ASSIGNMENT 26.1

To solve the both the tasks, we create the spark application, which has two Scala files with object **Even_Number_Line.scala** and **Offensive_Words_Count.scala.**

Description of the codes explained in the code file.

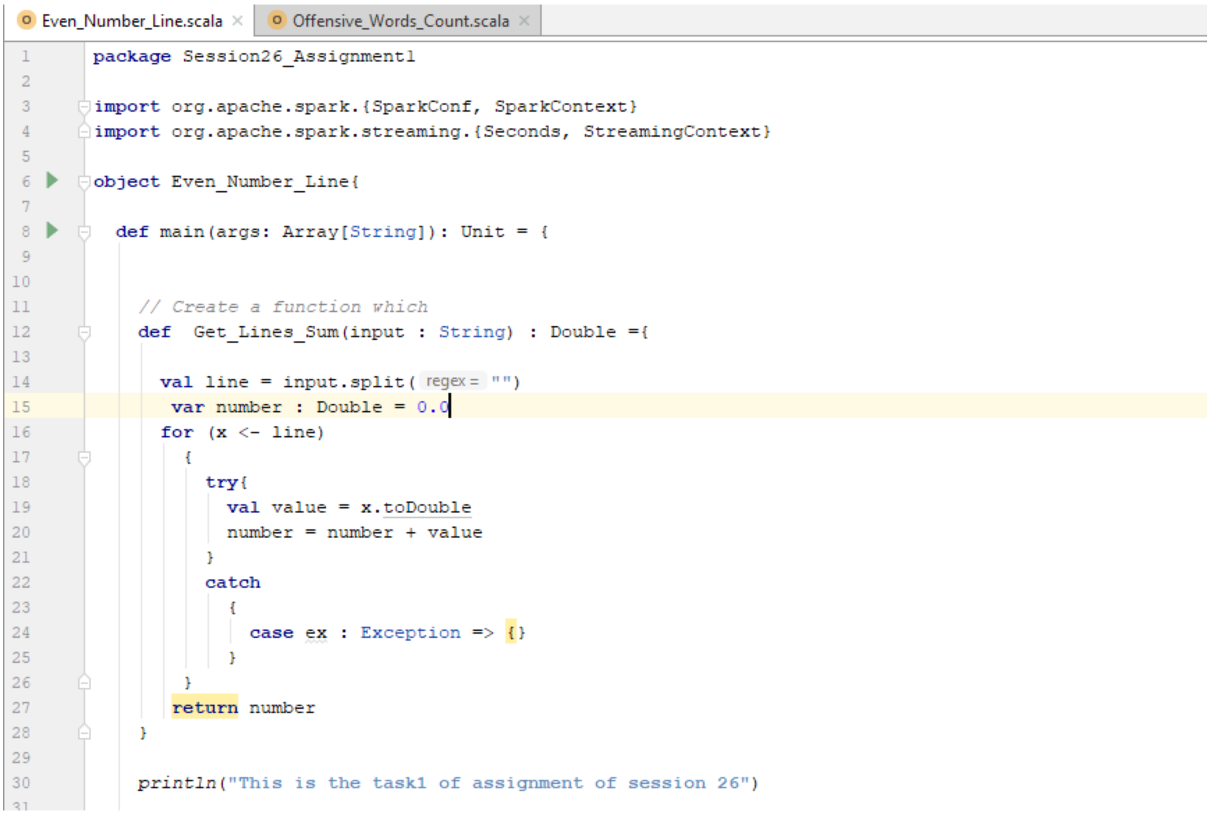
Both task solved in IDEA Intellij and we used the netcat on windows for real time input string streaming.

Task 1

Read a stream of Strings, fetch the words which can be converted to numbers. Filter out the rows, where the sum of numbers in that line is odd.

