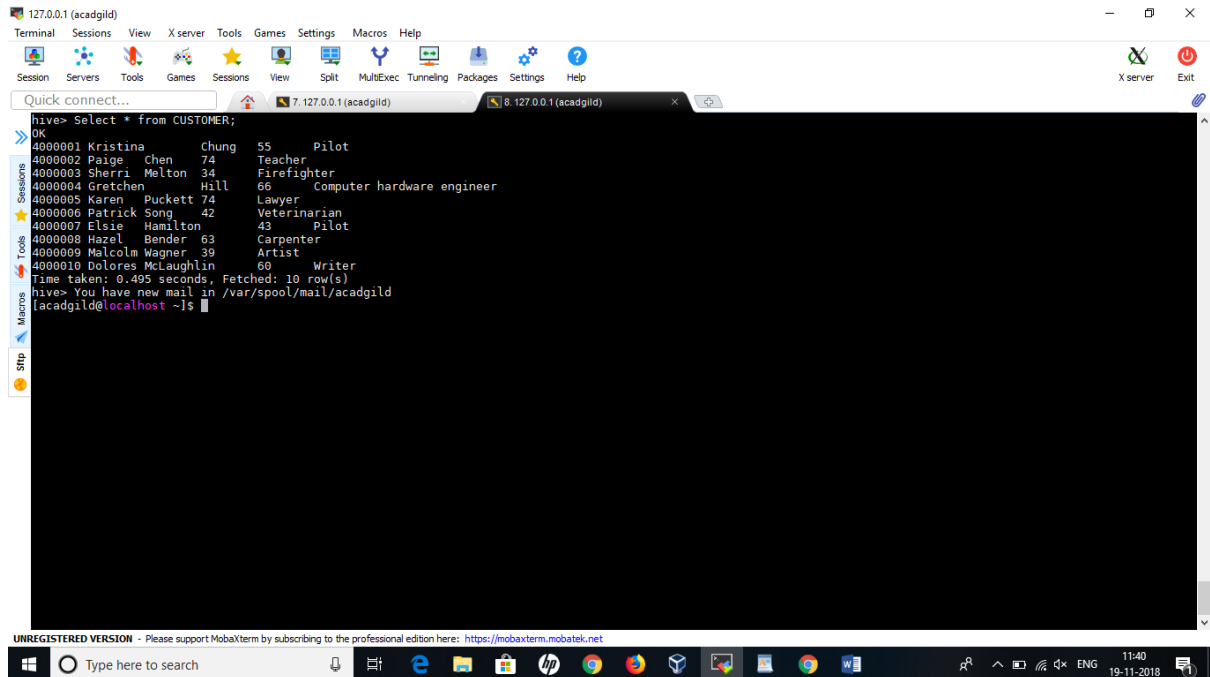


Case Study - Customer Transactions.

Let us take up the CUSTOMER and TRANSACTIONS table we have created in the

Let's Do Together section. Let us solve the following use cases using these tables:-

Solution: CUSTOMER TABLE: *Select * from CUSTOMER;*



The screenshot shows a MobaXterm terminal window with a Hive SQL query executed. The query is `hive> Select * from CUSTOMER;`. The output displays 10 rows of customer data, including ID, Name, Last Name, Age, and Profession. The terminal window has a menu bar (Terminal, Sessions, View, X server, Tools, Games, Settings, Macros, Help) and a toolbar. The Windows taskbar at the bottom shows the time as 11:40 on 19-11-2018.

```
hive> Select * from CUSTOMER;
OK
4000001 Kristina Chung 55 Pilot
4000002 Paige Chen 74 Teacher
4000003 Sherri Melton 34 Firefighter
4000004 Gretchen Hill 66 Computer hardware engineer
4000005 Karen Puckett 74 Lawyer
4000006 Patrick Song 42 Veterinarian
4000007 Elsie Hamilton 43 Pilot
4000008 Hazel Bender 63 Carpenter
4000009 Malcolm Wagner 39 Artist
4000010 Dolores McLaughlin 60 Writer
Time taken: 0.495 seconds, Fetched: 10 row(s)
hive> You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$
```

