MAP REDUCE ASSIGNMENT

Task 4:

We have used 2 methods for executing Map Reduce

1. **Commandline (question a,b,c)**
2. **Using hadoop streaming (questions d,e,f)**

Qa. Which vendors have the most trips, and what is the total revenue generated by that vendor?

- Considering the part 1 file , vendor 2 (VeriFone Inc) had most trips with 525037658.14 as total revenue
- Command:
    - cat yellow_tripdata_2017-01.csv yellow_tripdata_2017-02.csv yellow_tripdata_2017-03.csv yellow_tripdata_2017-04.csv yellow_tripdata_2017-05.csv yellow_tripdata_2017-06.csv | python mappervendor.py | python reducervendor.py > output_vendor

```
[root@ip-172-31-72-60 script]# cat output_vendor
1       26824089        430567016.43
2       32158202        525037658.14
```

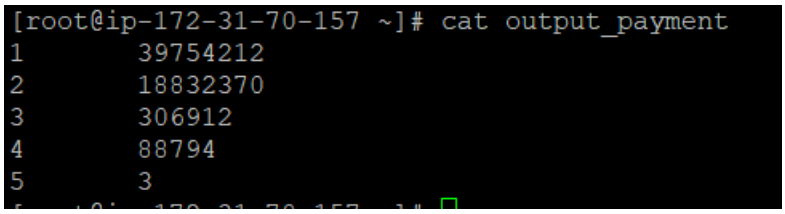
Qb. Which pickup location generates the most revenue?

- pickup location ID 132 has produced most revenue.
- Command:
    - cat yellow_tripdata_2017-01.csv yellow_tripdata_2017-02.csv yellow_tripdata_2017-03.csv yellow_tripdata_2017-04.csv yellow_tripdata_2017-05.csv yellow_tripdata_2017-06.csv | python mapperlocation.py | python reducerlocation.py > output_location

```
[root@ip-172-31-70-157 ~]# cat output_location
132     1378523 77196812.24
[root@ip-172-31-70-157 ~]#
```

Qc. What are the different payment types used by customers and their count? The final results should be in a sorted format.

- Most of the customer prefer payment type 1 i.e., credit card payment



Qd. What is the average trip time for different pickup locations?

- We have used hadoop streamin for executing.

  hadoop jar /lib/hadoop-mapreduce/hadoop-streaming-2.8.5-amzn-6.jar \

  -file mapper_d.py -mapper 'python mapper_d.py' \

  -file reducer_d.py -reducer 'python reducer_d.py' \

  -input /user/hadoop/input \

  -output /user/hadoop/output_d \

  -file combiner_d.py -combiner 'python combiner_d.py'

  Output screenshot:

```
215     48.06
212     20.64
218     20.21
227     14.07
131     13.66
137     13.29
134     15.36
26      11.57
224     13.22
20      13.66
23      12.34
29      21.45
8       17.01
```

  Complete Result is in output_d folder

Qe. Calculate the average tips to revenue ratio of the drivers for different locations in sorted format.

```
hadoop jar /lib/hadoop-mapreduce/hadoop-streaming-2.8.5-amzn-6.jar \
-file mapper_e.py -mapper 'python mapper_e.py' \
-file reducer_e.py -reducer 'python reducer_e.py' \
-input /user/hadoop/input \
-output /user/hadoop/output_e \
-file combiner_e.py -combiner 'python combiner_e.py'
```

Sample output:

```
2       5.47    51.82
5       12.20   70.30
8       3.63    33.44
11      2.38    35.95
14      2.02    22.36
17      1.15    13.81
20      1.43    19.01
23      6.70    56.37
26      1.14    17.41
29      2.05    31.13
32      1.25    20.42
35      2.03    20.21
38      2.48    42.75
41      1.17    12.95
44      3.66    57.10
47      0.54    15.13
50      1.57    14.40
53      1.80    23.28
```

Complete output is in output_e folder

Qe. How does revenue vary over time? Calculate the average trip revenue per month - analysing it by hour of the day (day vs night) and the day of the week (weekday vs weekend).

- From the pick up date and time, we have extracted month, day and night, day of the week using below scripts.

```
pickup = datetime.strptime(line[1],"%Y-%m-%d %H:%M:%S")
 pickup_date =pickup.date()
 month = pickup_date.strftime('%B')
 day = calendar.day_name[pickup_date.weekday()]
 hr = pickup.time().hour
 daytime = ( "day" if hr<=20 and hr>=6 else "night")
```

- below is the hadoop streaming command

  hadoop jar /lib/hadoop-mapreduce/hadoop-streaming-2.8.5-amzn-6.jar -files mappermonth.py,reducermonth.py -mapper 'python mappermonth.py' -reducer 'python reducermonth.py' -input /user/hadoop/input -output /user/hadoop/output_month3 -numReduceTasks 3 -partitioner org.apache.hadoop.mapred.lib.HashPartitioner

  Added index value 1 for month 2 for day of the week and 3 for day and night

  Sample output:

```
1March  16.22
2Monday 16.43
2Tuesday        15.90
2Friday 16.52
2Wednesday      16.57
2Thursday       17.00
2Sunday 15.97
2Saturday       14.87
3day    16.00
3night  16.84
```