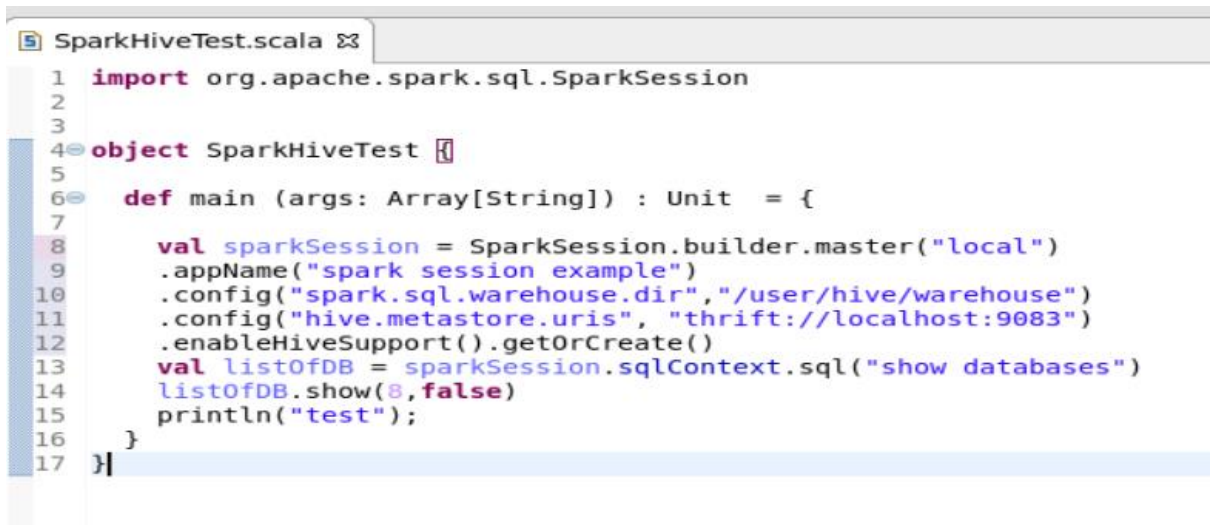


Assignment - 25.1

Task 1: As discussed in class integrate Spark Hive

Answer:

1. Open acadgild VM and start all daemons – Hadoop + Hbase using below commands.
 - Start-all.sh
 - Start-hbase.sh
2. Now open Eclipse in VM and create a project named “TestScalaProject” and import the source code “SparkHiveTest.scala”



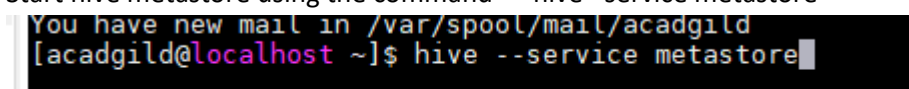
```
1 import org.apache.spark.sql.SparkSession
2
3
4 object SparkHiveTest {
5
6   def main (args: Array[String]) : Unit = {
7
8     val sparkSession = SparkSession.builder.master("local")
9       .appName("spark session example")
10      .config("spark.sql.warehouse.dir", "/user/hive/warehouse")
11      .config("hive.metastore.uris", "thrift://localhost:9083")
12      .enableHiveSupport().getOrCreate()
13     val listOfDB = sparkSession.sqlContext.sql("show databases")
14     listOfDB.show(8, false)
15     println("test");
16   }
17 }
```

3. Resolve all the compilation errors from external JARs which are required and then execute the class as a Scala Application.
4. Copy hive-site.xml file from \$HIVE_HOME/conf to \$SPARK_HOME/conf.
5. Add the following property to hive-site.xml on Spark side:

```
<property>
  <name>hive.metastore.uris</name>
  <value>thrift://localhost:9083</value>
  <description>password for connecting to mysql server</description>
</property>
```

This property helps to create the connection between Spark and Hive.

6. Start hive metastore using the command – “hive --service metastore”

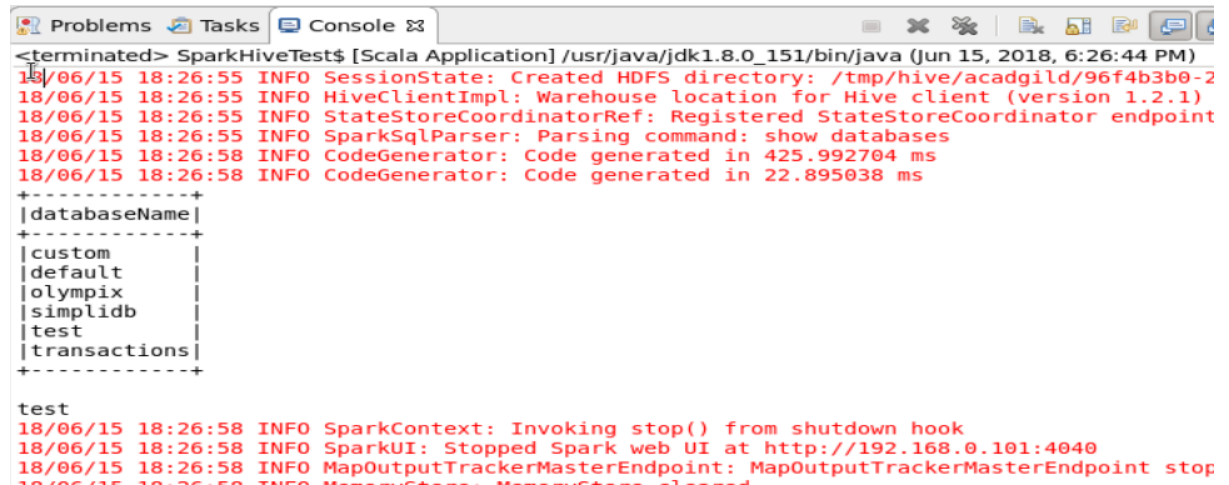


```
You have new mail in /var/spool/mail/acadgild
[acadgild@localhost ~]$ hive --service metastore
```

term by subscribing to the professional edition here: <https://mobaxterm.mobatek.net>

Make sure mysql service is running or else execute the command “sudo service mysql start”.

7. Run the class in eclipse as “Scala Application”.



```
<terminated> SparkHiveTest$ [Scala Application] /usr/java/jdk1.8.0_151/bin/java (Jun 15, 2018, 6:26:44 PM)
18/06/15 18:26:55 INFO SessionState: Created HDFS directory: /tmp/hive/acadgild/96f4b3b0-2
18/06/15 18:26:55 INFO HiveClientImpl: Warehouse location for Hive client (version 1.2.1)
18/06/15 18:26:55 INFO StateStoreCoordinatorRef: Registered StateStoreCoordinator endpoint
18/06/15 18:26:55 INFO SparkSqlParser: Parsing command: show databases
18/06/15 18:26:58 INFO CodeGenerator: Code generated in 425.992704 ms
18/06/15 18:26:58 INFO CodeGenerator: Code generated in 22.895038 ms

+-----+
|databaseName|
+-----+
|custom      |
|default     |
|olympix     |
|simplidb    |
|test        |
|transactions|
+-----+

test
18/06/15 18:26:58 INFO SparkContext: Invoking stop() from shutdown hook
18/06/15 18:26:58 INFO SparkUI: Stopped Spark web UI at http://192.168.0.101:4040
18/06/15 18:26:58 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stop
18/06/15 18:26:58 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stop
```

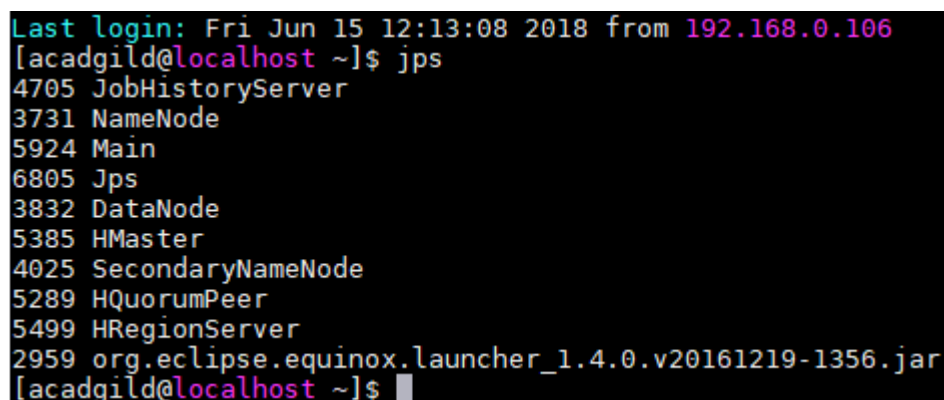
Here we could see the list of database in Hive using Apache Spark – Hadoop Integration.

Task 2: As discussed in class integrate Spark Hbase

Answer: Follow the below steps to proceed for Spark Hbase integration

8. Open acadgild VM and start all daemons – Hadoop + Hbase using below commands.
 - Start-all.sh
 - Start-hbase.sh
 - Mr-jobhistory-daemon.sh start history server

Check all daemons are running as in below screenshot.



```
Last login: Fri Jun 15 12:13:08 2018 from 192.168.0.106
[acadgild@localhost ~]$ jps
4705 JobHistoryServer
3731 NameNode
5924 Main
6805 Jps
3832 DataNode
5385 HMaster
4025 SecondaryNameNode
5289 HQuorumPeer
5499 HRegionServer
2959 org.eclipse.equinox.launcher_1.4.0.v20161219-1356.jar
[acadgild@localhost ~]$
```

9. Open HBase shell and execute the list command.

```

hbase(main):001:0> list
TABLE
SparkHBasesTable
txn_count
2 row(s) in 0.4380 seconds

=> ["SparkHBasesTable", "txn_count"]
hbase(main):002:0>

```

term by subscribing to the professional edition here: <https://mobaxterm.m>

10. Disable and drop the SparkHBasesTable.

```

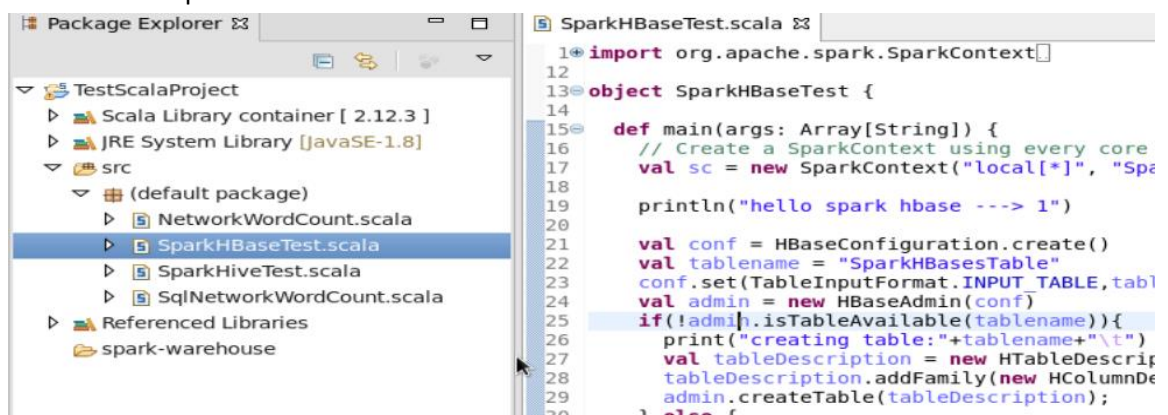
hbase(main):003:0> disable 'SparkHBasesTable'
0 row(s) in 2.4430 seconds

hbase(main):004:0> drop 'SparkHBasesTable'
0 row(s) in 1.2890 seconds

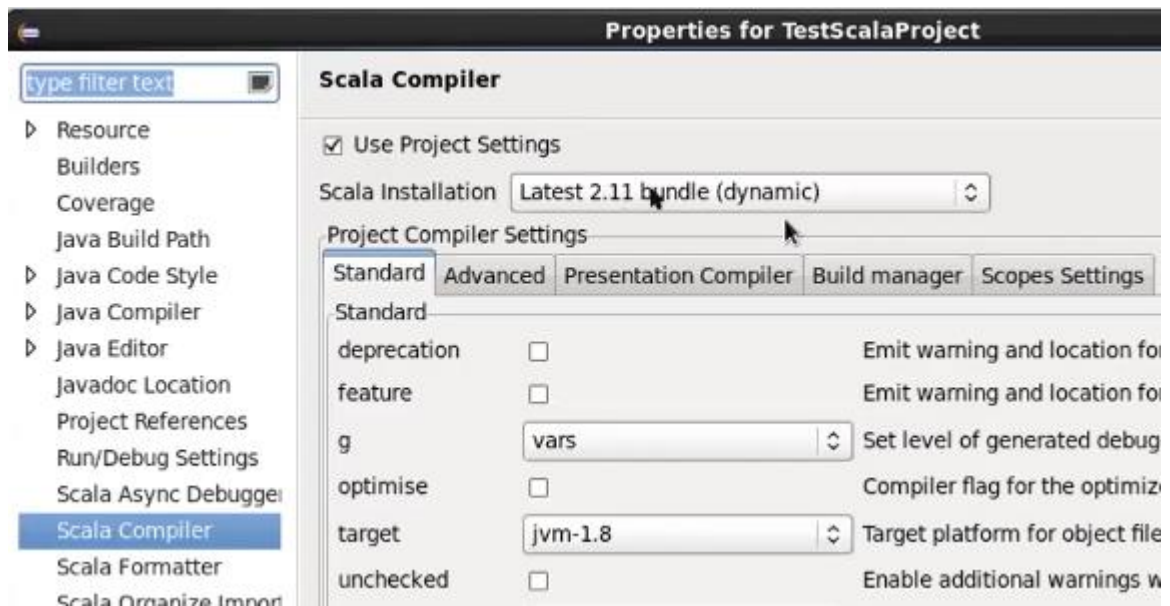
hbase(main):005:0>

```

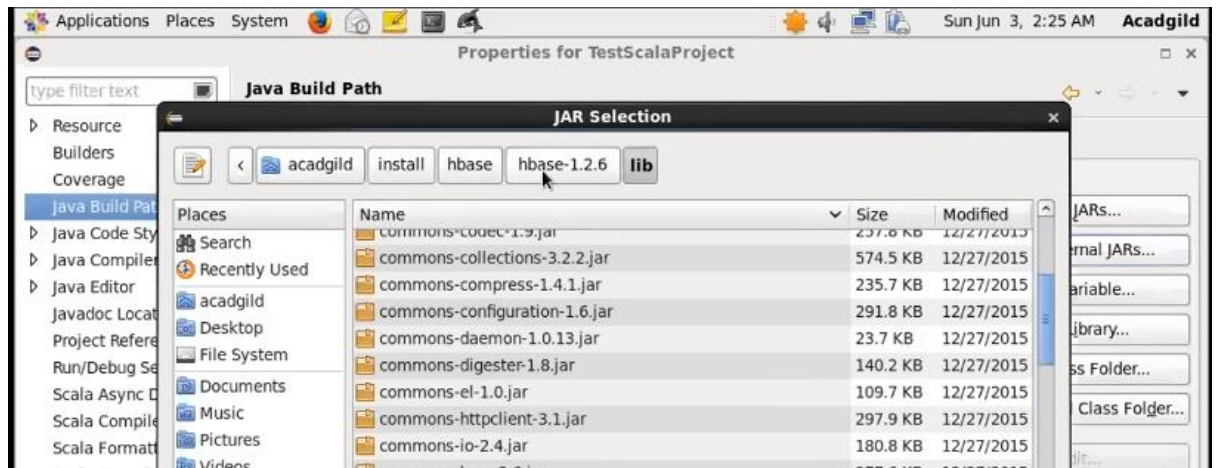
11. Now open Eclipse in VM and create a project named "TestScalaProject" and import the source code "SparkHBaseTest.scala"



12. Resolve all the error by adding on the external JARs from Spark folder and make sure the Scala compiler version is "Latest 2.11 bundle (dynamic)".



JAR details from this path:



13. Now run the particular class "SparkHBaseTest.scala" as Scala application.

```
*SparkHBaseTest.scala
15 def main(args: Array[String]) {
16   // Create a SparkContext using every core of the local machine, named RatingsCounter
17   val sc = new SparkContext("local[*]", "SparkHBaseTest")
18
19   println("hello spark hbase ---> 1")
20
21   val conf = HBaseConfiguration.create()
22   val tablename = "SparkHBasesTable"
23   conf.set(TableInputFormat.INPUT_TABLE, tablename)
24   val admin = new HBaseAdmin(conf)
25   if(!admin.isTableAvailable(tablename)){
26     print("creating table:"+tablename+"\n")
27     val tableDescription = new HTableDescriptor(tablename)
28     tableDescription.addFamily(new HColumnDescriptor("cf".getBytes()));
29     admin.createTable(tableDescription);
30   } else {
31     print("table already exists")
32   }
33
34   val table = new HTable(conf,tablename);
35   for(x <- 1 to 10){
36     var p = new Put(new String("row" + x).getBytes());
37     p.add("cf".getBytes(),"column1".getBytes(),new String("value" + x).getBytes());
38     table.put(p);
39     print("Data Entered In Table")
40   }
41   val hBaseRDD = sc.newAPIHadoopRDD(conf, classOf[TableInputFormat],
42     classOf[ImmutableBytesWritable],classOf[Result])
43   print("RecordCount-->"+hBaseRDD.count())
44   sc.stop()
}
```


14. Above code – creates a new table and enters input values to the table in HBase using Spark API. Below are the output screens for the same.

```
18/06/15 15:36:12 INFO BlockManager: Initialized BlockManager: BlockManagerId(driver, 192.168.0.101, 43519, None)
hello spark hbase ---> 1
18/06/15 15:36:12 INFO RecoverableZooKeeper: Process identifier=hconnection-0x55e7a35c connecting to ZooKeeper ensemble
18/06/15 15:36:12 INFO ZooKeeper: Client environment:zookeeper.version=3.4.6-1569965, built on 02/20/2014 09:09 GMT
18/06/15 15:36:12 INFO ZooKeeper: Client environment:host.name=localhost
18/06/15 15:36:12 INFO ZooKeeper: Client environment:java.version=1.8.0_151
18/06/15 15:36:12 INFO ZooKeeper: Client environment:java.vendor=Oracle Corporation
18/06/15 15:36:12 INFO ZooKeeper: Client environment:java.home=/usr/java/jdk1.8.0_151/jre
18/06/15 15:36:12 INFO ZooKeeper: Client environment:java.class.path=/home/acadgild/.p2/pool/plugins/org.scala-ide.
18/06/15 15:36:12 INFO ZooKeeper: Client environment:java.library.path=/usr/java/packages/lib/amd64:/usr/lib64:/lib64:/lib
18/06/15 15:36:12 INFO ZooKeeper: Client environment:java.io.tmpdir=/tmp
18/06/15 15:36:12 INFO ZooKeeper: Client environment:java.compiler=<NA>
18/06/15 15:36:12 INFO ZooKeeper: Client environment:os.name=Linux
18/06/15 15:36:12 INFO ZooKeeper: Client environment:os.arch=amd64
18/06/15 15:36:12 INFO ZooKeeper: Client environment:os.version=2.6.32-696.23.1.el6.x86_64
18/06/15 15:36:12 INFO ZooKeeper: Client environment:user.name=acadgild
18/06/15 15:36:12 INFO ZooKeeper: Client environment:user.home=/home/acadgild
18/06/15 15:36:12 INFO ZooKeeper: Client environment:user.dir=/home/acadgild/eclipse-workspace/TestScalaProject
18/06/15 15:36:12 INFO ZooKeeper: Initiating client connection, connectString=localhost:2181 sessionTimeout=90000 v
18/06/15 15:36:13 INFO ClientCnxn: Opening socket connection to server localhost/0:0:0:0:0:0:1:2181. Will not att
18/06/15 15:36:13 INFO ClientCnxn: Socket connection established to localhost/0:0:0:0:0:0:1:2181, initiating sess
18/06/15 15:36:13 INFO ClientCnxn: Session establishment complete on server localhost/0:0:0:0:0:0:1:2181, session
creating table:SparkHBasesTable 18/06/15 15:36:16 INFO HBaseAdmin: Created SparkHBasesTable
Data Entered In TableData Entered In TableData Entered In TableData Entered In TableData Entered In TableData Enter
18/06/15 15:36:17 INFO MemoryStore: Block broadcast 0 piece0 stored as bytes in memory (estimated size 28.4 KB, fre
18/06/15 15:36:17 INFO BlockManagerInfo: Added broadcast 0 piece0 in memory on 192.168.0.101:43519 (size: 28.4 KB,
18/06/15 15:36:17 INFO SparkContext: Created broadcast 0 from newAPIHadoopRDD at SparkHBaseTest.scala:41
18/06/15 15:36:17 INFO RecoverableZooKeeper: Process identifier=hconnection-0x2016f509 connecting to ZooKeeper ense
18/06/15 15:36:19 INFO TaskSchedulerImpl: Removed TaskSet 0.0, whose tasks have all completed, from pool
18/06/15 15:36:19 INFO DAGScheduler: ResultStage 0 (count at SparkHBaseTest.scala:42) finished in 0.614 s
18/06/15 15:36:19 INFO DAGScheduler: Job 0 finished: count at SparkHBaseTest.scala:42, took 0.928215 s
RecordCount-->1018/06/15 15:36:19 INFO SparkUI: Stopped Spark web UI at http://192.168.0.101:4040
18/06/15 15:36:19 INFO MapOutputTrackerMasterEndpoint: MapOutputTrackerMasterEndpoint stopped!
18/06/15 15:36:19 INFO MemoryStore: MemoryStore cleared
18/06/15 15:36:19 INFO BlockManager: BlockManager stopped
18/06/15 15:36:19 INFO BlockManagerMaster: BlockManagerMaster stopped
18/06/15 15:36:19 INFO OutputCommitCoordinator$OutputCommitCoordinatorEndpoint: OutputCommitCoordinator stoppe
18/06/15 15:36:19 INFO SparkContext: Successfully stopped SparkContext
18/06/15 15:36:19 INFO ShutdownHookManager: Shutdown hook called
18/06/15 15:36:19 INFO ShutdownHookManager: Deleting directory /tmp/spark-d3a6b10f-ce06-4460-94b6-86e45d80fcc1
```

Here we have got the output in the eclipse console.

Now check the same in the terminal using hbase shell.

```
=> ["SparkHBasesTable", "Cxn_Count"]
hbase(main):002:0> scan 'SparkHBasesTable'
ROW COLUMN+CELL
row1 column=cf:column1, timestamp=1529057176601, value=value1
row10 column=cf:column1, timestamp=1529057176659, value=value10
row2 column=cf:column1, timestamp=1529057176624, value=value2
row3 column=cf:column1, timestamp=1529057176628, value=value3
row4 column=cf:column1, timestamp=1529057176637, value=value4
row5 column=cf:column1, timestamp=1529057176641, value=value5
row6 column=cf:column1, timestamp=1529057176645, value=value6
row7 column=cf:column1, timestamp=1529057176648, value=value7
row8 column=cf:column1, timestamp=1529057176651, value=value8
row9 column=cf:column1, timestamp=1529057176655, value=value9
10 row(s) in 0.2680 seconds
hbase(main):003:0> █
```