

Session 19:

RDD DEEP DIVE

Assignment 1

Task 1

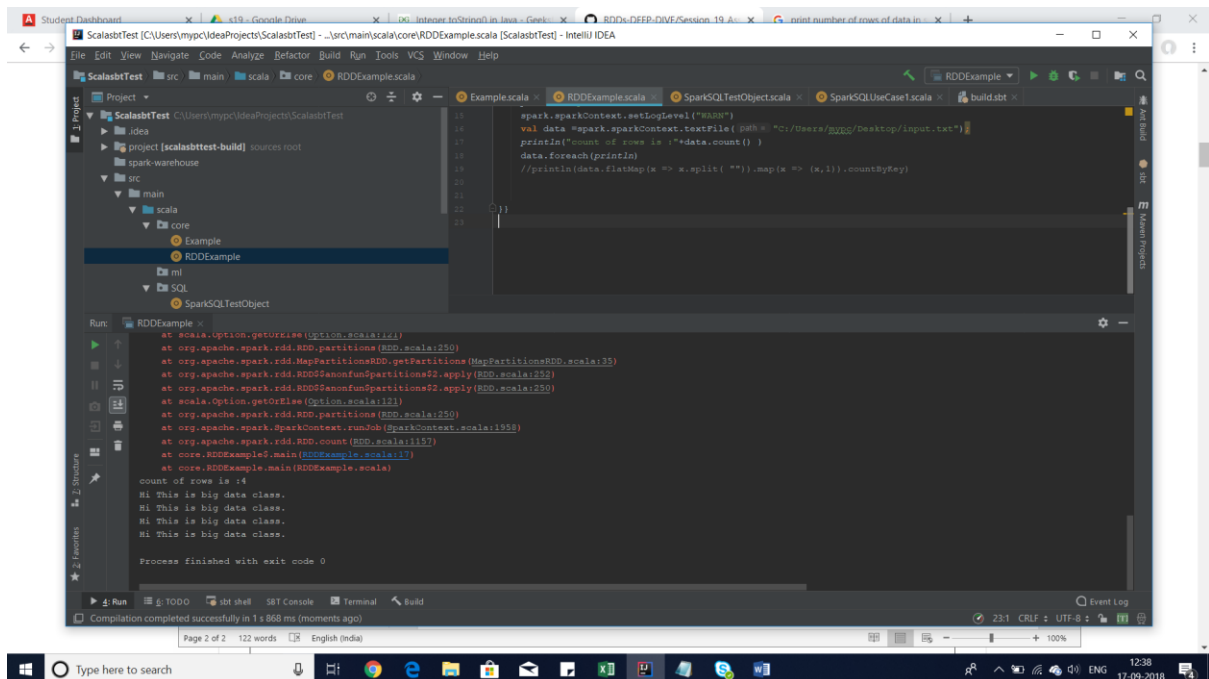
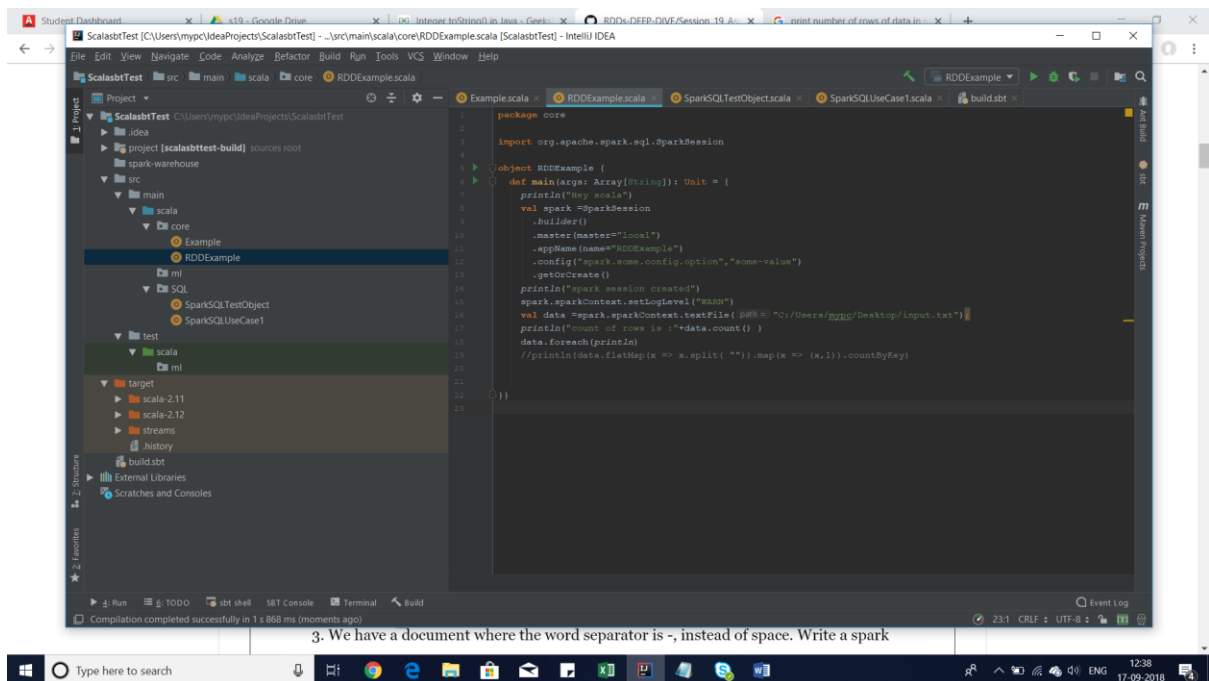
1. Write a program to read a text file and print the number of rows of data in the document.
2. Write a program to read a text file and print the number of words in the document.
3. We have a document where the word separator is -, instead of space. Write a spark code, to obtain the count of the total number of words present in the document.

```
package core
```

```
import org.apache.spark.sql.SparkSession
```

```
object RDDExample {  
  def main(args: Array[String]): Unit = {  
    println("Hey scala")  
    val spark = SparkSession  
      .builder()  
      .master(master="local")  
      .appName(name="RDDExample")  
      .config("spark.some.config.option", "some-value")  
      .getOrCreate()  
    println("spark session created")  
    spark.sparkContext.setLogLevel("WARN")  
    val data = spark.sparkContext.textFile("C:/Users/myipc/Desktop/input.txt");  
    println("count of rows is :"+data.count() )  
    data.foreach(println)  
    //println(data.flatMap(x => x.split( " ")).map(x => (x,1)).countByKey)
```

