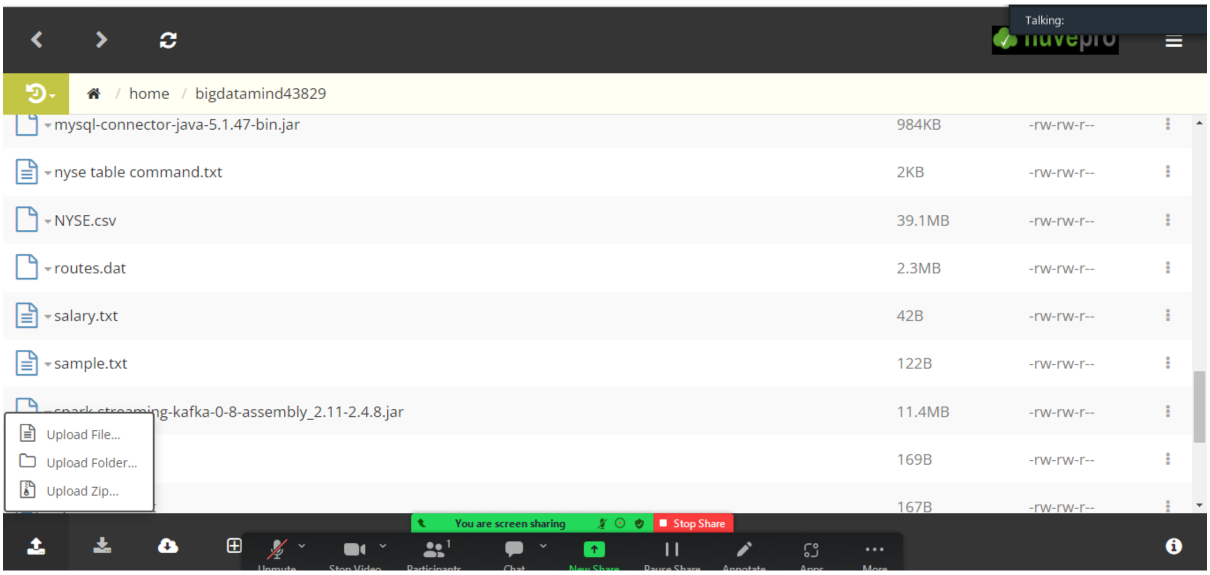
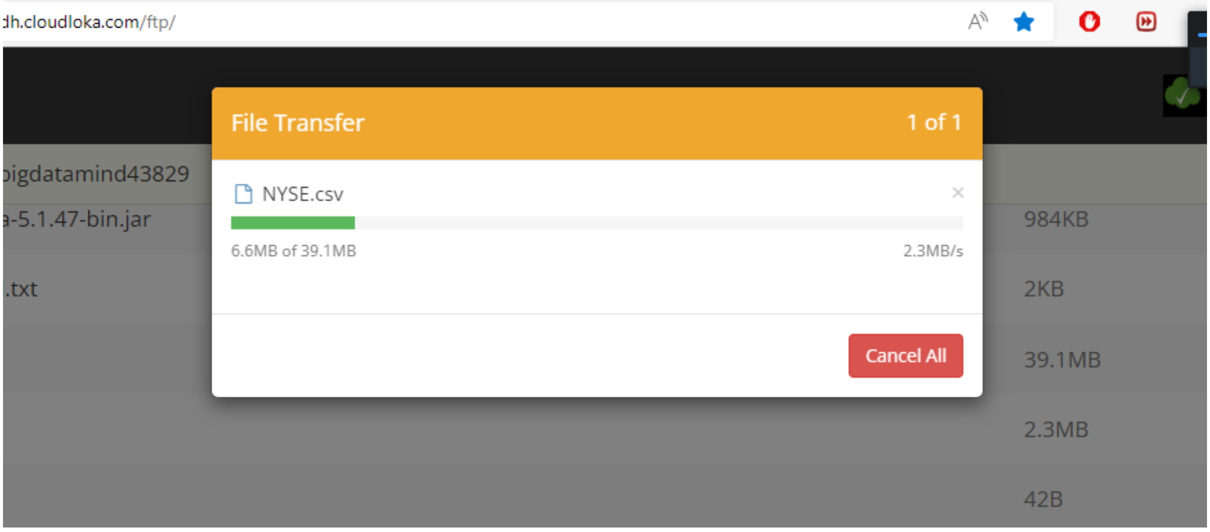
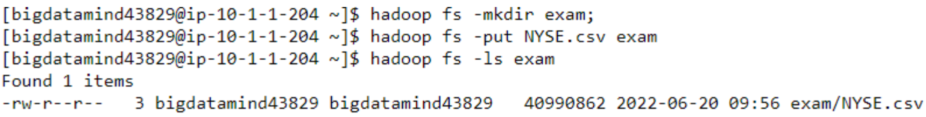
QUE 1}

```
package practice;
import java.io.*;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.DoubleWritable;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.fs.*;
import org.apache.hadoop.mapreduce.lib.input.*;
import org.apache.hadoop.mapreduce.lib.output.*;

public class AllTimeHigh {

      public static class MapClass extends 
Mapper<LongWritable,Text,Text,DoubleWritable>
         {
             private Text stock_id = new Text();
             private DoubleWritable High = new DoubleWritable();

            public void map(LongWritable key, Text value, Context context)
            {

               try{
                  String[] str = value.toString().split(",");
                  double high = Double.parseDouble(str[4]);
                  stock_id.set(str[1]);
                  High.set(high);

                  context.write(stock_id, High);
               }
               catch(Exception e)
               {
                  System.out.println(e.getMessage());
               }
            }
         }

        public static class ReduceClass extends 
Reducer<Text,DoubleWritable,Text,DoubleWritable>
         {
                private DoubleWritable result = new DoubleWritable();

                public void reduce(Text key, Iterable<DoubleWritable> 
values,Context context) throws IOException, InterruptedException {
                        double maxValue=0;
                        double temp_val=0;

                        for (DoubleWritable value : values) {
                              temp_val = value.get();
                              if (temp_val > maxValue) {
                                    maxValue = temp_val;
                              }
                        }
                        result.set(maxValue);
                  context.write(key, result);

                }
         }
        public static void main(String[] args) throws Exception {
                Configuration conf = new Configuration();

                Job job = Job.getInstance(conf, "Highest Price for each
stock");

                job.setJarByClass(AllTimeHigh.class);
                job.setMapperClass(MapClass.class);

                job.setReducerClass(ReduceClass.class);
                job.setNumReduceTasks(1);
                job.setOutputKeyClass(Text.class);
                job.setOutputValueClass(DoubleWritable.class);

                FileInputFormat.addInputPath(job, new Path(args[0]));
                FileOutputFormat.setOutputPath(job, new Path(args[1]));

                System.exit(job.waitForCompletion(true) ? 0 : 1);
              }

}
```

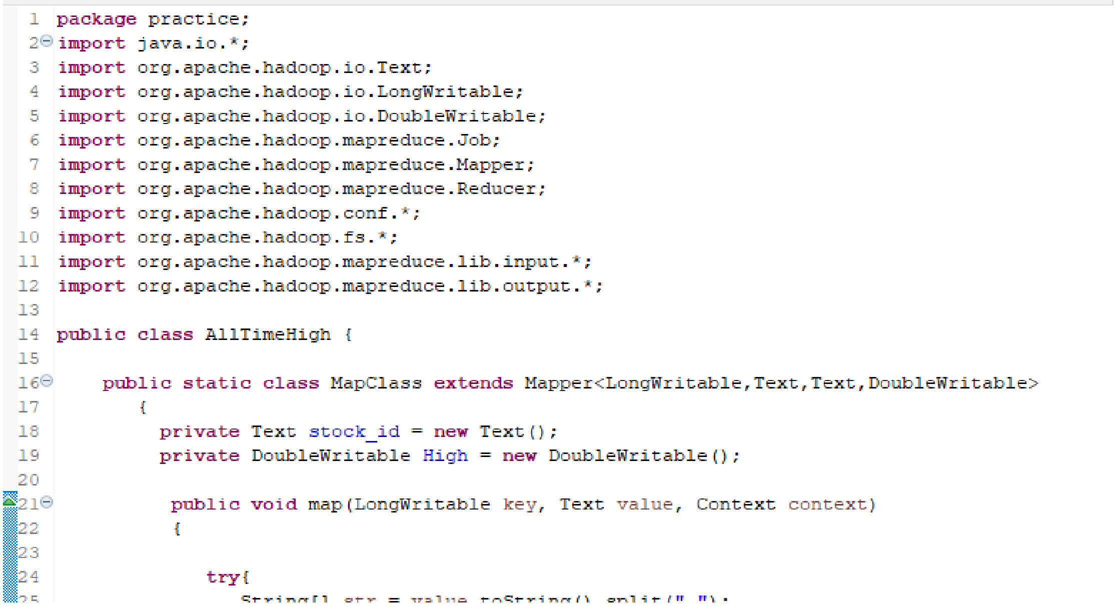


mysql-connector-java-5.1.47-bin.jar	984KB	-rw-rw-r--
nyse table command.txt	2KB	-rw-rw-r--
NYSE.csv	39.1MB	-rw-rw-r--
routes.dat	2.3MB	-rw-rw-r--
salary.txt	42B	-rw-rw-r--
sample.txt	122B	-rw-rw-r--
spark-streaming-kafka-0-8-assembly_2.11-2.4.8.jar	11.4MB	-rw-rw-r--
	169B	-rw-rw-r--
	167B	-rw-rw-r--

