

Task 1

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

The screenshot shows the IntelliJ IDEA interface. In the code editor, a Scala file named `Assignment19.scala` is open. The code defines a `main` method that creates a `SparkSession`, reads a text file from `F:\PDF Architect\Assignment19_DataSet.txt`, and prints the line count. The line count is highlighted with a yellow background.

```
1 import org.apache.spark.sql.SparkSession
2
3 object Class19Task {
4   def main(args: Array[String]): Unit = {
5     val sparkSession = SparkSession.builder.master( master = "local")
6       .appName( name = "spark session example")
7       .getOrCreate()
8     val sparkContext = sparkSession.sparkContext
9     val localFile = sparkContext.textFile( path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
10    localFile.foreach(println)
11    val lineCount = localFile.count()
12    println(lineCount)
13  }
14 }
```

In the run output window, the application's log is displayed. It shows the application starting, reading the file, and printing each line of the dataset. The final output shows the total line count of 837 bytes sent to the driver.

```
18/07/30 19:55:02 INFO TaskSetManager: Starting task 0.0 in stage 0.0 (TID 0, localhost, executor driver, partition 0, PROCESS_LOCAL, 7894 bytes)
18/07/30 19:55:02 INFO Executor: Running task 0.0 in stage 0.0 (TID 0)
18/07/30 19:55:02 INFO HadoopRDD: Input split: file:/F:/PDF Architect/Assignment19_DataSet.txt:0+624
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
18/07/30 19:55:02 INFO Executor: Finished task 0.0 in stage 0.0 (TID 0). 837 bytes result sent to driver
```

```
18/07/30 19:55:02 INFO DAGScheduler: MISSING parents: List()
18/07/30 19:55:02 INFO DAGScheduler: Submitting ResultStage 1 (F:\\PDF Architect\\Assignment19_DataSet.txt MapPartitionsRDD[1] at textFile at Assignment19.scala:9), which has no missing parents
18/07/30 19:55:02 INFO MemoryStore: Block broadcast_2 stored as values in memory (estimated size 3.1 KB, free 350.2 MB)
18/07/30 19:55:02 INFO MemoryStore: Block broadcast_2_piece0 stored as bytes in memory (estimated size 1929.0 B, free 350.2 MB)
22
18/07/30 19:55:02 INFO BlockManagerInfo: Added broadcast_2_piece0 in memory on 192.168.1.8:57824 (size: 1929.0 B, free: 350.4 MB)
18/07/30 19:55:02 INFO SparkContext: Created broadcast 2 from broadcast at DAGScheduler.scala:1039
18/07/30 19:55:02 INFO DAGScheduler: Submitting 1 missing tasks from ResultStage 1 (F:\\PDF Architect\\Assignment19_DataSet.txt MapPartitionsRDD[1] at textFile at Assignment19.scala:9) (first 10)
18/07/30 19:55:02 INFO TaskSchedulerImpl: Adding task set 1.0 with 1 tasks
18/07/30 19:55:02 INFO TaskSetManager: Starting task 0.0 in stage 1.0 (TID 1, localhost, executor driver, partition 0, PROCESS_LOCAL, 7894 bytes)
```

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

The screenshot shows a Scala development environment. At the top, there are tabs for 'Assignment19.scala' and 'Casestudy911.scala'. The code editor displays 'Assignment19.scala' with the following content:

```
1 import org.apache.spark.sql.SparkSession
2
3 object Class19Task {
4     def main(args: Array[String]): Unit = {
5         val sparkSession = SparkSession.builder.master( master = "local")
6             .appName( name = "spark session example")
7             .getOrCreate()
8         val sparkContext = sparkSession.sparkContext
9         val localFile = sparkContext.textFile( path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
10        val words = localFile.flatMap(_.split( regex = "," ))
11        val wordCount = words.map( x=> (x,1)).reduceByKey(_+_)
12        wordCount.foreach( println )
13    }
14 }
```

