

Roll No. 220940325075

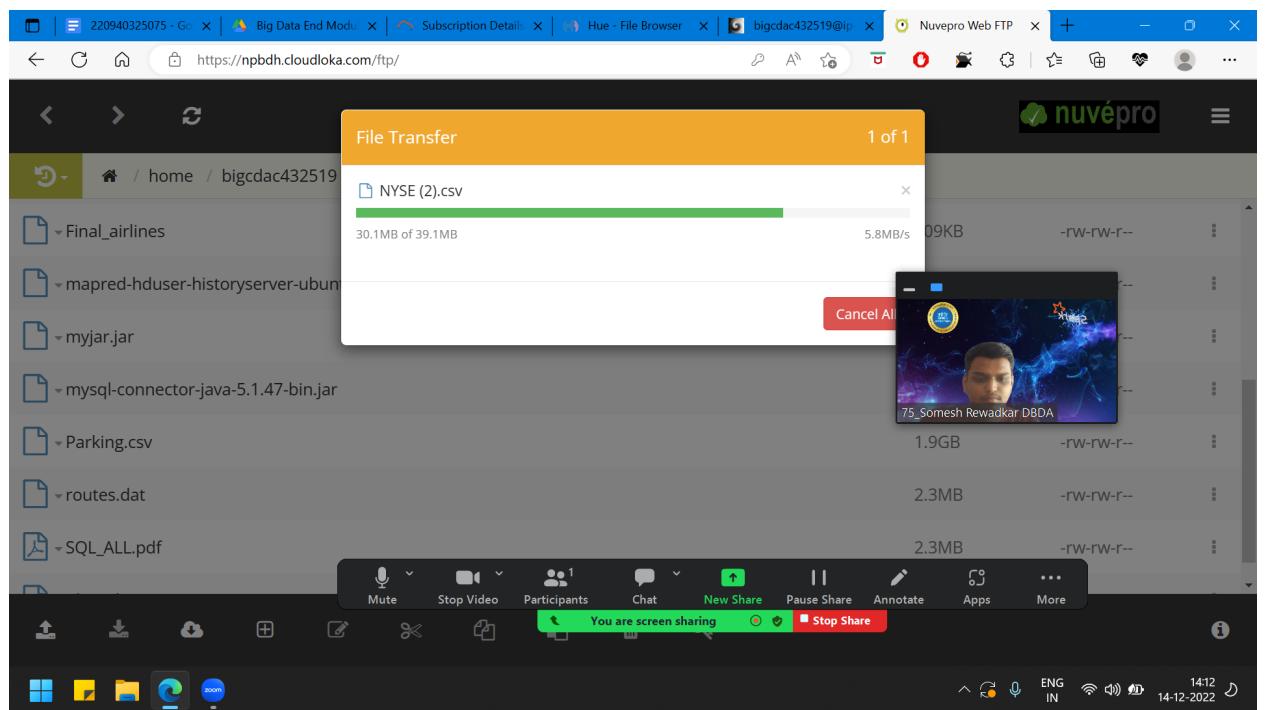
Q1. MapReduce

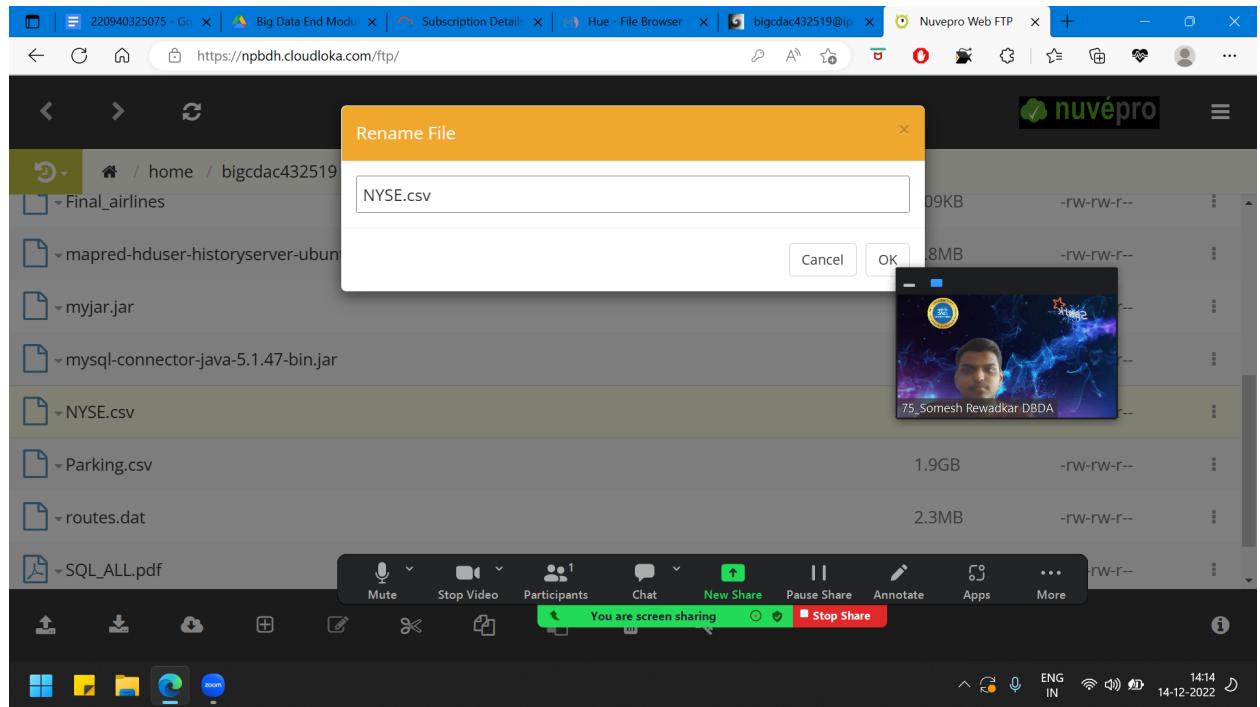
Step 1 : Upload data set to hadoop

Here file is uploading to FTP and renamed it to NYSE for convenience After that

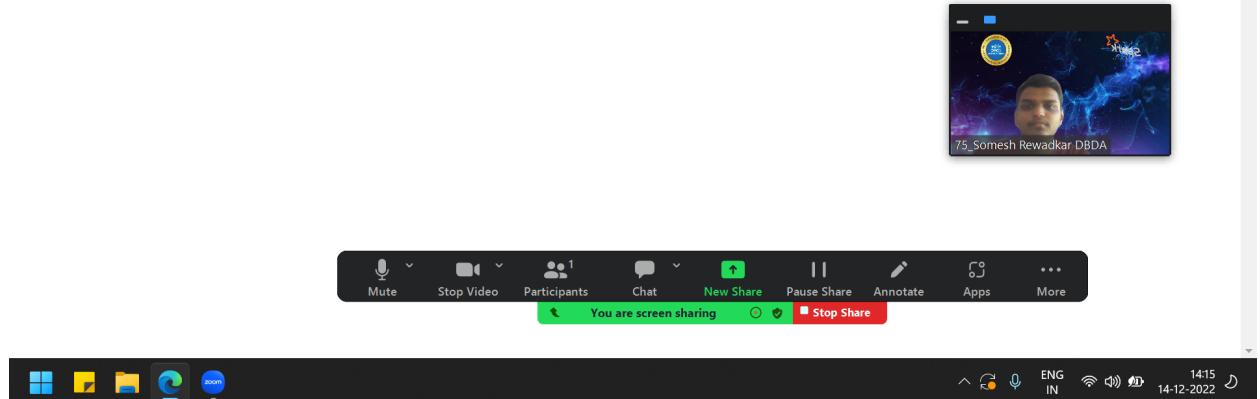
Below command is run in shell to upload it on hadoop

```
hadoop fs -put NYSE.csv training
```