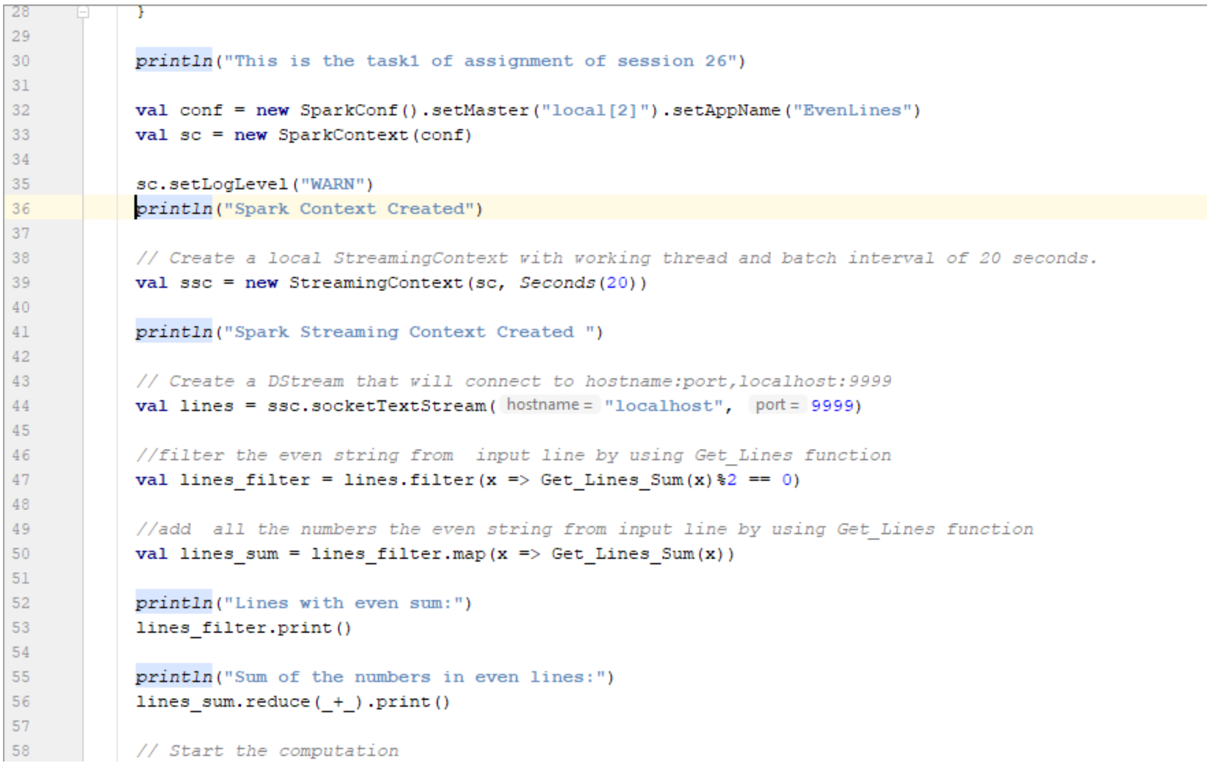
Provide the sum of all the remaining numbers in that batch.

Below screen shot shows the spark application to filter the lines containing even numbers

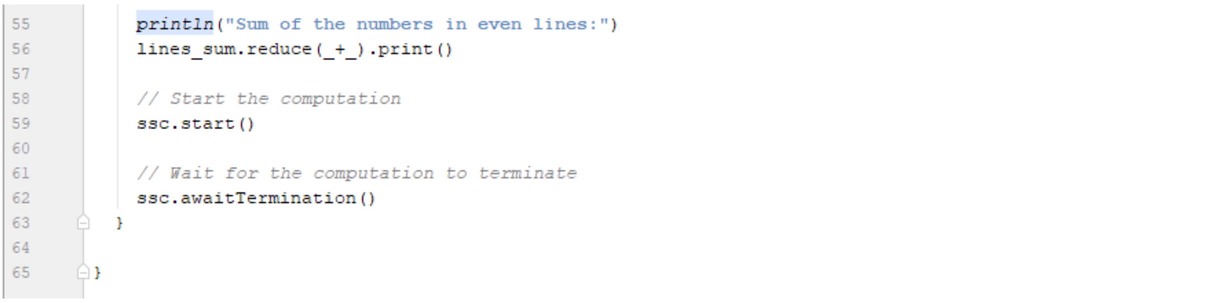


```
package Session26_Assignment1

import org.apache.spark.{SparkConf, SparkContext}
import org.apache.spark.streaming.{Seconds, StreamingContext}

object Even_Number_Line{

  def main(args: Array[String]): Unit = {


    // Create a function which
    def  Get_Lines_Sum(input : String) : Double ={

      val line = input.split(" ")
       var number : Double = 0.0
      for (x <- line)
        {
          try{
            val value = x.toDouble
            number = number + value
          }
          catch
            {
              case ex : Exception => {}
            }
        }
       return number
    }

    println("This is the task1 of assignment of session 26")
```

.



