NAME : PARIK SHARMA

ROLL NO : 454

COLLEGE : UIT SHIMLA

EMIAL : parik1950@gmail.com

Phone no. : 8894217922

PROJECT : FACEBOOK DATA ANALYSIS

Facebook Data analysis using the HDFS and using HIVE

DESCRIPTION OF DATASET

- There are 15 columns and 65536 rows.
- Columns are : unique_id, age, dob_day, dob_year, dob_month, gender, tenure, friend_count,friendship_initiated, likes, likes_recd, mobile_likes, mobile_likes_received, www_likes, www_likes_received.

INPUT FILES

pseudo_facebook
.csv

PROBLEM STATEMENT

- Find out the total number of user in this dataset ?
- Find out the number of facebook users above the age of 25 ?
- Do male facebook users tends to have more friends,or female users ?
- How many likes do young people receive to facebook opposed to older members ?
- Find out the count of facebook users for each birthday month ?
- Do young people use mobile phone or computers for facebook browsing?

ENVIRONMENT SETUP

- **Software Specification**
  - Oracle VirtualBox - version 6.1
  - Hadoop Version - Hadoop 2.9.1
  - Hive version - Hive 1.2.2
  - WinSCP - version 5.9.4

PROJECT MODULES

1. Placing the given dataset in HDFS
   a. Create directory in HDFS
   b. Placing the dataset in HDFS directory

2. Implementation in HIVE
   a. Creating HIVE database
   b. Creating and loading the HIVE tables with the given datasets

3. Problem Scenario 1 - Find out the total number of user in this dataset ?

4. Problem Scenario 2 - Find out the number of facebook users above the age of 25?

5. Problem Scenario 3 - Do male facebook users tends to have more friends,or female users?

6. Problem Scenario 4 - How many likes do young people receive to facebook opposed to older members ?
7. Problem Scenario 5 - Find out the count of facebook users for each birthday month?
8. Problem Scenario 6 - Do young people use mobile phone or computers for facebook browsing?

Placing the given dataset in HDFS

Create Directory in HDFS

Step 1 a: First of all we have started all the daemons in hadoop and then write this command to make a directory in HDFS.