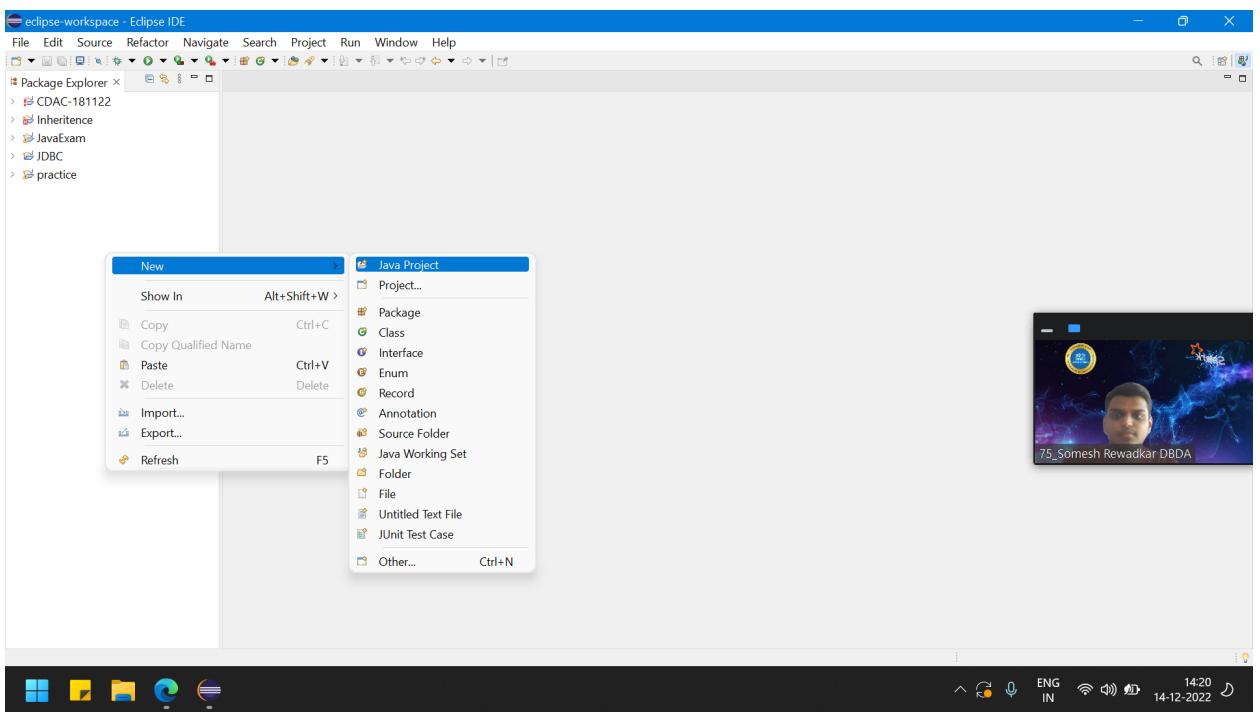


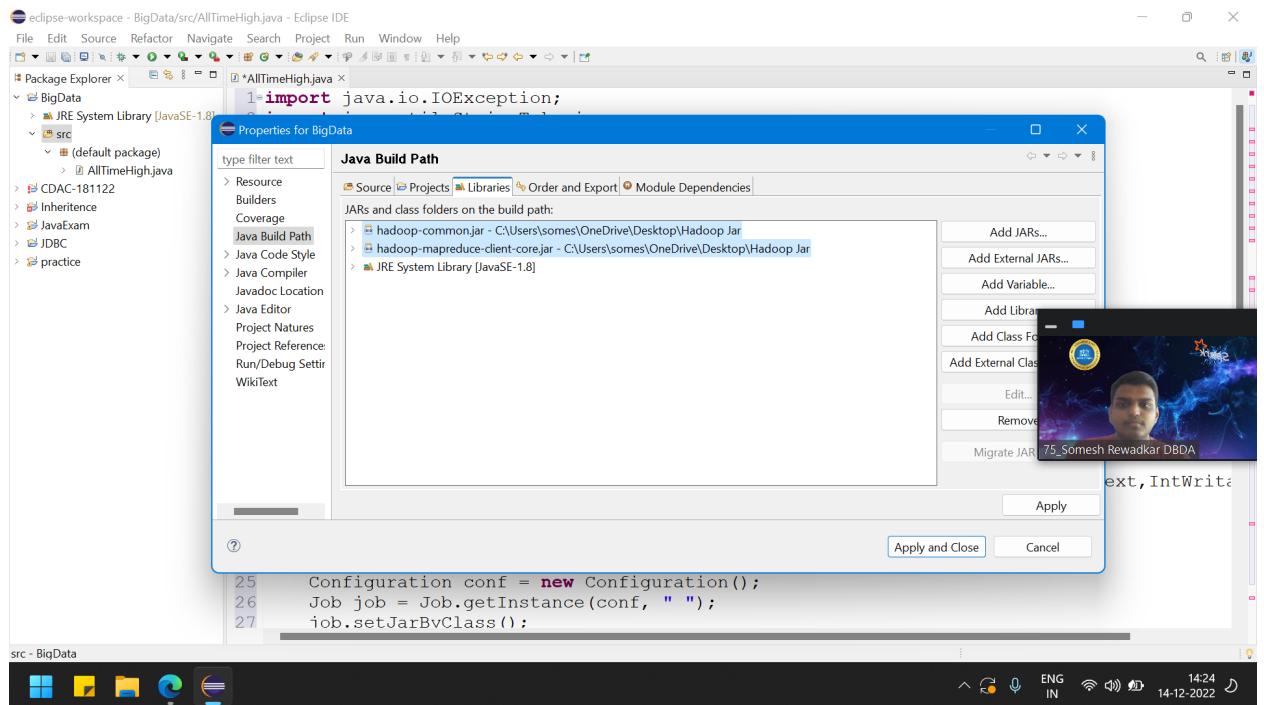
```
npbdh login: bigcdac432519
bigcdac432519@npbdh.cloudloka.com's password:
Last login: Wed Dec 14 06:07:04 2022 from ec2-65-1-45-35.ap-south-1.compute.amazonaws.com
[bigcdac432519@ip-10-1-1-204 ~]$ hadoop fs -put NYSE.csv training
-bash: hadoop: command not found
[bigcdac432519@ip-10-1-1-204 ~]$ hadoop fs -put NYSE.csv training
-bash: hadoop: command not found
[bigcdac432519@ip-10-1-1-204 ~]$ hadoop fs -put NYSE.csv training
[bigcdac432519@ip-10-1-1-204 ~]$ 
```



npbdh login: bigcdac432519
 bigcdac432519@npbdh.cloudloka.com's password:
 Last login: Wed Dec 14 06:07:04 2022 from ec2-65-1-45-35.ap-south-1.compute.amazonaws.com
 [bigcdac432519@ip-10-1-1-204 ~]\$ hadoop fs -put NYSE.csv training
 -bash: hadoop: command not found
 [bigcdac432519@ip-10-1-1-204 ~]\$ hadoop fs -put NYSE.csv training
 -bash: hadoop: command not found
 [bigcdac432519@ip-10-1-1-204 ~]\$ hadoop fs -put NYSE.csv training
 [bigcdac432519@ip-10-1-1-204 ~]\$ hadoop fs -ls training
 Found 11 items
 -rw-r--r-- 3 bigcdac432519 bigcdac432519 10585358 2022-11-23 06:08 training/AirLineData.csv
 -rw-r--r-- 3 bigcdac432519 bigcdac432519 40990862 2022-12-14 08:45 training/NYSE.csv
 -rw-r--r-- 3 bigcdac432519 bigcdac432519 1821 2022-12-08 05:35 training/airlines.csv
