

ASSIGNMENT 17.1

Problem Statement:

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.

Sample document:

This-is-my-first-assignment.

It-will-count-the-number-of-lines-in-this-document.

The-total-number-of-lines-is-3

Solution:

Here is a sample text file along with the content for first two problems:

Introduction to Spark.txt

Apache Spark is a cluster computing platform designed to be fast and general purpose. Spark Core contains the basic functionality of Spark, including components for task scheduling, memory management, fault recovery, interacting with storage systems, and more. Spark Core is also home to the API that defines resilient distributed datasets (RDDs), which are Spark's main programming abstraction. RDDs represent a collection of items distributed across many compute nodes that can be manipulated in parallel. Spark Core provides many APIs for building and manipulating these collections.

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

Spark code in Scala:

```
object SparkTextFileOperations {  
  
    def main(args: Array[String]): Unit = {
```

```

val conf = new SparkConf().setAppName("SparkSampleTest").setMaster("local[*]")

val sc = new SparkContext(conf)    // create spark context applying using spark config

val lines = sc.textFile("E:\\Acadgild\\Session 17\\Assignment 17.1\\Introduction to Spark.txt")

val numberOfRows = lines.count()   // count() returns number of elements in the input

                                     // RDD where each element is a line of type String

println("Number of rows in the given text file: " + numberOfRows) // print the count
}
}

```

Output:

Number of rows in the given text file: 8

The screenshot shows the SparkSampleModule project in an IDE. The file SparkTextFileOperations.scala is open, displaying the following code:

```

1 package org.spark_samples
2
3 import org.apache.spark.{SparkConf, SparkContext}
4
5 object SparkTextFileOperations {
6
7   def main(args: Array[String]): Unit = {
8
9     val conf = new SparkConf().setAppName("SparkSampleTest").setMaster("local[*]")
10    val sc = new SparkContext(conf)
11    val lines = sc.textFile("E:\\Acadgild\\Session 17\\Assignment 17.1\\Introduction to Spark.txt")
12    val numberOfRows = lines.count()
13    println("Number of rows in the given text file: " + numberOfRows)
14  }
15
16 }

```

The Run console shows the following output:

```

18/02/18 23:25:32 INFO Executor: finished task 1.0 in stage 0.0 (TID 1). 1041 bytes result sent to driver
18/02/18 23:25:32 INFO Executor: Finished task 0.0 in stage 0.0 (TID 0). 1041 bytes result sent to driver
18/02/18 23:25:32 INFO TaskSetManager: Finished task 1.0 in stage 0.0 (TID 1) in 370 ms on localhost (1/2)
18/02/18 23:25:32 INFO TaskSetManager: Finished task 0.0 in stage 0.0 (TID 0) in 485 ms on localhost (2/2)
18/02/18 23:25:32 INFO TaskSchedulerImpl: Removed TaskSet 0.0, whose tasks have all completed, from pool
18/02/18 23:25:32 INFO DAGScheduler: ResultStage 0 (count at SparkTextFileOperations.scala:12) finished in 0.544 s
18/02/18 23:25:32 INFO DAGScheduler: Job 0 finished: count at SparkTextFileOperations.scala:12, took 1.061275 s
Number of rows in the given text file: 8
18/02/18 23:25:32 INFO SparkContext: Invoking stop() from shutdown hook
18/02/18 23:25:32 INFO SparkUI: Stopped Spark web UI at http://192.168.43.50:4040
18/02/18 23:25:32 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
18/02/18 23:25:32 INFO MemoryStore: MemoryStore cleared
18/02/18 23:25:32 INFO BlockManager: BlockManager stopped
18/02/18 23:25:32 INFO BlockManagerMaster: BlockManagerMaster stopped
18/02/18 23:25:32 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
18/02/18 23:25:32 INFO SparkContext: Successfully stopped SparkContext
18/02/18 23:25:32 INFO ShutdownHookManager: Shutdown hook called

```

The output "Number of rows in the given text file: 8" is highlighted in yellow in the original image.

