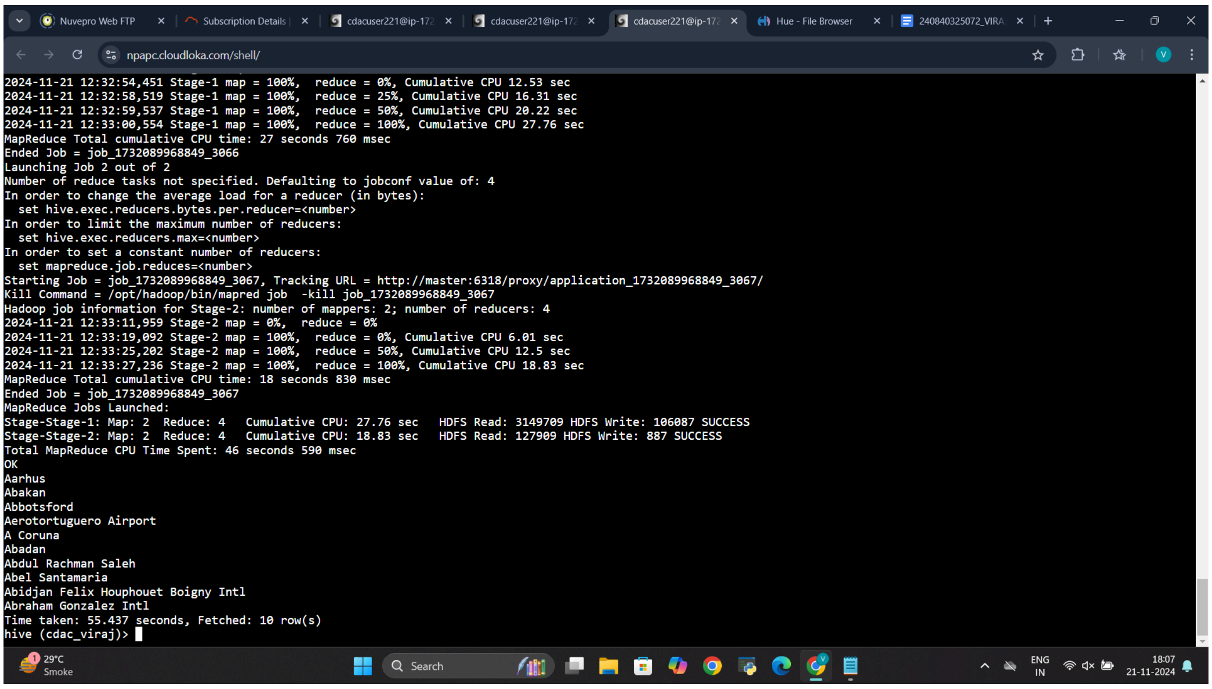
BIG DATA MODULE END PAPER

ROLL NO.240840325072
NAME:VIRAJ SHIRKE

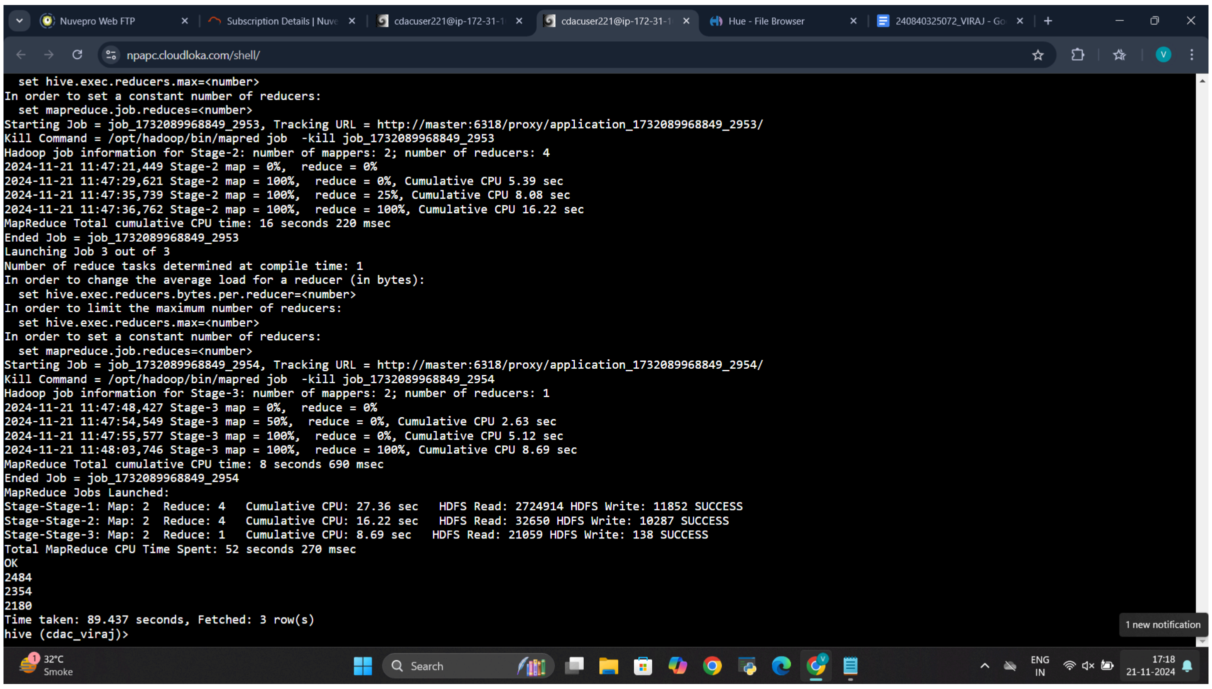
HIVE:
QUESTION 1:

1.
select distinct a.name from airport a join routes r on a.iata = r.src_airport_iata limit 10;