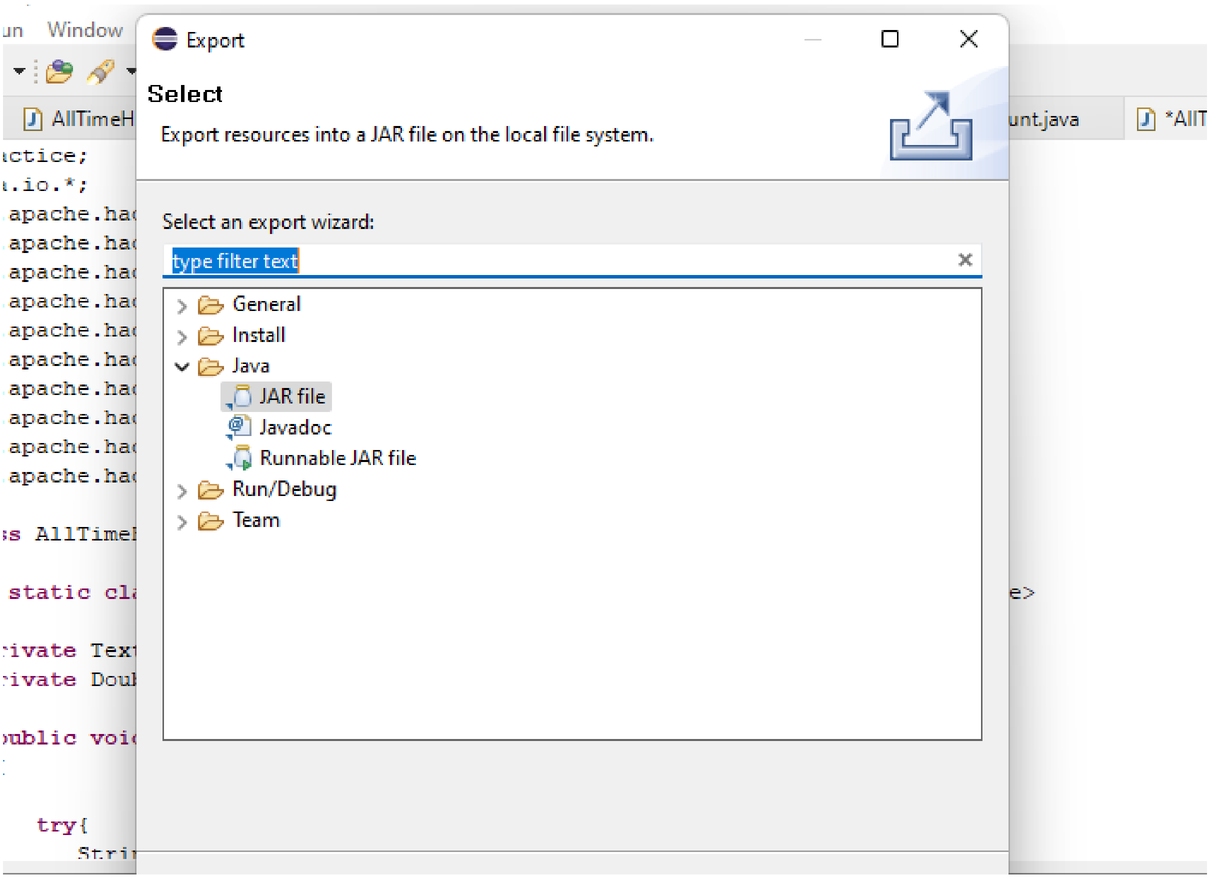




```
[bigdatamind43829@ip-10-1-1-204 ~]$ hadoop fs -mkdir exam;
[bigdatamind43829@ip-10-1-1-204 ~]$ hadoop fs -put NYSE.csv exam
[bigdatamind43829@ip-10-1-1-204 ~]$ hadoop fs -ls exam
Found 1 items
-rw-r--r--   3 bigdatamind43829 bigdatamind43829   40990862 2022-06-20 09:56 exam/NYSE.csv
```



```java
package practice;
import java.io.*;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.DoubleWritable;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.conf.*;
import org.apache.hadoop.fs.*;
import org.apache.hadoop.mapreduce.lib.input.*;
import org.apache.hadoop.mapreduce.lib.output.*;

public class AllTimeHigh {

    public static class MapClass extends Mapper<LongWritable,Text,Text,DoubleWritable>
       {
         private Text stock_id = new Text();
         private DoubleWritable High = new DoubleWritable();

          public void map(LongWritable key, Text value, Context context)
          {

             try{
                String[] str = value.toString().split(",");
```



Export

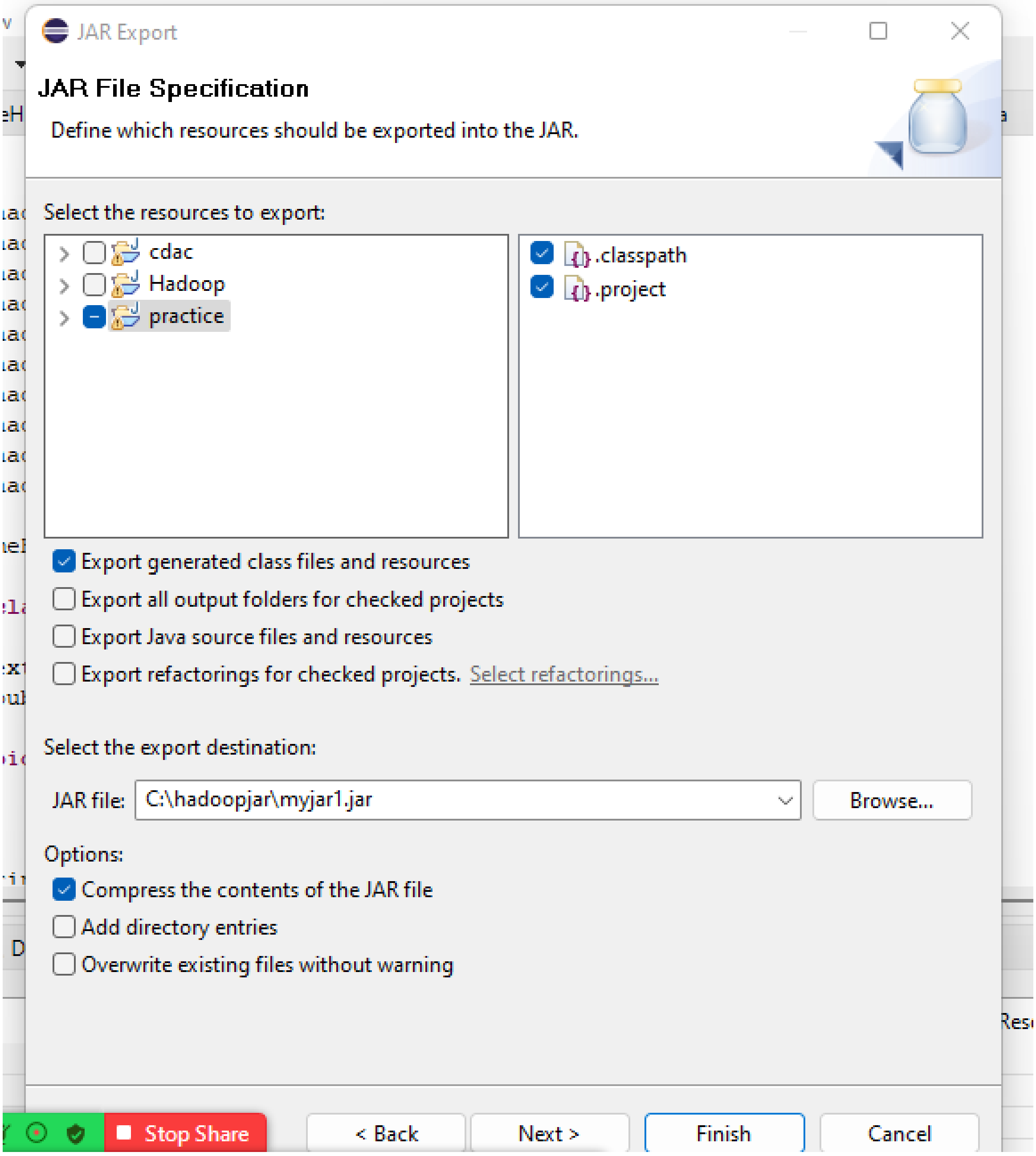
Select

Export resources into a JAR file on the local file system.

Select an export wizard:

type filter text

```
actice;
a.io.*;
 apache.had
 apache.had
 apache.had
 apache.had
 apache.had
 apache.had
 apache.had
 apache.had
 apache.had
 apache.had

ss AllTimeH

 static cla                                                              e>

rivate Text
rivate Doub

public voic
:

    try{
       Strin
```



JAR File Specification

Define which resources should be exported into the JAR.

Select the resources to export:

- □ cdac
- □ Hadoop
- ■ practice

☑ .classpath

☑ .project

☑ Export generated class files and resources

□ Export all output folders for checked projects

□ Export Java source files and resources

□ Export refactorings for checked projects. Select refactorings...