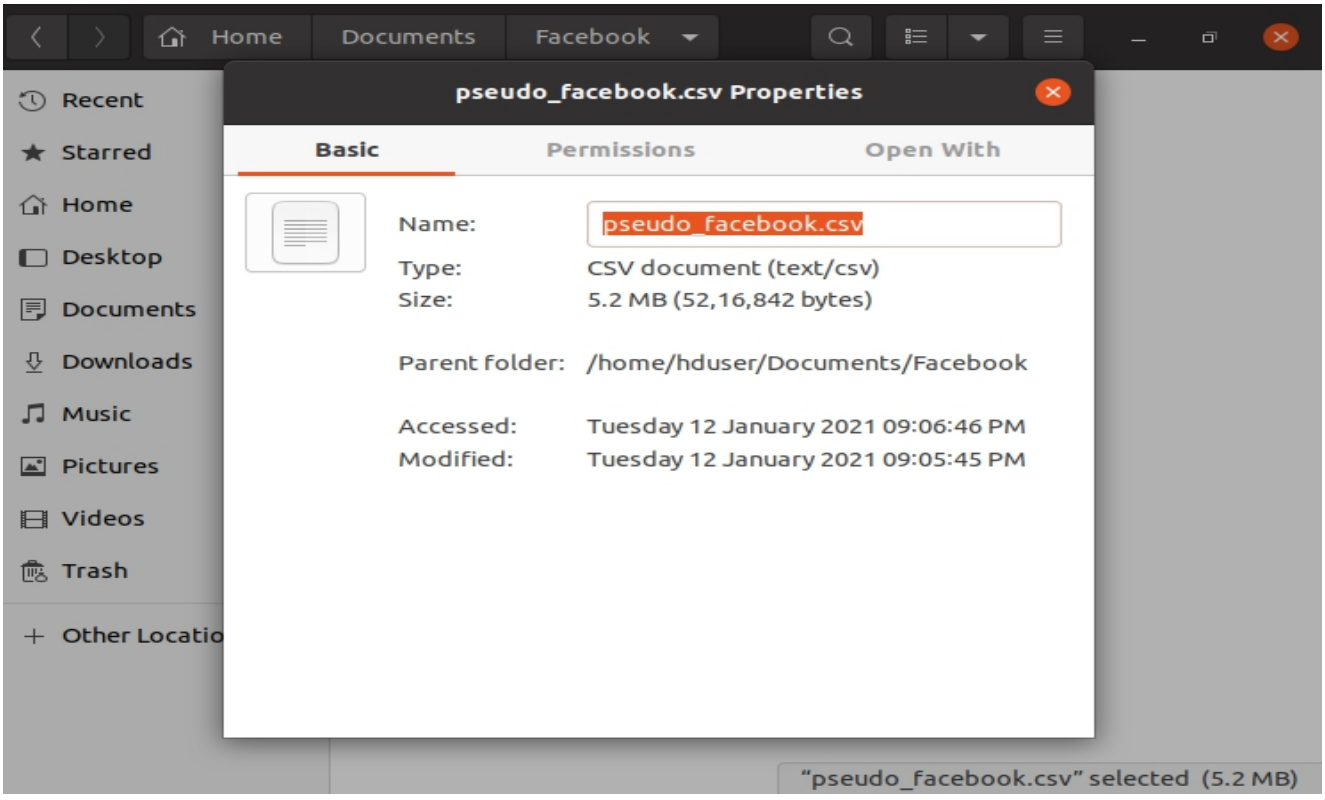
```
$ hdfs dfs -mkdir /facebookdata
```

Placing the dataset in HDFS directory

Step 1 b: Copying the dataset from local to HDFS directory in a separate directory. The code as follows :

```
$hdfs dfs -put Documents/Facebook/pseudo_facebook.csv /facebookdata
```

This image showing the location of dataset:



Facebook Data_analysis_by PARIK SHARMA

These are the commands to create a directory in HDFS and copying the data from local to HDFS :

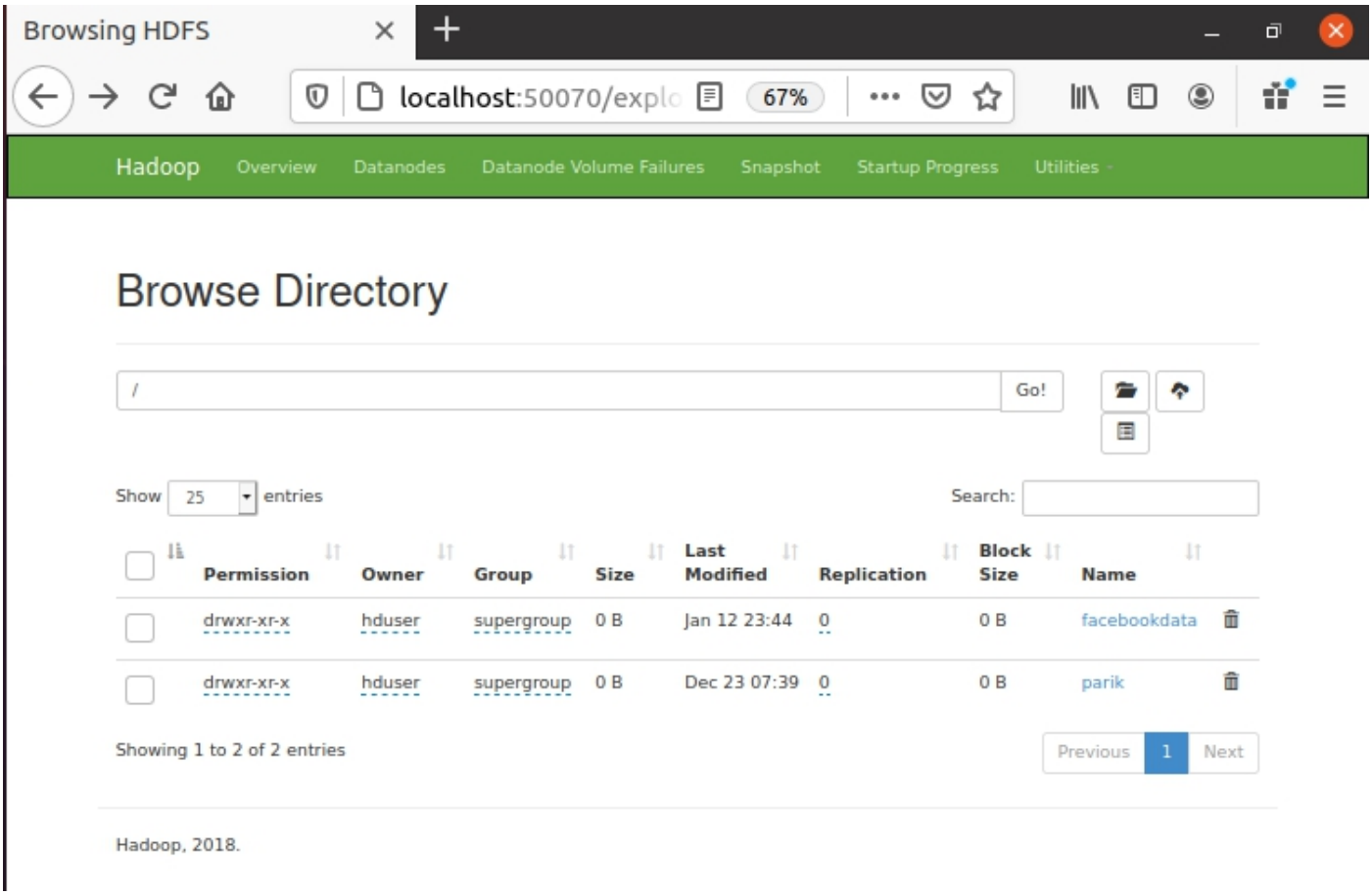
```
hduser@parik-VirtualBox:~$ hdfs dfs -mkdir /facebookdata
hduser@parik-VirtualBox:~$ hdfs dfs -put Documents/Facebook/pseudo_facebook.csv /
facebookdata
hduser@parik-VirtualBox:~$
```

