Case Study 2

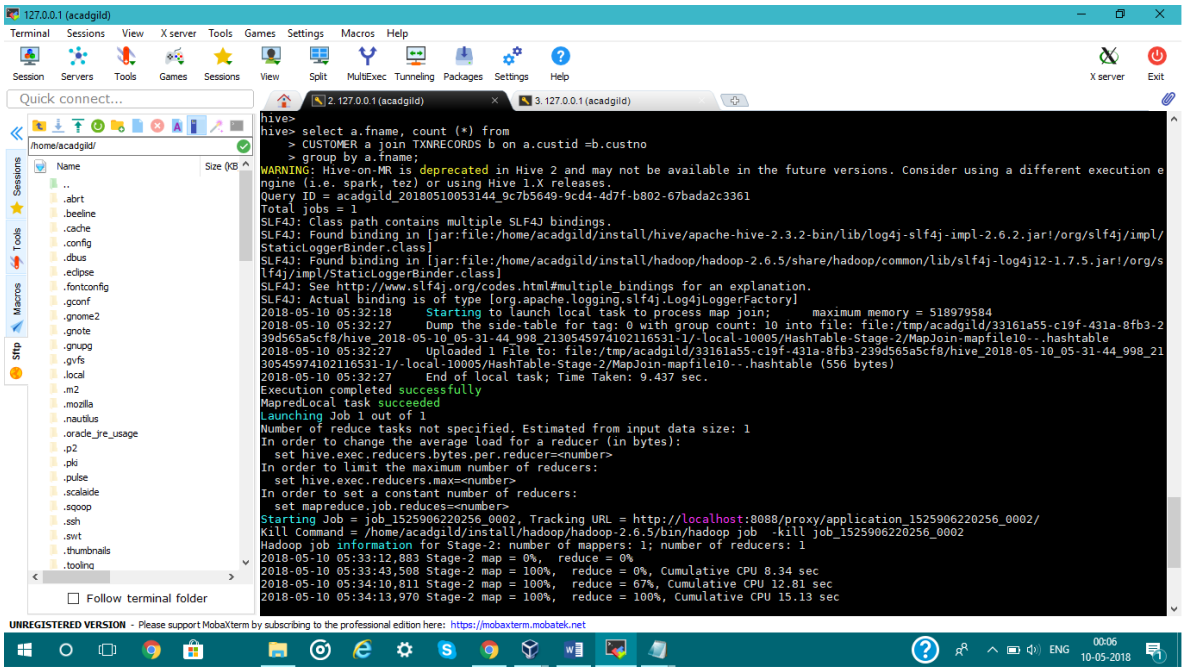
Task1

Find out the number of transaction done by each customer

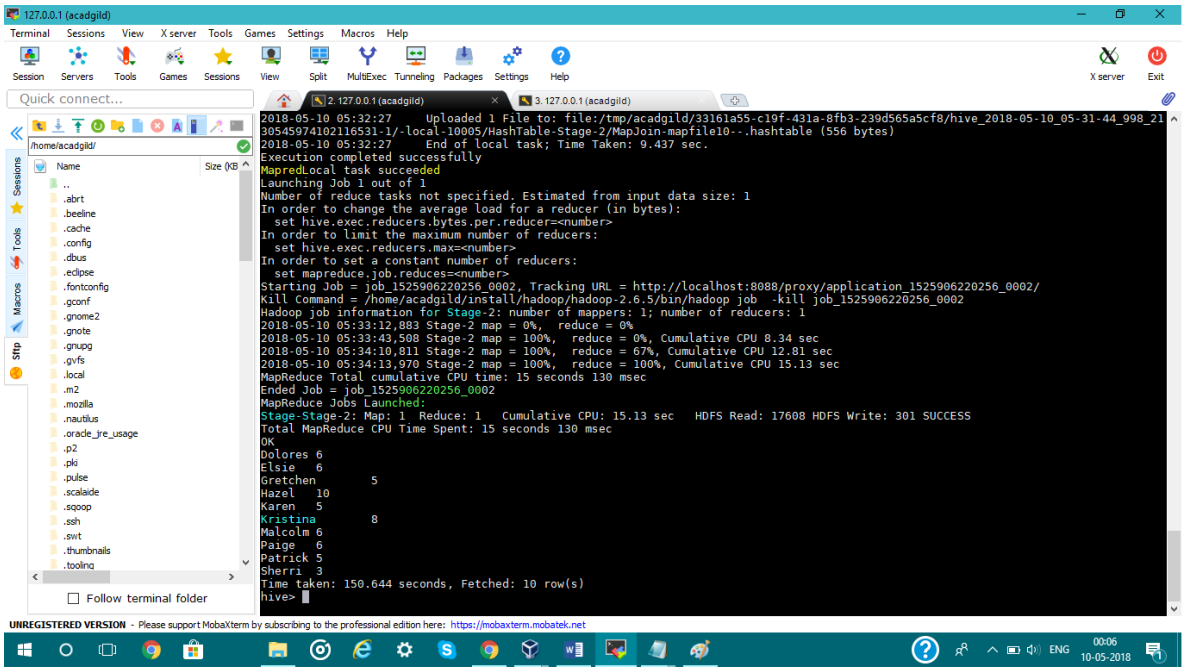
Command

Select a.fname, count(*) from Customers a join Transactions