M-4, 1st Floor, Old DLF Colony, Sector 14, Gurugram, Haryana 122001

Project report on Big data

Name

Jairam J S

Email ID

jsjairam01@gmail.com

Acknowledgement

Not all professionals do their work by themselves. Although they can be as prolific or as adept in their respective fields, they will still need assistance one way or another. For instance, writing a body of work takes a lot of research. They often depend on their assistants or subordinates to gather information about the subject matter. Aside from the research people, other individuals can also receive credit for their contributions to the writer's work.

CONTENTS

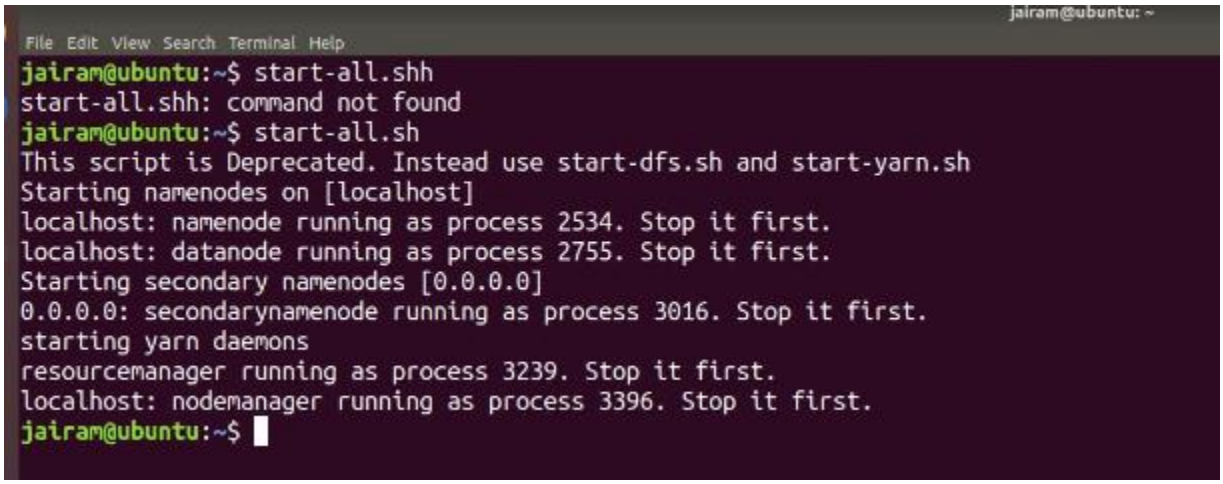
Titanic data analysis

Problem 1:

In this problem statement, we will find the average age of males and females who died in the Titanic tragedy.

1. Open terminal and type

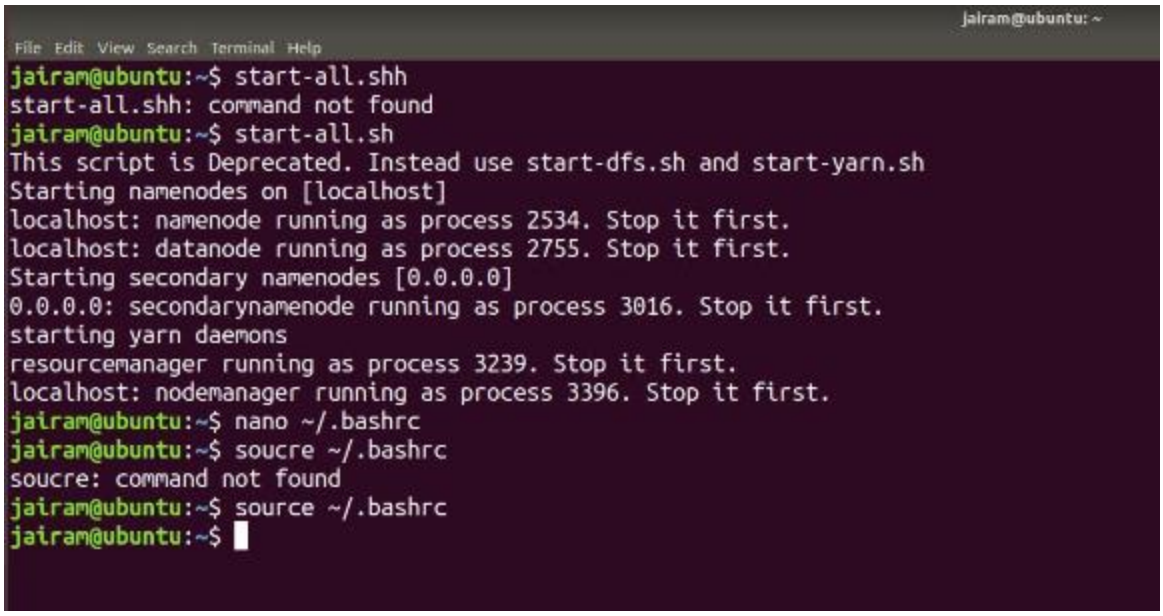
   Command: ***start-all.sh***



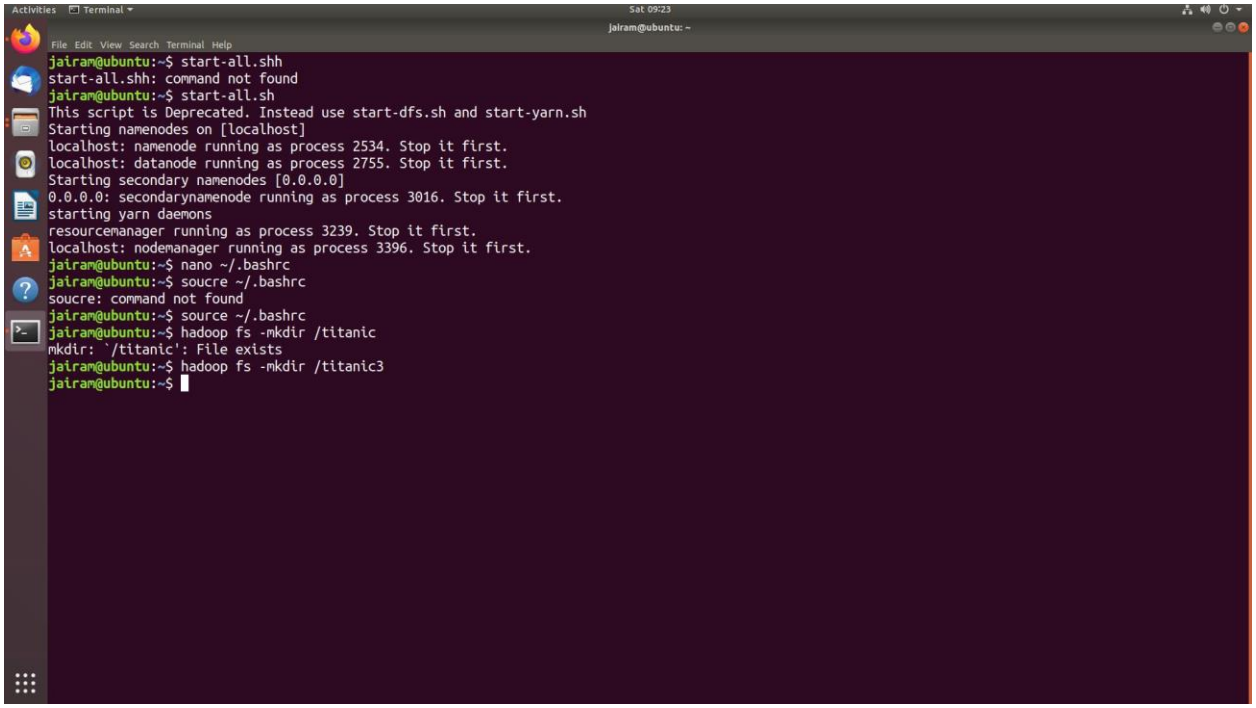
2. Open bashrc file by ***nano ~/.bashrc*** and type

   export HADOOP_CLASSPATH=$JAVA_HOME/lib/tools.jar

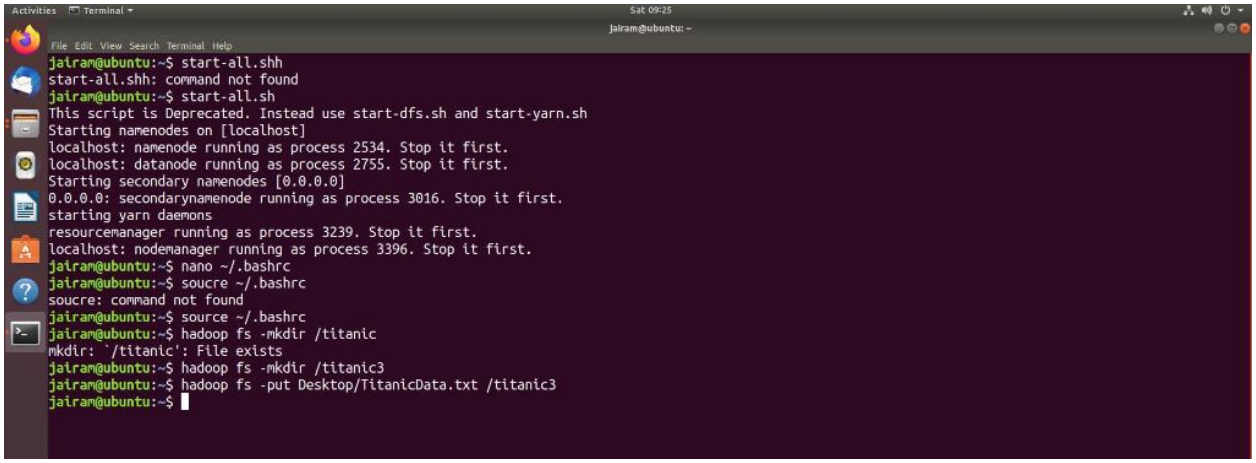
3. Then type ***source ~/.bashrc***



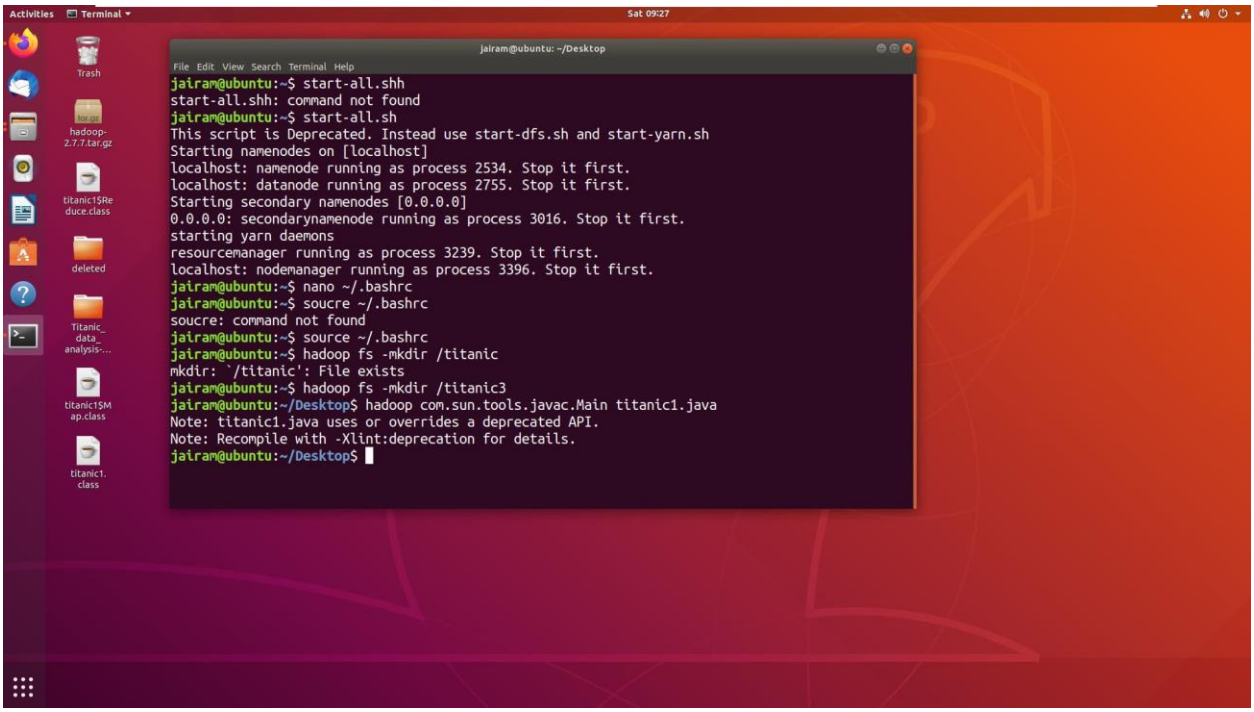
command: ***hadoop fs -mkdir /titanic3***



4. Command: ***hadoop fs -put Desktop/TitanicData.txt /titanic3***

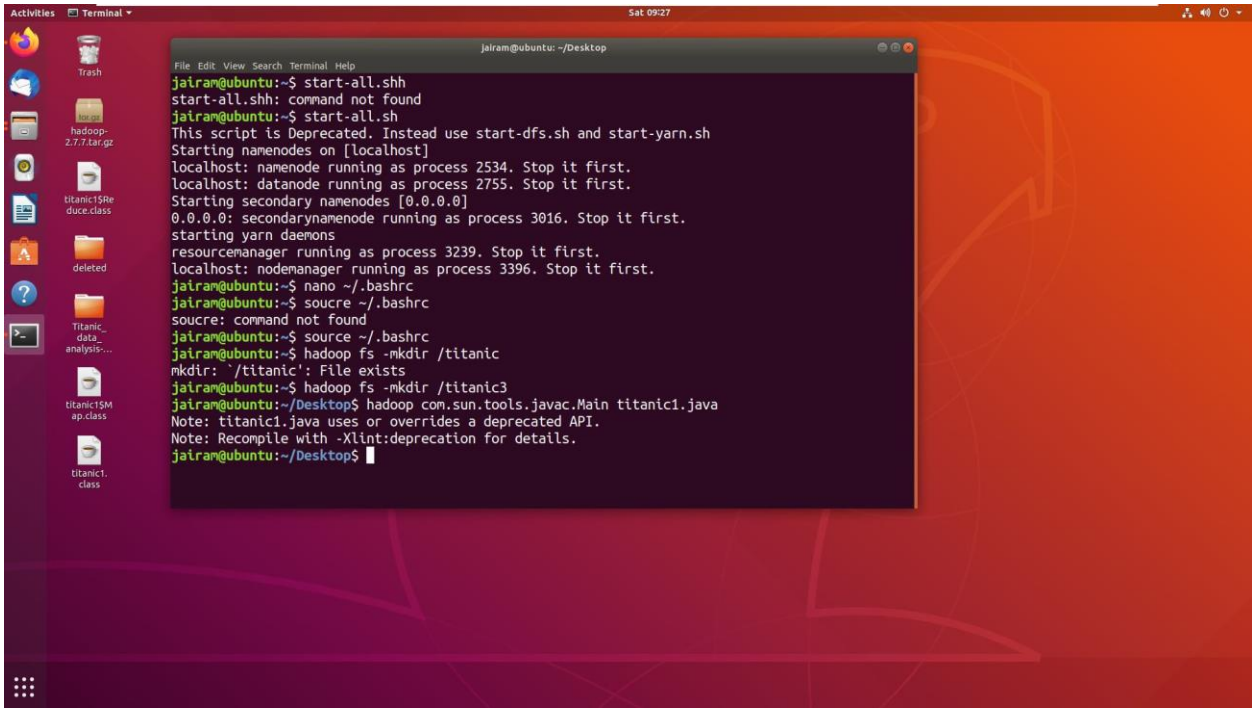


5. command***: cd Desktop***



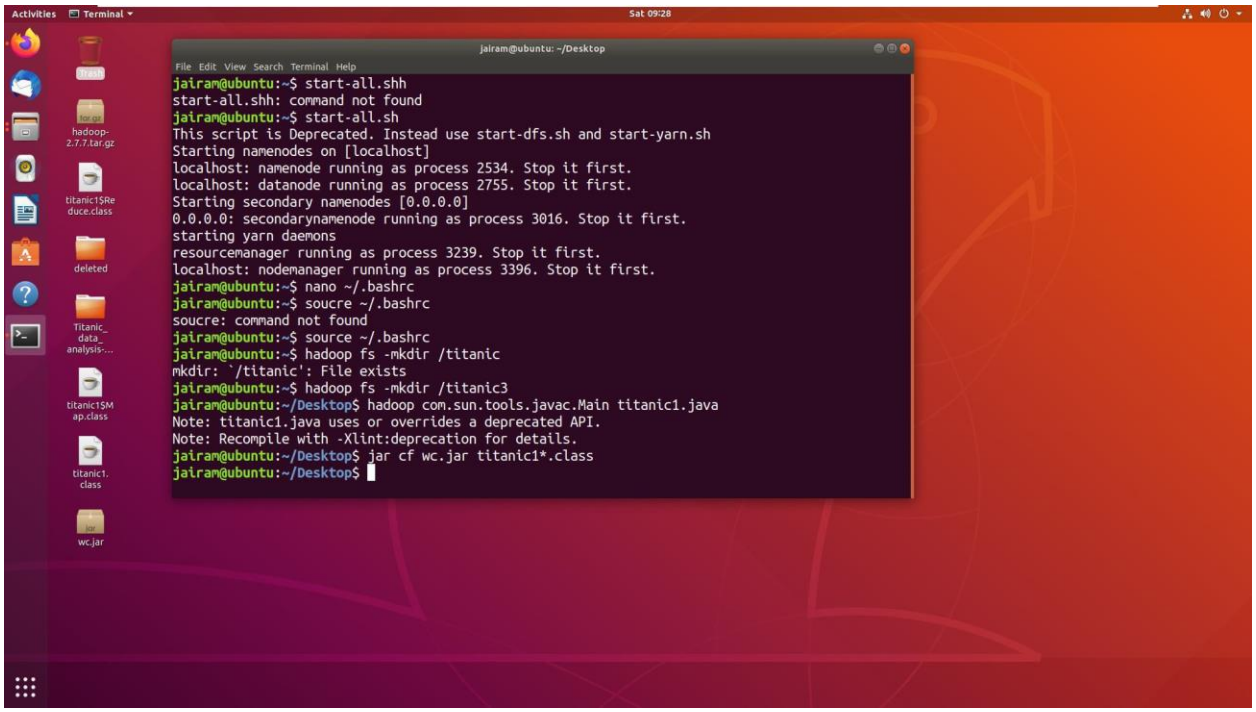
6. Now compile titanic1.java file suppose that file is on Desktop

Command: ***hadoop com.sun.tools.javac.Main titanic1.java***



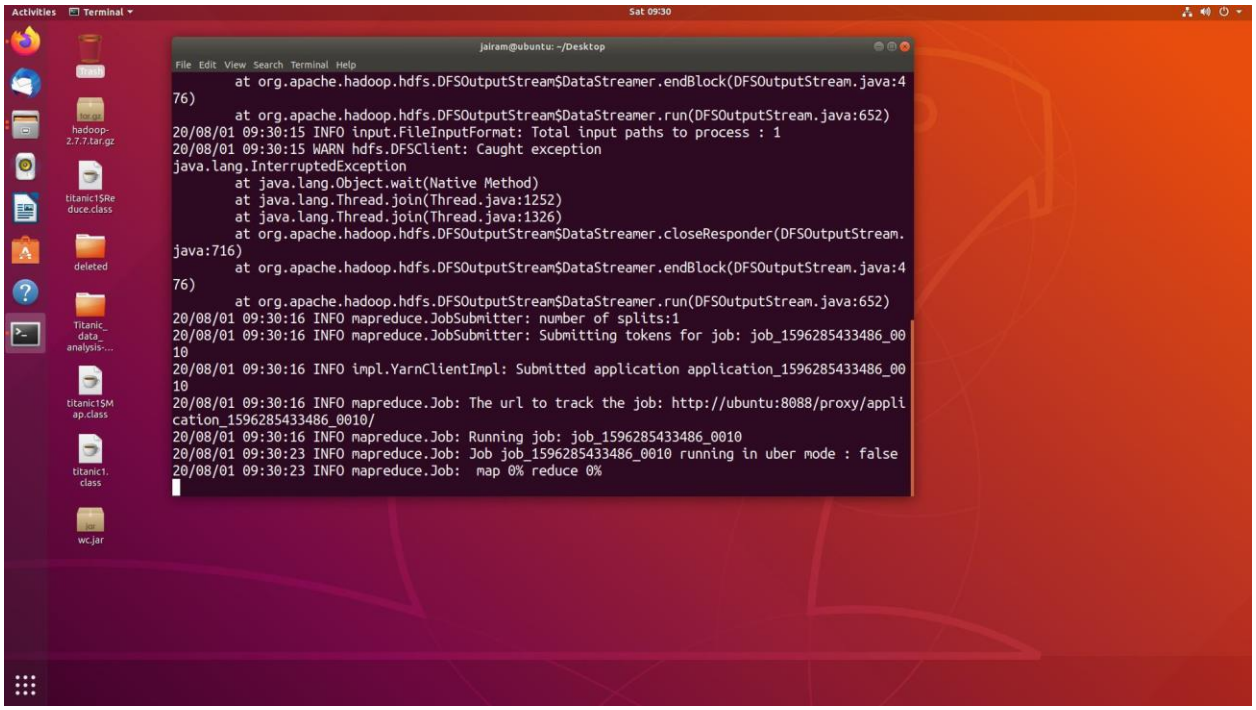
7. To combine all class files

Command**: *jar cf wc.jar titanic1*.class***



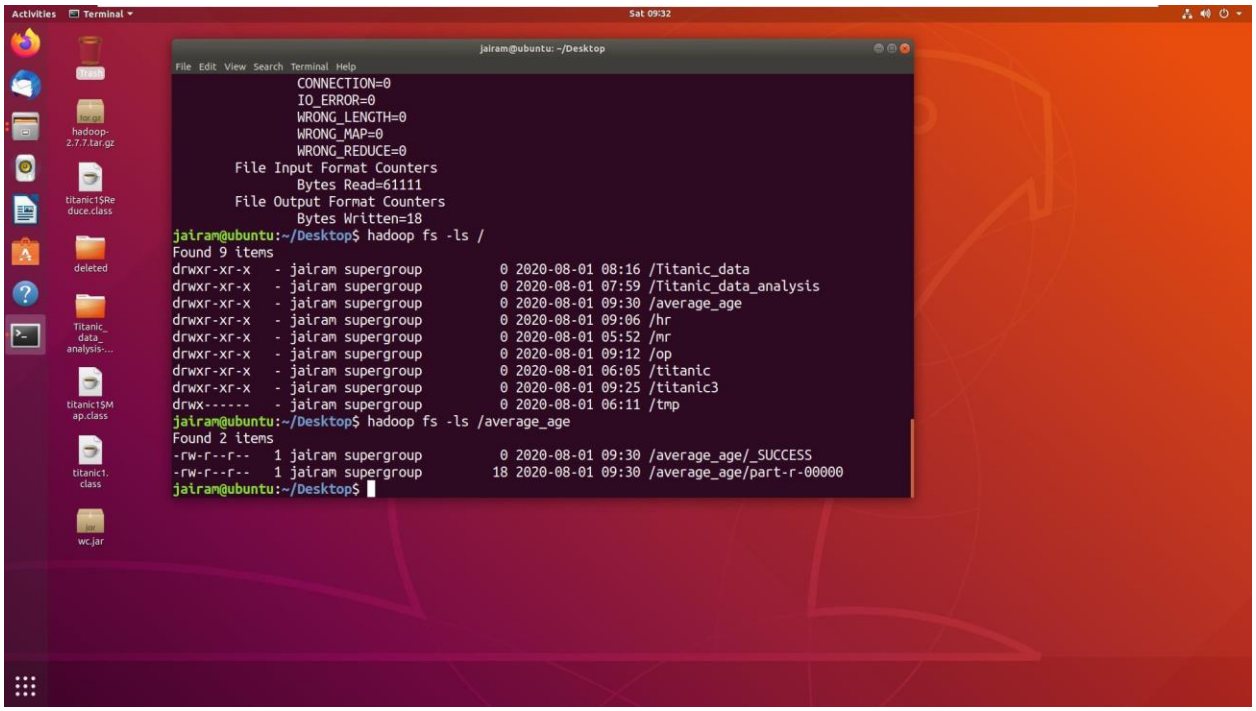
8. To execute

Command: ***hadoop jar wc.jar titanic1 /titanic/TitanicData.txt /average_age***

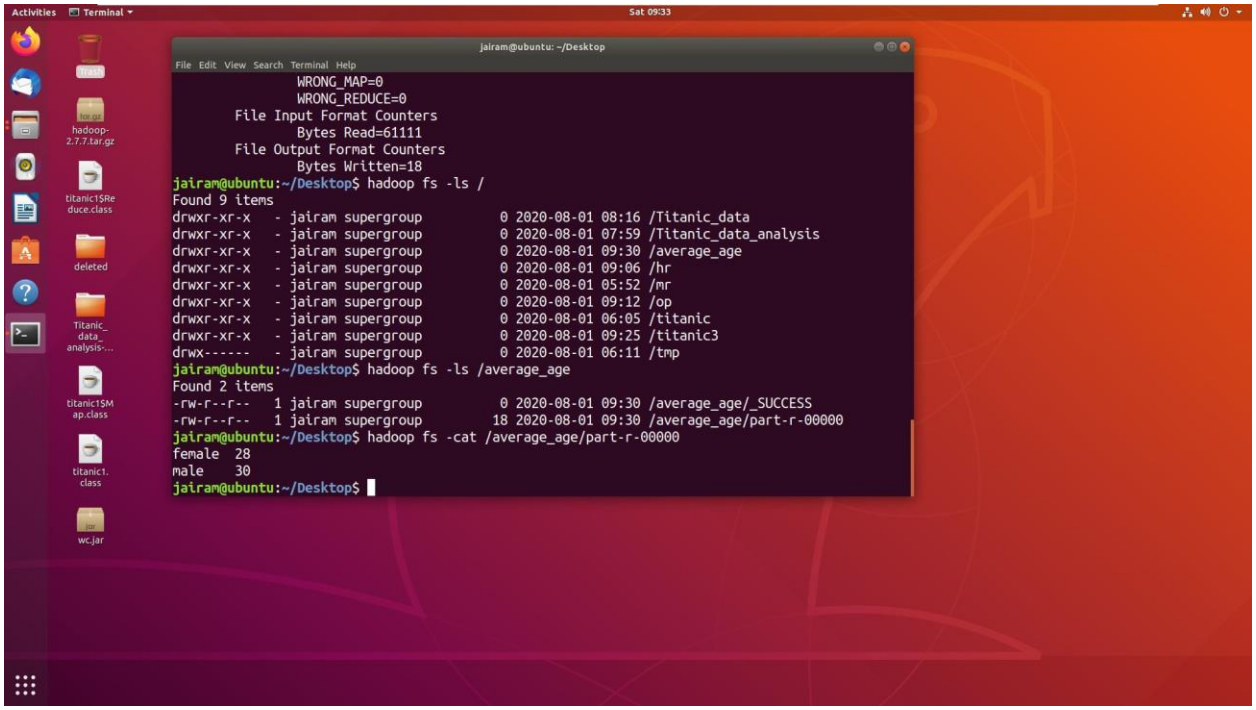


9. to check output folder is there or not

command: ***hadoop fs -ls /***



10.Command: ***hadoop fs -cat /average_age/part-r-00000***

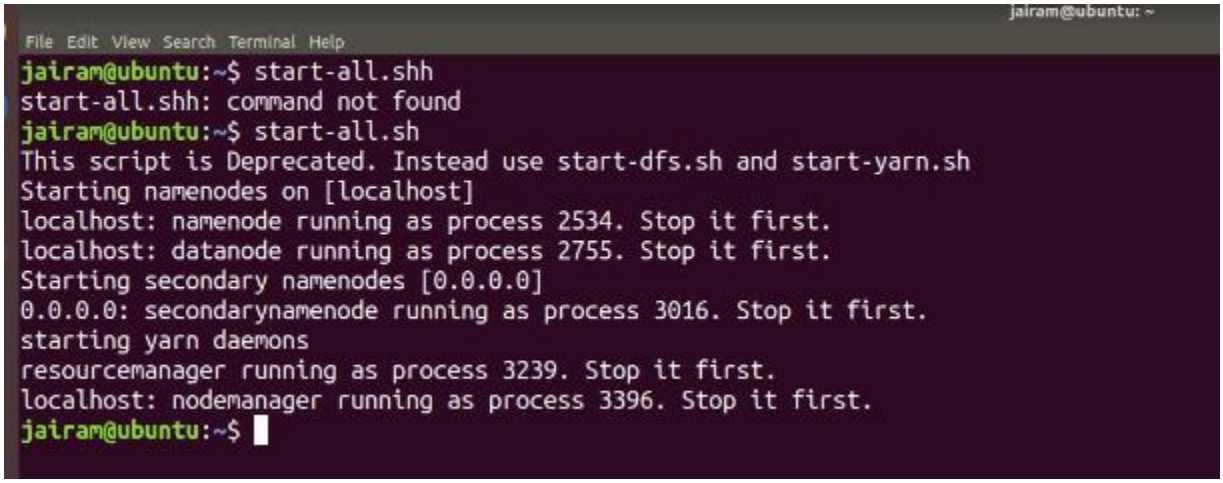


Problem 2:

In this problem statement, we will find the number of people died or survived in each class with their genders and ages.