The output window below shows the results of the execution:

```
Run: Class19Task
18/07/30 20:21:30 INFO ShutdownHookManager: Deleting directory C:\Users\Anupama Stanley\AppData\Local\Temp\spark-a94b698e-3a42-4570-85fe-cf00d0284062
(maths,6)
(14,3)
(67,1)
(98,1)
(15,1)
(Mark,4)
(grade-3,4)
(grade-1,9)
(35,1)
(history,8)
(Andrew,6)
(45,1)
(55,2)
(Mathew,4)
(16,1)
(86,1)
(92,1)
(34,1)
(science,8)
(26,1)
(87,1)
(grade-2,9)
(44,1)
(12,8)
(13,9)
(24,2)
(76,1)
(John,4)
(77,1)
(74,2)
(11,2)
(Lisa,4)
(23,2)
```

At the bottom, a message indicates the compilation was successful.

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 import org.apache.spark.sql.SparkSession
2
3 object Class19Task {
4   def main(args: Array[String]): Unit = {
5     val sparkSession = SparkSession.builder.master( master = "local")
6       .appName( name = "spark session example")
7       .getOrCreate()
8     val sparkContext = sparkSession.sparkContext
9     val localFile = sparkContext.textFile( path = "F:\\PDF Architect\\Assignment19_DataSet1.txt")
10    localFile.foreach(println)
11    val words = localFile.flatMap(_.split( regex = "[\\W]+"))
12    val wordCount = words.map( x=> (x,1)).reduceByKey(_+_)
13    wordCount.foreach( println )
14  }
15 }
```

Input File:

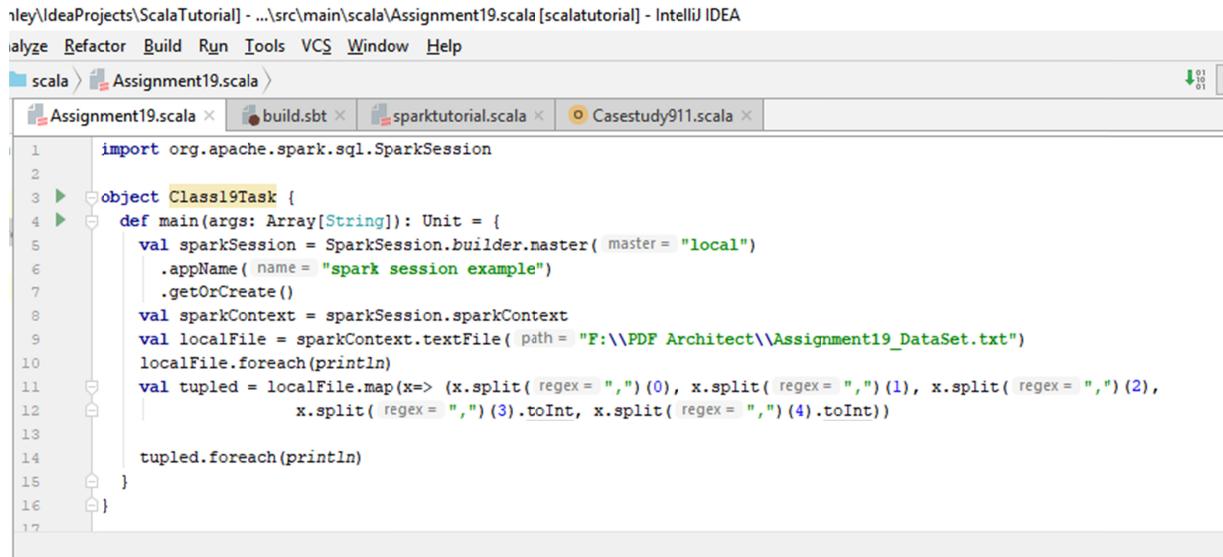
```
18/07/30 20:32:25 INFO TaskSchedulerImpl: Adding task set 0.0 with 1 tasks
18/07/30 20:32:29 INFO TaskSetManager: Starting task 0.0 in stage 0.0 (TID 0, localhost, executor drive
18/07/30 20:32:29 INFO Executor: Running task 0.0 in stage 0.0 (TID 0)
18/07/30 20:32:30 INFO HadoopRDD: Input split: file:/F:/PDF Architect/Assignment19_DataSet1.txt:0+140
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
18/07/30 20:32:30 INFO Executor: Finished task 0.0 in stage 0.0 (TID 0). 837 bytes result sent to drive
18/07/30 20:32:30 INFO TaskSetManager: Finished task 0.0 in stage 0.0 (TID 0) in 277 ms on localhost (e
18/07/30 20:32:30 INFO TaskSchedulerImpl: Removed TaskSet 0.0, whose tasks have all completed, from pool
```