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

Spark code in Scala:

```
val lines = sc.textFile("E:\\Acadgild\\Session 17\\Assignment 17.1\\Introduction to Spark.txt")

val words = lines.flatMap(x => x.split(" ")) //split each line by space between words

val initialWordCountRDD = words.map(x => (x, 1)) //assign initial count as 1 to each word

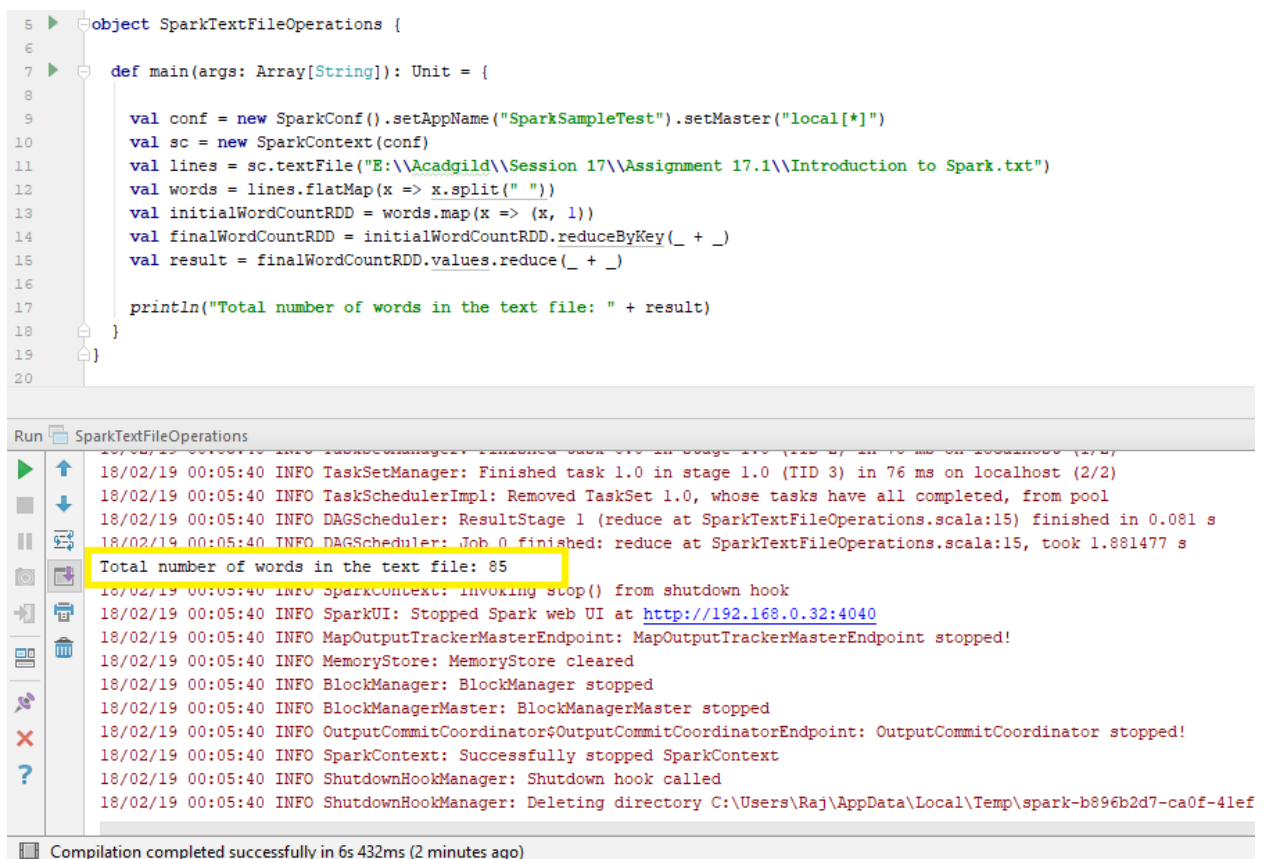
val finalWordCountRDD = initialWordCountRDD.reduceByKey(_ + _) //add count for same word

val result = finalWordCountRDD.values.reduce(_ + _) // take total count of each word and
// sum up all counts

println("Total number of words in the text file: " + result) // print result
```

Output:

Total number of words in the text file: 85



The screenshot displays an IDE with a Scala program and its execution output. The code defines an object `SparkTextFileOperations` with a `main` method that reads a text file, splits it into words, and calculates the total word count using RDD operations. The output console shows the execution of this program, with the final result `Total number of words in the text file: 85` highlighted in a yellow box. The output also includes various Spark logs such as task completion, DAG scheduling, and context shutdown.

```
object SparkTextFileOperations {
  def main(args: Array[String]): Unit = {
    val conf = new SparkConf().setAppName("SparkSampleTest").setMaster("local[*]")
    val sc = new SparkContext(conf)
    val lines = sc.textFile("E:\\Acadgild\\Session 17\\Assignment 17.1\\Introduction to Spark.txt")
    val words = lines.flatMap(x => x.split(" "))
    val initialWordCountRDD = words.map(x => (x, 1))
    val finalWordCountRDD = initialWordCountRDD.reduceByKey(_ + _)
    val result = finalWordCountRDD.values.reduce(_ + _)

    println("Total number of words in the text file: " + result)
  }
}
```