Select the export destination:

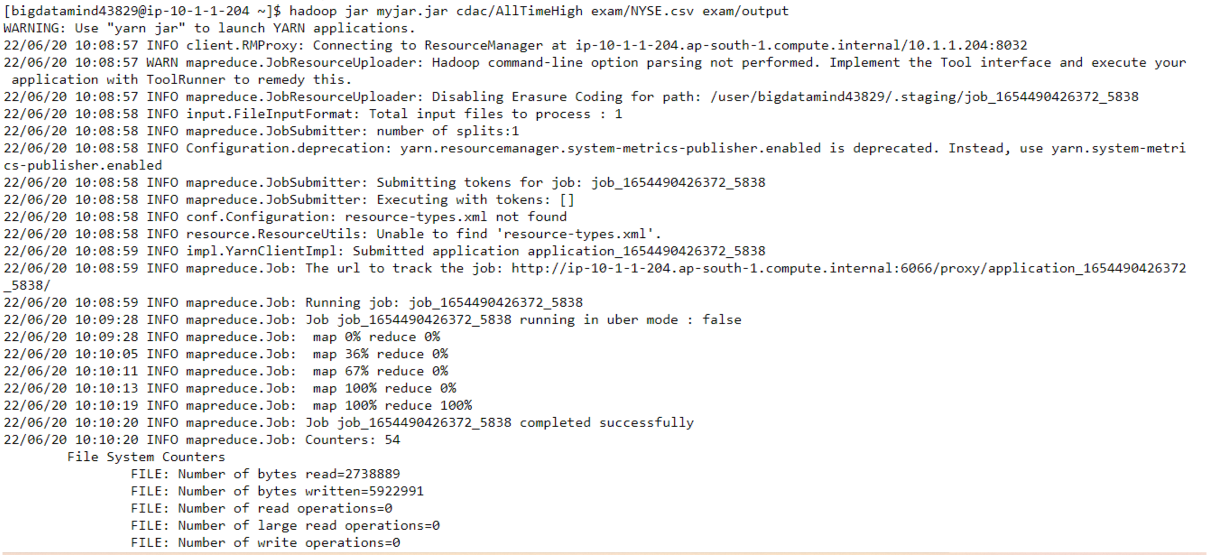
JAR file: C:\hadoopjar\myjar1.jar

Options:

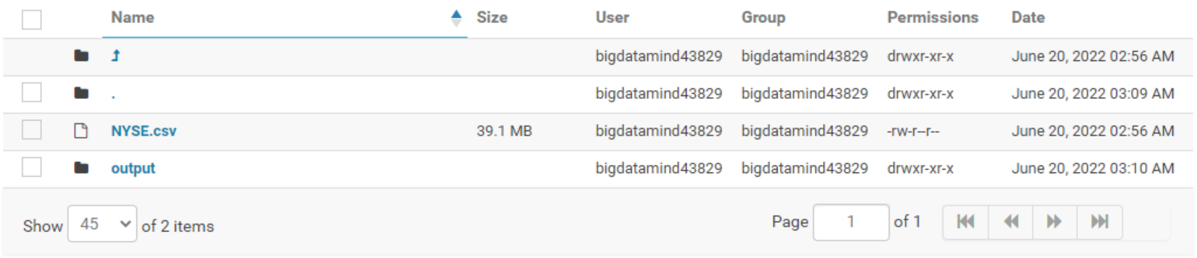
☑ Compress the contents of the JAR file

□ Add directory entries

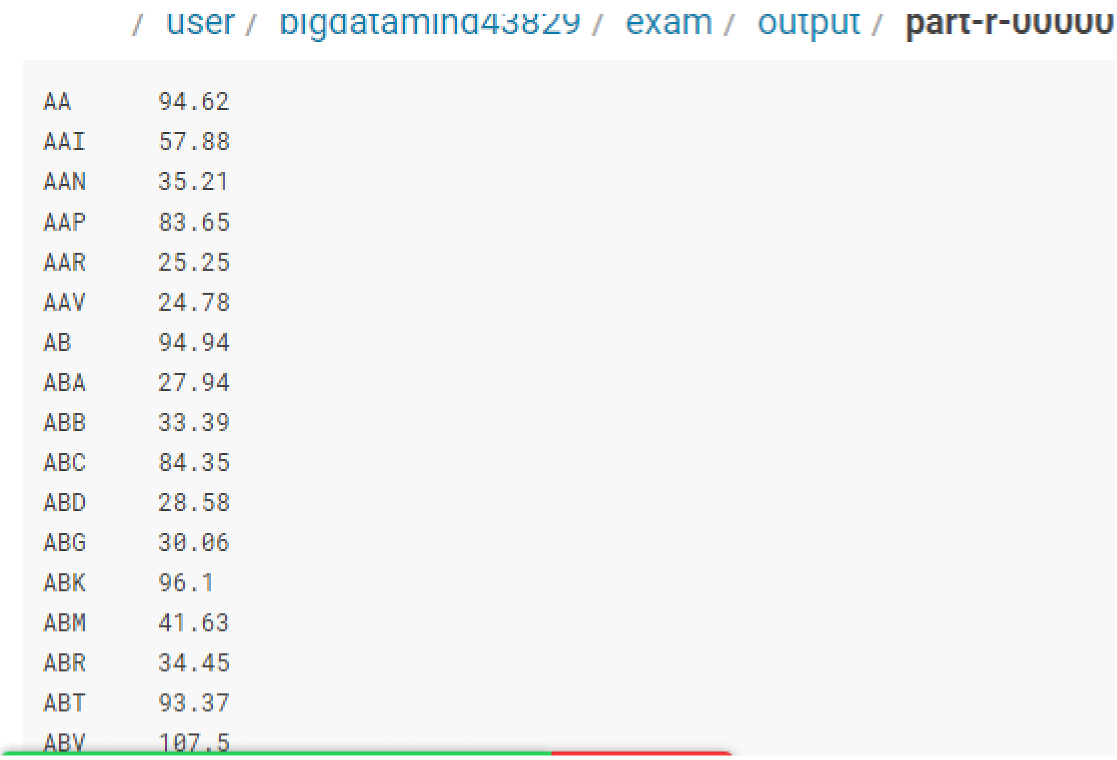
□ Overwrite existing files without warning



```
[bigdatamind43829@ip-10-1-1-204 ~]$ hadoop jar myjar.jar cdac/AllTimeHigh exam/NYSE.csv exam/output
WARNING: Use "yarn jar" to launch YARN applications.
22/06/20 10:08:57 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
22/06/20 10:08:57 WARN mapreduce.JobResourceUploader: Hadoop command-line option parsing not performed. Implement the Tool interface and execute your
 application with ToolRunner to remedy this.
22/06/20 10:08:57 INFO mapreduce.JobResourceUploader: Disabling Erasure Coding for path: /user/bigdatamind43829/.staging/job_1654490426372_5838
22/06/20 10:08:58 INFO input.FileInputFormat: Total input files to process : 1
22/06/20 10:08:58 INFO mapreduce.JobSubmitter: number of splits:1
22/06/20 10:08:58 INFO Configuration.deprecation: yarn.resourcemanager.system-metrics-publisher.enabled is deprecated. Instead, use yarn.system-metri
cs-publisher.enabled
22/06/20 10:08:58 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1654490426372_5838
22/06/20 10:08:58 INFO mapreduce.JobSubmitter: Executing with tokens: []
22/06/20 10:08:58 INFO conf.Configuration: resource-types.xml not found
22/06/20 10:08:58 INFO resource.ResourceUtils: Unable to find 'resource-types.xml'.
22/06/20 10:08:59 INFO impl.YarnClientImpl: Submitted application application_1654490426372_5838
22/06/20 10:08:59 INFO mapreduce.Job: The url to track the job: http://ip-10-1-1-204.ap-south-1.compute.internal:6066/proxy/application_1654490426372
_5838/
22/06/20 10:08:59 INFO mapreduce.Job: Running job: job_1654490426372_5838
22/06/20 10:09:28 INFO mapreduce.Job: Job job_1654490426372_5838 running in uber mode : false
22/06/20 10:09:28 INFO mapreduce.Job:  map 0% reduce 0%
22/06/20 10:10:05 INFO mapreduce.Job:  map 36% reduce 0%
22/06/20 10:10:11 INFO mapreduce.Job:  map 67% reduce 0%
22/06/20 10:10:13 INFO mapreduce.Job:  map 100% reduce 0%
22/06/20 10:10:19 INFO mapreduce.Job:  map 100% reduce 100%
22/06/20 10:10:20 INFO mapreduce.Job: Job job_1654490426372_5838 completed successfully
22/06/20 10:10:20 INFO mapreduce.Job: Counters: 54
        File System Counters
                FILE: Number of bytes read=2738889
                FILE: Number of bytes written=5922991
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
```



□		Name	Size	User	Group	Permissions	Date
		↥		bigdatamind43829	bigdatamind43829	drwxr-xr-x	June 20, 2022 02:56 AM
□		.		bigdatamind43829	bigdatamind43829	drwxr-xr-x	June 20, 2022 03:09 AM
□		NYSE.csv	39.1 MB	bigdatamind43829	bigdatamind43829	-rw-r--r--	June 20, 2022 02:56 AM
□		output		bigdatamind43829	bigdatamind43829	drwxr-xr-x	June 20, 2022 03:10 AM