Transactions table(TXNRECORDS): *Select * from TXNRECORDS;*

Time taken: 0.124 seconds
hive> select * from TXNRECORDS;

| | | | | | | | |
|------------|---------|--------|------------------------|-----------------------------------|----------------|----------------|--------|
| 06-26-2011 | 4000001 | 40.33 | Exercise & Fitness | Cardio Machine Accessories | Clarksville | Tennessee | credit |
| 05-26-2011 | 4000002 | 198.44 | Exercise & Fitness | Weightlifting Gloves | Long Beach | California | credit |
| 06-01-2011 | 4000002 | 5.58 | Exercise & Fitness | Weightlifting Machine Accessories | Anaheim | California | credit |
| 06-05-2011 | 4000003 | 198.19 | Gymnastics | Gymnastics Rings | Milwaukee | Wisconsin | credit |
| 12-17-2011 | 4000002 | 98.81 | Team Sports | Field Hockey | Nashville | Tennessee | credit |
| 02-14-2011 | 4000004 | 193.63 | Outdoor Recreation | Camping & Backpacking & Hiking | Chicago | Illinois | credit |
| 10-28-2011 | 4000005 | 27.89 | Puzzles Jigsaw Puzzles | Charleston | South Carolina | credit | |
| 07-14-2011 | 4000006 | 96.01 | Outdoor Play Equipment | Sandboxes | Columbus | Ohio | credit |
| 01-17-2011 | 4000006 | 10.44 | Winter Sports | Snowmobiling | Des Moines | Iowa | credit |
| 05-17-2011 | 4000006 | 152.46 | Jumping Bungee Jumping | St. Petersburg | Florida | credit | |
| 05-29-2011 | 4000007 | 180.28 | Outdoor Recreation | Archery | Reno | Nevada | credit |
| 06-18-2011 | 4000009 | 121.39 | Outdoor Play Equipment | Swing Sets | Columbus | Ohio | credit |
| 02-08-2011 | 4000009 | 41.52 | Indoor Games | Bowling | San Francisco | California | credit |
| 03-13-2011 | 4000010 | 107.8 | Team Sports | Field Hockey | Honolulu | Hawaii | credit |
| 02-25-2011 | 4000010 | 36.81 | Gymnastics | Vaulting Horses | Los Angeles | California | credit |
| 10-20-2011 | 4000001 | 137.64 | Combat Sports | Fencing | Honolulu | Hawaii | credit |
| 05-28-2011 | 4000010 | 35.56 | Exercise & Fitness | Free Weight Bars | Columbia | South Carolina | credit |
| 10-18-2011 | 4000008 | 75.55 | Water Sports | Scuba Diving & Snorkeling | Omaha | Nebraska | credit |
| 11-18-2011 | 4000008 | 88.65 | Team Sports | Baseball | Salt Lake City | Utah | credit |
| 08-28-2011 | 4000008 | 51.81 | Water Sports | Life Jackets | Newark | New Jersey | credit |
| 06-29-2011 | 4000005 | 41.55 | Exercise & Fitness | Weightlifting Belts | New Orleans | Louisiana | credit |
| 02-14-2011 | 4000005 | 45.79 | Air Sports | Parachutes | New York | New York | credit |
| 10-10-2011 | 4000009 | 19.64 | Water Sports | Kitesurfing | Saint Paul | Minnesota | credit |
| 05-02-2011 | 4000009 | 99.5 | Gymnastics | Gymnastics Rings | Springfield | Illinois | credit |
| 06-10-2011 | 4000003 | 151.2 | Water Sports | Surfing | Plano | Texas | credit |
| 10-14-2011 | 4000009 | 144.2 | Indoor Games | Darts | Phoenix | Arizona | credit |
| 10-11-2011 | 4000009 | 21.58 | Combat Sports | Wrestling | Orange | California | credit |
| 09-29-2011 | 4000010 | 66.4 | Games | Mahjong | Fremont | California | credit |
| 05-12-2011 | 4000008 | 79.78 | Team Sports | Cricket | Lexington | Kentucky | credit |
| 06-03-2011 | 4000001 | 126.9 | Outdoor Recreation | Hunting | Phoenix | Arizona | credit |
| 03-14-2011 | 4000001 | 47.05 | Water Sports | Swimming | Lincoln | Nebraska | credit |
| 11-28-2011 | 4000008 | 5.03 | Games | Dice & Dice Sets | Los Angeles | California | credit |
| 01-28-2011 | 4000008 | 29.13 | Team Sports | Soccer | Springfield | Illinois | credit |
| 06-15-2011 | 4000008 | 154.15 | Outdoor Recreation | Lawn Games | Nashville | Tennessee | credit |
| 05-06-2011 | 4000008 | 98.96 | Team Sports | Indoor Volleyball | Atlanta | Georgia | credit |

1. Find out the number of transaction done by each customer (These should be take up in module 8 itself)

Solution: *select fname, Count(*) from CUSTOMER a join TXNRECORDS b on a.custid = b.custno group by fname;*

hive> select fname,Count(*) from CUSTOMER a join TXNRECORDS b on a.custid = b.custno
> group by fname;

WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using Hive Tez releases.

