Task 1:

Created a new empty folder and run the scala code

Create a new file inside casestudy5 while the code is running.

Code:

```
import org.apache.spark.{SparkConf, SparkContext}
import org.apache.spark.streaming.{Seconds, StreamingContext}
import org.apache.log4j.{Level,Logger}

object CaseStudy5 {
  def main(args: Array[String]): Unit = {
    println("Spark Streaming")
    val directory = "/home/acadgild/casestudy5"
    println(s" $directory")
    val conf = new SparkConf().setMaster("local[2]").setAppName("CaseStudy5")
    val sc = new SparkContext(conf)
    val ssc = new StreamingContext(sc, Seconds(15))
    val lines = ssc.textFileStream(args(0))
    //lines.print()
    val words = lines.flatMap(_.split(" "))
    val wordCounts = words.map(x => (x, 1)).reduceByKey(_ + _)
    wordCounts.print()
    ssc.start()
    ssc.awaitTermination()
  }
}
```

Output

File contents:

```
[acadgild@localhost ~]$ cat test2.txt
I am a
Big Data
Developer
Spark is fun
So is scala
Lets do it
[acadgild@localhost ~]$
```

Created new file in the casestudy5 folder while the code is running:

```
[acadgild@localhost casestudy5]$ ls -lrt
total 8
-rw-rw-r--. 1 acadgild acadgild 88 Jun  7 12:04 test2.txt
-rw-rw-r--. 1 acadgild acadgild 68 Jun  7 12:07 test3.txt
[acadgild@localhost casestudy5]$ nano test5.txt
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost casestudy5]$ ls -lrt
total 12
-rw-rw-r--. 1 acadgild acadgild 88 Jun  7 12:04 test2.txt
-rw-rw-r--. 1 acadgild acadgild 68 Jun  7 12:07 test3.txt
-rw-rw-r--. 1 acadgild acadgild 89 Jun  7 12:33 test5.txt
[acadgild@localhost casestudy5]$ cat test5.txt
I am a
Big Data
Developer
Spark is fun
So is scala
Lets do it
```

```
CaseStudy5$ [Scala Application] /usr/java/jdk1.8.0_151/bin/java (Jun 7, 2018, 12:39:38 PM)
18/06/07 12:40:15 INFO Executor: Finished task 0.0 in stage 11.0 (TID 6). 1279 bytes result sent t
18/06/07 12:40:15 INFO TaskSetManager: Finished task 0.0 in stage 11.0 (TID 6) in 11 ms on localho
18/06/07 12:40:15 INFO TaskSchedulerImpl: Removed TaskSet 11.0, whose tasks have all completed, fr
-------------------------------------------
Time: 1528355415000 ms
-------------------------------------------
(Data,1)
(Developer,1)
(I,1)
(Lets,1)
(Big,1)
(It,1)
(is,2)
(Do,1)
(Spark,1)
(fun,1)
(Scala,1)
(a,1)
(So,1)
(am,1)
```

Task 2:

Code:

Created a new folder as casestudy5 in hadoop filesystem.