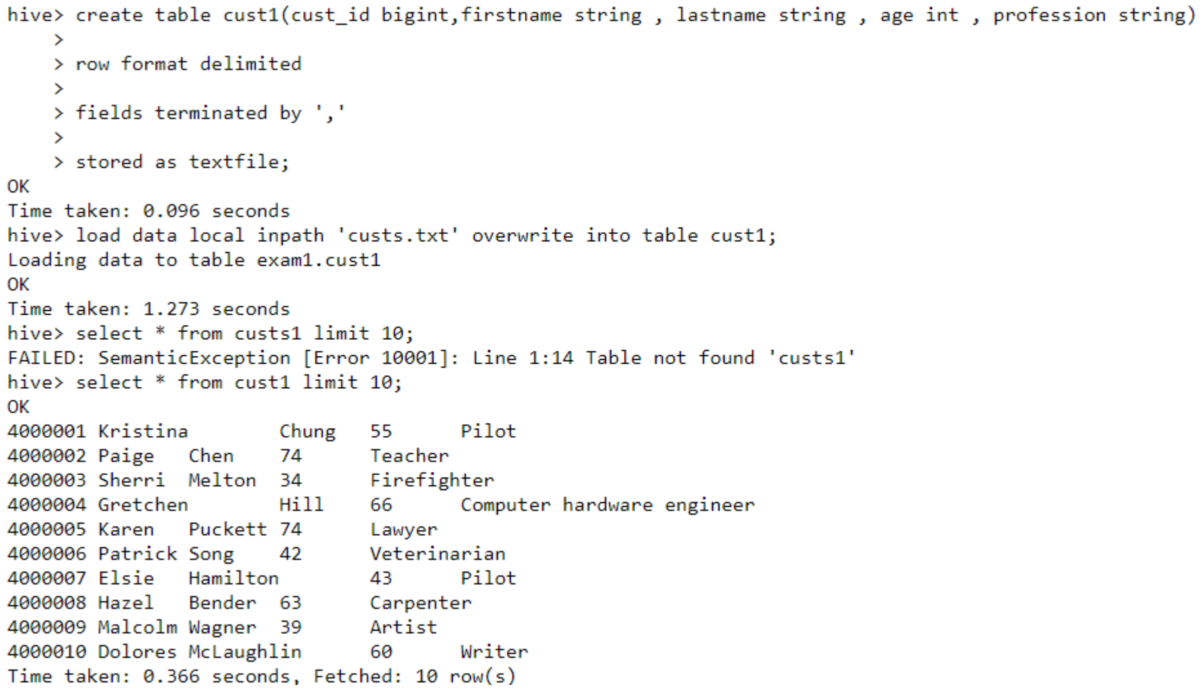
/ user / bigdatamind43829 / exam / output / part-r-00000

```
AA      94.62
AAI     57.88
AAN     35.21
AAP     83.65
AAR     25.25
AAV     24.78
AB      94.94
ABA     27.94
ABB     33.39
ABC     84.35
ABD     28.58
ABG     30.06
ABK     96.1
ABM     41.63
ABR     34.45
ABT     93.37
ABV     107.5
```

Hive Please find the customer data set.

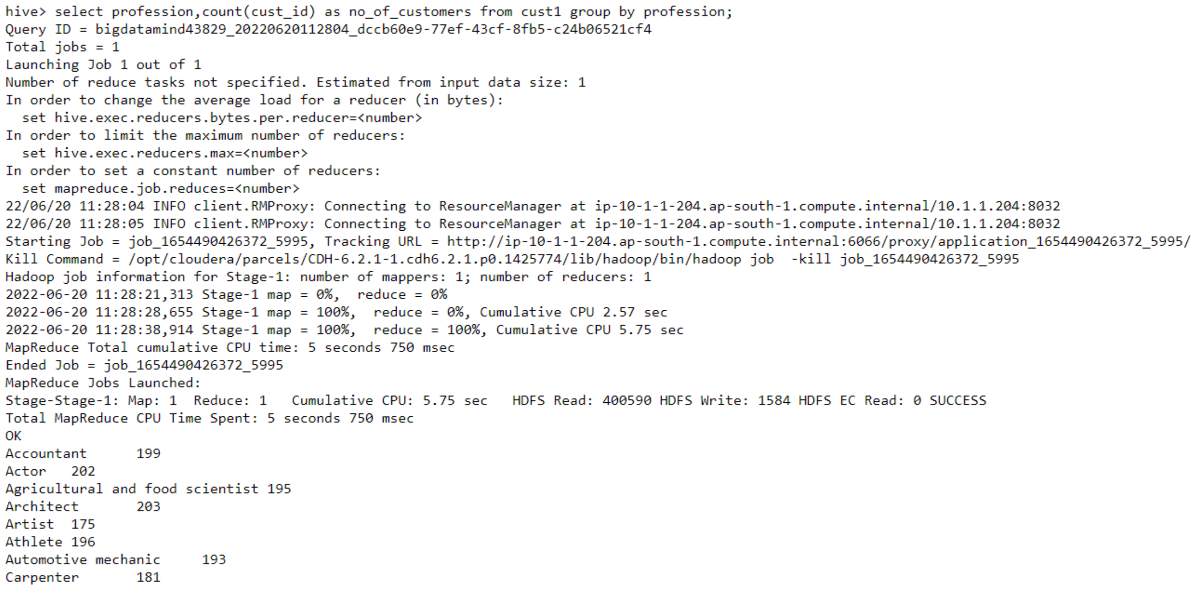
cust id firstname lastname age profession

1)Write a program to find the count of customers for each profession.

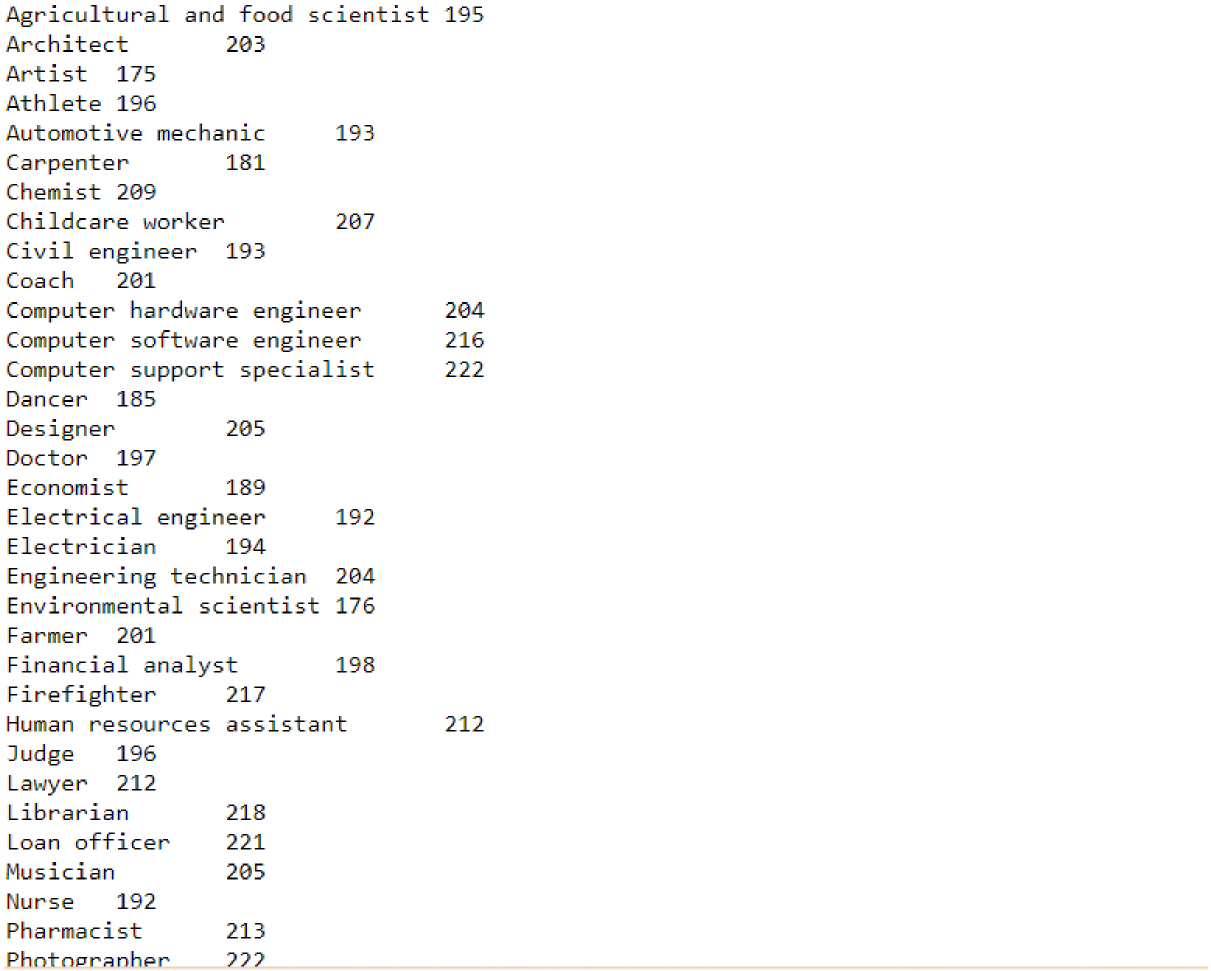


```
hive> create table cust1(cust_id bigint,firstname string , lastname string , age int , profession string)
    >
    > row format delimited
    >
    > fields terminated by ','
    >
    > stored as textfile;
OK
Time taken: 0.096 seconds
hive> load data local inpath 'custs.txt' overwrite into table cust1;
Loading data to table exam1.cust1
OK
Time taken: 1.273 seconds
hive> select * from custs1 limit 10;
FAILED: SemanticException [Error 10001]: Line 1:14 Table not found 'custs1'
hive> select * from cust1 limit 10;
OK
4000001 Kristina        Chung   55      Pilot
4000002 Paige   Chen    74      Teacher
4000003 Sherri  Melton  34      Firefighter
4000004 Gretchen        Hill    66      Computer hardware engineer
4000005 Karen   Puckett 74      Lawyer
4000006 Patrick Song    42      Veterinarian
4000007 Elsie   Hamilton        43      Pilot
4000008 Hazel   Bender  63      Carpenter
4000009 Malcolm Wagner  39      Artist
4000010 Dolores McLaughlin      60      Writer
Time taken: 0.366 seconds, Fetched: 10 row(s)
```

