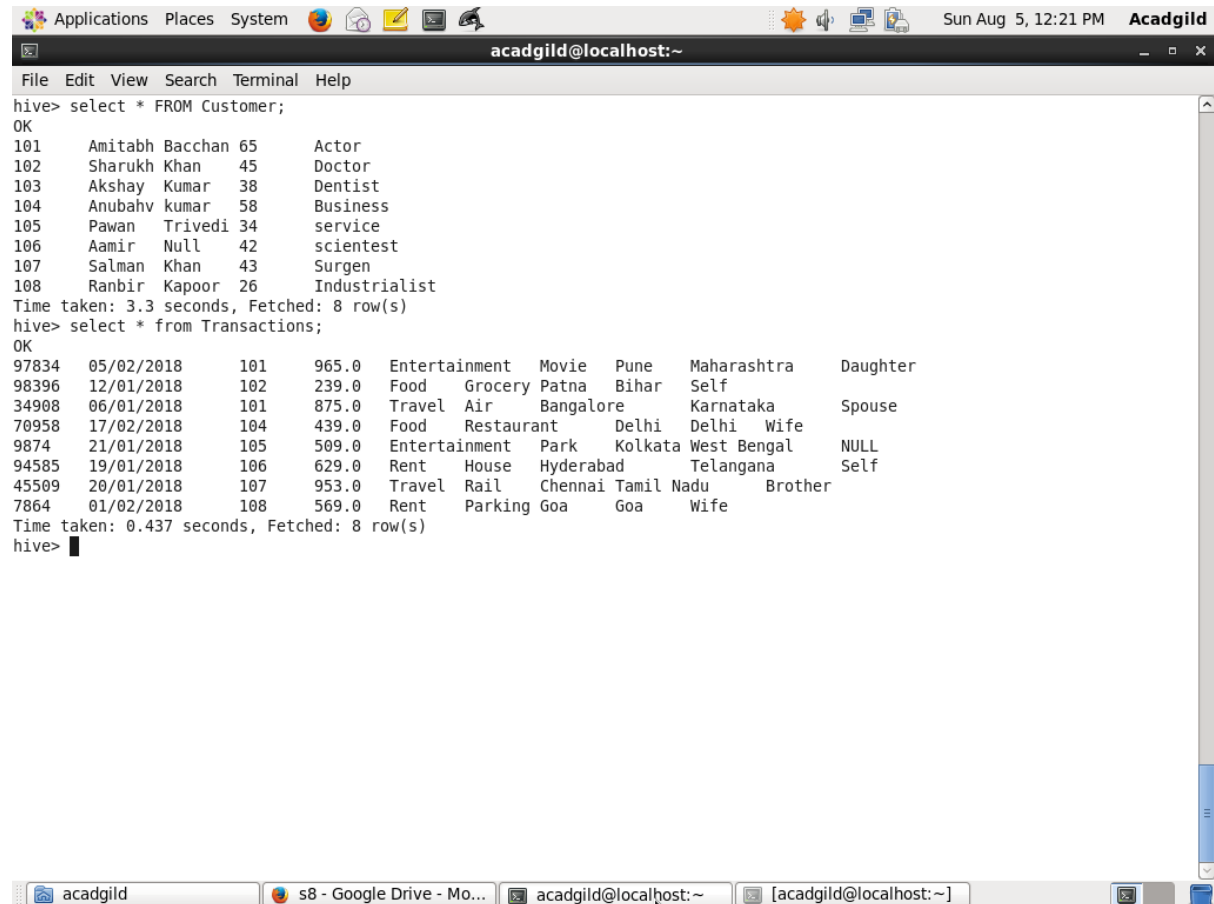


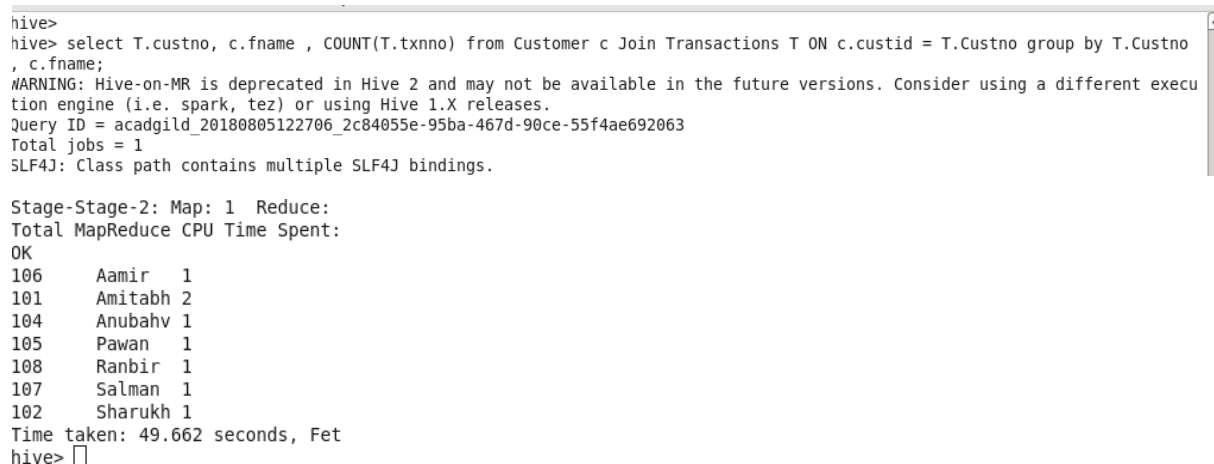
## Customer And Transaction

### Contents of Customer and Transactions Table:



```
Applications Places System [Icons] Sun Aug 5, 12:21 PM Acadgild
acadgild@localhost:~
File Edit View Search Terminal Help
hive> select * FROM Customer;
OK
101      Amitabh Bacchan 65      Actor
102      Sharukh Khan  45      Doctor
103      Akshay Kumar  38      Dentist
104      Anubahv kumar  58      Business
105      Pawan Trivedi  34      service
106      Aamir Null 42      scientest
107      Salman Khan  43      Surgen
108      Ranbir Kapoor 26      Industrialist
Time taken: 3.3 seconds, Fetched: 8 row(s)
hive> select * from Transactions;
OK
97834    05/02/2018    101    965.0    Entertainment    Movie    Pune    Maharashtra    Daughter
98396    12/01/2018    102    239.0    Food    Grocery    Patna    Bihar    Self
34908    06/01/2018    101    875.0    Travel    Air    Bangalore    Karnataka    Spouse
70958    17/02/2018    104    439.0    Food    Restaurant    Delhi    Delhi    Wife
9874     21/01/2018    105    509.0    Entertainment    Park    Kolkata    West Bengal    NULL
94585    19/01/2018    106    629.0    Rent    House    Hyderabad    Telangana    Self