```
Run: Class19Task
18/07/30 20:32:31 INFO BlockManagerInfo: Removed broadcast_2_piece0 on 192.168.1.8:58306 in memory (size: 2.8 KB, free: 350.4 MB)
18/07/30 20:32:31 INFO TaskSetManager: Finished task 0.0 in stage 2.0 (TID 2) in 139 ms on localhost (executor driver) (1/1)
18/07/30 20:32:31 INFO DAGScheduler: ResultStage 2 (foreach at Assignment19.scala:13) finished in 0.192 s
(math,1)
(14,1)
(Mark,2)
(history,2)
(45,1)
(55,1)
(Mathew,2)
(grade_1,2)
(science,2)
(John,1)
(12,2)
(13,3)
(76,1)
(grade_2,2)
(grade_3,1)
(23,1)
18/07/30 20:32:31 INFO TaskSchedulerImpl: Removed TaskSet 2.0, whose tasks have all completed, from pool
18/07/30 20:32:31 INFO DAGScheduler: Job 1 finished: foreach at Assignment19.scala:13, took 0.845802 s
```

Task 2

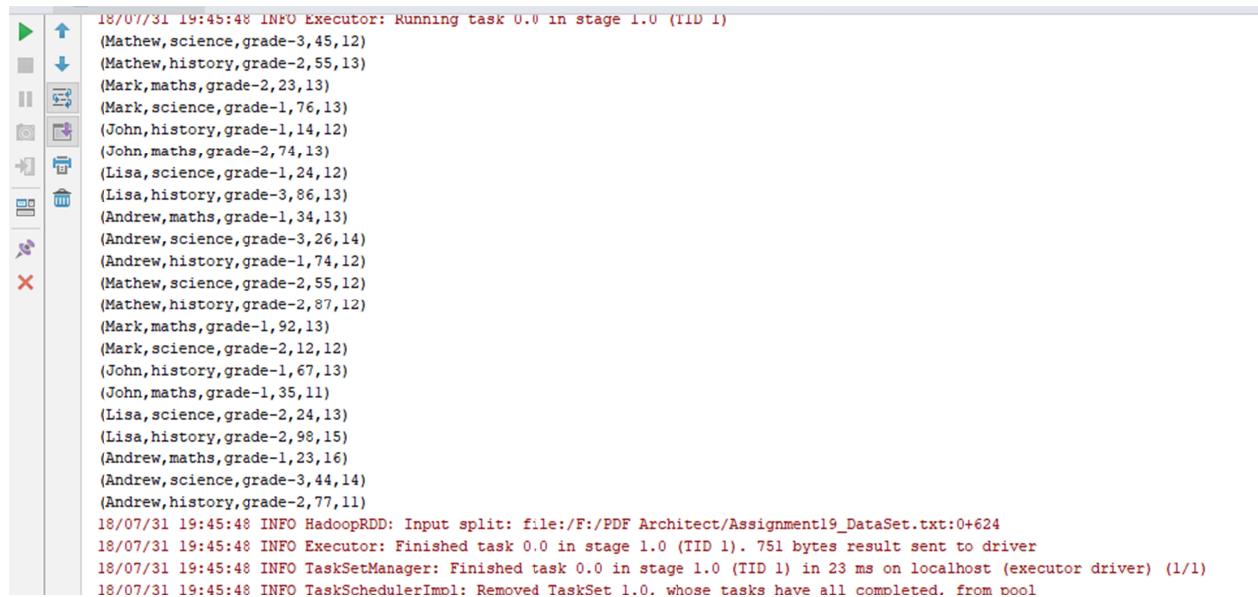
Problem Statement 1:

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