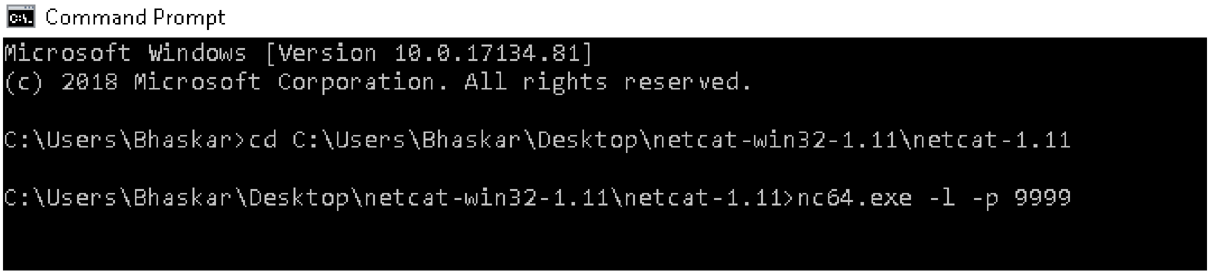
```
    }

    println("This is the task1 of assignment of session 26")

    val conf = new SparkConf().setMaster("local[2]").setAppName("EvenLines")
    val sc = new SparkContext(conf)

    sc.setLogLevel("WARN")
    println("Spark Context Created")

    // Create a local StreamingContext with working thread and batch interval of 20 seconds.
    val ssc = new StreamingContext(sc, Seconds(20))

    println("Spark Streaming Context Created ")

    // Create a DStream that will connect to hostname:port,localhost:9999
    val lines = ssc.socketTextStream( hostname = "localhost",  port = 9999)

    //filter the even string from  input line by using Get_Lines function
    val lines_filter = lines.filter(x => Get_Lines_Sum(x)%2 == 0)

    //add  all the numbers the even string from input line by using Get_Lines function
    val lines_sum = lines_filter.map(x => Get_Lines_Sum(x))