Run SparkTextFileOperations

```
18/02/19 00:05:40 INFO TaskSetManager: Finished task 1.0 in stage 1.0 (TID 3) in 76 ms on localhost (2/2)
18/02/19 00:05:40 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
18/02/19 00:05:40 INFO DAGScheduler: ResultStage 1 (reduce at SparkTextFileOperations.scala:15) finished in 0.081 s
18/02/19 00:05:40 INFO DAGScheduler: Job 0 finished: reduce at SparkTextFileOperations.scala:15, took 1.881477 s
Total number of words in the text file: 85
18/02/19 00:05:40 INFO SparkContext: Invoking stop() from shutdown hook
18/02/19 00:05:40 INFO SparkUI: Stopped Spark web UI at http://192.168.0.32:4040
18/02/19 00:05:40 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
18/02/19 00:05:40 INFO MemoryStore: MemoryStore cleared
18/02/19 00:05:40 INFO BlockManager: BlockManager stopped
18/02/19 00:05:40 INFO BlockManagerMaster: BlockManagerMaster stopped
18/02/19 00:05:40 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
18/02/19 00:05:40 INFO SparkContext: Successfully stopped SparkContext
18/02/19 00:05:40 INFO ShutdownHookManager: Shutdown hook called
18/02/19 00:05:40 INFO ShutdownHookManager: Deleting directory C:\Users\Raj\AppData\Local\Temp\spark-b896b2d7-ca0f-41ef
```

Compilation completed successfully in 6s 432ms (2 minutes ago)

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.

Spark code in Scala:

```
val lines = sc.textFile("E:\\Acadgild\\Session 17\\Assignment 17.1\\sample text file.txt")

val words = lines.flatMap(x => x.split("-"))           //split each line by hyphen between words

val initialWordCountRDD = words.map(x => (x, 1))        //assign initial count as 1 to each word

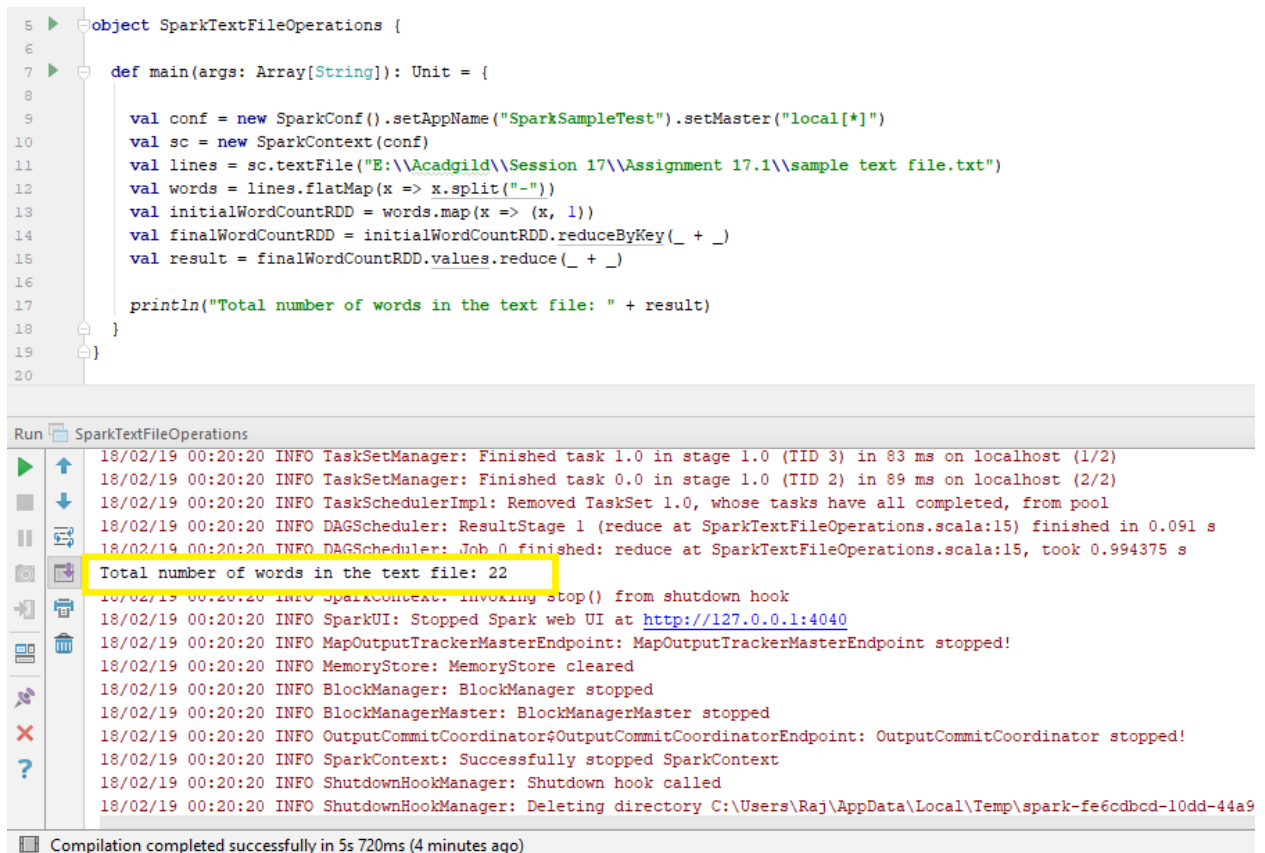
val finalWordCountRDD = initialWordCountRDD.reduceByKey(_ + _) //add count for same word

val result = finalWordCountRDD.values.reduce(_ + _)     // take total count of each word and
                                                         // sum up all counts

println("Total number of words in the text file: " + result) // print result
```