```
1 import org.apache.spark.sql.SparkSession
2
3 object Class19Task {
4   def main(args: Array[String]): Unit = {
5     val sparkSession = SparkSession.builder.master("local")
6       .appName("spark session example")
7       .getOrCreate()
8     val sparkContext = sparkSession.sparkContext
9     val localFile = sparkContext.textFile("F:\\PDF Architect\\Assignment19_DataSet.txt")
10    localFile.foreach(println)
11    val tupled = localFile.map(x=> (x.split(",")(0), x.split(",")(1), x.split(",")(2),
12      x.split(",")(3).toInt, x.split(",")(4).toInt))
13
14    tupled.foreach(println)
15  }
16}
17
```

Tupled rdd output



```
18/07/31 19:45:48 INFO Executor: Running task 0.0 in stage 1.0 (TID 1)
(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)
18/07/31 19:45:48 INFO HadoopRDD: Input split: file:/F:/PDF Architect/Assignment19_DataSet.txt:0+624
18/07/31 19:45:48 INFO Executor: Finished task 0.0 in stage 1.0 (TID 1). 751 bytes result sent to driver
18/07/31 19:45:48 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 1) in 23 ms on localhost (executor driver) (1/1)
18/07/31 19:45:48 INFO TaskSchedulerImpl: Removed TaskSet 1.0. whose tasks have all completed. from pool
```

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

File | IdeaProjects\ScalaTutorial] - ...\\src\\main\\scala\\Assignment19.scala [scalatutorial] - IntelliJ IDEA

Analyze Refactor Build Run Tools VCS Window Help

scala > Assignment19.scala

Assignment19.scala build.sbt sparktutorial.scala Casestudy911.scala

```

1 import org.apache.spark.sql.SparkSession
2
3 object Class19Task {
4   def main(args: Array[String]): Unit = {
5     val sparkSession = SparkSession.builder.master( master = "local")
6       .appName( name = "spark session example")
7       .getOrCreate()
8     val sparkContext = sparkSession.sparkContext
9     val localFile = sparkContext.textFile( path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
10    localFile.foreach(println)
11    val tupled = localFile.map(x=> (x.split( regex = ",")(0), x.split( regex = ",")(1), x.split( regex = ",") (2),
12      x.split( regex = ",") (3).toInt, x.split( regex = ",") (4).toInt))
13    val count = tupled.count()
14    println("No of rows in tupled rdd is " + count)
15  }
16}
17

```

18/07/31 19:48:42 INFO Executor: Running task 0.0 in stage 1.0 (TID 1)

18/07/31 19:48:42 INFO HadoopRDD: Input split: file:/F:/PDF Architect/Assignment19_DataSet.txt:0+624

No of rows in tupled rdd is 22

18/07/31 19:48:42 INFO Executor: Finished task 0.0 in stage 1.0 (TID 1). 832 bytes result sent to driver

18/07/31 19:48:42 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 1) in 24 ms on localhost (executor driver) (1/1)

18/07/31 19:48:42 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool

18/07/31 19:48:42 INFO DAGScheduler: ResultStage 1 (count at Assignment19.scala:13) finished in 0.047 s

18/07/31 19:48:42 INFO DAGScheduler: Job 1 finished: count at Assignment19.scala:13, took 0.054294 s

18/07/31 19:48:42 INFO SparkContext: Invoking stop() from shutdown hook

18/07/31 19:48:42 INFO SparkUI: Stopped Spark web UI at <http://192.168.1.8:4041>

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

18/07/31 20:02:48 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 8 ms

18/07/31 20:02:48 INFO Executor: Finished task 0.0 in stage 2.0 (TID 2). 1219 bytes result sent to driver

No of distinct subjects in school is 3

18/07/31 20:02:48 INFO TaskSetManager: Finished task 0.0 in stage 2.0 (TID 2) in 88 ms on localhost (executor driver) (1/1)

18/07/31 20:02:48 INFO TaskSchedulerImpl: Removed TaskSet 2.0, whose tasks have all completed, from pool

18/07/31 20:02:48 INFO DAGScheduler: ResultStage 2 (count at Assignment19.scala:14) finished in 0.119 s

18/07/31 20:02:48 INFO DAGScheduler: Job 1 finished: count at Assignment19.scala:14, took 0.962669 s

18/07/31 20:02:48 INFO SparkContext: Invoking stop() from shutdown hook

18/07/31 20:02:48 INFO SparkUI: Stopped Spark web UI at <http://192.168.1.8:4041>

18/07/31 20:02:48 INFO ContextCleaner: Cleaned accumulator 30