}}

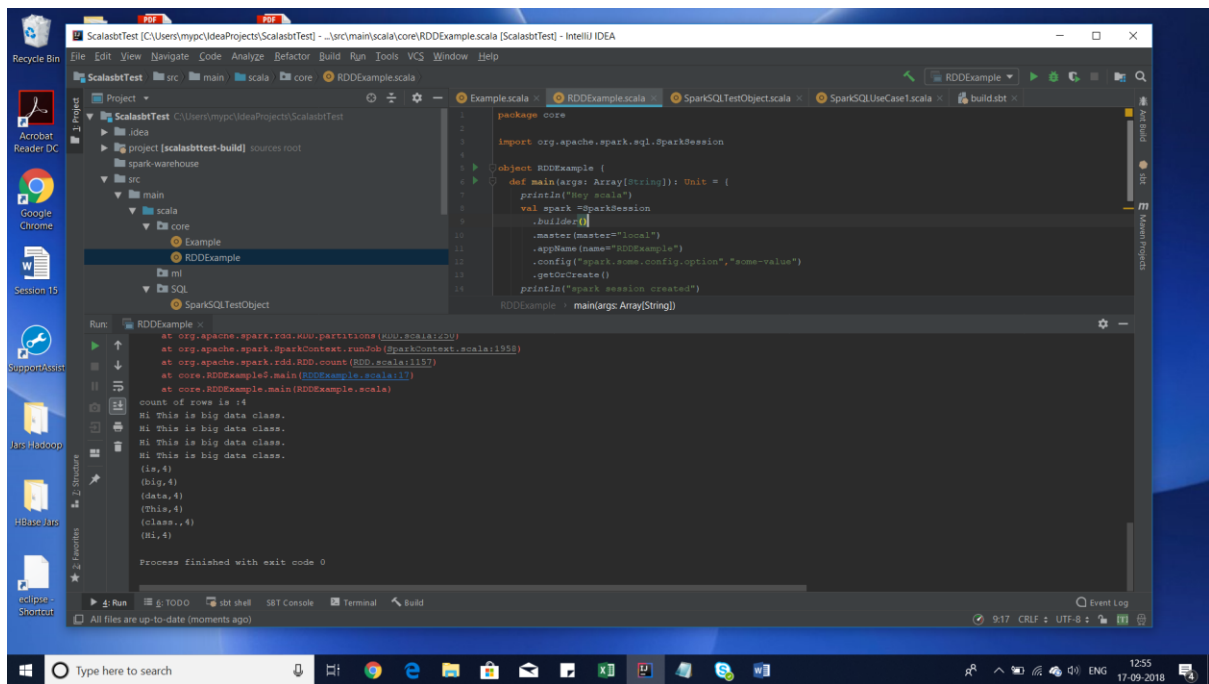
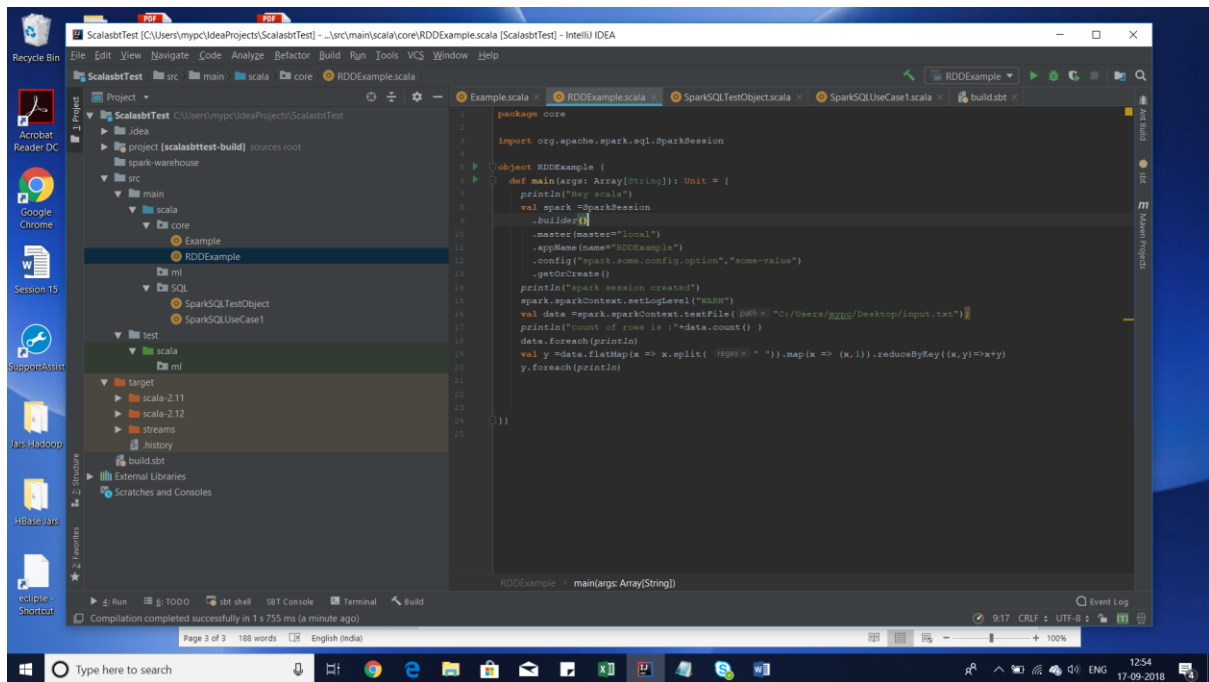


2. Write a program to read a text file and print the number of words in the document.

package core

import org.apache.spark.sql.{SparkSession}

```
object RDDExample {  
  def main(args: Array[String]): Unit = {  
    println("Hey scala")  
    val spark = SparkSession  
      .builder()  
      .master(master="local")  
      .appName(name="RDDExample")  
      .config("spark.some.config.option", "some-value")  
      .getOrCreate()  
    println("spark session created")  
    spark.sparkContext.setLogLevel("WARN")  
    val data = spark.sparkContext.textFile("C:/Users/myipc/Desktop/input.txt");  
    println("count of rows is :"+data.count() )  
    data.foreach(println)  
    val y = data.flatMap(x => x.split( " ")).map(x => (x,1)).reduceByKey((x,y)=>x+y)  
    y.foreach(println)  
  
  }  
}
```