    println("Lines with even sum:")
    lines_filter.print()

    println("Sum of the numbers in even lines:")
    lines_sum.reduce(_+_).print()

    // Start the computation
```



```
    println("Sum of the numbers in even lines:")
    lines_sum.reduce(_+_).print()

    // Start the computation
    ssc.start()

    // Wait for the computation to terminate
    ssc.awaitTermination()
  }

}
```

Before running the above application lets us start **netcat** as show below in screen shot.

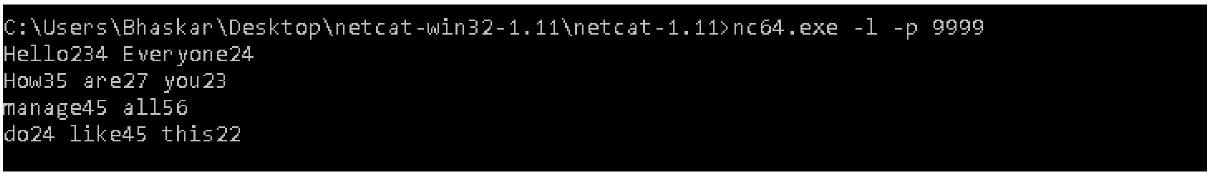
We will use **nc64.exe -l -p 9999** command to run netcat in window.



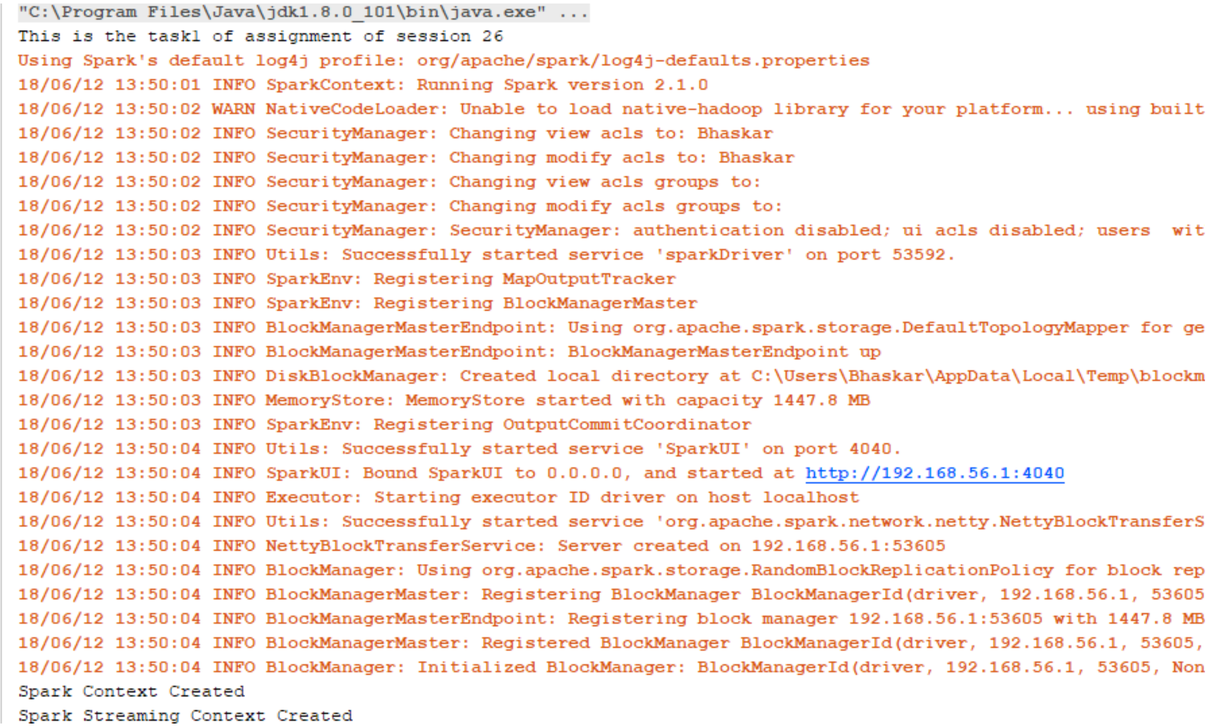
```
Microsoft Windows [Version 10.0.17134.81]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Bhaskar>cd C:\Users\Bhaskar\Desktop\netcat-win32-1.11\netcat-1.11

C:\Users\Bhaskar\Desktop\netcat-win32-1.11\netcat-1.11>nc64.exe -l -p 9999
```