1. Open terminal and type

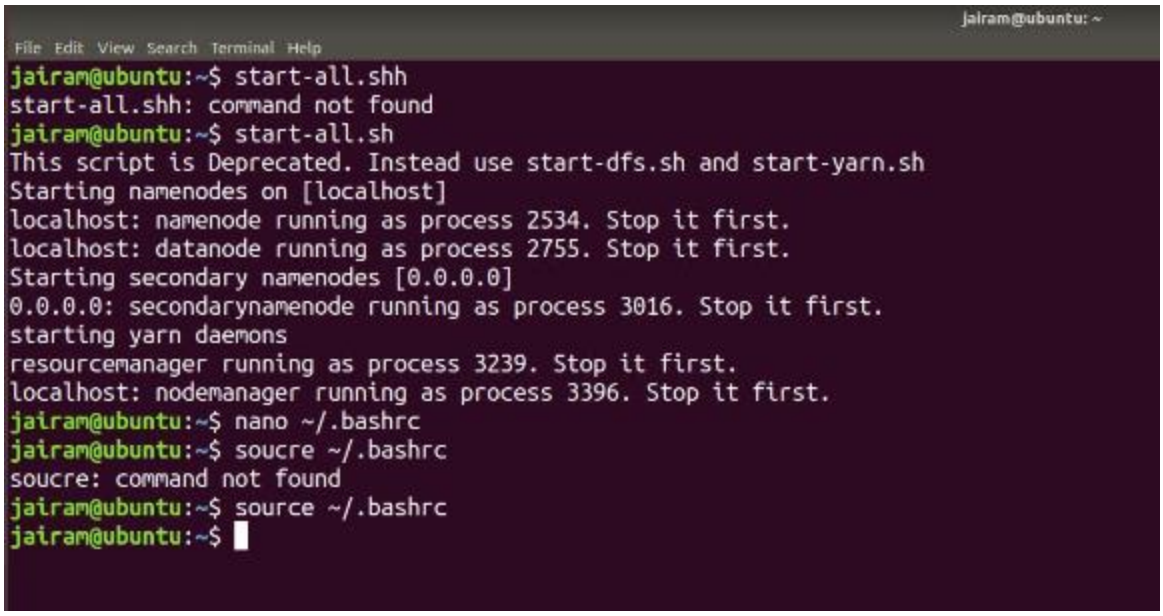
   Command: ***start-all.sh***



2. Open bashrc file by ***nano ~/.bashrc*** and type

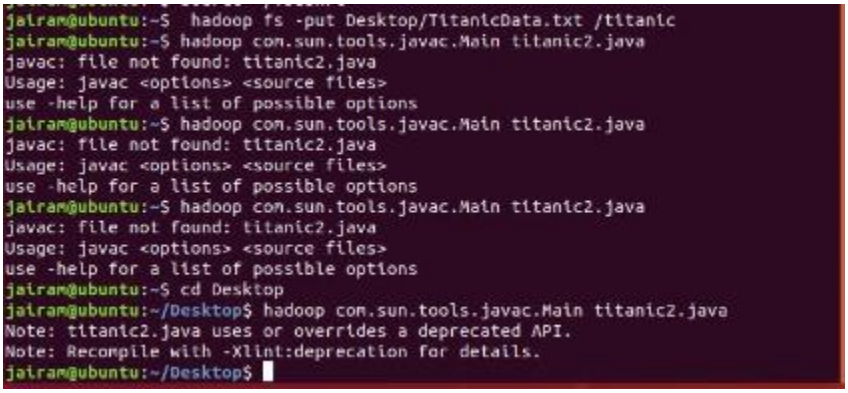
   export HADOOP_CLASSPATH=$JAVA_HOME/lib/tools.jar

3. Then type ***source ~/.bashrc***



Command: ***hadoop fs -mkdir /titanic***

4. Command: ***hadoop fs -put Desktop/TitanicData.txt /titanic***



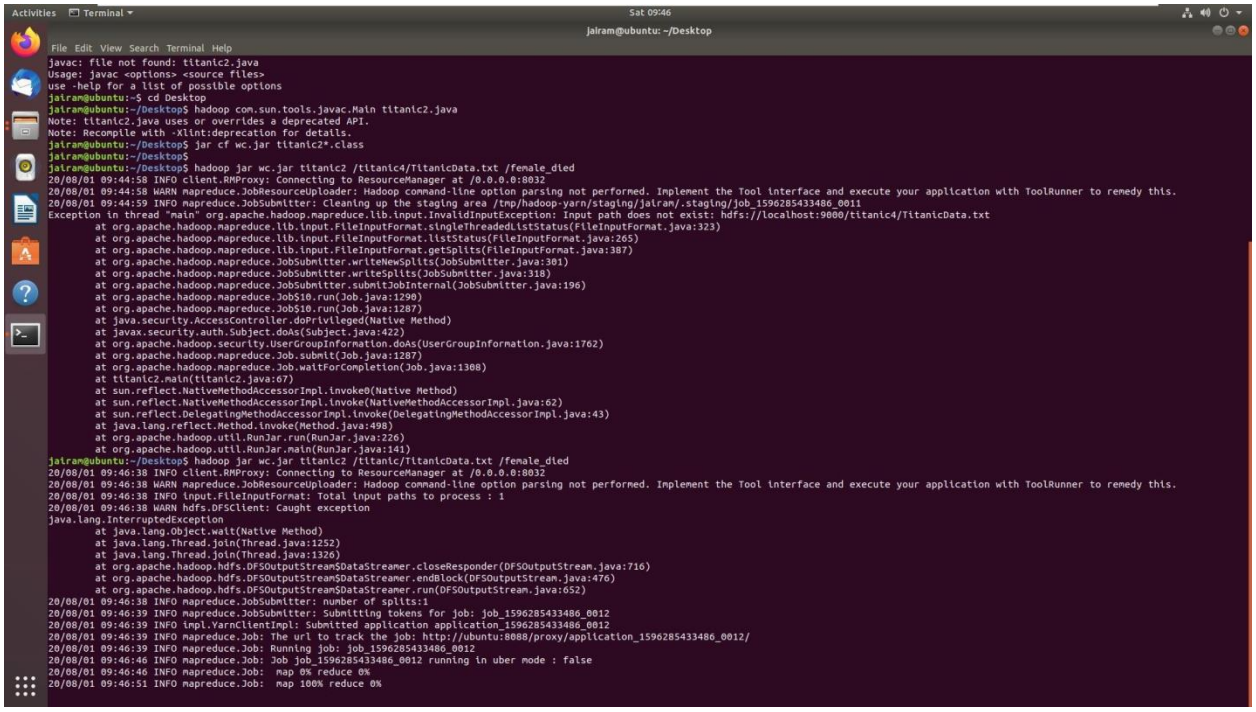
5. Command: ***cd Desktop***
6. Now compile titanic2.java file suppose that file is on Desktop

Command: ***hadoop com.sun.tools.javac.Main titanic2.java***

```
jairam@ubuntu:~$  hadoop fs -put Desktop/TitanicData.txt /titanic
jairam@ubuntu:~$ hadoop com.sun.tools.javac.Main titanic2.java
javac: file not found: titanic2.java
Usage: javac <options> <source files>
use -help for a list of possible options
jairam@ubuntu:~$ hadoop com.sun.tools.javac.Main titanic2.java
javac: file not found: titanic2.java
Usage: javac <options> <source files>
use -help for a list of possible options
jairam@ubuntu:~$ hadoop com.sun.tools.javac.Main titanic2.java
javac: file not found: titanic2.java
Usage: javac <options> <source files>
use -help for a list of possible options
jairam@ubuntu:~$ cd Desktop
jairam@ubuntu:~/Desktop$ hadoop com.sun.tools.javac.Main titanic2.java
Note: titanic2.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
jairam@ubuntu:~/Desktop$
```

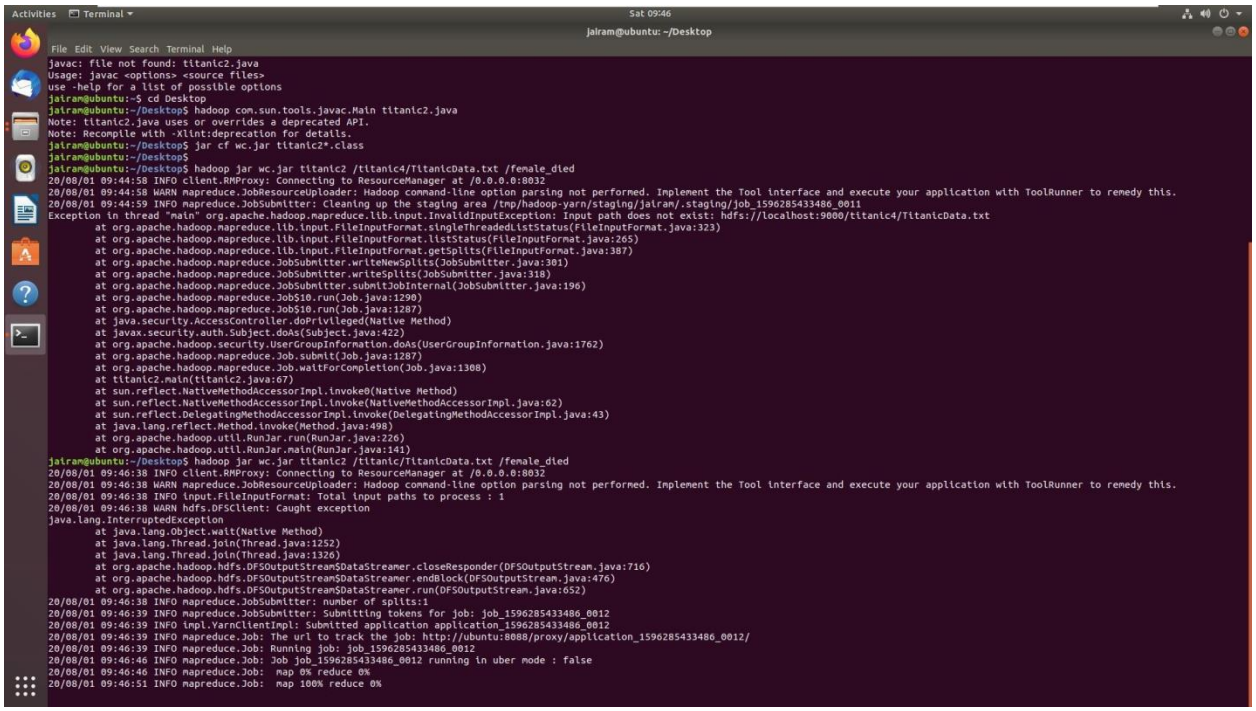
7. To combine all class files

Command*: **jar cf wc.jar titanic2*.class***



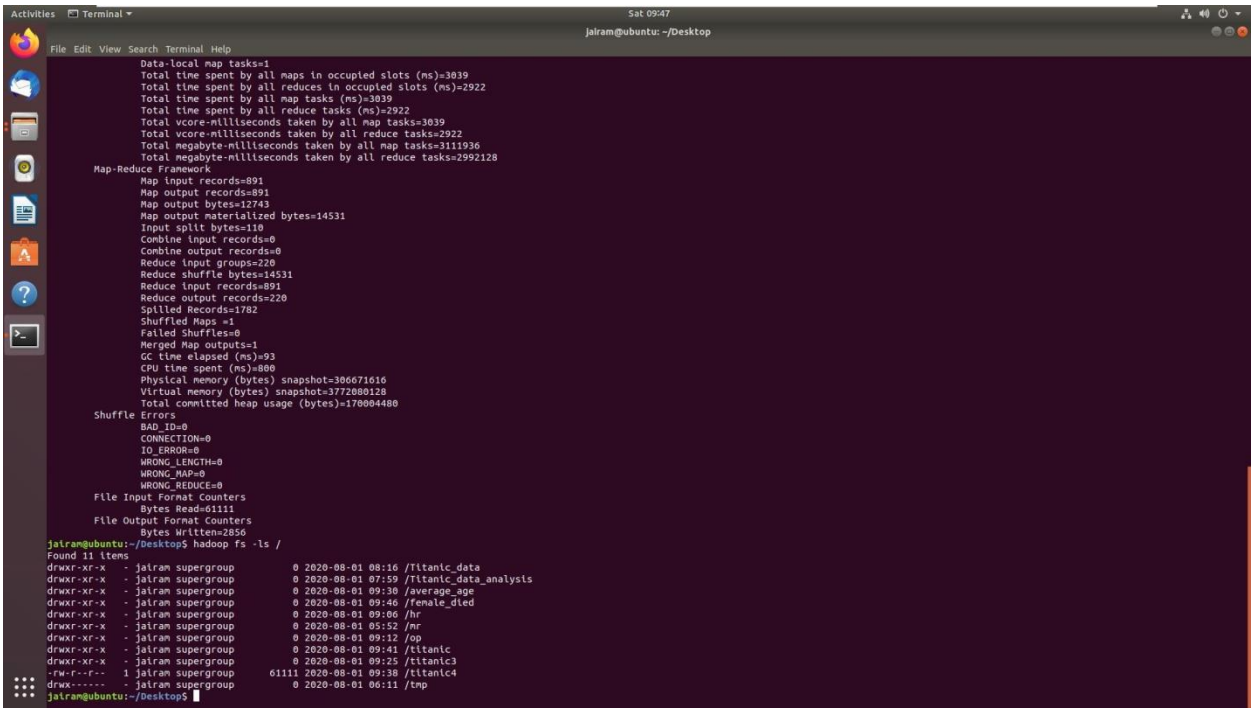
8. To execute

Command: ***hadoop jar wc.jar titanic2 /titanic/TitanicData.txt /female_died***

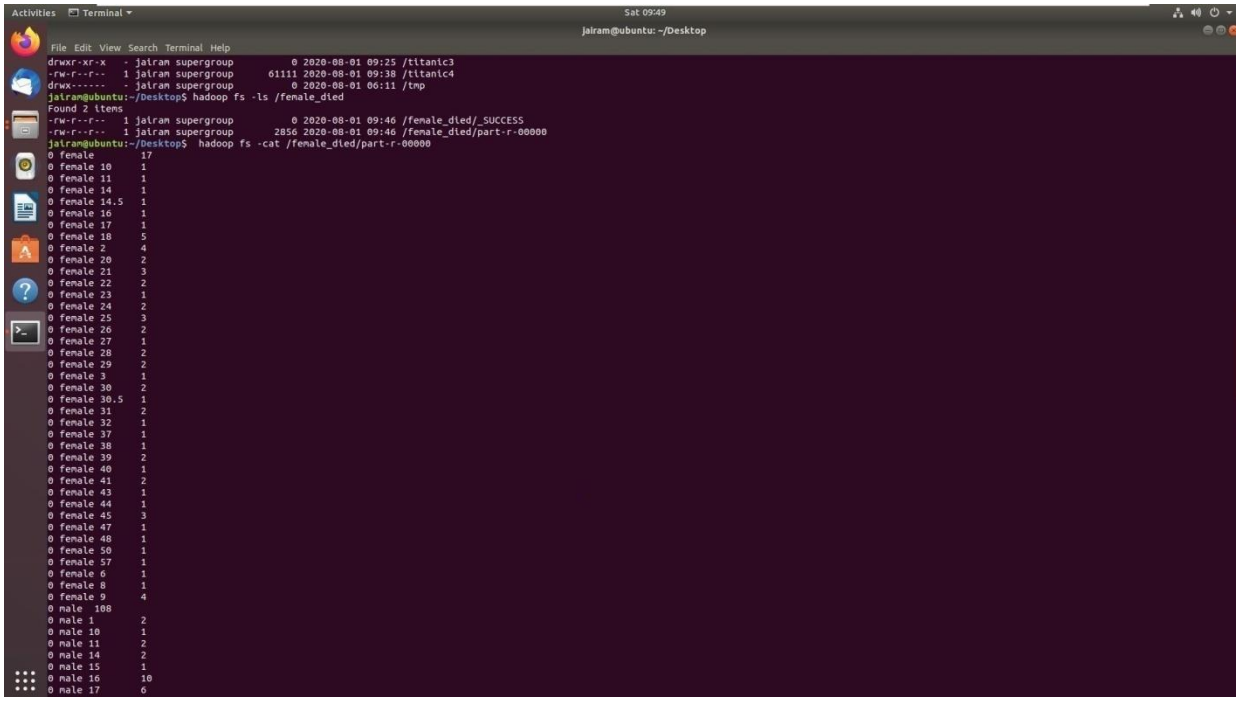


9. to check output folder is there or not

Command: ***hadoop fs -ls /***



10.command***: hadoop fs -cat /female_died/part-r-00000***



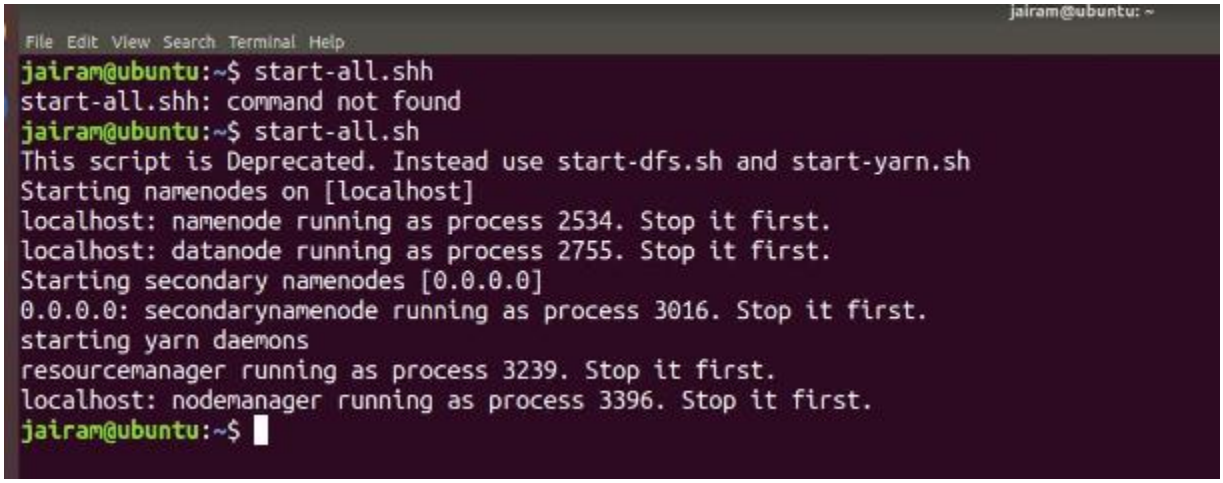
Uber data analysis

Problem 1:

In this problem statement, we will find the days on which each basement has more trips.

1. Open terminal and type

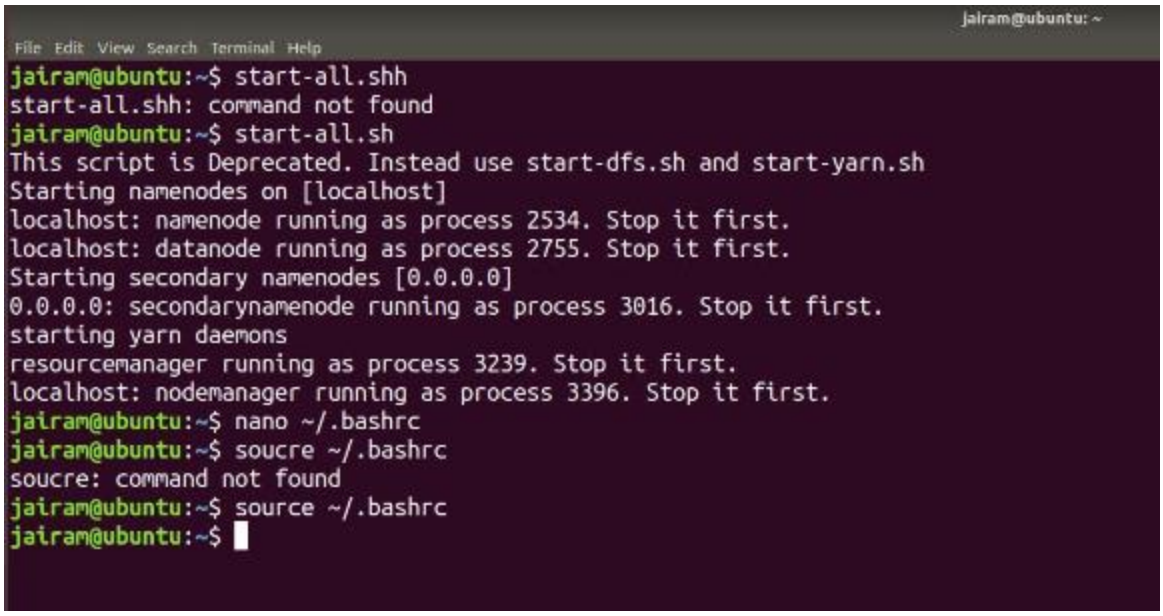
   Command: ***start-all.sh***



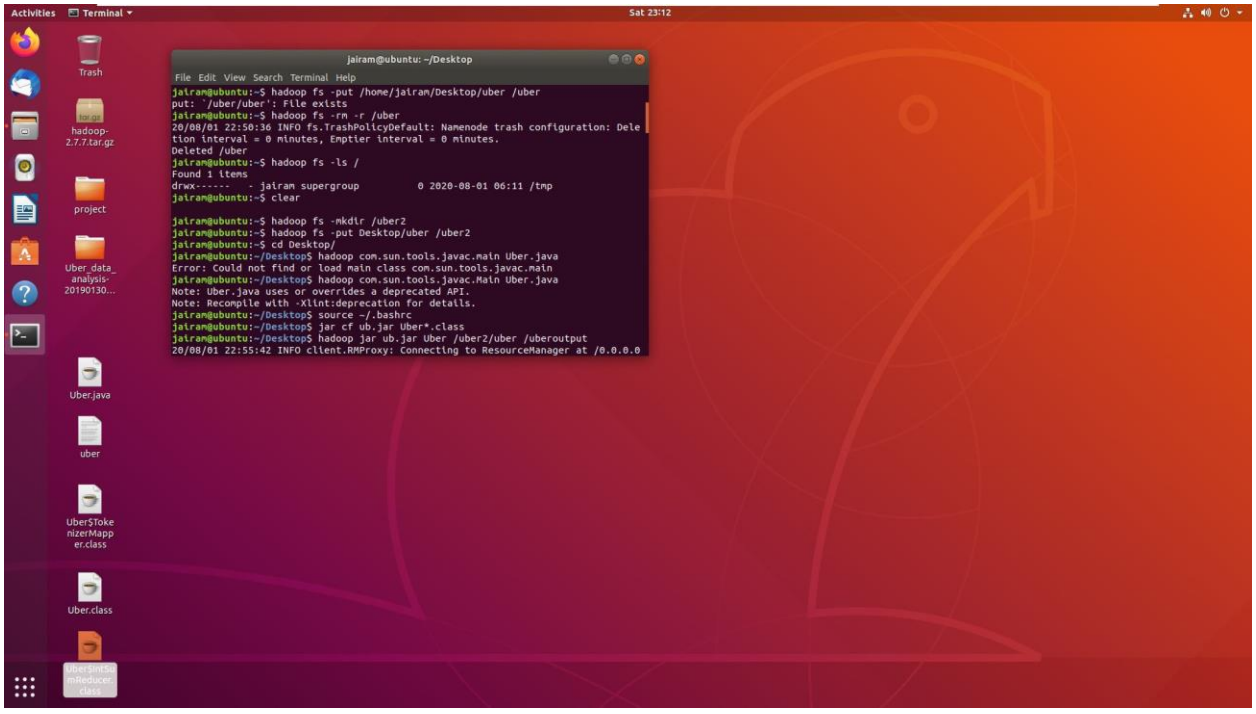
2. Open bashrc file by ***nano ~/.bashrc*** and type

   export HADOOP_CLASSPATH=$JAVA_HOME/lib/tools.jar

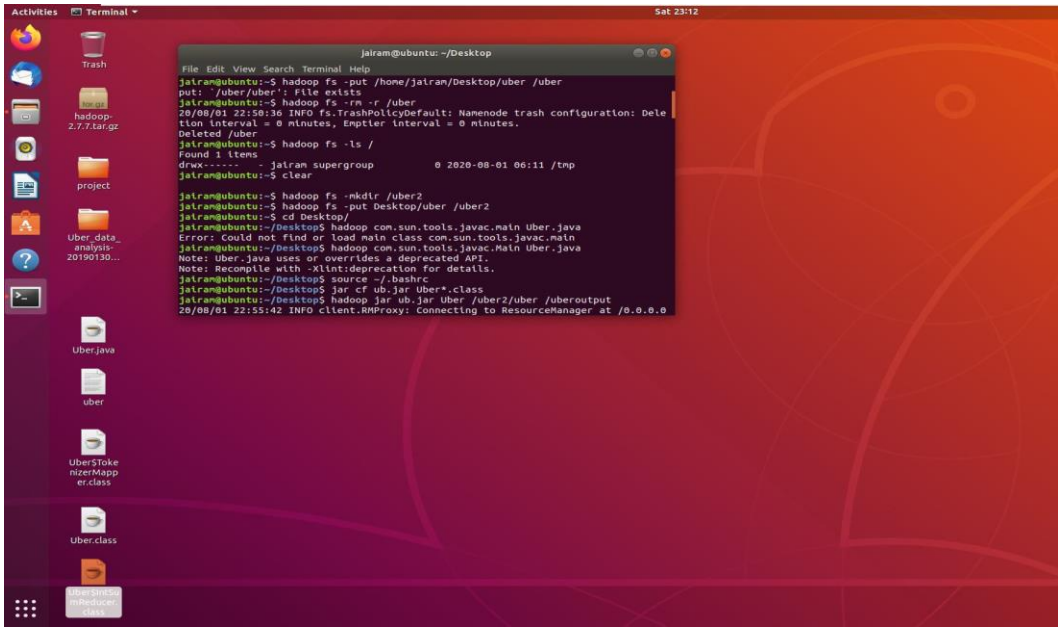
3. Then ***source ~/.bashrc***



4. command: ***hadoop fs -mkdir /uber2***



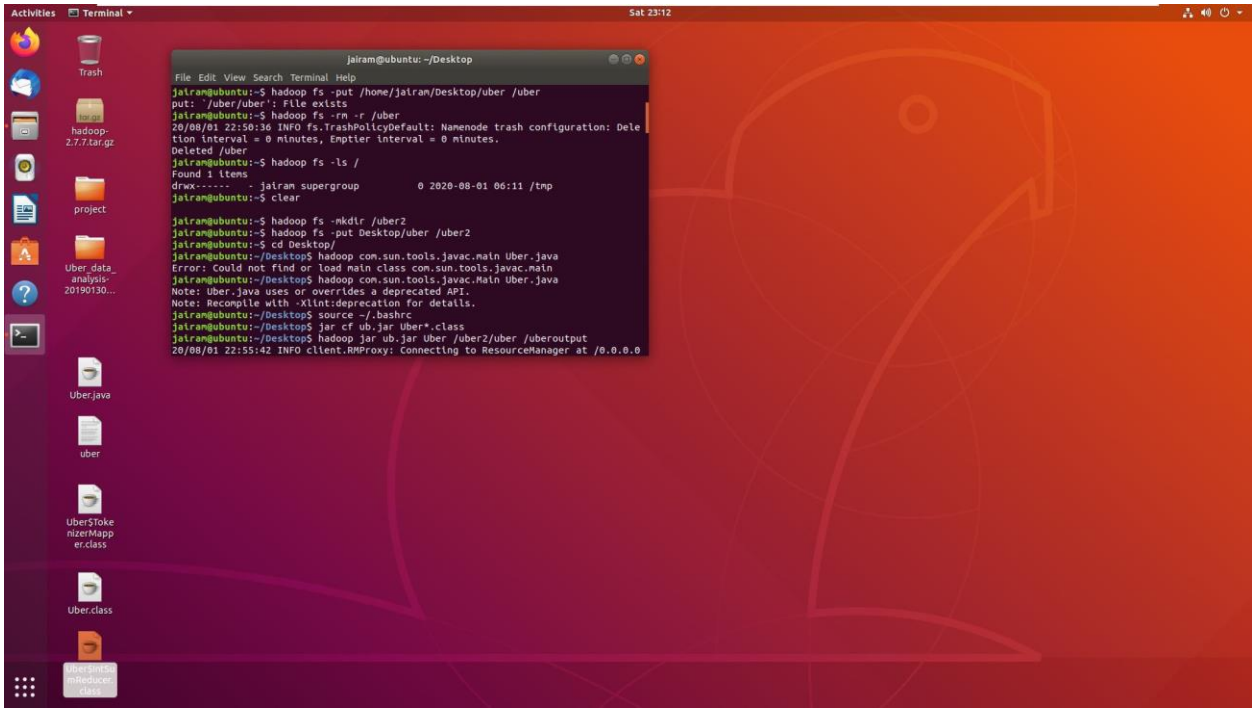
5. command***: hadoop fs -put Desktop/Uber.txt /uber2***



6. command: ***cd Desktop***

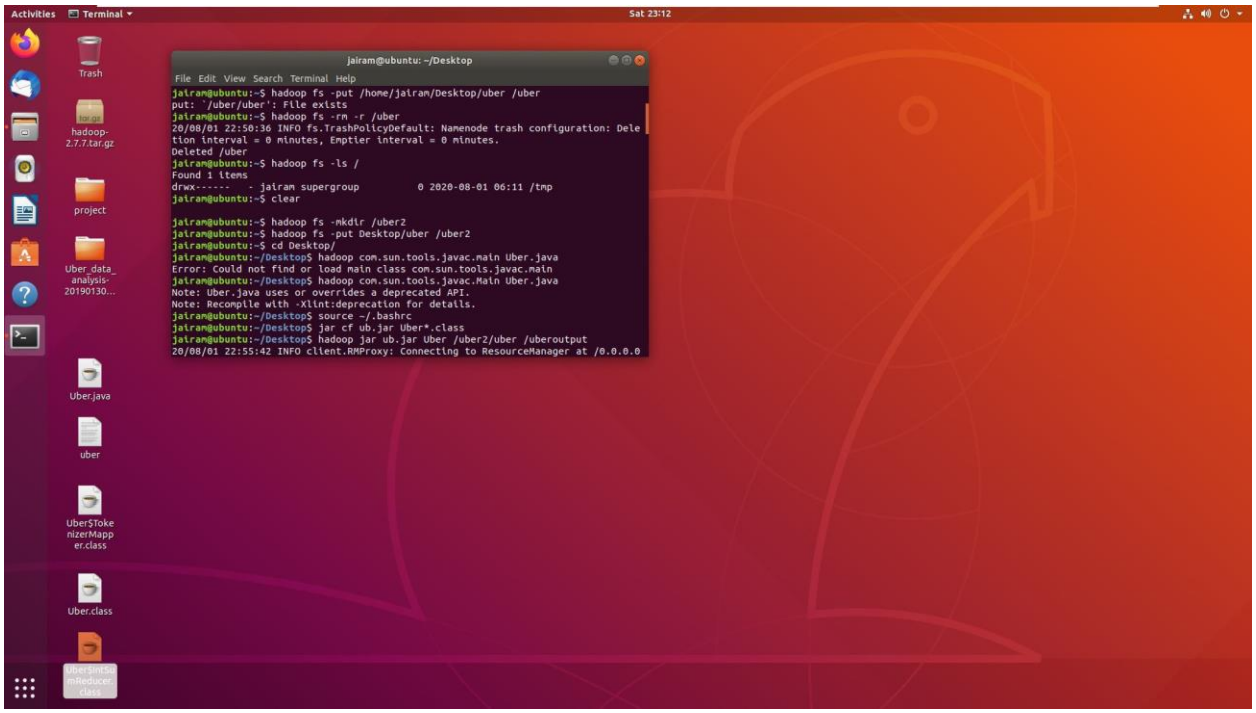
7. Now compile titanic1.java file suppose that file is on Desktop