b on a.custid=b.custno group by a.fname

Screenshot



Output



Task2

Create a new table called TRANSACTIONS_COUNT. This table should have

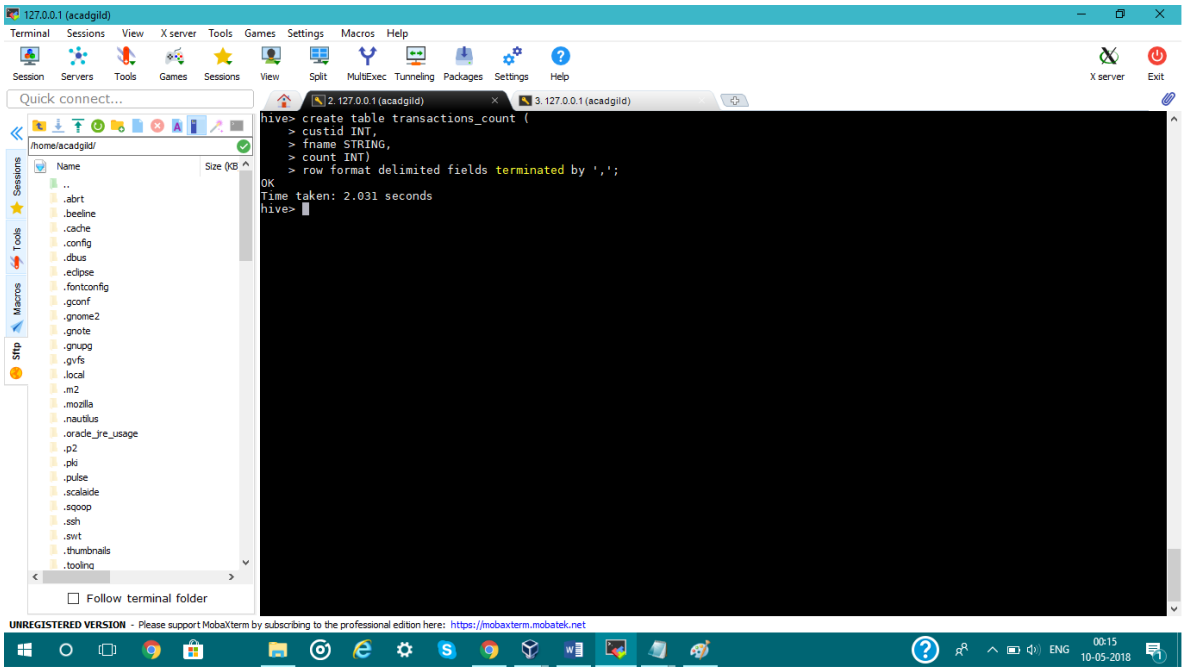
3 fields - custid, fname and count.