Put some string with numbers and start the above spark applications as shown below

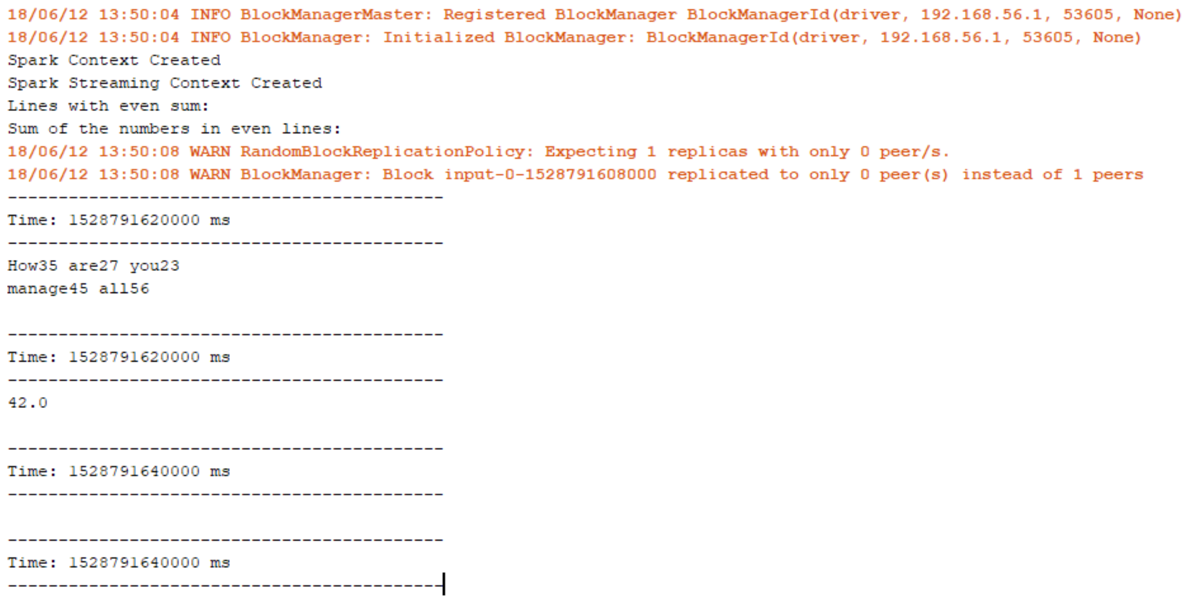
Hello234 Everyone24
How35 are27 you23
manage45 all56
do24 like45 this22



```
C:\Users\Bhaskar\Desktop\netcat-win32-1.11\netcat-1.11>nc64.exe -l -p 9999
Hello234 Everyone24
How35 are27 you23
manage45 all56
do24 like45 this22
```



```
"C:\Program Files\Java\jdk1.8.0_101\bin\java.exe" ...
This is the task1 of assignment of session 26
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
18/06/12 13:50:01 INFO SparkContext: Running Spark version 2.1.0
18/06/12 13:50:02 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using built
18/06/12 13:50:02 INFO SecurityManager: Changing view acls to: Bhaskar
18/06/12 13:50:02 INFO SecurityManager: Changing modify acls to: Bhaskar
18/06/12 13:50:02 INFO SecurityManager: Changing view acls groups to:
18/06/12 13:50:02 INFO SecurityManager: Changing modify acls groups to:
18/06/12 13:50:02 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users  wit
18/06/12 13:50:03 INFO Utils: Successfully started service 'sparkDriver' on port 53592.
18/06/12 13:50:03 INFO SparkEnv: Registering MapOutputTracker
18/06/12 13:50:03 INFO SparkEnv: Registering BlockManagerMaster
18/06/12 13:50:03 INFO BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for ge
18/06/12 13:50:03 INFO BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
18/06/12 13:50:03 INFO DiskBlockManager: Created local directory at C:\Users\Bhaskar\AppData\Local\Temp\blockm
18/06/12 13:50:03 INFO MemoryStore: MemoryStore started with capacity 1447.8 MB
18/06/12 13:50:03 INFO SparkEnv: Registering OutputCommitCoordinator
18/06/12 13:50:04 INFO Utils: Successfully started service 'SparkUI' on port 4040.
18/06/12 13:50:04 INFO SparkUI: Bound SparkUI to 0.0.0.0, and started at http://192.168.56.1:4040
18/06/12 13:50:04 INFO Executor: Starting executor ID driver on host localhost
18/06/12 13:50:04 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferS
18/06/12 13:50:04 INFO NettyBlockTransferService: Server created on 192.168.56.1:53605
18/06/12 13:50:04 INFO BlockManager: Using org.apache.spark.storage.RandomBlockReplicationPolicy for block rep
18/06/12 13:50:04 INFO BlockManagerMaster: Registering BlockManager BlockManagerId(driver, 192.168.56.1, 53605
18/06/12 13:50:04 INFO BlockManagerMasterEndpoint: Registering block manager 192.168.56.1:53605 with 1447.8 MB
18/06/12 13:50:04 INFO BlockManagerMaster: Registered BlockManager BlockManagerId(driver, 192.168.56.1, 53605,
18/06/12 13:50:04 INFO BlockManager: Initialized BlockManager: BlockManagerId(driver, 192.168.56.1, 53605, Non
Spark Context Created
Spark Streaming Context Created
```



```
18/06/12 13:50:04 INFO BlockManagerMaster: Registered BlockManager BlockManagerId(driver, 192.168.56.1, 53605, None)
18/06/12 13:50:04 INFO BlockManager: Initialized BlockManager: BlockManagerId(driver, 192.168.56.1, 53605, None)
Spark Context Created
Spark Streaming Context Created
Lines with even sum:
Sum of the numbers in even lines:
18/06/12 13:50:08 WARN RandomBlockReplicationPolicy: Expecting 1 replicas with only 0 peer/s.
18/06/12 13:50:08 WARN BlockManager: Block input-0-1528791608000 replicated to only 0 peer(s) instead of 1 peers
-------------------------------------------
Time: 1528791620000 ms
-------------------------------------------
How35 are27 you23
manage45 all56

-------------------------------------------
Time: 1528791620000 ms
-------------------------------------------
42.0

-------------------------------------------
Time: 1528791640000 ms
-------------------------------------------

-------------------------------------------
Time: 1528791640000 ms
-------------------------------------------
```

