

Assignment 13.3

Problem Statement:

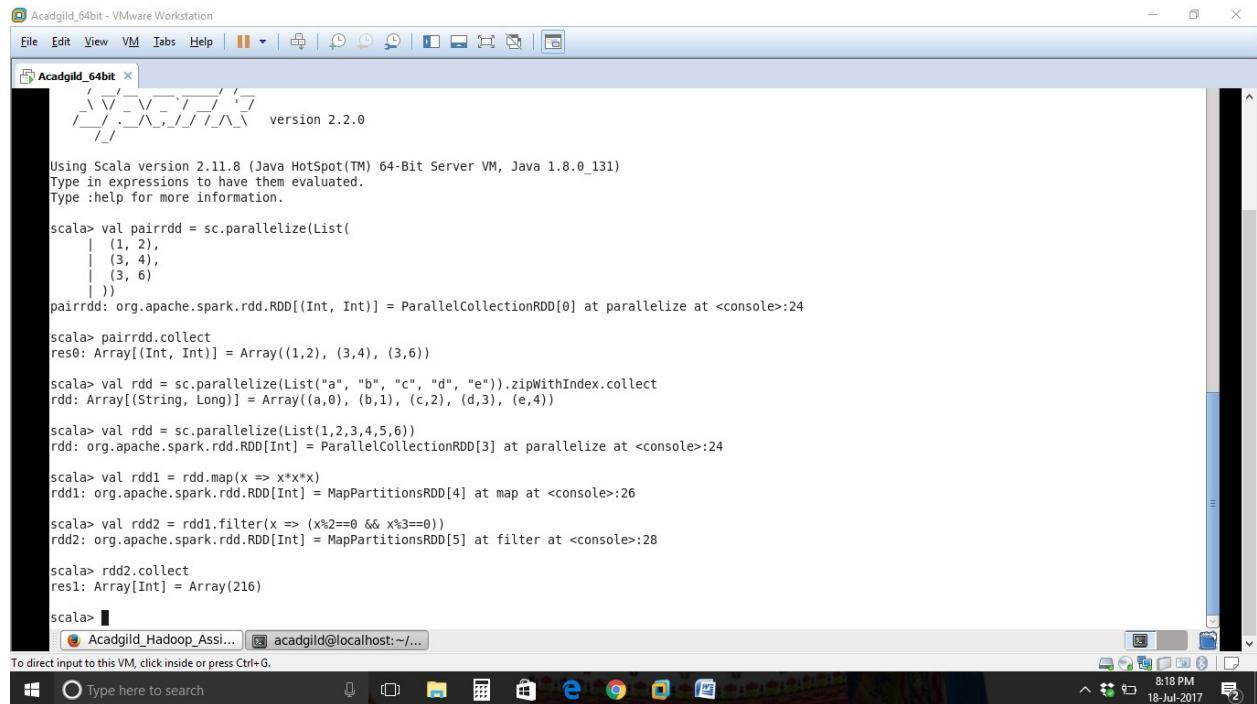
Write the code to Turn a collection into a RDD and perform map operation on it to cube every number and filter the number which are divided by two and three.

```
Val rdd = sc.parallelize(List(1,2,3,4,5,6))
```

```
Val rdd1 = rdd.map(x => x*x*x)
```

```
Val rdd2 = rdd1.filter( x => (x%2==0 && x%3==0))
```

```
Rdd2.collect
```



```
Acadgild_64bit - VMware Workstation
File Edit View VM Tabs Help
Acadgild_64bit x
version 2.2.0
Using Scala version 2.11.8 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_131)
Type in expressions to have them evaluated.
Type :help for more information.

scala> val pairrdd = sc.parallelize(List(
  | (1, 2),
  | (3, 4),
  | (3, 6)
  | ))
pairrdd: org.apache.spark.rdd.RDD[(Int, Int)] = ParallelCollectionRDD[0] at parallelize at <console>:24

scala> pairrdd.collect
res0: Array[(Int, Int)] = Array((1,2), (3,4), (3,6))

scala> val rdd = sc.parallelize(List("a", "b", "c", "d", "e")).zipWithIndex.collect
rdd: Array[(String, Long)] = Array((a,0), (b,1), (c,2), (d,3), (e,4))

scala> val rdd = sc.parallelize(List(1,2,3,4,5,6))
rdd: org.apache.spark.rdd.RDD[Int] = ParallelCollectionRDD[3] at parallelize at <console>:24

scala> val rdd1 = rdd.map(x => x*x*x)
rdd1: org.apache.spark.rdd.RDD[Int] = MapPartitionsRDD[4] at map at <console>:26

scala> val rdd2 = rdd1.filter(x => (x%2==0 && x%3==0))
rdd2: org.apache.spark.rdd.RDD[Int] = MapPartitionsRDD[5] at filter at <console>:28

scala> rdd2.collect
res1: Array[Int] = Array(216)

scala>
```