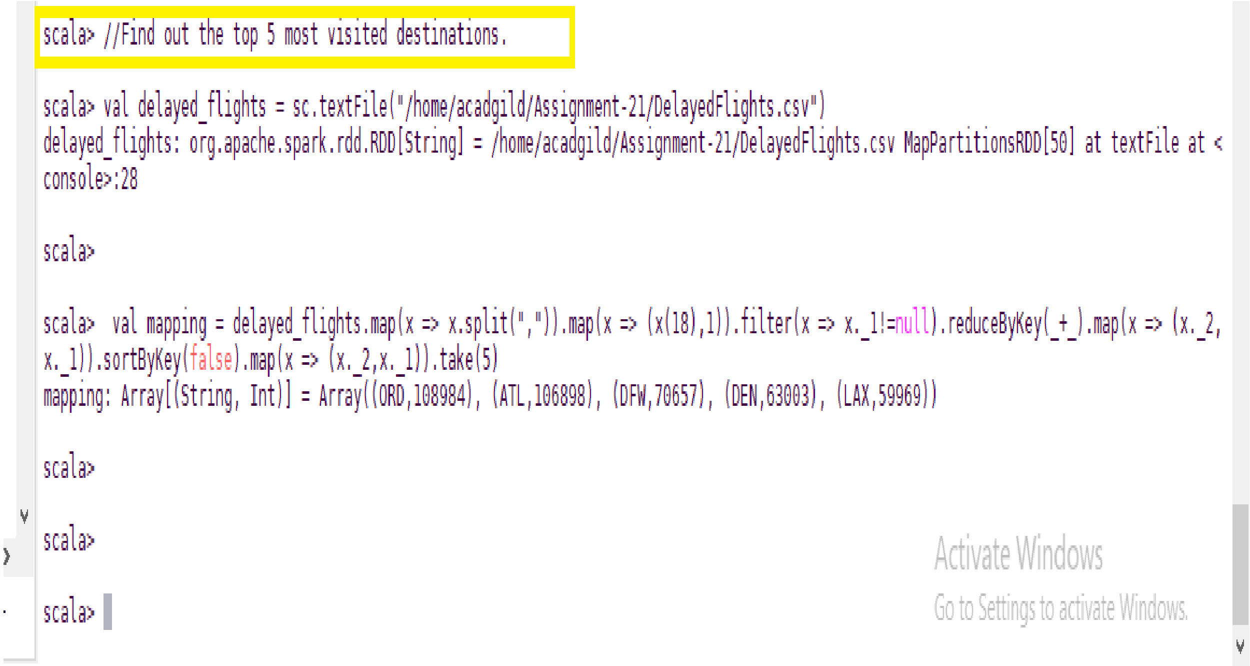
Problem Statement

Implement the below blog at your end and send the complete documentation

.