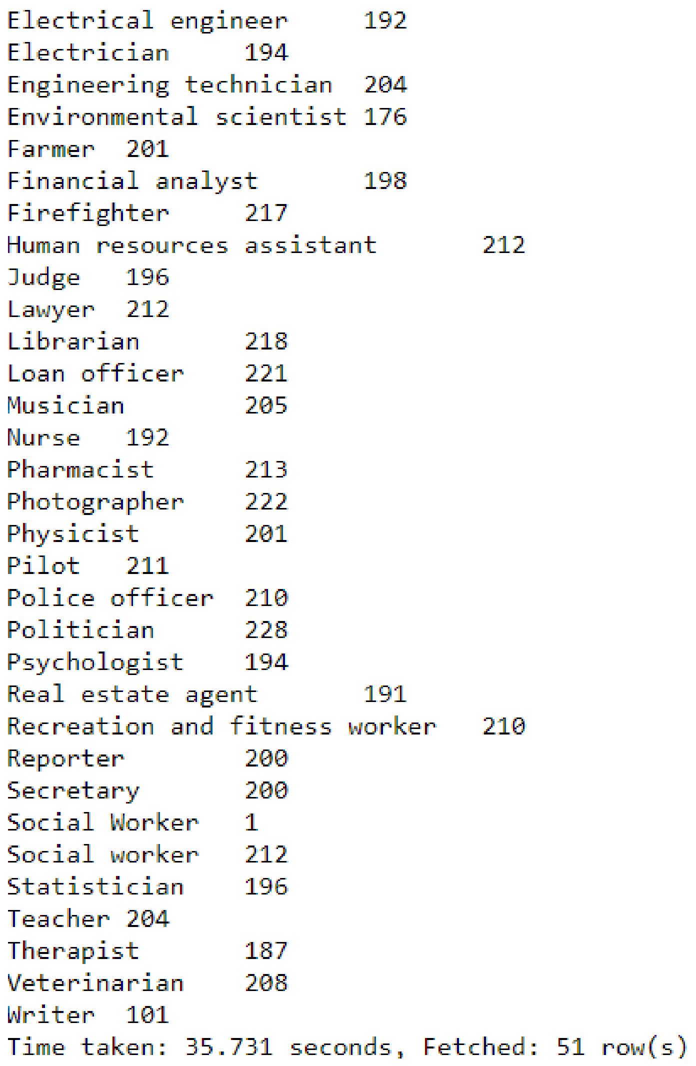
select profession,count(cust_id) as no_of_customers from cust1 group by profession;



```
hive> select profession,count(cust_id) as no_of_customers from cust1 group by profession;
Query ID = bigdatamind43829_20220620112804_dccb60e9-77ef-43cf-8fb5-c24b06521cf4
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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>
22/06/20 11:28:04 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
22/06/20 11:28:05 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
Starting Job = job_1654490426372_5995, Tracking URL = http://ip-10-1-1-204.ap-south-1.compute.internal:6066/proxy/application_1654490426372_5995/
Kill Command = /opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/bin/hadoop job  -kill job_1654490426372_5995
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-06-20 11:28:21,313 Stage-1 map = 0%,  reduce = 0%
2022-06-20 11:28:28,655 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 2.57 sec
2022-06-20 11:28:38,914 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 5.75 sec
MapReduce Total cumulative CPU time: 5 seconds 750 msec
Ended Job = job_1654490426372_5995
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 5.75 sec   HDFS Read: 400590 HDFS Write: 1584 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 5 seconds 750 msec
OK
Accountant      199
Actor   202
Agricultural and food scientist 195
Architect       203
Artist  175
Athlete 196
Automotive mechanic     193
Carpenter       181
```



```
Agricultural and food scientist 195
Architect       203
Artist  175
Athlete 196
Automotive mechanic     193
Carpenter       181
Chemist 209
Childcare worker        207
Civil engineer  193
Coach   201
Computer hardware engineer      204
Computer software engineer      216
Computer support specialist     222
Dancer  185
Designer        205
Doctor  197
Economist       189
Electrical engineer     192
Electrician     194
Engineering technician  204
Environmental scientist 176
Farmer  201
Financial analyst       198
Firefighter     217
Human resources assistant       212
Judge   196
Lawyer  212
Librarian       218
Loan officer    221
Musician        205
Nurse   192
Pharmacist      213
Photographer    222
```



```
Electrical engineer     192
Electrician     194
Engineering technician  204
Environmental scientist 176
Farmer  201
Financial analyst       198
Firefighter     217
Human resources assistant       212
Judge   196
Lawyer  212
Librarian       218
Loan officer    221
Musician        205
Nurse   192
Pharmacist      213
Photographer    222
Physicist       201
Pilot   211
Police officer  210
Politician      228
Psychologist    194
Real estate agent       191
Recreation and fitness worker   210
Reporter        200
Secretary       200
Social Worker   1
Social worker   212
Statistician    196
Teacher 204
Therapist       187
Veterinarian    208
Writer  101
Time taken: 35.731 seconds, Fetched: 51 row(s)
```

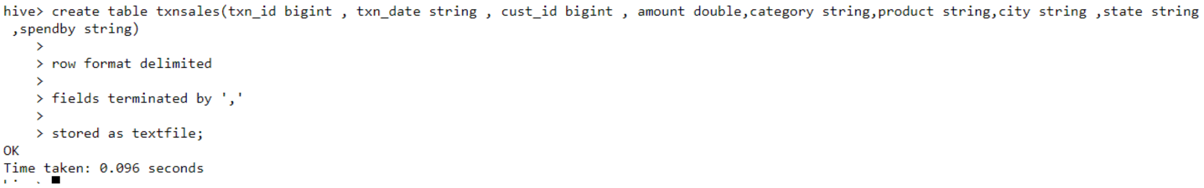
Please find the sales data set.

txn id
txn date
cust id
amount
category
product
city
state
spendby

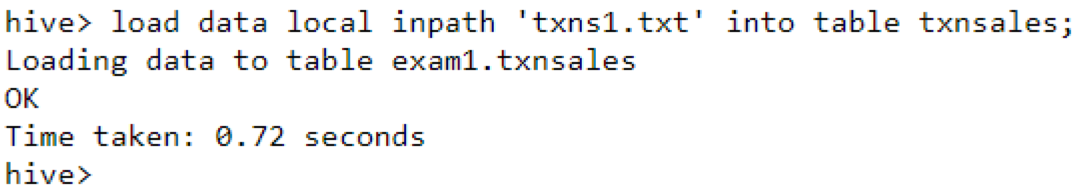
create table txnsales(txn_id bigint , txn_date string , cust_id bigint , amount double,category string,product string,city string ,state string ,spendby string) row format delimited

fields terminated by ','

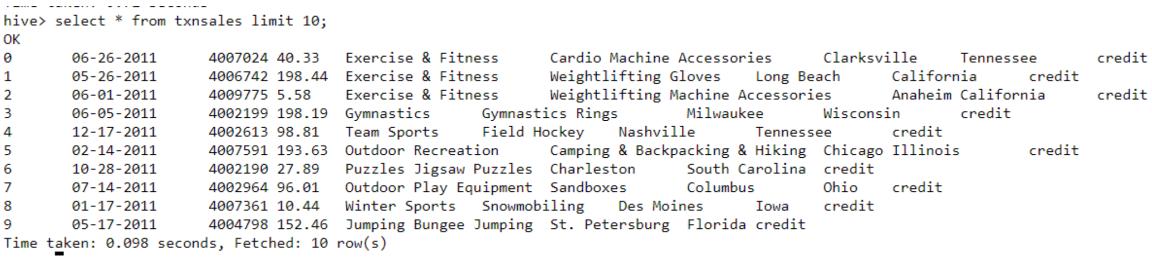
stored as textfile;



```
hive> create table txnsales(txn_id bigint , txn_date string , cust_id bigint , amount double,category string,product string,city string ,state string
 ,spendby string)
    >
    > row format delimited
    >
    > fields terminated by ','
    >
    > stored as textfile;
OK
Time taken: 0.096 seconds
```



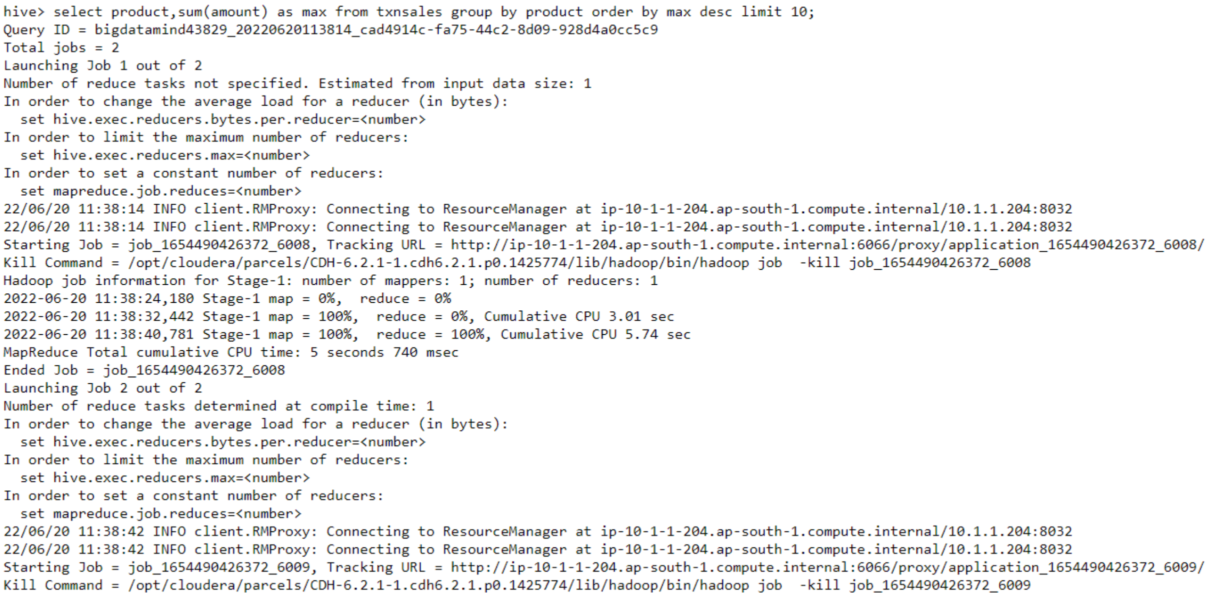
```
hive> load data local inpath 'txns1.txt' into table txnsales;
Loading data to table exam1.txnsales
OK
Time taken: 0.72 seconds
hive>
```