```

1
2
3 object Class19Task {
4   def main(args: Array[String]): Unit = {
5     val sparkSession = SparkSession.builder.master( master = "local")
6       .appName( name = "spark session example")
7       .getOrCreate()
8     val sparkContext = sparkSession.sparkContext
9     val localFile = sparkContext.textFile( path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
10    localFile.foreach(println)
11
12    val subjects = localFile.map( x=> x.split( regex = ",")(1))
13    val distinctSubjects = subjects.distinct()
14    val distinctCount = distinctSubjects.count()
15    println("No of distinct subjects in school is " + distinctCount)
16  }
17}
18

```

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

The screenshot shows a Scala IDE interface with several tabs at the top: Assignment19.scala, build.sbt, sparktutorial.scala, and Casestudy911.scala. The Assignment19.scala tab is active, displaying the following Scala code:

```
1 import org.apache.spark.sql.SparkSession
2
3 object Class19Task {
4   def main(args: Array[String]): Unit = {
5     val sparkSession = SparkSession.builder.master("local")
6       .appName("spark session example")
7       .getOrCreate()
8     val sparkContext = sparkSession.sparkContext
9     val localFile = sparkContext.textFile(path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
10
11     val students = localFile.map(x=> ((x.split(",")(0),x.split(",")(3).toInt),1))
12     val student = students.filter(x => x._1._1 == "Mathew" && x._1._2 == 55)
13     val studentagg = student.reduceByKey((x,y)=> x+y).foreach(println)
14   }
15 }
```

Below the code editor is a log window showing the execution output:

```
18/07/31 20:25:43 INFO ShuffleBlockFetcherIterator: Getting 1 non-empty blocks out of 1 blocks
18/07/31 20:25:43 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 10 ms
((Mathew,55),2)
18/07/31 20:25:43 INFO Executor: Finished task 0.0 in stage 1.0 (TID 1). 1138 bytes result sent to driver
18/07/31 20:25:43 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 1) in 116 ms on localhost (executor driver) (1/1)
18/07/31 20:25:43 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
18/07/31 20:25:43 INFO DAGScheduler: ResultStage 1 (foreach at Assignment19.scala:13) finished in 0.193 s
18/07/31 20:25:43 INFO DAGScheduler: Job 0 finished: foreach at Assignment19.scala:13, took 1.209210 s
18/07/31 20:25:43 INFO SparkContext: Invoking stop() from shutdown hook
18/07/31 20:25:43 INFO SparkUI: Stopped Spark web UI at http://192.168.1.8:4041
18/07/31 20:25:43 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
18/07/31 20:25:43 INFO MemoryStore: MemoryStore cleared
18/07/31 20:25:43 INFO BlockManager: BlockManager stopped
```

Problem Statement 2:

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

The screenshot shows a Scala IDE interface with several tabs at the top: scala, Assignment19.scala, build.sbt, sparktutorial.scala, and Casestudy911.scala. The Assignment19.scala tab is active, displaying the following Scala code:

```
1
2 object Class19Task {
3   def main(args: Array[String]): Unit = {
4     val sparkSession = SparkSession.builder.master("local")
5       .appName("spark session example")
6       .getOrCreate()
7     val sparkContext = sparkSession.sparkContext
8     val localFile = sparkContext.textFile(path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
9     val grade = localFile.map(x=> (x.split(",")(2),1)).reduceByKey((x,y)=> x+y)
10    grade.foreach(println)
11  }
12 }
13
14
15 }
```

The screenshot shows a Scala IDE interface with several tabs at the top: scala, Assignment19.scala, build.sbt, sparktutorial.scala, and Casestudy911.scala. The Assignment19.scala tab is active, displaying the following Scala code:

```
1
2 object Class19Task {
3   def main(args: Array[String]): Unit = {
4     val sparkSession = SparkSession.builder.master("local")
5       .appName("spark session example")
6       .getOrCreate()
7     val sparkContext = sparkSession.sparkContext
8     val localFile = sparkContext.textFile(path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
9     val grade = localFile.map(x=> (x.split(",")(2),1)).reduceByKey((x,y)=> x+y)
10    grade.foreach(println)
11  }
12 }
13
14
15 }
```

Below the code editor is a log window showing the execution output:

```
18/07/31 20:32:56 INFO ShuffleBlockFetcherIterator: Getting 1 non-empty blocks out of 1 blocks
18/07/31 20:32:56 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 8 ms
(grade-3,4)
(grade-1,9)
(grade-2,9)
18/07/31 20:32:56 INFO Executor: Finished task 0.0 in stage 1.0 (TID 1). 1138 bytes result sent to driver
18/07/31 20:32:56 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 1) in 76 ms on localhost (executor driver) (1/1)
18/07/31 20:32:56 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
18/07/31 20:32:56 INFO DAGScheduler: ResultStage 1 (foreach at Assignment19.scala:11) finished in 0.117 s
```

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

```

3  object Class19Task {
4    def main(args: Array[String]): Unit = {
5      val sparkSession = SparkSession.builder.master("local")
6        .appName("spark session example")
7        .getOrCreate()
8      val sparkContext = sparkSession.sparkContext
9      val localFile = sparkContext.textFile(path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
10     val file = localFile.map(x=> ((x.split(regex = ",") (0),x.split(regex = ",") (2)),x.split(regex = ",") (3).toInt))
11     val fileMap = file.mapValues(x=> (x,1))
12     val fileReduce = fileMap.reduceByKey((x,y)=>(x._1+y._1, x._2+ y._2))
13     val avg = fileReduce.mapValues{case(sum, count) => (1.0*sum)/count }
14     avg.foreach(println)
15   }
16 }

```

18/07/31 20:42:32 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
18/07/31 20:42:32 INFO DAGScheduler: ResultStage 1 (foreach at Assignment19.scala:14) finished in 0.105 s
((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),38.66666666666664)
((John,grade-2),74.0)
((Mark,grade-1),84.0)
((Andrew,grade-3),35.0)
((Mathew,grade-2),65.66666666666667)
18/07/31 20:42:32 INFO DAGScheduler: Job 0 finished: foreach at Assignment19.scala:14, took 1.249716 s
18/07/31 20:42:32 INFO SparkContext: Invoking stop() from shutdown hook

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

```

import org.apache.spark.sql.SparkSession

object Class19Task {
  def main(args: Array[String]): Unit = {
    val sparkSession = SparkSession.builder.master("local")
      .appName("spark session example")
      .getOrCreate()
    val sparkContext = sparkSession.sparkContext
    val localFile = sparkContext.textFile(path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
    val grade = localFile.map(x=> ((x.split(regex = ",") (0),x.split(regex = ",") (1)),(x.split(regex = ",") (3).toInt,1)))
    val gradeReduce = grade.reduceByKey((x,y)=>(x._1+y._1, x._2+ y._2))
    val avg = gradeReduce.mapValues{case(x,y) => (x.toFloat)/y }.sortByKey()
    avg.foreach(println)
  }
}

```

18/07/31 20:48:09 INFO ShuffleBlockFetcherIterator: Getting 1 non-empty blocks out of 1 blocks
18/07/31 20:48:09 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 0 ms
((Andrew,history),75.5)
18/07/31 20:48:10 INFO Executor: Finished task 0.0 in stage 2.0 (TID 2). 1095 bytes result sent to driver
((Andrew,maths),28.5)
((Andrew,science),35.0)
((John,history),40.5)
((John,maths),54.5)
((Lisa,history),92.0)
((Lisa,science),24.0)
((Mark,maths),57.5)
((Mark,science),44.0)
((Mathew,history),71.0)
((Mathew,science),50.0)
18/07/31 20:48:10 INFO TaskSetManager: Finished task 0.0 in stage 2.0 (TID 2) in 33 ms on localhost (executor driver) (1/1)

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

