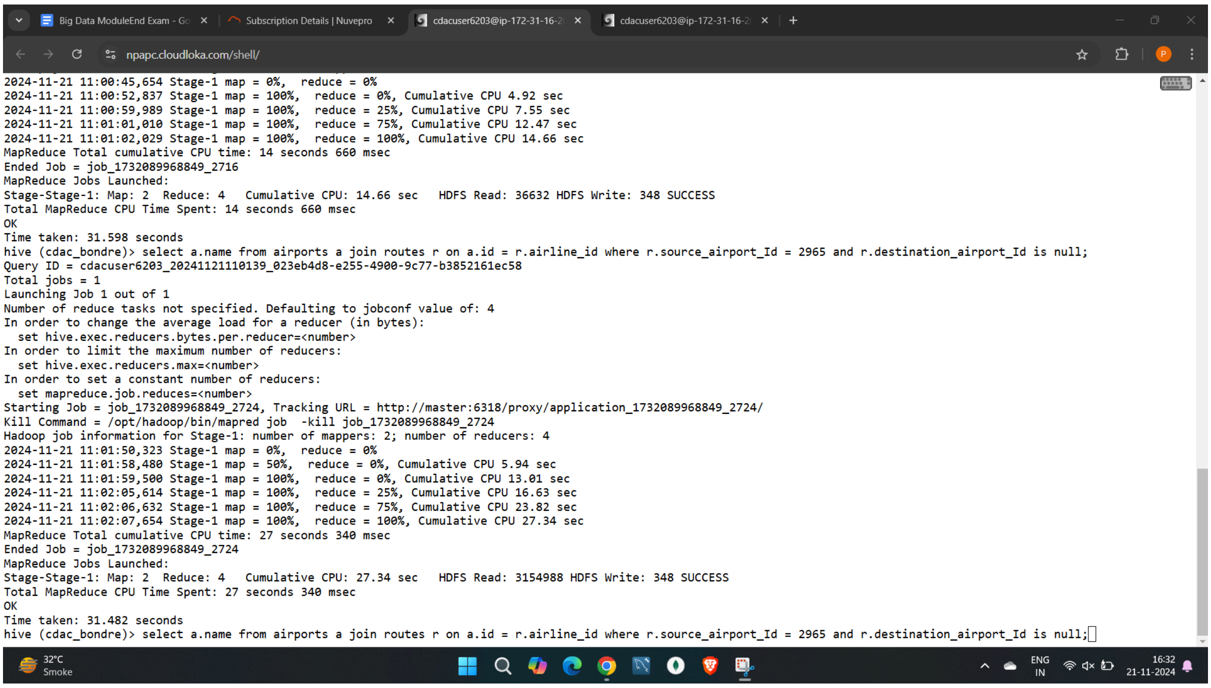
HIVE

Q1.

1.

→ select a.name from airports a join routes r on a.id = r.airline_id where r.source_airport_Id = 2965 and r.destination_airport_Id is null;