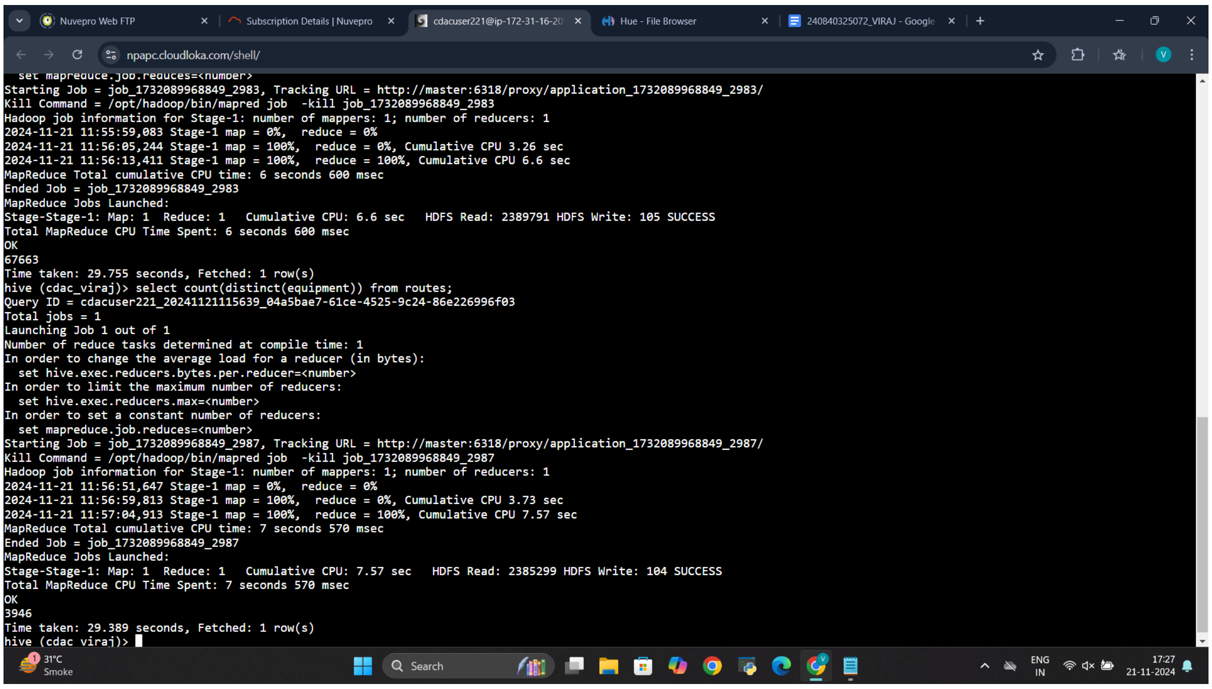
```
2024-11-21 12:32:54,451 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 12.53 sec
2024-11-21 12:32:58,519 Stage-1 map = 100%,  reduce = 25%, Cumulative CPU 16.31 sec
2024-11-21 12:32:59,537 Stage-1 map = 100%,  reduce = 50%, Cumulative CPU 20.22 sec
2024-11-21 12:33:00,554 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 27.76 sec
MapReduce Total cumulative CPU time: 27 seconds 760 msec
Ended Job = job_1732089968849_3066
Launching Job 2 out of 2
Number of reduce tasks not specified. Defaulting to jobconf value of: 4
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1732089968849_3067, Tracking URL = http://master:6318/proxy/application_1732089968849_3067/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_3067
Hadoop job information for Stage-2: number of mappers: 2; number of reducers: 4
2024-11-21 12:33:11,959 Stage-2 map = 0%,  reduce = 0%
2024-11-21 12:33:19,092 Stage-2 map = 100%,  reduce = 0%, Cumulative CPU 6.01 sec
2024-11-21 12:33:25,202 Stage-2 map = 100%,  reduce = 50%, Cumulative CPU 12.5 sec
2024-11-21 12:33:27,236 Stage-2 map = 100%,  reduce = 100%, Cumulative CPU 18.83 sec
MapReduce Total cumulative CPU time: 18 seconds 830 msec
Ended Job = job_1732089968849_3067
MapReduce Jobs Launched:
Stage-Stage-1: Map: 2  Reduce: 4   Cumulative CPU: 27.76 sec   HDFS Read: 3149709 HDFS Write: 106087 SUCCESS
Stage-Stage-2: Map: 2  Reduce: 4   Cumulative CPU: 18.83 sec   HDFS Read: 127909 HDFS Write: 887 SUCCESS
Total MapReduce CPU Time Spent: 46 seconds 590 msec
OK
Aarhus
Abakan
Abbotsford
Aerotortuguero Airport
A Coruna
Abadan
Abdul Rachman Saleh
Abel Santamaria
Abidjan Felix Houphouet Boigny Intl
Abraham Gonzalez Intl
Time taken: 55.437 seconds, Fetched: 10 row(s)
hive (cdac_viraj)>
```

2.select count(*) from airlines a join routes r on r.airline_id=a.airline_id group by a.airline_id order by count(*) desc limit 3;



```
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1732089968849_2953, Tracking URL = http://master:6318/proxy/application_1732089968849_2953/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2953
Hadoop job information for Stage-2: number of mappers: 2; number of reducers: 4
2024-11-21 11:47:21,449 Stage-2 map = 0%,  reduce = 0%
2024-11-21 11:47:29,621 Stage-2 map = 100%,  reduce = 0%, Cumulative CPU 5.39 sec
2024-11-21 11:47:35,739 Stage-2 map = 100%,  reduce = 25%, Cumulative CPU 8.08 sec
2024-11-21 11:47:36,762 Stage-2 map = 100%,  reduce = 100%, Cumulative CPU 16.22 sec
MapReduce Total cumulative CPU time: 16 seconds 220 msec
Ended Job = job_1732089968849_2953
Launching Job 3 out of 3
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1732089968849_2954, Tracking URL = http://master:6318/proxy/application_1732089968849_2954/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2954
Hadoop job information for Stage-3: number of mappers: 2; number of reducers: 1
2024-11-21 11:47:48,427 Stage-3 map = 0%,  reduce = 0%
2024-11-21 11:47:54,549 Stage-3 map = 50%,  reduce = 0%, Cumulative CPU 2.63 sec
2024-11-21 11:47:55,577 Stage-3 map = 100%,  reduce = 0%, Cumulative CPU 5.12 sec
2024-11-21 11:48:03,746 Stage-3 map = 100%,  reduce = 100%, Cumulative CPU 8.69 sec
MapReduce Total cumulative CPU time: 8 seconds 690 msec
Ended Job = job_1732089968849_2954
MapReduce Jobs Launched:
Stage-Stage-1: Map: 2  Reduce: 4   Cumulative CPU: 27.36 sec   HDFS Read: 2724914 HDFS Write: 11852 SUCCESS
Stage-Stage-2: Map: 2  Reduce: 4   Cumulative CPU: 16.22 sec   HDFS Read: 32650 HDFS Write: 10287 SUCCESS
Stage-Stage-3: Map: 2  Reduce: 1   Cumulative CPU: 8.69 sec   HDFS Read: 21059 HDFS Write: 138 SUCCESS
Total MapReduce CPU Time Spent: 52 seconds 270 msec
OK
2484
2354
2180
Time taken: 89.437 seconds, Fetched: 3 row(s)
hive (cdac_viraj)>
```

3.select count(distinct(equipment)) from routes;