Command

create table transactions_count (custid INT, fname STRING, count INT)

row format delimited fields terminated by ‘,’;

Screenshot



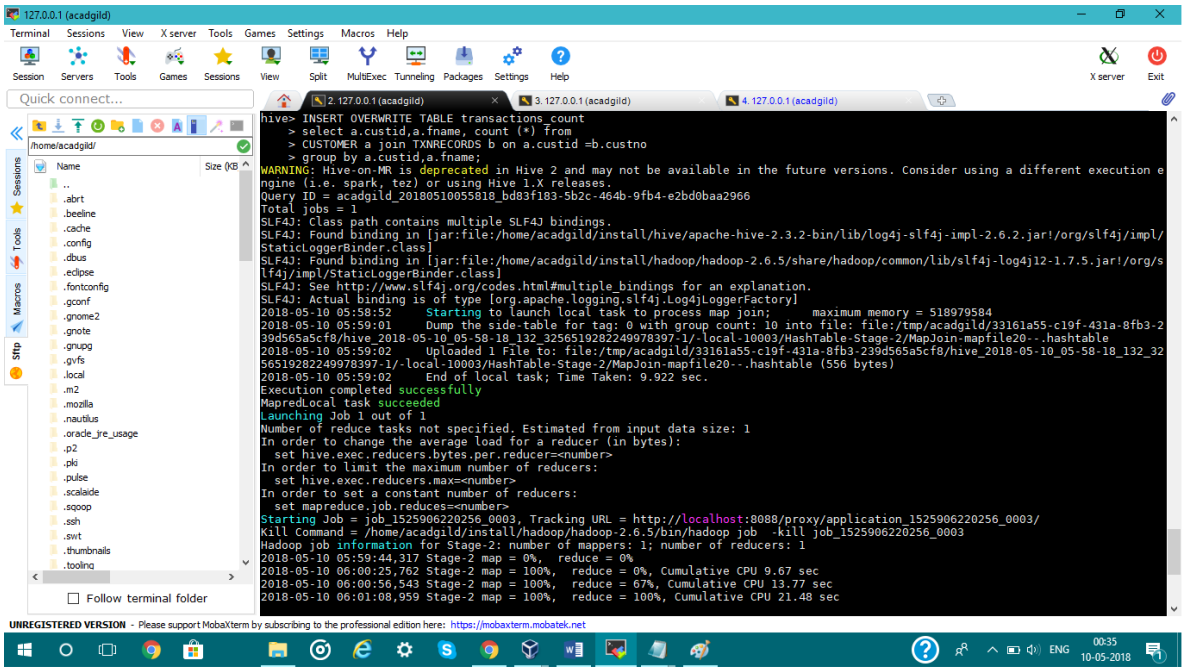
Task3

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.