In above screenshot we are able to see that lines containing sum of odd numbers are filter, even number is displayed, and sum of the number is displayed in the next line.

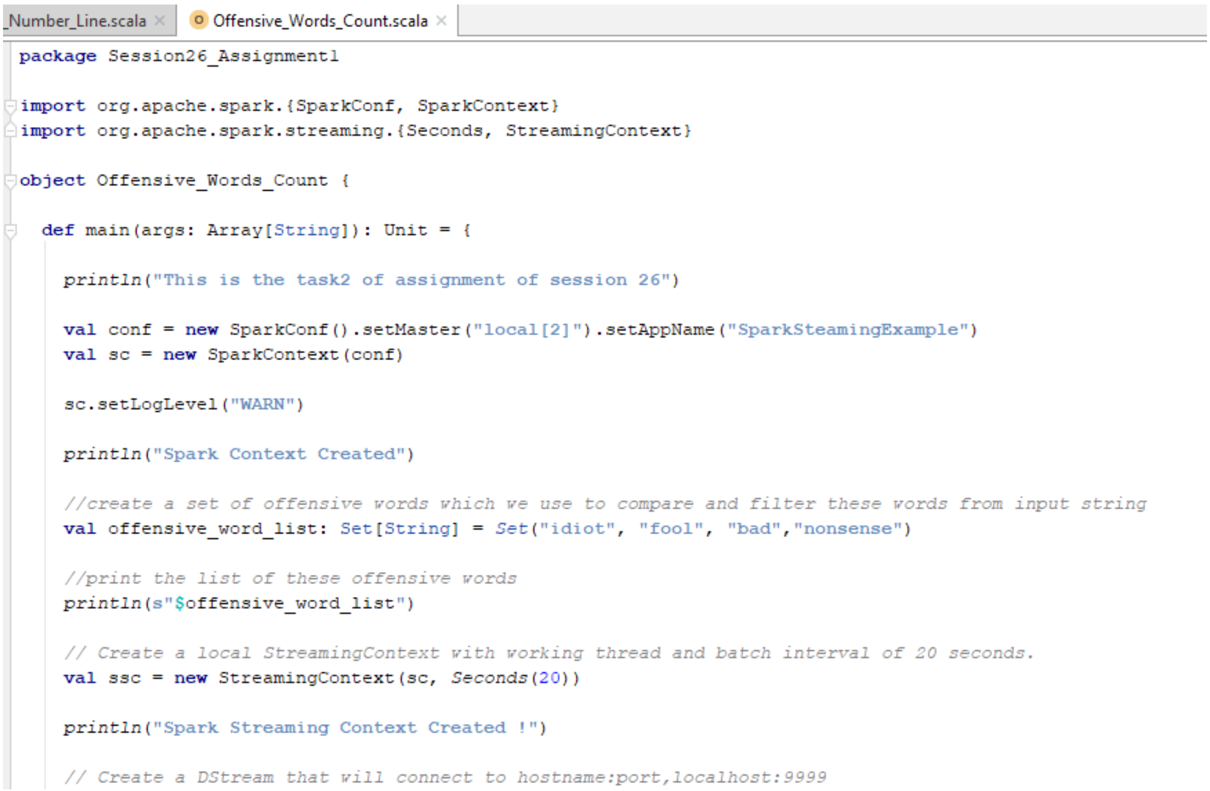
Task 2

Read two streams

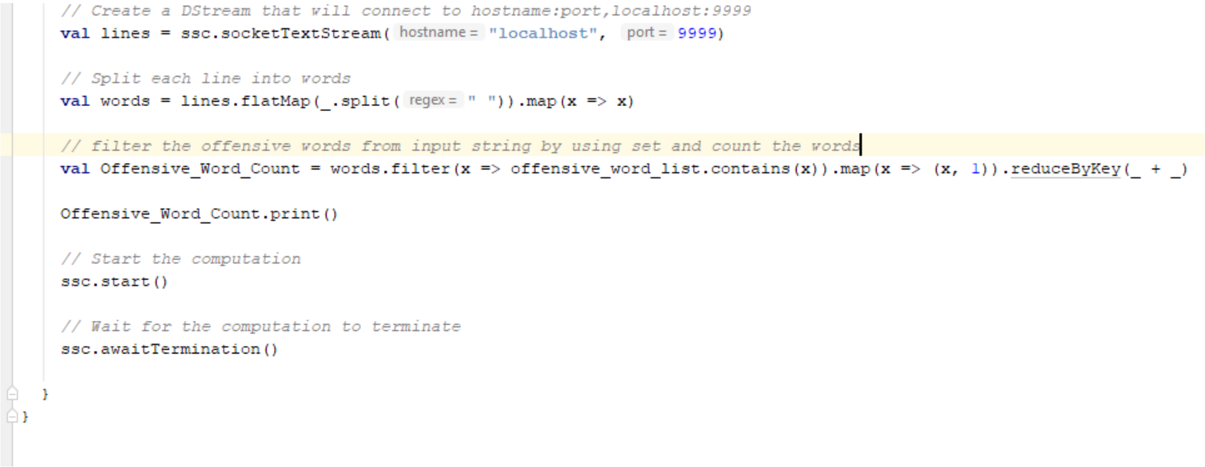
1. List of strings input by user
2. Real-time set of offensive words

Find the word count of the offensive words inputted by the user as per the real-time set of offensive words

Below screen shot shows the spark application to filter the offensive words form input string entered by user



```scala
package Session26_Assignment1

import org.apache.spark.{SparkConf, SparkContext}
import org.apache.spark.streaming.{Seconds, StreamingContext}

object Offensive_Words_Count {

  def main(args: Array[String]): Unit = {

    println("This is the task2 of assignment of session 26")

    val conf = new SparkConf().setMaster("local[2]").setAppName("SparkSteamingExample")
    val sc = new SparkContext(conf)

    sc.setLogLevel("WARN")

    println("Spark Context Created")

    //create a set of offensive words which we use to compare and filter these words from input string
    val offensive_word_list: Set[String] = Set("idiot", "fool", "bad","nonsense")

    //print the list of these offensive words
    println(s"$offensive_word_list")

    // Create a local StreamingContext with working thread and batch interval of 20 seconds.
    val ssc = new StreamingContext(sc, Seconds(20))

    println("Spark Streaming Context Created !")

    // Create a DStream that will connect to hostname:port,localhost:9999
```