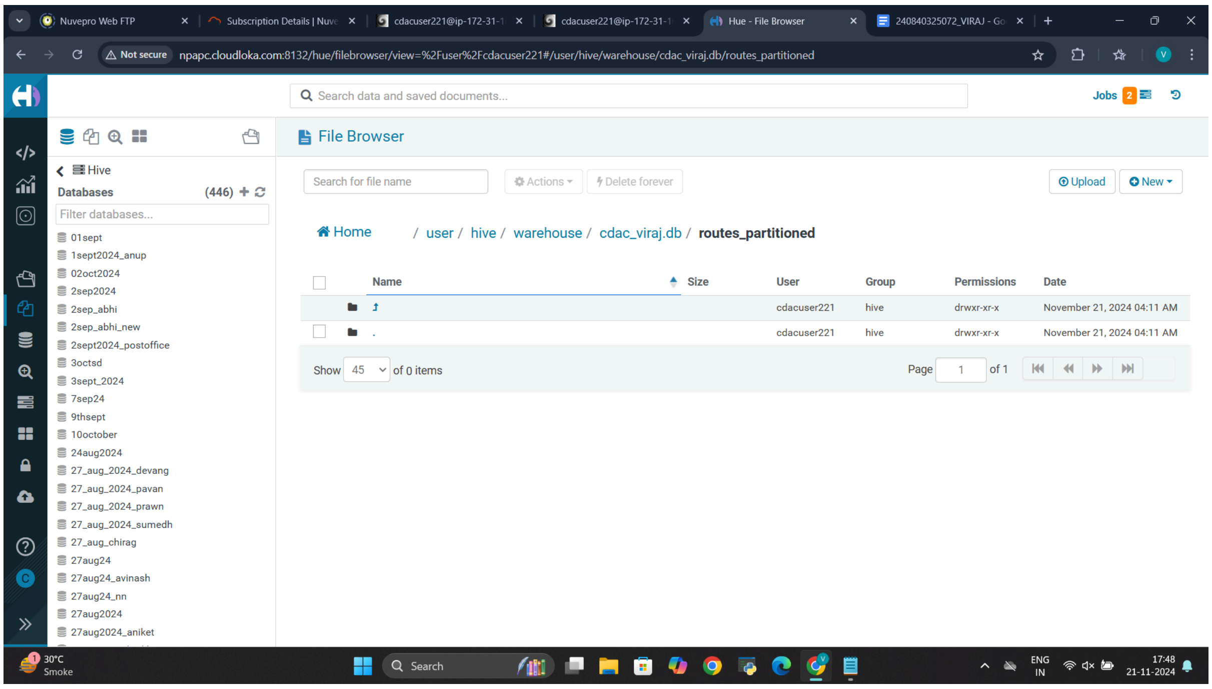


```
  set mapreduce.job.reduces=<number>
Starting Job = job_1732089968849_2983, Tracking URL = http://master:6318/proxy/application_1732089968849_2983/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2983
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-11-21 11:55:59,083 Stage-1 map = 0%,  reduce = 0%
2024-11-21 11:56:05,244 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 3.26 sec
2024-11-21 11:56:13,411 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 6.6 sec
MapReduce Total cumulative CPU time: 6 seconds 600 msec
Ended Job = job_1732089968849_2983
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 6.6 sec   HDFS Read: 2389791 HDFS Write: 105 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 600 msec
OK
67663
Time taken: 29.755 seconds, Fetched: 1 row(s)
hive (cdac_viraj)> select count(distinct(equipment)) from routes;
Query ID = cdacuser221_20241121115639_04a5bae7-61ce-4525-9c24-86e226996f03
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<number>
In order to limit the maximum number of reducers:
  set hive.exec.reducers.max=<number>
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1732089968849_2987, Tracking URL = http://master:6318/proxy/application_1732089968849_2987/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2987
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-11-21 11:56:51,647 Stage-1 map = 0%,  reduce = 0%
2024-11-21 11:56:59,813 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 3.73 sec
2024-11-21 11:57:04,913 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 7.57 sec
MapReduce Total cumulative CPU time: 7 seconds 570 msec
Ended Job = job_1732089968849_2987
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 7.57 sec   HDFS Read: 2385299 HDFS Write: 104 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 570 msec
OK
3946
Time taken: 29.389 seconds, Fetched: 1 row(s)
hive (cdac_viraj)>
```

QUESTION 2:

1.create table routes_partitioned(airline_iata string ,airline_id int ,src_airport_iata string,src_airport_id int , dest_airport_id int , codeshare st
ring , stops int , equipment string)
partitioned by (dest_airport_iata)
row format delimited
fields terminated by ','

stored as textfile;



npapc.cloudloka.com:8132/hue/filebrowser/view=%2Fuser%2Fcdacuser221#/user/hive/warehouse/cdac_viraj.db/routes_partitioned

File Browser

Home / user / hive / warehouse / cdac_viraj.db / routes_partitioned

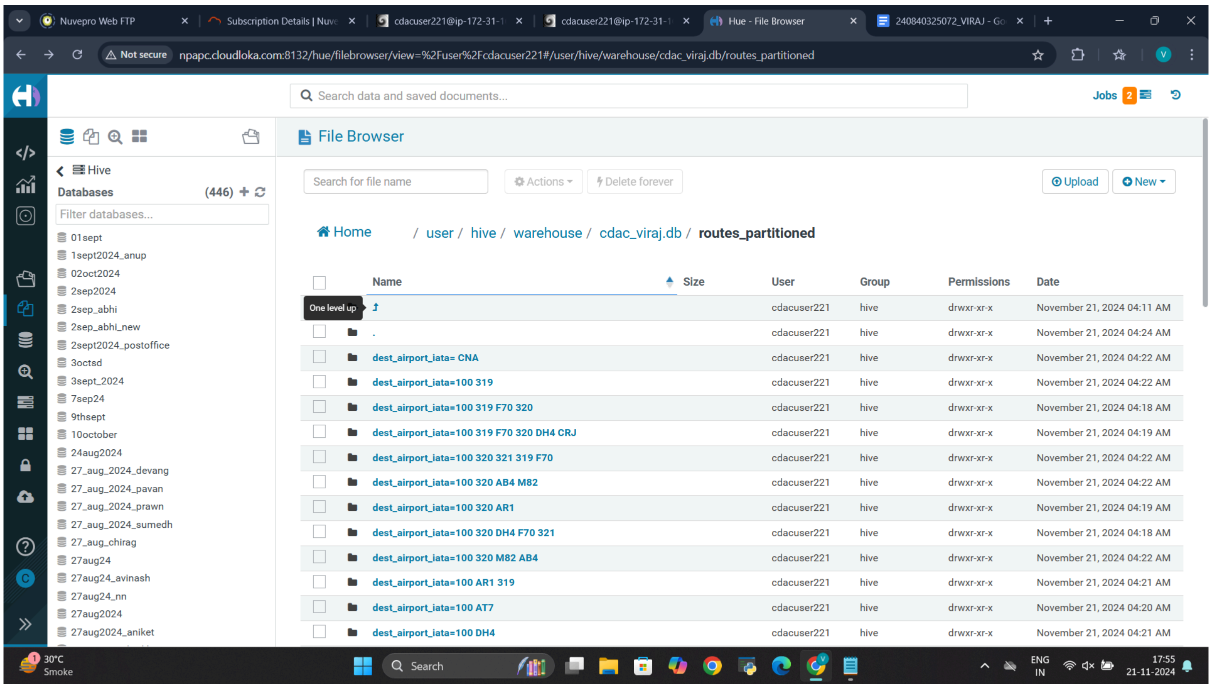
Name	Size	User	Group	Permissions	Date
↥		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:11 AM
.		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:11 AM

Show 45 of 0 items   Page 1 of 1

Hive Databases (446): 01sept, 1sept2024_anup, 02oct2024, 2sep2024, 2sep_abhi, 2sep_abhi_new, 2sept2024_postoffice, 3octsd, 3sept_2024, 7sep24, 9thsept, 10october, 24aug2024, 27_aug_2024_devang, 27_aug_2024_pavan, 27_aug_2024_prawn, 27_aug_2024_sumedh, 27_aug_chirag, 27aug24, 27aug24_avinash, 27aug24_nn, 27aug2024, 27aug2024_aniket

2.