https://drive.google.com/file/d/0B_Qjau8wv1KoUThzZ24tT1NsZGs/view?usp=sharing

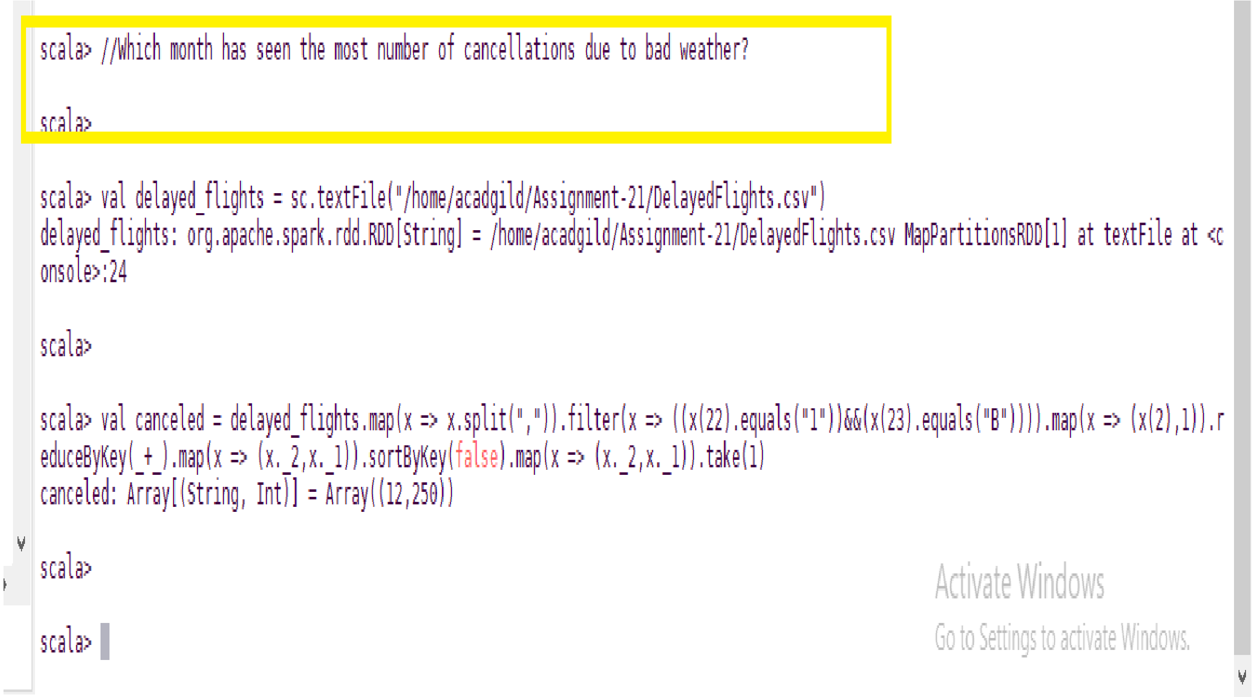
Question 1 Solution Screen-shot



```
scala> //Find out the top 5 most visited destinations.

scala> val delayed_flights = sc.textFile("/home/acadgild/Assignment-21/DelayedFlights.csv")
delayed_flights: org.apache.spark.rdd.RDD[String] = /home/acadgild/Assignment-21/DelayedFlights.csv MapPartitionsRDD[50] at textFile at <
console>:28

scala>

scala>  val mapping = delayed_flights.map(x => x.split(",")).map(x => (x(18),1)).filter(x => x._1!=null).reduceByKey(_+_).map(x => (x._2,
x._1)).sortByKey(false).map(x => (x._2,x._1)).take(5)
mapping: Array[(String, Int)] = Array((ORD,108984), (ATL,106898), (DFW,70657), (DEN,63003), (LAX,59969))

scala>

scala>

scala>
```

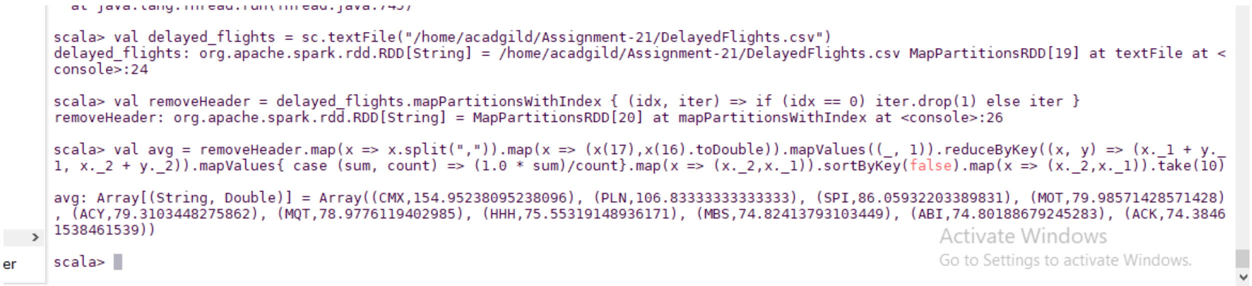
Accadgild_Session_21_Assignment_21.2_Solutions