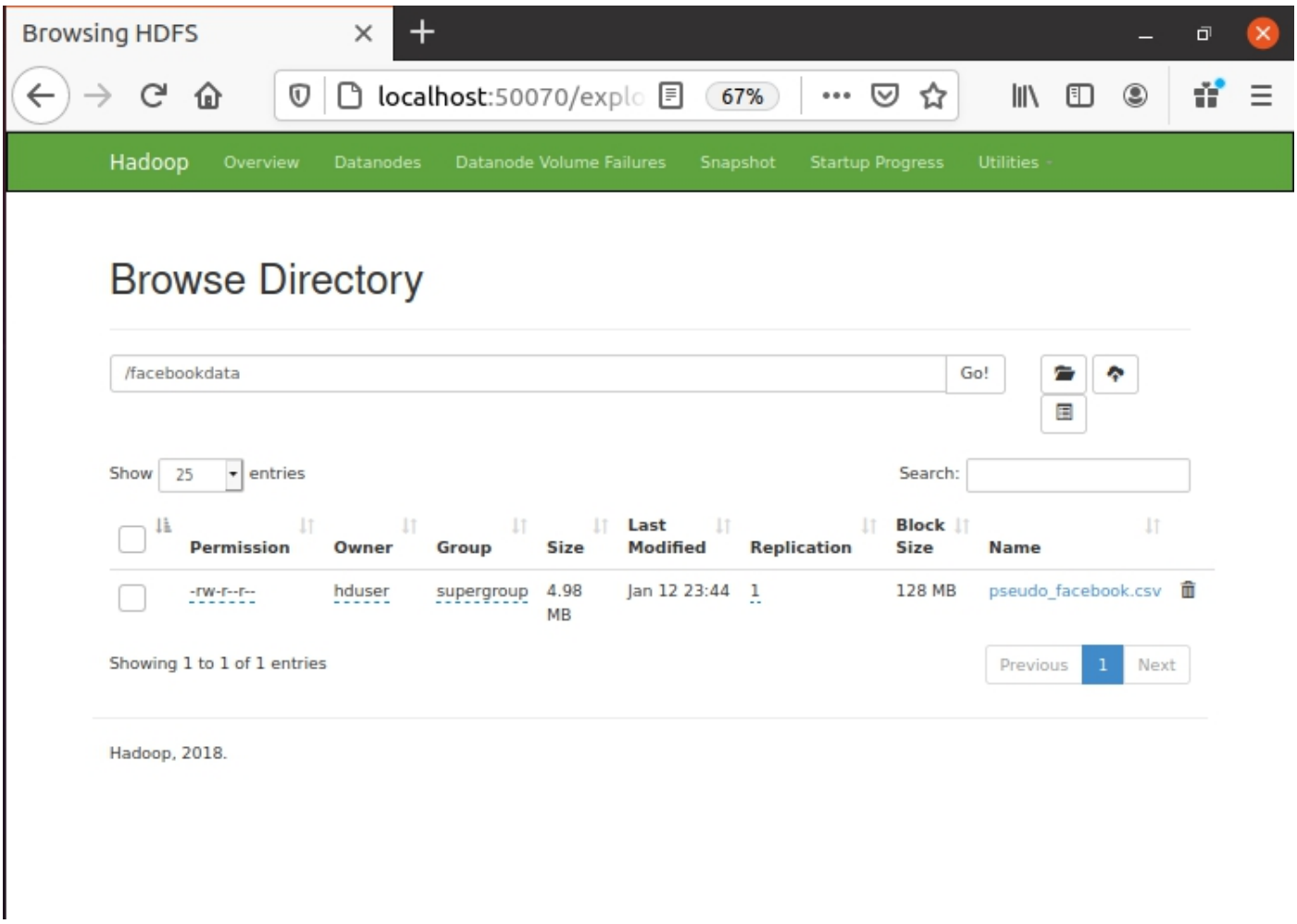
This command shows the directories/files in the HDFS :