Output:

Total number of words in the text file: 22



```
5 ▶ object SparkTextFileOperations {
6
7 ▶ def main(args: Array[String]): Unit = {
8
9     val conf = new SparkConf().setAppName("SparkSampleTest").setMaster("local[*]")
10    val sc = new SparkContext(conf)
11    val lines = sc.textFile("E:\\Acadgild\\Session 17\\Assignment 17.1\\sample text file.txt")
12    val words = lines.flatMap(x => x.split("-"))
13    val initialWordCountRDD = words.map(x => (x, 1))
14    val finalWordCountRDD = initialWordCountRDD.reduceByKey(_ + _)
15    val result = finalWordCountRDD.values.reduce(_ + _)
16
17    println("Total number of words in the text file: " + result)
18  }
19 }
20
```

Run SparkTextFileOperations

```
18/02/19 00:20:20 INFO TaskSetManager: Finished task 1.0 in stage 1.0 (TID 3) in 83 ms on localhost (1/2)
18/02/19 00:20:20 INFO TaskSetManager: Finished task 0.0 in stage 1.0 (TID 2) in 89 ms on localhost (2/2)
18/02/19 00:20:20 INFO TaskSchedulerImpl: Removed TaskSet 1.0, whose tasks have all completed, from pool
18/02/19 00:20:20 INFO DAGScheduler: ResultStage 1 (reduce at SparkTextFileOperations.scala:15) finished in 0.091 s
18/02/19 00:20:20 INFO DAGScheduler: Job 0 finished: reduce at SparkTextFileOperations.scala:15, took 0.994375 s
Total number of words in the text file: 22
18/02/19 00:20:20 INFO SparkContext: Invoking stop() from shutdown hook
18/02/19 00:20:20 INFO SparkUI: Stopped Spark web UI at http://127.0.0.1:4040
18/02/19 00:20:20 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
18/02/19 00:20:20 INFO MemoryStore: MemoryStore cleared
18/02/19 00:20:20 INFO BlockManager: BlockManager stopped
18/02/19 00:20:20 INFO BlockManagerMaster: BlockManagerMaster stopped
18/02/19 00:20:20 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stopped!
18/02/19 00:20:20 INFO SparkContext: Successfully stopped SparkContext
18/02/19 00:20:20 INFO ShutdownHookManager: Shutdown hook called
18/02/19 00:20:20 INFO ShutdownHookManager: Deleting directory C:\Users\Raj\AppData\Local\Temp\spark-fe6cdbc-d10dd-44a9
```

Compilation completed successfully in 5s 720ms (4 minutes ago)