Command: ***hadoop com.sun.tools.javac.Main uber1.java***



8. To combine all class files

Command: ***jar cf wc.jar uber1*.class***

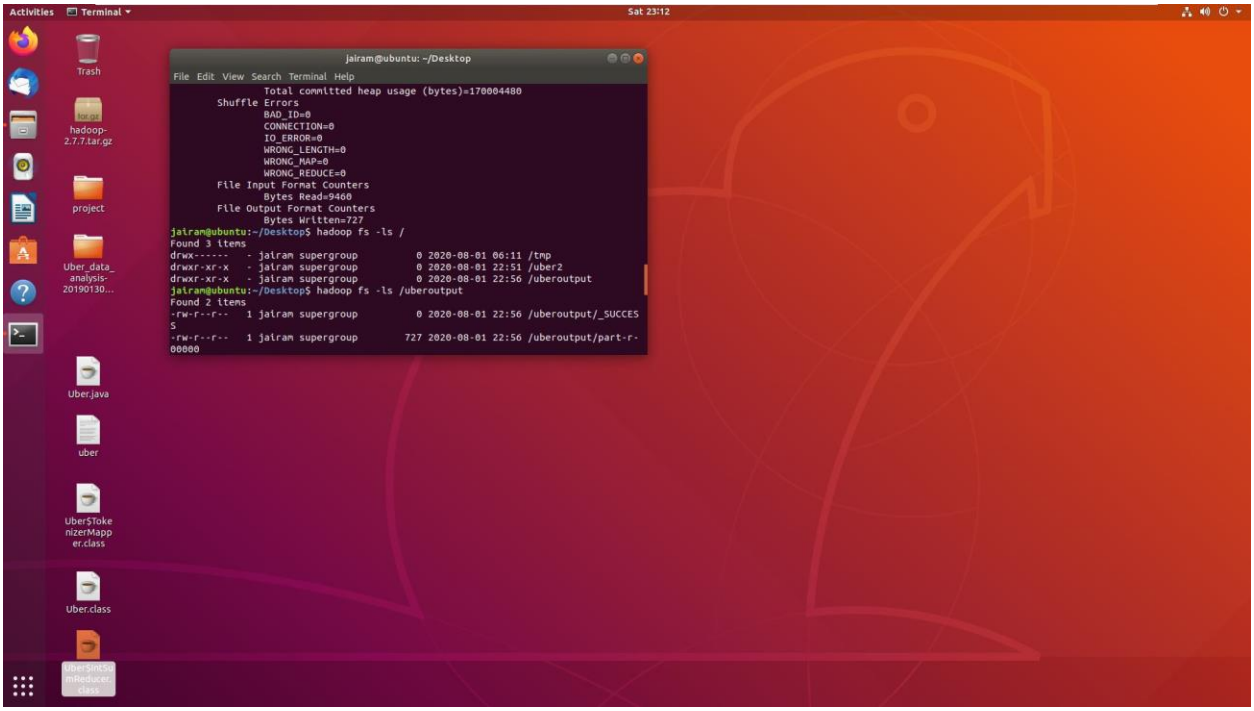


9. To execute

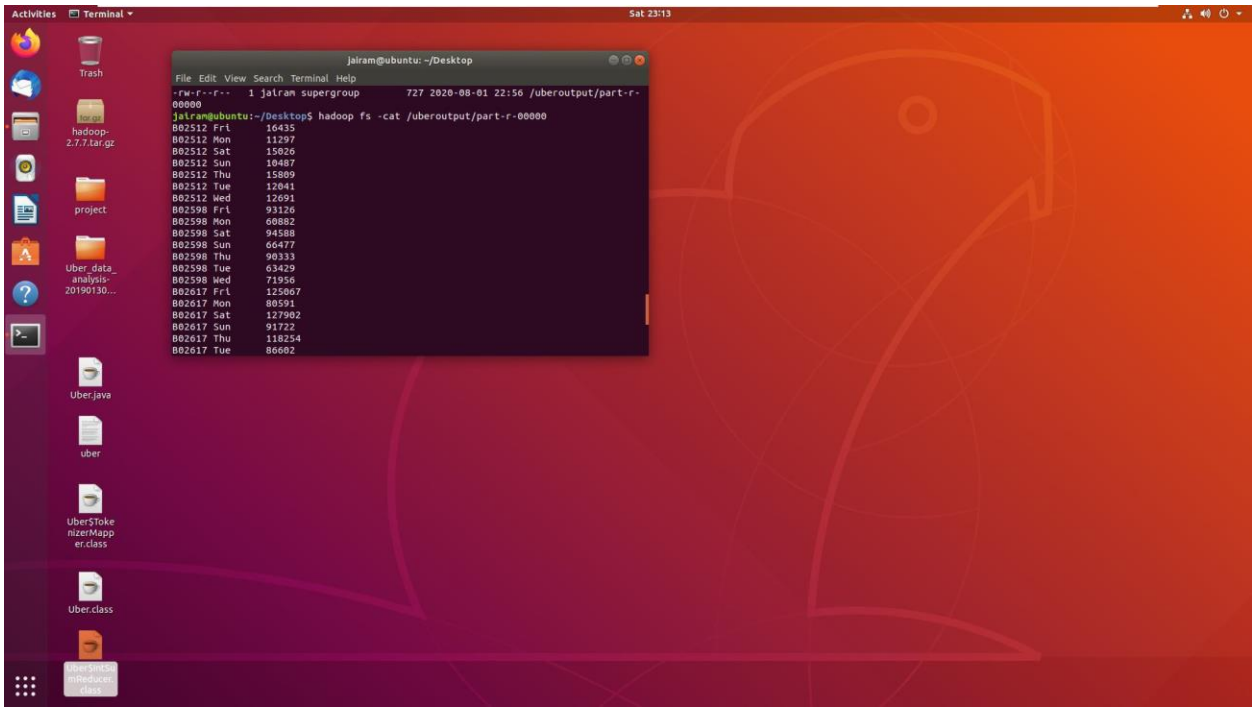
Command: ***hadoop jar wc.jar uber1 /uber2/Uber.txt /uberoutput***

11.to check output folder is there or not

Command: ***hadoop fs -ls /***



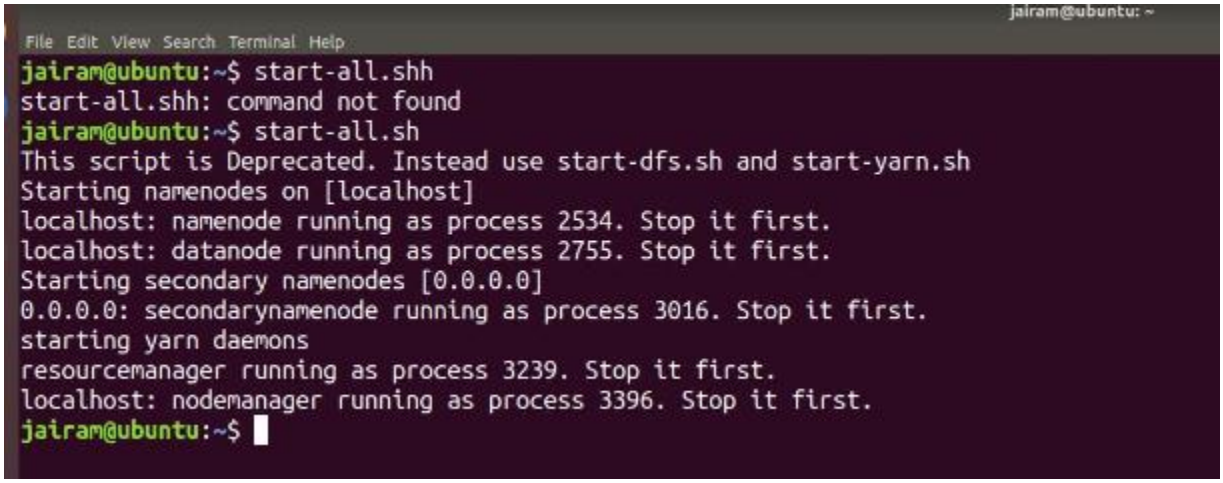
11.command: ***hadoop fs -cat /uberoutput/part-r-00000***


Activities
Terminal
Sat 23:13
Trash
hadoop-2.7.7.tar.gz
project
Uber_data_analysis-20190130...
Uber.java
uber
Uber$TokenizerMapper.class
Uber.class
Uber$IntSumReducer.class
jairam@ubuntu: ~/Desktop
File Edit View Search Terminal Help
-rw-r--r-- 1 jairam supergroup 727 2020-08-01 22:56 /uberoutput/part-r-00000
jairam@ubuntu:~/Desktop$ hadoop fs -cat /uberoutput/part-r-00000
B02512 Fri 16435
B02512 Mon 11297
B02512 Sat 15026
B02512 Sun 10487
B02512 Thu 15809
B02512 Tue 12041
B02512 Wed 12691
B02598 Fri 93126
B02598 Mon 60882
B02598 Sat 94588
B02598 Sun 66477
B02598 Thu 90333
B02598 Tue 63429
B02598 Wed 71956
B02617 Fri 125067
B02617 Mon 80591
B02617 Sat 127902
B02617 Sun 91722
B02617 Thu 118254
B02617 Tue 86602

Problem 2:

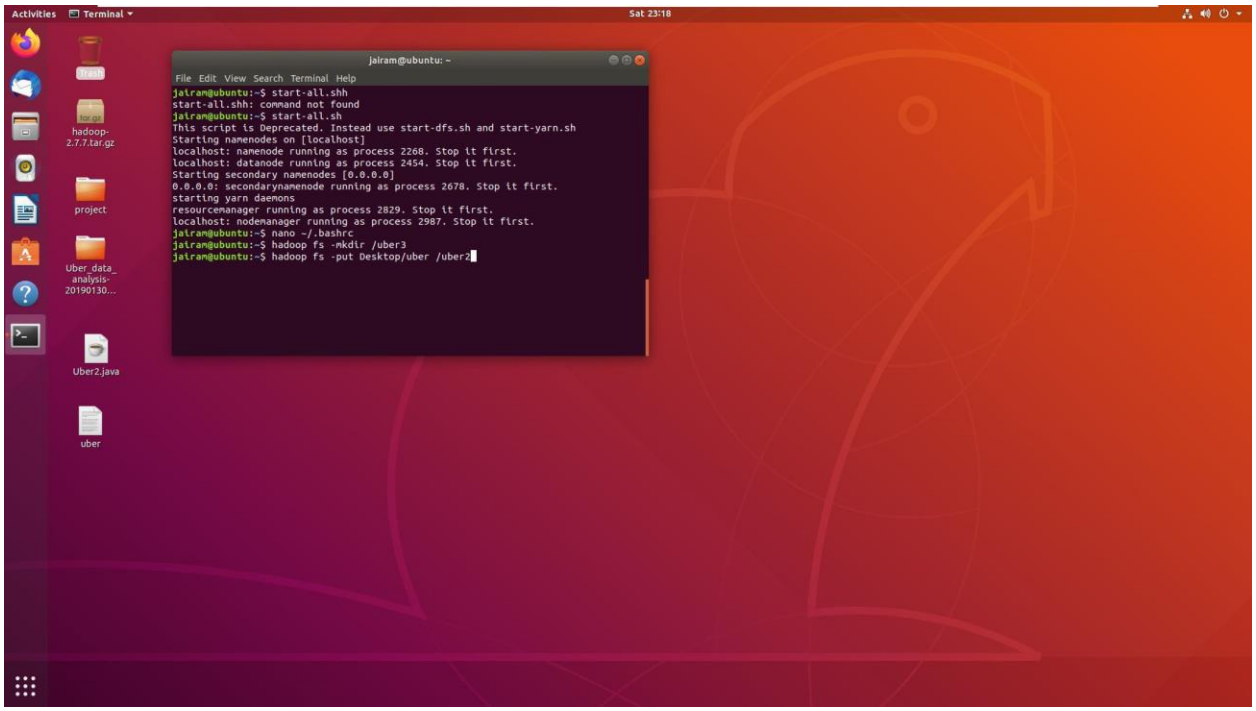
In this problem statement, we will find the days on which each basement has more number of active vehicles.

1. Open terminal and type Command: ***start-all.sh***

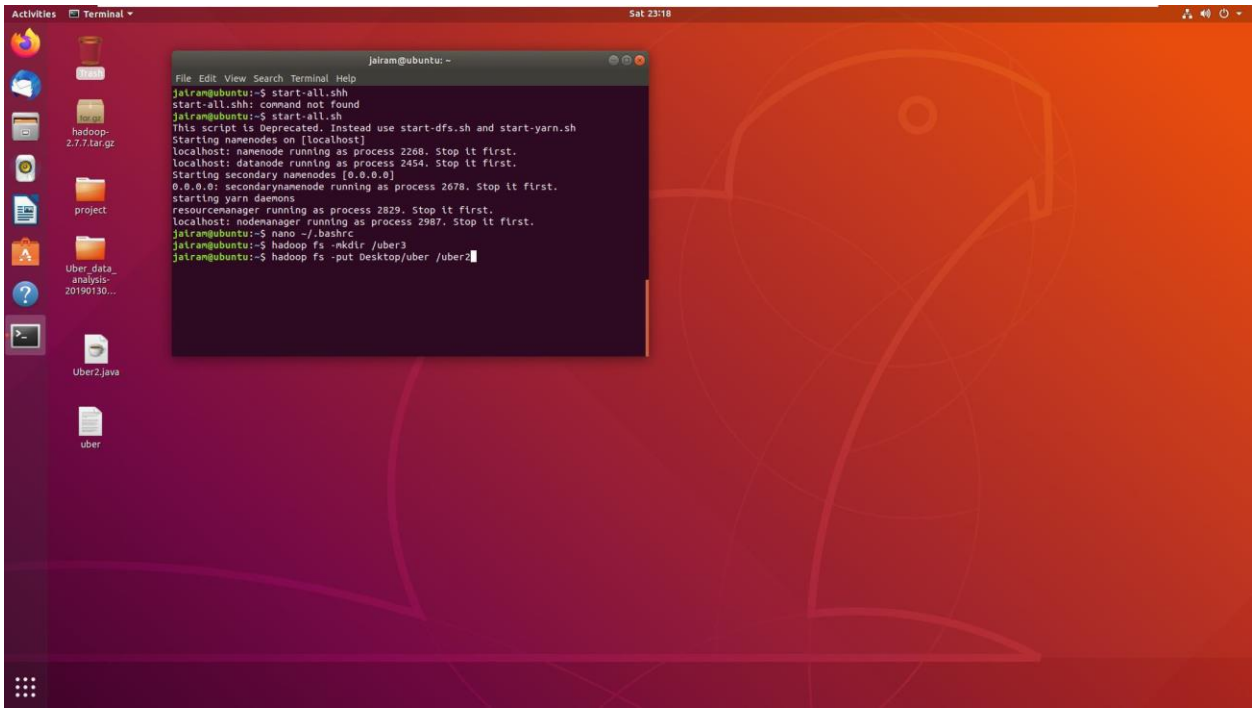


2. Open bashrc file by ***nano ~/.bashrc*** and type

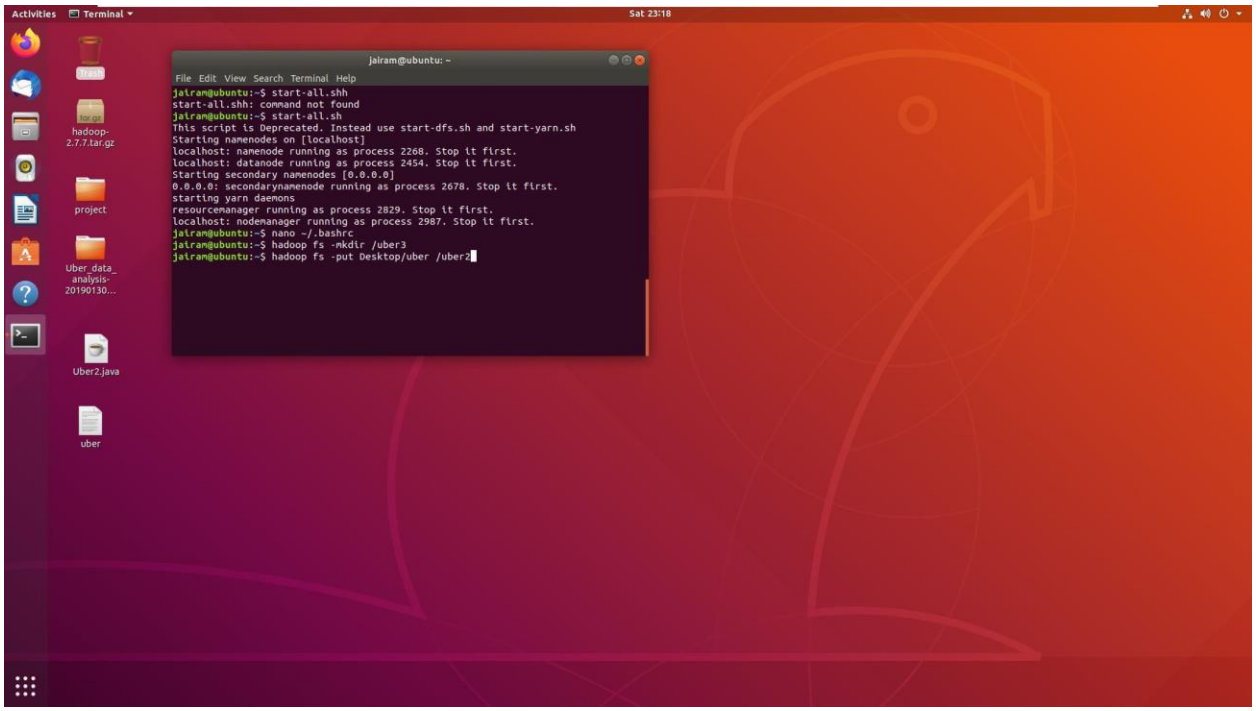
   export HADOOP_CLASSPATH=$JAVA_HOME/lib/tools.jar



3. Command: ***hadoop fs -mkdir /uber3***

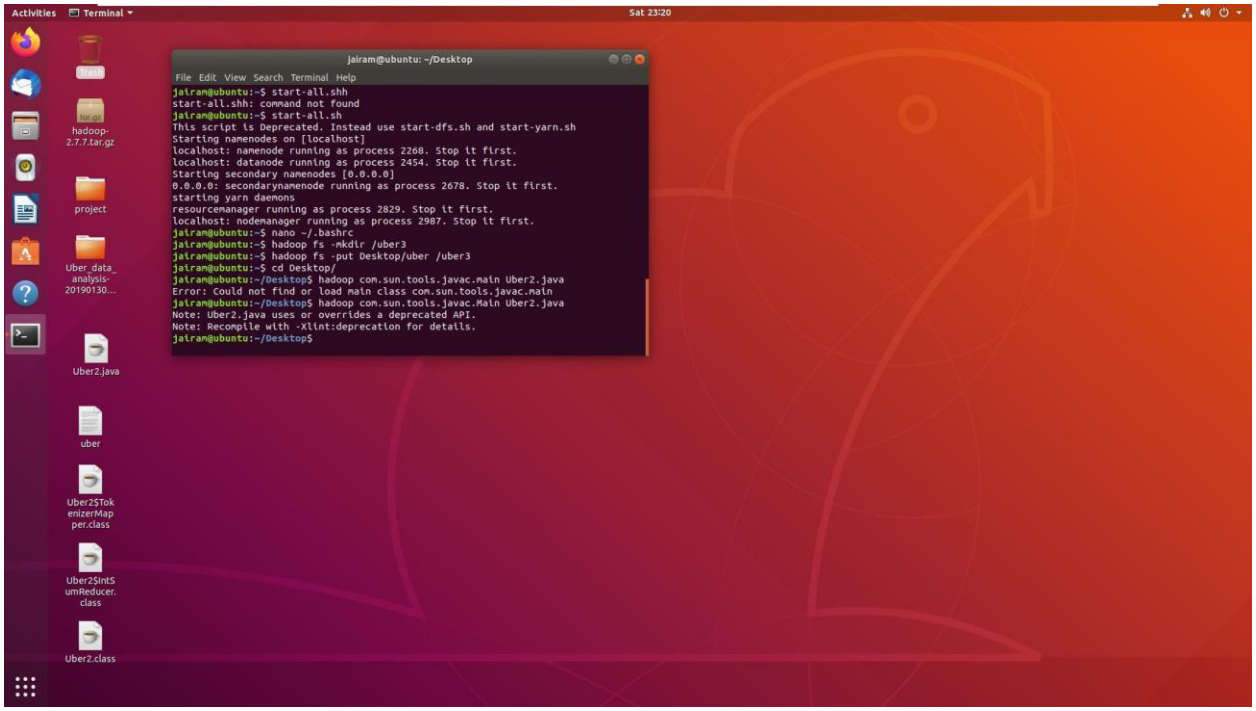


4. Command: ***hadoop fs -put Desktop/Uber.txt /uber3***



5. Command***: cd Desktop***
6. Now compile titanic1.java file suppose that file is on Desktop

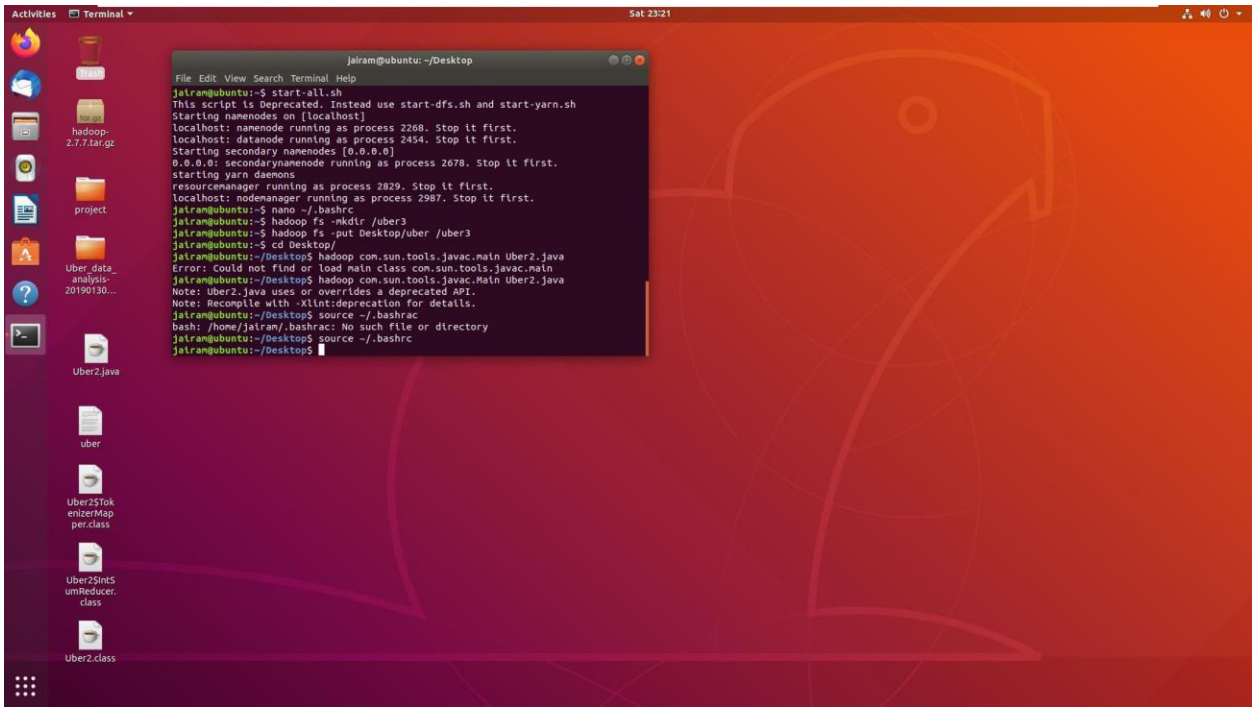
   hadoop com.sun.tools.javac.Main uber2.java



7. Then ***source ~/.bashrc***

8. To combine all class files

jar cf wc.jar uber2*.class

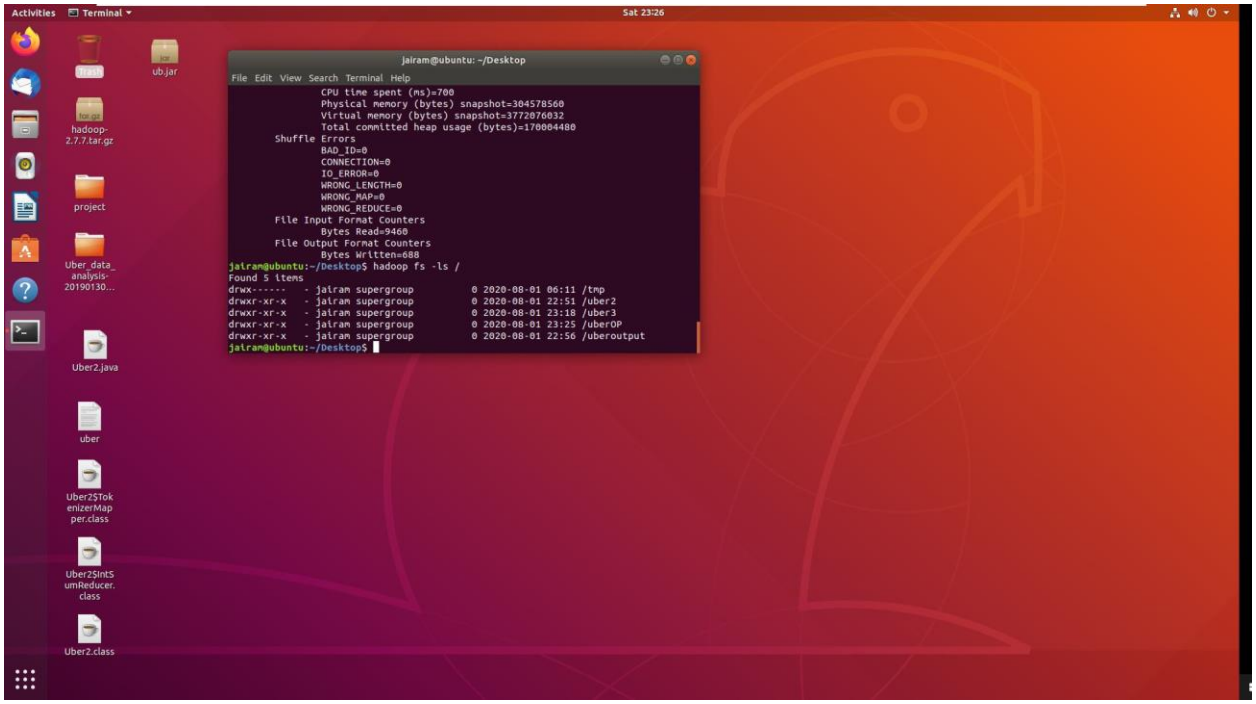


9. To execute

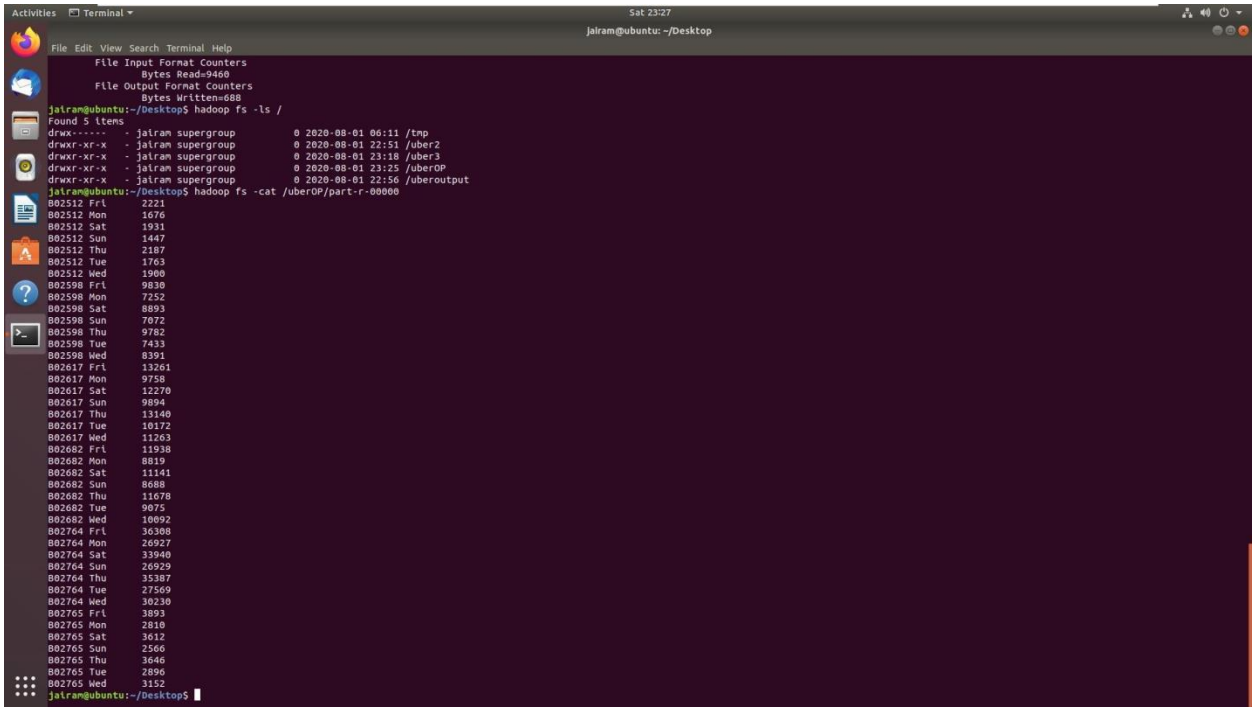
hadoop jar wc.jar uber2 /uber3/Uber.txt /uberop