Query ID = acadgild_20181118223436_081c501a-ef19-46cb-954b-4c568ec11fcb

Total jobs = 1

SLF4J: Class path contains multiple SLF4J bindings.

SLF4J: Found binding in [jar:file:/home/acadgild/install/hive/apache-hive-2.3.2-bin/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: Found binding in [jar:file:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]

SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.

SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]

2018-11-18 22:34:55 Starting to launch local task to process map join; maximum memory = 518979584

2018-11-18 22:34:58 Dump the side-table for tag: 0 with group count: 10 into file: file:/tmp/acadgild/b4459aa9-ff7b-408c-ae9-82b3884382da/hive_2018-11-18_22-34-36_629_6830541097363522879-1/-local-10005/HashTable-Stage-2/MapJoin-mapfile000-..hashtable

2018-11-18 22:34:58 Uploaded 1 file to: file:/tmp/acadgild/b4459aa9-ff7b-408c-ae9-82b3884382da/hive_2018-11-18_22-34-36_629_6830541097363522879-1/-local-10005/HashTable-Stage-2/MapJoin-mapfile000-..hashtable (556 bytes)

2018-11-18 22:34:58 End of local task; Time Taken: 3.417 sec.

Execution completed successfully

MapredLocal task succeeded

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>

Job running in-process (local Hadoop)

2018-11-18 22:35:04,253 Stage-2 map = 0%, reduce = 0%

2018-11-18 22:35:05,500 Stage-2 map = 100%, reduce = 100%

Ended Job = job_local1281783256_0001

MapReduce Jobs Launched:

Stage-Stage-2: HDFS Read: 22504 HDFS Write: 0 SUCCESS

Total MapReduce CPU Time Spent: 0 msec

OK

Dolores 6

Elsie 6

```
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
2018-11-18 22:34:55 Starting to launch local task to process map join; maximum memory = 518979584
2018-11-18 22:34:58 Dump the side-table for tag: 0 with group count: 10 into file: file:/tmp/acadgild/b4459aa9-ff7b-408c-ae9-82b3884382da/hive_2018-11-18_22-34-36_620_6830541097363522879-1/-local-10005/HashTable-Stage-2/MapJoin-mapfile000--.hashtable
2018-11-18 22:34:58 Uploaded 1 File to: file:/tmp/acadgild/b4459aa9-ff7b-408c-ae9-82b3884382da/hive_2018-11-18_22-34-36_620_6830541097363522879-1/-local-10005/HashTable-Stage-2/MapJoin-mapfile000--.hashtable (556 bytes)
2018-11-18 22:34:58 End of local task; Time Taken: 3.417 sec.
Execution completed successfully
MapredLocal task succeeded
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>
Job running in-process (local Hadoop)
2018-11-18 22:35:04,253 Stage-2 map = 0%, reduce = 0%
2018-11-18 22:35:05,500 Stage-2 map = 100%, reduce = 100%
Ended Job = job_local1281783256_0001
MapReduce Jobs Launched:
Stage-Stage-2:  HDFS Read: 22504 HDFS Write: 0 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Dolores 6
Elsie 6
Gretchen 5
Hazel 10
Karen 5
Kristina 8
Malcolm 6
Paige 6
Patrick 5
Sherri 3
Time taken: 29.016 seconds, Fetched: 10 row(s)
hive>
```

2. Create a new table called `TRANSACTIONS_COUNT`. This table should have 3 fields - `custid`, `fname` and `count`. (Again to be done in module 8)