```

1
2
3  object Class19Task {
4    def main(args: Array[String]): Unit = {
5      val sparkSession = SparkSession.builder.master( master = "local")
6        .appName( name = "spark session example")
7        .getOrCreate()
8      val sparkContext = sparkSession.sparkContext
9      val localFile = sparkContext.textFile( path = "F:\\\\PDF Architect\\\\Assignment19_DataSet.txt")
10     val grade = localFile.map(x=> ((x.split( regex = "," )(1),x.split( regex = "," )(2)),(x.split( regex = "," )(3).toInt,1)))
11     val gradeReduce = grade.reduceByKey((x,y)=>(x._1+y._1, x._2+y._2))
12     val avg = gradeReduce.mapValues{case(x,y) => (x.toFloat)/y }.sortByKey()
13     avg.foreach(println)
14   }
15 }
16

```

Output window:

```

18/07/31 20:51:52 INFO Executor: Running task 0.0 in stage 2.0 (TID 2)
18/07/31 20:51:52 INFO ShuffleBlockFetcherIterator: Getting 1 non-empty blocks out of 1 blocks
18/07/31 20:51:52 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 1 ms
((history,grade-1),51.666668)
((history,grade-2),79.25)
((history,grade-3),86.0)
((maths,grade-1),46.0)
((maths,grade-2),48.5)
((science,grade-1),50.0)
((science,grade-2),30.333334)
((science,grade-3),38.333332)
18/07/31 20:51:52 INFO Executor: Finished task 0.0 in stage 2.0 (TID 2). 1095 bytes result sent to driver
18/07/31 20:51:52 INFO TaskSetManager: Finished task 0.0 in stage 2.0 (TID 2) in 36 ms on localhost (executor driver) (1/1)
18/07/31 20:51:52 INFO TaskSchedulerImpl: Removed TaskSet 2.0, whose tasks have all completed, from pool
18/07/31 20:51:52 INFO DAGScheduler: ResultStage 2 (foreach at Assignment19.scala:13) finished in 0.055 s

```

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

```

1
2
3  object Class19Task {
4    def main(args: Array[String]): Unit = {
5      val sparkSession = SparkSession.builder.master( master = "local")
6        .appName( name = "spark session example")
7        .getOrCreate()
8      val sparkContext = sparkSession.sparkContext
9      val localFile = sparkContext.textFile( path = "F:\\\\PDF Architect\\\\Assignment19_DataSet.txt")
10     val grade = localFile.map(x=> ((x.split( regex = "," )(0),x.split( regex = "," )(2)),(x.split( regex = "," )(3).toInt,1)))
11     val gradeReduce = grade.reduceByKey((x,y)=>(x._1+y._1, x._2+y._2))
12     val avg = gradeReduce.mapValues{case(x,y) => (x.toFloat)/y }.sortByKey()
13     val avgGrade2 = avg.filter(x=> x._1._2 == "grade-2" && x._2 > 50 )
14     avgGrade2.foreach(println)
15     println("No of Students with avg score greater than 50 is : "+avgGrade2.count() )
16   }
17

```

Output window:

```

18/07/31 20:55:22 INFO Executor: Running task 0.0 in stage 2.0 (TID 2)
18/07/31 20:55:22 INFO ShuffleBlockFetcherIterator: Getting 1 non-empty blocks out of 1 blocks
18/07/31 20:55:22 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 0 ms
((Andrew,grade-2),77.0)
((John,grade-2),74.0)
((Lisa,grade-2),61.0)
((Mathew,grade-2),65.666664)
18/07/31 20:55:22 INFO Executor: Finished task 0.0 in stage 2.0 (TID 2). 1095 bytes result sent to driver
18/07/31 20:55:22 INFO TaskSetManager: Finished task 0.0 in stage 2.0 (TID 2) in 40 ms on localhost (executor driver) (1/1)
18/07/31 20:55:22 INFO TaskSchedulerImpl: Removed TaskSet 2.0, whose tasks have all completed, from pool
18/07/31 20:55:22 INFO DAGScheduler: ResultStage 2 (foreach at Assignment19.scala:14) finished in 0.058 s

```