$hdfs dfs -ls /

```
hduser@parik-VirtualBox:~$ hdfs dfs -ls /
Found 2 items
drwxr-xr-x   - hduser supergroup          0 2021-01-12 21:06 /facebookdata
drwxr-xr-x   - hduser supergroup          0 2020-12-23 07:39 /parik
hduser@parik-VirtualBox:~$ hdfs dfs -ls /facebookdata
Found 1 items
-rw-r--r--   1 hduser supergroup    5216842 2021-01-12 21:06 /facebookdata/pseudo
_facebook.csv
hduser@parik-VirtualBox:~$
```

Below two images show the HDFS directory in the GUI interface :-





Implementation in HIVE

Creating HIVE DB

Step 2 a: Create a database in name “Facebook ” in HIVE. The code as follows :

```
Hive > create database Facebook;
```

```
hduser@parik-VirtualBox:~$ hive

Logging initialized using configuration in jar:file:/home/hduser/apache-hive-1.2.
2-bin/lib/hive-common-1.2.2.jar!/hive-log4j.properties
hive> show databases;
OK
default
Time taken: 2.386 seconds, Fetched: 1 row(s)
hive> create database facebook;
OK
Time taken: 0.682 seconds
hive> show databases;
OK
default
facebook
Time taken: 0.049 seconds, Fetched: 2 row(s)
hive> use facebook;
OK
Time taken: 0.03 seconds
hive> show tables;
OK
Time taken: 0.193 seconds
```

Creating and Loading the HIVE tables with the given datasets

Step 2 b: Create a table in name “fb” in HIVE. The code as follows:

Hive> create table fb(id int, age int, day int, year int, month int, gender string, tenure int, friends int, friend_init int, likes int, likes_recd int, mlikes int, mlikes_recd int, wlikes_int int, wlikes_recd int) row format delimited fileds terminated by ‘,’ stored as textfile location ‘/facebookdata/’;

```
hive> create table fb(id int, age int,day int, year int, month int, gender string
, tenure int, friends int, friend_init int, likes int, likes_recd int, mlikes int
, mlikes_recd int, wlikes int, wlikes_recd int) row format delimited fields termi
nated by ',' stored as textfile location '/facebookdata/';
OK
Time taken: 1.106 seconds
hive> select * from fb limit 5;
OK
2094382 14      19      1999    11      male    266     0       0       0       0
0       0       0       0
1192601 14      2       1999    11      female  6       0       0       0       0
0       0       0       0
2083884 14      16      1999    11      male    13      0       0       0       0
0       0       0       0
1203168 14      25      1999    12      female  93      0       0       0       0
0       0       0       0
1733186 14      4       1999    12      male    82      0       0       0       0
0       0       0       0
Time taken: 2.079 seconds, Fetched: 5 row(s)
hive>
```