Insert overwrite table routes_partitioned PARTITION(dest_airport_iata) select r.airline_iata ,r.airline_id ,r.src_airport_iata ,r.src_airport_id ,r.dest_airport_iata, r.dest_airport_id , r.codeshare ,r.stops,r.equipment from routes r DISTRIBUTE By dest_airport_iata;

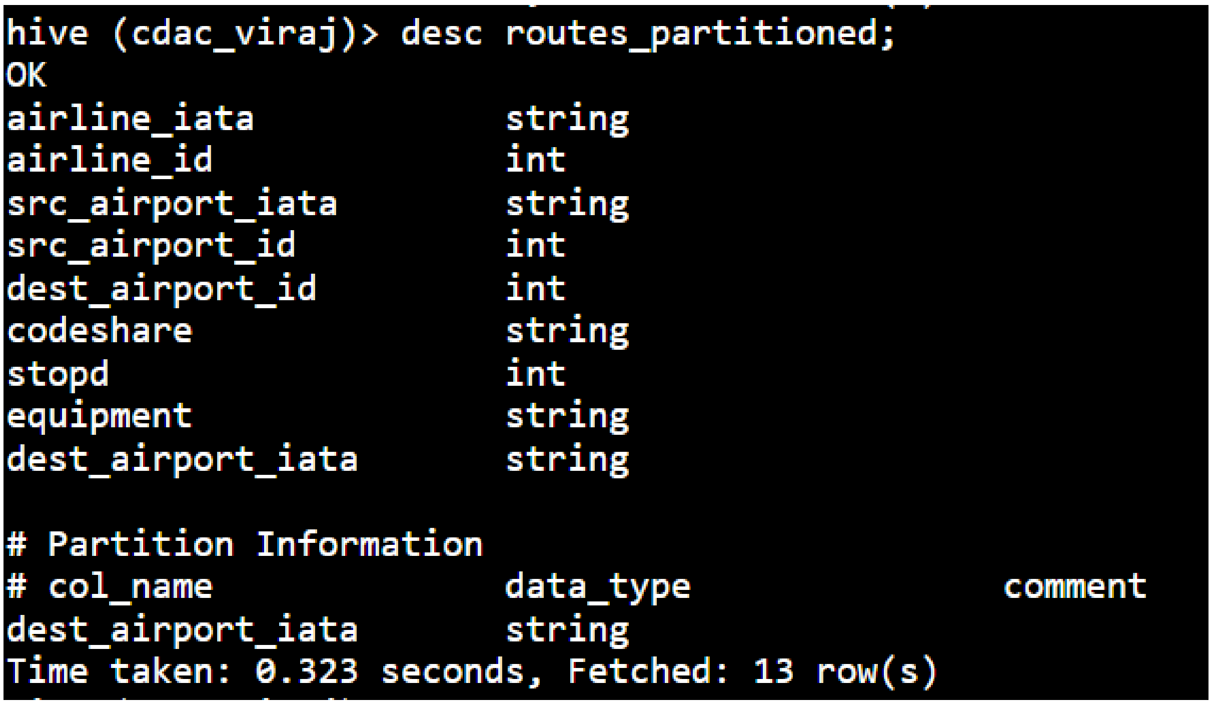


npapc.cloudloka.com:8132/hue/filebrowser/view=%2Fuser%2Fcdacuser221#/user/hive/warehouse/cdac_viraj.db/routes_partitioned

File Browser

Home / user / hive / warehouse / cdac_viraj.db / routes_partitioned

Name	Size	User	Group	Permissions	Date
↥		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:11 AM
.		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:24 AM
dest_airport_iata= CNA		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:22 AM
dest_airport_iata=100 319		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:22 AM
dest_airport_iata=100 319 F70 320		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:18 AM
dest_airport_iata=100 319 F70 320 DH4 CRJ		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:19 AM
dest_airport_iata=100 320 321 319 F70		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:22 AM
dest_airport_iata=100 320 AB4 M82		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:22 AM
dest_airport_iata=100 320 AR1		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:19 AM
dest_airport_iata=100 320 DH4 F70 321		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:18 AM
dest_airport_iata=100 320 M82 AB4		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:22 AM
dest_airport_iata=100 AR1 319		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:21 AM
dest_airport_iata=100 AT7		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:20 AM
dest_airport_iata=100 DH4		cdacuser221	hive	drwxr-xr-x	November 21, 2024 04:21 AM



```
hive (cdac_viraj)> desc routes_partitioned;
OK
airline_iata            string
airline_id              int
src_airport_iata        string
src_airport_id          int
dest_airport_id         int
codeshare               string
stopd                   int
equipment               string
dest_airport_iata       string

# Partition Information
# col_name              data_type               comment
dest_airport_iata       string
Time taken: 0.323 seconds, Fetched: 13 row(s)
```