10. to check output folder is there or not

hadoop fs -ls /

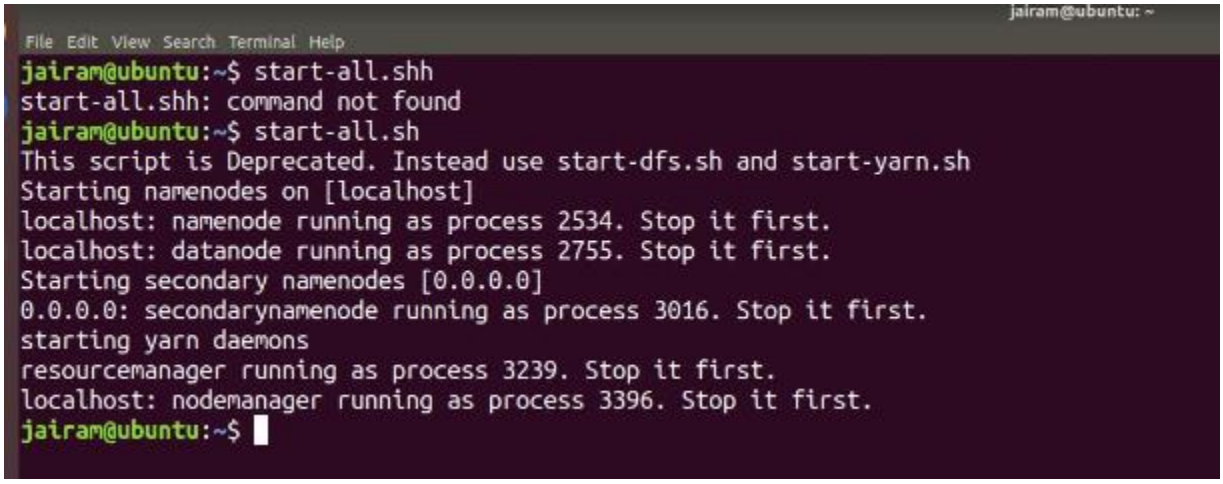


11. command: ***hadoop fs -cat /uberop/part-r-00000***

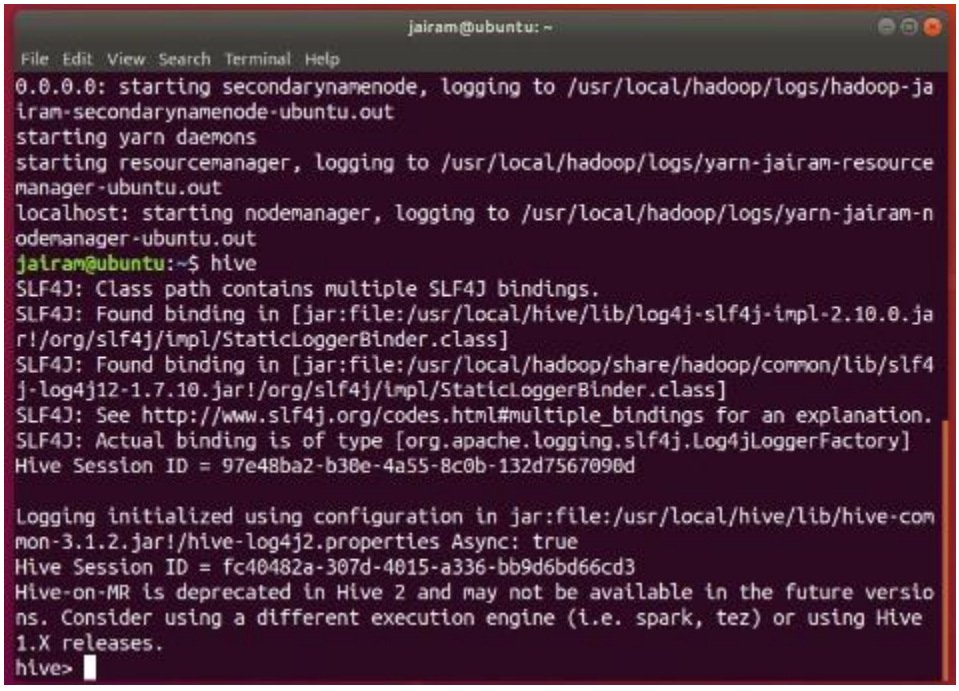


Pokemon data analysis

1. Open terminal and type Command: ***start-all.sh***

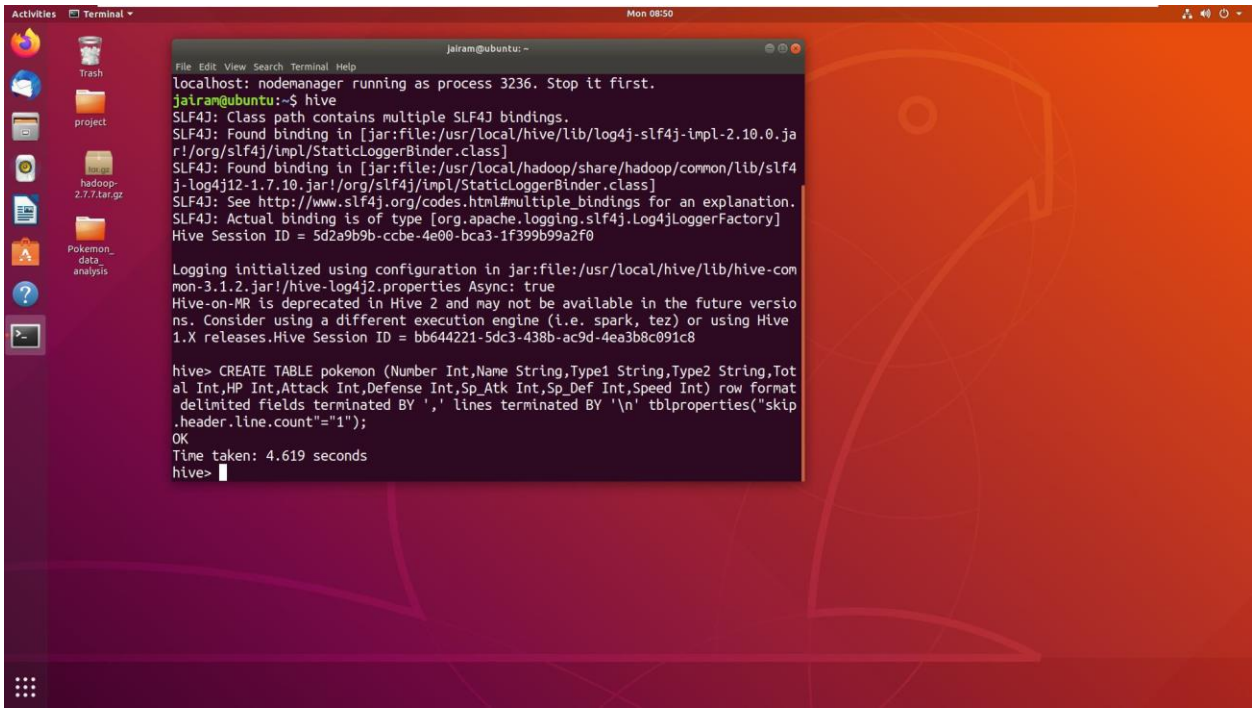


2. Command: ***hive*** for lauch hive



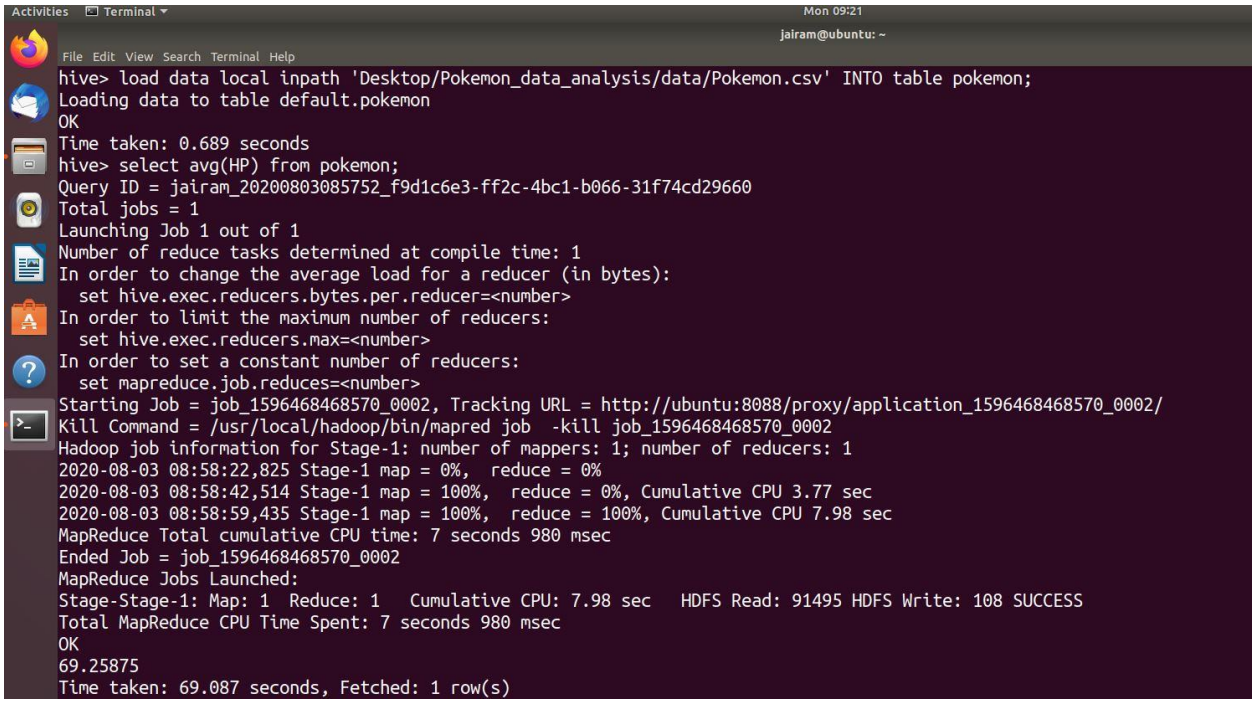
3. to create table and type

CREATE TABLE pokemon (Number Int,Name String,Type1 String,Type2 String,Total Int,HP Int,Attack Int,Defense Int,Sp_Atk Int,Sp_Def Int,Speed Int) row format delimited fields terminated BY ',' lines terminated BY '\n' tblproperties("skip.header.line.count"="1");



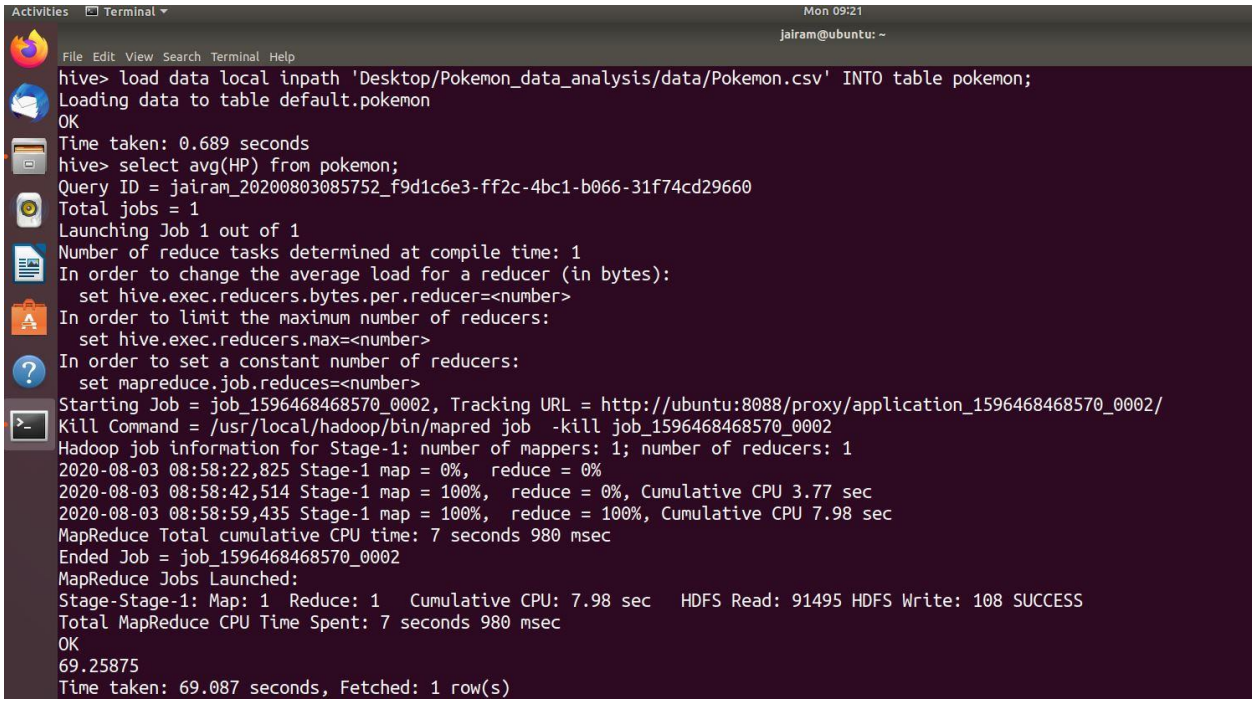
4. to load data and type

load data local inpath 'Desktop/Pokemon_data_analysis/Pokemon.csv ' INTO table pokemon;



5. Find out the average HP (Hit points) of all the Pokémon, using the below query.

Command: ***select avg(HP) from pokemon;***

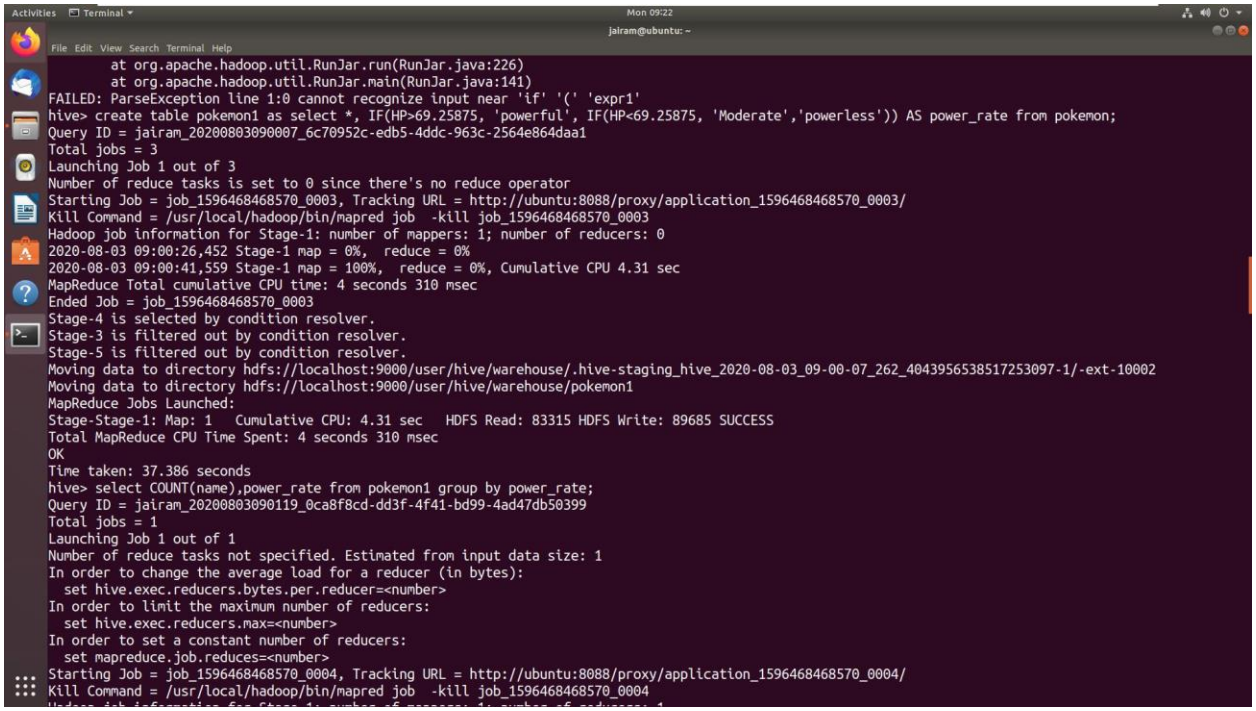


6. Create and insert values of existing table ‘pokemon’ into a new table ‘pokemon1’, with an additional column ‘power_rate’ to find the count of ‘powerful’ and ‘moderate’ from the table ‘pokemon1’.Now, based on the average hit points, we will create another column called ‘power_rate’.In order to segregate the Pokémon, we will use if condition inside the select statement, which will create one more column in our dataset. The if condition should be used in the following manner inside a Hive query.

Command: ***if(expr1,expr2,expr3)***

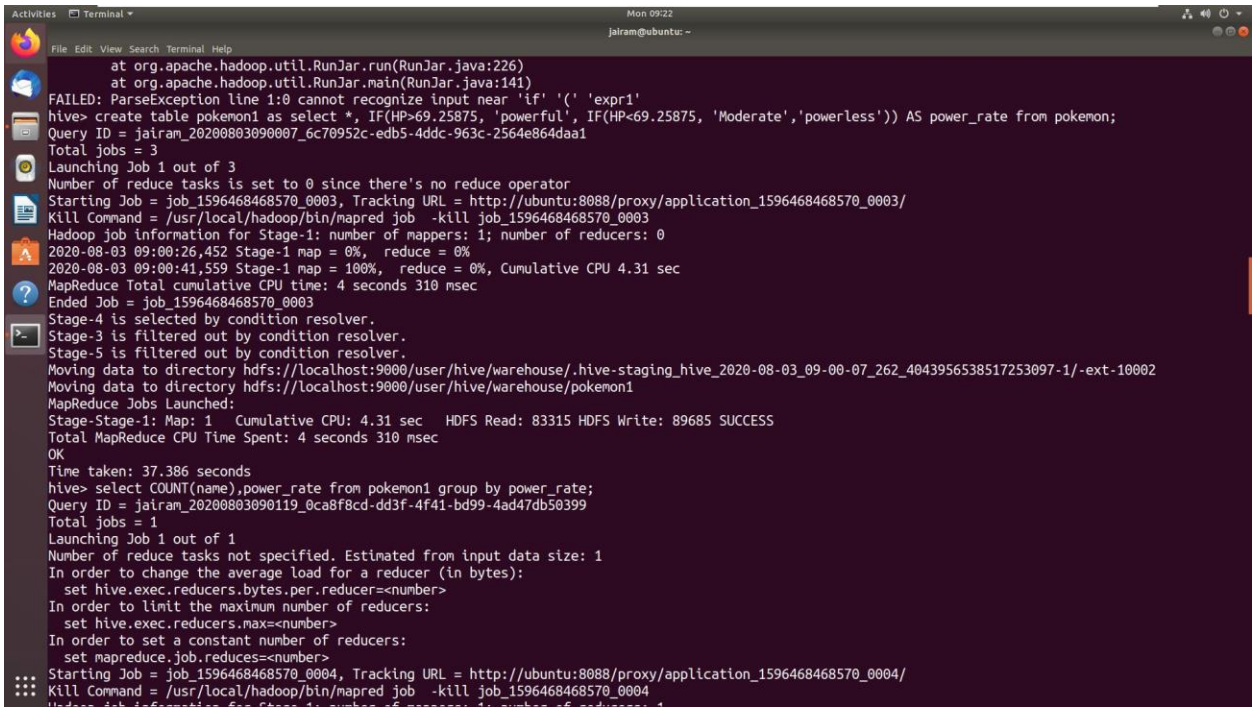
7. Now, we will create a table based on the condition that if the HP is greater than the average HP, then it is powerful, and if the HP is less than the average, then it is Moderate and a neutral condition is considered as powerless. The same is given as a Hive query below.

Command: ***create table pokemon1 as select *, IF(HP>69.25875, 'powerful', IF(HP<69.25875, 'Moderate','powerless')) AS power_rate from pokemon;***



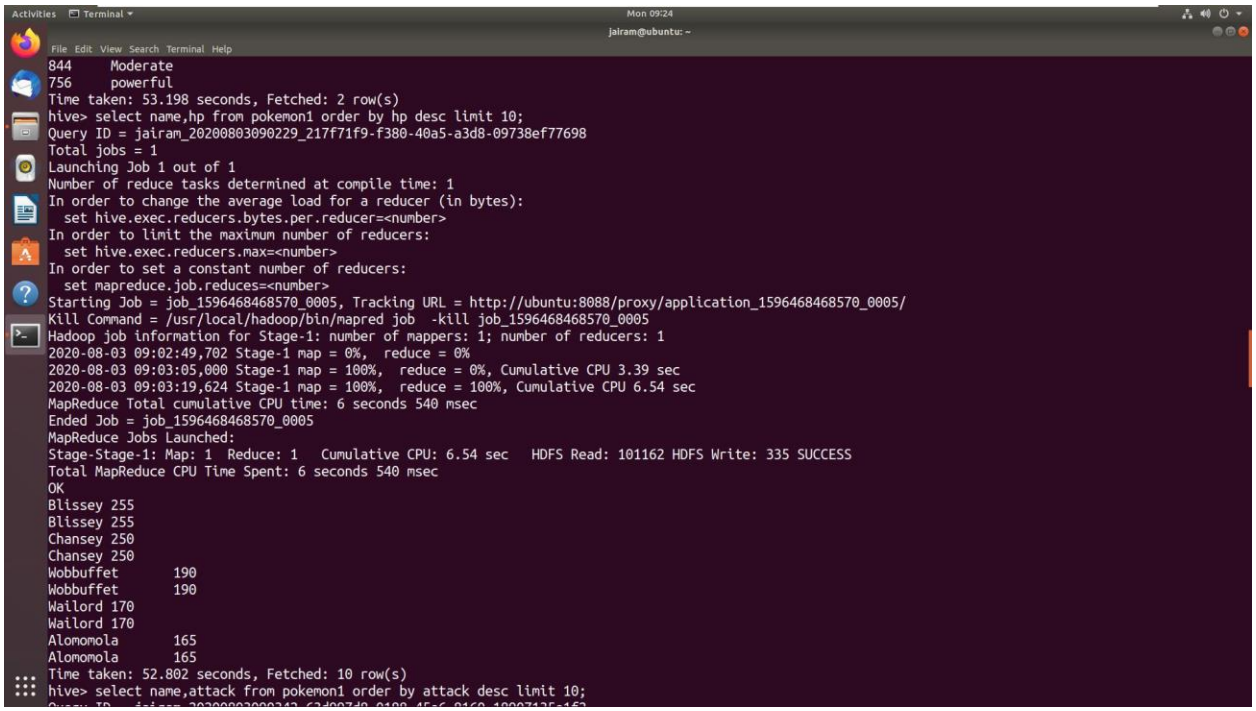
8. With the above query, a new column power_rate has been created.Now, we will find out the number of powerful and moderate HP Pokémons present, using the below query.

Command: ***select COUNT(name),power_rate from pokemon1 group by power_rate;***



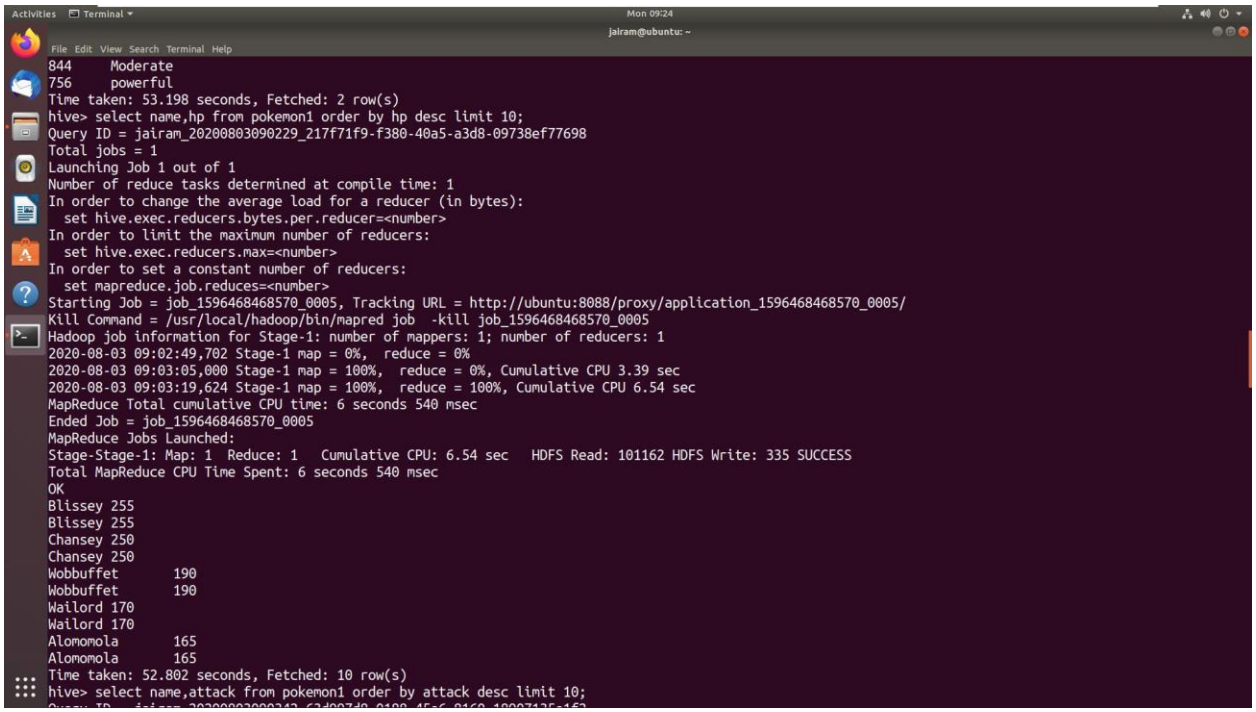
9. Find out the top 10 Pokémons according to their HP's using the below query.

Command: ***select name,hp from pokemon1 order by hp desc limit 10;***



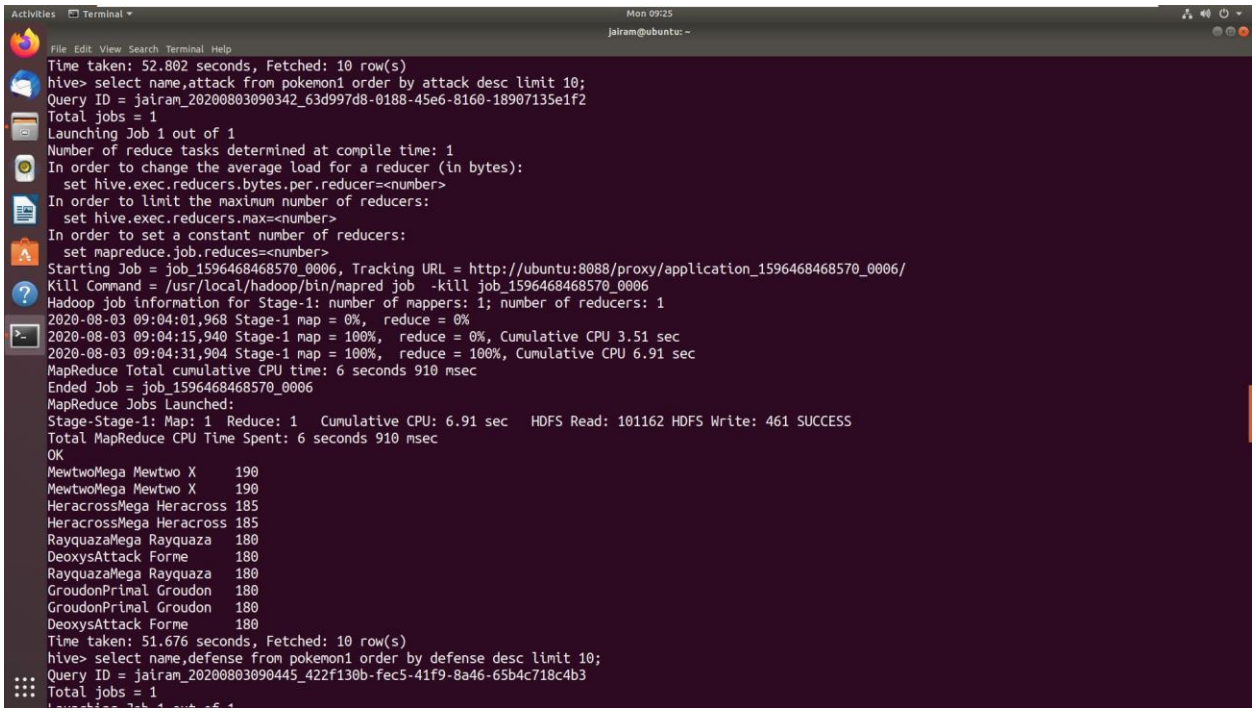
10.Find out the top 10 Pokémons based on their Attack stat, using the below query.

Command: ***select name,attack from pokemon1 order by attack desc limit 10;***



11.Find out the top 10 Pokémons based on their Defense stat, using the below query. Now, we will see the list of top 10 Pokémons according to their defense, using the below query.

Command: ***select name,defense from pokemon1 order by defense desc limit 10;***

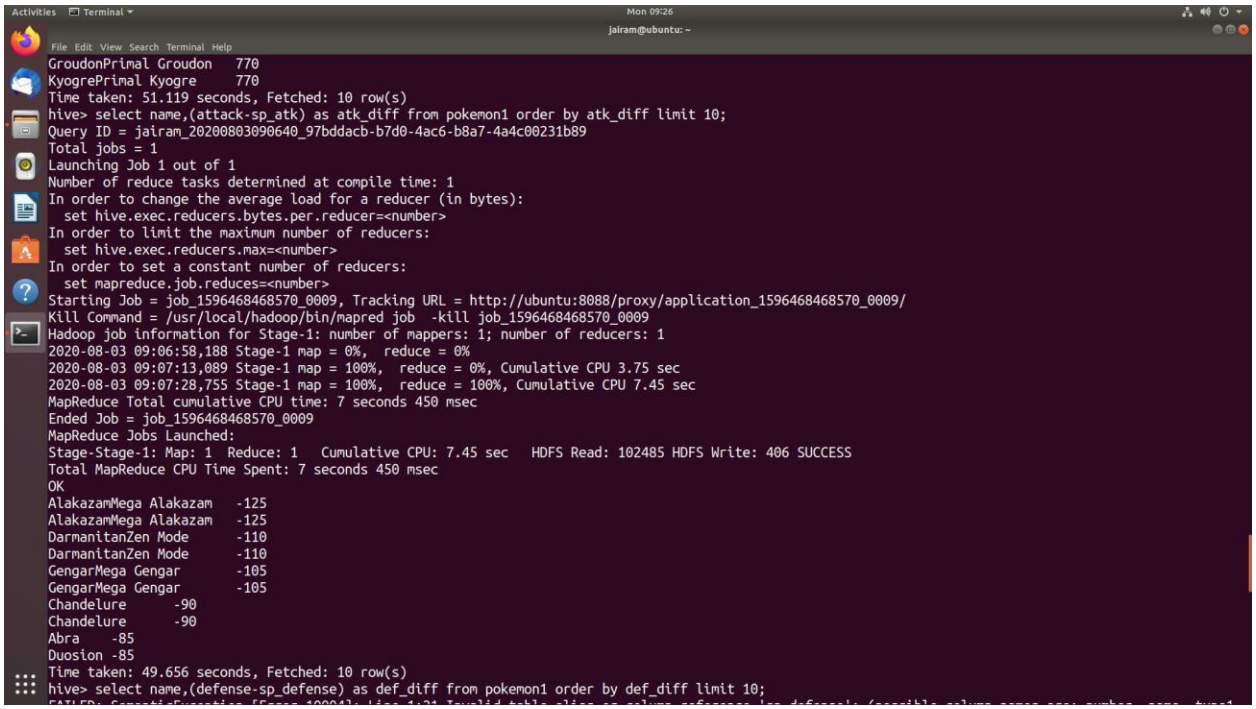


12.Find out the top 10 Pokémons based on their total power.

Command: ***select name,total from pokemon1 order by total desc limit 10;***

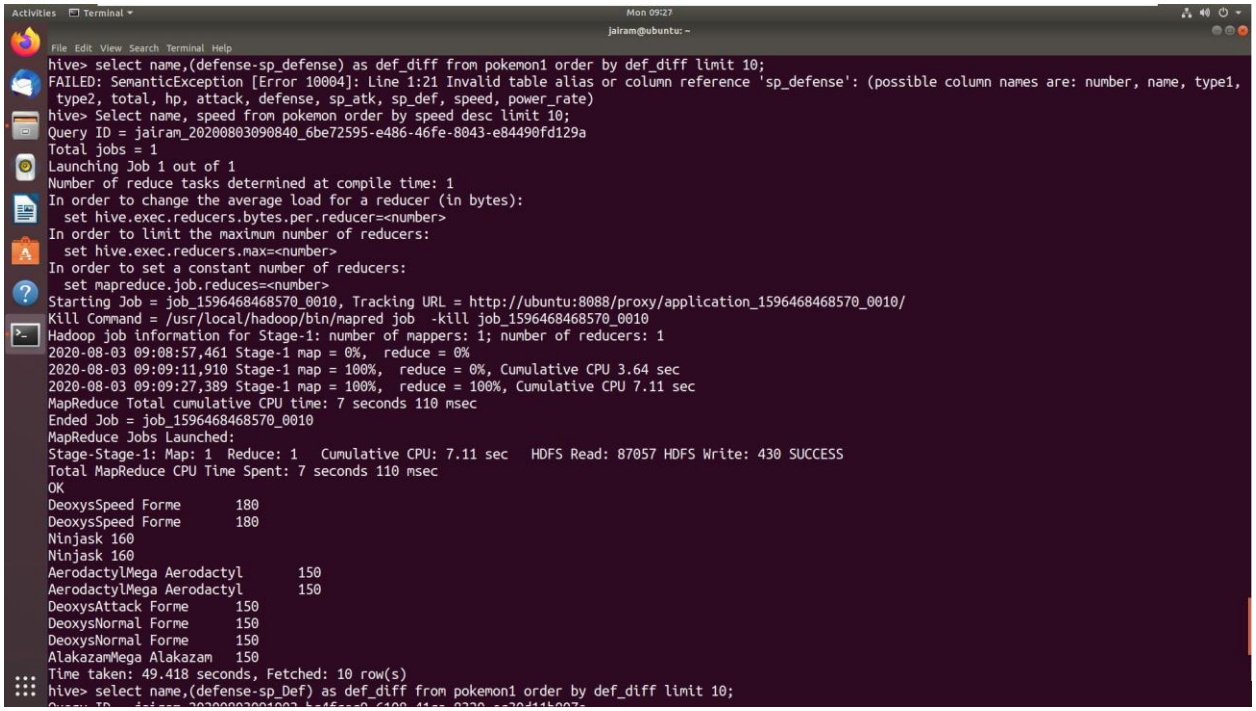
13.Find out the top 10 Pokémons having a drastic change in their attack and sp.attack, using the below query.

Command: ***select name,(attack-sp_atk) as atk_diff from pokemon1 order by atk_diff limit 10;***



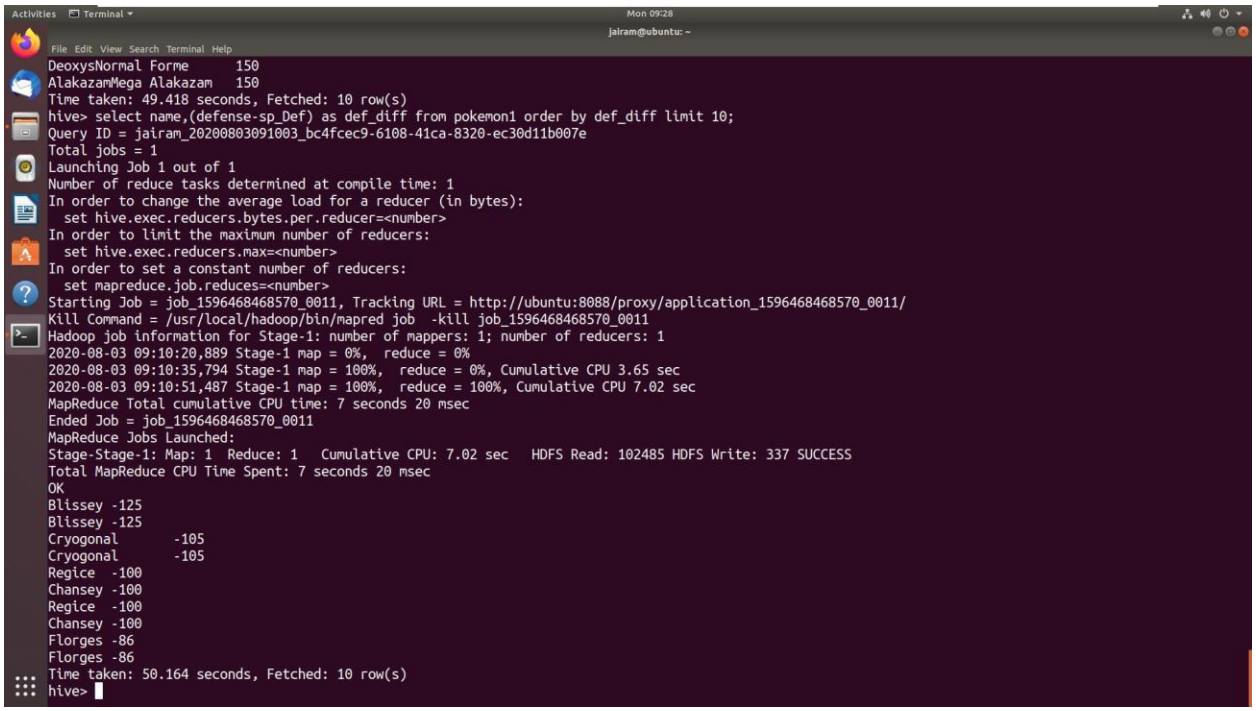
14.Find out the top 10 Pokémons having a drastic change in their defense and sp.defense, using the below query.