Problem Scenario 1 :-

In problem 1 we have to find the total number of user in this dataset. Since each row represent a user so we write the following command to get the result:

Hive > select count(*) from fb;

```
hive> select count(*) from fb;
Query ID = hduser_20210112232752_665983e1-a418-4cdd-a44a-cc7fc036153d
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>
Job running in-process (local Hadoop)
2021-01-12 23:27:57,414 Stage-1 map = 0%,  reduce = 0%
2021-01-12 23:28:00,529 Stage-1 map = 100%,  reduce = 0%
2021-01-12 23:28:01,541 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1408633107_0001
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 10441876 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
99003
Time taken: 9.601 seconds, Fetched: 1 row(s)
hive>
```

Answer :- Total number of user are *99003*

Problem Scenario 2 :-

In second problem we have to find out the facebook users which are above 25 years, so it is similar to problem 1 but here we have to add a condition at the last using WHERE keyword. So the code to this problem is :-

Hive > select count(*) from fb where age >= 35;

```
hive> select count(*) from fb where age>=25;
Query ID = hduser_20210112232836_24ce82f4-8a7d-4acc-b040-b30f178d2e75
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>
Job running in-process (local Hadoop)
2021-01-12 23:28:40,472 Stage-1 map = 0%,  reduce = 0%
2021-01-12 23:28:41,482 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local286643655_0002
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 20875560 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
60317
Time taken: 4.751 seconds, Fetched: 1 row(s)
hive>
```

Answer :- Total number of users which are above 25 years are *60317*

Problem Scenario 3:

In this problem we have to find which gender I.e male or female have more numbers of friends . So we can solve this problem using two ways:-

I) Hive > select avg(friends) from fb where gender='male' ;
Hive > select avg(friends) from fb where gender='female';

II.)Hive > select gender,avg(friends) from fb group by gender;

```
hive> select avg(friends) from fb where gender='male';
Query ID = hduser_20210112232935_2f51edf6-9f73-415c-b044-145b0fda29d2
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>
Job running in-process (local Hadoop)
2021-01-12 23:29:37,628 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local919783616_0003
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 31309244 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
165.03545941885477
Time taken: 2.202 seconds, Fetched: 1 row(s)
hive>
```

```
hive> select avg(friends) from fb where gender='female';
Query ID = hduser_20210112232958_749457ac-7db7-466b-8dc7-62300701c59d
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>
Job running in-process (local Hadoop)
2021-01-12 23:30:00,812 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1102783658_0004
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 41742928 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
241.96994087544095
Time taken: 2.328 seconds, Fetched: 1 row(s)
hive>
```

Answer :- By looking at output we can clearly see that female users have more friends than the male users. The average friends in male users is *165.035* and the average friends in female users Is *241.9699*

Problem Scenario 4 :-

In this problem we have to find which young user or old user have more number of likes we can do it by appling a condition using WHERE keyword.Code for Problem is as follows :

Hive > select avg(likes_recd) from fb where age >= 14 AND age <=25;
Hive > select avg(likes_recd) from fb where age >= 35;

```
hive> select avg(likes_recd) from fb where age>=14 AND age <= 25;
Query ID = hduser_20210112233112_8966020e-87c8-4334-a8fb-9d5729d63b0c
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>
Job running in-process (local Hadoop)
2021-01-12 23:31:14,024 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local103230178_0005
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 52176612 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
200.7832373395789
Time taken: 1.895 seconds, Fetched: 1 row(s)
hive>
```