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.

package core

import org.apache.spark.sql.{SparkSession, SparkContext}

```

object RDDExample {
  def main(args: Array[String]): Unit = {
    println("Hey scala")
    val spark = SparkSession
      .builder()
      .master(master="local")
      .appName(name="RDDExample")
      .config("spark.some.config.option", "some-value")
      .getOrCreate()
    println("spark session created")
    spark.sparkContext.setLogLevel("WARN")
    val data = spark.sparkContext.textFile("C:/Users/myipc/Desktop/input.txt");
    println("count of rows is :"+data.count() )
    data.foreach(println)
    //val y =data.flatMap(x => x.split(" ")).map(x => (x,1)).reduceByKey((x,y)=>x+y)
    //y.foreach(println)
    val z =data.flatMap(x => x.split("-")).map(x => (x,1)).reduceByKey((x,y)=>x+y)
    z.foreach(println)

  }
}

```

```

package core

import org.apache.spark.sql.SparkSession

object RDDExample {
  def main(args: Array[String]): Unit = {
    println("hey scala")
    val spark = SparkSession
      .builder()
      .master("local")
      .appName("name=RDDExample")
      .config("spark.some.config.option", "some-value")
      .getOrCreate()

    println("spark session created")
    spark.sparkContext.setLogLevel("WARN")
    val data = spark.sparkContext.textFile("HDFS://C:/Users/pooja/Desktop/input.txt")
    println("count of rows is " + data.count())
    data.foreach(println)

    //val y = data.flatMap(x => x.split(" ").map(x => (x, 1))).reduceByKey((x, y) => x + y)
    //y.foreach(println)
    val z = data.flatMap(x => x.split(" (age=) ")).map(x => (x, 1)).reduceByKey((x, y) => x + y)
    z.foreach(println)
  }
}

```

```

Run: RDDExample x
at org.apache.spark.sql.SparkSession.runJob(SparkSession.scala:1958)
at org.apache.spark.rdd.RDD.count(RDD.scala:1157)
at core.RDDExample$.main(RDDExample.scala:17)
at core.RDDExample.main(RDDExample.scala)

count of rows is 4
Hi-This-is-big-data-class.
Hi-This-is-big-data-class.
Hi-This-is-big-data-class.
Hi-This-is-big-data-class.
(i.e, 4)
(big, 4)
(data, 4)
(This, 4)
(class, 4)
(Hi, 4)

Process finished with exit code 0

```

Task 2

Problem Statement 1:

1. Read the text file, and create a tupled rdd.
2. Find the count of total number of rows present.
3. What is the distinct number of subjects present in the entire school
4. What is the count of the number of students in the school, whose name is Mathew and

marks is 55

Solution:

package core

```
import org.apache.spark.sql.Session
```

```
object RDDExample {
```

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

```
    println("Hey scala")
```

```
    val spark = SparkSession
```

```
      .builder()
```

```
      .master(master="local")
```

```
      .appName(name="RDDExample")
```

```
      .config("spark.some.config.option", "some-value")
```

```
      .getOrCreate()
```

```
    println("spark session created")
```

```
    spark.sparkContext.setLogLevel("WARN")
```

```
    val data = spark.sparkContext.textFile("C:/Users/myipc/Desktop/Dataset.txt");
```

```
    println("count of rows is :"+data.count() )
```

```
    //data.foreach(println)
```