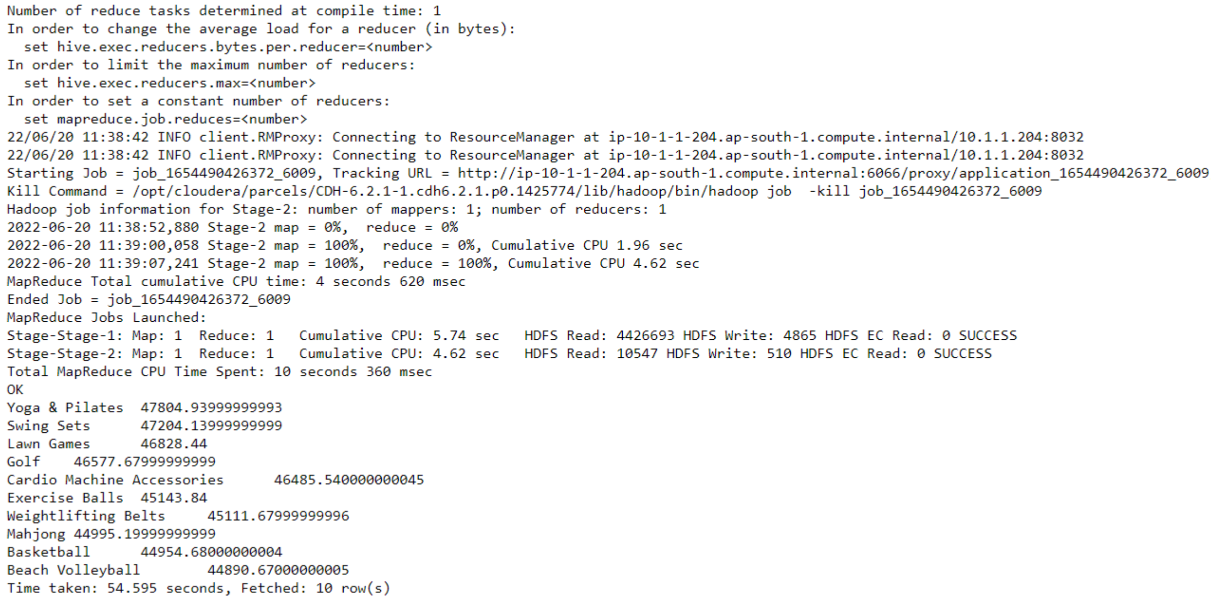
```
hive> select * from txnsales limit 10;
OK
0       06-26-2011      4007024 40.33   Exercise & Fitness      Cardio Machine Accessories      Clarksville     Tennessee       credit
1       05-26-2011      4006742 198.44  Exercise & Fitness      Weightlifting Gloves    Long Beach      California      credit
2       06-01-2011      4009775 5.58    Exercise & Fitness      Weightlifting Machine Accessories       Anaheim California      credit
3       06-05-2011      4002199 198.19  Gymnastics      Gymnastics Rings        Milwaukee       Wisconsin       credit
4       12-17-2011      4002613 98.81   Team Sports     Field Hockey    Nashville       Tennessee       credit
5       02-14-2011      4007591 193.63  Outdoor Recreation      Camping & Backpacking & Hiking  Chicago Illinois        credit
6       10-28-2011      4002190 27.89   Puzzles Jigsaw Puzzles  Charleston      South Carolina  credit
7       07-14-2011      4002964 96.01   Outdoor Play Equipment  Sandboxes       Columbus        Ohio    credit
8       01-17-2011      4007361 10.44   Winter Sports   Snowmobiling    Des Moines      Iowa    credit
9       05-17-2011      4004798 152.46  Jumping Bungee Jumping  St. Petersburg  Florida credit
Time taken: 0.098 seconds, Fetched: 10 row(s)
```

2) Write a program to find the top 10 products sales wise

select product,sum(amount) as max from txnsales group by product order by max desc limit 10;



```
hive> select product,sum(amount) as max from txnsales group by product order by max desc limit 10;
Query ID = bigdatamind43829_20220620113814_cad4914c-fa75-44c2-8d09-928d4a0cc5c9
Total jobs = 2
Launching Job 1 out of 2
Number of reduce tasks not specified. Estimated from input data size: 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>
22/06/20 11:38:14 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
22/06/20 11:38:14 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
Starting Job = job_1654490426372_6008, Tracking URL = http://ip-10-1-1-204.ap-south-1.compute.internal:6066/proxy/application_1654490426372_6008/
Kill Command = /opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/bin/hadoop job  -kill job_1654490426372_6008
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-06-20 11:38:24,180 Stage-1 map = 0%,  reduce = 0%
2022-06-20 11:38:32,442 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 3.01 sec
2022-06-20 11:38:40,781 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 5.74 sec
MapReduce Total cumulative CPU time: 5 seconds 740 msec
Ended Job = job_1654490426372_6008
Launching Job 2 out of 2
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>
22/06/20 11:38:42 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
22/06/20 11:38:42 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
Starting Job = job_1654490426372_6009, Tracking URL = http://ip-10-1-1-204.ap-south-1.compute.internal:6066/proxy/application_1654490426372_6009/
Kill Command = /opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/bin/hadoop job  -kill job_1654490426372_6009
```



```
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>
22/06/20 11:38:42 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
22/06/20 11:38:42 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
Starting Job = job_1654490426372_6009, Tracking URL = http://ip-10-1-1-204.ap-south-1.compute.internal:6066/proxy/application_1654490426372_6009
Kill Command = /opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/bin/hadoop job  -kill job_1654490426372_6009
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2022-06-20 11:38:52,880 Stage-2 map = 0%,  reduce = 0%
2022-06-20 11:39:00,058 Stage-2 map = 100%,  reduce = 0%, Cumulative CPU 1.96 sec
2022-06-20 11:39:07,241 Stage-2 map = 100%,  reduce = 100%, Cumulative CPU 4.62 sec
MapReduce Total cumulative CPU time: 4 seconds 620 msec
Ended Job = job_1654490426372_6009
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 5.74 sec   HDFS Read: 4426693 HDFS Write: 4865 HDFS EC Read: 0 SUCCESS
Stage-Stage-2: Map: 1  Reduce: 1   Cumulative CPU: 4.62 sec   HDFS Read: 10547 HDFS Write: 510 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 10 seconds 360 msec
OK
Yoga & Pilates  47804.93999999993
Swing Sets      47204.13999999999
Lawn Games      46828.44
Golf    46577.67999999999
Cardio Machine Accessories      46485.540000000045
Exercise Balls  45143.84
Weightlifting Belts     45111.67999999996
Mahjong 44995.19999999999
Basketball      44954.68000000004
Beach Volleyball        44890.67000000005
Time taken: 54.595 seconds, Fetched: 10 row(s)
```

3) Write a program to create partiioned table on category

set hive.exec.dynamic.partition.mode=nonstrict;

set hive.exec.dynamic.partition=true;

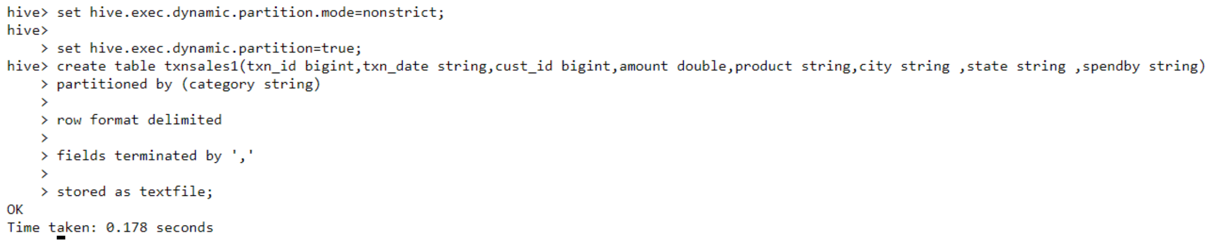
create table txnsales1(txn_id bigint,txn_date string,cust_id bigint,amount double,product string,city string ,state string ,spendby string)

partitioned by (category string)

row format delimited

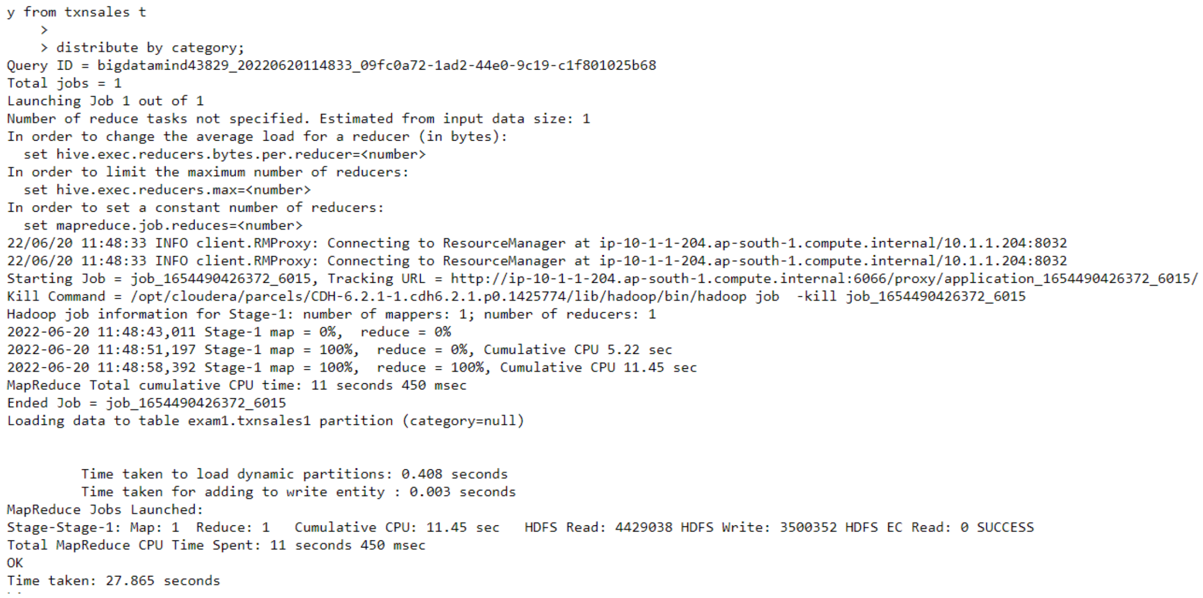
fields terminated by ','

stored as textfile;