Had a file in local folder so that it can be moved to hdfs on the fly and used for wordcount.

```
import java.io.File

import org.apache.spark.{SparkConf, SparkContext}

import scala.io.Source._
import org.apache.log4j.{Level, Logger}

object CaseStudy5part2 {
  private var localFilePath: File = new
File("/home/acadgild/casestudy5/test6.txt")
  private var dfsDirPath: String = "hdfs://localhost:8020/casestudy5"
  //private val NPARAMS = 2

  def main(args: Array[String]): Unit = {
    println("HDFSWordCountComparison : Main Called Successfully")

    println("Performing local word count")
    val fileContents = readFile(localFilePath.toString())

    println("Performing local word count - File Content ->>" + fileContents)
    val localWordCount = runLocalWordCount(fileContents)

    println("SparkHDFSWordCountComparison : Main Called Successfully -> Local
Word Count is ->>" + localWordCount)
    println("Performing local word count Completed !!")

    println("Creating Spark Context")
    val conf = new
SparkConf().setMaster("local[2]").setAppName("SparkHDFSWordCountComparisonApp"
)
    val sc = new SparkContext(conf)
    val rootLogger = Logger.getRootLogger()
    rootLogger.setLevel(Level.ERROR)

    println("Spark Context Created")

    println("Writing local file to DFS")
    val dfsFilename = dfsDirPath + "/local_to_hdfs"
    val fileRDD = sc.parallelize(fileContents)
    fileRDD.saveAsTextFile(dfsFilename)
    println("Writing local file to DFS Completed")

    println("Reading file from DFS and running Word Count")
    val readFileRDD = sc.textFile(dfsFilename)
```

```
    val dfsWordCount = readFileRDD
      .flatMap(_.split(" "))
      .flatMap(_.split("\t"))
      .filter(_.nonEmpty)
      .map(w => (w, 1))
      .countByKey()
      .values
      .sum

    sc.stop()

    if (localWordCount == dfsWordCount) {
      println(s"Success! Local Word Count ($localWordCount) " +
        s"and DFS Word Count ($dfsWordCount) are same.")
    } else {
      println(s"Failure! Local Word Count ($localWordCount) " +
        s"and DFS Word Count ($dfsWordCount) are not same.")
    }
  }
  private def printUsage(): Unit = {
    val usage: String = "DFS Read-Write Test\n" +
      "\n" +
      "Usage: localFile dfsDir\n" +
      "\n" +
      "localFile - (string) local file to use in test\n" +
      "dfsDir - (string) DFS directory for read/write tests\n"

    println(usage)
  }

  private def readFile(filename: String): List[String] = {
    val lineIter: Iterator[String] = fromFile(filename).getLines()
    val lineList: List[String] = lineIter.toList
    lineList
  }

  def runLocalWordCount(fileContents: List[String]): Int = {
    fileContents.flatMap(_.split(" "))
      .flatMap(_.split("\t"))
      .filter(_.nonEmpty)
      .groupBy(w => w)
      .mapValues(_.size)
      .values
      .sum
  }
}
```

Output from hdfs:

```
[acadgild@localhost ~]$ hadoop fs -ls /casestudy5/local_to_hdfs
18/06/07 13:30:59 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classe
s where applicable
Found 3 items
-rw-r--r--   3 acadgild supergroup          0 2018-06-07 13:26 /casestudy5/local_to_hdfs/_SUCCESS
-rw-r--r--   3 acadgild supergroup         47 2018-06-07 13:26 /casestudy5/local_to_hdfs/part-00000
-rw-r--r--   3 acadgild supergroup         41 2018-06-07 13:26 /casestudy5/local_to_hdfs/part-00001
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ hadoop fs -cat /casestudy5/local_to_hdfs/part-00000
18/06/07 13:31:24 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classe
s where applicable
I am a
Big Data
Developer
[acadgild@localhost ~]$ hadoop fs -cat /casestudy5/local_to_hdfs/part-00001
18/06/07 13:31:36 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classe
s where applicable
Spark is fun
So is scala
Lets do it
[acadgild@localhost ~]$
```

Console output:

```
<terminated> CaseStudy5part2$ [Scala Application] /usr/java/jdk1.8.0_151/bin/java (Jun 7, 2018, 1:26:23 PM)
HDFSWordCountComparison : Main Called Successfully
Performing local word count
Performing local word count - File Content ->>List(I am a, Big Data, Developer, Spark is fun, So is scala,
SparkHDFSWordCountComparison : Main Called Successfully -> Local Word Count is ->>15
Performing local word count Completed !!
Creating Spark Context
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
```

File comparison:

```
Spark Context Created
Writing local file to DFS
Writing local file to DFS Completed
Reading file from DFS and running Word Count
Success! Local Word Count (15) and DFS Word Count (15) are same.
```