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

Solution-

We have taken a dataset and kept in local file in below location-

```
[acadgild@localhost assignment-17.1]$ pwd
/home/acadgild/assignment-17.1
[acadgild@localhost assignment-17.1]$ ls -l
total 4
-rw-rw-r--. 1 acadgild acadgild 113 Dec 25 00:30 Sample-Data.txt
```

In below screenshot we are creating a RDD using spark context to read the file-

```
scala> val input = sc.textFile("/home/acadgild/assignment-17.1/Sample-Data.txt")
input: org.apache.spark.rdd.RDD[String] = /home/acadgild/assignment-17.1/Sample-Data.txt MapPartitionsRDD[1] at textFile at <console>:24

scala>
```

In below screenshot we are using function count() to count the number of rows in the file.

The result is 3-

```
scala> val rowCount = input.count()
rowCount: Long = 3

scala>
```

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

Solution-

```
scala> val input = sc.textFile("/home/acadgild/assignment-17.1/Sample-Data.txt")
input: org.apache.spark.rdd.RDD[String] = /home/acadgild/assignment-17.1/Sample-Data.txt MapPartitionsRDD[1] at textFile at <console>:24

scala>
```

We are using above file only for this problem also. In order to count the words we are first splitting it using the delimiter "-" and doing a flatmap-

```
scala> val words = input.flatMap(x => x.split("-"))
words: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[2] at flatMap at <console>:26
```

Now below screenshot shows the result of above RDD i.e. the words after splitting-

```
scala> words.collect()
res1: Array[String] = Array(This, is, my, first, assignment., It, will, count, the, number, of, lines, in, this, document., The, total, number, of, lines, is, 3)

scala>
```

Now we are using count() function to count the number of words present in the file-

```
scala> val count2 = words.count()
count2: Long = 22

scala>
```

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.

Solution-

```
scala> val input = sc.textFile("/home/acadgild/assignment-17.1/Sample-Data.txt")
input: org.apache.spark.rdd.RDD[String] = /home/acadgild/assignment-17.1/Sample-Data.txt MapPartitionsRDD[1] at textFile at <console>:24

scala>
```

Here also we are again using above file only to do further operation and using split function to separate the words based on delimiter “-“.

```
scala> val words = input.flatMap(x => x.split("-"))
words: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[2] at flatMap at <console>:26
```

Now we are mapping each word with the value 1 which will create a tuple. After that we are using reduceByKey to add those numbers of occurrences as shown below-

```
scala> val result = words.map(x => (x,1)).reduceByKey((x,y) => x+y)
result: org.apache.spark.rdd.RDD[(String, Int)] = ShuffledRDD[4] at reduceByKey at <console>:28
```

Below screenshot shows the final result-

```
scala> result.foreach(println)
(this,1)
(is,2)
(will,1)
(This,1)
(first,1)
(total,1)
(my,1)
(lines,2)
(The,1)
(document.,1)
(assignment.,1)
(number,2)
(in,1)
(3,1)
(of,2)
(It,1)
(count,1)
(the,1)
```