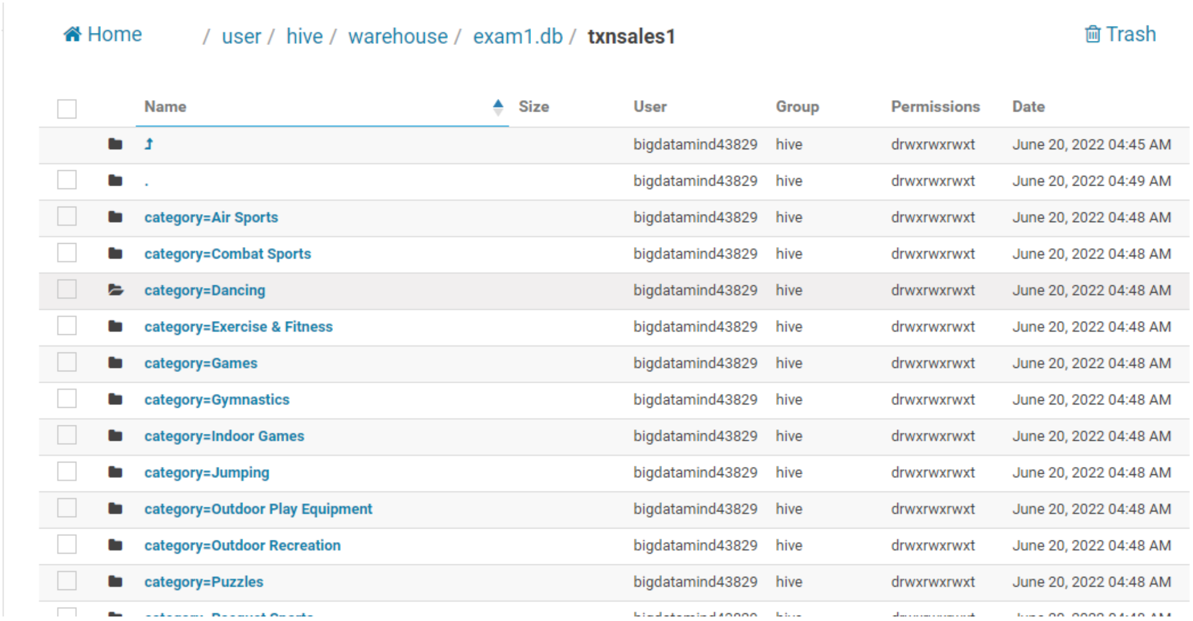


```
hive> set hive.exec.dynamic.partition.mode=nonstrict;
hive>
    > set hive.exec.dynamic.partition=true;
hive> create table txnsales1(txn_id bigint,txn_date string,cust_id bigint,amount double,product string,city string ,state string ,spendby string)
    > partitioned by (category string)
    >
    > row format delimited
    >
    > fields terminated by ','
    >
    > stored as textfile;
OK
Time taken: 0.178 seconds
```

insert overwrite table txnsales1 partition(category) select t.txn_id,t.txn_date,t.cust_id,t.amount,t.product,t.city,t.state,t.spendby,t.category from txnsales t distribute by category;



```
y from txnsales t
    >
    > distribute by category;
Query ID = bigdatamind43829_20220620114833_09fc0a72-1ad2-44e0-9c19-c1f801025b68
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks not specified. Estimated from input data size: 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>
22/06/20 11:48:33 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
22/06/20 11:48:33 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
Starting Job = job_1654490426372_6015, Tracking URL = http://ip-10-1-1-204.ap-south-1.compute.internal:6066/proxy/application_1654490426372_6015/
Kill Command = /opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/bin/hadoop job  -kill job_1654490426372_6015
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-06-20 11:48:43,011 Stage-1 map = 0%,  reduce = 0%
2022-06-20 11:48:51,197 Stage-1 map = 100%,  reduce = 0%, Cumulative CPU 5.22 sec
2022-06-20 11:48:58,392 Stage-1 map = 100%,  reduce = 100%, Cumulative CPU 11.45 sec
MapReduce Total cumulative CPU time: 11 seconds 450 msec
Ended Job = job_1654490426372_6015
Loading data to table exam1.txnsales1 partition (category=null)


         Time taken to load dynamic partitions: 0.408 seconds
         Time taken for adding to write entity : 0.003 seconds
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1  Reduce: 1   Cumulative CPU: 11.45 sec   HDFS Read: 4429038 HDFS Write: 3500352 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 11 seconds 450 msec
OK
Time taken: 27.865 seconds
```



/ user / hive / warehouse / exam1.db / txnsales1

□	Name	Size	User	Group	Permissions	Date
	↥		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:45 AM
□	.		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:49 AM
□	category=Air Sports		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Combat Sports		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Dancing		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Exercise & Fitness		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Games		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Gymnastics		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Indoor Games		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Jumping		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Outdoor Play Equipment		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Outdoor Recreation		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Puzzles		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM
□	category=Racquet Sports		bigdatamind43829	hive	drwxrwxrwxt	June 20, 2022 04:48 AM

