Session 17

Assignment 2

|  |  |
| --- | --- |
| **Prepared For:** | AcadGild |
|  |  |
| **Document Approval:** | **AcadGild** |
|  |  |
|  |  |
|  |  |
|  |  |
| **Project Title:** | Session 17 – Assignment 2 |
|  |  |
| **Prepared By:** | Duncan Burgess |
|  |  |
|  | dburgess@duncb.com |
|  |  |
| **Primary Engineer:** | Duncan Burgess |
|  |  |
| **Document Reference:** | **Session 17 – Assignment 2** |
|  |  |
| **Start Date:** | 16/10/2017 |
|  |  |
|  |  |



# 

# Contents

[Contents 2](#_Toc495962518)

[Change History 3](#_Toc495962519)

[1. Problem Statement 4](#_Toc495962520)

[2. Dataset 4](#_Toc495962521)

[3. Solution 5](#_Toc495962522)

[4. Result 7](#_Toc495962523)

# Change History

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Document Revision** | **Date** | **Authored By** | **Authorised By** | **Sections Affected** | **Reason for Change** |
| Rev 01 | 16/10/2017 | Duncan Burgess |  | All | Initial release. |
|  |  |  |  |  |  |
|  |  |  |  |  |  |
|  |  |  |  |  |  |

# Problem Statement

Given a dataset of college students as a text file (name, subject, grade, marks) :

Dataset

**Problem Statement 1:**

* Read the text file, and create a tupled rdd.
* Find the count of total number of rows present.
* What is the distinct number of subjects present in the entire school
* What is the count of the number of students in the school, whose name is Mathew and marks is 55

**Problem Statement 2:**

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

**Problem Statement 3:**

* Are there any students in the college that satisfy the below criteria :
* Average score per student\_name across all grades is same as average score per student\_name per grade

Hint - Use Intersection Property.

# Dataset

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

# Solution

To enhance my learning I have again written a solution that answers all questions in one using the supplied dataset.

**Code created**

***package*** *com.duncb.spark*

***import*** *org.apache.spark.\_*

***import*** *org.apache.spark.SparkContext.\_*

***import*** *org.apache.log4j.\_*

*//\*\* Compute the the following*

***object*** *studentData {*

*/\*\* A function that splits a line of input into (name,subject,grade,mark) tuples. \*/*

***def*** *parseLine(line: String) = {*

*// Split by commas*

***val*** *fields = line.split(",")*

*// Extract the name, subject and grade to Strings and marks to integers*

***val*** *name = fields(0).toString*

***val*** *subject = fields(1).toString*

***val*** *grade = fields (2) .toString*

***val*** *marks = fields (3) .toInt*

***val*** *score = fields (4) .toInt*

*// Create a tuple that is our result.*

*(name,subject,grade,marks,score)*

*}*

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

*// Set the log level to only print errors*

*Logger.getLogger("org").setLevel(Level.ERROR)*

*// Create a SparkContext using every core of the local machine*

***val*** *sc =* ***new*** *SparkContext("local[\*]", "StudentData")*

*// Load each line of the source data into an RDD*

***val*** *lines = sc.textFile("file:///N:/Datasets/17.2\_Dataset.txt")*

***val*** *rdd = lines.map(parseLine)*

*rdd.foreach(println)*

***val*** *numberOfLInes= rdd.count*

*println("Number of lines = " + numberOfLInes)*

***val*** *subjects = rdd.map(x=>x.\_2).distinct*

*subjects.foreach (println)*

***val*** *numberOfSubjects= subjects.count*

*println ("Number of subjects is " + numberOfSubjects)*

***val*** *Mathew55 = rdd.map(x=>(x.\_1,x.\_2,x.\_4)).filter(x => x.\_1=="Mathew" && x.\_3 == 55).collect*

*println (Mathew55)*

***val*** *countMatt55 = rdd.map(x=>(x.\_1,x.\_4)).filter(x => x.\_1=="Mathew" && x.\_2== 55).count*

*println ("Number of students called Mathew who scoresd 55 is " + countMatt55)*

***val*** *studentsPerGrade= rdd.map(x=>(x.\_3,x.\_1)).distinct.map(x=>(x.\_1,1)).reduceByKey((x,y)=> x+y)*

*println ("No of students per Grade")*

*studentsPerGrade.foreach(println)*

*println ("Average of each Student in each subject all grades")*

***val*** *averageScore = rdd.map(x =>((x.\_1,x.\_2),(x.\_4 + x.\_5))).groupByKey.map (x=> (x.\_1,(x.\_2.sum.toDouble/(x.\_2.size\*2)))).sortByKey()*

*averageScore.foreach(println)*

*println ("Average score Student each Student Grade")*

***val*** *averageScoreStudGrad = rdd.map(x=> ((x.\_1,x.\_2,x.\_3),(x.\_4 + x.\_5))).groupByKey.map(x=> (x.\_1,(x.\_2.sum.toDouble /(x.\_2.size \*2)))).sortByKey()*

*averageScoreStudGrad.foreach(println)*

*println("Average score in Grade 2 over 50")*

***val*** *grade2Greater50 = rdd.filter(x=> x.\_3 == "grade-2").map (x=> (x.\_1,(x.\_4 + x.\_5))).groupByKey.map(x=> (x.\_1,(x.\_2.sum.toDouble/(x.\_2.size \*2)))).filter(x => x.\_2 > 50).collect*

*grade2Greater50.foreach(println)*

*println ("Average score satisfying the below criteria")*

***val*** *avgScoreStudentGrade = rdd.map(x=> (x.\_1,(x.\_4 + x.\_5))).groupByKey.map(x=> (x.\_1,(x.\_2.sum.toDouble/(x.\_2.size \*2))))*

*avgScoreStudentGrade.foreach(println)*

*}*

*}*

# Result

Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties

(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)

Number of lines = 22

history

science

maths

Number of subjects is 3

Number of students called Mathew who scored 55 is 2

No of students per Grade

(grade-3,3)

(grade-1,4)

(grade-2,5)

*A*verage of each Student in each subject all grades

((Lisa,science),18.25)

((Mark,maths),35.25)

((Mark,science),28.25)

((Mathew,history),41.75)

((Mathew,science),31.0)

((Andrew,history),43.5)

((Andrew,maths),21.5)

((Andrew,science),24.5)

((John,history),26.5)

((John,maths),33.25)

((Lisa,history),53.0)

Average score Student each Student Grade

((Lisa,science,grade-1),18.0)

((Lisa,science,grade-2),18.5)

((Mark,maths,grade-1),52.5)

((Mark,maths,grade-2),18.0)

((Mark,science,grade-1),44.5)

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

((Mathew,history,grade-2),41.75)

((Mathew,science,grade-2),33.5)

((Mathew,science,grade-3),28.5)

((Andrew,history,grade-1),43.0)

((Andrew,history,grade-2),44.0)

((Andrew,maths,grade-1),21.5)

((Andrew,science,grade-3),24.5)

((John,history,grade-1),26.5)

((John,maths,grade-1),23.0)

((John,maths,grade-2),43.5)

((Lisa,history,grade-2),56.5)

((Lisa,history,grade-3),49.5)

Average score in Grade 2 over 50

None where achieved

Average score satisfying the below criteria

(Mark,31.75)

(Andrew,29.833333333333332)

(John,29.875)

(Lisa,35.625)

(Mathew,36.375)