45509    20/01/2018    107    953.0    Travel    Rail    Chennai    Tamil Nadu    Brother
7864     01/02/2018    108    569.0    Rent    Parking    Goa    Goa    Wife
Time taken: 0.437 seconds, Fetched: 8 row(s)
hive>
```

### 1. Find out the number of transaction done by each customer



```
hive>
hive> select T.custno, c.fname , COUNT(T.txnno) from Customer c Join Transactions T ON c.custid = T.Custno group by T.Custno
, c.fname;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execu
tion engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = acadgild_20180805122706_2c84055e-95ba-467d-90ce-55f4ae692063
Total jobs = 1
SLF4J: Class path contains multiple SLF4J bindings.

Stage-Stage-2: Map: 1 Reduce:
Total MapReduce CPU Time Spent:
OK
106      Aamir 1
101      Amitabh 2
104      Anubahv 1
105      Pawan 1
108      Ranbir 1
107      Salman 1
102      Sharukh 1
Time taken: 49.662 seconds, Fet
hive>
```

### 2. Create a new table called TRANSACTIONS\_COUNT. This table should have 3 fields - custid, fname and count.

```
hive> CREATE TABLE TRANSACTIONS_COUNT
> (
> Custid INT,
> fname STRING,
> Count INT
> );
```

```
OK
Time taken: 0.257 seconds
```