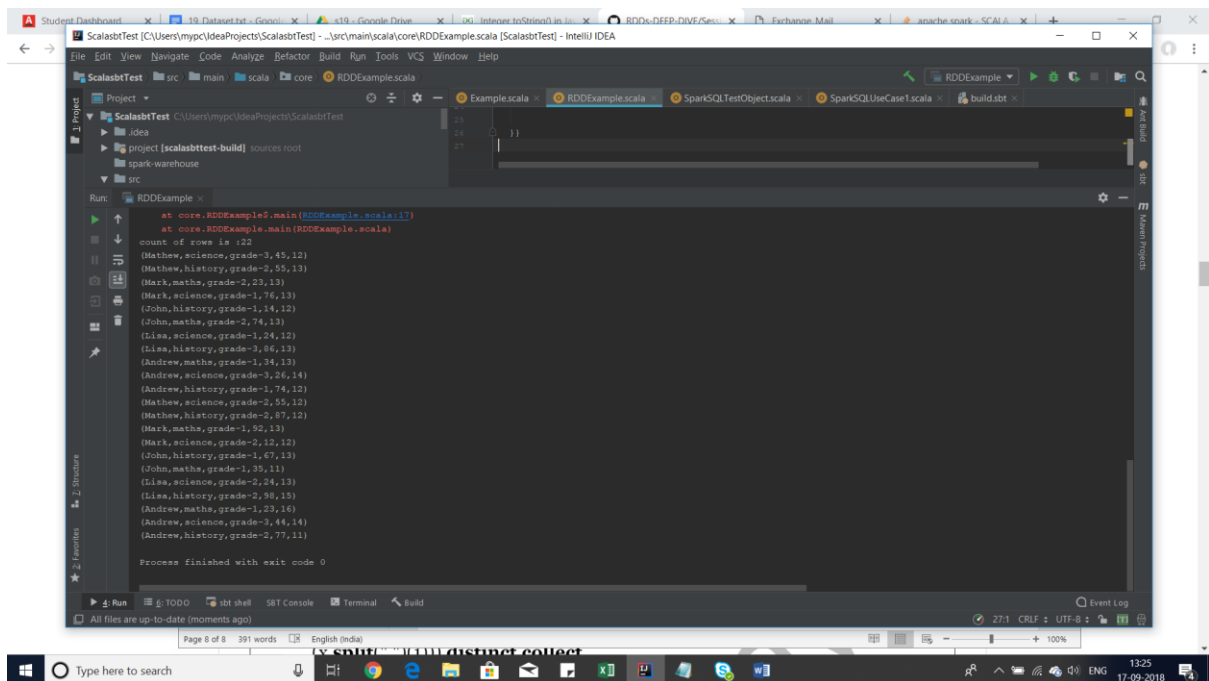
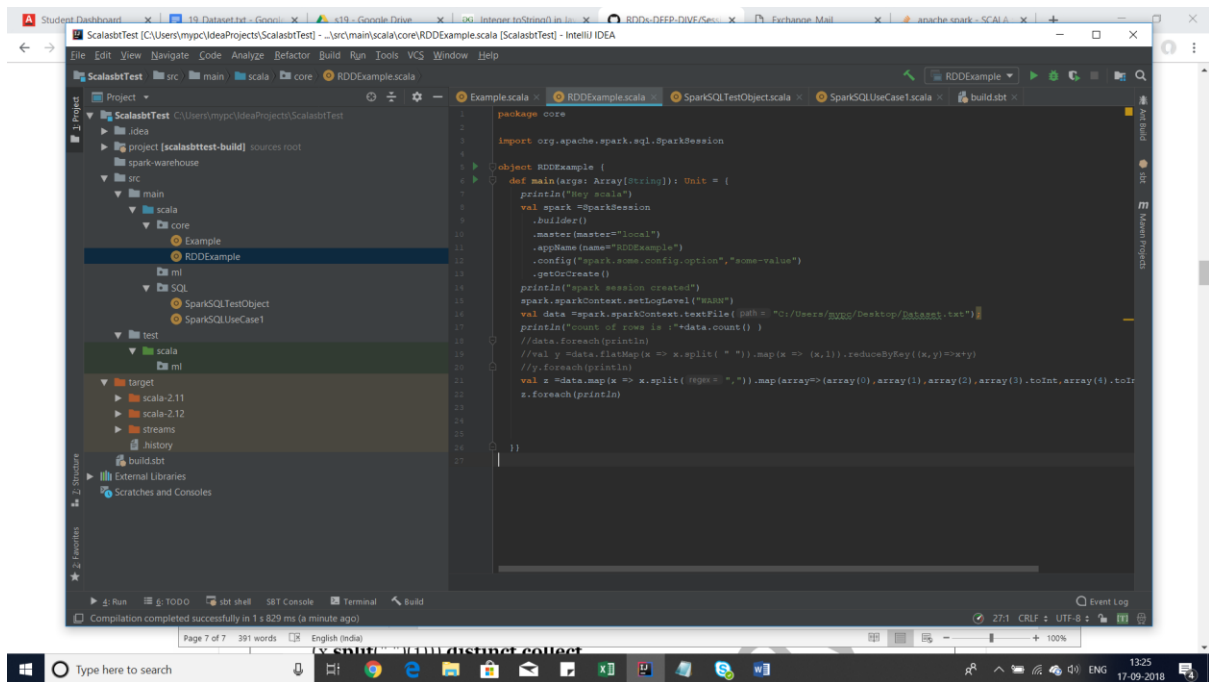
```
    //val y = data.flatMap(x => x.split(" ")).map(x => (x,1)).reduceByKey((x,y)=>x+y)
```

```
    //y.foreach(println)
```

```
    val z = data.map(x =>
x.split(",")).map(array=>(array(0),array(1),array(2),array(3).toInt,array(4).toInt)).collect
```

```
    z.foreach(println)
```

```
  }
```

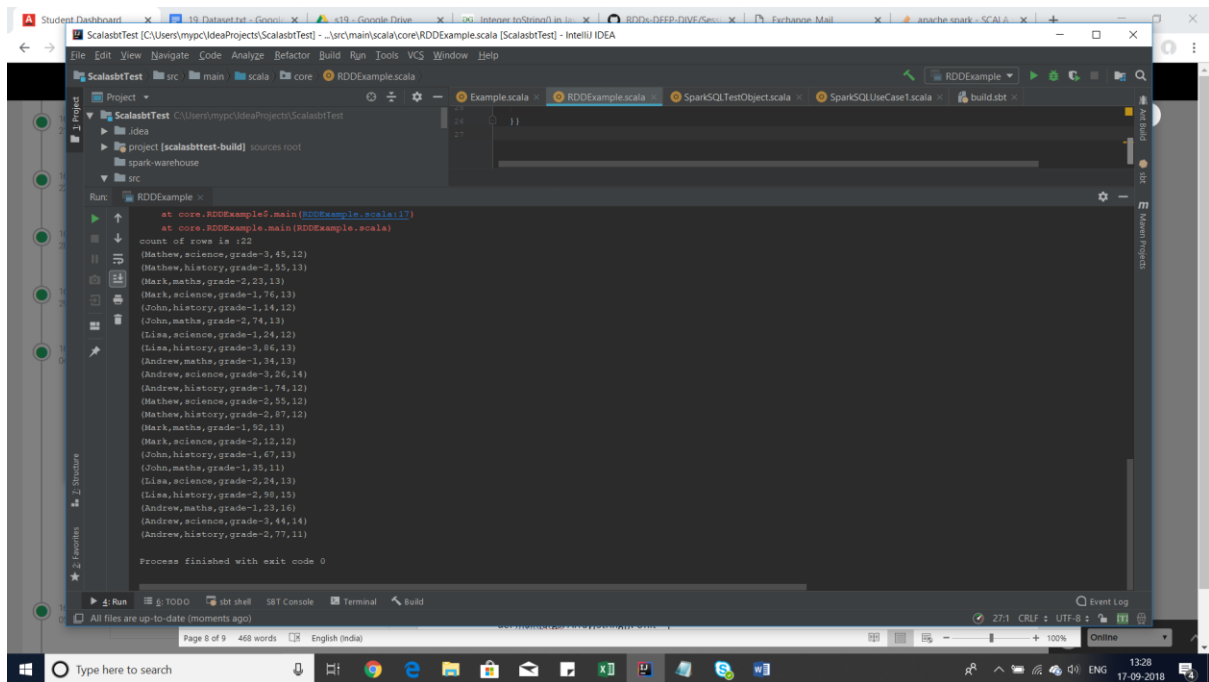


2. Find the count of total number of rows present.

Solution: We can find the count of rows using

“println("count of rows is :"+data.count())” in the program


```
object RDDExample {  
  def main(args: Array[String]): Unit = {  
    println("Hey scala")  
    val spark = SparkSession  
      .builder()  
      .master(master="local")  
      .appName(name="RDDExample")  
      .config("spark.some.config.option", "some-value")  
      .getOrCreate()  
    println("spark session created")  
    spark.sparkContext.setLogLevel("WARN")  
    val data = spark.sparkContext.textFile("C:/Users/myipc/Desktop/Dataset.txt");  
    println("count of rows is :"+data.count() )  
    val z = data.map(x =>  
x.split(",")).map(array=>(array(0),array(1),array(2),array(3).toInt,array(4).toInt)).collect  
    z.foreach(println)  
  
  }  
}
```



