Hadoop rmr2 Lab

Step 0: Transfer the data set files to hdfs directory /user/share/student. The files are:

testData.csv	- data files with header row and 12 data rows in csv format
testDataNoHdr.csv	- data file with no header row and 12 data rows in csv format
test_25K.csv	- data file with no header row and ~25K data rows in csv format
test_203694.csv	- data file with no header row and ~203694 data rows in csv format

Download these files from: https://cse.sc.edu/~rose/587/CSV/

Step 1: Review the map and reduce functions we discussed in class (**Oct 27: rmr2 Airline counting example**) to count how many flights per unique carrier (airline) a data set contains. This was basically the word counting problem that you've already seen.

Also be aware that your map function must always return a key-value pair unless you want rmr2 to choke. As far as I can tell, rmr2 will choke if you throw away rows. (The data set we use does not contain a header row).

Hints:

1) Dealing with missing values in rmr2: In R, missing values can be excluded using the keyword "na.rm=TRUE". In rmr2, this can be used in the reduce function: myreduce = function(carrier, counts) {keyval(carrier, sum(counts,na.rm=TRUE))}
2) In the case of reading csv files, the input to the map function is a list. Be sure to use double square brackets "[[]]" to access a list member directly (using a single bracket "[]" returns the slice containing the member) for details see: http://www.r-tutor.com/r-introduction/list

Problem 1: create a set of map, reduce, mapreduce functions to process the data in test_203694.csv. In the case of this problem you are to determine how many cancelled flights there are for each unique carrier (airline). Look at the map function we saw in class for counting flights for hints. The goal of this map-reduce problem is to determine the number of cancelled flights for each unique carrier in the data set. To figure out which columns correspond to unique carrier (**UniqueCarrier**) and cancellation (**Cancelled**), look at testData.csv which has header information.

Start by testing your code on the small data set testDataNoHdr.csv. When your code works with testData.csv, try your code with test_25K.csv. Finally, when everything works run your mapreduce code on the large dataset 2004.csv.

Assuming that you used the line `out = mr(hdfs.data, hdfs.out)`to invoke you map reduce job, output your results using:

```
results = from.dfs(out)
results.df = as.data.frame(results, stringsAsFactors=F)
colnames(results.df) = c('Carrier', 'cancellations')
results.df
```

Problem 2: create another set of map, reduce, mapreduce functions (**with different names**) to process the data in test_203694.csv. In the case of this problem you should find the mean departure delay by carrier. To figure out which columns correspond to unique carrier (**UniqueCarrier**) and departure delay (**DepDelay**), look at testData.csv which has header information. Define the column names for output using:

```
colnames(results.df) = c('Carrier', 'Mean Departure Delay')
```