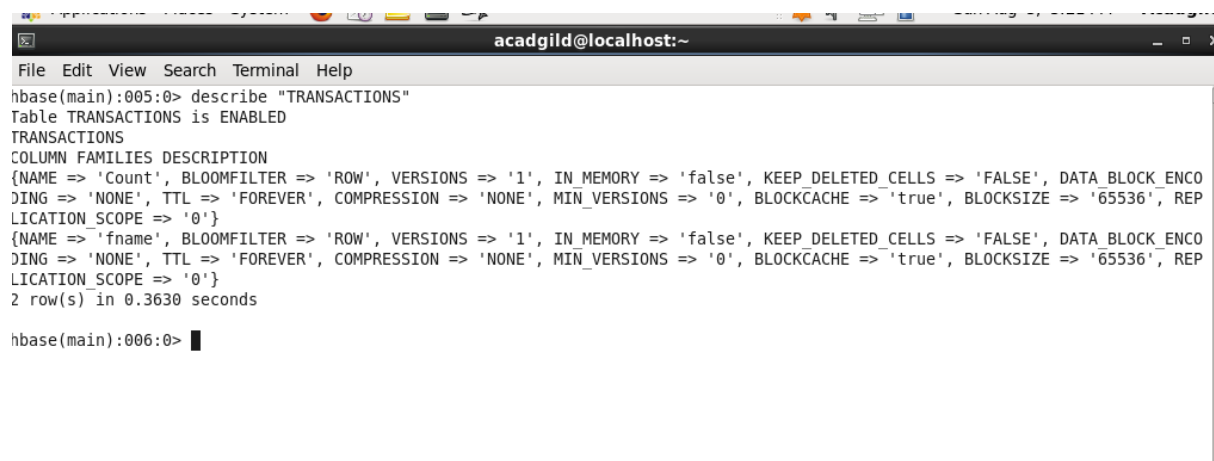
### 3. Now write a hive query in such a way that the query populates the data obtained in Step 1 above and populate the table in step 2 above.

```
hive> Insert Into TRANSACTIONS_COUNT select T.custno, c.fname, COUNT(T.txnno) from Customer c Join Transactions T ON c.custid = T.Custno group by T.Custno, c.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 1.X releases.
Query ID = acadqild 20180805124455 f95408ae-9eal-4b37-a274-6931c8de5224
```

```
hive> select * From Transactions_count;
OK
106    Aamir    1
101    Amitabh  2
104    Anubahv  1
105    Pawan   1
108    Ranbir   1
107    Salman   1
102    Sharukh  1
Time taken: 0.316 seconds, Fetched: 7 row(s)
hive> █
```

### 4. Now lets make the TRANSACTIONS\_COUNT table Hbase compliant. 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.

```
Time taken: 0.063 seconds
hive> Create table hbase value 1(custid int,fname string,Count int)
> STORED BY 'org.apache.hadoop.hive.hbase.HBaseStorageHandler'
> with serdeproperties
> ("hbase.columns.mapping"=":key,fname:name,Count:Count")
> tblproperties("hbase.table.name"="TRANSACTIONS");
OK
Time taken: 6.612 seconds
hive> █
```



```
acadgild@localhost:~
File Edit View Search Terminal Help
hbase(main):005:0> describe "TRANSACTIONS"
Table TRANSACTIONS is ENABLED
TRANSACTIONS
COLUMN FAMILIES DESCRIPTION
{NAME => 'Count', BLOOMFILTER => 'ROW', VERSIONS => '1', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'FALSE', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE', MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0'}
{NAME => 'fname', BLOOMFILTER => 'ROW', VERSIONS => '1', IN_MEMORY => 'false', KEEP_DELETED_CELLS => 'FALSE', DATA_BLOCK_ENCODING => 'NONE', TTL => 'FOREVER', COMPRESSION => 'NONE', MIN_VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE => '65536', REPLICATION_SCOPE => '0'}
2 row(s) in 0.3630 seconds
hbase(main):006:0> █
```

### 5. Now insert the data in TRANSACTIONS\_COUNT table using the query in step 3 again, this should populate the Hbase TRANSACTIONS table automatically

```

hive> INSERT OVERWRITE TABLE hbase_value_1 select T.custno, c.fname , COUNT(T.txnno) from Customer c Join Transactions T ON c
.custid = T.Custno group by T.Custno , c.fname;
WARNING: Hive-on-MR is deprecated in Hive 2 and may not be available in the future versions. Consider using a different execu
tion engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = acadgild_20180805171959_655a6add-73b9-4f66-b709-737b3a11b032
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]

OK
Time taken: 75.193 seconds
hive> select * from hbase_value_1;
OK
101      Amitabh 2
102      Sharukh 1
104      Anubahv 1
105      Pawan   1
106      Aamir   1
107      Salman  1
108      Ranbir  1