 -rw-r--r-- 3 bigcdac432519 bigcdac432519 207106008 2022-11-19 19:37 training/eclipse.gz
 -rw-r--r-- 3 bigcdac432519 bigcdac432519 1888276 2022-12-07 08:31 training/mapred-hduser-historyserver-ubuntu.log
 drwxr-xr-x - bigcdac432519 bigcdac432519 0 2022-11-22 11:16 training/out1
 drwxr-xr-x - bigcdac432519 bigcdac432519 0 2022-12-10 09:08 training/out2
 drwxr-xr-x - bigcdac432519 bigcdac432519 0 2022-11-23 06:25 training/out3
 drwxr-xr-x - bigcdac432519 bigcdac432519 0 2022-12-08 05:49 training/spark2
 -rw-r--r-- 3 bigcdac432519 bigcdac432519 4418144 2022-12-07 08:43 training/txts1.txt
 -rw-r--r-- 3 bigcdac432519 bigcdac432519 9511 2022-12-07 09:05 training/uber_data
 [bigcdac432519@ip-10-1-1-204 ~]\$

Step 2 : In eclipse java project is created with MapReduce code





Step 3: MapReduce code for finding out all time high price of each stock is written

```

import java.io.IOException;
import java.util.StringTokenizer;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.fs.Path;
import org.apache.hadoop.io.DoubleWritable;
import org.apache.hadoop.io.IntWritable;
import org.apache.hadoop.io.LongWritable;
import org.apache.hadoop.io.Text;
import org.apache.hadoop.mapreduce.Job;
import org.apache.hadoop.mapreduce.Mapper;
import org.apache.hadoop.mapreduce.Reducer;
import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;

public class AllTimeHigh {