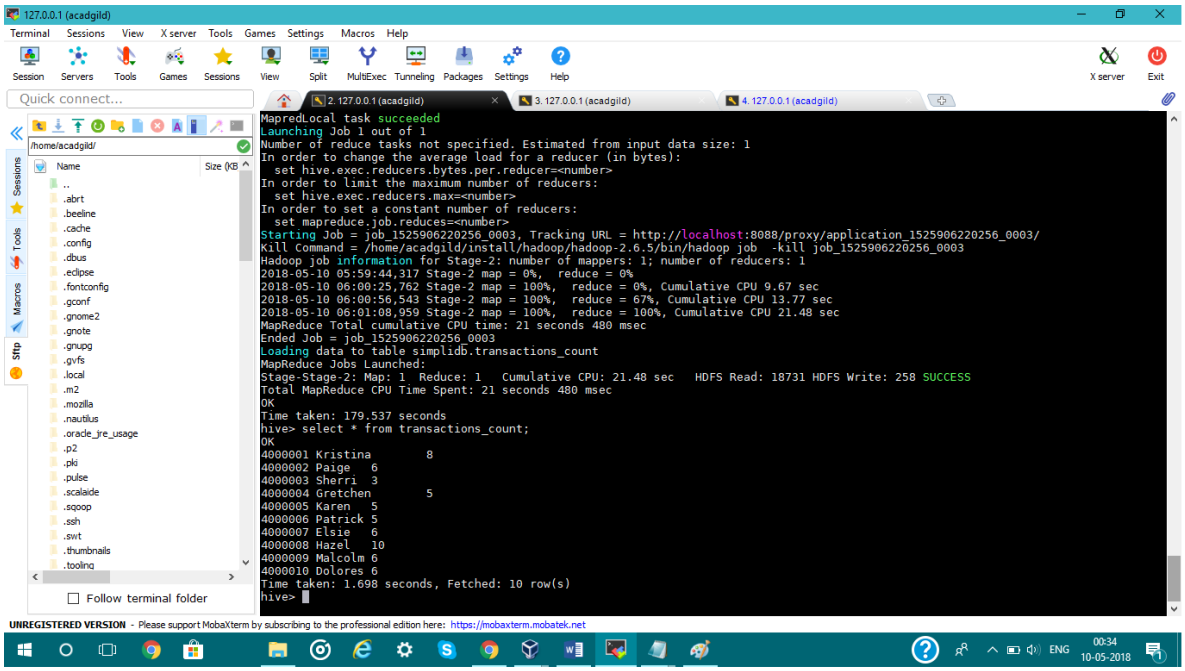
Command

Insert overwrite table transactions_count Select a.custid, a.fname, count(*) from Customers a join Transactions b on a.custid=b.custno group by a.custid, a.fname