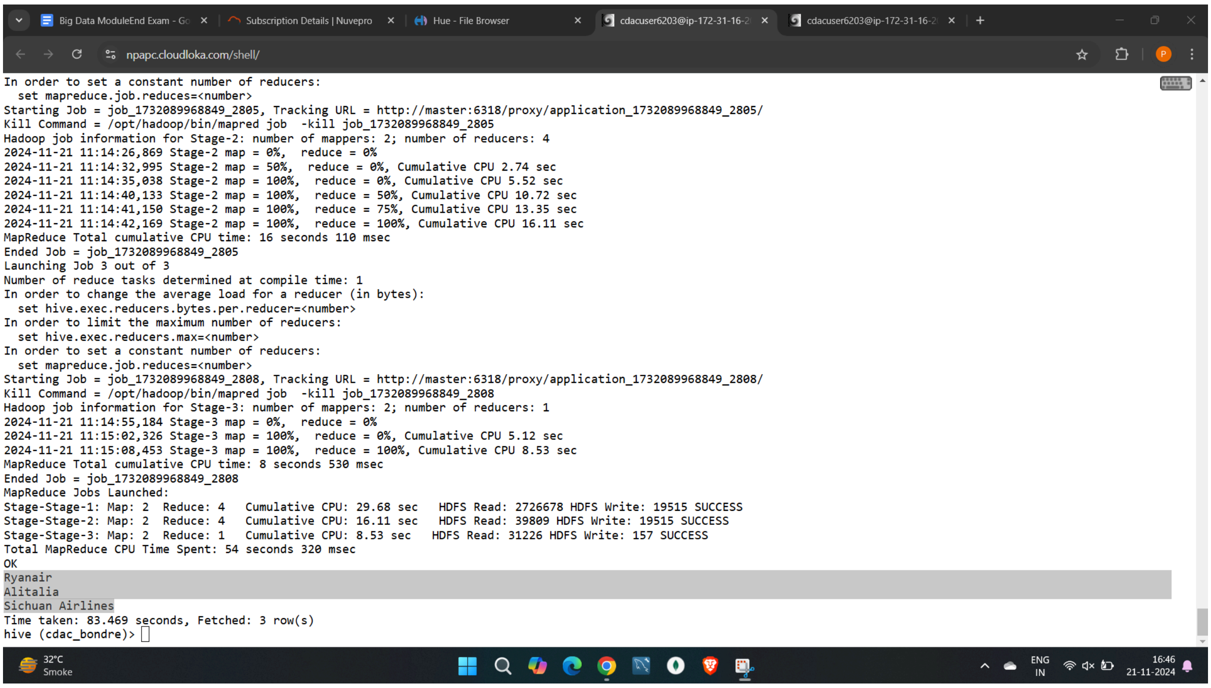


Big Data ModuleEnd Exam - Go
Subscription Details | Nuvepro
cdacuser6203@ip-172-31-16-2
cdacuser6203@ip-172-31-16-2
npapc.cloudloka.com/shell/
32°C
Smoke
ENG
IN
16:32
21-11-2024

```
2024-11-21 11:00:45,654 Stage-1 map = 0%,  reduce = 0%
2024-11-21 11:00:52,837 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 4.92 sec
2024-11-21 11:00:59,989 Stage-1 map = 100%,  reduce = 25%, Cumulative CPU 7.55 sec
2024-11-21 11:01:01,010 Stage-1 map = 100%,  reduce = 75%, Cumulative CPU 12.47 sec
2024-11-21 11:01:02,029 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 14.66 sec
MapReduce Total cumulative CPU time: 14 seconds 660 msec
Ended Job = job_1732089968849_2716
MapReduce Jobs Launched:
Stage-Stage-1: Map: 2  Reduce: 4   Cumulative CPU: 14.66 sec   HDFS Read: 36632 HDFS Write: 348 SUCCESS
Total MapReduce CPU Time Spent: 14 seconds 660 msec
OK
Time taken: 31.598 seconds
hive (cdac_bondre)> select a.name from airports a join routes r on a.id = r.airline_id where r.source_airport_Id = 2965 and r.destination_airport_Id is null;
Query ID = cdacuser6203_20241121110139_023eb4d8-e255-4900-9c77-b3852161ec58
Total jobs = 1
Launching Job 1 out of 1
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_2724, Tracking URL = http://master:6318/proxy/application_1732089968849_2724/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2724
Hadoop job information for Stage-1: number of mappers: 2; number of reducers: 4
2024-11-21 11:01:50,323 Stage-1 map = 0%,  reduce = 0%
2024-11-21 11:01:58,480 Stage-1 map = 50%,  reduce = 0%, Cumulative CPU 5.94 sec
2024-11-21 11:01:59,500 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 13.01 sec
2024-11-21 11:02:05,614 Stage-1 map = 100%,  reduce = 25%, Cumulative CPU 16.63 sec
2024-11-21 11:02:06,632 Stage-1 map = 100%,  reduce = 75%, Cumulative CPU 23.82 sec
2024-11-21 11:02:07,654 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 27.34 sec
MapReduce Total cumulative CPU time: 27 seconds 340 msec
Ended Job = job_1732089968849_2724
MapReduce Jobs Launched:
Stage-Stage-1: Map: 2  Reduce: 4   Cumulative CPU: 27.34 sec   HDFS Read: 3154988 HDFS Write: 348 SUCCESS
Total MapReduce CPU Time Spent: 27 seconds 340 msec
OK
Time taken: 31.482 seconds
hive (cdac_bondre)> select a.name from airports a join routes r on a.id = r.airline_id where r.source_airport_Id = 2965 and r.destination_airport_Id is null;
```

2.

→ select distinct(a.name) from airlines a join routes r on a.id = r.airline_id order by max(source_airport_id) desc limit 3 ;

