogLevel("WARN")

    val txt_file =spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt")


    println("Below is the program to read a text file and print the number of rows
of data in the document")

    val row_count = txt_file.count()

    println("row count "+row_count)
```

Output :

Below is the program to read a text file and print the number of rows of data in the document

row count **22**

Assignment 19.1
RDD Deep Dive

2. Write a program to read a text file and print the number of words in the document.

Here we have split data by comma (‘, ’) and similarly we have used count() function and printed the number of words in the document.

Code :

```
println("Below is the program to read a text file and print the number of words in
the document")

val word_count = txt_file.flatMap(x => x.split(",")).count()

println("word count "+word_count)
```

Output :

```
Below is the program to read a text file and print the number of words in the document

word count 110
```

3. We have a document where the word separator is -, instead of space. Write a spark

code, to obtain the count of the total number of words present in the document.

Here we have split data by ‘, ’ and similarly we have used count() function and printed the number of words in the document.

Code :

```
println("Below is the program to read a text file and print the number of words in
the document where word separator is '-'")

val word_count_with_hypen = txt_file.flatMap(x => x.split("-")).count()

println("word count with '-' as separator "+word_count_with_hypen)
```

Output :

```
Below is the program to read a text file and print the number of words in the document where word
separator is '-'

word count with '-' as separator 44
```

Assignment 19.1
RDD Deep Dive

Task 2 :

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.

1. Read the text file, and create a tupled rdd.

Code :

```
val tupledRDD =  spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt").map(x =>

(x.split(",")(0),(x.split(",")(1),x.split(",")(2),x.split(",")(3).toInt,x.split(",")(4).toInt)))

 println("Below we read the text file and create a tupled rdd")

 tupledRDD.foreach(println)
```

Output :

Below we read the text file and create a tupled rdd

```
(Mathew,(science,grade-3,45,12))
(Mathew,(history,grade-2,55,13))
(Mark,(maths,grade-2,23,13))
(Mark,(science,grade-1,76,13))
(John,(history,grade-1,14,12))
(John,(maths,grade-2,74,13))
(Lisa,(science,grade-1,24,12))
(Lisa,(history,grade-3,86,13))
(Andrew,(maths,grade-1,34,13))
(Andrew,(science,grade-3,26,14))
(Andrew,(history,grade-1,74,12))
(Mathew,(science,grade-2,55,12))
(Mathew,(history,grade-2,87,12))
(Mark,(maths,grade-1,92,13))
(Mark,(science,grade-2,12,12))
(John,(history,grade-1,67,13))
(John,(maths,grade-1,35,11))
(Lisa,(science,grade-2,24,13))
(Lisa,(history,grade-2,98,15))
(Andrew,(maths,grade-1,23,16))
(Andrew,(science,grade-3,44,14))
(Andrew,(history,grade-2,77,11))
```

Assignment 19.1
RDD Deep Dive

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

Code :

```
println("Below is the count of total number of rows present")

val tupledRDD_count =   tupledRDD.count()

println("Row count of tupled RDD is "+tupledRDD_count)
```

Output :

```
Below is the count of total number of rows present
Row count of tupled RDD is 22
```

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

Code :

```
println("Below is the distinct number of subjects present in the entire school")

val subjectRDD = spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt").map(x => (x.split(",")(1)))

val distint_subjects = subjectRDD.map(x => (x,1)).reduceByKey((a,b) => a +
b).foreach(println)
```

Output :

```
Below is the distinct number of subjects present in the entire school
(maths,6)
(history,8)
(science,8)
```

Assignment 19.1
RDD Deep Dive

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

Code :

```
println("Below is the count of the number of students in the school, whose name is
Mathew and marks is 55")

val studentRDD = spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt").map(x =>
  ((x.split(",")(0),x.split(",")(3).toInt),1))

val student_count =  studentRDD.filter(x => x._1._1 == "Mathew" && (x._1._2 == 55
)).reduceByKey((x,y) => x + y).foreach(println)
```

Output :

```
Below is the count of the number of students in the school, whose name is Mathew and marks is
55
((Mathew,55),2)
```

Assignment 19.1
RDD Deep Dive

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?

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

Code:

```
  println("Below is the count of students per grade")

 val  student_count_grd = spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt").map(x=> x.split(",")(2)).map(x => (x,1))

 val grd_reduce = student_count_grd.reduceByKey((x,y) => x + y).foreach(println)
```

Output :

```
Below is the count of students per grade
(grade-3,4)
(grade-1,9)
(grade-2,9)
```

Assignment 19.1
RDD Deep Dive

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

Code:

```
 println("Below is the average of each student (Note - Mathew is grade-1, is
different from Mathew in some other grade!) ")

 val averageRDD = spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt").map(x=>((x.split(",")(0),x.split(",")(2)),x.s
plit(",")(3).toInt))


 val grade = averageRDD.mapValues(x=>(x,1))


 val gradeReduce = grade.reduceByKey((x,y)=> (x._1+y._1,x._2+y._2))

 val grade_reduce_fnl = gradeReduce.mapValues{case(sum,count) =>
((1.0*sum)/count)}.foreach(println)
```

Output :

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

((Lisa,grade-1),24.0)

((Mark,grade-2),17.5)

((Lisa,grade-2),61.0)

((Mathew,grade-3),45.0)

((Andrew,grade-2),77.0)

((Andrew,grade-1),43.666666666666664)

((Lisa,grade-3),86.0)

((John,grade-1),38.666666666666664)

((John,grade-2),74.0)

((Mark,grade-1),84.0)

((Andrew,grade-3),35.0)

((Mathew,grade-2),65.66666666666667)
```

