Week 4 Assessment Task

Part A - MRJob with text (6 marks)

Detecting popular and trending topics from the news articles is an important task for public opinion monitoring. In Part A, your task is to perform text data analysis over a dataset of Australian news from ABC (Australian Broadcasting Corporation) using MRJob.

The dataset you will use contains data from news headlines published over several years. In this text file, each line is a **headline of a news article**, in the format of "*date,* term1 term2 ... ... ". The lines are sorted by the date, and the terms are separated by the space character. A sample file is like the below:

```
20191124,woman stabbed adelaide shopping centre
20191204,economy continue teetering edge recession
20200401,coronanomics learnt coronavirus economy
20200401,coronavirus home test kits selling chinese community
20201015,coronavirus pacific economy foriegn aid china
20201016,china builds pig apartment blocks guard swine flu
20211216,economy starts bounce unemployment
20211224,online shopping rise due coronavirus
20211229,china close encounters elon musks
```

When you click the panel on the right you'll get a connection to a server that has, in your home directory, a text file called "abcnews.txt", containing some sample text (feel free to open the file and explore its contents). The entire dataset can be downloaded from https://www.kaggle.com/therohk/million-headlines.

Your task is to **compute for each term, in which year it appears the most.** That is, for each term, you count how many articles contain this word in each year, and then select the year that has the most articles with this term (note that if an article contains a term multiple times, it only contributes 1 to the frequency). If the term appears in several years with the same frequency, select the earliest year as the result.

In your output, each line contains a key-value pair, where the key is the term, and the value is a pair of the year and this term's frequency in this year. For example, given the above data set, the output should be (there is no need to remove the quotation marks):

```
"adelaide"    "2019:1"
"aid"    "2020:1"
"apartment"    "2020:1"
"blocks"    "2020:1"
"bounce"    "2021:1"
"builds"    "2020:1"
```

```
"centre"    "2019:1"
"china"    "2020:2"
"chinese"    "2020:1"
"close"    "2021:1"
"community"    "2020:1"
"continue"    "2019:1"
"coronanomics"    "2020:1"
"coronavirus"    "2020:3"
"due"    "2021:1"
"economy"    "2019:1"
"edge"    "2019:1"
"elon"    "2021:1"
"encounters"    "2021:1"
"flu"    "2020:1"
"foriegn"    "2020:1"
"guard"    "2020:1"
"home"    "2020:1"
"kits"    "2020:1"
"learnt"    "2020:1"
"musks"    "2021:1"
"online"    "2021:1"
"pacific"    "2020:1"
"pig"    "2020:1"
"recession"    "2019:1"
"rise"    "2021:1"
"selling"    "2020:1"
"shopping"    "2019:1"
"stabbed"    "2019:1"
"starts"    "2021:1"
"swine"    "2020:1"
"teetering"    "2019:1"
"test"    "2020:1"
"unemployment"    "2021:1"
"woman"    "2019:1"
```

Write an MRJob job to do this. A file called "job.py" has been created for you - you just need to fill in the details. You can test your job locally by running the following command (it tells Python to execute job.py, using abcnews.txt as input, but the results may not be sorted by years):

```
$ python job.py abcnews.txt
```

To run your code on Hadoop MapReduce, you can use the following command (the results would be sorted as you can see in "output"):

```
$ python job.py abcnews.txt -r hadoop > output
```

Part B - MRJob with CSV (4 marks)

In Part B your task is to answer a question about the data in a CSV file, first using MRJob, and then using Hive. By using both to answer the same question about the same file you can more readily see how the two techniques compare.

When you click the panel on the right you'll get a connection to a server that has, in your home directory, a CSV file called "orders.csv", containing data about book orders (feel free to open the file and explore its contents).

Here are the fields in the file:

```
OrderDate (date)
ISBN (string)
Title (string)
Category (string)
PriceEach (decimal(5,2))
Quantity (integer)
FirstName (string)
LastName (string)
City (string)
```

Your task is to compute the **average cost of books per customer**, i.e., the total spent for books of a customer divided by the number of books purchased by the customer.

The result should be rounded to two decimal places, with **round(x,2)**, as shown below (MRJob output):

```
"BECCA NELSON"    34.18
"BONITA MORALES"    36.55
"CINDY GIRARD"    20.63
"GREG MONTIASA"    30.98
"JAKE LUCAS"    70.62
"JASMINE LEE"    55.95
"JENNIFER SMITH"    55.95
"KENNETH FALAH"    66.62
"KENNETH JONES"    19.95
"LEILA SMITH"    72.22
"REESE MCGOVERN"    55.95
"STEVE SCHELL"    8.95
"TAMMY GIANA"    48.91
"THOMAS PIERSON"    19.95
```

Write an MRJob job to do this. A file called "job.py" has been created for you - you just need to fill in the details. **Note that you are required to implement a combiner to do this task.**

You can test your job locally by running the following command (it tells Python to execute job.py

locally, using orders.csv as the input):

```
$ python job.py orders.csv
```

To run your code on Hadoop, you can use the following command (the results would be sorted by keys as you can see in "output"):

```
$ python job.py orders.csv -r hadoop > output
```