```

```

    public static class MapClass extends Mapper<LongWritable, Text, Text,
DoubleWritable>
    {
        try
        {
            String [] str = value.toString().split(",");
            double high = Double.parseDouble(str[4]);
            context.write(new Text str([1]),new DoubleWritable(high));
        }
        catch(Exception e)
        {
            System.out.println(e.getMessage());
        }
    }

    public static class ReduceClass extends
Reducer<Text,IntWritable,Text,IntWritable>
{
    private DoubleWritable result = new DoubleWritable();
    public void reduce(Text key, Iterable<DoubleWritable> values,Context context)
throws IOException, InterruptedException
    {
        double max=0.00;

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

    public static void main(String[] args) throws Exception {
        Configuration conf = new Configuration();
        Job job = Job.getInstance(conf, "All time High price of each stock ");
        job.setJarByClass(AllTimeHigh.class);
    }
}

```

```

job.setMapperClass(MapClass.class);
job.setReducerClass(ReduceClass.class);

job.setNumReduceTasks(2);
//job.setMapOutputKeyClass(Text.class);
//job.setMapOutputValueClass(DoubleWritable.class);
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);
}

```

eclipse-workspace - BigData/src/AllTimeHigh.java - Eclipse IDE

File Edit Source Refactor Navigate Project Run Window Help

Package Explorer X

- BigData
 - JRE System Library [JavaSE-1.8]
 - src
 - (default package)
 - AllTimeHigh.java
 - Referenced Libraries
 - CDAC-181122
 - Inheritance
 - JavaExam
 - JDBC
 - practice

AllTimeHigh.java x

```

1 import java.io.IOException;
2 import java.util.StringTokenizer;
3 import org.apache.hadoop.conf.Configuration;
4 import org.apache.hadoop.fs.Path;
5 import org.apache.hadoop.io.DoubleWritable;
6 import org.apache.hadoop.io.IntWritable;
7 import org.apache.hadoop.io.LongWritable;
8 import org.apache.hadoop.io.Text;
9 import org.apache.hadoop.mapreduce.Job;
10 import org.apache.hadoop.mapreduce.Mapper;
11 import org.apache.hadoop.mapreduce.Reducer;
12 import org.apache.hadoop.mapreduce.lib.input.FileInputFormat;
13 import org.apache.hadoop.mapreduce.lib.output.FileOutputFormat;
14
15 public class AllTimeHigh {
16     public static class MapClass extends Mapper<LongWritable, Text, Text, IntWritable> {
17         private Text stock_id = new Text();
18         private DoubleWritable High = new DoubleWritable();
19
20         try {
21             String [] str = value.toString().split(",");
22             double high = Double.parseDouble(str[4]);
23             stock_id.set(str[1]);
24             High.set(high);
25             context.write(stock_id, High);
26         }
27         catch(Exception e) {
28             System.out.println(e.getMessage());
29         }
30     }
31 }

```

Writable Smart Insert 63:20:2120

ENG IN 14-12-2022 14:55

eclipse-workspace - BigData/src/AllTimeHigh.java - Eclipse IDE

```
File Edit Source Refactor Navigate Project Run Window Help
Package Explorer X AllTimeHigh.java X
BigData
  JRE System Library [JavaSE-1.8]
  src
    (default package)
      AllTimeHigh.java
  Referenced Libraries
  CDAC-181122
  Inheritance
  JavaExam
  JDBC
  practice

  23      context.write(stock_id, high);
  24  }
  25  catch(Exception e)
  26  {
  27      System.out.println(e.getMessage());
  28  }
  29
  30
  31
  32  }
  33
  34  public static class ReduceClass extends Reducer<Text,IntWritable,Text,IntWritable>
  35  {
  36      private DoubleWritable result = new DoubleWritable();
  37      public void reduce(Text key, Iterable<DoubleWritable> values, Context context) throws IOException Inte
  38
  39      double maxValue=0.00;
  40      double temp_val=0.00;
  41
  42      for (DoubleWritable value: values) {
  43          temp_val = value.get();
  44          if (temp_val > maxValue)
  45          {
  46              maxValue = temp_val;
  47          }
  48          result.set(maxValue);
  49          context.write(key, result);
  50      }
  51  }
  52
  53  public static void main(String[] args) throws Exception {
  54      Configuration conf = new Configuration();
```