Command***: select name,(defense-sp_defense) as def_diff from pokemon1 order by def_diff limit 10;***



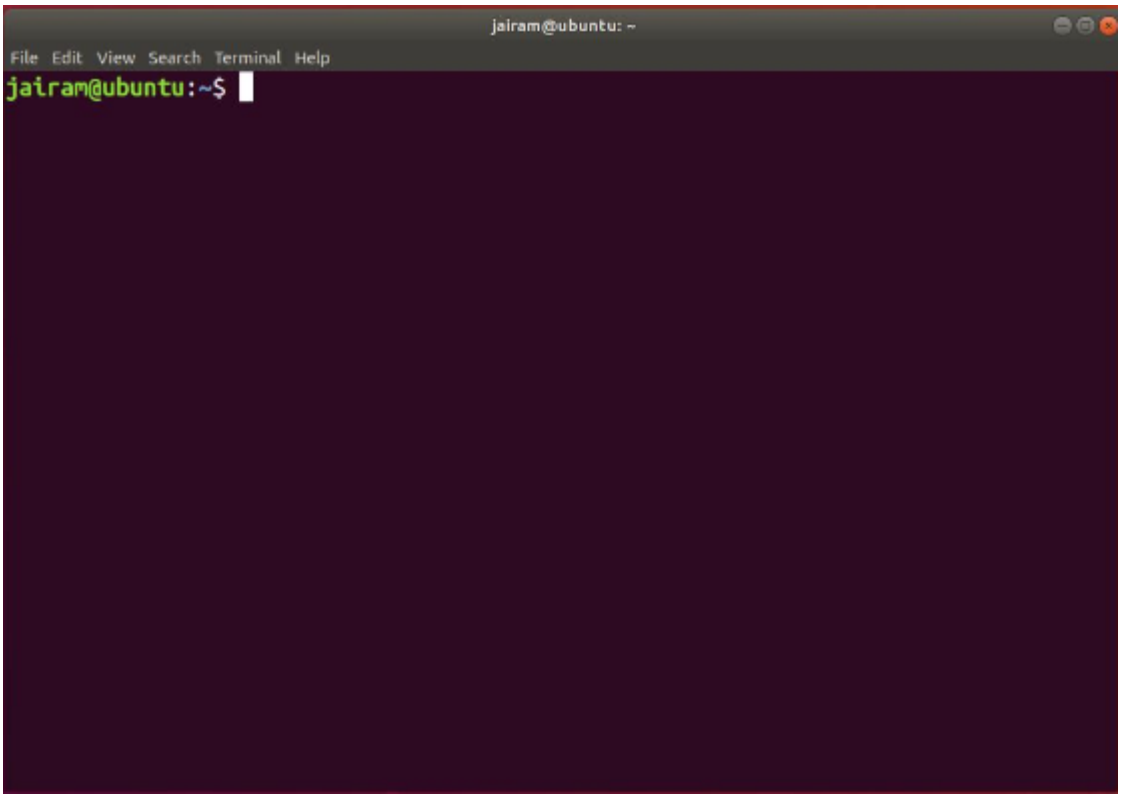
15.Find out the top 10 fastest Pokémons, using the below query.

Command: ***Select name, speed from pokemon order by speed desc limit 10;***


Activities
Terminal
Mon 09:28
jairam@ubuntu: ~
File Edit View Search Terminal Help
DeoxysNormal Forme 150
AlakazamMega Alakazam 150
Time taken: 49.418 seconds, Fetched: 10 row(s)
hive> select name,(defense-sp_Def) as def_diff from pokemon1 order by def_diff limit 10;
Query ID = jairam_20200803091003_bc4fcec9-6108-41ca-8320-ec30d11b007e
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_1596468468570_0011, Tracking URL = http://ubuntu:8088/proxy/application_1596468468570_0011/
Kill Command = /usr/local/hadoop/bin/mapred job -kill job_1596468468570_0011
Hadoop job information for Stage-1: number of mappers: 1; number of reducers: 1
2020-08-03 09:10:20,889 Stage-1 map = 0%, reduce = 0%
2020-08-03 09:10:35,794 Stage-1 map = 100%, reduce = 0%, Cumulative CPU 3.65 sec
2020-08-03 09:10:51,487 Stage-1 map = 100%, reduce = 100%, Cumulative CPU 7.02 sec
MapReduce Total cumulative CPU time: 7 seconds 20 msec
Ended Job = job_1596468468570_0011
MapReduce Jobs Launched:
Stage-Stage-1: Map: 1 Reduce: 1 Cumulative CPU: 7.02 sec HDFS Read: 102485 HDFS Write: 337 SUCCESS
Total MapReduce CPU Time Spent: 7 seconds 20 msec
OK
Blissey -125
Blissey -125
Cryogonal -105
Cryogonal -105
Regice -100
Chansey -100
Regice -100
Chansey -100
Florges -86
Florges -86
Time taken: 50.164 seconds, Fetched: 10 row(s)
hive>

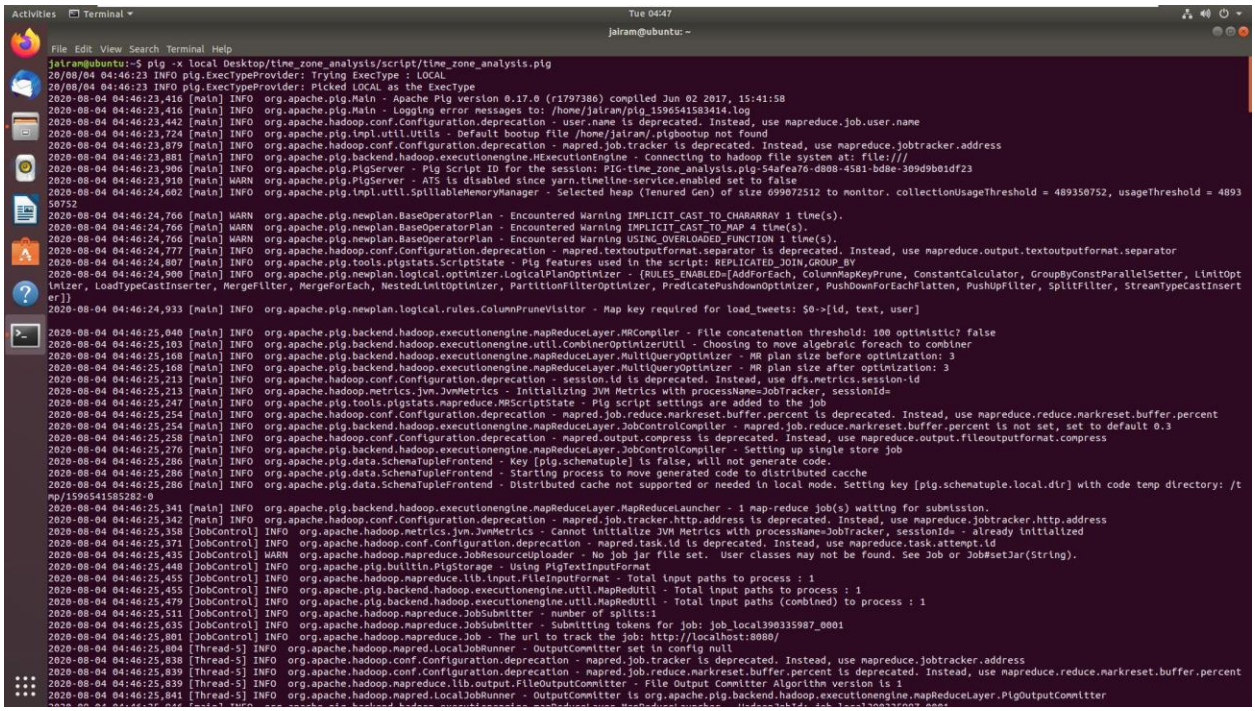
Time zone analysis

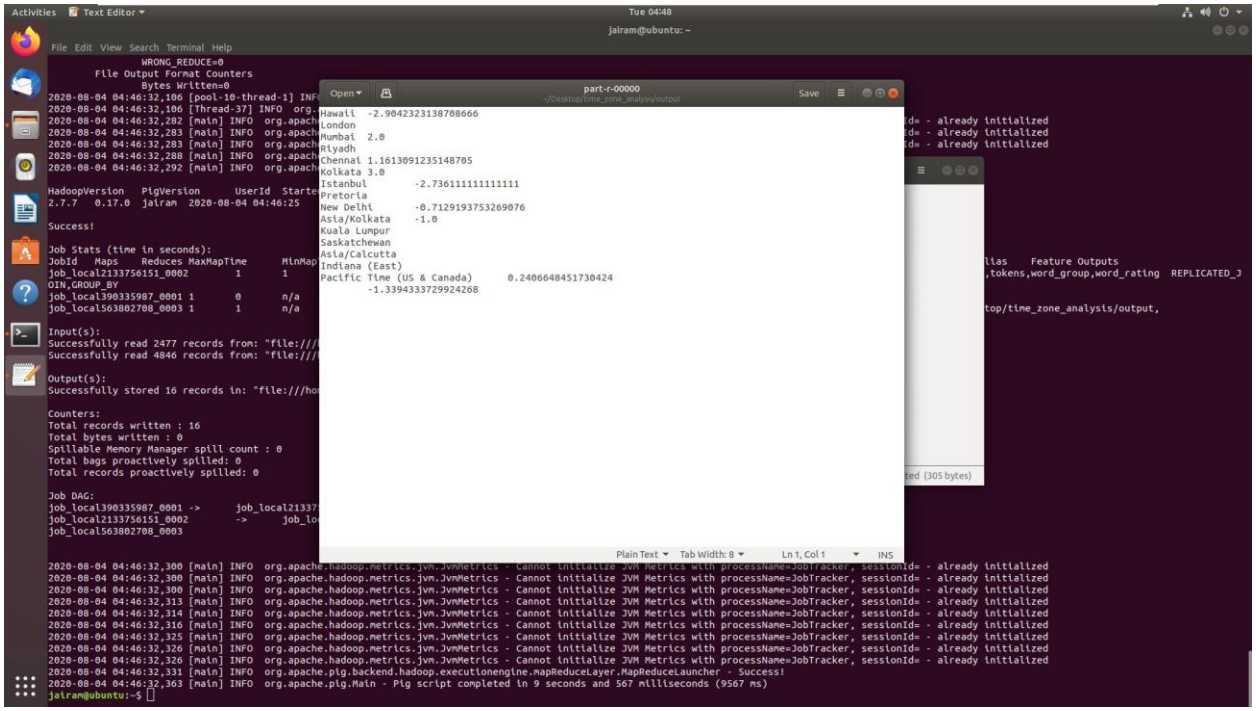
1. Paste the time_zone_analysis folder on Desktop
2. Open terminal



3. run the script file

Command :***pig -x local Desktop/time_zone_analysis/script/time_zone_analysis.pig***

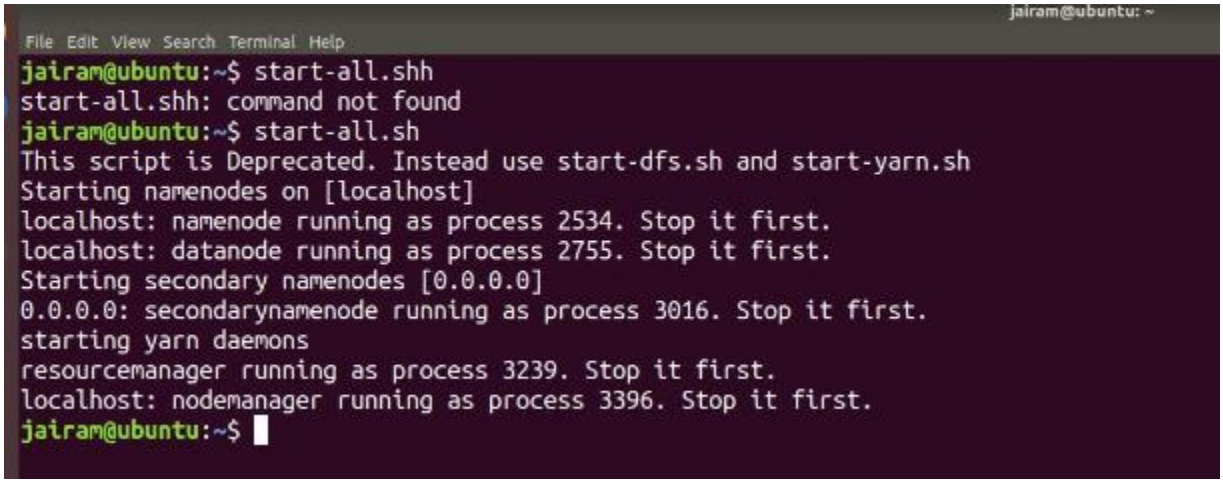

Activities
Terminal
Tue 04:47
jairam@ubuntu: ~
File Edit View Search Terminal Help
jairam@ubuntu:~$ pig -x local Desktop/time_zone_analysis/script/time_zone_analysis.pig
20/08/04 04:46:23 INFO pig.ExecTypeProvider: Trying ExecType : LOCAL
20/08/04 04:46:23 INFO pig.ExecTypeProvider: Picked LOCAL as the ExecType
2020-08-04 04:46:23,416 [main] INFO org.apache.pig.Main - Apache Pig version 0.17.0 (r1797386) compiled Jun 02 2017, 15:41:58
2020-08-04 04:46:23,416 [main] INFO org.apache.pig.Main - Logging error messages to: /home/jairam/pig_1596541583414.log
2020-08-04 04:46:23,442 [main] INFO org.apache.hadoop.conf.Configuration.deprecation - user.name is deprecated. Instead, use mapreduce.job.user.name
2020-08-04 04:46:23,724 [main] INFO org.apache.pig.impl.util.Utils - Default bootup file /home/jairam/.pigbootup not found
2020-08-04 04:46:23,879 [main] INFO org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.address
2020-08-04 04:46:23,881 [main] INFO org.apache.pig.backend.hadoop.executionengine.HExecutionEngine - Connecting to hadoop file system at: file:///
2020-08-04 04:46:23,906 [main] INFO org.apache.pig.PigServer - Pig Script ID for the session: PIG-time_zone_analysis.pig-54afea76-d808-4581-bd8e-309d9b01df23
2020-08-04 04:46:23,910 [main] WARN org.apache.pig.PigServer - ATS is disabled since yarn.timeline-service.enabled set to false
2020-08-04 04:46:24,602 [main] INFO org.apache.pig.impl.util.SpillableMemoryManager - Selected heap (Tenured Gen) of size 699072512 to monitor. collectionUsageThreshold = 489350752, usageThreshold = 489350752
2020-08-04 04:46:24,766 [main] WARN org.apache.pig.newplan.BaseOperatorPlan - Encountered Warning IMPLICIT_CAST_TO_CHARARRAY 1 time(s).
2020-08-04 04:46:24,766 [main] WARN org.apache.pig.newplan.BaseOperatorPlan - Encountered Warning IMPLICIT_CAST_TO_MAP 4 time(s).
2020-08-04 04:46:24,766 [main] WARN org.apache.pig.newplan.BaseOperatorPlan - Encountered Warning USING_OVERLOADED_FUNCTION 1 time(s).
2020-08-04 04:46:24,777 [main] INFO org.apache.hadoop.conf.Configuration.deprecation - mapred.textoutputformat.separator is deprecated. Instead, use mapreduce.output.textoutputformat.separator
2020-08-04 04:46:24,807 [main] INFO org.apache.pig.tools.pigstats.ScriptState - Pig features used in the script: REPLICATED_JOIN,GROUP_BY
2020-08-04 04:46:24,900 [main] INFO org.apache.pig.newplan.logical.optimizer.LogicalPlanOptimizer - {RULES_ENABLED=[AddForEach, ColumnMapKeyPrune, ConstantCalculator, GroupByConstParallelSetter, LimitOptimizer, LoadTypeCastInserter, MergeFilter, MergeForEach, NestedLimitOptimizer, PartitionFilterOptimizer, PredicatePushdownOptimizer, PushDownForEachFlatten, PushUpFilter, SplitFilter, StreamTypeCastInserter]}
2020-08-04 04:46:24,933 [main] INFO org.apache.pig.newplan.logical.rules.ColumnPruneVisitor - Map key required for load_tweets: $0->[id, text, user]
2020-08-04 04:46:25,040 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MRCompiler - File concatenation threshold: 100 optimistic? false
2020-08-04 04:46:25,103 [main] INFO org.apache.pig.backend.hadoop.executionengine.util.CombinerOptimizerUtil - Choosing to move algebraic foreach to combiner
2020-08-04 04:46:25,168 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MultiQueryOptimizer - MR plan size before optimization: 3
2020-08-04 04:46:25,168 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MultiQueryOptimizer - MR plan size after optimization: 3
2020-08-04 04:46:25,213 [main] INFO org.apache.hadoop.conf.Configuration.deprecation - session.id is deprecated. Instead, use dfs.metrics.session-id
2020-08-04 04:46:25,213 [main] INFO org.apache.hadoop.metrics.jvm.JvmMetrics - Initializing JVM Metrics with processName=JobTracker, sessionId=
2020-08-04 04:46:25,247 [main] INFO org.apache.pig.tools.pigstats.mapreduce.MRScriptState - Pig script settings are added to the job
2020-08-04 04:46:25,254 [main] INFO org.apache.hadoop.conf.Configuration.deprecation - mapred.job.reduce.markreset.buffer.percent is deprecated. Instead, use mapreduce.reduce.markreset.buffer.percent
2020-08-04 04:46:25,254 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.JobControlCompiler - mapred.job.reduce.markreset.buffer.percent is not set, set to default 0.3
2020-08-04 04:46:25,258 [main] INFO org.apache.hadoop.conf.Configuration.deprecation - mapred.output.compress is deprecated. Instead, use mapreduce.output.fileoutputformat.compress
2020-08-04 04:46:25,276 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.JobControlCompiler - Setting up single store job
2020-08-04 04:46:25,286 [main] INFO org.apache.pig.data.SchemaTupleFrontend - Key [pig.schematuple] is false, will not generate code.
2020-08-04 04:46:25,286 [main] INFO org.apache.pig.data.SchemaTupleFrontend - Starting process to move generated code to distributed cacche
2020-08-04 04:46:25,286 [main] INFO org.apache.pig.data.SchemaTupleFrontend - Distributed cache not supported or needed in local mode. Setting key [pig.schematuple.local.dir] with code temp directory: /tmp/1596541585282-0
2020-08-04 04:46:25,341 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - 1 map-reduce job(s) waiting for submission.
2020-08-04 04:46:25,342 [main] INFO org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker.http.address is deprecated. Instead, use mapreduce.jobtracker.http.address
2020-08-04 04:46:25,358 [JobControl] INFO org.apache.hadoop.metrics.jvm.JvmMetrics - Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
2020-08-04 04:46:25,371 [JobControl] INFO org.apache.hadoop.conf.Configuration.deprecation - mapred.task.id is deprecated. Instead, use mapreduce.task.attempt.id
2020-08-04 04:46:25,435 [JobControl] WARN org.apache.hadoop.mapreduce.JobResourceUploader - No job jar file set. User classes may not be found. See Job or Job#setJar(String).
2020-08-04 04:46:25,448 [JobControl] INFO org.apache.pig.builtin.PigStorage - Using PigTextInputFormat
2020-08-04 04:46:25,455 [JobControl] INFO org.apache.hadoop.mapreduce.lib.input.FileInputFormat - Total input paths to process : 1
2020-08-04 04:46:25,455 [JobControl] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths to process : 1
2020-08-04 04:46:25,479 [JobControl] INFO org.apache.pig.backend.hadoop.executionengine.util.MapRedUtil - Total input paths (combined) to process : 1
2020-08-04 04:46:25,511 [JobControl] INFO org.apache.hadoop.mapreduce.JobSubmitter - number of splits:1
2020-08-04 04:46:25,635 [JobControl] INFO org.apache.hadoop.mapreduce.JobSubmitter - Submitting tokens for job: job_local390335987_0001
2020-08-04 04:46:25,801 [JobControl] INFO org.apache.hadoop.mapreduce.Job - The url to track the job: http://localhost:8080/
2020-08-04 04:46:25,804 [Thread-5] INFO org.apache.hadoop.mapred.LocalJobRunner - OutputCommitter set in config null
2020-08-04 04:46:25,838 [Thread-5] INFO org.apache.hadoop.conf.Configuration.deprecation - mapred.job.tracker is deprecated. Instead, use mapreduce.jobtracker.address
2020-08-04 04:46:25,839 [Thread-5] INFO org.apache.hadoop.conf.Configuration.deprecation - mapred.job.reduce.markreset.buffer.percent is deprecated. Instead, use mapreduce.reduce.markreset.buffer.percent
2020-08-04 04:46:25,839 [Thread-5] INFO org.apache.hadoop.mapreduce.lib.output.FileOutputCommitter - File Output Committer Algorithm version is 1
2020-08-04 04:46:25,841 [Thread-5] INFO org.apache.hadoop.mapred.LocalJobRunner - OutputCommitter is org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.PigOutputCommitter


Activities
Text Editor
Tue 04:48
jairam@ubuntu: ~
File Edit View Search Terminal Help
WRONG_REDUCE=0
File Output Format Counters
Bytes Written=0
2020-08-04 04:46:32,106 [pool-10-thread-1] INFO
2020-08-04 04:46:32,106 [Thread-37] INFO org.
2020-08-04 04:46:32,282 [main] INFO org.apach
2020-08-04 04:46:32,283 [main] INFO org.apach
2020-08-04 04:46:32,283 [main] INFO org.apach
2020-08-04 04:46:32,288 [main] INFO org.apach
2020-08-04 04:46:32,292 [main] INFO org.apach
HadoopVersion PigVersion UserId StartedAt
2.7.7 0.17.0 jairam 2020-08-04 04:46:25
Success!
Job Stats (time in seconds):
JobId Maps Reduces MaxMapTime MinMap
job_local2133756151_0002 1 1
OIN,GROUP_BY
job_local390335987_0001 1 0 n/a
job_local563802708_0003 1 1 n/a
Input(s):
Successfully read 2477 records from: "file:///
Successfully read 4846 records from: "file:///
Output(s):
Successfully stored 16 records in: "file:///ho
Counters:
Total records written : 16
Total bytes written : 0
Spillable Memory Manager spill count : 0
Total bags proactively spilled: 0
Total records proactively spilled: 0
Job DAG:
job_local390335987_0001 -> job_local2133
job_local2133756151_0002 -> job_lo
job_local563802708_0003
Open
part-r-00000
~/Desktop/time_zone_analysis/output
Save
Hawaii -2.9042323138708666
London
Mumbai 2.0
Riyadh
Chennai 1.161309123514870 5
Kolkata 3.0
Istanbul -2.736111111111111
Pretoria
New Delhi -0.7129193753269076
Asia/Kolkata -1.0
Kuala Lumpur
Saskatchewan
Asia/Calcutta
Indiana (East)
Pacific Time (US & Canada) 0.2406648451730424
-1.3394333729924268
Plain Text
Tab Width: 8
Ln 1, Col 1
INS
llas Feature Outputs
,tokens,word_group,word_rating REPLICATED_J
top/time_zone_analysis/output,
ted (305 bytes)
2020-08-04 04:46:32,300 [main] INFO org.apache.hadoop.metrics.jvm.JvmMetrics - Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
2020-08-04 04:46:32,313 [main] INFO org.apache.hadoop.metrics.jvm.JvmMetrics - Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
2020-08-04 04:46:32,314 [main] INFO org.apache.hadoop.metrics.jvm.JvmMetrics - Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
2020-08-04 04:46:32,316 [main] INFO org.apache.hadoop.metrics.jvm.JvmMetrics - Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
2020-08-04 04:46:32,325 [main] INFO org.apache.hadoop.metrics.jvm.JvmMetrics - Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
2020-08-04 04:46:32,326 [main] INFO org.apache.hadoop.metrics.jvm.JvmMetrics - Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
2020-08-04 04:46:32,326 [main] INFO org.apache.hadoop.metrics.jvm.JvmMetrics - Cannot initialize JVM Metrics with processName=JobTracker, sessionId= - already initialized
2020-08-04 04:46:32,331 [main] INFO org.apache.pig.backend.hadoop.executionengine.mapReduceLayer.MapReduceLauncher - Success!
2020-08-04 04:46:32,363 [main] INFO org.apache.pig.Main - Pig script completed in 9 seconds and 567 milliseconds (9567 ms)
jairam@ubuntu:~$

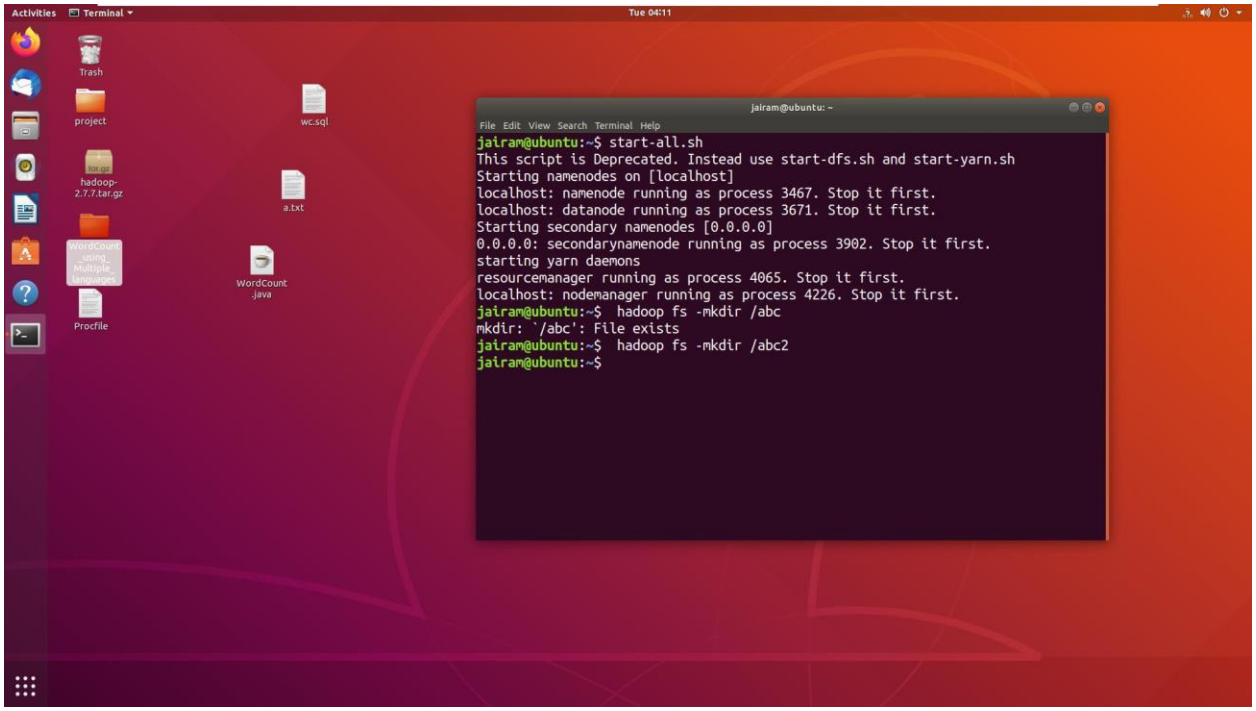
WordCount using java

1. Create a text file containing a paragraph i.e a.txt

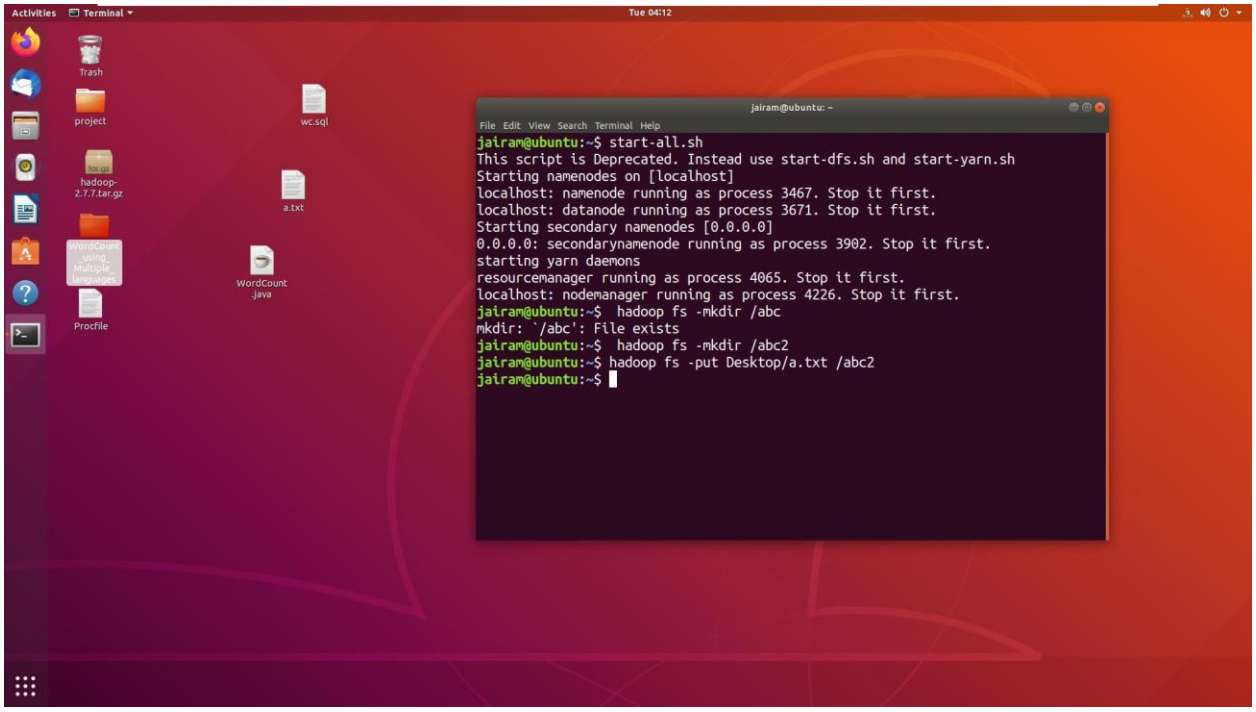
2. Open terminal and type Command: ***start-all.sh***