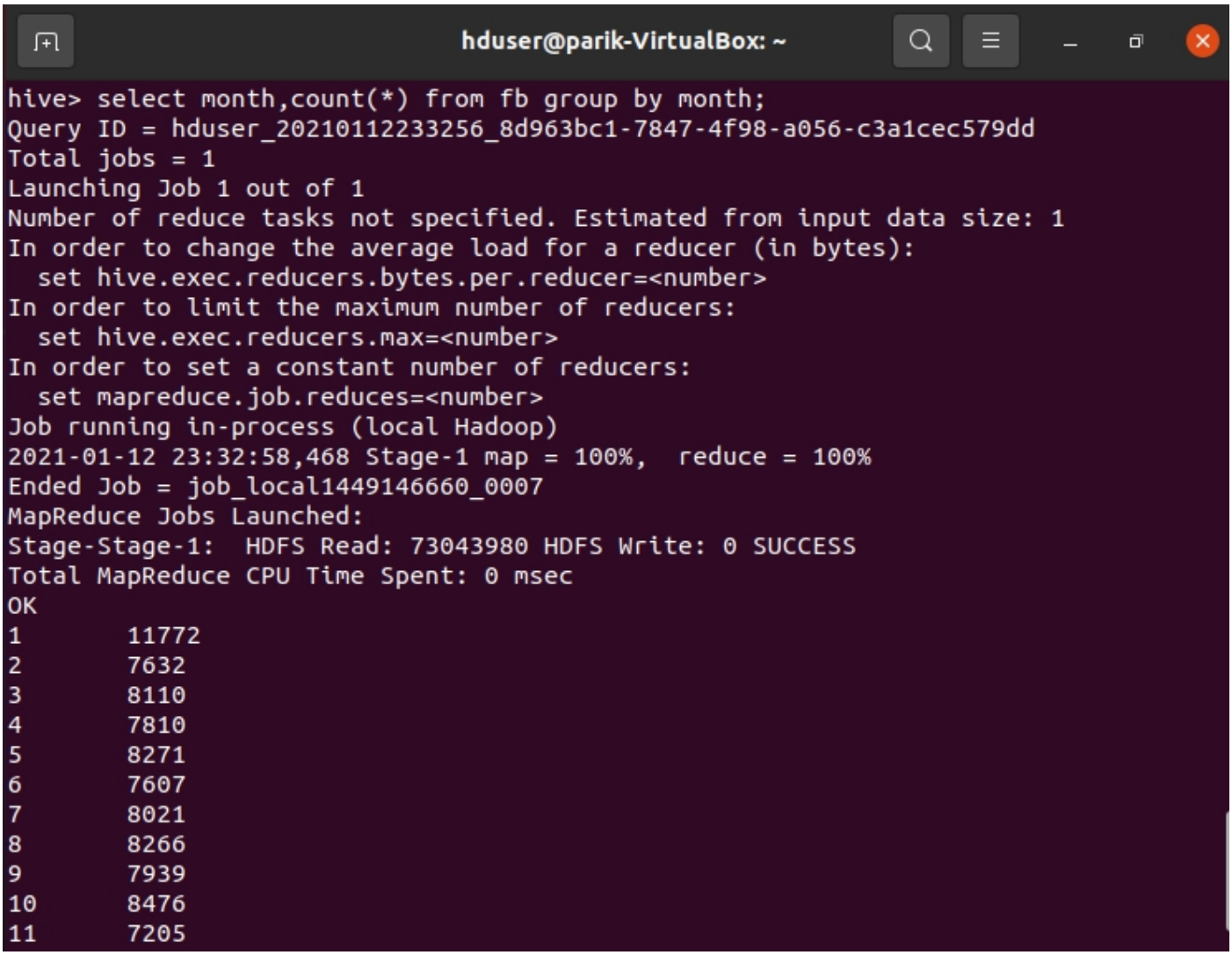
```
hive> select avg(likes_recd) from fb where age >= 35;
Query ID = hduser_20210112233140_7ae88c5d-bad4-42e8-9f5b-1ffd9c270afe
Total jobs = 1
Launching Job 1 out of 1
Number of reduce tasks determined at compile time: 1
In order to change the average load for a reducer (in bytes):
  set hive.exec.reducers.bytes.per.reducer=<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>
Job running in-process (local Hadoop)
2021-01-12 23:31:41,831 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1094614664_0006
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 62610296 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
103.89021217994491
Time taken: 1.61 seconds, Fetched: 1 row(s)
hive>
```