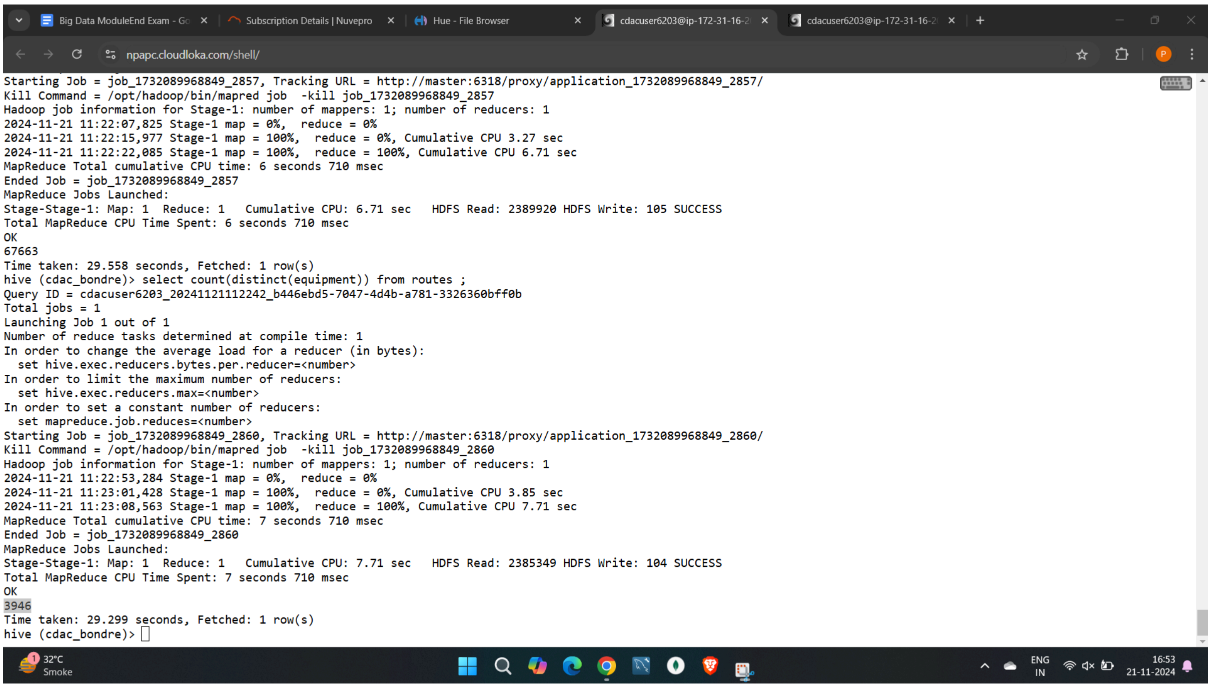


Big Data ModuleEnd Exam - Go
Subscription Details | Nuvepro
Hue - File Browser
cdacuser6203@ip-172-31-16-2
cdacuser6203@ip-172-31-16-2
npapc.cloudloka.com/shell/
32°C
Smoke
ENG
IN
16:46
21-11-2024

```
In order to set a constant number of reducers:
  set mapreduce.job.reduces=<number>
Starting Job = job_1732089968849_2805, Tracking URL = http://master:6318/proxy/application_1732089968849_2805/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2805
Hadoop job information for Stage-2: number of mappers: 2; number of reducers: 4
2024-11-21 11:14:26,869 Stage-2 map = 0%,  reduce = 0%
2024-11-21 11:14:32,995 Stage-2 map = 50%,  reduce = 0%, Cumulative CPU 2.74 sec
2024-11-21 11:14:35,038 Stage-2 map = 100%,  reduce = 0%, Cumulative CPU 5.52 sec
2024-11-21 11:14:40,133 Stage-2 map = 100%,  reduce = 50%, Cumulative CPU 10.72 sec
2024-11-21 11:14:41,150 Stage-2 map = 100%,  reduce = 75%, Cumulative CPU 13.35 sec
2024-11-21 11:14:42,169 Stage-2 map = 100%,  reduce = 100%, Cumulative CPU 16.11 sec
MapReduce Total cumulative CPU time: 16 seconds 110 msec
Ended Job = job_1732089968849_2805
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_2808, Tracking URL = http://master:6318/proxy/application_1732089968849_2808/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2808
Hadoop job information for Stage-3: number of mappers: 2; number of reducers: 1
2024-11-21 11:14:55,184 Stage-3 map = 0%,  reduce = 0%
2024-11-21 11:15:02,326 Stage-3 map = 100%,  reduce = 0%, Cumulative CPU 5.12 sec
2024-11-21 11:15:08,453 Stage-3 map = 100%,  reduce = 100%, Cumulative CPU 8.53 sec
MapReduce Total cumulative CPU time: 8 seconds 530 msec
Ended Job = job_1732089968849_2808
MapReduce Jobs Launched:
Stage-Stage-1: Map: 2  Reduce: 4   Cumulative CPU: 29.68 sec   HDFS Read: 2726678 HDFS Write: 19515 SUCCESS
Stage-Stage-2: Map: 2  Reduce: 4   Cumulative CPU: 16.11 sec   HDFS Read: 39809 HDFS Write: 19515 SUCCESS
Stage-Stage-3: Map: 2  Reduce: 1   Cumulative CPU: 8.53 sec   HDFS Read: 31226 HDFS Write: 157 SUCCESS
Total MapReduce CPU Time Spent: 54 seconds 320 msec
OK
Ryanair
Alitalia
Sichuan Airlines
Time taken: 83.469 seconds, Fetched: 3 row(s)
hive (cdac_bondre)> 
```

3 .

→ select count(distinct(equipment)) from routes ;