3. Command :**hadoop fs -mkdir /abc2**

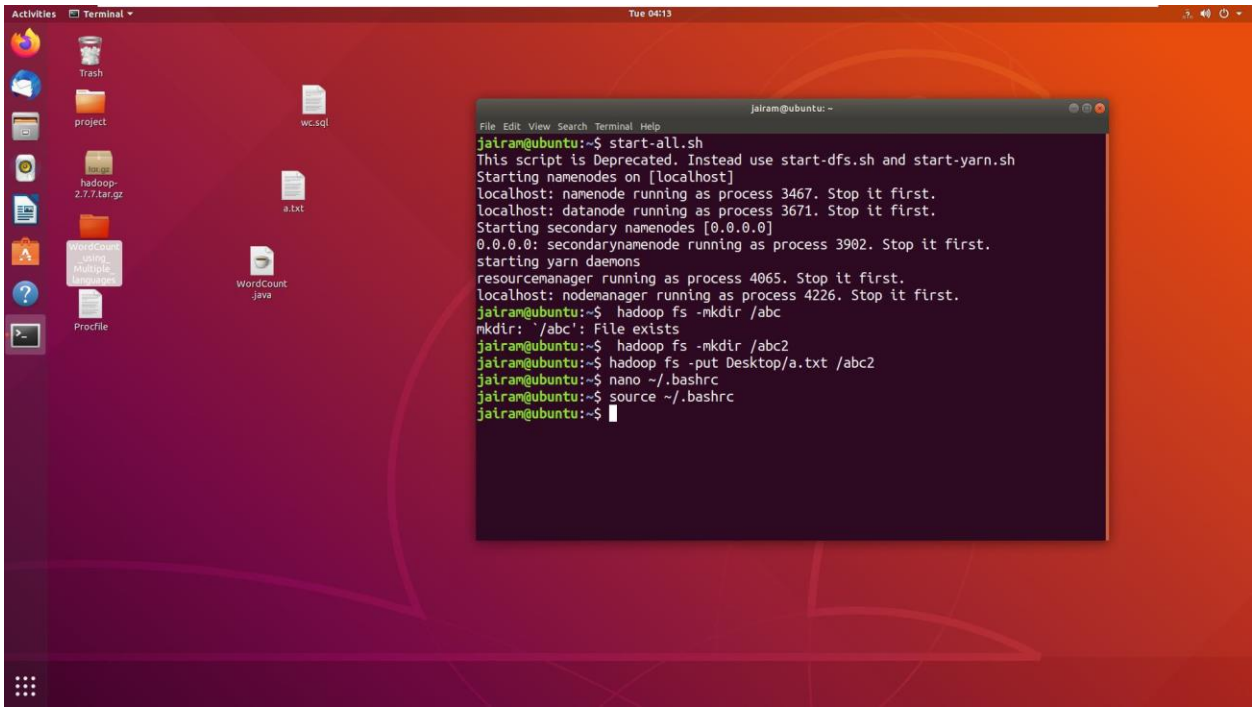


4. Command :***hadoop fs -put Desktop/a.txt /abc2***



5. Open bashrc file by ***nano ~/.bashrc*** and type

   export HADOOP_CLASSPATH=$JAVA_HOME/lib/tools.jar

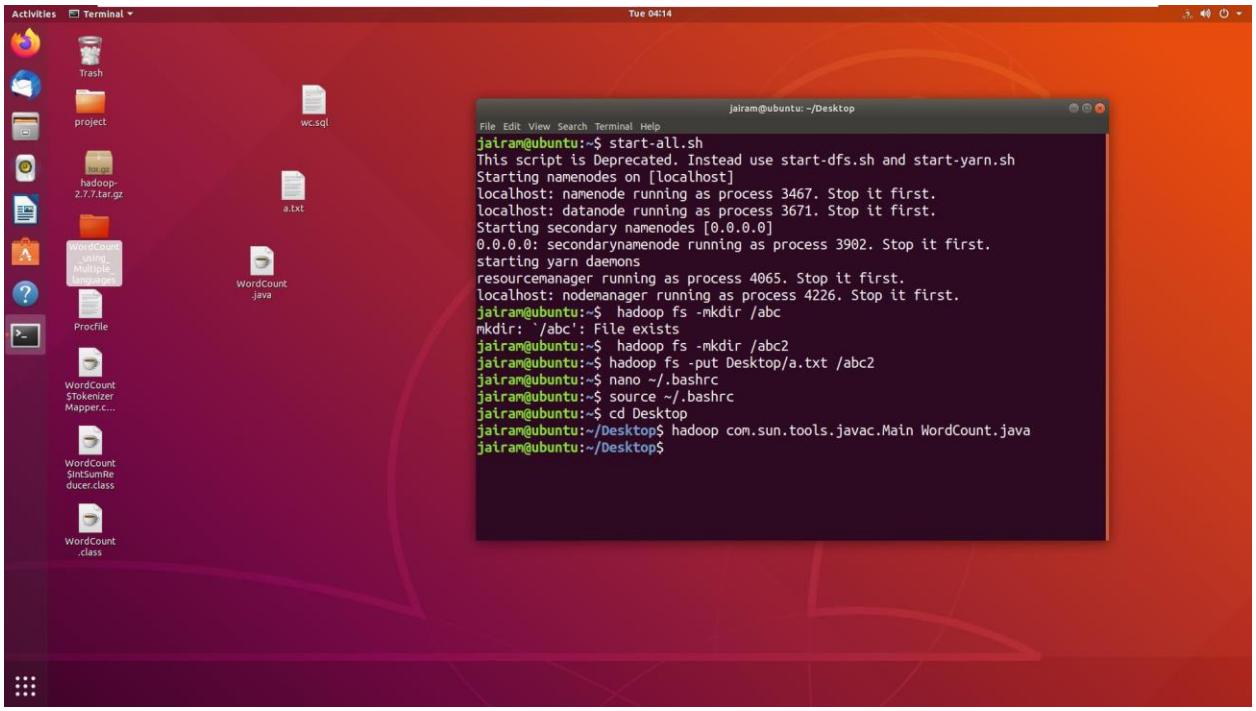


6. Then command :***source ~/.bashrc***

7. command :***cd Desktop***

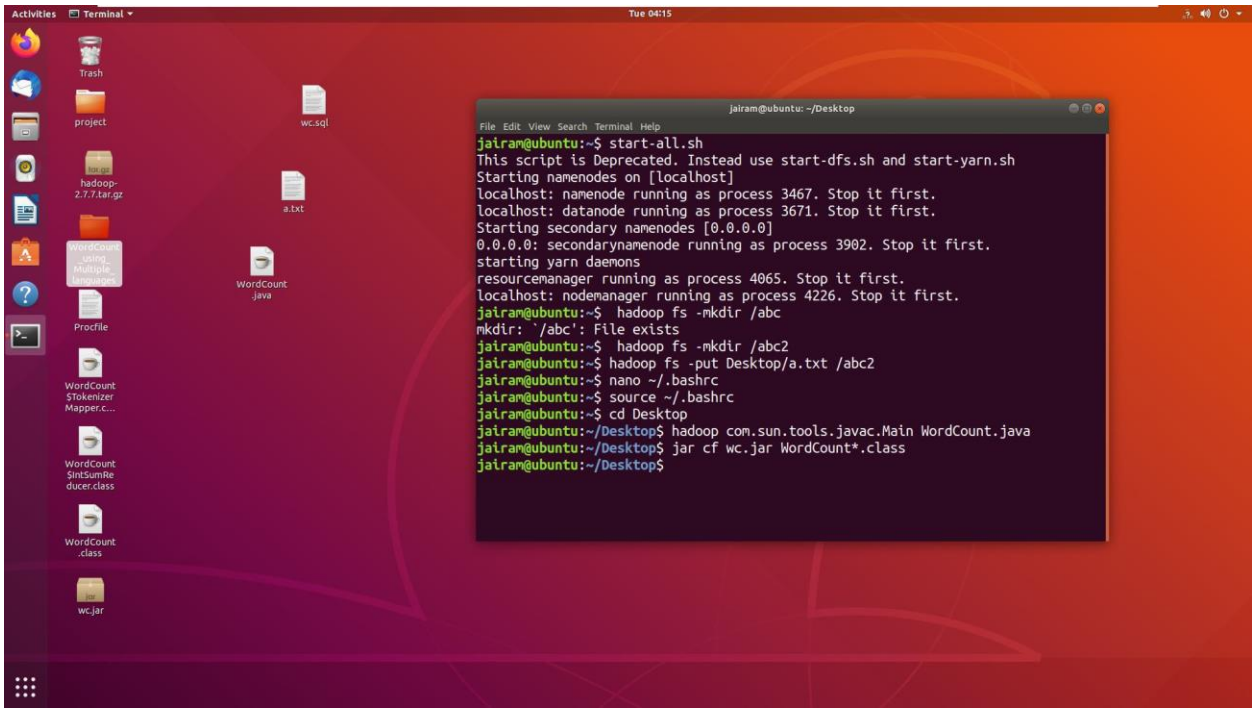
8. Now compile Wordcount.java file suppose that file is on Desktop

Command: ***hadoop com.sun.tools.javac.Main WordCount.java***



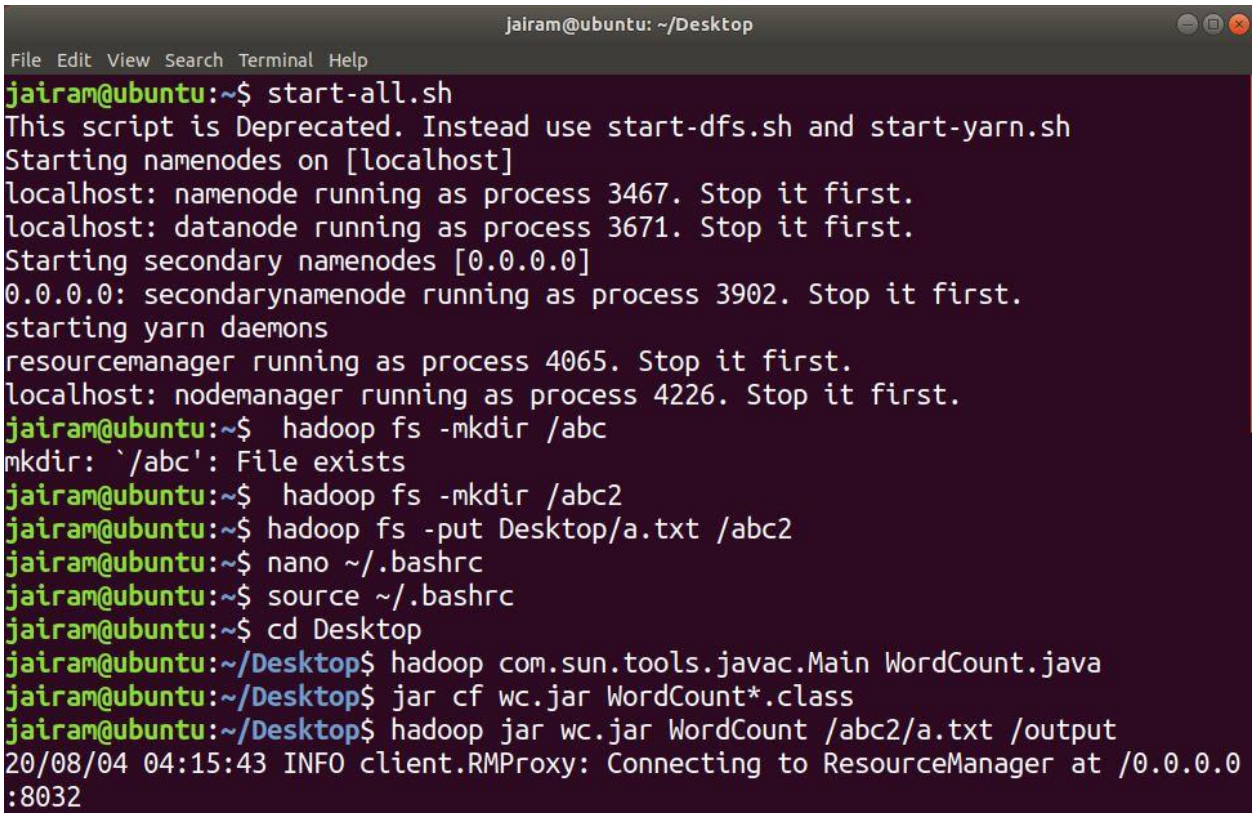
9. To combine all class files

Command:***jar cf wc.jar WordCount*.class***



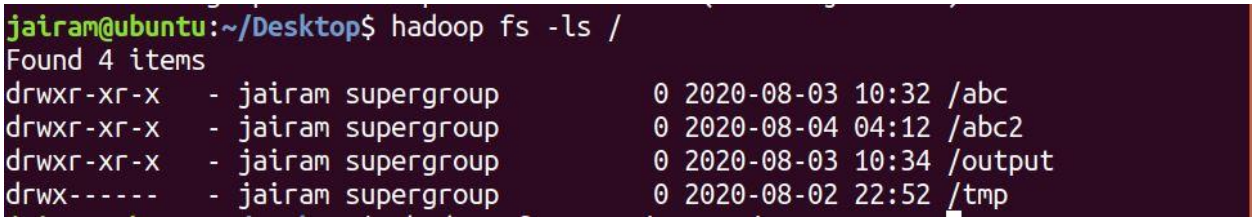
10. To execute

Command: ***hadoop jar wc.jar WordCount /abc2/a.txt /output***

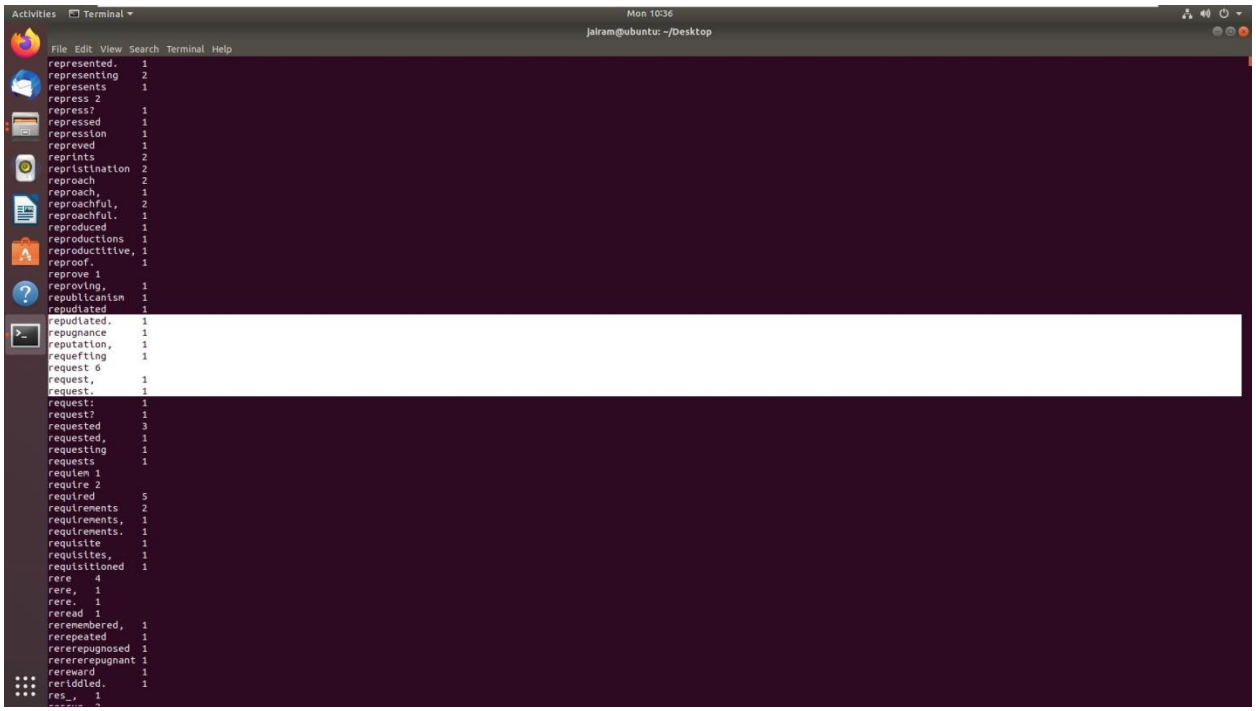


11. To check output folder is there or not

Command: ***hadoop fs -ls /***

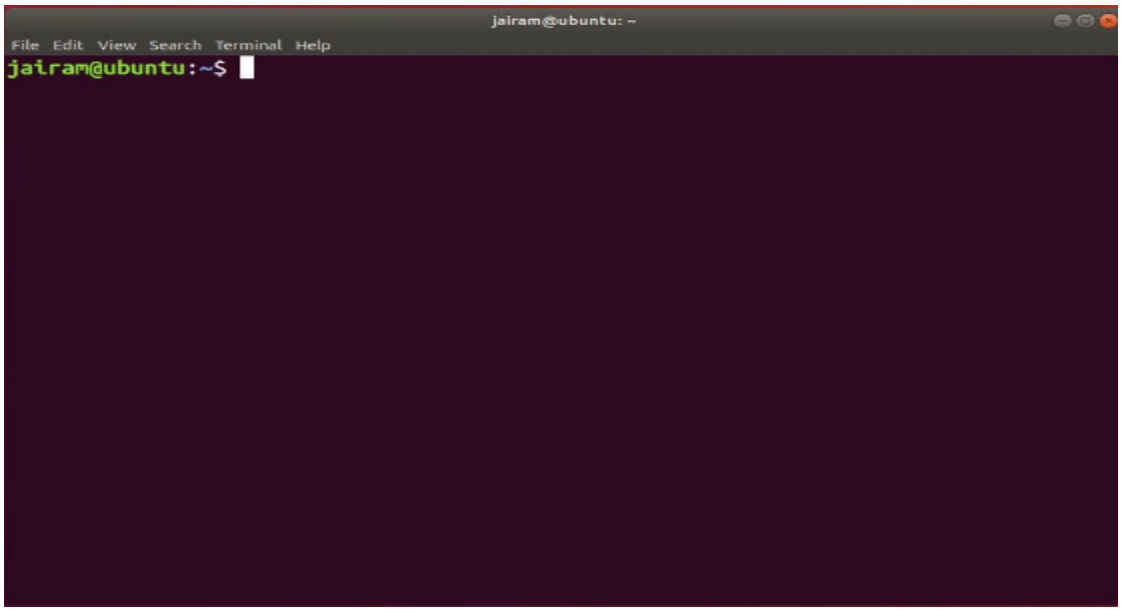


12.hadoop fs -cat /output/part-r-00000

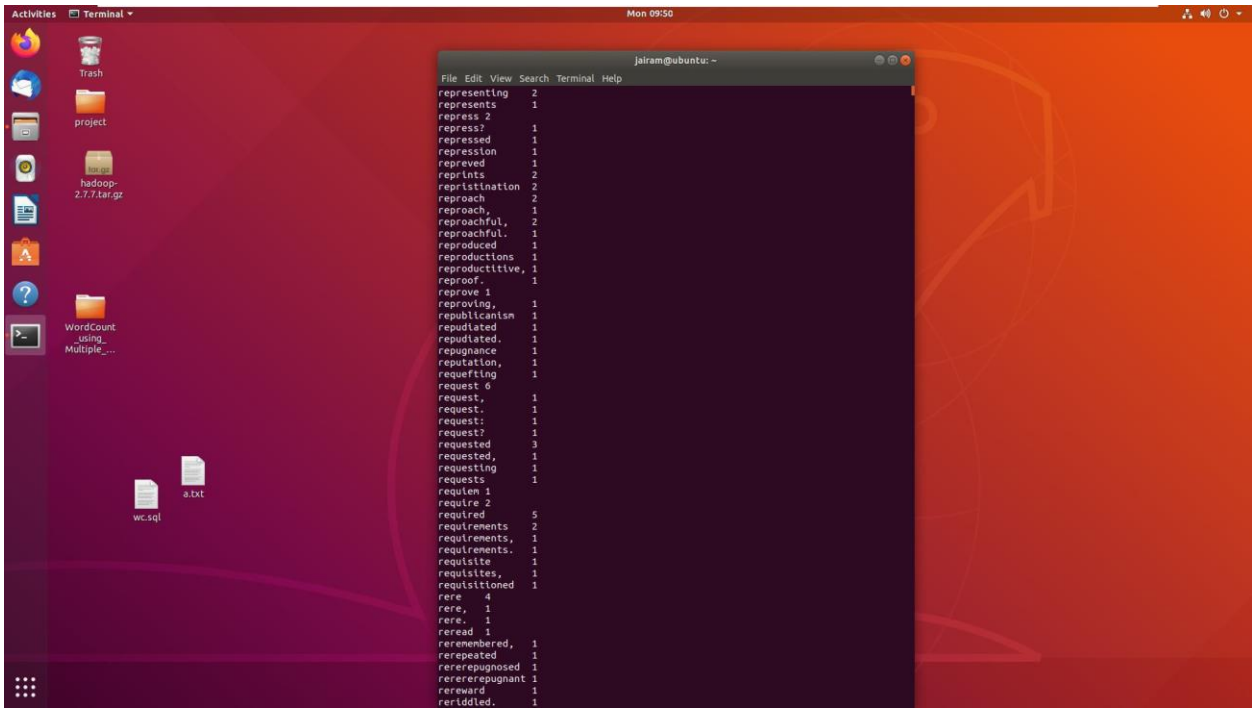

Activities
Terminal
Mon 10:36
jairam@ubuntu: ~/Desktop
File Edit View Search Terminal Help
represented. 1
representing 2
represents 1
repress 2
repress? 1
repressed 1
repression 1
repreved 1
reprints 2
repristination 2
reproach 2
reproach, 1
reproachful, 2
reproachful. 1
reproduced 1
reproductions 1
reproductitive, 1
reproof. 1
reprove 1
reproving, 1
republicanism 1
repudiated 1
repudiated. 1
repugnance 1
reputation, 1
requefting 1
request 6
request, 1
request. 1
request: 1
request? 1
requested 3
requested, 1
requesting 1
requests 1
requiem 1
require 2
required 5
requirements 2
requirements, 1
requirements. 1
requisite 1
requisites, 1
requisitioned 1
rere 4
rere, 1
rere. 1
reread 1
reremembered, 1
rerepeated 1
rererepugnosed 1
rererepugnant 1
rereward 1
reriddled. 1
res_ 1

WordCount using hive

1. . Open terminal

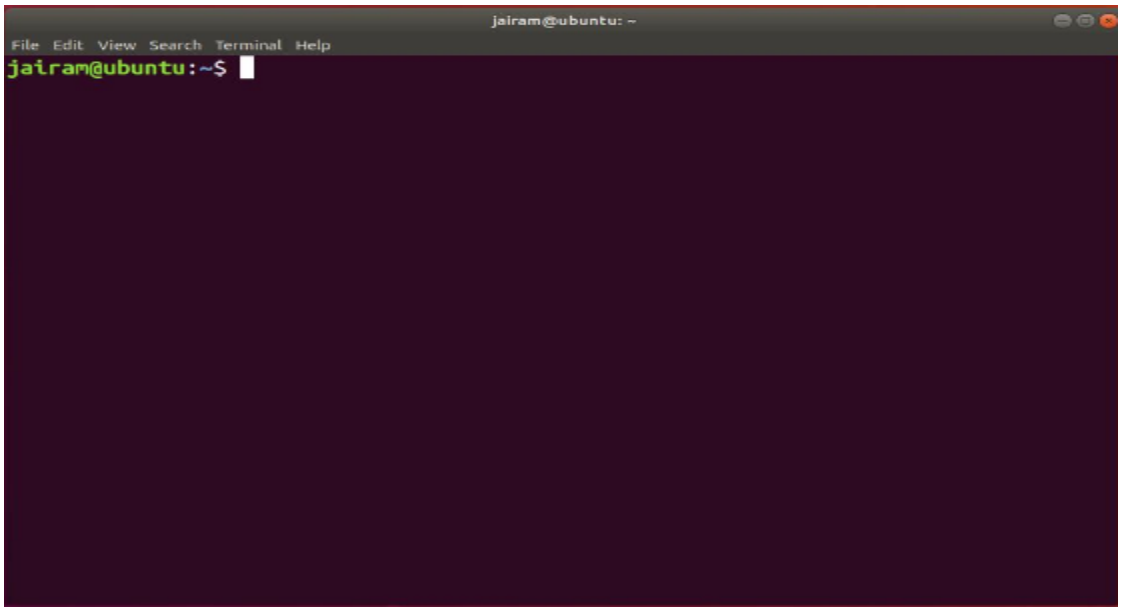


2. Command: ***hive -f Desktop/wc.sql***

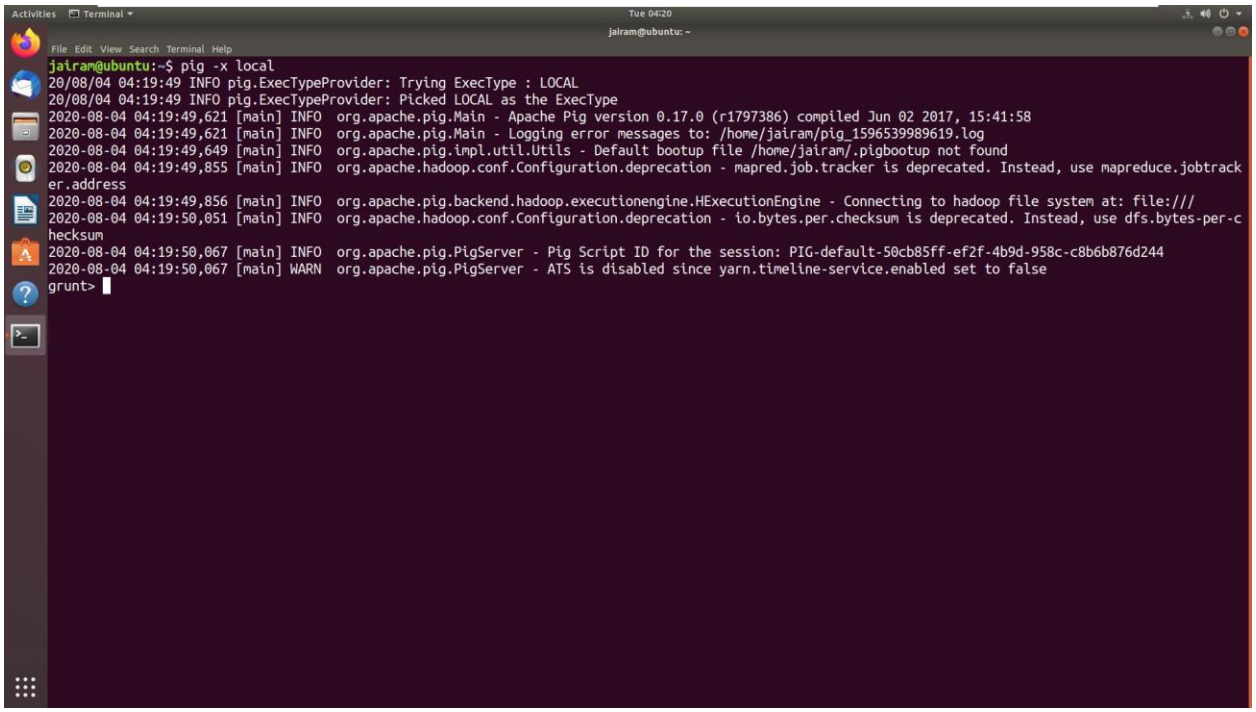


WordCount using pig

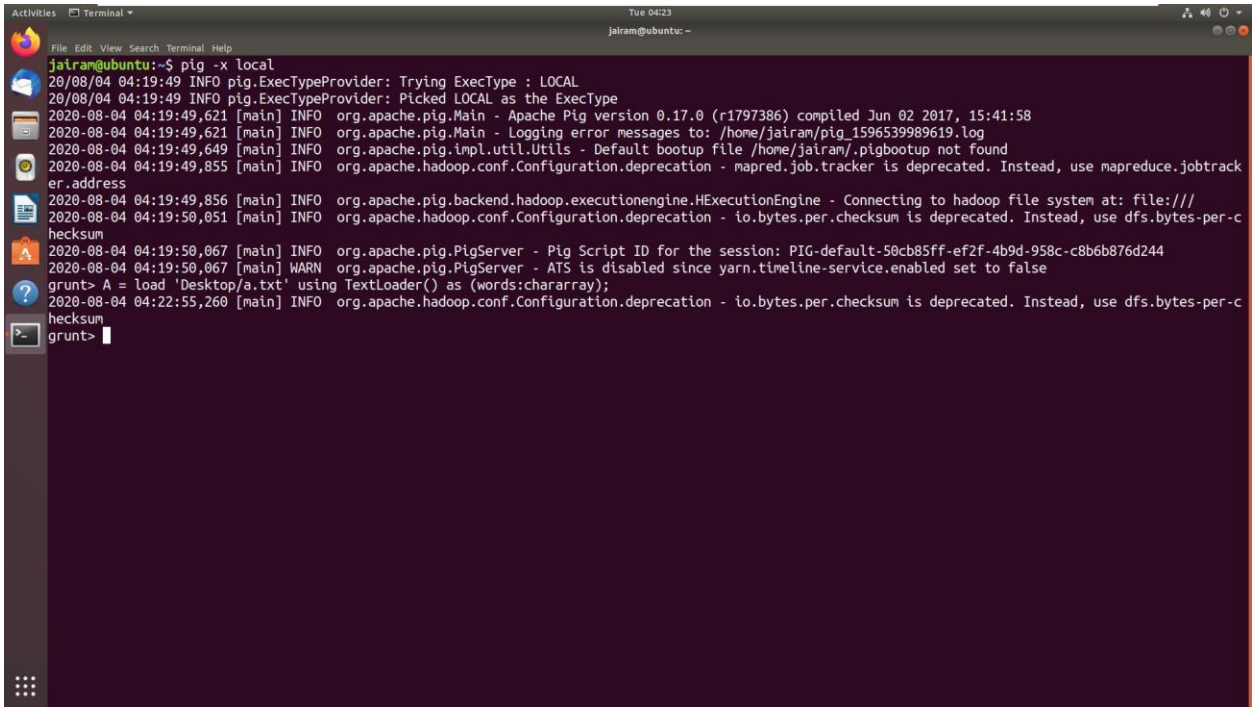
1. . Open terminal

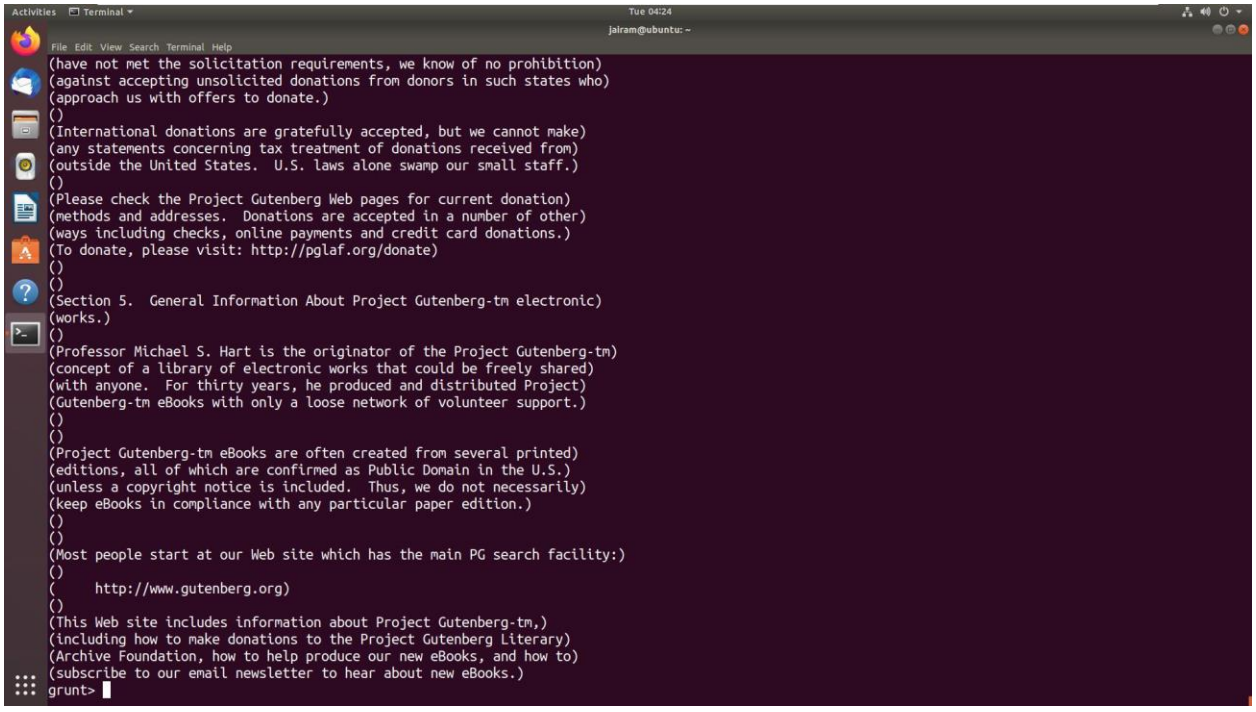


2. Command***: pig –x local*** to lauch the pig



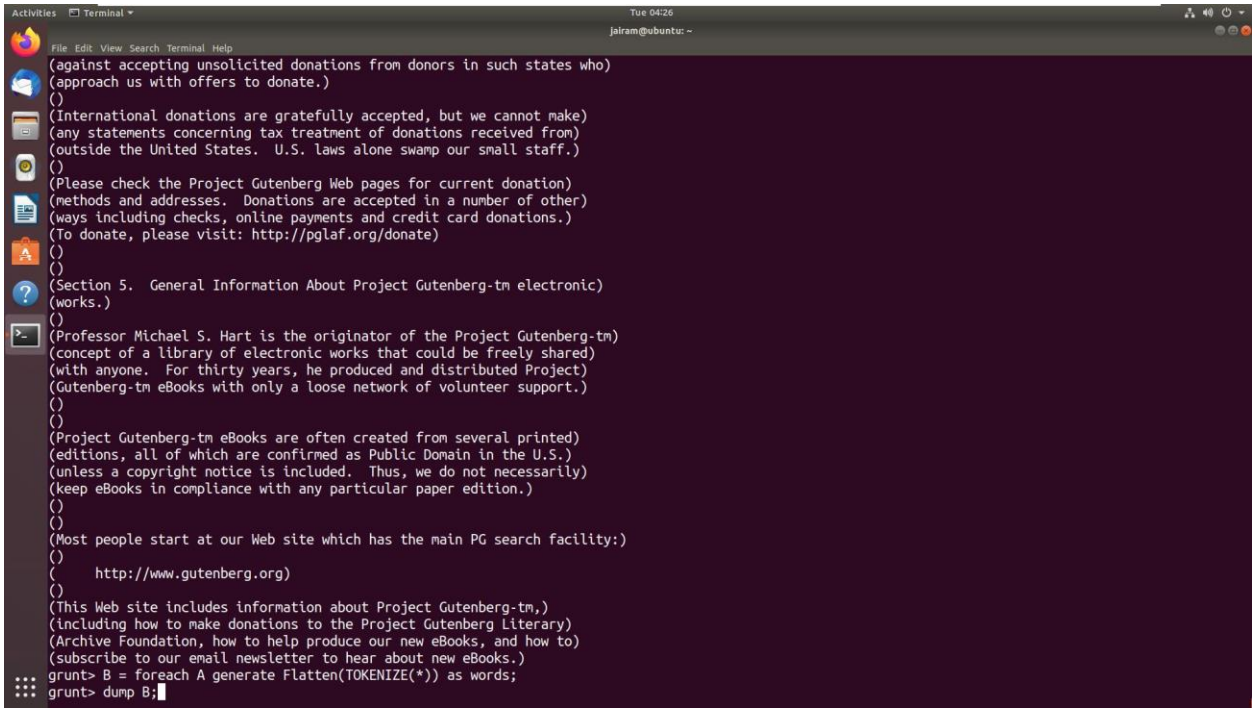
3. Command***: grunt > A = load 'Desktop/a.txt' using TextLoader() as (word:chararray);***