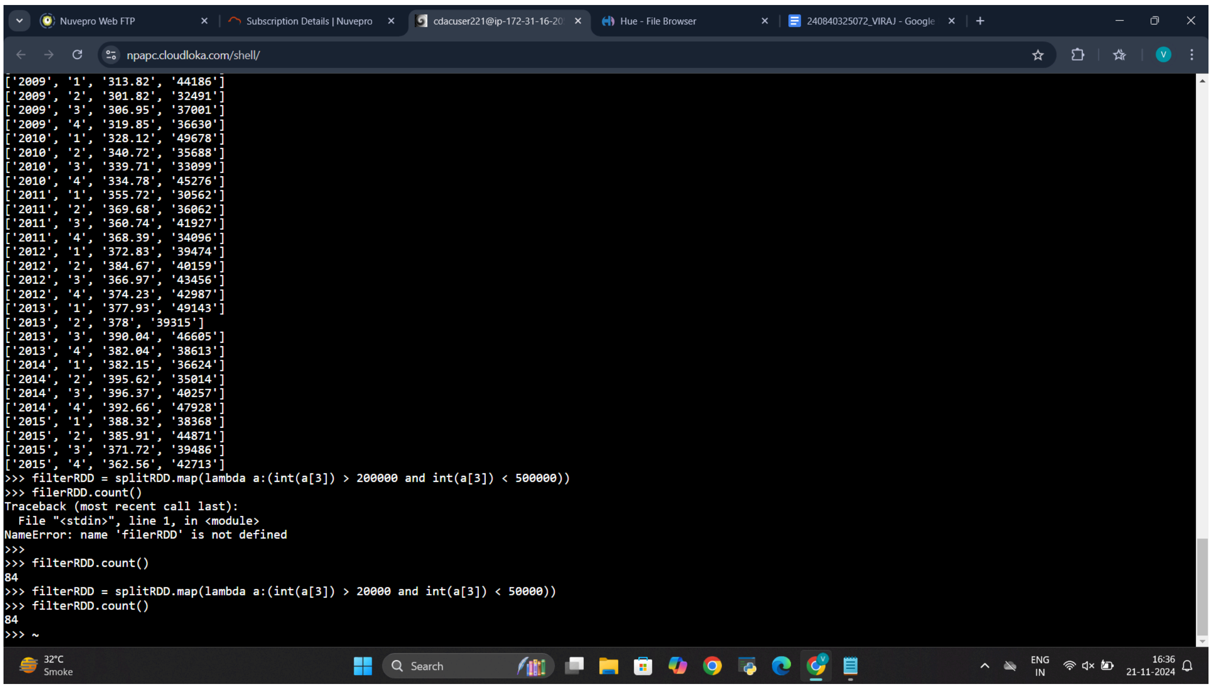
3.

PYSPARK :

QUESTION 1:

1.

filterRDD = splitRDD.map(lambda a:(int(a[3]) > 20000 and int(a[3]) < 50000)



```
['2009', '1', '313.82', '44186']
['2009', '2', '301.82', '32491']
['2009', '3', '306.95', '37001']
['2009', '4', '319.85', '36630']
['2010', '1', '328.12', '49678']
['2010', '2', '340.72', '35688']
['2010', '3', '339.71', '33099']
['2010', '4', '334.78', '45276']
['2011', '1', '355.72', '30562']
['2011', '2', '369.68', '36062']
['2011', '3', '360.74', '41927']
['2011', '4', '368.39', '34096']
['2012', '1', '372.83', '39474']
['2012', '2', '384.67', '40159']
['2012', '3', '366.97', '43456']
['2012', '4', '374.23', '42987']
['2013', '1', '377.93', '49143']
['2013', '2', '378', '39315']
['2013', '3', '390.04', '46605']
['2013', '4', '382.04', '38613']
['2014', '1', '382.15', '36624']
['2014', '2', '395.62', '35014']
['2014', '3', '396.37', '40257']
['2014', '4', '392.66', '47928']
['2015', '1', '388.32', '38368']
['2015', '2', '385.91', '44871']
['2015', '3', '371.72', '39486']
['2015', '4', '362.56', '42713']
>>> filterRDD = splitRDD.map(lambda a:(int(a[3]) > 200000 and int(a[3]) < 500000))
>>> filerRDD.count()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'filerRDD' is not defined
>>>
>>> filterRDD.count()
84
>>> filterRDD = splitRDD.map(lambda a:(int(a[3]) > 20000 and int(a[3]) < 50000))
>>> filterRDD.count()
84
>>> ~
```

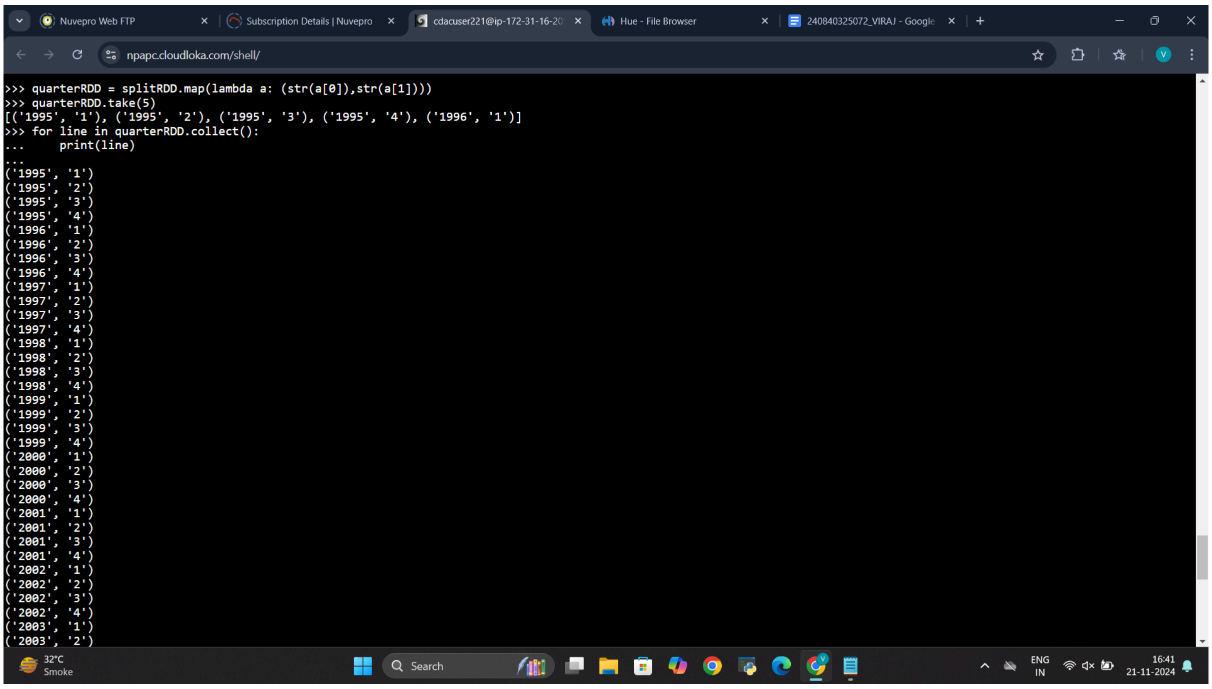
Nuvepro Web FTP
Subscription Details | Nuvepro
cdacuser221@ip-172-31-16-20
Hue - File Browser
240840325072_VIRAJ - Google
npapc.cloudloka.com/shell/
32°C
Smoke
Search
ENG
IN
16:36
21-11-2024

2.

quarterRDD = splitRDD.map(lambda a: (str[0],str[1]))

for line in quarterRDD .collect():