Solution: `CREATE TABLE TRANSACTIONS_COUNT (`

`> custid INT,`

`> fname STRING,`

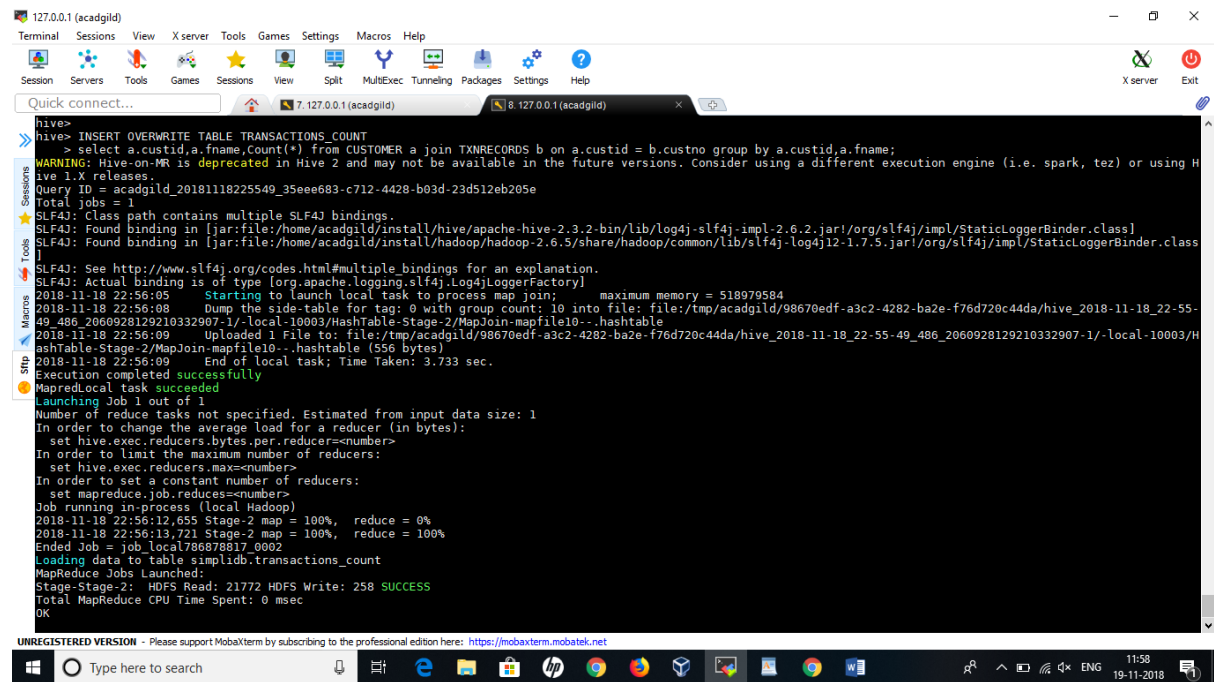
`> count INT)`

`> row format delimited fields terminated by ',';`

```
hive> CREATE TABLE TRANSACTIONS_COUNT (
>   custid INT,
>   fname STRING,
>   count INT)
> row format delimited fields terminated by ',';
OK
Time taken: 2.023 seconds
hive> select * from TRANSACTIONS_COUNT;
OK
Time taken: 4.986 seconds
hive>
```

3. Now write a hive query in such a way that the query populates the data obtained in Step 1 above and populate the table in step 2 above. (This has to be done in module 9).