4. Command***: grunt > B = foreach A generate flatten(TOKENIZE(*)) as words;***

grunt> dump B;

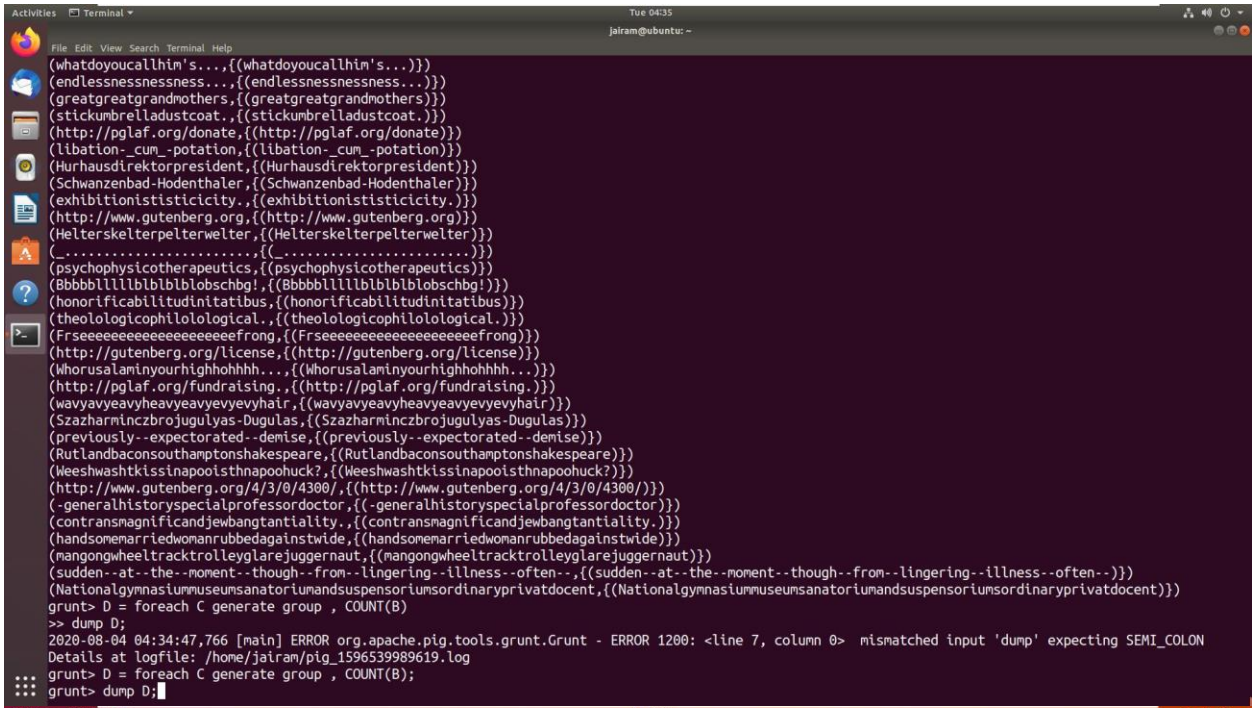


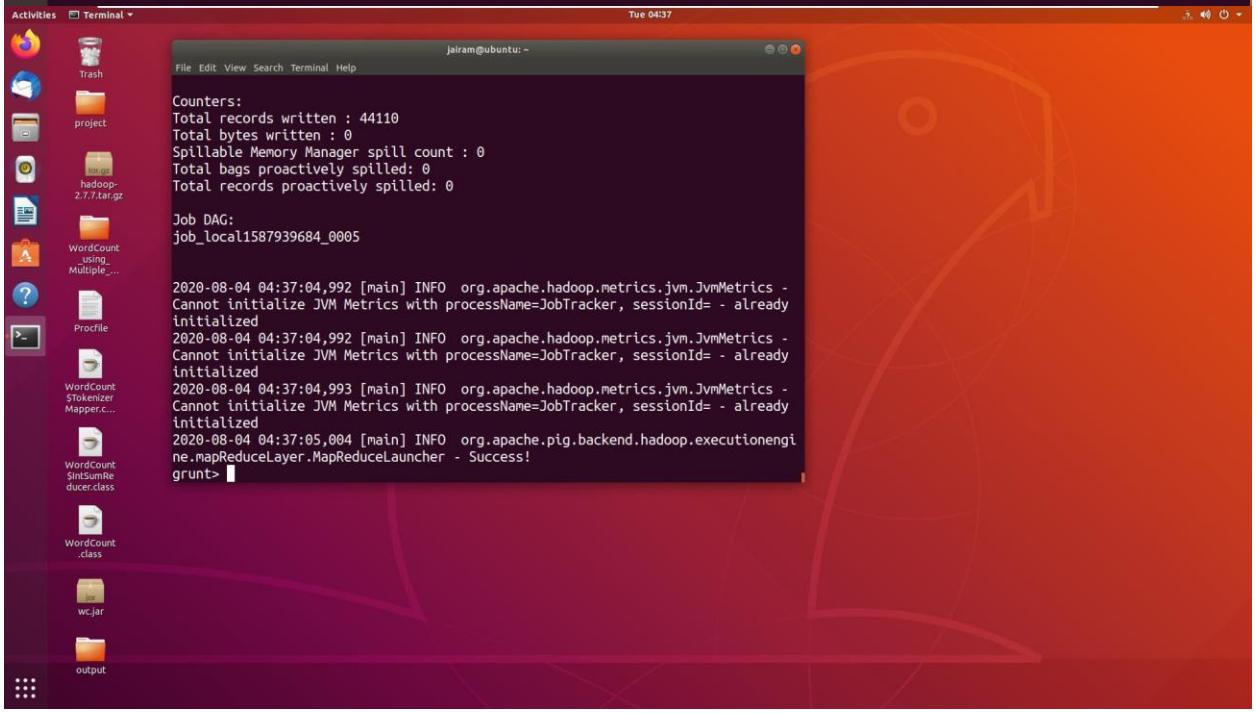
5. Command***: grunt >C = group B by words***

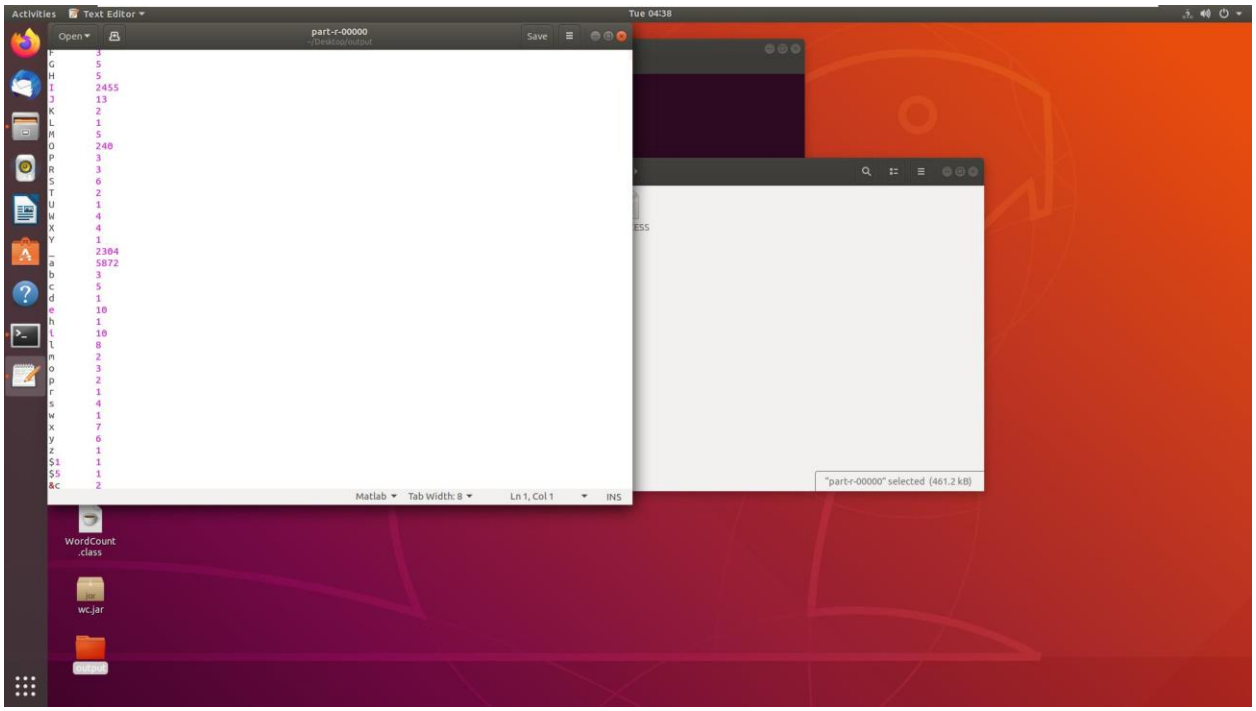
grunt > dump C;

6. Command***: grunt > forceach c generate group , COUNT(B);***

grunt > dump D;

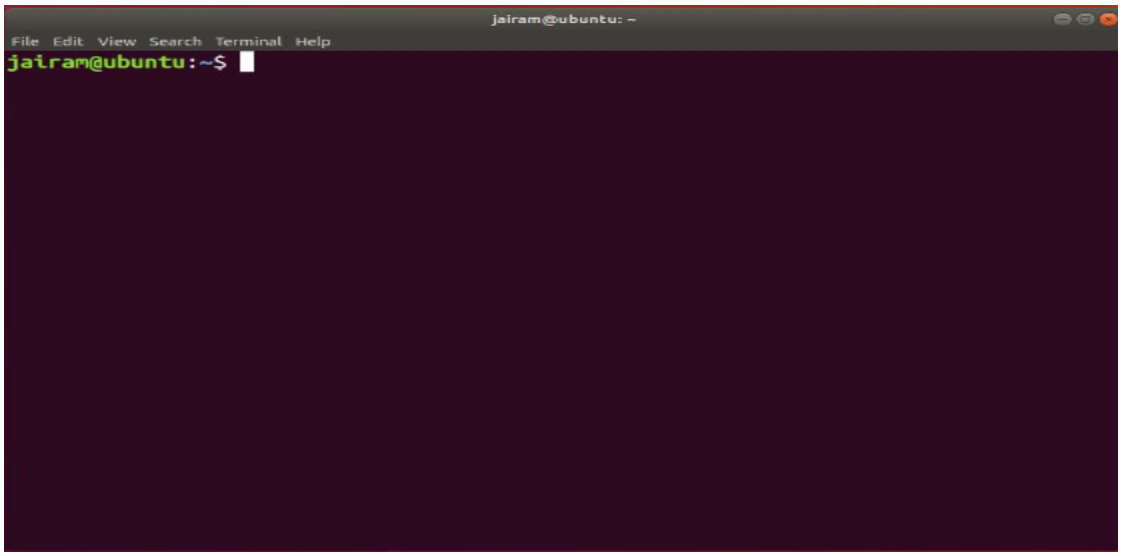




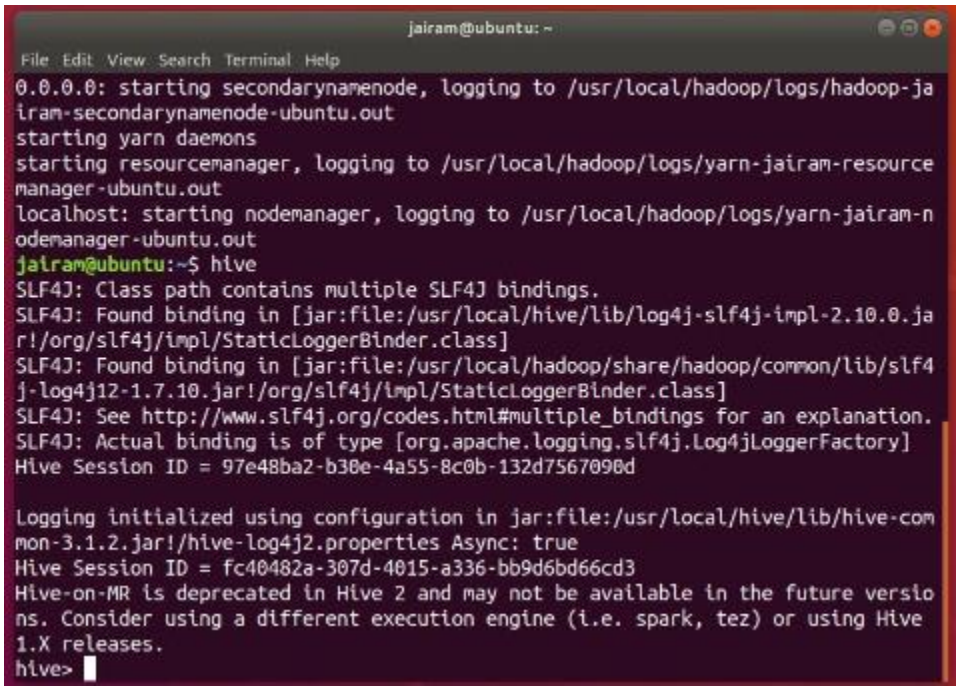

Activities
Text Editor
Tue 04:38
Open
part-r-00000
~/Desktop/output
Save
F 3
G 5
H 5
I 2455
J 13
K 2
L 1
M 5
O 240
P 3
R 3
S 6
T 2
U 1
W 4
X 4
Y 1
_ 2304
a 5872
b 3
c 5
d 1
e 10
h 1
i 10
l 8
m 2
o 3
p 2
r 1
s 4
w 1
x 7
y 6
z 1
$1 1
$$ 1
&c 2
Matlab
Tab Width: 8
Ln 1, Col 1
INS
"part-r-00000" selected (461.2 kB)
WordCount
.class
wc.jar
output

Data compression using Hive

1. Open terminal

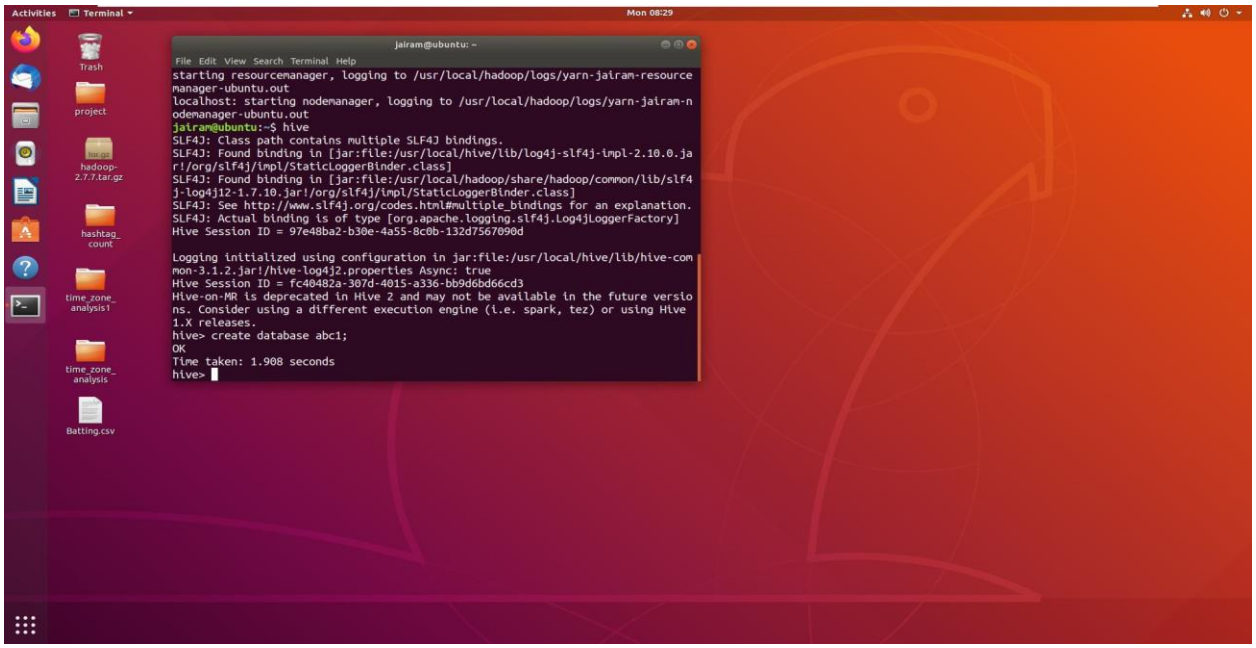


2. Command: ***hive*** for lauching hive

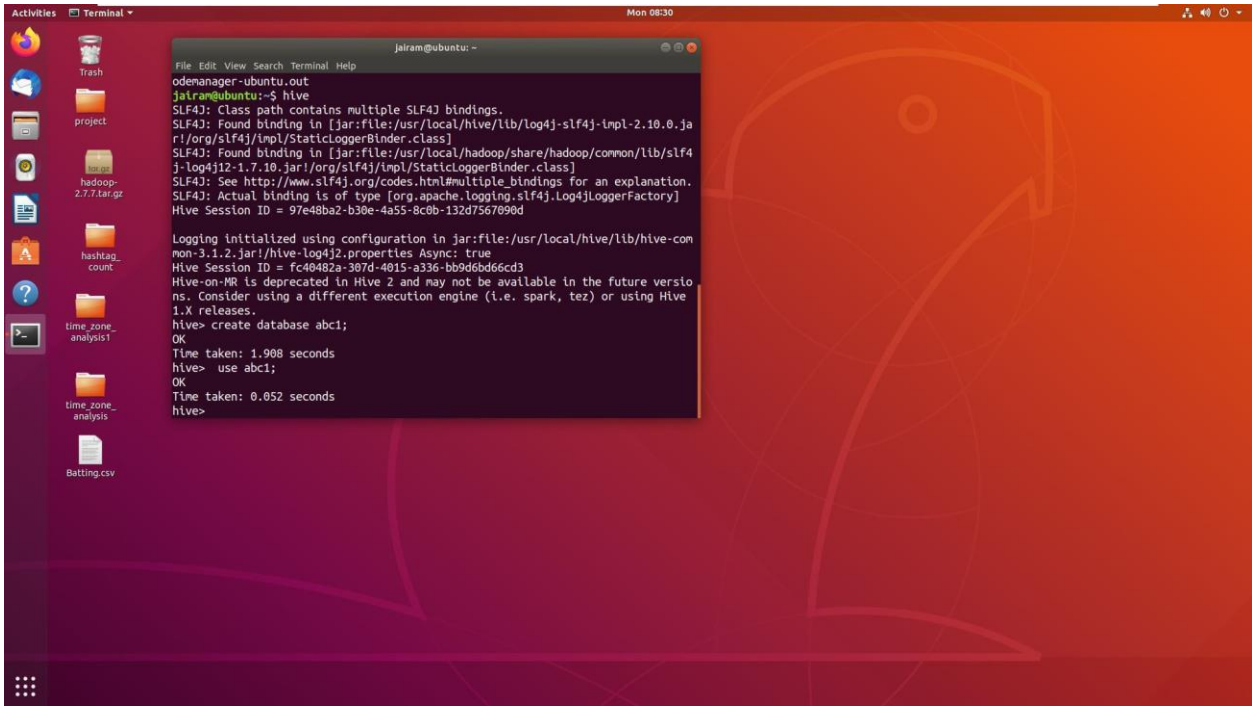


3. To create database...

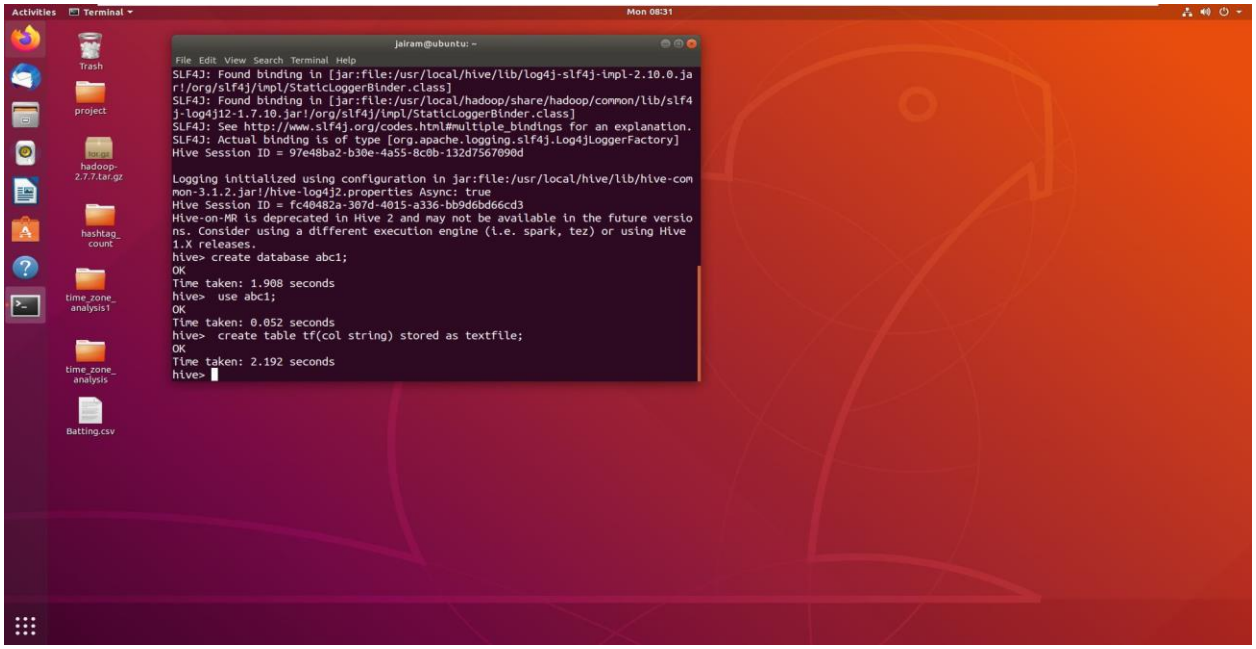
Command: ***create database abc;***



4. Command: use abc:

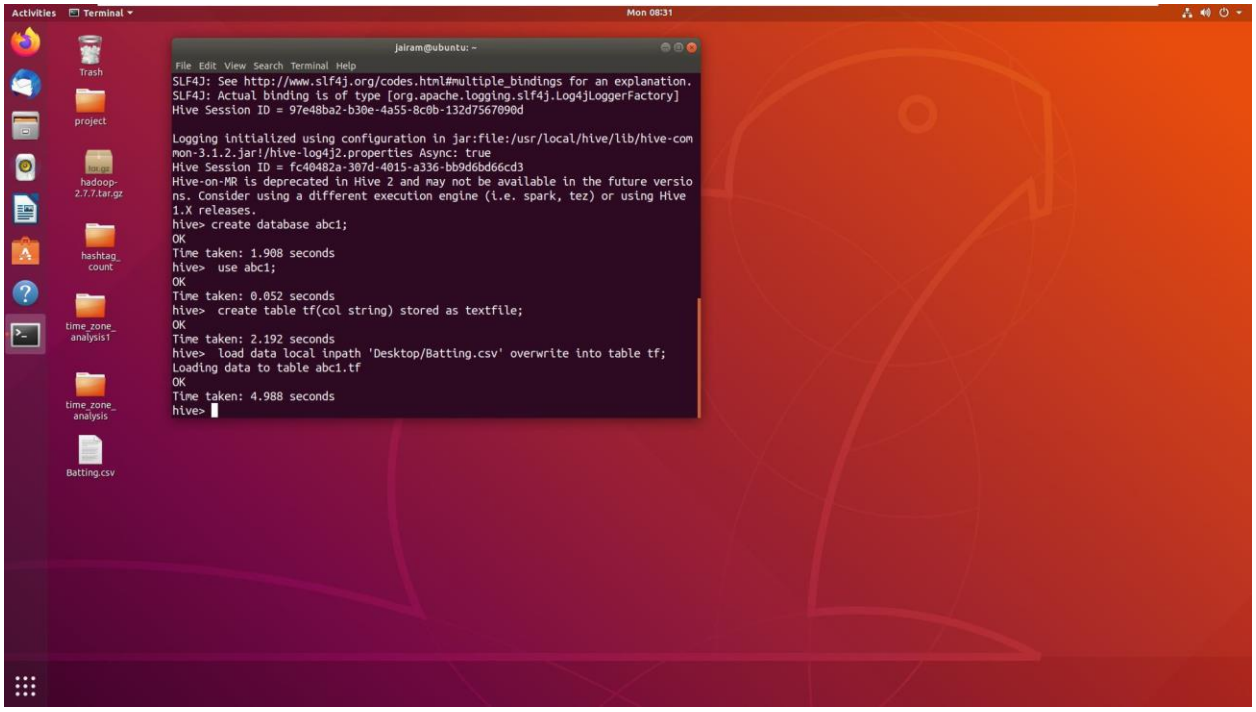


5. Command: ***create table tf(col string) stored as textfile;***



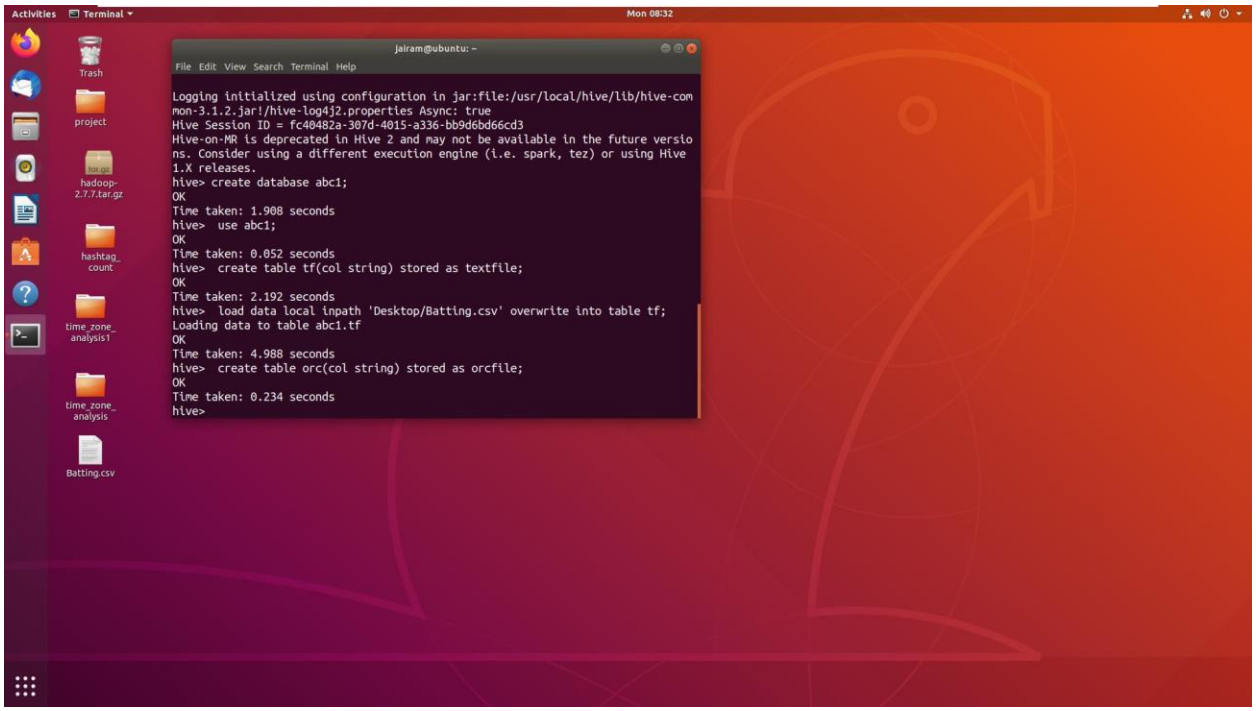
6. To create optimized row column file...