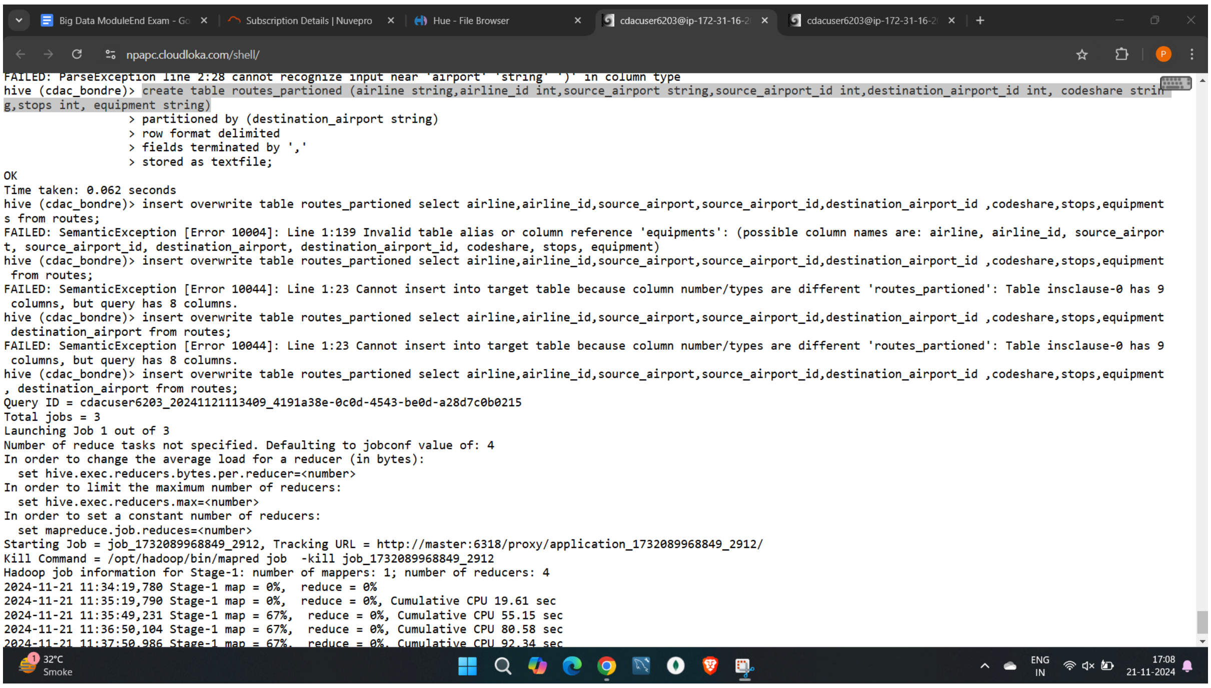
Big Data ModuleEnd Exam - Go
Subscription Details | Nuvepro
Hue - File Browser
cdacuser6203@ip-172-31-16-2
cdacuser6203@ip-172-31-16-2
npapc.cloudloka.com/shell/
32°C
Smoke
ENG
IN
16:53
21-11-2024

```
Starting Job = job_1732089968849_2857, Tracking URL = http://master:6318/proxy/application_1732089968849_2857/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2857
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-11-21 11:22:07,825 Stage-1 map = 0%,  reduce = 0%
2024-11-21 11:22:15,977 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 3.27 sec
2024-11-21 11:22:22,085 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 6.71 sec
MapReduce Total cumulative CPU time: 6 seconds 710 msec
Ended Job = job_1732089968849_2857
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 6.71 sec   HDFS Read: 2389920 HDFS Write: 105 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 710 msec
OK
67663
Time taken: 29.558 seconds, Fetched: 1 row(s)
hive (cdac_bondre)> select count(distinct(equipment)) from routes ;
Query ID = cdacuser6203_20241121112242_b446ebd5-7047-4d4b-a781-3326360bff0b
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_2860, Tracking URL = http://master:6318/proxy/application_1732089968849_2860/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2860
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2024-11-21 11:22:53,284 Stage-1 map = 0%,  reduce = 0%
2024-11-21 11:23:01,428 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 3.85 sec
2024-11-21 11:23:08,563 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 7.71 sec
MapReduce Total cumulative CPU time: 7 seconds 710 msec
Ended Job = job_1732089968849_2860
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 7.71 sec   HDFS Read: 2385349 HDFS Write: 104 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 710 msec
OK
3946
Time taken: 29.299 seconds, Fetched: 1 row(s)
hive (cdac_bondre)> 
```

Q2

1.

```
→ create table routes_partioned (airline string,airline_id
int,source_airport string,source_airport_id
int,destination_airport_id int, codeshare strin
g,stops int, equipment string)
                          > partitioned by (destination_airport string)
                          > row format delimited
                          > fields terminated by ','
                          > stored as textfile;
```