```

```

hbase(main):007:0> SCAN "TRANSACTIONS"
NoMethodError: undefined method `SCAN' for #<Object:0x4391a2d8>

```

```

hbase(main):008:0> scan "TRANSACTIONS"
ROW                                COLUMN+CELL
101                                column=Count:Count, timestamp=1533469871499, value=2
101                                column=fname:name, timestamp=1533469871499, value=Amitabh
102                                column=Count:Count, timestamp=1533469871499, value=1
102                                column=fname:name, timestamp=1533469871499, value=Sharukh
104                                column=Count:Count, timestamp=1533469871499, value=1
104                                column=fname:name, timestamp=1533469871499, value=Anubahv
105                                column=Count:Count, timestamp=1533469871499, value=1
105                                column=fname:name, timestamp=1533469871499, value=Pawan
106                                column=Count:Count, timestamp=1533469871499, value=1
106                                column=fname:name, timestamp=1533469871499, value=Aamir
107                                column=Count:Count, timestamp=1533469871499, value=1
107                                column=fname:name, timestamp=1533469871499, value=Salman
108                                column=Count:Count, timestamp=1533469871499, value=1
108                                column=fname:name, timestamp=1533469871499, value=Ranbir
7 row(s) in 0.4390 seconds

```

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

**Code:**

#### JAVA API to SCAN

```

import java.io.IOException;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.hbase.HBaseConfiguration;
import org.apache.hadoop.hbase.HTableDescriptor;
import org.apache.hadoop.hbase.client.HBaseAdmin;
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 ScanT {

    public static void main(String args[])

```

```

        throws IOException{

    Configuration c = HBaseConfiguration.create();    // Instantiate Configuration class

    HTable table = new HTable(c,"TRANSACTIONS");    // Instantiate HTable class

    Scan scan = new Scan();    // Instantiate the Scan class


    HBaseAdmin admin = new HBaseAdmin(c);

    // Getting all the list of tables using HBaseAdmin object
    HTableDescriptor[] tableDescriptor = admin.listTables();

    // printing all the table names.
    for (int i=0; i<tableDescriptor.length;i++){
        System.out.println(tableDescriptor[i].getNameAsString());
    }

    scan.addColumn(Bytes.toBytes("fname"),Bytes.toBytes("name")); // Scan the required
columns
    scan.addColumn(Bytes.toBytes("Count"),Bytes.toBytes("Count")); // Scan the required
columns

    ResultScanner scanner = table.getScanner(scan);    // Get scan result

    // Reading values from scan result

    for (Result result = scanner.next();result != null;
        result = scanner.next())

    System.out.println( result);

    scanner.close();    //close the scanner

}

}

```

```

You have new mail in /var/spool/mail/acadgild
[acadgild@localhost src]$ javac ScanT.java
Note: ScanT.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
[acadgild@localhost src]$ java ScanT
log4j:WARN No appenders could be found for logger (org.apache.hadoop.security.Groups).
log4j:WARN Please initialize the log4j system properly.
log4j:WARN See http://logging.apache.org/log4j/1.2/faq.html#noconfig for more info.
TRANSACTIONS
keyvalues={101/Count:Count/1533469871499/Put/vlen=1/seqid=0, 101/fname:name/1533469871499/Put/vlen=7/seqid=0}
keyvalues={102/Count:Count/1533469871499/Put/vlen=1/seqid=0, 102/fname:name/1533469871499/Put/vlen=7/seqid=0}
keyvalues={104/Count:Count/1533469871499/Put/vlen=1/seqid=0, 104/fname:name/1533469871499/Put/vlen=7/seqid=0}
keyvalues={105/Count:Count/1533469871499/Put/vlen=1/seqid=0, 105/fname:name/1533469871499/Put/vlen=5/seqid=0}
keyvalues={106/Count:Count/1533469871499/Put/vlen=1/seqid=0, 106/fname:name/1533469871499/Put/vlen=5/seqid=0}
keyvalues={107/Count:Count/1533469871499/Put/vlen=1/seqid=0, 107/fname:name/1533469871499/Put/vlen=6/seqid=0}
keyvalues={108/Count:Count/1533469871499/Put/vlen=1/seqid=0, 108/fname:name/1533469871499/Put/vlen=6/seqid=0}
[acadgild@localhost src]$ █

```