Screenshot



Output



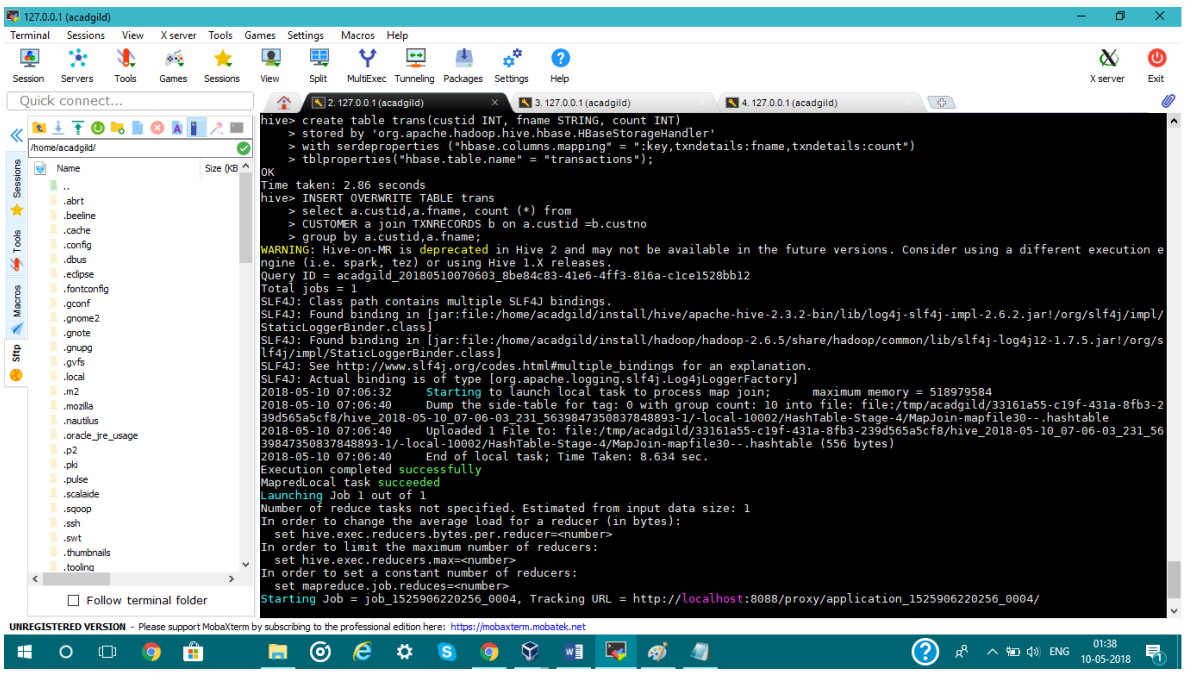
Task4

Make the TRANSACTIONS_COUNT table Hbase complaint

Command

Create table trans (custid INT, fname STRING, count INT) stored by ‘org.apache.hadoop.hive.hbase.HBasestorageHandler’ with serdeproperties (‘hbase.columns.mapping”=:”key.txndetails:fname, txndetails:count”) tblproperties(“hbase.table.name”=”transactions”);

Screenshot



Task5

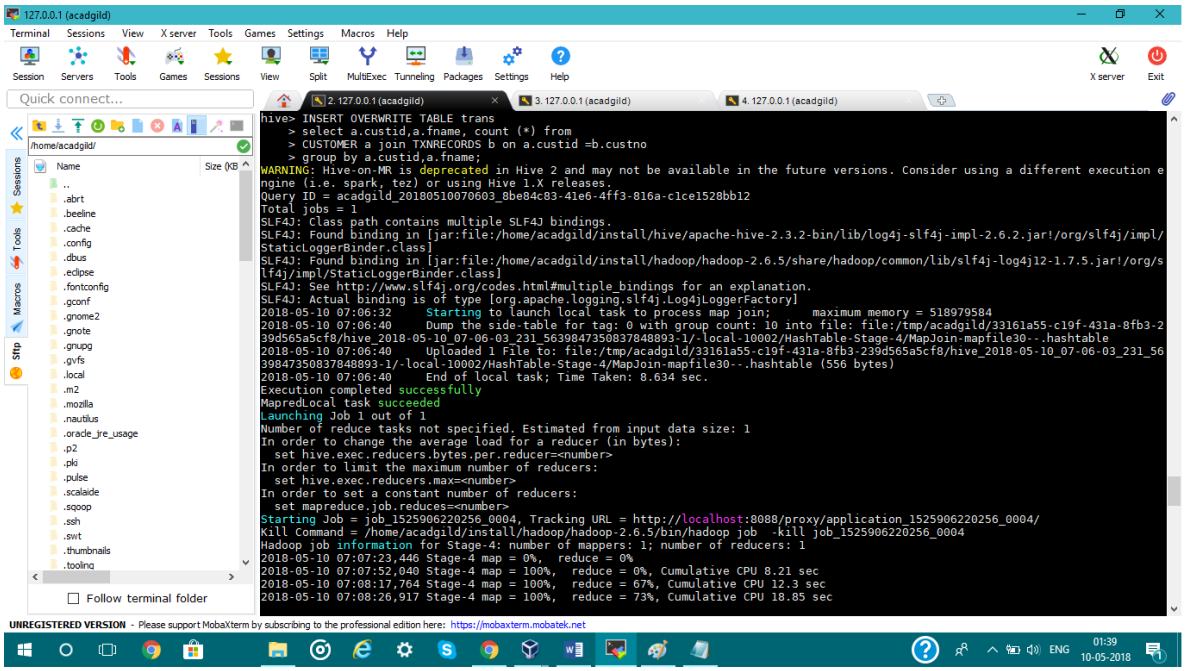
Now insert the data in TRANSACTIONS_COUNT table using the query in step