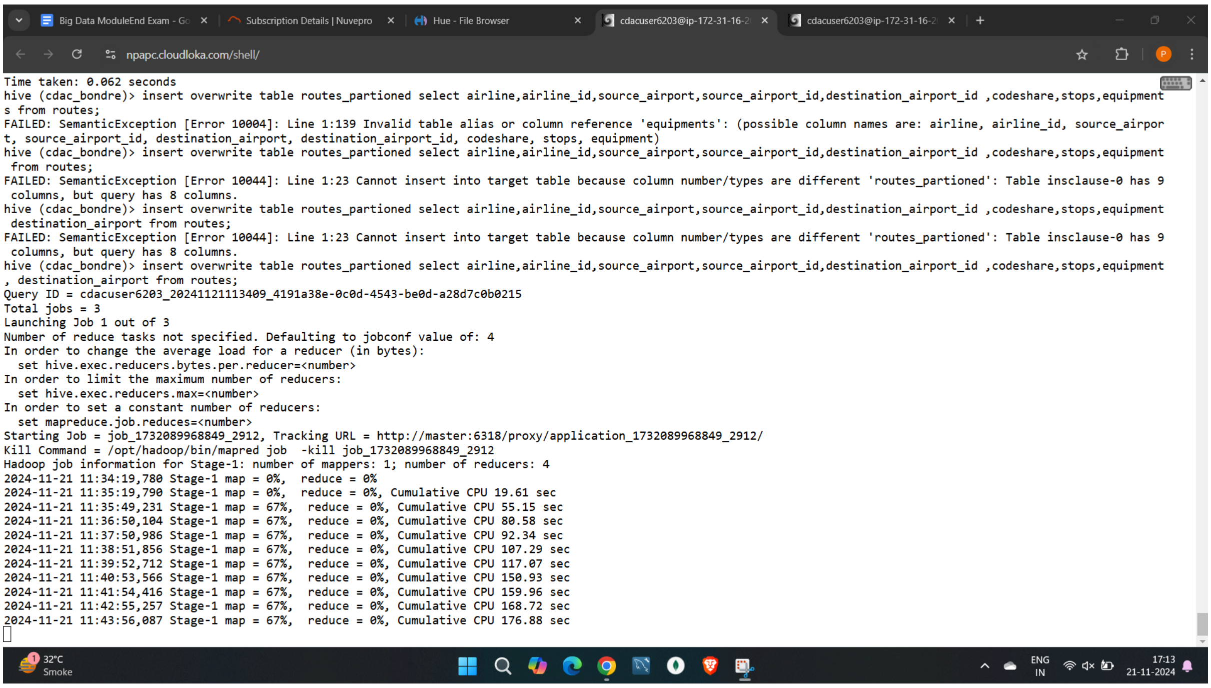

Big Data ModuleEnd Exam - Go
Subscription Details | Nuvepro
Hue - File Browser
cdacuser6203@ip-172-31-16-2
cdacuser6203@ip-172-31-16-2
npapc.cloudloka.com/shell/
32°C
Smoke
ENG
IN
17:08
21-11-2024

```
FAILED: ParseException line 2:28 cannot recognize input near 'airport' 'string' ')' in column type
hive (cdac_bondre)> create table routes_partioned (airline string,airline_id int,source_airport string,source_airport_id int,destination_airport_id int, codeshare strin
g,stops int, equipment string)
                  > partitioned by (destination_airport string)
                  > row format delimited
                  > fields terminated by ','
                  > stored as textfile;
OK
Time taken: 0.062 seconds
hive (cdac_bondre)> insert overwrite table routes_partioned select airline,airline_id,source_airport,source_airport_id,destination_airport_id ,codeshare,stops,equipment
s from routes;
FAILED: SemanticException [Error 10004]: Line 1:139 Invalid table alias or column reference 'equipments': (possible column names are: airline, airline_id, source_airpor
t, source_airport_id, destination_airport, destination_airport_id, codeshare, stops, equipment)
hive (cdac_bondre)> insert overwrite table routes_partioned select airline,airline_id,source_airport,source_airport_id,destination_airport_id ,codeshare,stops,equipment
 from routes;
FAILED: SemanticException [Error 10044]: Line 1:23 Cannot insert into target table because column number/types are different 'routes_partioned': Table insclause-0 has 9
 columns, but query has 8 columns.
hive (cdac_bondre)> insert overwrite table routes_partioned select airline,airline_id,source_airport,source_airport_id,destination_airport_id ,codeshare,stops,equipment
 destination_airport from routes;
FAILED: SemanticException [Error 10044]: Line 1:23 Cannot insert into target table because column number/types are different 'routes_partioned': Table insclause-0 has 9
 columns, but query has 8 columns.
hive (cdac_bondre)> insert overwrite table routes_partioned select airline,airline_id,source_airport,source_airport_id,destination_airport_id ,codeshare,stops,equipment
, destination_airport from routes;
Query ID = cdacuser6203_20241121113409_4191a38e-0c0d-4543-be0d-a28d7c0b0215
Total jobs = 3
Launching Job 1 out of 3
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_2912, Tracking URL = http://master:6318/proxy/application_1732089968849_2912/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2912
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 4
2024-11-21 11:34:19,780 Stage-1 map = 0%,  reduce = 0%
2024-11-21 11:35:19,790 Stage-1 map = 0%,  reduce = 0%, Cumulative CPU 19.61 sec
2024-11-21 11:35:49,231 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 55.15 sec
2024-11-21 11:36:50,104 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 80.58 sec
2024-11-21 11:37:50,986 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 92.34 sec
```

2.

```
→  insert overwrite table routes_partioned select
airline,airline_id,source_airport,source_airport_id,destination_ai
rport_id ,codeshare,stops,equipment
s from routes;
```