Question 2 Solution Screen-Shot



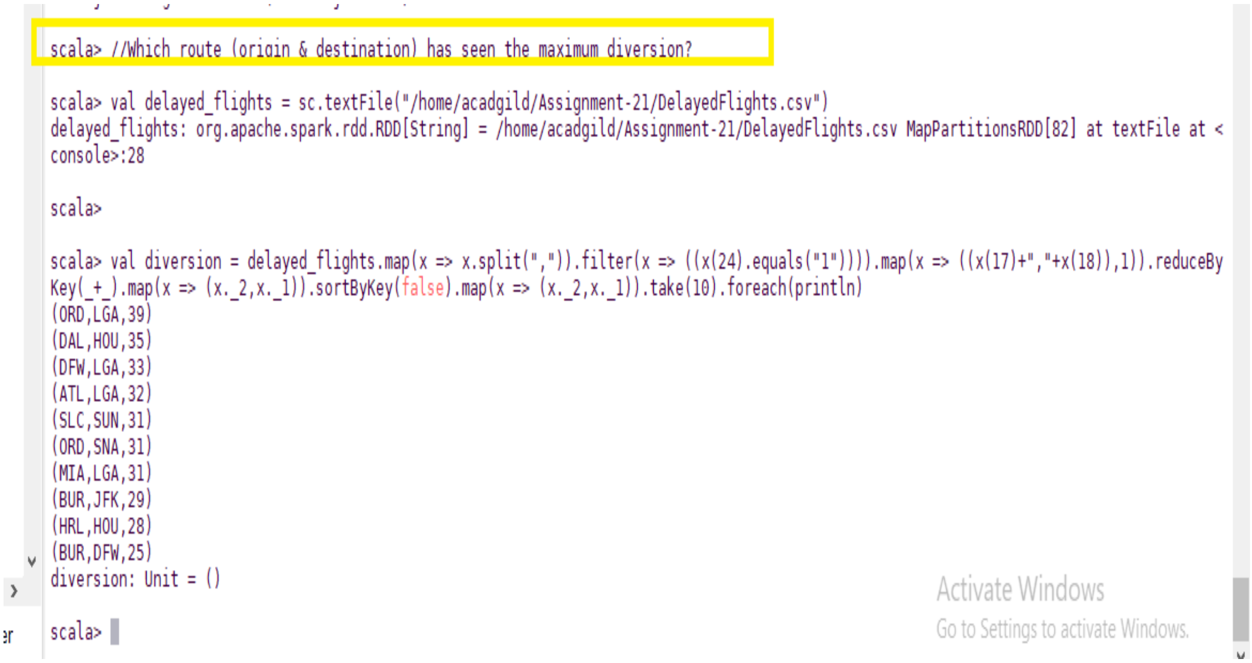
```
scala> //Which month has seen the most number of cancellations due to bad weather?

scala>

scala> val delayed_flights = sc.textFile("/home/acadgild/Assignment-21/DelayedFlights.csv")
delayed_flights: org.apache.spark.rdd.RDD[String] = /home/acadgild/Assignment-21/DelayedFlights.csv MapPartitionsRDD[1] at textFile at <c
onsole>:24

scala>

scala> val canceled = delayed_flights.map(x => x.split(",")).filter(x => ((x(22).equals("1"))&&(x(23).equals("B")))).map(x => (x(2),1)).r
educeByKey(_+_).map(x => (x._2,x._1)).sortByKey(false).map(x => (x._2,x._1)).take(1)
canceled: Array[(String, Int)] = Array((12,250))

scala>

scala>
```

Question 3 Solution Screen-Shot



```
  at java.lang.Thread.run(Thread.java:745)

scala> val delayed_flights = sc.textFile("/home/acadgild/Assignment-21/DelayedFlights.csv")
delayed_flights: org.apache.spark.rdd.RDD[String] = /home/acadgild/Assignment-21/DelayedFlights.csv MapPartitionsRDD[19] at textFile at <
console>:24

scala> val removeHeader = delayed_flights.mapPartitionsWithIndex { (idx, iter) => if (idx == 0) iter.drop(1) else iter }
removeHeader: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[20] at mapPartitionsWithIndex at <console>:26

scala> val avg = removeHeader.map(x => x.split(",")).map(x => (x(17),x(16).toDouble)).mapValues((_, 1)).reduceByKey((x, y) => (x._1 + y._
1, x._2 + y._2)).mapValues{ case (sum, count) => (1.0 * sum)/count}.map(x => (x._2,x._1)).sortByKey(false).map(x => (x._2,x._1)).take(10)

avg: Array[(String, Double)] = Array((CMX,154.95238095238096), (PLN,106.83333333333333), (SPI,86.05932203389831), (MOT,79.98571428571428)
, (ACY,79.3103448275862), (MQT,78.9776119402985), (HHH,75.55319148936171), (MBS,74.82413793103449), (ABI,74.80188679245283), (ACK,74.3846
1538461539))

scala>
```

Question 4 Solution Screen-Shot



```
scala> //Which route (origin & destination) has seen the maximum diversion?

scala> val delayed_flights = sc.textFile("/home/acadgild/Assignment-21/DelayedFlights.csv")
delayed_flights: org.apache.spark.rdd.RDD[String] = /home/acadgild/Assignment-21/DelayedFlights.csv MapPartitionsRDD[82] at textFile at <
console>:28

scala>

scala> val diversion = delayed_flights.map(x => x.split(",")).filter(x => ((x(24).equals("1")))).map(x => ((x(17)+","+x(18)),1)).reduceBy
Key(_+_).map(x => (x._2,x._1)).sortByKey(false).map(x => (x._2,x._1)).take(10).foreach(println)
(ORD,LGA,39)
(DAL,HOU,35)
(DFW,LGA,33)
(ATL,LGA,32)
(SLC,SUN,31)
(ORD,SNA,31)
(MIA,LGA,31)
(BUR,JFK,29)
(HRL,HOU,28)
(BUR,DFW,25)
diversion: Unit = ()

scala>
```

Submitted By

Shishir Jha