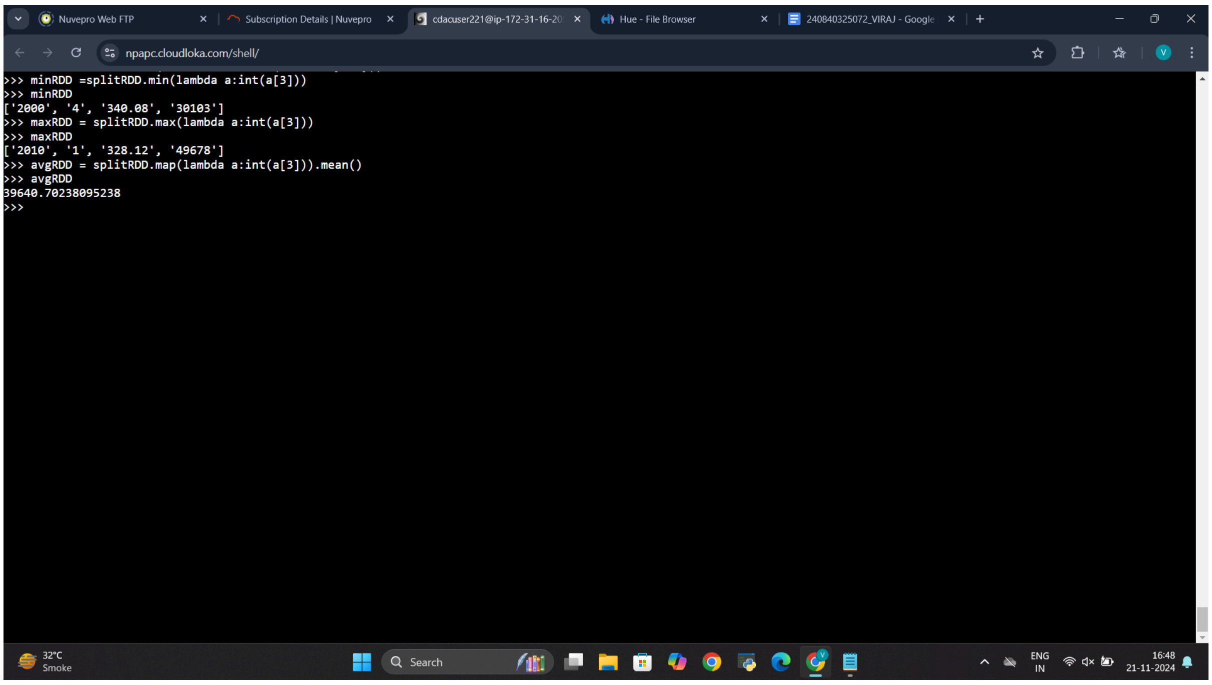
print(line)



```
>>> quarterRDD = splitRDD.map(lambda a: (str(a[0]),str(a[1])))
>>> quarterRDD.take(5)
[('1995', '1'), ('1995', '2'), ('1995', '3'), ('1995', '4'), ('1996', '1')]
>>> for line in quarterRDD.collect():
...     print(line)
...
('1995', '1')
('1995', '2')
('1995', '3')
('1995', '4')
('1996', '1')
('1996', '2')
('1996', '3')
('1996', '4')
('1997', '1')
('1997', '2')
('1997', '3')
('1997', '4')
('1998', '1')
('1998', '2')
('1998', '3')
('1998', '4')
('1999', '1')
('1999', '2')
('1999', '3')
('1999', '4')
('2000', '1')
('2000', '2')
('2000', '3')
('2000', '4')
('2001', '1')
('2001', '2')
('2001', '3')
('2001', '4')
('2002', '1')
('2002', '2')
('2002', '3')
('2002', '4')
('2003', '1')
('2003', '2')
```

QUESTION 2:

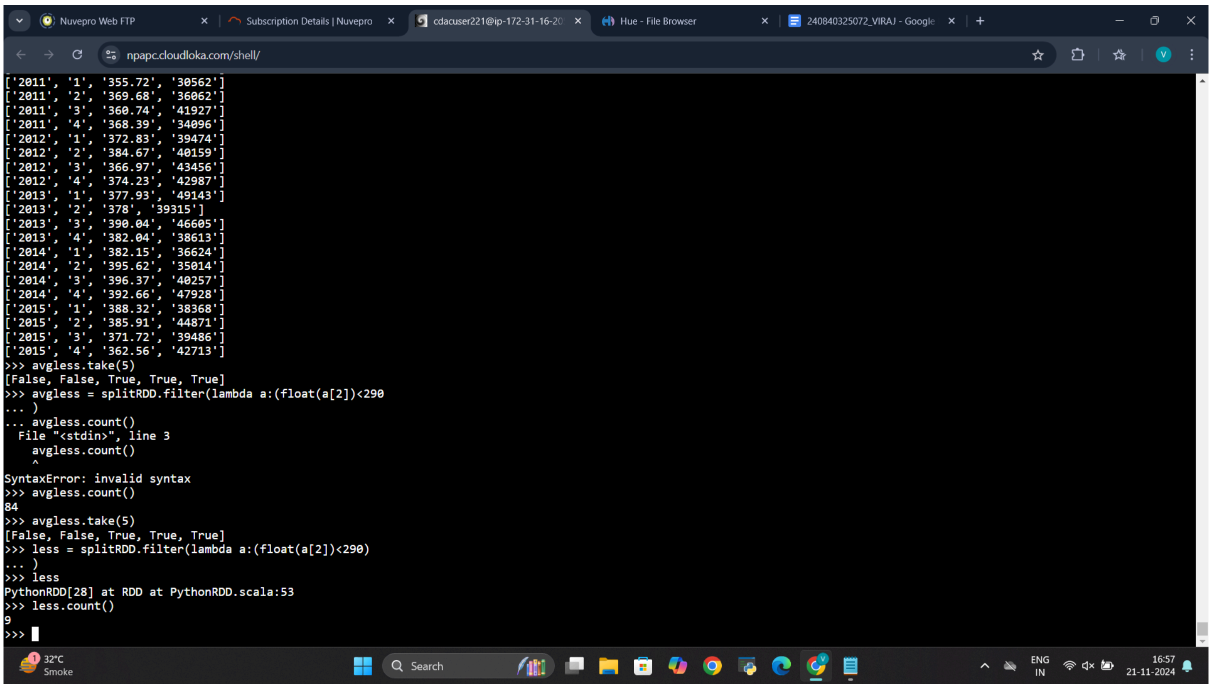
1.
Min,Max,Avg:
minRDD =splitRDD.min(lambda a:int(a[3]))
maxRDD = splitRDD.max(lambda a:int(a[3]))
avgRDD = splitRDD.max(lambda a:int(a[3])).mean()



```
>>> minRDD =splitRDD.min(lambda a:int(a[3]))
>>> minRDD
['2000', '4', '340.08', '30103']
>>> maxRDD = splitRDD.max(lambda a:int(a[3]))
>>> maxRDD
['2010', '1', '328.12', '49678']
>>> avgRDD = splitRDD.map(lambda a:int(a[3])).mean()
>>> avgRDD
39640.70238095238
>>>
```