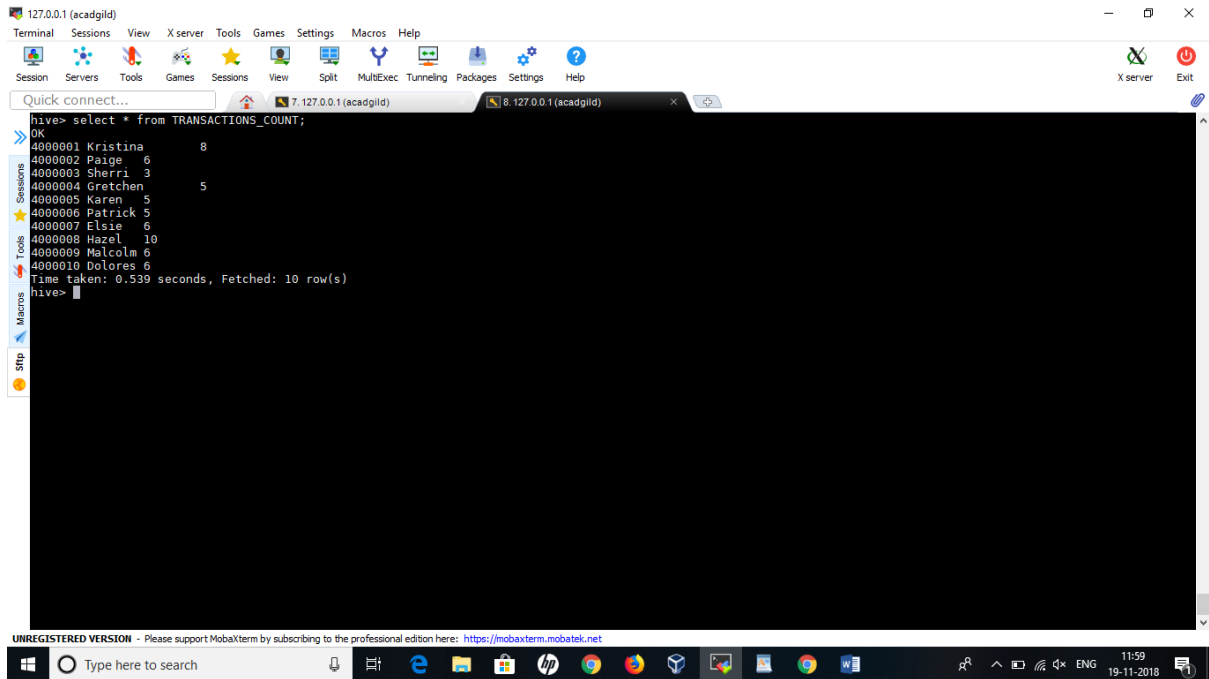
Solution: `INSERT OVERWRITE TABLE TRANSACTIONS_COUNT
select a.custid,a.fname,Count(*) from CUSTOMER a join TXNRECORDS b
on a.custid = b.custno group by a.custid,a.fname;`



```
127.0.0.1 (acadgild)
Terminal Sessions View X server Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect... 7. 127.0.0.1 (acadgild) 8. 127.0.0.1 (acadgild)

>> hive> INSERT OVERWRITE TABLE TRANSACTIONS_COUNT
>> > select a.custid,a.fname,Count(*) from CUSTOMER a join TXNRECORDS b
on a.custid = b.custno group by a.custid,a.fname;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using H
ive 1.X releases.
Query ID = acadgild_20181118225549_35eee683-c712-4428-b03d-23d512eb205e
Total jobs = 1
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/acadgild/install/hive/apache-hive-2.3.2-bin/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
2018-11-18 22:56:05 Starting to launch local task to process map join; maximum memory = 518970584
2018-11-18 22:56:08 Dump the side-table for tag: 0 with group count: 10 into file: file:/tmp/acadgild/98670edf-a3c2-4282-ba2e-f76d720c44da/hive_2018-11-18_22-55-
49_486_2060928129210332907-1/-local-10003/HashTable-Stage-2/MapJoin-mapfile10-..hashtable
2018-11-18 22:56:09 Uploaded 1 File to: file:/tmp/acadgild/98670edf-a3c2-4282-ba2e-f76d720c44da/hive_2018-11-18_22-55-49_486_2060928129210332907-1/-local-10003/H
ashTable-Stage-2/MapJoin-mapfile10-..hashtable (556 bytes)
2018-11-18 22:56:09 End of local task; Time Taken: 3.733 sec.
Execution completed successfully
MapredLocal task succeeded
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>
Job running in-process (local Hadoop)
2018-11-18 22:56:12,655 Stage-2 map = 100%, reduce = 0%
2018-11-18 22:56:13,721 Stage-2 map = 100%, reduce = 100%
Ended Job = job_local780878817_0002
Loading data to table simplidb.transactions_count
MapReduce Jobs Launched:
Stage-Stage-2: HDFS Read: 21772 HDFS Write: 258 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
```

TRANSACTIONS_COUNT Table: `select * from TRANSACTIONS_COUNT`



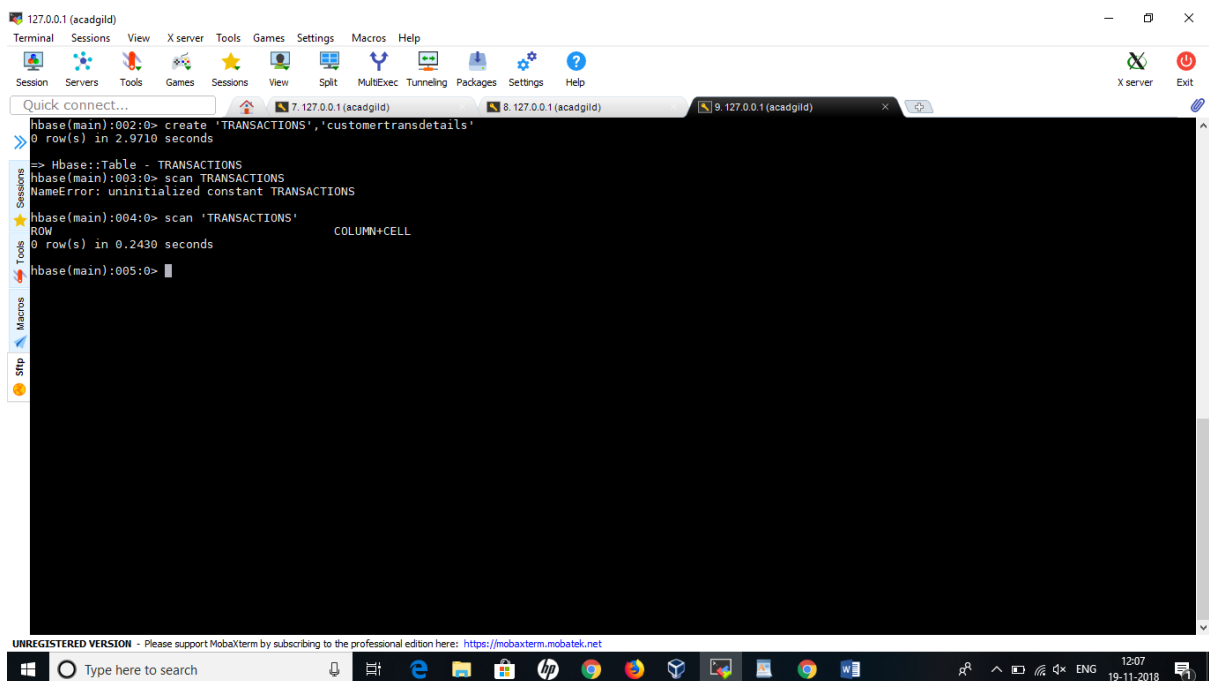
The screenshot shows a MobaXterm window with a terminal session. The terminal is running a Hive query: `hive> select * from TRANSACTIONS_COUNT;`. The output displays a list of names and their corresponding counts. The data is as follows:

| ID | Name | Count |
|---------|----------|-------|
| 4000001 | Kristina | 8 |
| 4000002 | Paige | 6 |
| 4000003 | Sherri | 3 |
| 4000004 | Gretchen | 5 |
| 4000005 | Karen | 5 |
| 4000006 | Patrick | 5 |
| 4000007 | Elsie | 6 |
| 4000008 | Hazel | 10 |
| 4000009 | Malcolm | 6 |
| 4000010 | Dolores | 6 |

The terminal also shows the execution time: `Time taken: 0.539 seconds, Fetched: 10 row(s)`. The MobaXterm interface includes a menu bar, a toolbar, and a sidebar with session and tool icons.

4. Now lets make the TRANSACTIONS_COUNT table Hbase complaint. In the sence, use Ser Des And Storate handler features of hive to change the TRANSACTIONS_COUNT table to be able to create a TRANSACTIONS table in Hbase. (This has to be done in module 10)

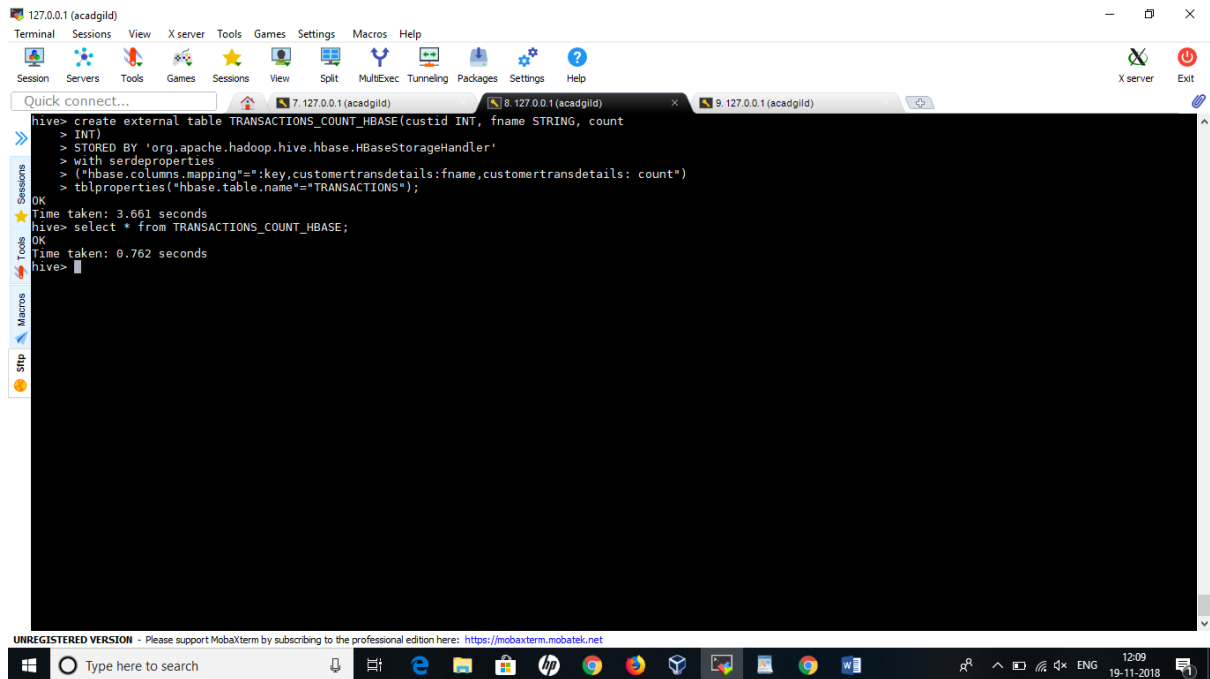
Solution: HBASE TABLE: `create 'TRANSACTIONS','customertransdetails'`