The query got stucked in between the i have shared the query with ScreenShot



Big Data ModuleEnd Exam - Go
Subscription Details | Nuvepro
Hue - File Browser
cdacuser6203@ip-172-31-16-2
cdacuser6203@ip-172-31-16-2
npapc.cloudloka.com/shell/
32°C
Smoke
ENG
IN
17:13
21-11-2024

```
Time taken: 0.062 seconds
hive (cdac_bondre)> insert overwrite table routes_partioned select airline,airline_id,source_airport,source_airport_id,destination_airport_id ,codeshare,stops,equipment
s from routes;
FAILED: SemanticException [Error 10004]: Line 1:139 Invalid table alias or column reference 'equipments': (possible column names are: airline, airline_id, source_airpor
t, source_airport_id, destination_airport, destination_airport_id, codeshare, stops, equipment)
hive (cdac_bondre)> insert overwrite table routes_partioned select airline,airline_id,source_airport,source_airport_id,destination_airport_id ,codeshare,stops,equipment
 from routes;
FAILED: SemanticException [Error 10044]: Line 1:23 Cannot insert into target table because column number/types are different 'routes_partioned': Table insclause-0 has 9
 columns, but query has 8 columns.
hive (cdac_bondre)> insert overwrite table routes_partioned select airline,airline_id,source_airport,source_airport_id,destination_airport_id ,codeshare,stops,equipment
 destination_airport from routes;
FAILED: SemanticException [Error 10044]: Line 1:23 Cannot insert into target table because column number/types are different 'routes_partioned': Table insclause-0 has 9
 columns, but query has 8 columns.
hive (cdac_bondre)> insert overwrite table routes_partioned select airline,airline_id,source_airport,source_airport_id,destination_airport_id ,codeshare,stops,equipment
, destination_airport from routes;
Query ID = cdacuser6203_20241121113409_4191a38e-0c0d-4543-be0d-a28d7c0b0215
Total jobs = 3
Launching Job 1 out of 3
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_2912, Tracking URL = http://master:6318/proxy/application_1732089968849_2912/
Kill Command = /opt/hadoop/bin/mapred job  -kill job_1732089968849_2912
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 4
2024-11-21 11:34:19,780 Stage-1 map = 0%,  reduce = 0%
2024-11-21 11:35:19,790 Stage-1 map = 0%,  reduce = 0%, Cumulative CPU 19.61 sec
2024-11-21 11:35:49,231 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 55.15 sec
2024-11-21 11:36:50,104 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 80.58 sec
2024-11-21 11:37:50,986 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 92.34 sec
2024-11-21 11:38:51,856 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 107.29 sec
2024-11-21 11:39:52,712 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 117.07 sec
2024-11-21 11:40:53,566 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 150.93 sec
2024-11-21 11:41:54,416 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 159.96 sec
2024-11-21 11:42:55,257 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 168.72 sec
2024-11-21 11:43:56,087 Stage-1 map = 67%,  reduce = 0%, Cumulative CPU 176.88 sec
```

Q3

→

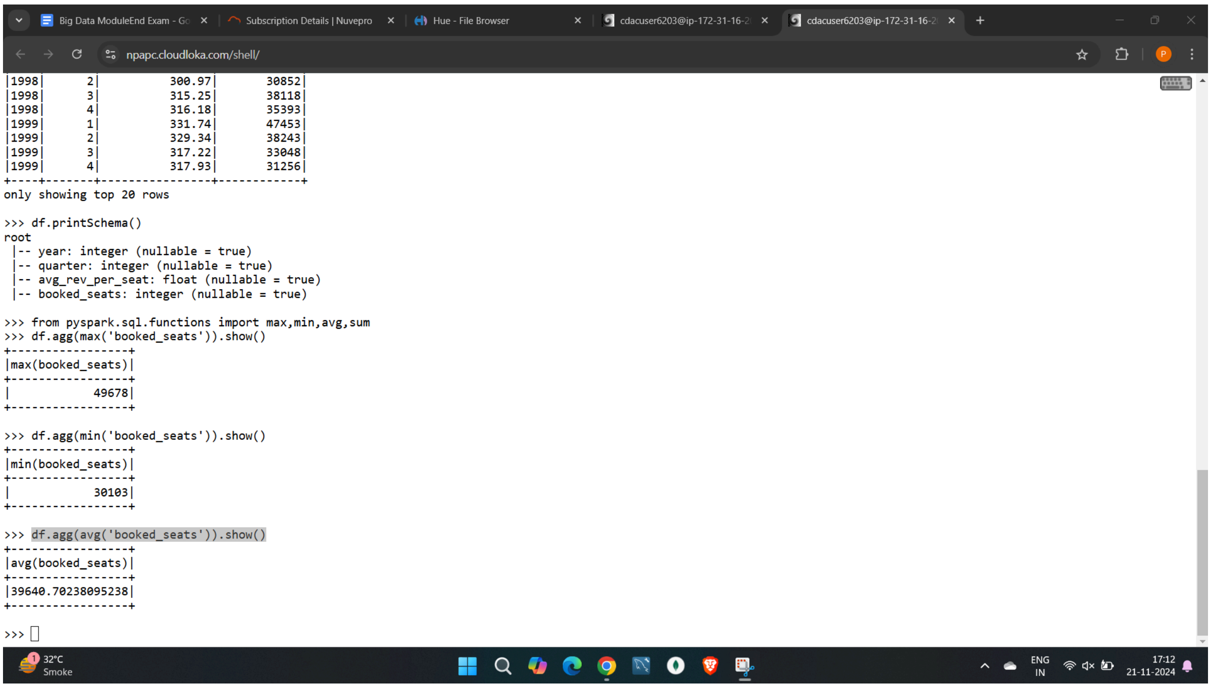
SPARK

Q2

1.

→

```
 Max- df.agg(max('booked_seats')).show()
 Min- df.agg(min('booked_seats')).show()
 Avg- df.agg(avg('booked_seats')).show()
```