Writable Smart Insert 63:20:2120

ENG IN 14:55 14-12-2022

eclipse-workspace - BigData/src/AllTimeHigh.java - Eclipse IDE

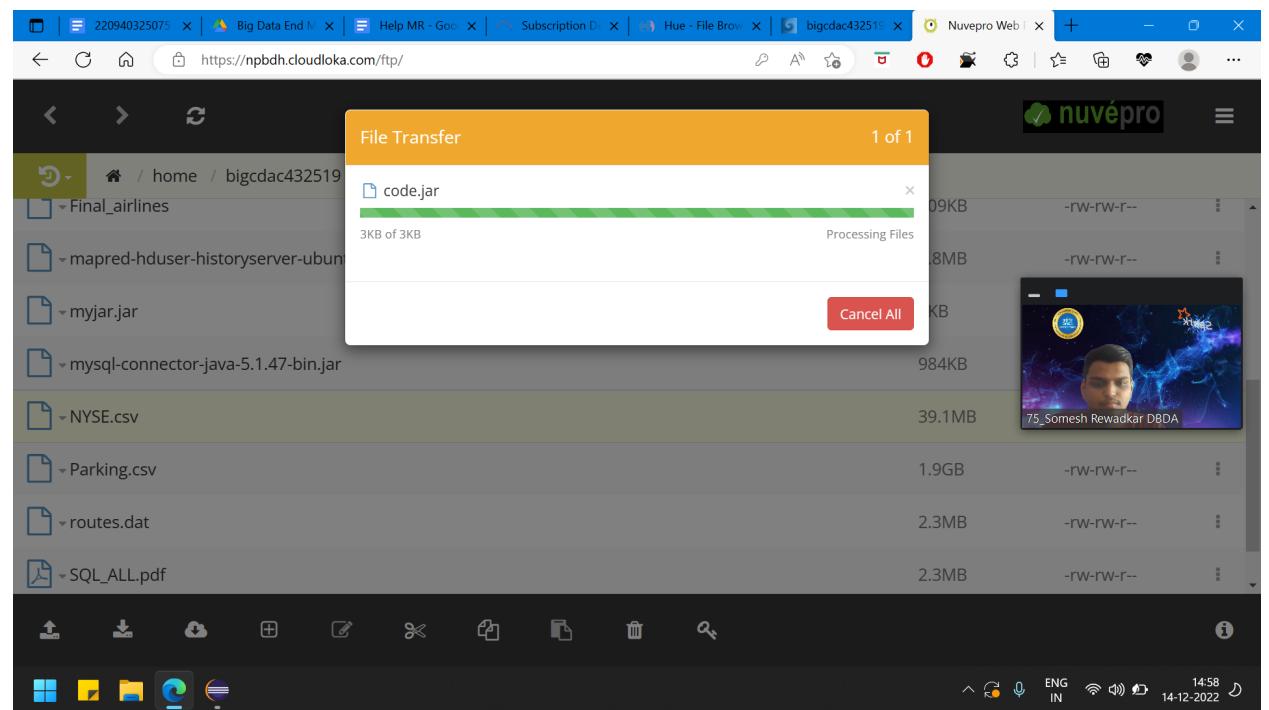
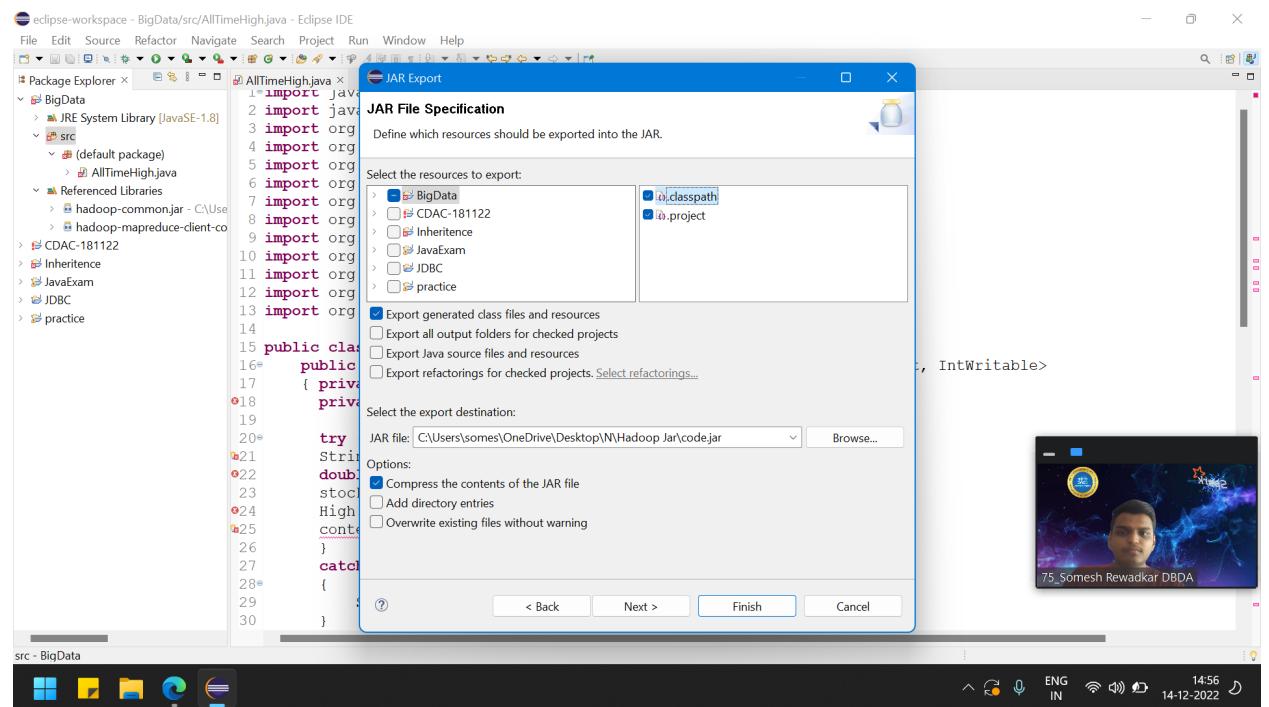
```
File Edit Source Refactor Navigate Project Run Window Help
Package Explorer X AllTimeHigh.java X
BigData
  JRE System Library [JavaSE-1.8]
  src
    (default package)
      AllTimeHigh.java
  Referenced Libraries
  CDAC-181122
  Inheritance
  JavaExam
  JDBC
  practice

  40      double temp_val=0.00;
  41
  42      for (DoubleWritable value: values) {
  43          temp_val = value.get();
  44          if (temp_val > maxValue)
  45          {
  46              maxValue = temp_val;
  47          }
  48          result.set(maxValue);
  49          context.write(key, result);
  50      }
  51  }
  52
  53  public static void main(String[] args) throws Exception {
  54      Configuration conf = new Configuration();
  55      Job job = Job.getInstance(conf, "High price of each stock ");
  56      job.setJarByClass(AlltimeHigh.class);
  57      job.setMapperClass(MapClass.class);
  58      job.setReducerClass(ReduceClass.class);
  59
  60      job.setNumReduceTasks(1);
  61      //job.setMapOutputKeyClass(Text.class);
  62      //job.setMapOutputValueClass(DoubleWritable.class);
  63      job.setOutputKeyClass(Text.class);
  64      job.setOutputValueClass(DoubleWritable.class);
  65      FileInputFormat.addInputPath(job, new Path(args[0]));
  66      FileOutputFormat.setOutputPath(job, new Path(args[1]));
  67      System.exit(job.waitForCompletion(true) ? 0 : 1);
  68  }
  69
```