The screenshot shows a MobaXterm window with a terminal session. The terminal is running Hbase commands. The first command is `hbase(main):002:0> create 'TRANSACTIONS','customertransdetails'`, which returns `0 row(s) in 2.9710 seconds`. The second command is `hbase(main):003:0> scan 'TRANSACTIONS'`, which returns an error: `NameError: uninitialized constant TRANSACTIONS`. The third command is `hbase(main):004:0> scan 'TRANSACTIONS'`, which returns `0 row(s) in 0.2430 seconds`. The fourth command is `hbase(main):005:0>`, which is currently empty. The MobaXterm interface includes a menu bar, a toolbar, and a sidebar with session and tool icons.

```
HIVE TABLE: create external table TRANSACTIONS_COUNT_HBASE(custid
INT, fname STRING, count
INT)
STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler'
with serdeproperties
("hbase.columns.mapping"=":key,customertransdetails:fname,customertr
ansdetails: count")
tblproperties("hbase.table.name"="TRANSACTIONS");

select * from TRANSACTIONS_COUNT_HBASE;
```



The screenshot shows a MobaXterm terminal window with three tabs, all connected to 127.0.0.1 (acadgild). The active terminal shows the following commands and output:

```
hive> create external table TRANSACTIONS_COUNT_HBASE(custid INT, fname STRING, count
> INT)
> STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler'
> with serdeproperties
> ("hbase.columns.mapping"=":key,customertransdetails:fname,customertransdetails: count")
> tblproperties("hbase.table.name"="TRANSACTIONS");
OK
Time taken: 3.661 seconds
hive> select * from TRANSACTIONS_COUNT_HBASE;
OK
Time taken: 0.762 seconds
hive>
```

The terminal window has a standard Windows taskbar at the bottom with the date 19-11-2018 and time 12:09.

5. Now insert the data in TRANSACTIONS_COUNT table using the query in step 3 again, this should populate the Hbase TRANSACTIONS table automatically (This has to be done in module 10).