What is the distinct number of subjects present in the entire school

Solution:

package core

import org.apache.spark.sql.SparkSession

object RDDExample {

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

println("Hey scala")

val spark =SparkSession

.builder()

.master(master="local")

.appName(name="RDDExample")

.config("spark.some.config.option","some-value")

.getOrCreate()

println("spark session created")

spark.sparkContext.setLogLevel("WARN")

```

val data = spark.sparkContext.textFile("C:/Users/myopc/Desktop/Dataset.txt");

println("count of rows is :"+data.count() )

val z = data.map(x => x.split(",")).map(array=>(array(1))).distinct.collect

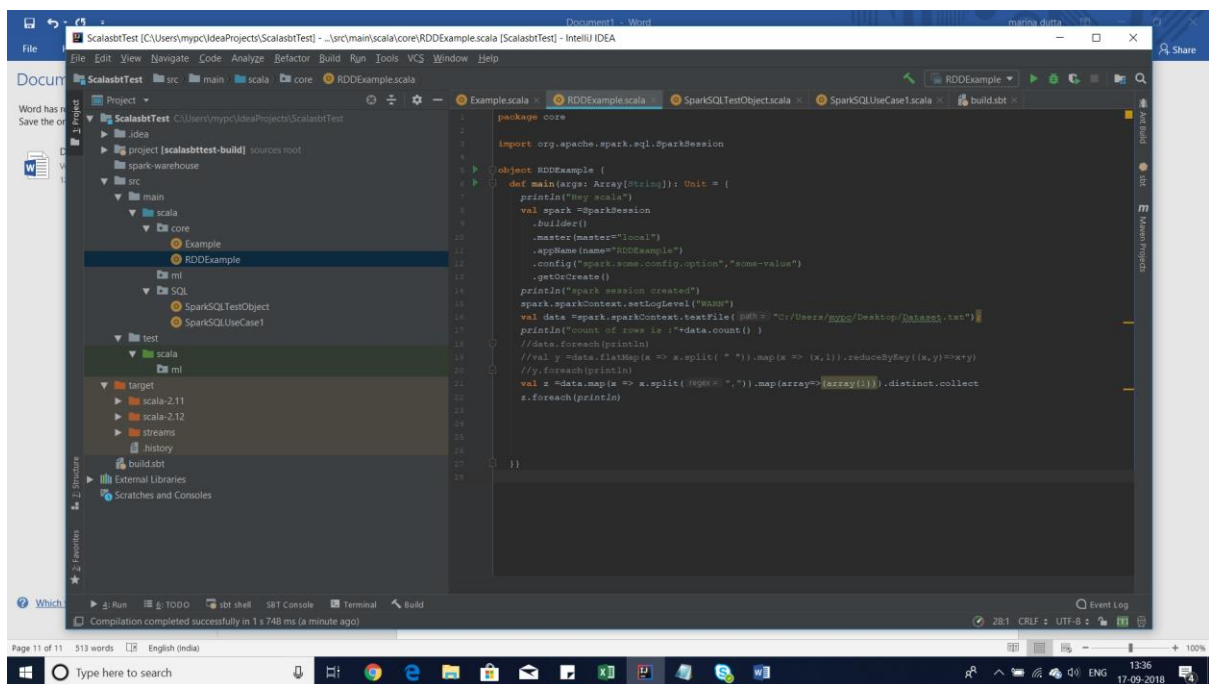
z.foreach(println)

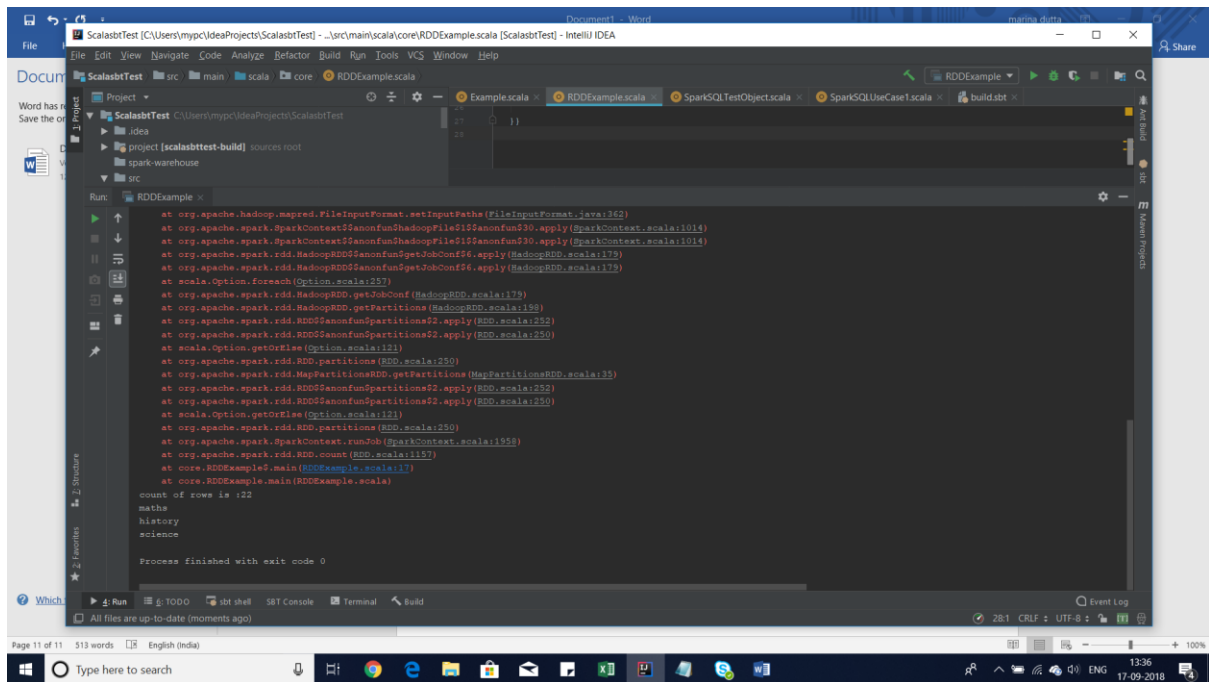
```

```

}}