Writable Smart Insert 63:20:2120

ENG IN 14:54 14-12-2022

Step 4 : Jar file of the code is created and Jar file uploaded on FTP



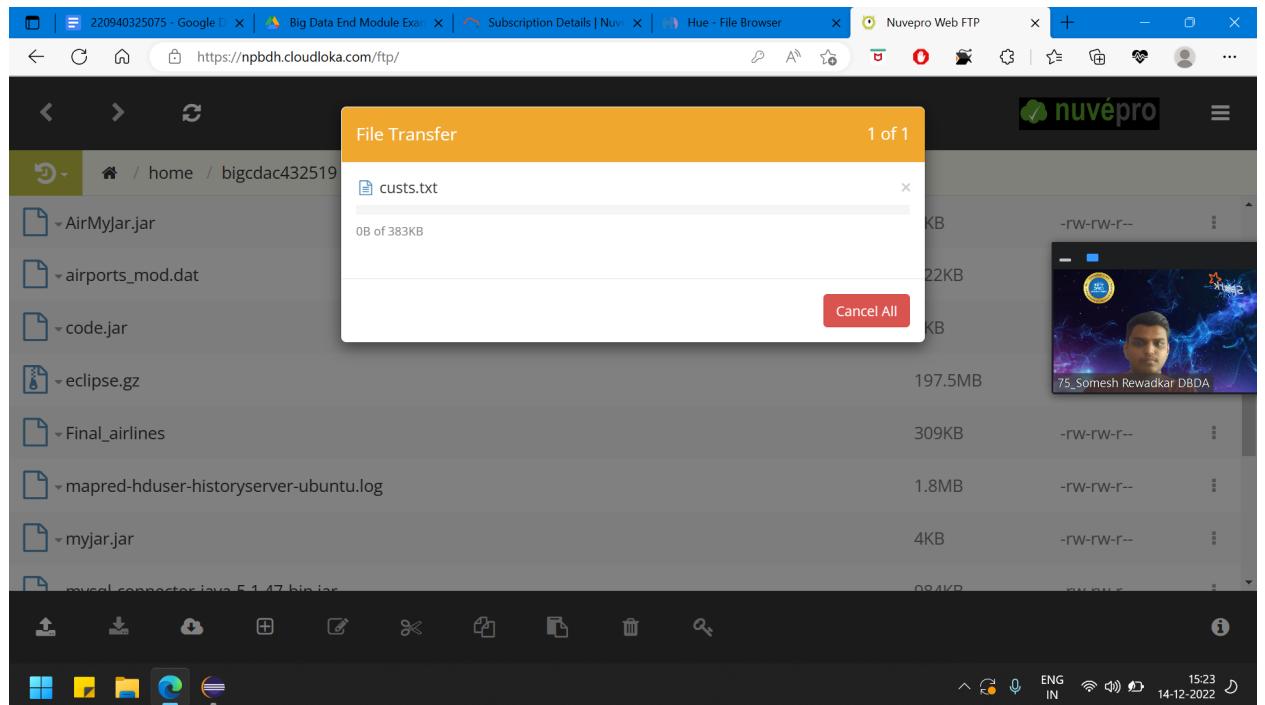
Q2. Hive

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

Query - SELECT count(cid), profession from customer group by profession;

Implementation →

Step 1 : uploading data set on FTP



Step 2: Creating Hive table And Mapping data set to that table

```
LOAD DATA LOCAL INPATH 'custs.txt' overwrite into table  
customer;
```

The screenshot shows a Windows desktop environment. In the center is a terminal window for Apache Hive. The command entered is:

```
hive> create table customer(cid int,FirstName String,LastName String,Age int,Profession String) row format delimited fields terminated  
> by ',' stored as textfile;
```

The terminal window also displays the following output:

```
OK  
Time taken: 0.451 seconds  
hive> show tables;  
OK  
customer  
parking  
stkvol  
Time taken: 0.195 seconds, Fetched: 3 row(s)
```

At the bottom of the terminal window, there is a standard Windows taskbar with icons for File Explorer, Task View, Start, and others.

```
220940325075 - Go X | Big Data End Modu... X | Subscription Details X | bigcdac432519@ip... X | Hue - File Browser X | Nuvepro Web FTP X | + - □ ×
← ↻ ⌂ https://npbdh.cloudloka.com:4200
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at org.apache.hadoop.util.RunJar.run(RunJar.java:313)
at org.apache.hadoop.util.RunJar.main(RunJar.java:227)
FAILED: ParseException line 1:33 cannot recognize input near 'fields' 'terminated' 'by' in column type
hive> show tables
    >;
OK
parking
stkvol
Time taken: 0.08 seconds, Fetched: 2 row(s)
hive> create table customer(cid int,FirstName String,LastName String,Age int,Profession String) row format delimiter
    > by ',', stored as textfile;
OK
Time taken: 0.451 seconds
hive> show tables;
OK
customer
parking
stkvol
Time taken: 0.195 seconds, Fetched: 3 row(s)
hive> SELECT * FROM CUSTOMER LIMIT 4;
OK
Time taken: 0.538 seconds
hive> LOAD DATA LOCAL INPATH 'custs.txt' insert into table customer;
FAILED: ParseException line 1:35 extraneous input 'insert' expecting INTO near '<EOF>'
hive> LOAD DATA LOCAL INPATH 'custs.txt' overwirte into table customer;
Loading data to table training432519.customer
OK
Time taken: 1.354 seconds
hive> □
```