Solution: `INSERT OVERWRITE TABLE TRANSACTIONS_COUNT_HBASE`
`select a.custid,a.fname,Count(*) from CUSTOMER a join TXNRECORDS b`
`on a.custid = b.custno group by a.custid,a.fname;`

```
127.0.0.1 (acadgild)
Terminal Sessions View Xserver Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect... 7. 127.0.0.1 (acadgild) 8. 127.0.0.1 (acadgild) 9. 127.0.0.1 (acadgild)
hive> INSERT OVERWRITE TABLE TRANSACTIONS_COUNT_HBASE
> select a.custid,a.fname,count(*) from CUSTOMER a join TXNRECORDS b on a.custid = b.custno group by a.custid,a.fname;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execution engine (i.e. spark, tez) or using H
ive 1.X releases.
Query ID = acadgild_20181118230938_fc03f172-1948-45d0-8f16-2d2533dab03f
Total jobs = 1
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/acadgild/install/hive/apache-hive-2.3.2-bin/lib/log4j-slf4j-impl-2.6.2.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: See http://www.slf4j.org/codes.html#multiple_bindings for an explanation.
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
2018-11-18 23:09:57 Starting to launch local task to process map join; maximum memory = 518979584
2018-11-18 23:10:02 Dump the side-table for tag: 0 with group count: 10 into file: file:/tmp/acadgild/98670edf-a3c2-4282-ba2e-f76d720c44da/hive_2018-11-18_23-09-
38_352_6020650838966490622-1/-local-10002/HashTable-Stage-4/MapJoin-mapfile20--.hashtable
2018-11-18 23:10:02 Uploaded 1 file to: file:/tmp/acadgild/98670edf-a3c2-4282-ba2e-f76d720c44da/hive_2018-11-18_23-09-38_352_6020650838966490622-1/-local-10002/H
ashTable-Stage-4/MapJoin-mapfile20--.hashtable (556 bytes)
2018-11-18 23:10:02 End of local task; Time Taken: 4.845 sec.
Execution completed successfully
MapredLocal task succeeded
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>
Job running in-process (local Hadoop)
2018-11-18 23:10:07,739 Stage-4 map = 100%, reduce = 0%
2018-11-18 23:10:08,757 Stage-4 map = 100%, reduce = 100%
Ended Job = job_local1534598966_0003
MapReduce Jobs Launched:
Stage-Stage-4: HDFS Read: 33174 HDFS Write: 516 SUCCESS
Total MapReduce CPU Time Spent: 0 msec
OK
Time taken: 30.45 seconds
hive>
```

```
127.0.0.1 (acadgild)
Terminal Sessions View Xserver Tools Games Settings Macros Help
Session Servers Tools Games Sessions View Split MultiExec Tunneling Packages Settings Help
Quick connect... 7. 127.0.0.1 (acadgild) 8. 127.0.0.1 (acadgild) 9. 127.0.0.1 (acadgild)
hive> select * from TRANSACTIONS_COUNT_HBASE;
OK
4000001 Kristina      8
4000002 Paige        6
4000003 Sherri       3
4000004 Gretchen     5
4000005 Karen        5
4000006 Patrick      5
4000007 Elsie        6
4000008 Hazel       10
4000009 Malcolm      6
4000010 Dolores      6
Time taken: 0.617 seconds, Fetched: 10 row(s)
hive>
```

```
hbase(main):005:0> scan 'TRANSACTIONS'
ROW
4000001  COLUMN+CELL
column=customertransdetails: count, timestamp=1542562807709, value=8
column=customertransdetails: fname, timestamp=1542562807709, value=Kristina
4000002  column=customertransdetails: count, timestamp=1542562807709, value=6
column=customertransdetails: fname, timestamp=1542562807709, value=Paige
4000003  column=customertransdetails: count, timestamp=1542562807709, value=3
column=customertransdetails: fname, timestamp=1542562807709, value=Sherri
4000004  column=customertransdetails: count, timestamp=1542562807709, value=5
column=customertransdetails: fname, timestamp=1542562807709, value=Gretchen
4000005  column=customertransdetails: count, timestamp=1542562807709, value=5
column=customertransdetails: fname, timestamp=1542562807709, value=Karen
4000006  column=customertransdetails: count, timestamp=1542562807709, value=5
column=customertransdetails: fname, timestamp=1542562807709, value=Patrick
4000007  column=customertransdetails: count, timestamp=1542562807709, value=6
column=customertransdetails: fname, timestamp=1542562807709, value=Elsie
4000008  column=customertransdetails: count, timestamp=1542562807709, value=10
column=customertransdetails: fname, timestamp=1542562807709, value=Hazel
4000009  column=customertransdetails: count, timestamp=1542562807709, value=6
column=customertransdetails: fname, timestamp=1542562807709, value=Malcolm
4000010  column=customertransdetails: count, timestamp=1542562807709, value=6
column=customertransdetails: fname, timestamp=1542562807709, value=Dolores
10 row(s) in 0.2030 seconds
hbase(main):006:0>
```

6. Now from the Hbase level, write the Hbase java API code to access and scan the TRANSACTIONS table data from java level.

Solution: package com.hbase.api;

import java.io.IOException;

import org.apache.hadoop.conf.Configuration;

import org.apache.hadoop.hbase.HBaseConfiguration;

import org.apache.hadoop.hbase.client.HTable;

import org.apache.hadoop.hbase.client.Result;

import org.apache.hadoop.hbase.client.ResultScanner;

import org.apache.hadoop.hbase.client.Scan;

import org.apache.hadoop.hbase.util.Bytes;

public class ScanData {

public static void main(String args[])throws IOException {

Configuration conf = HBaseConfiguration.create();

@SuppressWarnings("deprecation")

HTable table = new HTable(conf,"TRANSACTIONS");

Scan scan = new Scan();

scan.addColumn(Bytes.toBytes("customertransdetails"),Bytes.toBytes("count"));

scan.addColumn(Bytes.toBytes("customertransdetails"),Bytes.toBytes("fname"));


```

ResultScanner result = table.getScanner(scan);

for(Result res:result){

byte[] val = res.getValue(Bytes.toBytes("customertransdetails"),
Bytes.toBytes("count"));

byte[] val1 = res.getValue(Bytes.toBytes("customertransdetails"),
Bytes.toBytes("fname"));

System.out.println("Row-value : "+Bytes.toString(val));

System.out.println("Row-value : "+Bytes.toString(val1));

System.out.println(res);

}

table.close();

}

}

```