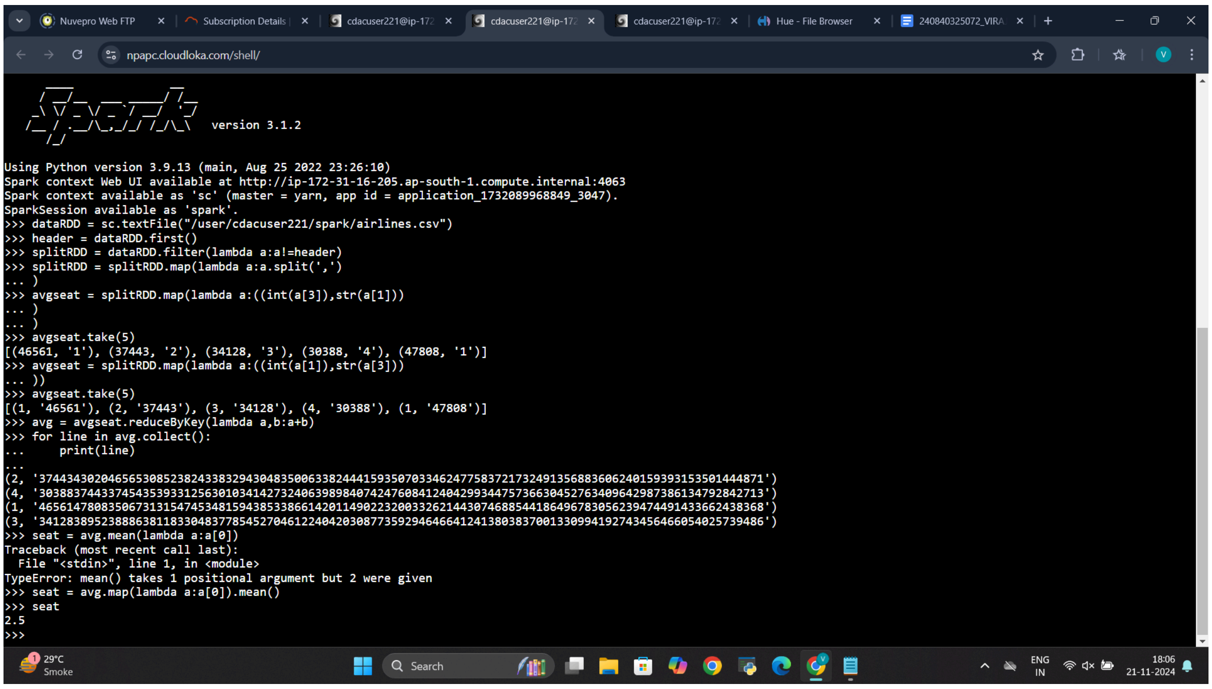
2.

less = splitRDD.filter(lambda a:(float(a[2])<290)



```
['2011', '1', '355.72', '30562']
['2011', '2', '369.68', '36062']
['2011', '3', '360.74', '41927']
['2011', '4', '368.39', '34096']
['2012', '1', '372.83', '39474']
['2012', '2', '384.67', '40159']
['2012', '3', '366.97', '43456']
['2012', '4', '374.23', '42987']
['2013', '1', '377.93', '49143']
['2013', '2', '378', '39315']
['2013', '3', '390.04', '46605']
['2013', '4', '382.04', '38613']
['2014', '1', '382.15', '36624']
['2014', '2', '395.62', '35014']
['2014', '3', '396.37', '40257']
['2014', '4', '392.66', '47928']
['2015', '1', '388.32', '38368']
['2015', '2', '385.91', '44871']
['2015', '3', '371.72', '39486']
['2015', '4', '362.56', '42713']
>>> avgless.take(5)
[False, False, True, True, True]
>>> avgless = splitRDD.filter(lambda a:(float(a[2])<290
... )
... avgless.count()
  File "<stdin>", line 3
    avgless.count()
    ^
SyntaxError: invalid syntax
>>> avgless.count()
84
>>> avgless.take(5)
[False, False, True, True, True]
>>> less = splitRDD.filter(lambda a:(float(a[2])<290)
... )
>>> less
PythonRDD[28] at RDD at PythonRDD.scala:53
>>> less.count()
9
>>> █
```

3.



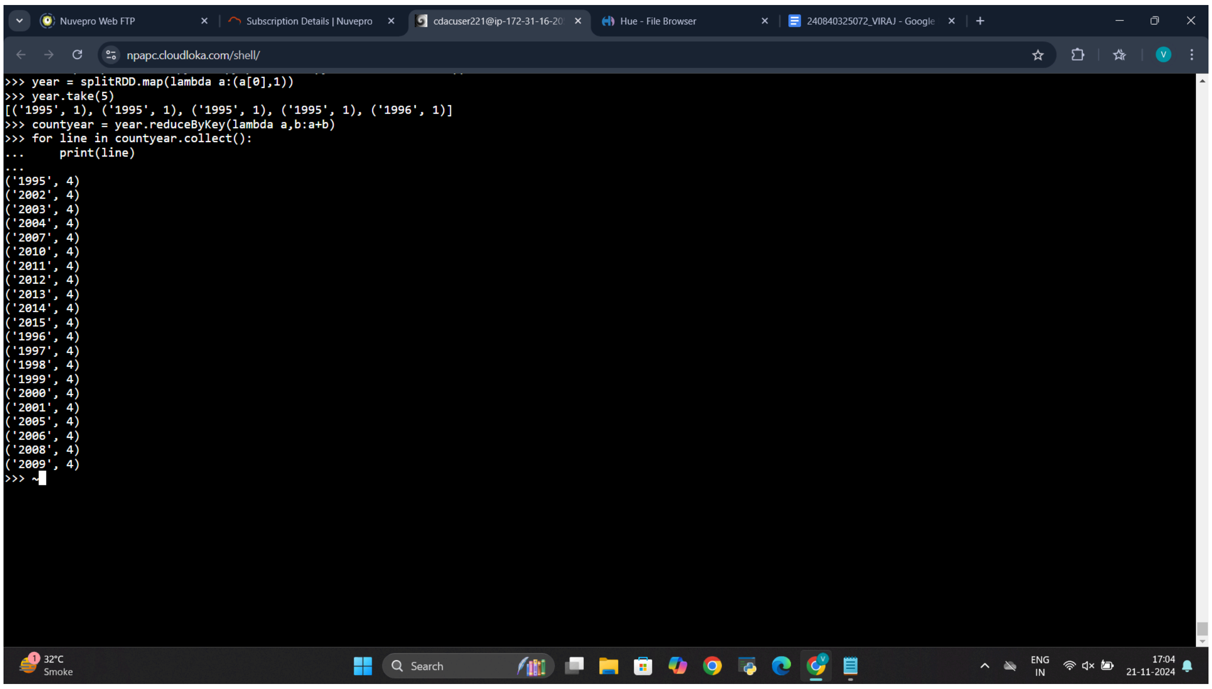
```
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 3.1.2
      /_/

Using Python version 3.9.13 (main, Aug 25 2022 23:26:10)
Spark context Web UI available at http://ip-172-31-16-205.ap-south-1.compute.internal:4063
Spark context available as 'sc' (master = yarn, app id = application_1732089968849_3047).
SparkSession available as 'spark'.
>>> dataRDD = sc.textFile("/user/cdacuser221/spark/airlines.csv")
>>> header = dataRDD.first()
>>> splitRDD = dataRDD.filter(lambda a:a!=header)
>>> splitRDD = splitRDD.map(lambda a:a.split(',')
... )
>>> avgseat = splitRDD.map(lambda a:((int(a[3]),str(a[1]))
... )
... )
>>> avgseat.take(5)
[(46561, '1'), (37443, '2'), (34128, '3'), (30388, '4'), (47808, '1')]
>>> avgseat = splitRDD.map(lambda a:((int(a[1]),str(a[3]))
... ))
>>> avgseat.take(5)
[(1, '46561'), (2, '37443'), (3, '34128'), (4, '30388'), (1, '47808')]
>>> avg = avgseat.reduceByKey(lambda a,b:a+b)
>>> for line in avg.collect():
...     print(line)
...
(2, '374434302046565308523824338329430483500633824441593507033462477583721732491356883606240159393153501444871')
(4, '303883744337454353933125630103414273240639898407424760841240429934475736630452763409642987386134792842713')
(1, '465614780835067313154745348159438533866142011490223200332621443074688544186496783056239474491433662438368')
(3, '341283895238886381183304837785452704612240420308773592946466412413803837001330994192743456466054025739486')
>>> seat = avg.mean(lambda a:a[0])
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: mean() takes 1 positional argument but 2 were given
>>> seat = avg.map(lambda a:a[0]).mean()
>>> seat
2.5
>>>
```