75_Somesh Rewadkar DBDA



15:34
ENG IN 14-12-2022

```
220940325075 - Go X | Big Data End Modu... X | Subscription Details X | bigcdac432519@ip... X | Hue - File Browser X | Nuvepro Web FTP X | + - □ ×
← ↻ ⌂ https://npbdh.cloudloka.com:4200
Time taken: 0.117 seconds, Fetched: 4 row(s)
hive> SELECT count(cid), profession from customer group by profession;
Query ID = bigcdac432519_20221214100703_a1920e29-a98f-4aee-853e-eeb8ff21a34c
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>
22/12/14 10:07:04 INFO client.RMProxy: Connecting to ResourceManager at ip-10-1-1-204.ap-south-1.compute.internal/10.1.1.204:8032
22/12/14 10:07:04 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_1663041244711_22919, Tracking URL = http://ip-10-1-1-204.ap-south-1.compute.internal:6066/proxy/application_1663041244711_22919/
Kill Command = /opt/cloudera/parcels/CDH-6.2.1-1.cdh6.2.1.p0.1425774/lib/hadoop/bin/hadoop job -kill job_1663041244711_22919
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2022-12-14 10:07:59,592 Stage-1 map = 0%, reduce = 0%
2022-12-14 10:08:25,979 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.66 sec
2022-12-14 10:09:22,030 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 6.64 sec
MapReduce Total cumulative CPU time: 6 seconds 640 msec
Ended Job = job_1663041244711_22919
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 6.64 sec HDFS Read: 400764 HDFS Write: 1584 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 640 msec
OK
199      Accountant
202      Actor
195      Agricultural and food scientist
203      Architect
175      Artist
```



75_Somesh Rewadkar DBDA



15:40
ENG IN 14-12-2022

```
220940325075 | Big Data End M... | Subscription D... | bigcdac432519 | bigcdac432519 | Hue - File Brow... | Nuvepro Web... | + - □ ×
https://npbdh.cloudloka.com:4200
2022-12-14 10:08:25,979 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.66 sec
2022-12-14 10:09:22,030 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 6.64 sec
MapReduce Total cumulative CPU time: 6 seconds 640 msec
Ended Job = job_1663041244711_22919
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 6.64 sec HDFS Read: 400764 HDFS Write: 1584 HDFS EC Read: 0 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 640 msec
OK
199 Accountant
202 Actor
195 Agricultural and food scientist
203 Architect
175 Artist
196 Athlete
193 Automotive mechanic
181 Carpenter
209 Chemist
207 Childcare worker
193 Civil engineer
201 Coach
204 Computer hardware engineer
216 Computer software engineer
222 Computer support specialist
185 Dancer
205 Designer
197 Doctor
189 Economist
192 Electrical engineer
194 Electrician
204 Engineering technician
176 Environmental scientist
201 Farmer
```



```
220940325075 | Big Data End M... | Subscription D... | bigcdac432519 | bigcdac432519 | Hue - File Brow... | Nuvepro Web... | + - □ ×
https://npbdh.cloudloka.com:4200
176 Environmental scientist
201 Farmer
198 Financial analyst
217 Firefighter
212 Human resources assistant
196 Judge
212 Lawyer
218 Librarian
221 Loan officer
205 Musician
192 Nurse
213 Pharmacist
222 Photographer
201 Physicist
211 Pilot
210 Police officer
228 Politician
194 Psychologist
191 Real estate agent
210 Recreation and fitness worker
200 Reporter
200 Secretary
1 Social Worker
212 Social worker
196 Statistician
204 Teacher
187 Therapist
208 Veterinarian
101 Writer
Time taken: 141.731 seconds, Fetched: 51 row(s)
hive> 
```



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

Query - SELECT PRODUCT, SUM(AMOUNT) AS MAX FROM txn group by product order by MAX DESC;

Implementation→

Step 1: Creating Hive table of txns1.txt and loading data

```
create table txn(txn_id bigint, txn_date  
date,cust_id String,amount double,category  
String, product String,city String, State  
String,spendby String) row format delimited  
fields terminated by ',' stored as textfile
```

```
LOAD DATA LOCAL INPATH 'txns1.txt' OVERWRITE  
INTO TABLE txn;
```

```
at org.apache.hadoop.hive.ql.Driver.compile(Driver.java:564)
at org.apache.hadoop.hive.ql.Driver.compileInternal(Driver.java:1425)
at org.apache.hadoop.hive.ql.Driver.runInternal(Driver.java:1493)
at org.apache.hadoop.hive.ql.Driver.run(Driver.java:1339)
at org.apache.hadoop.hive.ql.Driver.run(Driver.java:1328)
at org.apache.hadoop.hive.cli.CliDriver.processLocalCmd(CliDriver.java:239)
at org.apache.hadoop.hive.cli.CliDriver.processCmd(CliDriver.java:187)
at org.apache.hadoop.hive.cli.CliDriver.processLine(CliDriver.java:409)
at org.apache.hadoop.hive.cli.CliDriver.executeDriver(CliDriver.java:836)
at org.apache.hadoop.hive.cli.CliDriver.run(CliDriver.java:772)
at org.apache.hadoop.hive.cli.CliDriver.main(CliDriver.java:699)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at org.apache.hadoop.util.RunJar.run(RunJar.java:313)
at org.apache.hadoop.util.RunJar.main(RunJar.java:227)
FAILED: ParseException line 1:40 cannot recognize input near ',' 'cust_id' 'String' in column type
hive> create table txn(txn_id bigint, txn_date String,cust_id String,amount double,category String, product String,city String, State String
,spendby String) row format delimited fields terminated by ',' stored as textfile
> ;
OK
Time taken: 0.19 seconds
hive> show tables;
OK
customer
parking
stkvol
txn
Time taken: 0.041 seconds, Fetched: 4 row(s)
hive> 
```