Big Data ModuleEnd Exam - Go
Subscription Details | Nuvepro
Hue - File Browser
cdacuser6203@ip-172-31-16-2
cdacuser6203@ip-172-31-16-2
npapc.cloudloka.com/shell/
32°C
Smoke
ENG
IN
17:12
21-11-2024

```
|1998|      2|          300.97|       30852|
|1998|      3|          315.25|       38118|
|1998|      4|          316.18|       35393|
|1999|      1|          331.74|       47453|
|1999|      2|          329.34|       38243|
|1999|      3|          317.22|       33048|
|1999|      4|          317.93|       31256|
+----+-------+----------------+------------+
only showing top 20 rows

>>> df.printSchema()
root
 |-- year: integer (nullable = true)
 |-- quarter: integer (nullable = true)
 |-- avg_rev_per_seat: float (nullable = true)
 |-- booked_seats: integer (nullable = true)

>>> from pyspark.sql.functions import max,min,avg,sum
>>> df.agg(max('booked_seats')).show()
+-----------------+
|max(booked_seats)|
+-----------------+
|            49678|
+-----------------+

>>> df.agg(min('booked_seats')).show()
+-----------------+
|min(booked_seats)|
+-----------------+
|            30103|
+-----------------+

>>> df.agg(avg('booked_seats')).show()
+-----------------+
|avg(booked_seats)|
+-----------------+
|39640.70238095238|
+-----------------+

>>> 
```

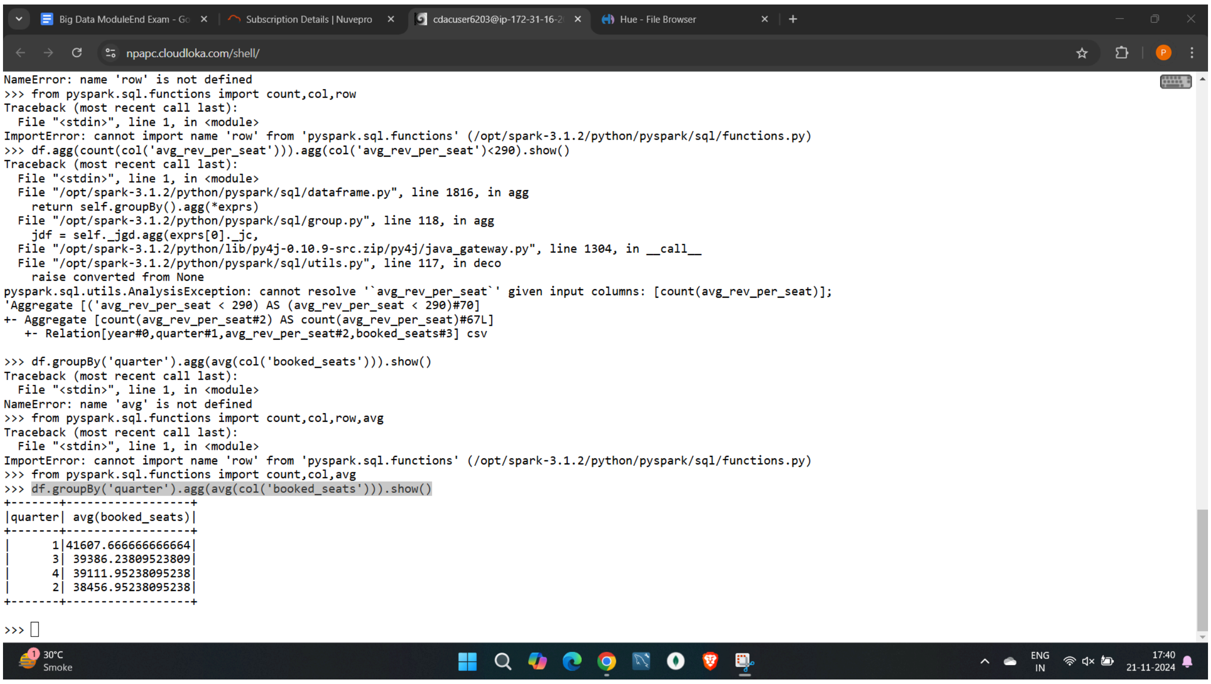
2.

→

```
df.agg(count(col('avg_rev_per_seat'))).agg(col('avg_rev_per_seat')
<290).show()
```

3.

```
→ df.groupBy('quarter').agg(avg(col('booked_seats'))).show()
```


Big Data ModuleEnd Exam - Go
Subscription Details | Nuvepro
cdacuser6203@ip-172-31-16-2
Hue - File Browser
npapc.cloudloka.com/shell/
30°C
Smoke
ENG
IN
17:40
21-11-2024