4.

year = splitRDD.map(lambda a:(a[0],1))

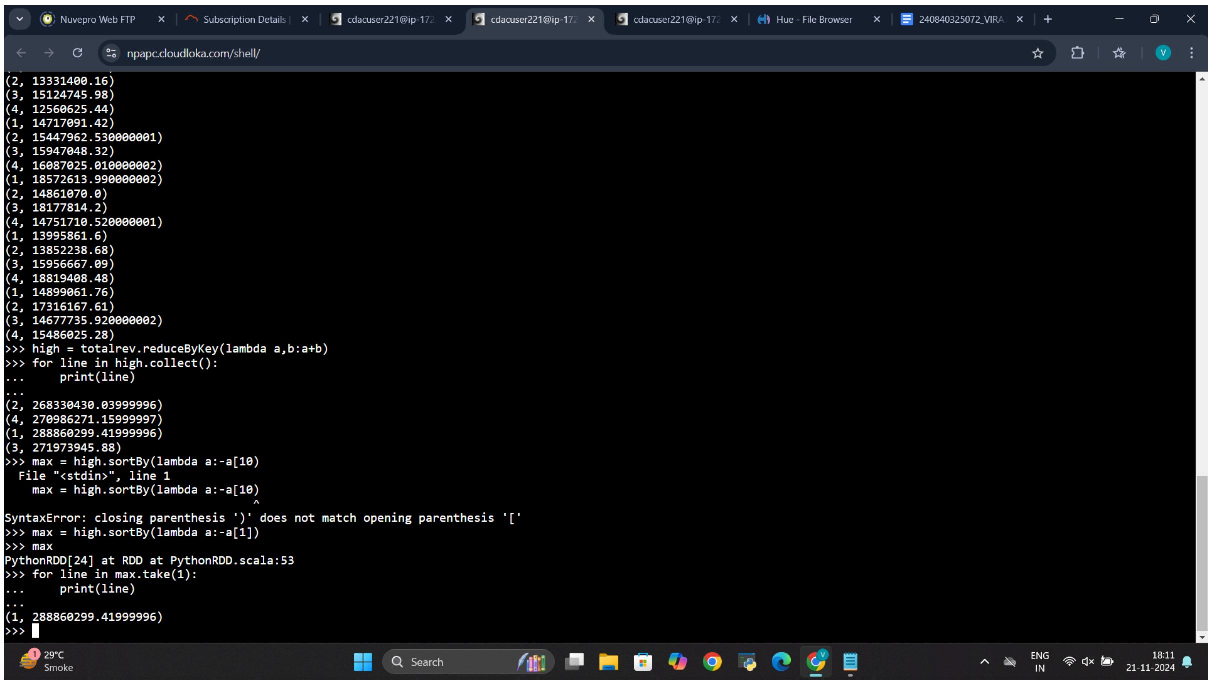
countyear = year.reduceByKey(lambda a,b:a+b)



```
>>> year = splitRDD.map(lambda a:(a[0],1))
>>> year.take(5)
[('1995', 1), ('1995', 1), ('1995', 1), ('1995', 1), ('1996', 1)]
>>> countyear = year.reduceByKey(lambda a,b:a+b)
>>> for line in countyear.collect():
...     print(line)
...
('1995', 4)
('2002', 4)
('2003', 4)
('2004', 4)
('2007', 4)
('2010', 4)
('2011', 4)
('2012', 4)
('2013', 4)
('2014', 4)
('2015', 4)
('1996', 4)
('1997', 4)
('1998', 4)
('1999', 4)
('2000', 4)
('2001', 4)
('2005', 4)
('2006', 4)
('2008', 4)
('2009', 4)
>>> ~
```

5.

totalrev = splitRDD.map(lambda a: ((int(a[1])), float(a[2])*int(a[3])))
high = totalrev.reduceByKey(lambda a,b:a+b)
max = high.sortBy(lambda a:-a[1])



```
(2, 13331400.16)
(3, 15124745.98)
(4, 12560625.44)
(1, 14717091.42)
(2, 15447962.530000001)
(3, 15947048.32)
(4, 16087025.010000002)
(1, 18572613.990000002)
(2, 14861070.0)
(3, 18177814.2)
(4, 14751710.520000001)
(1, 13995861.6)
(2, 13852238.68)
(3, 15956667.09)
(4, 18819408.48)
(1, 14899061.76)
(2, 17316167.61)
(3, 14677735.920000002)
(4, 15486025.28)
>>> high = totalrev.reduceByKey(lambda a,b:a+b)
>>> for line in high.collect():
...     print(line)
...
(2, 268330430.03999996)
(4, 270986271.15999997)
(1, 288860299.41999996)
(3, 271973945.88)
>>> max = high.sortBy(lambda a:-a[10)
  File "<stdin>", line 1
    max = high.sortBy(lambda a:-a[10)
                                    ^
SyntaxError: closing parenthesis ')' does not match opening parenthesis '['
>>> max = high.sortBy(lambda a:-a[1])
>>> max
PythonRDD[24] at RDD at PythonRDD.scala:53
>>> for line in max.take(1):
...     print(line)
...
(1, 288860299.41999996)
>>> 
```