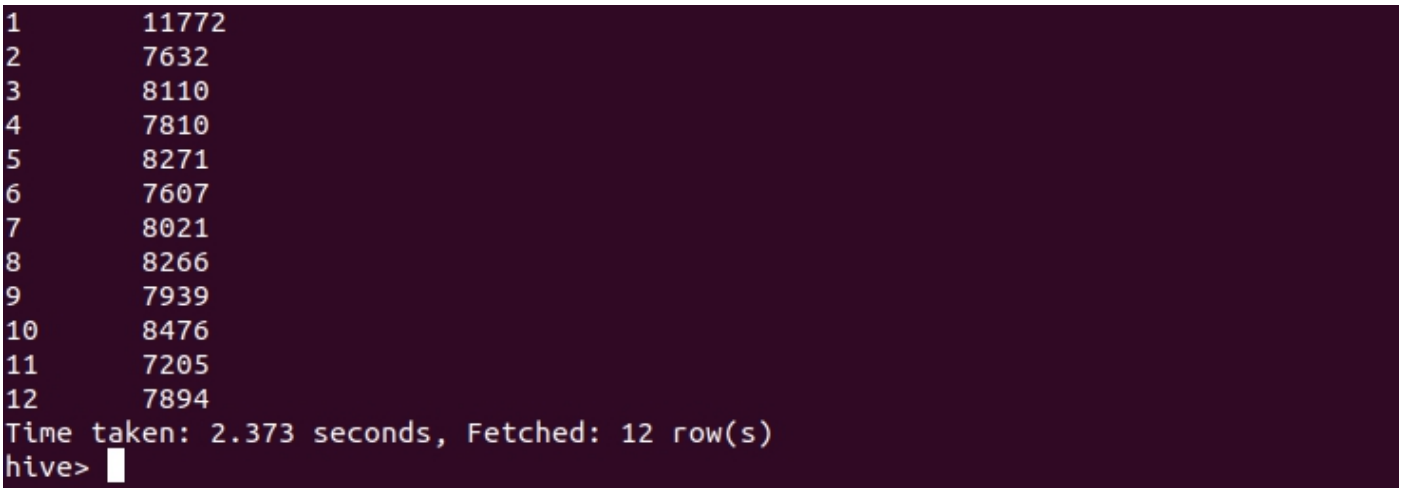
Answer :- From output we can see that younger user get an average of *200.783* likes and on the other hand old user get *103.89* likes. So younger user get more number of likes.

Problem Scenario 5 :-

In this problem we have to count the facebook users in each birthday month. This problem can be solved by using GROUP BY keyword . The code for this problem is :-

Hive > select month, count(*) from fb group by month;





Answer :- Result can be seen from the picture and from this picture we can say that maximum users have their birthday in *january*.

Problem Scenario 6 :-

In this problem we have to find that do young people use mobile or laptop for facebook browsing. We can analys it by using mlikes and wlikes columns in the dataset . We just calculate the average of likes in both the columns based on age and then we can find if the young or old user use mobile or laptop more.Code for this problem is as follows : -

Hive>select avg(mlikes), avg(wlikes) from fb where age >= 13 AND age <= 25;
Hive>select avg(mlikes), avg(wlikes) from fb where age >= 25;

```
hive> select avg(mlikes),avg(wlikes) from fb where age >=13 AND age <=25;
Query ID = hduser_20210112233500_98bbc3e9-9ec9-46b0-92dc-cfbdc11b08c4
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>
Job running in-process (local Hadoop)
2021-01-12 23:35:02,190 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local257942695_0008
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 83477664 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
123.98981737425284      55.50010631511801
Time taken: 1.706 seconds, Fetched: 1 row(s)
hive>
```

```
hive> select avg(mlikes),avg(wlikes) from fb where age >= 35;
Query ID = hduser_20210112233522_5155c87f-84ce-459e-b674-de61982fb21c
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>
Job running in-process (local Hadoop)
2021-01-12 23:35:24,031 Stage-1 map = 100%,  reduce = 100%
Ended Job = job_local1066779229_0009
MapReduce Jobs Launched:
Stage-Stage-1:  HDFS Read: 93911348 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
94.55878302560441       56.50313679485872
Time taken: 1.848 seconds, Fetched: 1 row(s)
hive>
```

Answer :- From output we can clearly see that young users have average of *123.93* likes from mobile and *55.5* likes from web , on the other hand old users have average of *94.55* likes from mobile and *56.5* likes from web. So from these result we can say that old users use laptop more than the young user. And young user use more mobiles for facebook browsing .

THANK YOU