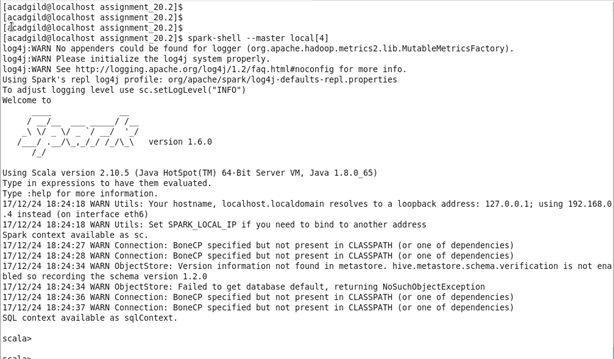
Assignment 20.1

spark-shell --master local[4]



Step2:

import org.apache.spark.\_

import org.apache.spark.streaming.\_

import org.apache.spark.streaming.StreamingContext.\_

Step3:

Declare accumulator totalEvenLinesWordNumber which will keep track of sum of number of word numbers in lines so far

val totalEvenLinesWordNumber = sc.accumulator(0)

Step4:

val wordNumberMap = Map("Hi" -> 1, "my" -> 2, "name" -> 3, "is" -> 4, "Hello" ->5, "Monimoy" -> 6, "John" -> 7, "Bob"->8, "Vibhu" ->9)

val wordNumberMapBroadcast = sc.broadcast(wordNumberMap)

Step5:

def lineWordNumberTotal(line:String):Int = {

var sum:Int = 0

var words = line.split(" ")

for (word <- words) sum += wordNumberMapBroadcast.value.get(word).getOrElse(0)

sum

}

Screenshot of steps 2 to 4 is as below:



Step6:

val ssc = new StreamingContext(sc, Seconds(15))

val stream = ssc.socketTextStream("localhost", 9999)

Step7:

stream.foreachRDD(line => {

val lineStr = line.collect().toList.mkString("")

if (lineStr != "") {

var numTotal = lineWordNumberTotal(lineStr)

if (numTotal % 2 == 1) println(lineStr)

else {

totalEvenLinesWordNumber += numTotal

println("Sum of lines with even word number so far =" + totalEvenLinesWordNumber.value.toInt)

}

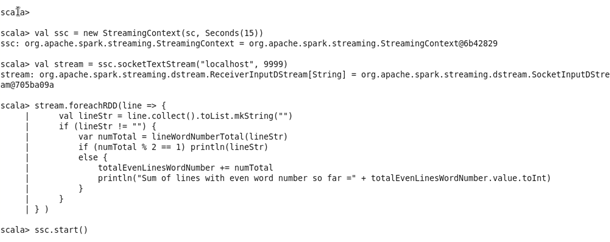
}

} )

Step8:

ssc.start()

ssc.awaitTermination()



Step9:

nc -lk 9999

The screenshot is as below:



Step10:

