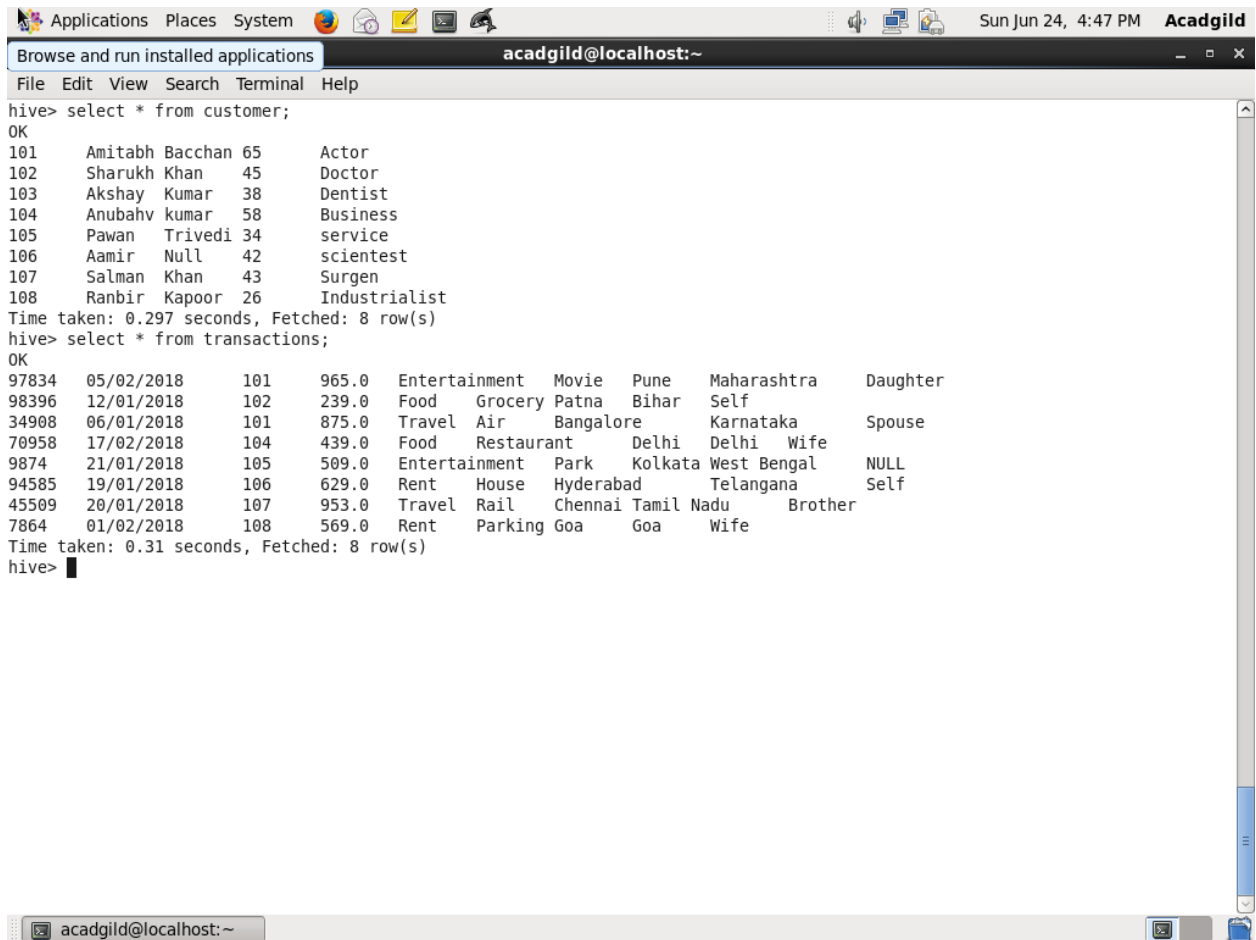


Data in customer and transactions table :



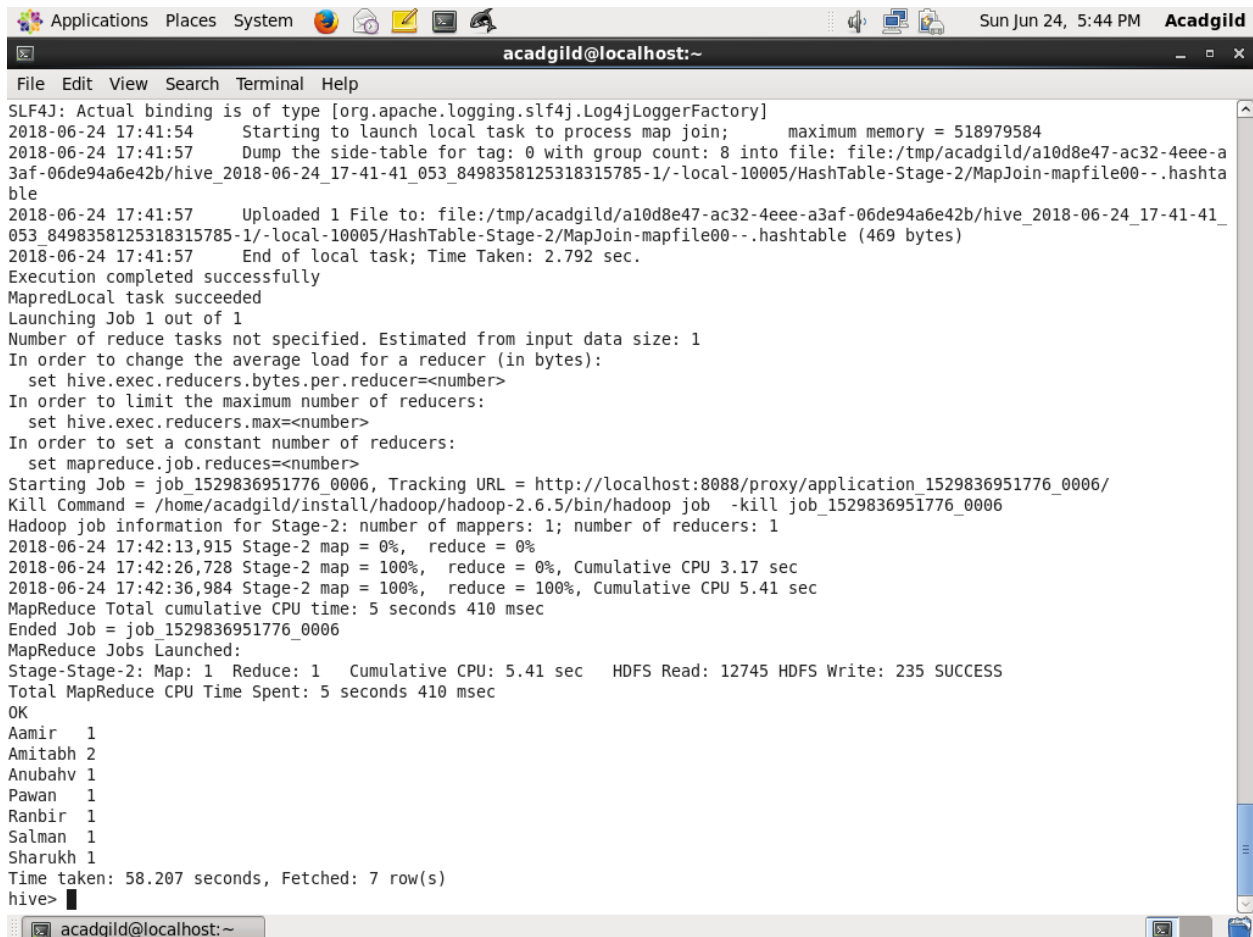
The screenshot shows a terminal window titled "acadgild@localhost:~" with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal displays the execution of two Hive SQL queries. The first query, "select * from customer;", returns 8 rows of customer data. The second query, "select * from transactions;", returns 8 rows of transaction data. The terminal output is as follows:

```
hive> select * from customer;
OK
101    Amitabh Bacchan 65      Actor
102    Sharukh Khan   45      Doctor
103    Akshay Kumar   38      Dentist
104    Anubhav 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: 0.297 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: 0.31 seconds, Fetched: 8 row(s)
hive>
```

Output of number of transactions done by customer :

Query :

```
select a.fname, count(*) from customer a left join transactions b on a.custid =  
b.custno group by a.fname;
```



The screenshot shows a terminal window titled 'acadgild@localhost:~' with a menu bar (File, Edit, View, Search, Terminal, Help). The terminal displays the following output:

```
SLF4J: Actual binding is of type [org.apache.logging.slf4j.Log4jLoggerFactory]
2018-06-24 17:41:54 Starting to launch local task to process map join; maximum memory = 518979584
2018-06-24 17:41:57 Dump the side-table for tag: 0 with group count: 8 into file: file:/tmp/acadgild/a10d8e47-ac32-4eee-a3af-06de94a6e42b/hive_2018-06-24_17-41-41_053_8498358125318315785-1/-local-10005/HashTable-Stage-2/MapJoin-mapfile00--.hashtable
2018-06-24 17:41:57 Uploaded 1 File to: file:/tmp/acadgild/a10d8e47-ac32-4eee-a3af-06de94a6e42b/hive_2018-06-24_17-41-41_053_8498358125318315785-1/-local-10005/HashTable-Stage-2/MapJoin-mapfile00--.hashtable (469 bytes)
2018-06-24 17:41:57 End of local task; Time Taken: 2.792 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>
Starting Job = job_1529836951776_0006, Tracking URL = http://localhost:8088/proxy/application_1529836951776_0006/
Kill Command = /home/acadgild/install/hadoop/hadoop-2.6.5/bin/hadoop job -kill job_1529836951776_0006
Hadoop job information for Stage-2: number of mappers: 1; number of reducers: 1
2018-06-24 17:42:13,915 Stage-2 map = 0%, reduce = 0%
2018-06-24 17:42:26,728 Stage-2 map = 100%, reduce = 0%, Cumulative CPU 3.17 sec
2018-06-24 17:42:36,984 Stage-2 map = 100%, reduce = 100%, Cumulative CPU 5.41 sec
MapReduce Total cumulative CPU time: 5 seconds 410 msec
Ended Job = job_1529836951776_0006
MapReduce Jobs Launched:
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 5.41 sec HDFS Read: 12745 HDFS Write: 235 SUCCESS
Total MapReduce CPU Time Spent: 5 seconds 410 msec
OK
Aamir 1
Amitabh 2
Anubahv 1
Pawan 1
Ranbir 1
Salman 1
Sharukh 1
Time taken: 58.207 seconds, Fetched: 7 row(s)
hive>
```

Creating new table TRANSACTIONS_COUNT

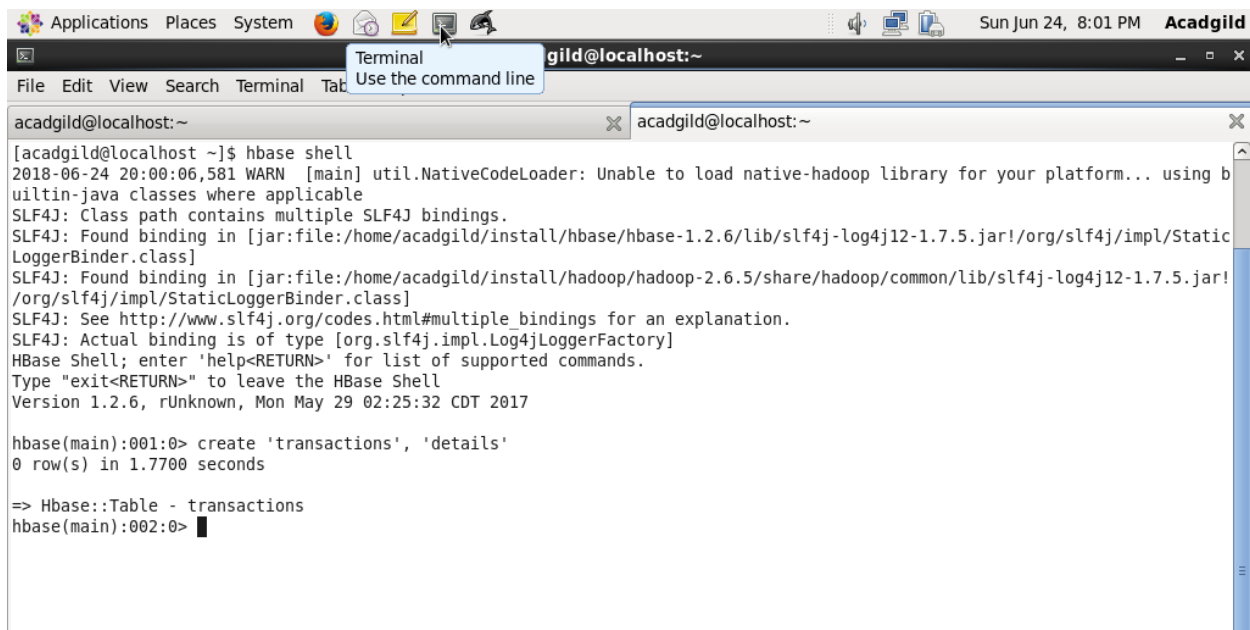
Populating TRANSACTIONS_COUNT table from above query

```
Applications Places System acadgild@localhost:~
File Edit View Search Terminal Help
hive> create table TRANSACTIONS_COUNT(
  > custid INT,
  > fname STRING,
  > count INT);
OK
Time taken: 0.008 seconds
hive> insert overwrite table transactions_count
  > select a.custid, a.fname, count(*) from CUSTOMER a join TRANSACTIONS 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 Hive 1.X releases.
Query ID = acadgild_20180624175910_961db93c-ac9e-4f3b-a3dc-f835d8de04d3
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]
2018-06-24 17:59:21 Starting to launch local task to process map join; maximum memory = 518979584
2018-06-24 17:59:24 Dump the side-table for tag: 0 with group count: 8 into file: file:/tmp/acadgild/a10d8e47-ac32-4eee-a3af-06de94a6e42b/hive_2018-06-24_17-59-10_296_7001499669338651690-1/-local-10003/HashTable-Stage-2/MapJoin-mapfile10--.hashtable
2018-06-24 17:59:24 Uploaded 1 File to: file:/tmp/acadgild/a10d8e47-ac32-4eee-a3af-06de94a6e42b/hive_2018-06-24_17-59-10_296_7001499669338651690-1/-local-10003/HashTable-Stage-2/MapJoin-mapfile10--.hashtable (469 bytes)
2018-06-24 17:59:24 End of local task; Time Taken: 3.245 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>
Starting Job = job_1529836951776_0008, Tracking URL = http://localhost:8088/proxy/application_1529836951776_0008/
Kill Command = /home/acadgild/install/hadoop/hadoop-2.6.5/bin/hadoop job -kill job_1529836951776_0008
acadgild@localhost:~
```

Table populated:

```
MapReduce Total cumulative CPU time: 6 seconds 100 msec
Ended Job = job_1529836951776_0008
Loading data to table acadgild.transactions_count
MapReduce Jobs Launched:
Stage-Stage-2: Map: 1 Reduce: 1 Cumulative CPU: 6.1 sec HDFS Read: 13871 HDFS Write: 177 SUCCESS
Total MapReduce CPU Time Spent: 6 seconds 100 msec
OK
Time taken: 50.229 seconds
hive> select * from transactions_count;
OK
101 Amitabh 2
102 Sharukh 1
104 Anubhav 1
105 Pawan 1
106 Aamir 1
107 Salman 1
108 Ranbir 1
Time taken: 0.271 seconds, Fetched: 7 row(s)
hive>
```

Creating Hbase table :



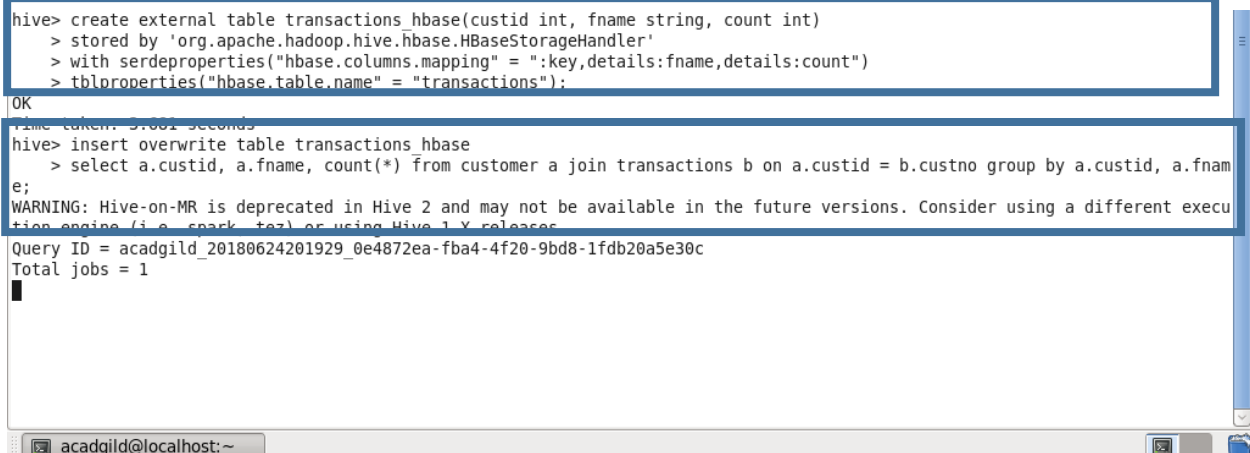
```
Applications Places System gild@localhost:~
Terminal Use the command line
acacgild@localhost:~
[acacgild@localhost ~]$ hbase shell
2018-06-24 20:00:06,581 WARN [main] util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using b
uiltin-java classes where applicable
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/acacgild/install/hbase/hbase-1.2.6/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/Static
LoggerBinder.class]
SLF4J: Found binding in [jar:file:/home/acacgild/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.slf4j.impl.Log4jLoggerFactory]
HBase Shell; enter 'help<RETURN>' for list of supported commands.
Type "exit<RETURN>" to leave the HBase Shell
Version 1.2.6, rUnknown, Mon May 29 02:25:32 CDT 2017

hbase(main):001:0> create 'transactions', 'details'
0 row(s) in 1.7700 seconds

=> Hbase::Table - transactions
hbase(main):002:0> █
```

Creating hive table which will be integrated with hbase table

Populating integrated table with above query:



```
hive> create external table transactions hbase(custid int, fname string, count int)
> stored by 'org.apache.hadoop.hive.hbase.HBaseStorageHandler'
> with serdeproperties("hbase.columns.mapping" = ":key,details:fname,details:count")
> tblproperties("hbase.table.name" = "transactions");
OK
Time taken: 0.001 seconds
hive> insert overwrite table transactions hbase
> select a.custid, a.fname, count(*) from customer a join transactions 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 execu
tion engine (i.e. spark, tez) or using Hive 1.X releases.
Query ID = acacgild_20180624201929_0e4872ea-fba4-4f20-9bd8-1fdb20a5e30c
Total jobs = 1
█
```

Output:

Both Hive and Hbase table populated with same data

```
-----
Total MapReduce CPU Time Spent: 8 seconds 420 msec
OK
Time taken: 72.666 seconds
hive> select * from transactions_hbase;
OK
101    Amitabh 2
102    Sharukh 1
104    Anubahv 1
105    Pawan   1
106    Aamir   1
107    Salman  1
108    Ranbir  1
Time taken: 0.629 seconds, Fetched: 7 row(s)
hive> █

acadgild@localhost:~
```



```
Applications  Places  System  Sun Jun 24, 8:22 PM  Acadgild
acadgild@localhost:~
File Edit View Search Terminal Tabs Help

acadgild@localhost:~  acadgild@localhost:~

[acadgild@localhost ~]$ hbase shell
2018-06-24 20:00:06,581 WARN [main] util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using b
ultin-java classes where applicable
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/home/acadgild/install/hbase/hbase-1.2.6/lib/slf4j-log4j12-1.7.5.jar!/org/slf4j/impl/Static
LoggerBinder.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.slf4j.impl.Log4jLoggerFactory]
HBase Shell; enter 'help<RETURN>' for list of supported commands.
Type "exit<RETURN>" to leave the HBase Shell
Version 1.2.6, rUnknown, Mon May 29 02:25:32 CDT 2017

hbase(main):001:0> create 'transactions', 'details'
0 row(s) in 1.7700 seconds

=> Hbase::Table - transactions
hbase(main):002:0> scan 'transactions'
ROW                                COLUMN+CELL
101                                column=details:count, timestamp=1529851840100, value=2
101                                column=details:fname, timestamp=1529851840100, value=Amitabh
102                                column=details:count, timestamp=1529851840100, value=1
102                                column=details:fname, timestamp=1529851840100, value=Sharukh
104                                column=details:count, timestamp=1529851840100, value=1
104                                column=details:fname, timestamp=1529851840100, value=Anubahv
105                                column=details:count, timestamp=1529851840100, value=1
105                                column=details:fname, timestamp=1529851840100, value=Pawan
106                                column=details:count, timestamp=1529851840100, value=1
106                                column=details:fname, timestamp=1529851840100, value=Aamir
107                                column=details:count, timestamp=1529851840100, value=1
107                                column=details:fname, timestamp=1529851840100, value=Salman
108                                column=details:count, timestamp=1529851840100, value=1
108                                column=details:fname, timestamp=1529851840100, value=Ranbir
7 row(s) in 0.4160 seconds

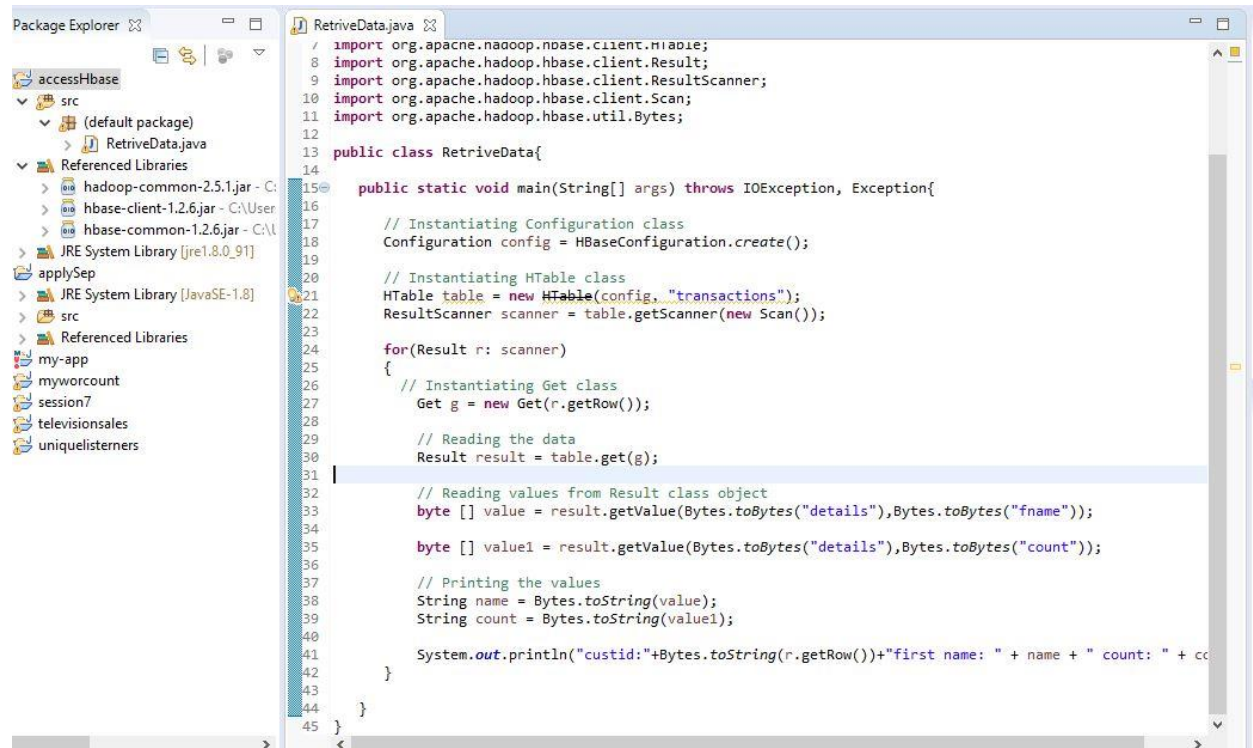
hbase(main):003:0> █

acadgild@localhost:~
```

Hbase java api code to access and scan the transactions table from java level:

Using eclipse to create jar file, added necessary jars from \$HBASE_HOME/lib/

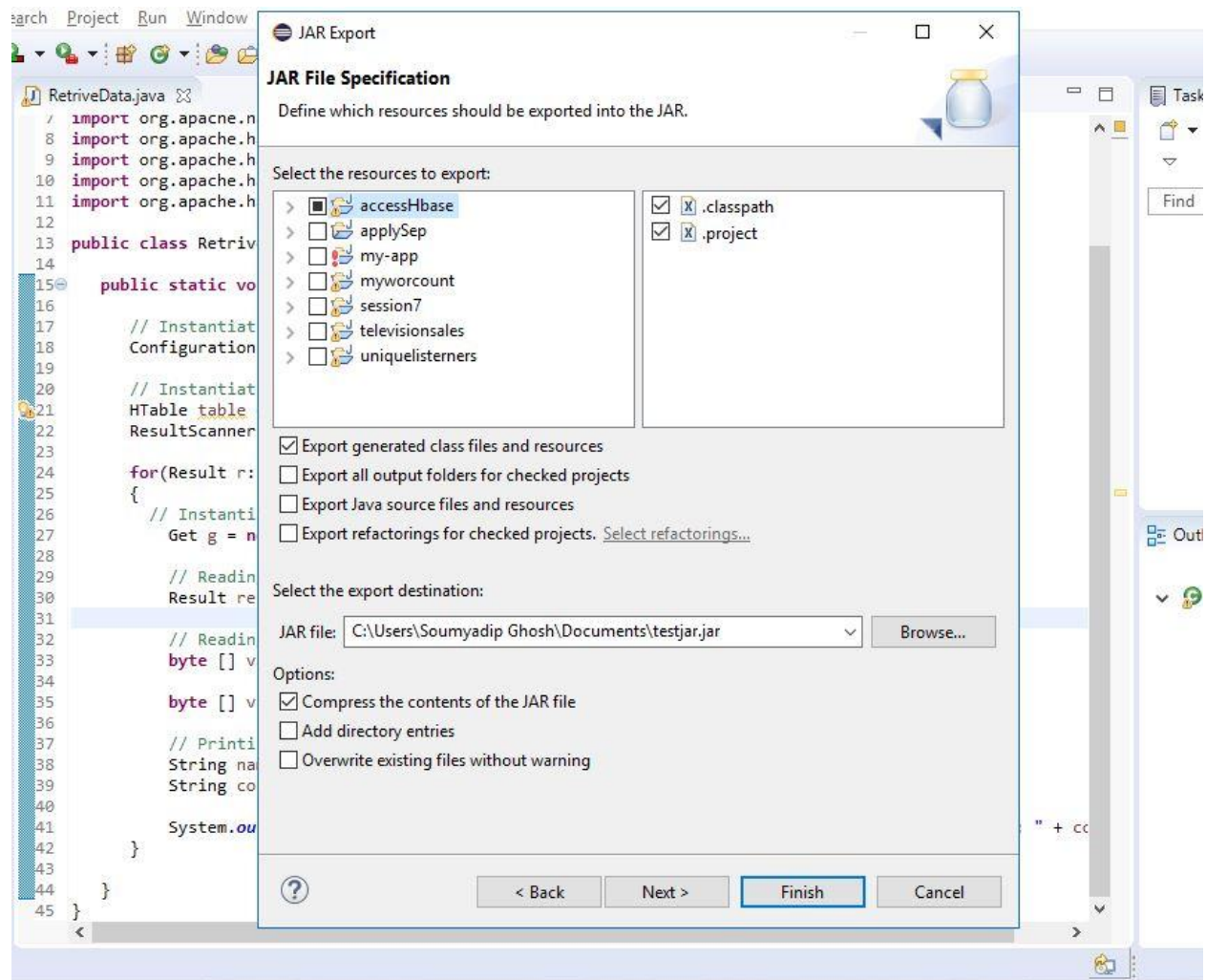
Code and added jar files:



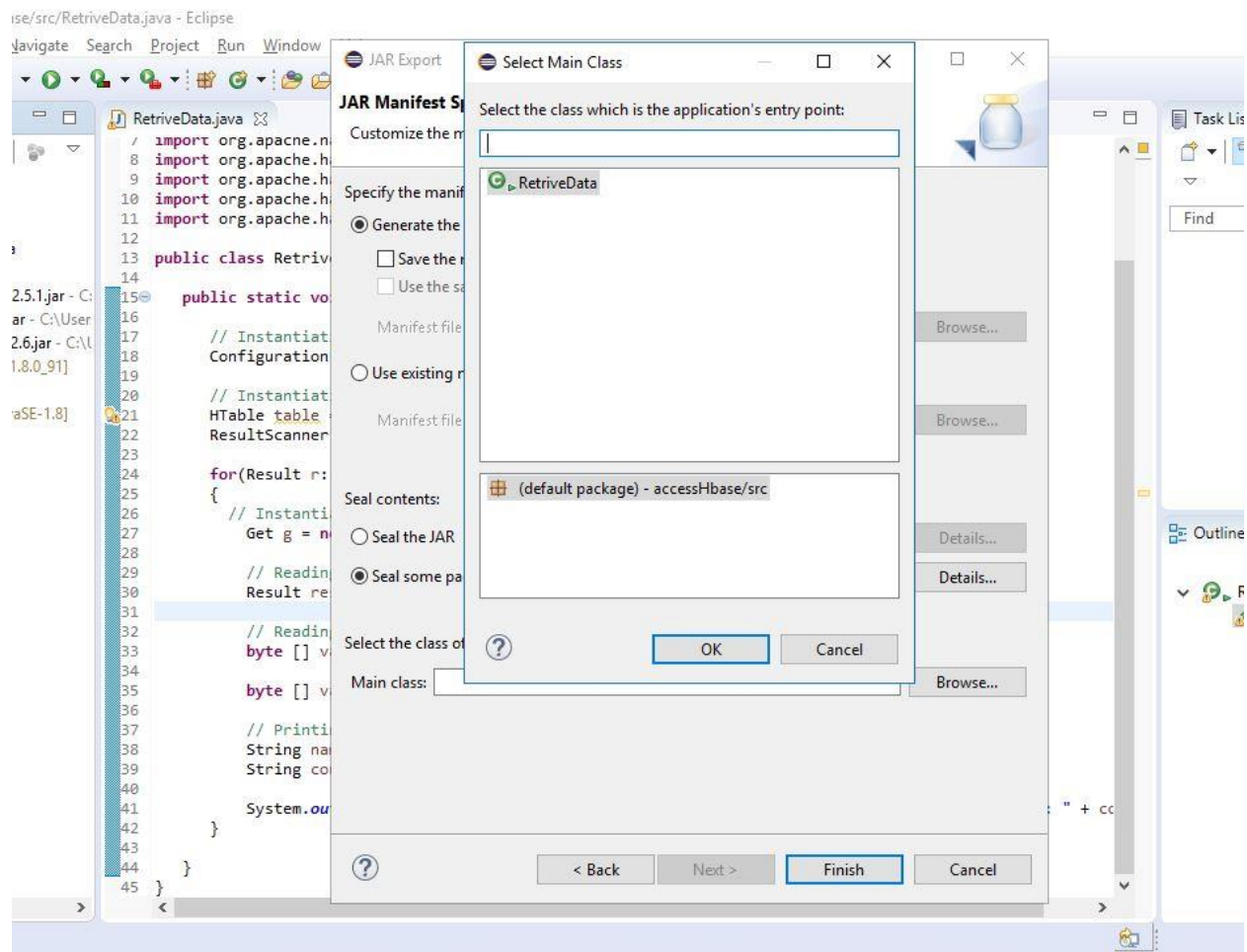
```
1  import org.apache.hadoop.hbase.client.HTable;
2  import org.apache.hadoop.hbase.client.Result;
3  import org.apache.hadoop.hbase.client.ResultScanner;
4  import org.apache.hadoop.hbase.client.Scan;
5  import org.apache.hadoop.hbase.util.Bytes;
6
7  public class RetriveData{
8
9      public static void main(String[] args) throws IOException, Exception{
10
11          // Instantiating Configuration class
12          Configuration config = HBaseConfiguration.create();
13
14          // Instantiating HTable class
15          HTable table = new HTable(config, "transactions");
16          ResultScanner scanner = table.getScanner(new Scan());
17
18          for(Result r: scanner)
19          {
20              // Instantiating Get class
21              Get g = new Get(r.getRow());
22
23              // Reading the data
24              Result result = table.get(g);
25
26              // Reading values from Result class object
27              byte [] value = result.getValue(Bytes.toBytes("details"),Bytes.toBytes("fname"));
28
29              byte [] value1 = result.getValue(Bytes.toBytes("details"),Bytes.toBytes("count"));
30
31              // Printing the values
32              String name = Bytes.toString(value);
33              String count = Bytes.toString(value1);
34
35              System.out.println("custid:"+Bytes.toString(r.getRow())+"first name: " + name + " count: " + count);
36          }
37      }
38  }
```

Exporting to create jar file:

Given name testjar.jar

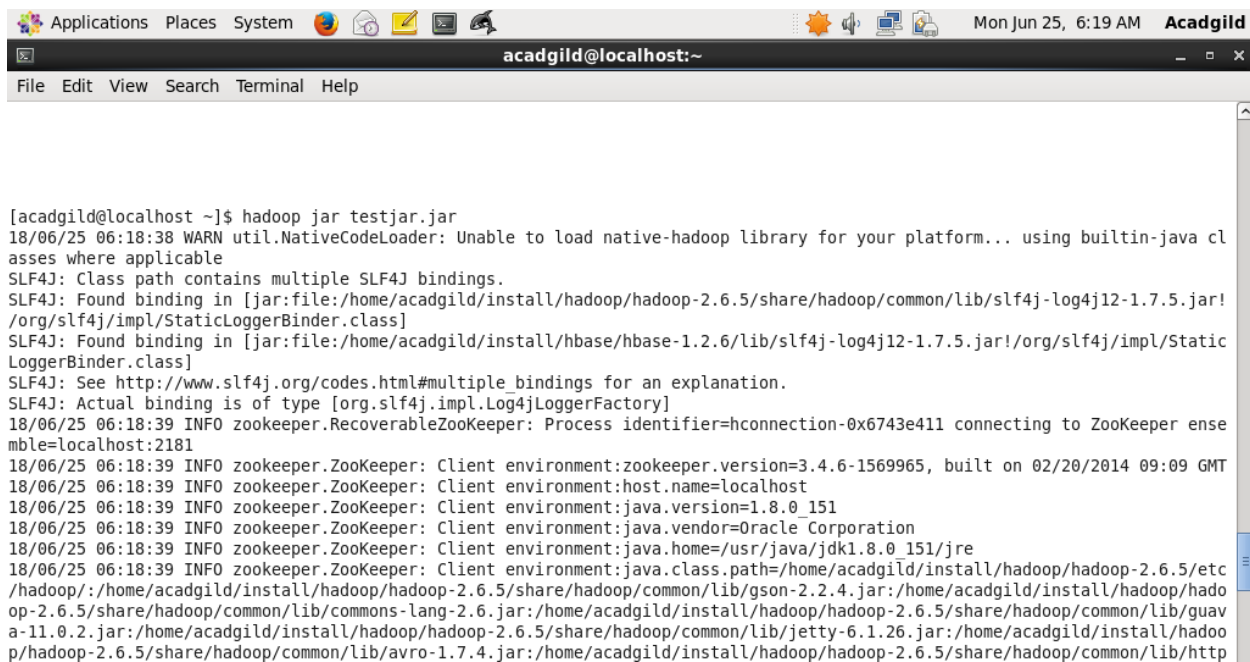


Selecting main class :



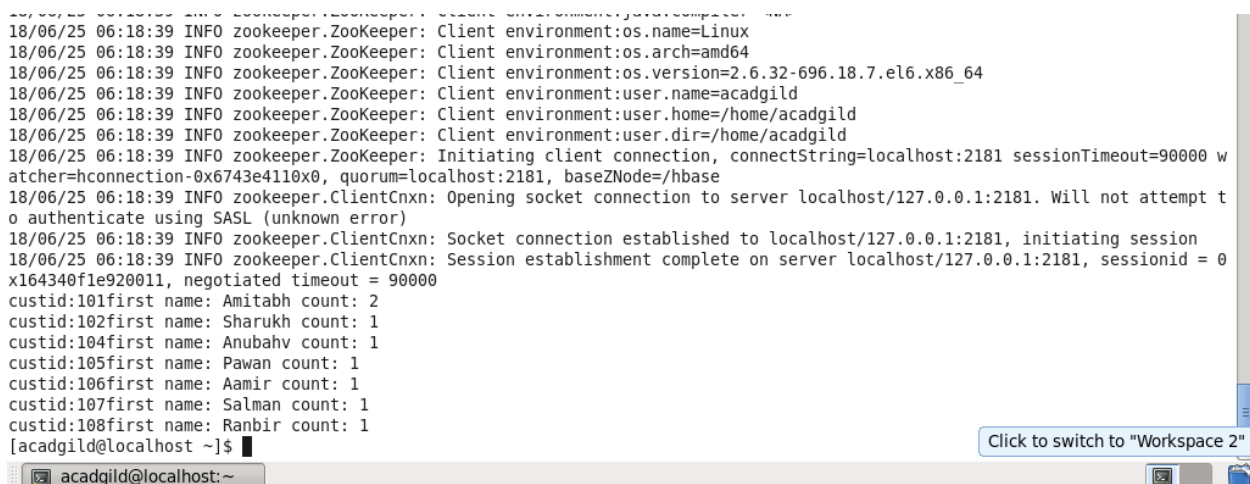
Output:

Running jar by command `hadoop jar <file-name>`

A terminal window titled 'acadgild@localhost:~' with a menu bar (File, Edit, View, Search, Terminal, Help). The output shows the execution of 'hadoop jar testjar.jar'. It includes warnings about native code loading, SLF4J binding information, and a series of INFO logs from the ZooKeeper client environment, detailing version, host, and classpath information.

```
[acadgild@localhost ~]$ hadoop jar testjar.jar
18/06/25 06:18:38 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
SLF4J: Class path contains multiple SLF4J bindings.
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: Found binding in [jar:file:/home/acadgild/install/hbase/hbase-1.2.6/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.slf4j.impl.Log4jLoggerFactory]
18/06/25 06:18:39 INFO zookeeper.RecoverableZooKeeper: Process identifier=hconnection-0x6743e411 connecting to ZooKeeper ensemble=localhost:2181
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:zookeeper.version=3.4.6-1569965, built on 02/20/2014 09:09 GMT
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:host.name=localhost
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:java.version=1.8.0_151
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:java.vendor=Oracle Corporation
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:java.home=/usr/java/jdk1.8.0_151/jre
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:java.class.path=/home/acadgild/install/hadoop/hadoop-2.6.5/etc/hadoop/:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/gson-2.2.4.jar:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/commons-lang-2.6.jar:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/guava-11.0.2.jar:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/jetty-6.1.26.jar:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/avro-1.7.4.jar:/home/acadgild/install/hadoop/hadoop-2.6.5/share/hadoop/common/lib/http
```

As you can see the data from hbase table 'transactions' has been read from java level

A terminal window titled 'acadgild@localhost:~' showing the continuation of the Hadoop command output. It displays INFO logs from the ZooKeeper client environment, including details about the session establishment, connection to the ZooKeeper ensemble, and the list of transactions read from the 'transactions' table.

```
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:java.vendor=Oracle Corporation
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:os.name=Linux
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:os.arch=amd64
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:os.version=2.6.32-696.18.7.el6.x86_64
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:user.name=acadgild
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:user.home=/home/acadgild
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Client environment:user.dir=/home/acadgild
18/06/25 06:18:39 INFO zookeeper.ZooKeeper: Initiating client connection, connectString=localhost:2181 sessionTimeout=90000 watcher=hconnection-0x6743e4110x0, quorum=localhost:2181, baseZNode=/hbase
18/06/25 06:18:39 INFO zookeeper.ClientCnxn: Opening socket connection to server localhost/127.0.0.1:2181. Will not attempt to authenticate using SASL (unknown error)
18/06/25 06:18:39 INFO zookeeper.ClientCnxn: Socket connection established to localhost/127.0.0.1:2181, initiating session
18/06/25 06:18:39 INFO zookeeper.ClientCnxn: Session establishment complete on server localhost/127.0.0.1:2181, sessionId = 0x164340f1e920011, negotiated timeout = 90000
custid:101first name: Amitabh count: 2
custid:102first name: Sharukh count: 1
custid:104first name: Anubhav count: 1
custid:105first name: Pawan count: 1
custid:106first name: Aamir count: 1
custid:107first name: Salman count: 1
custid:108first name: Ranbir count: 1
[acadgild@localhost ~]$
```