```





4. What is the count of the number of students in the school, whose name is Mathew and marks is 55

Solution:

package core

```
import org.apache.spark.sql.SparkSession
```

```
object RDDExample {  
  
  def main(args: Array[String]): Unit = {  
  
    println("Hey scala")  
  
    val spark = SparkSession  
  
      .builder()  
  
      .master(master="local")  
  
      .appName(name="RDDExample")  
  
      .config("spark.some.config.option", "some-value")  
  
      .getOrCreate()  
  
  }  
  
}
```

```
println("spark session created")
```

```
spark.sparkContext.setLogLevel("WARN")
```

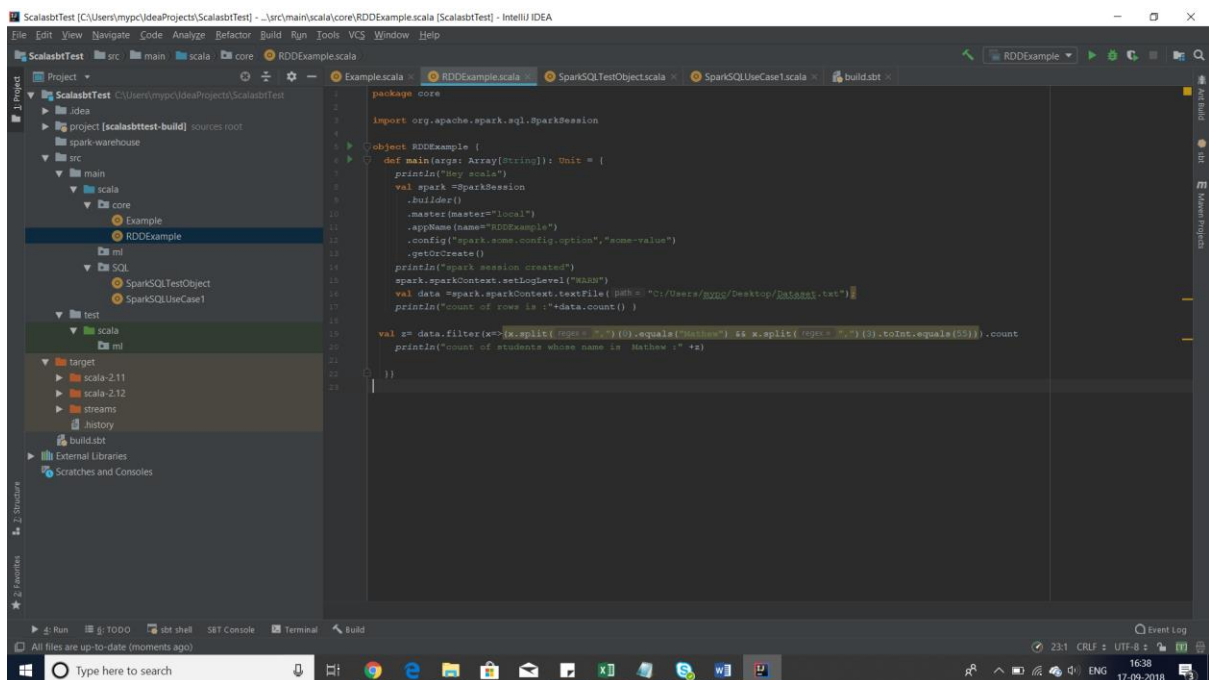
```
val data = spark.sparkContext.textFile("C:/Users/myopc/Desktop/Dataset.txt");
```

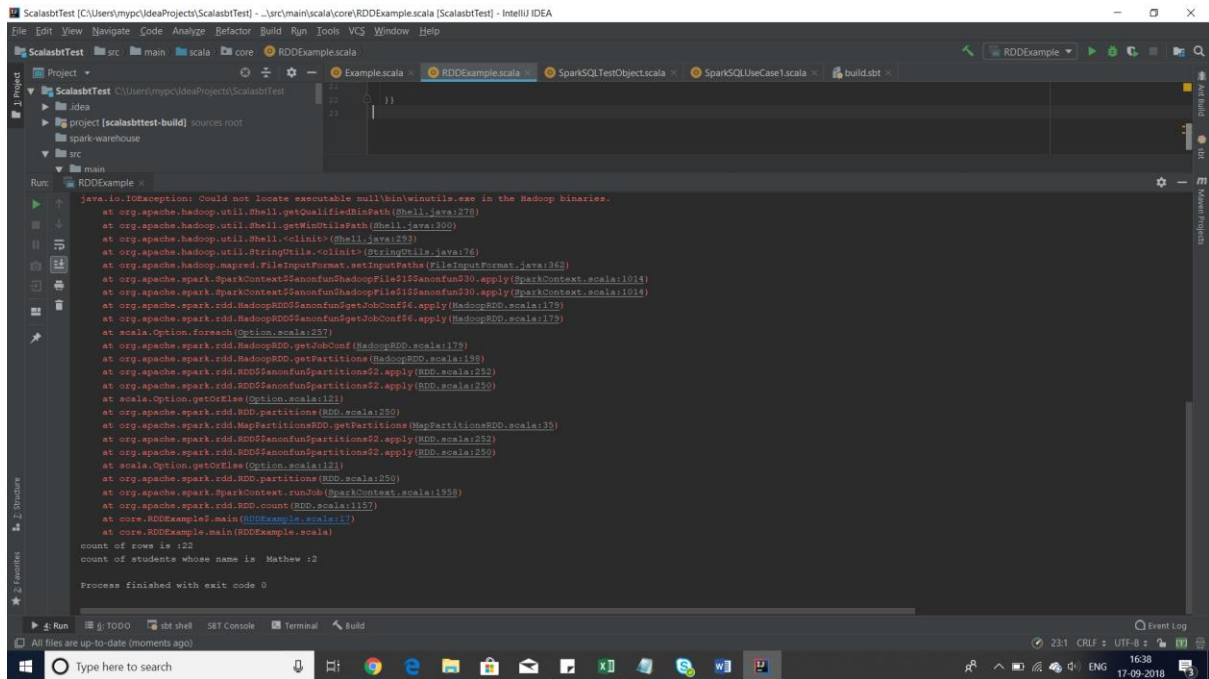
```
println("count of rows is :"+data.count() )
```

```
val z= data.filter(x=>(x.split(",")(0).equals("Mathew") && x.split(",")(3).toInt.equals(55))).count
```

```
println("count of students whose name is Mathew :"+z)
```

```
}}
```