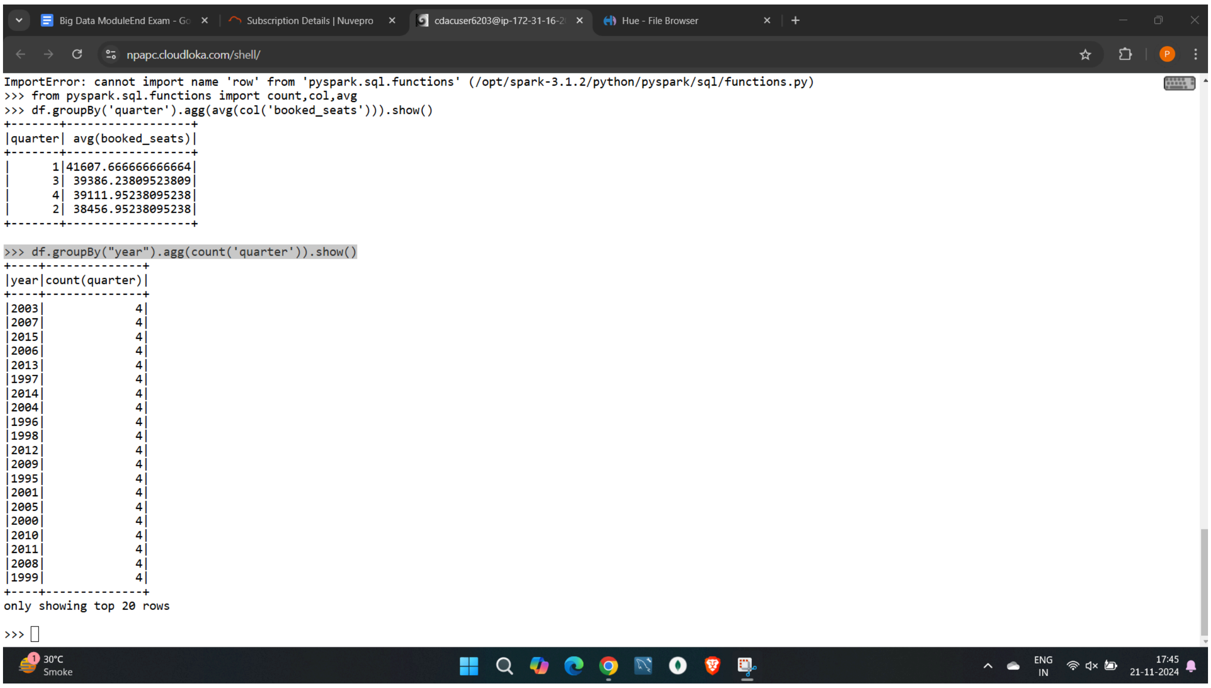
```
NameError: name 'row' is not defined
>>> from pyspark.sql.functions import count,col,row
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: cannot import name 'row' from 'pyspark.sql.functions' (/opt/spark-3.1.2/python/pyspark/sql/functions.py)
>>> df.agg(count(col('avg_rev_per_seat'))).agg(col('avg_rev_per_seat')<290).show()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "/opt/spark-3.1.2/python/pyspark/sql/dataframe.py", line 1816, in agg
    return self.groupBy().agg(*exprs)
  File "/opt/spark-3.1.2/python/pyspark/sql/group.py", line 118, in agg
    jdf = self._jgd.agg(exprs[0]._jc,
  File "/opt/spark-3.1.2/python/lib/py4j-0.10.9-src.zip/py4j/java_gateway.py", line 1304, in __call__
  File "/opt/spark-3.1.2/python/pyspark/sql/utils.py", line 117, in deco
    raise converted from None
pyspark.sql.utils.AnalysisException: cannot resolve '`avg_rev_per_seat`' given input columns: [count(avg_rev_per_seat)];
'Aggregate [('avg_rev_per_seat < 290) AS (avg_rev_per_seat < 290)#70]
+- Aggregate [count(avg_rev_per_seat#2) AS count(avg_rev_per_seat)#67L]
   +- Relation[year#0,quarter#1,avg_rev_per_seat#2,booked_seats#3] csv

>>> df.groupBy('quarter').agg(avg(col('booked_seats'))).show()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'avg' is not defined
>>> from pyspark.sql.functions import count,col,row,avg
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ImportError: cannot import name 'row' from 'pyspark.sql.functions' (/opt/spark-3.1.2/python/pyspark/sql/functions.py)
>>> from pyspark.sql.functions import count,col,avg
>>> df.groupBy('quarter').agg(avg(col('booked_seats'))).show()
+-------+-----------------+
|quarter| avg(booked_seats)|
+-------+-----------------+
|      1|41607.666666666664|
|      3| 39386.23809523809|
|      4| 39111.95238095238|
|      2| 38456.95238095238|
+-------+-----------------+

>>> 
```

4.

→ df.groupBy("year").agg(count('quarter')).show()


Big Data ModuleEnd Exam - Go
Subscription Details | Nuvepro
cdacuser6203@ip-172-31-16-2
Hue - File Browser
npapc.cloudloka.com/shell/
30°C
Smoke
ENG
IN
17:45
21-11-2024

```
ImportError: cannot import name 'row' from 'pyspark.sql.functions' (/opt/spark-3.1.2/python/pyspark/sql/functions.py)
>>> from pyspark.sql.functions import count,col,avg
>>> df.groupBy('quarter').agg(avg(col('booked_seats'))).show()
+-------+-----------------+
|quarter| avg(booked_seats)|
+-------+-----------------+
|      1|41607.666666666664|
|      3| 39386.23809523809|
|      4| 39111.95238095238|
|      2| 38456.95238095238|
+-------+-----------------+

>>> df.groupBy("year").agg(count('quarter')).show()
+----+--------------+
|year|count(quarter)|
+----+--------------+
|2003|             4|
|2007|             4|
|2015|             4|
|2006|             4|
|2013|             4|
|1997|             4|
|2014|             4|
|2004|             4|
|1996|             4|
|1998|             4|
|2012|             4|
|2009|             4|
|1995|             4|
|2001|             4|
|2005|             4|
|2000|             4|
|2010|             4|
|2011|             4|
|2008|             4|
|1999|             4|
+----+--------------+
only showing top 20 rows

>>> 
```

5.

```
→ df.agg('quarter').agg(sum(col('avg_rev_per_seat') * col('booked_seats')).alias("revenue")).agg(max(col("revenue"))).show()
```