PySpark

Please find the AIRLINES data set

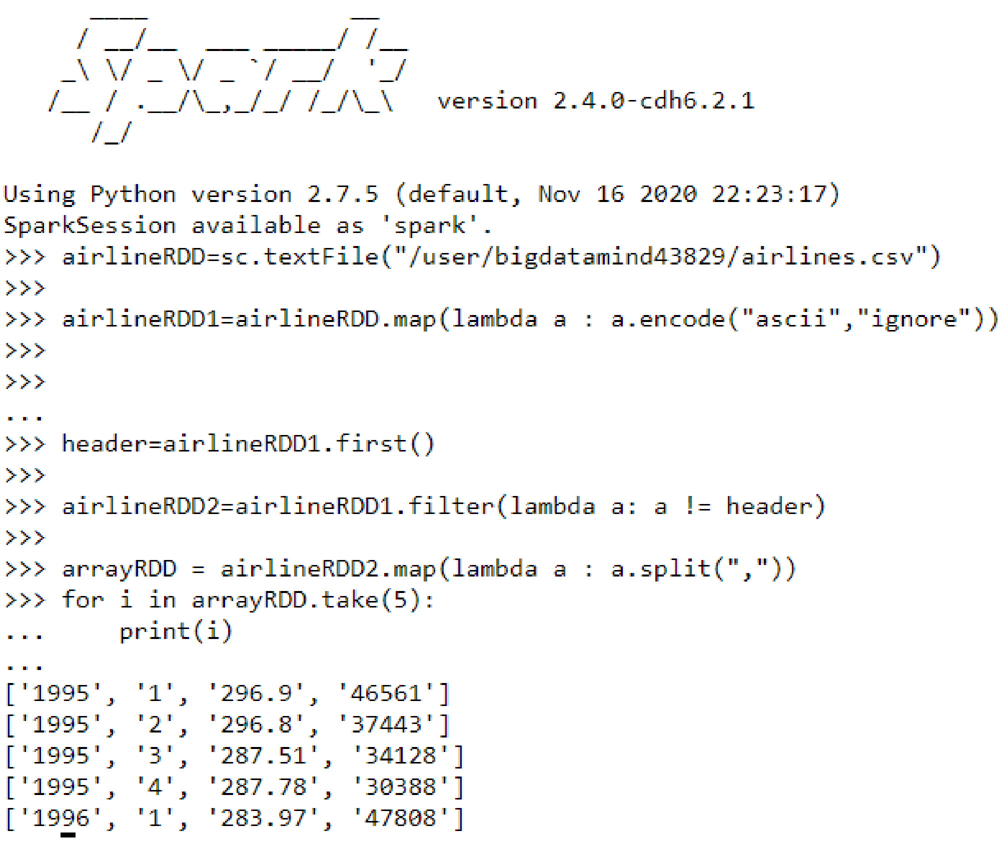
Year

Quarter

Average

revenue per seat

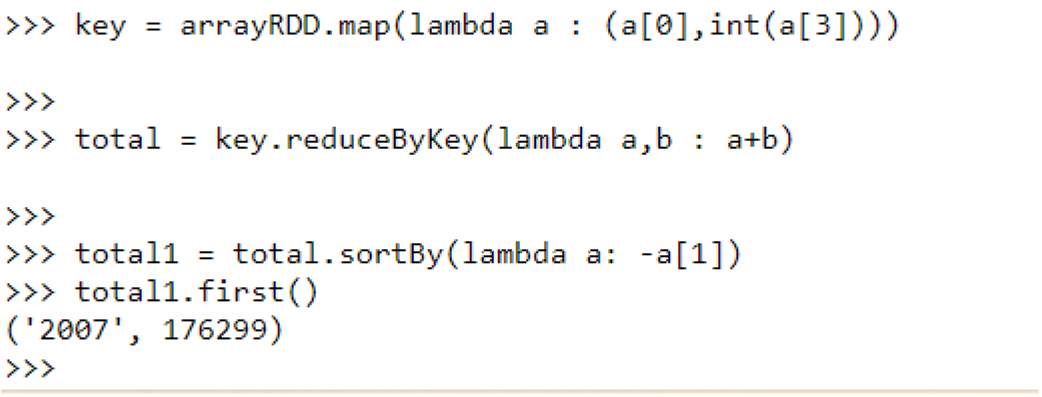
Total number of booked seats



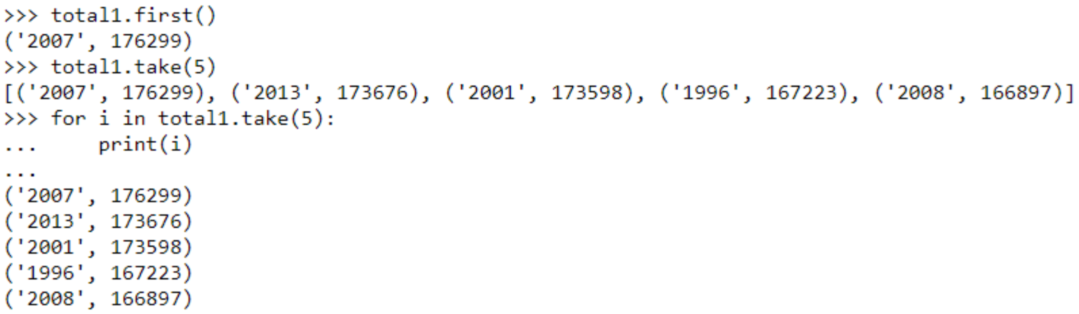
```
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /__ / .__/\_,_/_/ /_/\_\   version 2.4.0-cdh6.2.1
      /_/

Using Python version 2.7.5 (default, Nov 16 2020 22:23:17)
SparkSession available as 'spark'.
>>> airlineRDD=sc.textFile("/user/bigdatamind43829/airlines.csv")
>>>
>>> airlineRDD1=airlineRDD.map(lambda a : a.encode("ascii","ignore"))
>>>
>>>
...
>>> header=airlineRDD1.first()
>>>
>>> airlineRDD2=airlineRDD1.filter(lambda a: a != header)
>>>
>>> arrayRDD = airlineRDD2.map(lambda a : a.split(","))
>>> for i in arrayRDD.take(5):
...     print(i)
...
['1995', '1', '296.9', '46561']
['1995', '2', '296.8', '37443']
['1995', '3', '287.51', '34128']
['1995', '4', '287.78', '30388']
['1996', '1', '283.97', '47808']
```

1) What was the highest number of people travelled in which year?

```
key = arrayRDD.map(lambda a : (a[0],int(a[3])))

total = key.reduceByKey(lambda a,b : a+b)

total1 = total.sortBy(lambda a: -a[1])

total1.first()
```

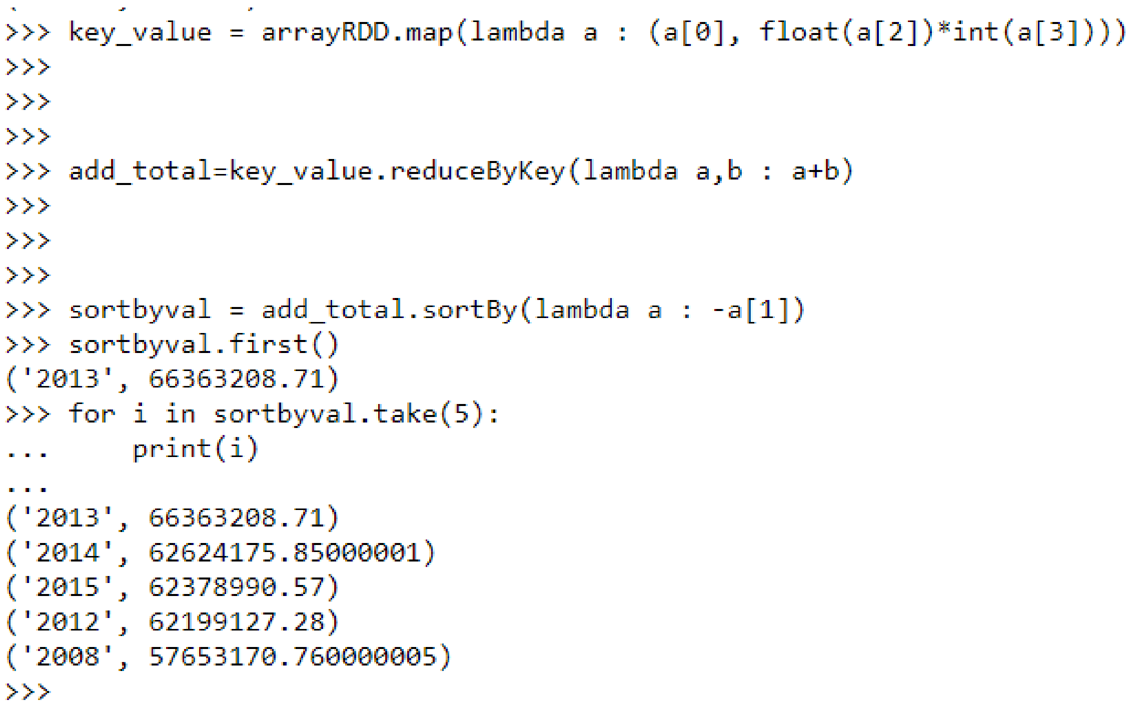


```
>>> key = arrayRDD.map(lambda a : (a[0],int(a[3])))

>>>
>>> total = key.reduceByKey(lambda a,b : a+b)

>>>
>>> total1 = total.sortBy(lambda a: -a[1])
>>> total1.first()
('2007', 176299)
>>>
```



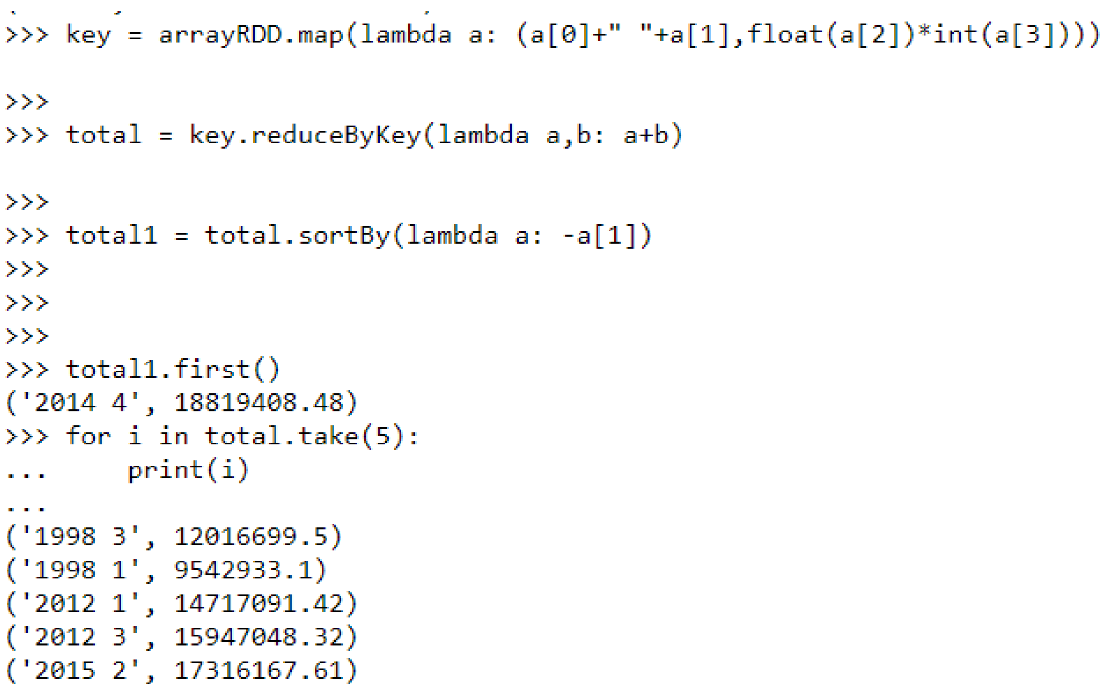
```
>>> total1.first()
('2007', 176299)
>>> total1.take(5)
[('2007', 176299), ('2013', 173676), ('2001', 173598), ('1996', 167223), ('2008', 166897)]
>>> for i in total1.take(5):
...     print(i)
...
('2007', 176299)
('2013', 173676)
('2001', 173598)
('1996', 167223)
('2008', 166897)
```

2) Identifying the highest revenue generation for which year



```
>>> key_value = arrayRDD.map(lambda a : (a[0], float(a[2])*int(a[3])))
>>>
>>>
>>>
>>> add_total=key_value.reduceByKey(lambda a,b : a+b)
>>>
>>>
>>>
>>> sortbyval = add_total.sortBy(lambda a : -a[1])
>>> sortbyval.first()
('2013', 66363208.71)
>>> for i in sortbyval.take(5):
...     print(i)
...
('2013', 66363208.71)
('2014', 62624175.85000001)
('2015', 62378990.57)
('2012', 62199127.28)
('2008', 57653170.760000005)
>>>
```

3) Identifying the highest revenue generation for which year and quarter (Common group)



```
>>> key = arrayRDD.map(lambda a: (a[0]+" "+a[1],float(a[2])*int(a[3])))

>>>
>>> total = key.reduceByKey(lambda a,b: a+b)

>>>
>>> total1 = total.sortBy(lambda a: -a[1])
>>>
>>>
>>>
>>> total1.first()
('2014 4', 18819408.48)
>>> for i in total.take(5):
...     print(i)
...
('1998 3', 12016699.5)
('1998 1', 9542933.1)
('2012 1', 14717091.42)
('2012 3', 15947048.32)
('2015 2', 17316167.61)
```