Problem Statement 2:

1. What is the count of students per grade in the school?
import org.apache.spark.sql.Session

```
object RDDExample {
  def main(args: Array[String]): Unit = {
    println("Hey scala")
    val spark = SparkSession
      .builder()
      .master(master="local")
      .appName(name="RDDExample")
      .config("spark.some.config.option", "some-value")
      .getOrCreate()
    println("spark session created")
    spark.sparkContext.setLogLevel("WARN")
    val data = spark.sparkContext.textFile("C:/Users/myopc/Desktop/Dataset.txt");
    println("count of rows is :"+data.count() )

    val z = data.map(x=>x.split(",")(2)).map(x=>(x,1)).reduceByKey((x,y)=>x+y)

    println("count of students per grade is : " + " val y = z.foreach(println)")
    val y = z.foreach(println)

  }
}
```

The screenshot shows the IntelliJ IDEA interface with the `RDDExample.scala` file open. The code defines an `RDDExample` object with a `main` method. The `main` method performs the following steps:

- Imports `org.apache.spark.sql.SparkSession`.
- Creates a `SparkSession` with `local` as the master and `RDDExample` as the app name.
- Loads a dataset from a text file located at `"C:/Users/ggg/Desktop/Dataset.txt"`.
- Prints the count of rows in the dataset.
- Maps each row (split by comma) to a tuple `(x, y)` where `x` is the grade and `y` is the student name.
- Reduces the data by grade to calculate the count of students per grade.
- Prints the count of students per grade.

The Run console shows the following output:

```
at org.apache.spark.rdd.RDD.count(RDD.scala:1157)
at core.RDDExample$.main(RDDExample.scala:12)
at core.RDDExample.main(RDDExample.scala)
count of rows is 122
count of students per grade is : val y = z.foreach(println)
(grade-3,4)
(grade-1,9)
(grade-2,9)
Process finished with exit code 0
```

This screenshot is identical to the one above, showing the same code and execution output for the `RDDExample.scala` file.

2. Find the average of each student (Note - Mathew is grade-1, is different from Mathew in some other grade!)

```
import org.apache.spark.sql.SparkSession

object RDDExample {
  def main(args: Array[String]): Unit = {
    println("Hey scala")
    val spark = SparkSession
      .builder()
      .master(master = "local")
      .appName(name = "RDDExample")
      .config("spark.some.config.option", "some-value")
      .getOrCreate()
    println("spark session created")
    spark.sparkContext.setLogLevel("WARN")
    val data = spark.sparkContext.textFile("C:/Users/myipc/Desktop/Dataset.txt")
    println("count of rows is : " + data.count())

    val z = data.map(x => ((x.split(",")(0), x.split(",")(2)), x.split(",")(3).toDouble)).groupByKey().map(x
=> (x._1, x._2.sum / x._2.size)).foreach(println)

  }
}
```



```

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

  println("Hey scala")

  val spark = SparkSession

    .builder()

    .master(master = "local")

    .appName(name = "RDDEExample")

    .config("spark.some.config.option", "some-value")

    .getOrCreate()

  println("spark session created")

  spark.sparkContext.setLogLevel("WARN")

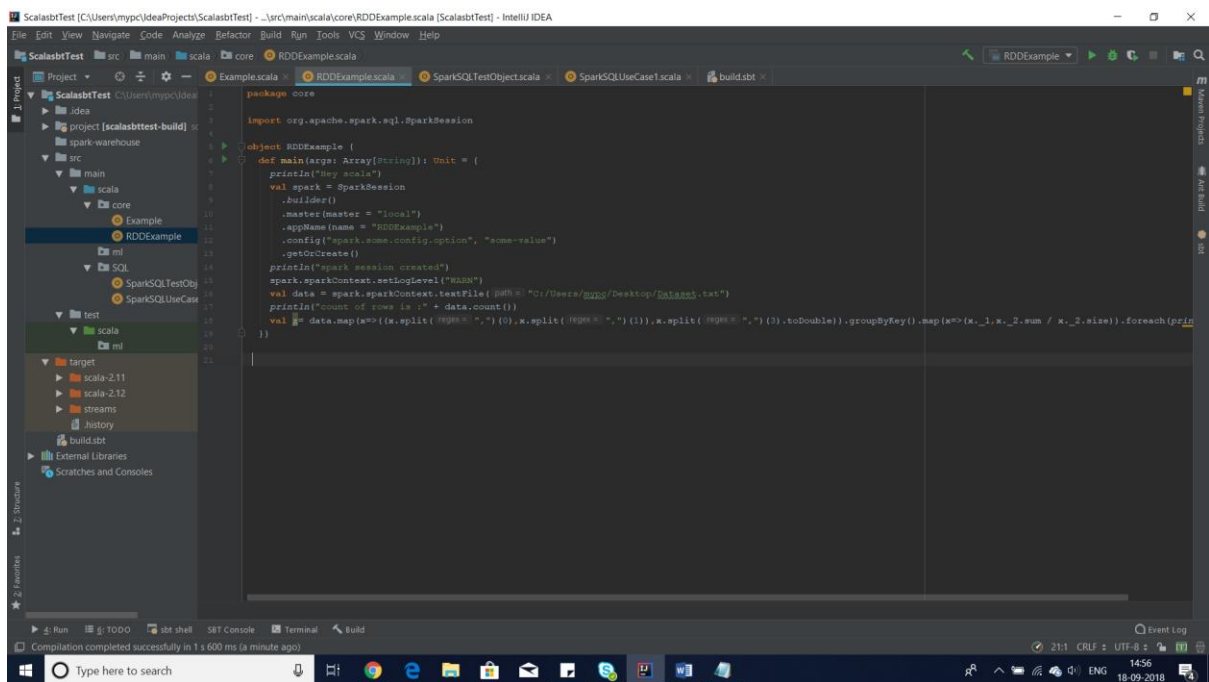
  val data = spark.sparkContext.textFile("C:/Users/myipc/Desktop/Dataset.txt")

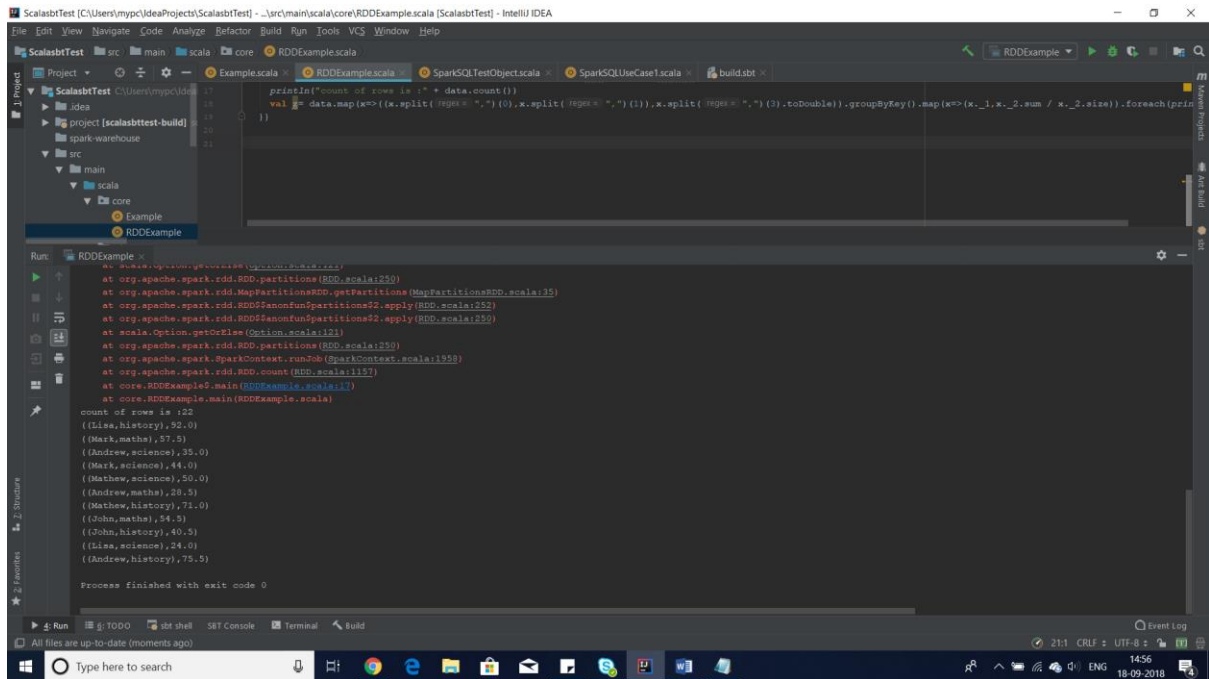
  println("count of rows is : " + data.count())

  val z=
data.map(x=>((x.split(",")(0),x.split(",")(1)),x.split(",")(3).toDouble)).groupByKey().map(x=>(x._1,x._2.
sum / x._2.size)).foreach(println)

}