```

18/07/31 21:00:00 INFO TaskSchedulerImpl: Cancelling task 0.0 in stage 5.0 (TID 3)
18/07/31 21:00:00 INFO Executor: Running task 0.0 in stage 5.0 (TID 3)
18/07/31 21:00:00 INFO ShuffleBlockFetcherIterator: Getting 1 non-empty blocks out of 1 blocks
18/07/31 21:00:00 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 0 ms
No of Students with avg score greater than 50 is : 4
18/07/31 21:00:00 INFO Executor: Finished task 0.0 in stage 5.0 (TID 3). 1176 bytes result sent to driver
18/07/31 21:00:00 INFO TaskSetManager: Finished task 0.0 in stage 5.0 (TID 3) in 29 ms on localhost (executor driver) (1/1)
18/07/31 21:00:00 INFO TaskSchedulerImpl: Removed TaskSet 23.0, whose tasks have all completed, from pool

```

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

```

Assignment19.scala
Assignment19.scala build.sbt sparktutorial.scala Casestudy911.scala

object Class19Task {
  def main(args: Array[String]): Unit = {
    val sparkSession = SparkSession.builder.master( master = "local")
      .appName( name = "spark session example")
      .getOrCreate()
    val sparkContext = sparkSession.sparkContext
    val localFile = sparkContext.textFile( path = "F:\\PDF Architect\\Assignment19_DataSet.txt")
    val fileRDD = localFile.map( x =>x.split( regex = ","))
    val studentsl = fileRDD.map( x => ((x(2),x(0)),x(3).toInt))
    val suml = studentsl.distinct.groupByKey().mapValues(x => x.sum)
    suml.foreach(x => println(x))
    val students2 = fileRDD.map(x=> ((x(2), x(0)),1))
    val countl = students2.groupByKey().mapValues( x => x.sum )
    countl.foreach( x => println(x))
    val joinedRDD = suml.join(countl)
    joinedRDD.foreach( x => println(x))
    val avgStd = joinedRDD.map( x => ((x._1),(x._2._1/x._2._2)))
    val avgGrade = avgStd.map( x => (x._1._2, x._2 ))
    val sumPerStd = fileRDD.map( x => ( (x(0), x(3).toInt)).groupByKey().mapValues( y => y.sum))
    sumPerStd.foreach( x => println(x))
    val countStd = fileRDD.map( x => (x(0), 1)).groupByKey().mapValues( y => y.sum )
    countStd.foreach( x => println(x))
    val avgStdName = sumPerStd.join(countStd).map( x => (x._1, x._2._1/x._2._2))
    avgStdName.foreach( x => println(x))
    val commonStd = avgStdName.intersection(avgGrade)
    println( "count of Students with same avg per name and grade " + commonStd.count())
    commonStd.foreach( x => println("Students with same avg per name and grade " +x))
  }
}

```

```

18/07/31 23:07:23 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 0 ms
18/07/31 23:07:23 INFO ShuffleBlockFetcherIterator: Getting 1 non-empty blocks out of 1 blocks
18/07/31 23:07:23 INFO ShuffleBlockFetcherIterator: Started 0 remote fetches in 1 ms
18/07/31 23:07:23 INFO Executor: Finished task 0.0 in stage 23.0 (TID 13). 1176 bytes result sent to driver
18/07/31 23:07:23 INFO TaskSetManager: Finished task 0.0 in stage 23.0 (TID 13) in 18 ms on localhost (executor driver) (1/1)
18/07/31 23:07:23 INFO TaskSchedulerImpl: Removed TaskSet 23.0, whose tasks have all completed, from pool
count of Students with same avg per name and grade 0
18/07/31 23:07:23 INFO DAGScheduler: ResultStage 23 (count at Assignment19.scala:28) finished in 0.030 s
18/07/31 23:07:23 INFO DAGScheduler: Job 6 finished: count at Assignment19.scala:28, took 0.202995 s
18/07/31 23:07:23 INFO SparkContext: Starting job: foreach at Assignment19.scala:29

```