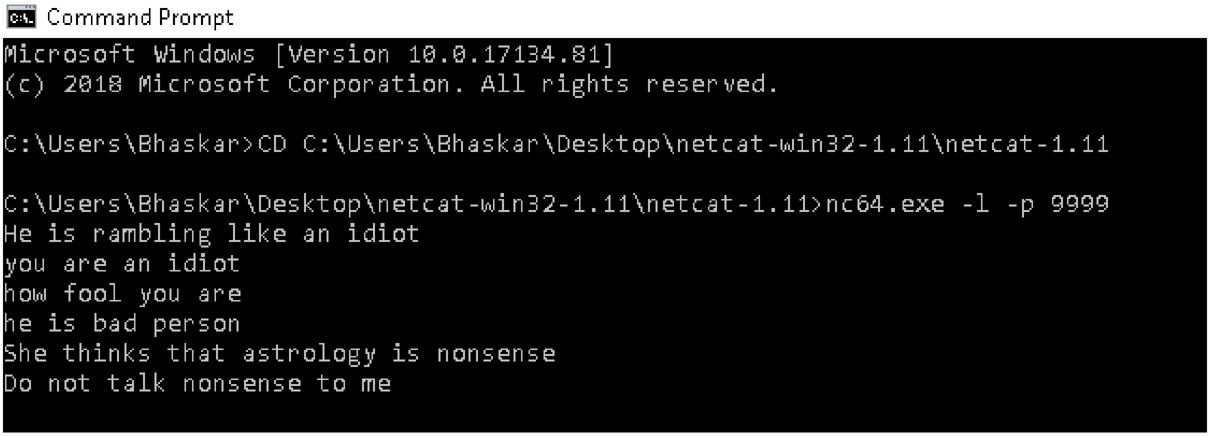
```
    // Create a DStream that will connect to hostname:port,localhost:9999
    val lines = ssc.socketTextStream( hostname = "localhost",  port = 9999)

    // Split each line into words
    val words = lines.flatMap(_.split( regex = " ")).map(x => x)

    // filter the offensive words from input string by using set and count the words
    val Offensive_Word_Count = words.filter(x => offensive_word_list.contains(x)).map(x => (x, 1)).reduceByKey(_ + _)

    Offensive_Word_Count.print()

    // Start the computation
    ssc.start()

    // Wait for the computation to terminate
    ssc.awaitTermination()

  }
}
```

In the above spark application, we have a set of words that we considered as offensive words.

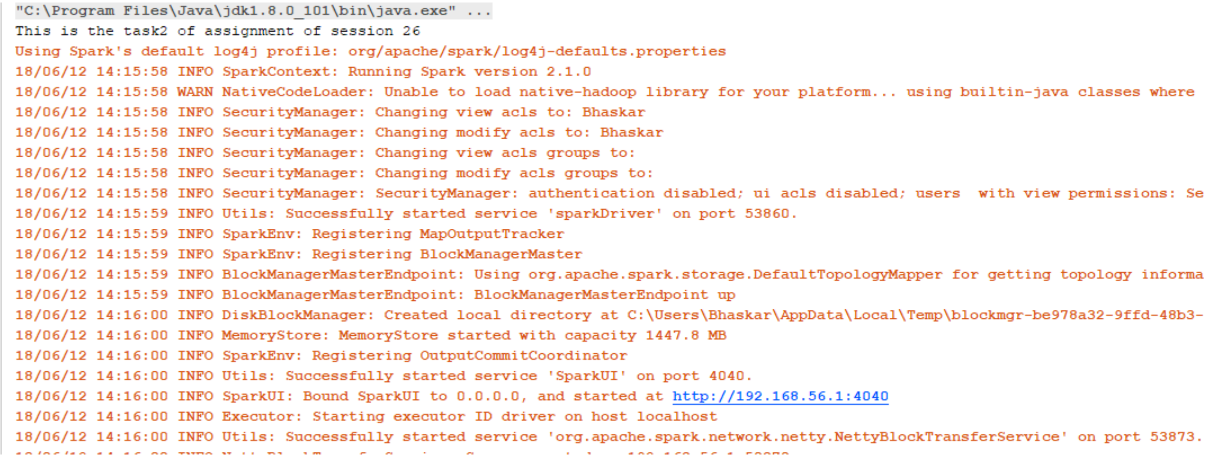
“idiot”,”fool”,”bad” ,“nonsense”

Let us start the **netcat** and put some input string containing above offensive words, which will be count by the spark application as shown below.