```





4. What is the average score of students in each subject per grade?

package core

import org.apache.spark.sql.Session

object RDDExample {

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

println("Hey scala")

val spark = SparkSession

.builder()

.master(master = "local")

.appName(name = "RDDExample")

.config("spark.some.config.option", "some-value")

.getOrCreate()

println("spark session created")

spark.sparkContext.setLogLevel("WARN")

```

val data = spark.sparkContext.textFile("C:/Users/mypc/Desktop/Dataset.txt")

println("count of rows is :" + data.count())

val z=
data.map(x=>((x.split(",")(0),x.split(",")(1),x.split(",")(2)),x.split(",")(3).toDouble)).groupByKey().map(
x=>(x._1,x._2.sum / x._2.size)).foreach(println)
}

```

```

package core

import org.apache.spark.sql.SparkSession

object RDDExample {
  def main(args: Array[String]): Unit = {
    println("key scala")
    val spark = SparkSession
      .builder()
      .master(master = "local")
      .appName(name = "RDDExample")
      .config("spark.some.config.option", "some-value")
      .getOrCreate()

    println("spark session created")
    spark.sparkContext.setLogLevel("WARN")
    val data = spark.sparkContext.textFile("C:/Users/mypc/Desktop/Dataset.txt")
    println("count of rows is :" + data.count())
    val z = data.map(x => ((x.split(",")(0), x.split(",")(1), x.split(",")(2)), x.split(",")(3).toDouble)).groupByKey().map(x => (x._1, x._2.sum / x._2.size))
  }
}

```

```

Run: RDDExample
at org.apache.spark.sql.execution.python.PythonUDFRunner.run(PythonUDFRunner.scala:222)
at org.apache.spark.rdd.RDD.count(RDD.scala:1157)
at core.RDDExample$.main(RDDExample.scala:17)
at core.RDDExample.main(RDDExample.scala)

count of rows is 122
((Lisa,history,grade-3),86.0)
((John,history,grade-1),40.5)
((Andrew,history,grade-2),77.0)
((John,maths,grade-2),74.0)
((Mark,maths,grade-2),23.0)
((Andrew,maths,grade-1),20.5)
((Andrew,science,grade-3),35.0)
((Mark,science,grade-2),12.0)
((Mathew,science,grade-2),45.0)
((Mathew,history,grade-2),71.0)
((Andrew,history,grade-1),74.0)
((John,maths,grade-1),35.0)
((Mark,maths,grade-1),52.0)
((Mark,science,grade-1),76.0)
((Mathew,science,grade-2),35.0)
((Lisa,science,grade-2),24.0)
((Lisa,history,grade-2),98.0)
((Lisa,science,grade-1),24.0)

Process finished with exit code 0

```

5. For all students in grade-2, how many have average score greater than 50?

ackage core

import org.apache.spark.sql.SparkSession

object RDDEExample {

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

println("Hey scala")

val spark = SparkSession

.builder()

.master(master = "local")

.appName(name = "RDDEExample")

.config("spark.some.config.option", "some-value")

.getOrCreate()

println("spark session created")

spark.sparkContext.setLogLevel("WARN")

val data = spark.sparkContext.textFile("C:/Users/myipc/Desktop/Dataset.txt")

println("count of rows is :" + data.count())

val z= data.filter(x=>(x.split(",")(2)).equals("grade-2")).map(x=>(x.split(",")(0),x.split(",")(3).toDouble)).groupByKey().map(x=>(x._1,x._2.sum / x._2.size)).filter(x=>(x._2>50)).foreach(println)

}}