15:49
ENG IN 14-12-2022

```
at org.apache.hadoop.hive.ql.Driver.run(Driver.java:1328)
at org.apache.hadoop.hive.cli.CliDriver.processLocalCmd(CliDriver.java:239)
at org.apache.hadoop.hive.cli.CliDriver.processCmd(CliDriver.java:187)
at org.apache.hadoop.hive.cli.CliDriver.processLine(CliDriver.java:409)
at org.apache.hadoop.hive.cli.CliDriver.executeDriver(CliDriver.java:836)
at org.apache.hadoop.hive.cli.CliDriver.run(CliDriver.java:772)
at org.apache.hadoop.hive.cli.CliDriver.main(CliDriver.java:699)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at org.apache.hadoop.util.RunJar.run(RunJar.java:313)
at org.apache.hadoop.util.RunJar.main(RunJar.java:227)
FAILED: ParseException line 1:40 cannot recognize input near ',' 'cust_id' 'String' in column type
hive> create table txn(txn_id bigint, txn_date String,cust_id String,amount double,category String, product String,city String, State String
,spendby String) row format delimited fields terminated by ',' stored as textfile
> ;
OK
Time taken: 0.19 seconds
hive> show tables;
OK
customer
parking
stkvol
txn
Time taken: 0.041 seconds, Fetched: 4 row(s)
hive> LOAD DATA LOCAL INPATH 'txns1.txt' OVERWRITE INTO TABLE txn;
Loading data to table training432519.txn
OK
Time taken: 1.056 seconds
hive> 
```

15:52
ENG IN 14-12-2022

```
220940325075 - Go | Big Data End Modu | Subscription Detail | bigcdac432519@ip | Hue - File Browser | Nuvepro Web FTP | +  
https://npbdh.cloudloka.com:4200  
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 8.43 sec HDFS Read: 4426694 HDFS Write: 4865 HDFS EC Read: 0 SUCCESS  
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 6.34 sec HDFS Read: 10388 HDFS Write: 5333 HDFS EC Read: 0 SUCCESS  
Total MapReduce CPU Time Spent: 14 seconds 770 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 44954.6800000004  
Beach Volleyball 44890.6700000005  
Badminton 44786.18999999997  
Boxing 44516.86999999966  
Stopwatches 44443.52  
Hockey 44144.75000000015  
Balance Beams 44052.89999999999  
Rugby 43752.18999999996  
Water Polo 43577.83  
Cross-Country Skiing 43562.2300000002  
Swimming 43486.88999999985  
Weight Benches 43473.69  
Deck Shuffleboard 43440.5200000003  
Table Shuffleboard 43405.14999999994  
Abdominal Equipment 43304.10999999986  
Darts 43243.41999999999  
Gymnastics Mats 43224.54999999999  
Bobsledding 43157.45999999999  
Foosball 43055.95999999997  
Boating 43049.06999999934
```

```

Running 39169.11
Soccer 39094.65
Hunting 39081.74
Wetsuits 38980.34000000004
Bungee Jumping 38975.59000000004
Pommel Horses 38899.8999999995
Hang Gliding 38758.90999999996
Track & Field 38687.81000000002
Billiards 38614.52000000004
Sledding 38509.56000000002
Rock Climbing 38451.42999999964
Weightlifting Gloves 38438.72
Equestrian 38381.93999999995
Curling 38207.25
Team Handball 38121.0499999999
Snowboarding 38064.8099999998
Springboards 37890.76000000001
Baseball 37843.8199999998
Kitesurfing 37730.8899999995
Exercise Bands 37679.75000000015
Fishing 37144.6099999998
Archery 37088.65999999945
Cricket 37061.5800000001
Downhill Skiing 36870.3599999996
Tetherball 35611.9299999999
Surfing 35069.07999999965
Disc Golf 33944.0200000002
Mechanical Puzzles 20381.66999999984
Air Suits 19371.57000000007
Time taken: 115.691 seconds, Fetched: 125 row(s)
hive>

```

Windows taskbar icons: File Explorer, File Manager, FileZilla, and others.

System tray icons: ENG IN, battery level, signal strength, 14-12-2022, 15:59.

3) Write a program to create partitioned table on category

Query -

```

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

```

```

create table txnPartition(txn_id bigint, txn_date
date,cust_id String,amount double,product String,city
String, State String,spendby String)
Partitioned by (category String)
row format delimited fields terminated by ','
stored as textfile;

```

```
220940325075 - Go X | Big Data End Modu... X | Subscription Detail... X | bigcdac432519@ip... X | Hue - File Browser X | Nuvepro Web FTP X | + - □ ×
https://npbdh.cloudloka.com:4200
Archer 37088.659999999945
Cricket 37061.5800000001
Downhill Skiing 36870.3599999996
Tetherball 35611.9299999999
Surfing 35069.07999999965
Disc Golf 33944.0200000002
Mechanical Puzzles 20381.66999999984
Air Suits 19371.570000000007
Time taken: 115.691 seconds, Fetched: 125 row(s)
hive>
hive> set hive.exec.dynamic.partition=true;
hive> create table txn(txn_id bigint, txn_date date,cust_id String,amount double,product String,city String, State String)
> Partitioned by (category String)
> row format delimited
> fields terminated by ','
> stored as textfile;
FAILED: Execution Error, return code 1 from org.apache.hadoop.hive.ql.exec.DDLTask. AlreadyExistsException(message:Table txn already exists)
hive> create table txnPartition(txn_id bigint, txn_date date,cust_id String,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: 0.119 seconds
hive>
```



16:08
ENG
IN 14-12-2022

Uploading DATASET on FTP

The screenshot shows a web browser window displaying an FTP directory listing. The URL in the address bar is <https://npbdh.cloudloka.com/ftp/>. The page title is "nuvépro". The directory path shown is /home/bigcdac432519. The table lists the following files:

Name	Size	Permissions
..		
out2		
AirLineData.csv	10.1MB	75_Somesh Rewadkar DBDA
airlines.csv	2KB	-rW-rW-r--
AirMyJar.jar	4KB	-rW-rW-r--
airports_mod.dat	722KB	-rW-rW-r--
code.jar	4KB	-rW-rW-r--

Below the table are standard file management icons: upload, download, cloud, folder, file, edit, delete, and search. At the bottom right, there is a system tray showing the date and time (14-12-2022, 16:13), language (ENG IN), and connectivity status.