Assignemt 17.1

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

Create a sample file with two lines

```
$ hdfs dfs -cat sample.txt
this is a sample file for testing
number of lines in this file is 2
```

val word = sc.textFile("sample.txt")
word.count

```
scala> val word = sc.textFile("sample.txt")
word: org.apache.spark.rdd.RDD[String] = sample.txt MapPartitionsRDD[1] at textFile at <console>:24

scala> word.count
res0: Long = 2
```

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

val word=sc.textFile("sample.txt")
var flat_map= word.flatMap(row=>row.split(" "))
var map=flat_map.map(word=>(word,1))
var count=map.reduceByKey(_+_)
count.collect().foreach(println)

```
scala> var flat_map= word.flatMap(row=>row.split(" "))
flat_map: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[2] at flatMap at <console>:26

scala> var map=flat_map.map(word=>(word,1))
map: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[3] at map at <console>:28

scala> var count=map.reduceByKey(_+_)
count: org.apache.spark.rdd.RDD[(String, Int)] = ShuffledRDD[4] at reduceByKey at <console>:30
```

```
scala> count.collect().foreach(println)
(this,2)
(is,2)
(2,1)
(testing,1)
(file,2)
(sample,1)
(lines,1)
(a,1)
(number,1)
(in,1)
(of,1)
(for,1)
```

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.

```
val word= sc.textFile("sample.txt")
val words= word.flatMap(line=>line.split("-"))
val wordCounts= words.map(word=>(word,1)).reduceByKey{case(x,y)=>x+y}
 wordCounts.saveAsTextFile("Output")
```

```
scala> val word = sc.textFile("sample.txt")
word: org.apache.spark.rdd.RDD[String] = sample.txt MapPartitionsRDD[7] at textFile at <console>:24

scala> val words= word.flatMap(line=>line.split("-"))
words: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[8] at flatMap at <console>:26

scala> val wordCounts= words.map(word=>(word,1)).reduceByKey{case(x,y)=>x+y}
wordCounts: org.apache.spark.rdd.RDD[(String, Int)] = ShuffledRDD[10] at reduceByKey at <console>:28

scala> wordCounts.saveAsTextFile("Output")
```

OUTPUT

```
$ hdfs  dfs -ls Output
Found 3 items
-rw-r--r--                hadoop          0 2018-01-17 04:21 Output/_SUCCESS
-rw-r--r--                hadoop         61 2018-01-17 04:21 Output/part-00000
-rw-r--r--                hadoop        106 2018-01-17 04:21 Output/part-00001

$ hdfs dfs -cat Output/part-00000
(this,1)
(is,2)
(will,1)
(This,1)
(first,1)
(total,1)
(my,1)
$ hdfs dfs -cat Output/part-00001
(lines,2)
(The,1)
(document. ,1)
(number,2)
(assignment. ,1)
(in,1)
(3,1)
(of,2)
(It,1)
(count,1)
(the,1)
$
```