Assignment 19.1
RDD Deep Dive

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

Code:

```
 println("Below is the average score of students in each subject across all
grades")

 val  avg_score_stud = spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt").map(x=>
((x.split(",")(0),x.split(",")(1)),x.split(",")(3).toInt)).mapValues(y => (y,1))


val avg_reduce = avg_score_stud.reduceByKey((x,y) => (x._1 + y._1, x._2 + y._2))

 val avg_reduce_fnl = avg_reduce.mapValues{case(sum,count) =>
((1.0*sum)/count)}.foreach(println)
```

Output :

```
Below is the average score of students in each subject across all grades

((Lisa,history),92.0)

((Mark,maths),57.5)

((Andrew,science),35.0)

((Mark,science),44.0)

((Mathew,science),50.0)

((Andrew,maths),28.5)

((Mathew,history),71.0)

((John,maths),54.5)

((John,history),40.5)

((Lisa,science),24.0)

((Andrew,history),75.5)
```

Assignment 19.1
RDD Deep Dive

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

Code :

```
println("Below is the average score of students in each subject per grade")

val averageRDDFile = spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt")

val averageRDD1 = averageRDDFile.map((x => ((x.split(",")(0),
x.split(",")(1),x.split(",")(2)), x.split(",")(3).toInt)))

val averageRDDMapVal = averageRDD1.mapValues(x => (x,1))

val averageRDDReduceKey = averageRDDMapVal.reduceByKey((x,y) => (x._1 + y._1, x._2
+ y._2))

val averageRDDFinal = averageRDDReduceKey.mapValues{case(sum,count) =>
(1.0*sum/count)}.foreach(println)
```

Output :

```
Below is the average score of students in each subject per grade

((Lisa,history,grade-3),86.0)

((John,history,grade-1),40.5)

((Andrew,history,grade-2),77.0)

((John,maths,grade-2),74.0)

((Andrew,maths,grade-1),28.5)

((Mark,maths,grade-2),23.0)

((Mark,science,grade-2),12.0)

((Andrew,science,grade-3),35.0)

((Mathew,science,grade-3),45.0)

((Mathew,history,grade-2),71.0)

((Andrew,history,grade-1),74.0)

((John,maths,grade-1),35.0)

((Mark,maths,grade-1),92.0)

((Mark,science,grade-1),76.0)

((Mathew,science,grade-2),55.0)

((Lisa,science,grade-2),24.0)

((Lisa,history,grade-2),98.0)

((Lisa,science,grade-1),24.0)
```

Assignment 19.1
RDD Deep Dive

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

Code :

```
println("Below is list from students in grade-2, having average score greater than
50  ")

val  avg50Grade2File = spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt")

val  avg50Grade3 = avg50Grade2File.map(x => ((( x.split(",")(0), x.split(",")(2)),
x.split(",")(3).toInt)))

val  avg50Grade2MapV = avg50Grade3.mapValues( x => (x,1))

val  avg50Grade2Reduce  = avg50Grade2MapV.reduceByKey((x,y) => (x._1 + y._1, x._2 +
y._2))

val RDDavg = avg50Grade2Reduce.mapValues{case(sum,count)=>(1.0*sum)/count}

val  avg50Grade2Filter = RDDavg.filter( x => x._1._2 == "grade-2" && x._2 >
50).foreach(println)
```

Output :

```
Below is list from students in grade-2, having average score greater than 50

((Lisa,grade-2),61.0)

((Andrew,grade-2),77.0)

((John,grade-2),74.0)

((Mathew,grade-2),65.66666666666667)
```

Assignment 19.1
RDD Deep Dive

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

Code :

```
println("Below is list of students who satisfies condition of having Average score
per student_name across all grades is same as average score per grade")

val studentAvg = spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt").map(x=>(x.split(",")(0),x.split(",")(3).toInt
))

val studentAvgMapV = studentAvg.mapValues(x=>(x,1))

 val studentReduce = studentAvgMapV.reduceByKey((x,y)=> (x._1+y._1,x._2+y._2))

 val AvgStudent = studentReduce.mapValues{case (sum,count) => (1.0 * sum)/count}

val AvgStudentExt = AvgStudent.map(x=> x._1 + "," + x._2)



val StudAvgPerGrade = spark.sparkContext.textFile("C:\\AcadGild
Hadoop\\Assignments\\19_Dataset.txt").map(x=>((x.split(",")(0),x.split(",")(2)),
x.split(",")(3).toInt))

val StudAvgPerGradeMapV = StudAvgPerGrade.mapValues(x=>(x,1))

val StudAvgPerGradeReduce = StudAvgPerGradeMapV.reduceByKey((x,y)=>
(x._1+y._1,x._2+y._2))

val AvgStudentPerGrade = StudAvgPerGradeReduce.mapValues{case (sum,count) => (1.0 *
sum)/count}

val AvgStudentPerGradeExt = AvgStudentPerGrade.map(x=> x._1._1 + "," +
x._2.toDouble)

val commanval = AvgStudentPerGradeExt.intersection(AvgStudentExt).foreach(println)
```

Output :

Below is list of students who satisfies condition of having Average score per student_name across all grades is same as average score per grade

(Here , we did not get any output, no rows)