Command: ***load data local inpath 'Desktop/Batting.csv' overwrite into table tf;***

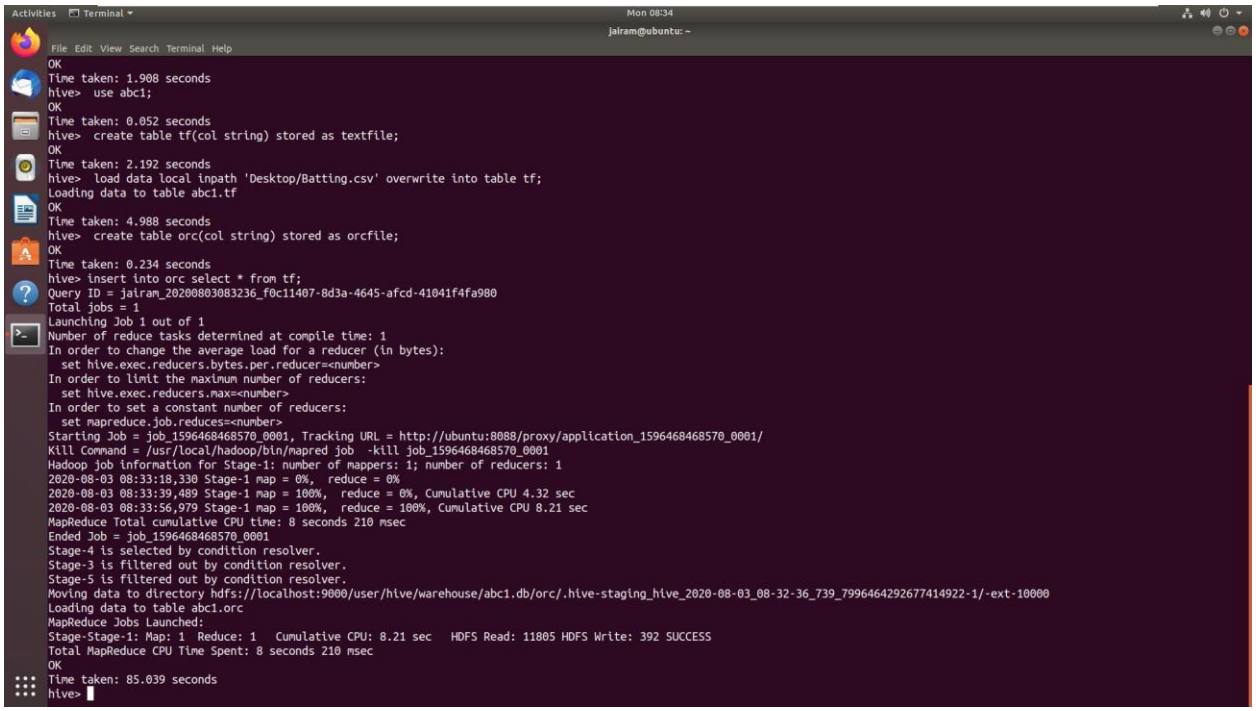


7. you can't load direct data in orc table so by using this command you can insert data in orc table

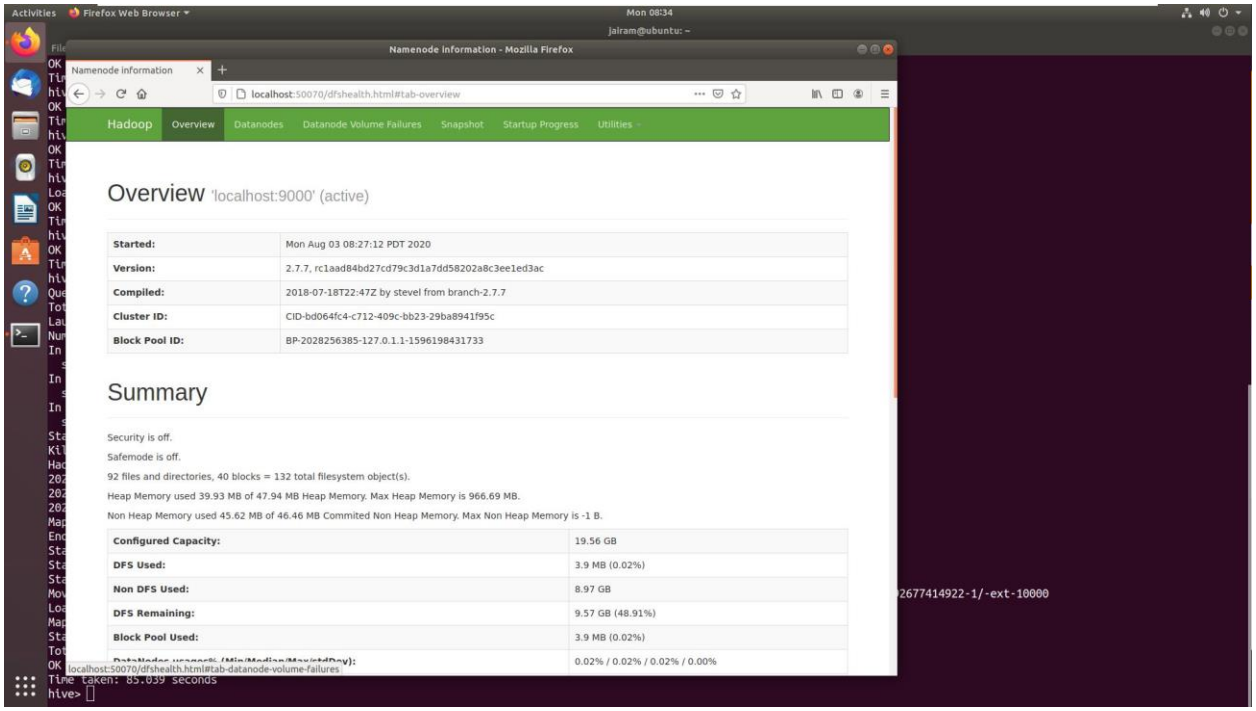
command: ***create table orc(col string) stored as orcfile;***



8. command: ***insert into orc select * from tf;***

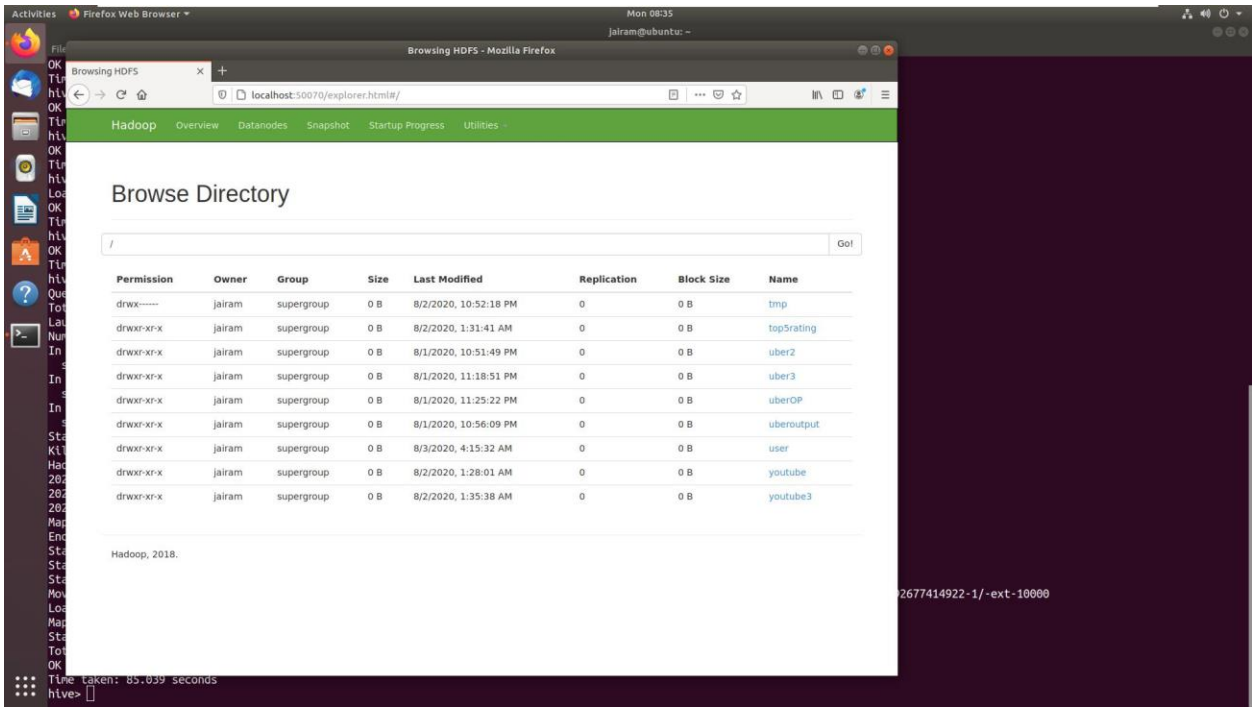


9. now open browser then type localhost:50070



10.then utilities

11.then browse the file system



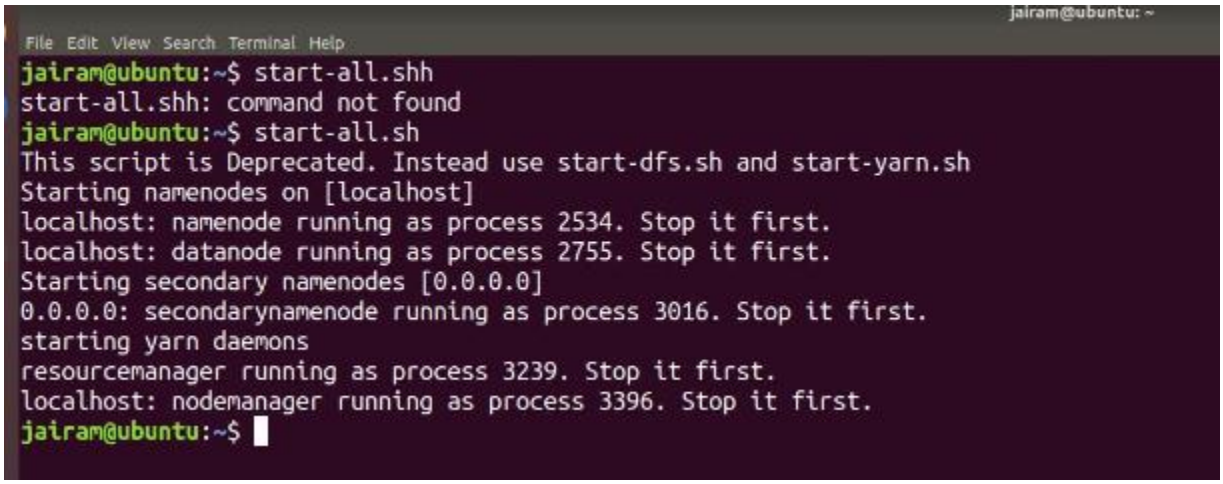
12.then

/user/hive/warehouse/abc.db/tf //// here check the size of file

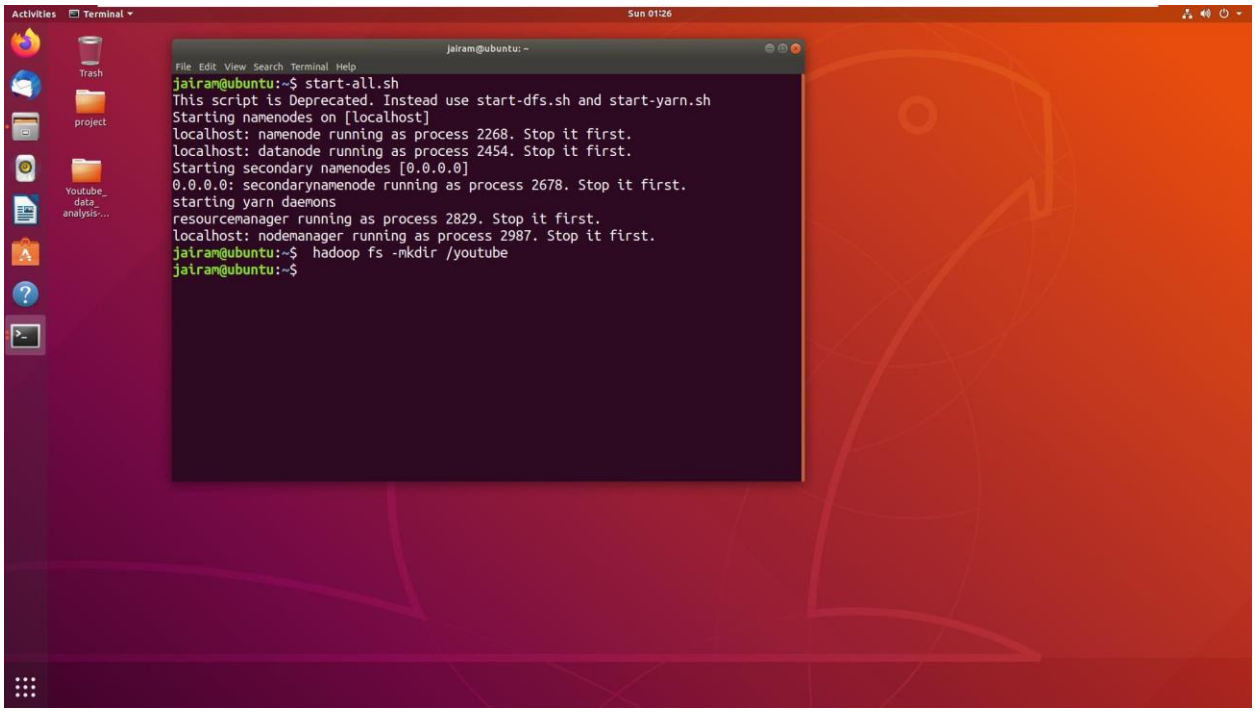
/user/hive/warehouse/abc.db/orc //// now check the compressed form of data the data is now converted into binary format and also its size also decreases.

Youtube data analysis

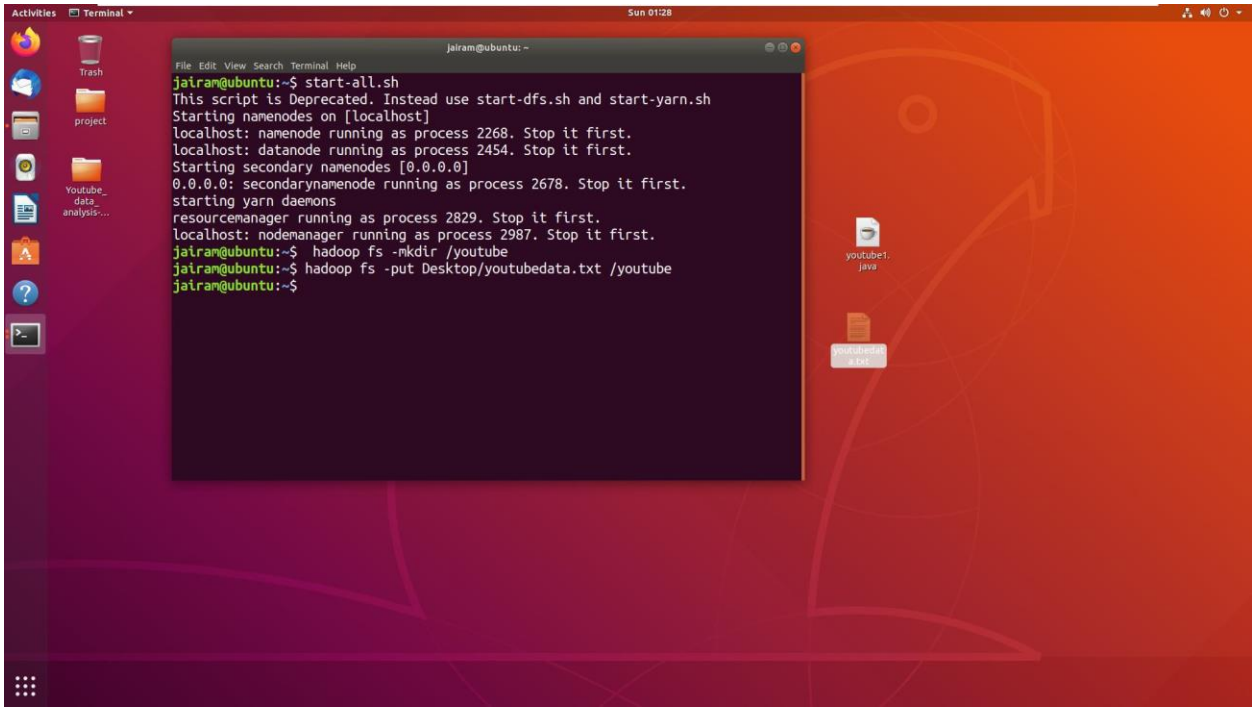
1. Open terminal and type Command: ***start-all.sh***



2. command: ***hadoop fs -mkdir /youtube***



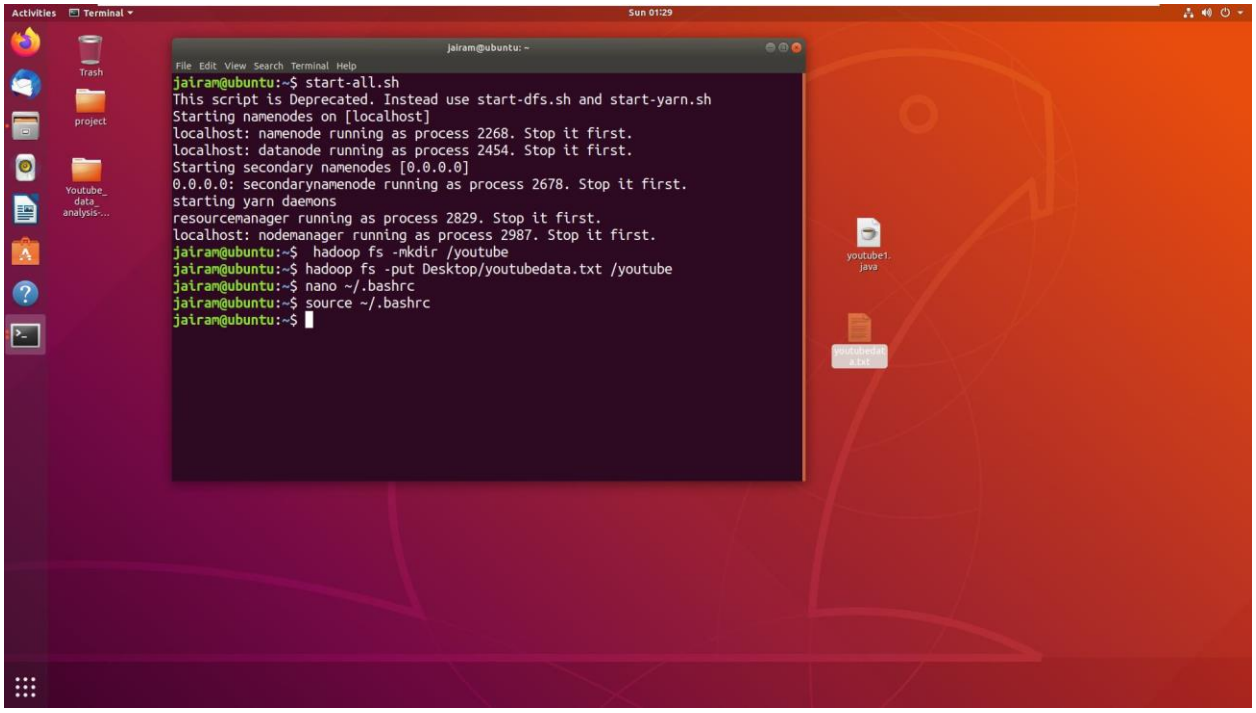
3. command: ***hadoop fs -put Desktop/youtubedata.txt /youtube***



4. Open bashrc file by ***nano ~/. bashrc*** and type

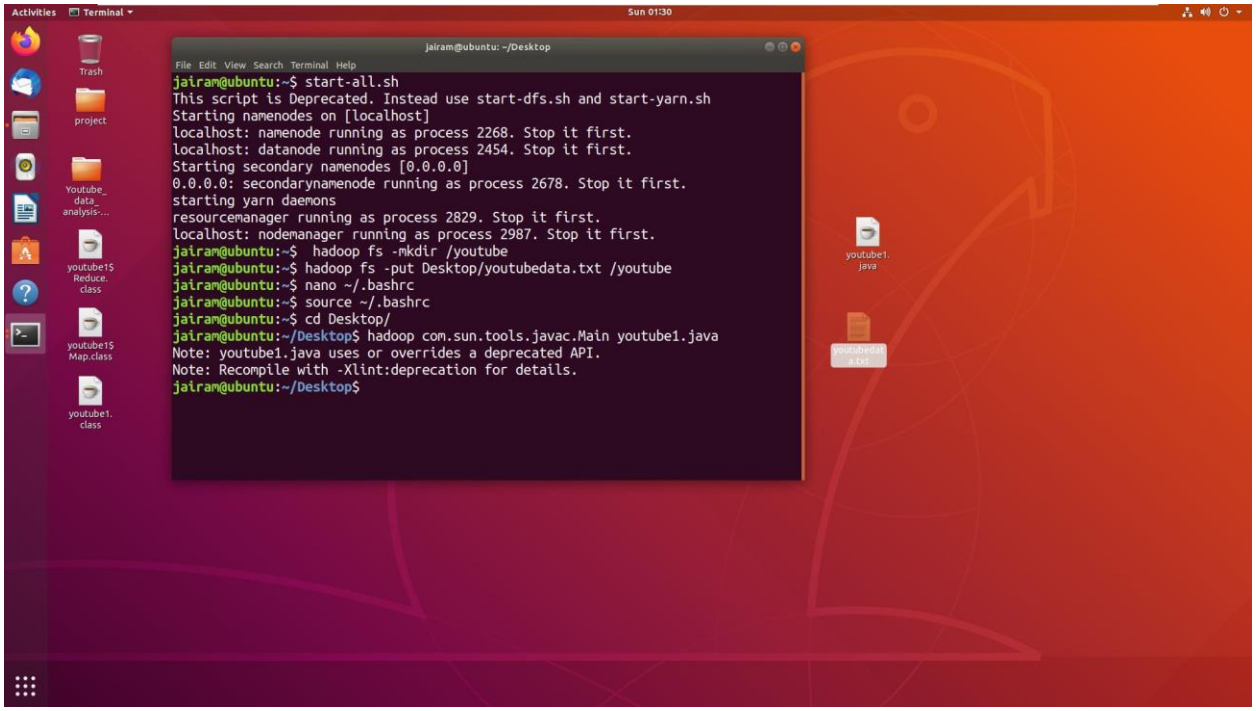
   export HADOOP_CLASSPATH=$JAVA_HOME/lib/tools.jar

5. Then commnd: ***source ~/.bashrc***



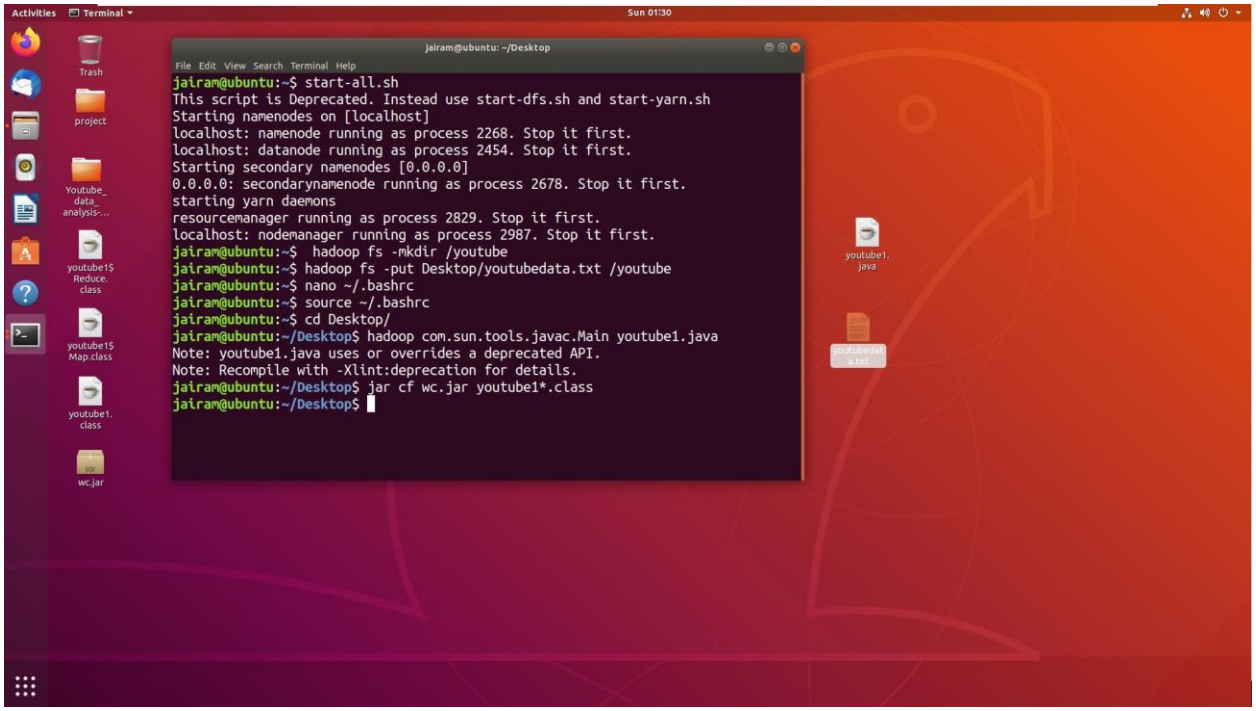
6. Command: ***cd Desktop***
7. Now compile youtube1.java file suppose that file is on Desktop

   Command: ***hadoop com.sun.tools.javac.Main youtube1.java***



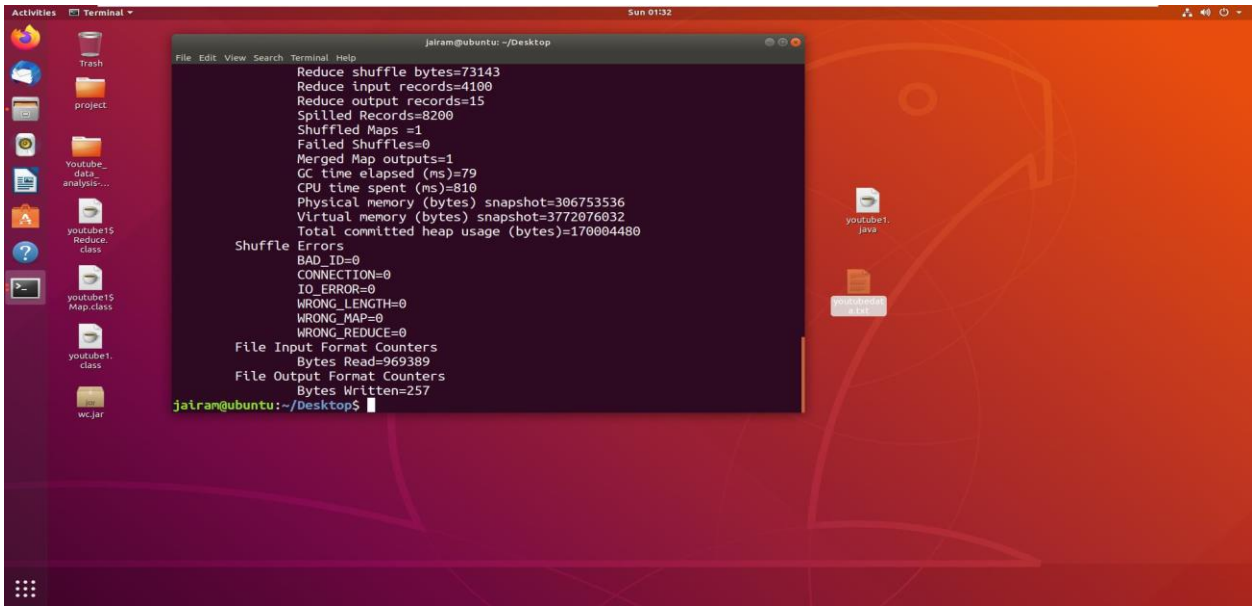
8. To combine all class files

Command; ***jar cf wc.jar youtube1*.class***



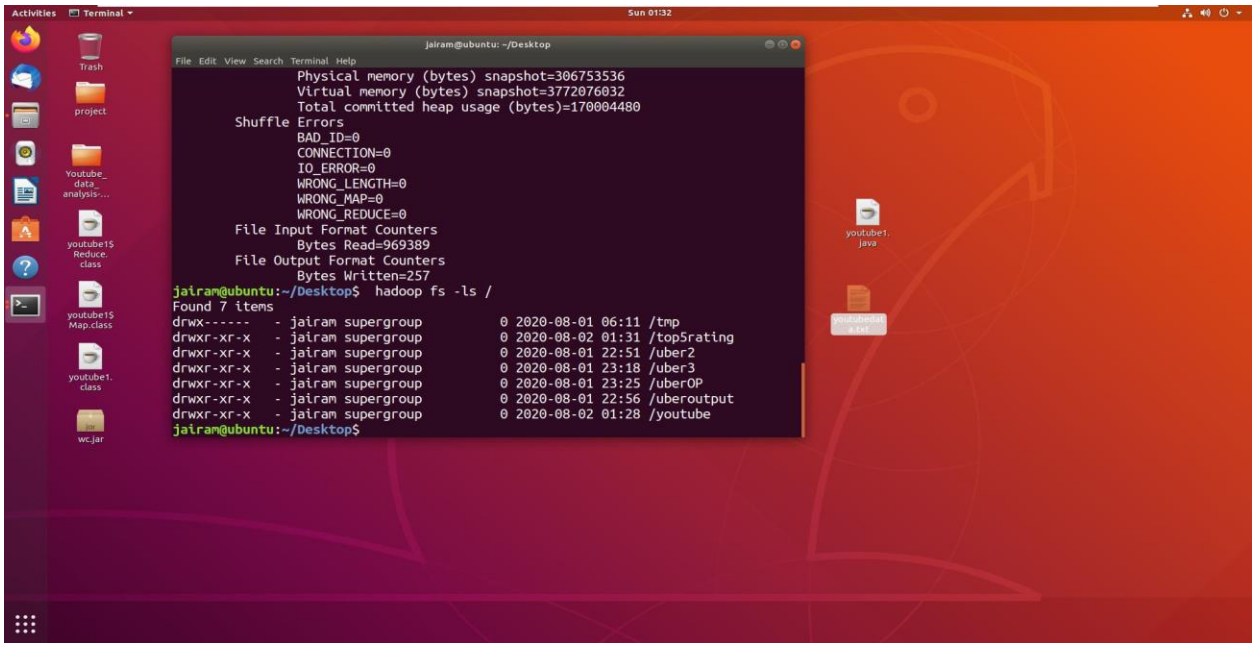
9. To execute

Command: ***hadoop jar wc.jar youtube1 /youtube/youtubedata.txt /top5rating***

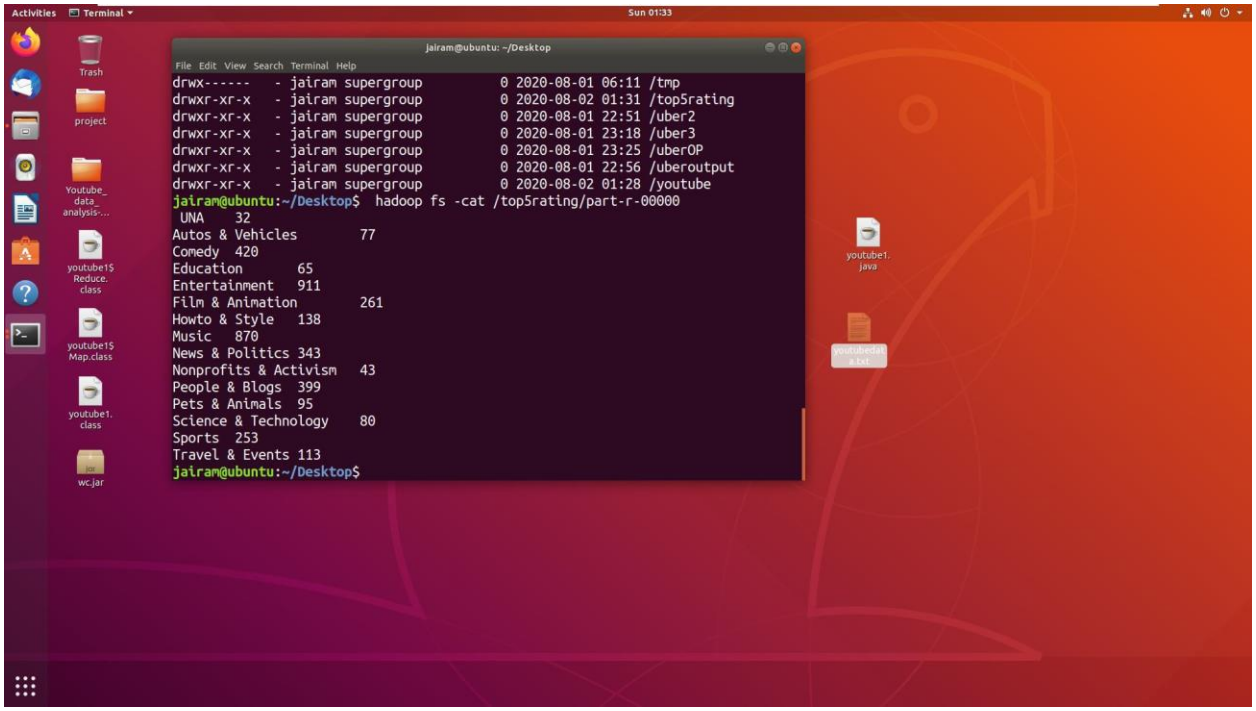


10. To check output folder is there or not

Command: ***hadoop fs -ls /***



11.command: hadoop fs -cat /top5rating/part-r-00000



Bibliography And Reference Taken.

Jairam J S(2020) - Beginner's Guide to Big data

Project Workbook Of Big data by cosmic skill.