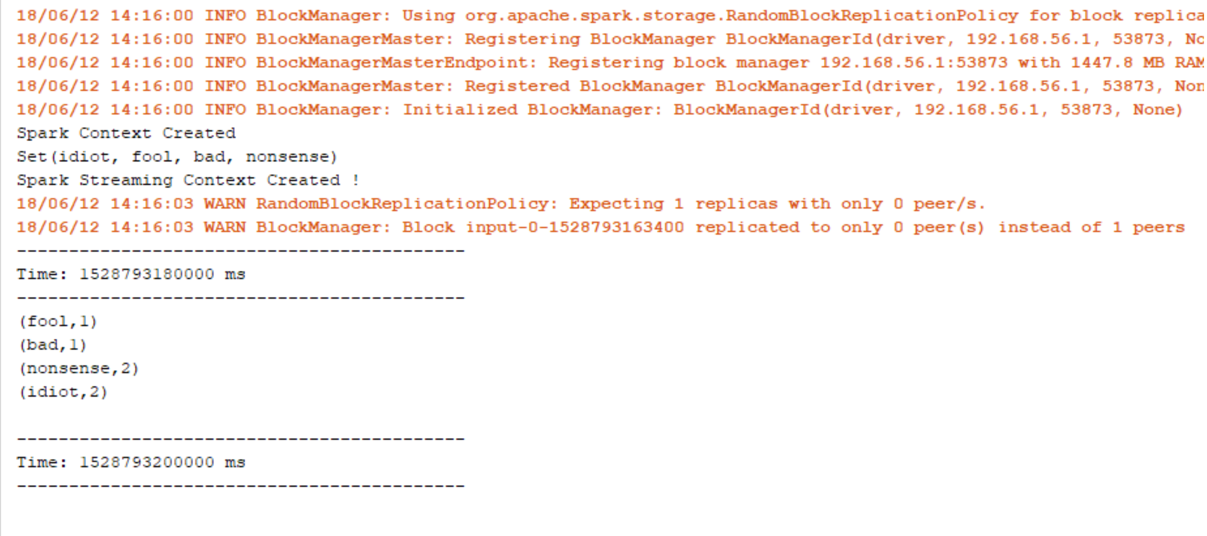
He is rambling like an idiot
You are an idiot
How fool you are
He is bad person
She thinks that astrology is nonsense
Do not talk nonsense to me



```
Microsoft Windows [Version 10.0.17134.81]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Bhaskar>CD C:\Users\Bhaskar\Desktop\netcat-win32-1.11\netcat-1.11

C:\Users\Bhaskar\Desktop\netcat-win32-1.11\netcat-1.11>nc64.exe -l -p 9999
He is rambling like an idiot
you are an idiot
how fool you are
he is bad person
She thinks that astrology is nonsense
Do not talk nonsense to me
```



```
"C:\Program Files\Java\jdk1.8.0_101\bin\java.exe" ...
This is the task2 of assignment of session 26
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
18/06/12 14:15:58 INFO SparkContext: Running Spark version 2.1.0
18/06/12 14:15:58 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where
18/06/12 14:15:58 INFO SecurityManager: Changing view acls to: Bhaskar
18/06/12 14:15:58 INFO SecurityManager: Changing modify acls to: Bhaskar
18/06/12 14:15:58 INFO SecurityManager: Changing view acls groups to:
18/06/12 14:15:58 INFO SecurityManager: Changing modify acls groups to:
18/06/12 14:15:58 INFO SecurityManager: SecurityManager: authentication disabled; ui acls disabled; users  with view permissions: Se
18/06/12 14:15:59 INFO Utils: Successfully started service 'sparkDriver' on port 53860.
18/06/12 14:15:59 INFO SparkEnv: Registering MapOutputTracker
18/06/12 14:15:59 INFO SparkEnv: Registering BlockManagerMaster
18/06/12 14:15:59 INFO BlockManagerMasterEndpoint: Using org.apache.spark.storage.DefaultTopologyMapper for getting topology informa
18/06/12 14:15:59 INFO BlockManagerMasterEndpoint: BlockManagerMasterEndpoint up
18/06/12 14:16:00 INFO DiskBlockManager: Created local directory at C:\Users\Bhaskar\AppData\Local\Temp\blockmgr-be978a32-9ffd-48b3-
18/06/12 14:16:00 INFO MemoryStore: MemoryStore started with capacity 1447.8 MB
18/06/12 14:16:00 INFO SparkEnv: Registering OutputCommitCoordinator
18/06/12 14:16:00 INFO Utils: Successfully started service 'SparkUI' on port 4040.
18/06/12 14:16:00 INFO SparkUI: Bound SparkUI to 0.0.0.0, and started at http://192.168.56.1:4040
18/06/12 14:16:00 INFO Executor: Starting executor ID driver on host localhost
18/06/12 14:16:00 INFO Utils: Successfully started service 'org.apache.spark.network.netty.NettyBlockTransferService' on port 53873.
```



```
18/06/12 14:16:00 INFO BlockManager: Using org.apache.spark.storage.RandomBlockReplicationPolicy for block replica
18/06/12 14:16:00 INFO BlockManagerMaster: Registering BlockManager BlockManagerId(driver, 192.168.56.1, 53873, No
18/06/12 14:16:00 INFO BlockManagerMasterEndpoint: Registering block manager 192.168.56.1:53873 with 1447.8 MB RAM
18/06/12 14:16:00 INFO BlockManagerMaster: Registered BlockManager BlockManagerId(driver, 192.168.56.1, 53873, Non
18/06/12 14:16:00 INFO BlockManager: Initialized BlockManager: BlockManagerId(driver, 192.168.56.1, 53873, None)
Spark Context Created
Set(idiot, fool, bad, nonsense)
Spark Streaming Context Created !
18/06/12 14:16:03 WARN RandomBlockReplicationPolicy: Expecting 1 replicas with only 0 peer/s.
18/06/12 14:16:03 WARN BlockManager: Block input-0-1528793163400 replicated to only 0 peer(s) instead of 1 peers
-------------------------------------------
Time: 1528793180000 ms
-------------------------------------------
(fool,1)
(bad,1)
(nonsense,2)
(idiot,2)

-------------------------------------------
Time: 1528793200000 ms
-------------------------------------------
```

In above screen shot, we are able to see that spark application has count the number of offensive words occur as per the input string provided by the user.