****

**SPARK RDD ASSIGNMENT SOLUTION:**

1. **Print each country’s suicide value?**

val data = sc.textFile("Address of the file")

val data1 = data.map(x => x.split(",")).map(x => (x(0),x(4).toInteger))

data1.collect.foreach(println)

1. **Find out the top ten countries which suicides cases have been reported highly?**

val data = sc.textFile("Address of the file")

val country = data.map(\_.split(",")(0))

val ccount = country.map(ccount => (ccount,1))

val countryCount = ccount.reduceByKey((x,y)=> x+y).map(tup.\_2,tup.\_1))sortByKey(false)

1. **Find how many male and female victims?**

val countMF = sc.textFile(“Address of the file”)

val gender = countMF.map(x => x.split(“,”)(2))

val count1 = gender.map(x => (x,1))

val count2 = count1.reducByKey(\_+\_)sortBy(\_.\_2)

count2.collect().foreach(println)

1. **Find out which country is having max, min population?**

val countryrdd = sc.textFile(“Address of the file”)

val Head = countryrdd.first()

val Nohead = countryrdd.filter(line => !line.equals(Head))

val max\_pop = Nohead.map(x => x.split(“,”)).map(x => (x(5).toInt,x(0)))

println(“ The country which is having highest population:”+max\_pop.max())

println(“The country which is having lowest population:”+max\_pop.min())

1. **Find out which country is having max, min gdp\_per\_capit, gdp\_per\_year?**

val gdpdata = Nohead.map(x => x.split(“,”)).map(x => (x(0).toString,x(11)))

println(“The country which is having max gdp\_per\_capita:”+gdpdata.max())

println(“The country which is having min gdp\_per\_capita:”+gdpdata.min())

val gdp1data = Nohead.map(x => x.split(“,”)).map(x => (x(0).toString,x(10)))

println(“The country which is having max gdp\_per\_year:”+gdp1data.max())

println(“The country which is having min gdp\_per\_year:”+gdp1data.min())