3 again, this should populate the Hbase TRANSACTIONS table automatically

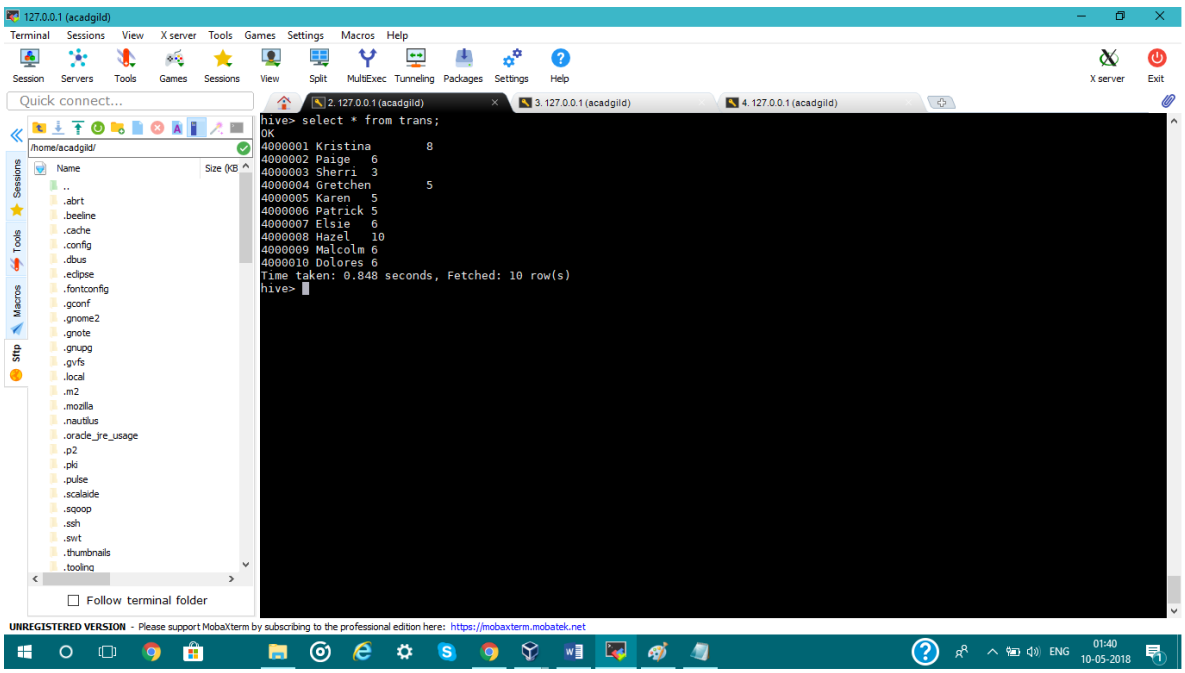
Command

Insert overwrite table trans Select a.custid, a.fname, count(*) from Customers a join Transactions b on a.custid=b.custno group by a.custid, a.fname

Screenshot



Output



Task6

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

Code

```
//importing all the packages
import java.io.IOException;
import java.util.ArrayList;
import java.util.List;
import org.apache.hadoop.conf.Configuration;
import org.apache.hadoop.hbase.HBaseConfiguration;
import org.apache.hadoop.hbase.HColumnDescriptor;
import org.apache.hadoop.hbase.HTableDescriptor;
import org.apache.hadoop.hbase.KeyValue;
import org.apache.hadoop.hbase.MasterNotRunningException;
import org.apache.hadoop.hbase.ZooKeeperConnectionException;
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 HbaseTable {                              //defining a class
 private static Configuration conf = null;  //declaring configurations
```

```
static {
        conf = HBaseConfiguration.create();//initialising configuration
    }
//scanning a HBase table
    public static void getAllRecord (String tableName) {
        try{
            HTable table = new HTable(conf, tableName);
            Scan s = new Scan();
            ResultScanner ss = table.getScanner(s);
            for(Result r:ss){
                for(KeyValue kv : r.raw()){
                    System.out.print(new String(kv.getRow()) + " ");         //for
fetching rowkey
                    System.out.print(new String(kv.getFamily()) + ":");      //for
fetching column family
                    System.out.print(new String(kv.getQualifier()) + " ");    //for
fetching columns
                    System.out.print(kv.getTimestamp() + " ");
                    System.out.println(new String(kv.getValue()));
                }
            }
        } catch (IOException e){
            e.printStackTrace();
        }
    }

    public static void main(String[] agrs) {
```

```
        try {
            String tablename = "transactions";          //initialising the name of the table in HBase
            String[] families = { "custid", "fname","count" };//initialising the column families
            System.out.println("===========show all record========");
            HbaseTable.getAllRecord(tablename);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

Screenshot