topPayment class

```
package sparkPayment1;


import java.util.ArrayList;
import java.util.Arrays;
import java.util.List;

import org.apache.spark.SparkConf;
import org.apache.spark.api.java.JavaPairRDD;
import org.apache.spark.api.java.JavaRDD;
import org.apache.spark.api.java.JavaSparkContext;

import scala.Tuple2;

public class topPayment {
        public static void main (String args[]) {
                //SparkConf conf = new
SparkConf().setAppName("PaymentCount").setMaster("local[*]");
                SparkConf conf = new SparkConf().setAppName("Payment Count");
                @SuppressWarnings("resource")
                JavaSparkContext sc = new JavaSparkContext(conf);
                // provide text file paths to be read to RDD, separated by comma
                //String files = "data/yellow_tripdata_2017-11.csv";
                //JavaRDD<String> records = sc.textFile(files);

                //args[0] take the input file
                JavaRDD<String> records = sc.textFile(args[0]);
                JavaPairRDD<String, Double> userRdd = records.mapToPair(
                                data -> {
                                        List<String> dataList = new ArrayList<String>();
                                        dataList = Arrays.asList(data.split(","));
                                        //for(dataList.size() == 17);
                                        return data.trim().isEmpty() ?  new Tuple2<String,
Double>("Blank", 0.0) : new Tuple2<String, Double>(dataList.get(9), 1.0);
                                        //return Tuple2<String, Double>(dataList.get(9), 1.0);
                                });


                JavaPairRDD<String, Double> userCntRdd = userRdd.reduceByKey((x,y) -> x+y);

                // Sorting elements based on count

                List<Tuple2<String, Double>> userCntList = userCntRdd.top(10, new TupleSorter());
                //List<Tuple2<String, Double>> userCntListForStoring= null;// =new
List<Tuple2<String, Double>>();// = userCntList
                int counter = 0;

                for (Tuple2<String, Double> t : userCntList) {
                        if(counter>=2 )
```

```
            {
                System.out.println("Payment Type: "+t._1+" Cnt: "+t._2);

            }
            counter++;
            userCntRdd.saveAsTextFile(args[1]);
        }

        //Consider we need 1 output files
        //JavaPairRDD<String, Double> newData = userCntRdd.coalesce(1);

    }

}
```

Tuple Sorter Class

```
package sparkPayment1;


import java.io.Serializable;
import java.util.Comparator;

import scala.Tuple2;

public class TupleSorter implements Comparator<Tuple2<String, Double>>, Serializable {

    private static final long serialVersionUID = 1L;

    public int compare(Tuple2<String, Double> arg0, Tuple2<String, Double> arg1) {
        // TODO Auto-generated method stub
        if (arg1._2 > arg0._2)
            return 1;
        else if (arg1._2 < arg0._2)
            return -1;
        else
            return 0;
    }

}
```