CASE STUDY 3

Sensor Data Analysis

Assignment 22.3: Case Study Sensor Data Analysis

Use Case Description :

The Use Case consist of temperatures collected every minute, from 20 top buildings all over the world.

HVAC (heating, ventilating/ventilation, and **air conditioning**) is the technology of indoor and vehicular environmental comfort. Its goal is to provide thermal comfort and acceptable indoor air quality. Through the HVAC sensors, we will get the temperature of the buildings

The required two datasets are :

- **building.csv :** contains the details of the top 20 buildings all over the world
- **HVAC.csv** : contains the target temperature and the actual temperature along with the building Id.

We will be performing analysis on the HVAC datasets to obtain the temperature changes in the building. We are performing this analysis using Spark SQL.

Dataset:

- **Building.csv – BuildingID, BuildingMgr, BuildingAge, HVACproduct,Country**
- **HVAC.csv – Date, Time, TargetTemp, ActualTemp, System, SystemAge, BuildingID**

Initial Execution:

```
[acadgild@localhost ~]$ jps
2964 Jps
[acadgild@localhost ~]$ sudo service sshd start
[sudo] password for acadgild:
[acadgild@localhost ~]$ start-all.sh
This script is Deprecated. Instead use start-dfs.sh and start-yarn.sh
18/09/07 21:05:23 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your
platform... using builtin-java classes where applicable
Starting namenodes on [localhost]
localhost: starting namenode, logging to
```

/
home/acadgild/install/hadoop/hadoop-2.6.5/logs/hadoop-acadgild-namenode-localhost.localdomain.out
localhost: starting datanode, logging to
/home/acadgild/install/hadoop/hadoop-2.6.5/logs/hadoop-acadgild-datanode-localhost.localdomain.out
Starting secondary namenodes [0.0.0.0]
0.0.0.0: starting secondarynamenode, logging to
/home/acadgild/install/hadoop/hadoop-2.6.5/logs/hadoop-acadgild-secondarynamenode-localhost.localdomain.out
18/09/07 21:05:54 WARN util.NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
starting yarn daemons
starting resourcemanager, logging to
/home/acadgild/install/hadoop/hadoop-2.6.5/logs/yarn-acadgild-resourcemanager-localhost.localdomain.out
localhost: starting nodemanager, logging to
/home/acadgild/install/hadoop/hadoop-2.6.5/logs/yarn-acadgild-nodemanager-localhost.localdomain.out
[acadgild@localhost ~]$ jps
3680 Jps
3545 ResourceManager
3386 SecondaryNameNode
3115 NameNode
3212 DataNode
3646 NodeManager

[acadgild@localhost ~]$ spark-shell
ording to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.
Sat Sep 08 04:16:48 IST 2018 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.
Sat Sep 08 04:16:52 IST 2018 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.
Sat Sep 08 04:16:52 IST 2018 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with

existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.
Sat Sep 08 04:16:53 IST 2018 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.
Sat Sep 08 04:16:53 IST 2018 WARN: Establishing SSL connection without server's identity verification is not recommended. According to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You need either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificate verification.
18/09/08 04:17:02 ERROR DataNucleus.Datastore: An exception was thrown while adding/validating class(es) : Specified key was too long; max key length is 3072 bytes
com.mysql.jdbc.exceptions.jdbc4.MySQLSyntaxErrorException: Specified key was too long; max key length is 3072 bytes
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.java:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at com.mysql.jdbc.Util.handleNewInstance(Util.java:425)
	at com.mysql.jdbc.Util.getInstance(Util.java:408)
	at com.mysql.jdbc.SQLError.createSQLException(SQLError.java:944)
	at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:3976)
	at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:3912)
	at com.mysql.jdbc.MysqlIO.sendCommand(MysqlIO.java:2530)
	at com.mysql.jdbc.MysqlIO.sqlQueryDirect(MysqlIO.java:2683)
	at com.mysql.jdbc.ConnectionImpl.execSQL(ConnectionImpl.java:2482)
	at com.mysql.jdbc.ConnectionImpl.execSQL(ConnectionImpl.java:2440)
	at com.mysql.jdbc.StatementImpl.executeInternal(StatementImpl.java:845)
	at com.mysql.jdbc.StatementImpl.execute(StatementImpl.java:745)
	at com.jolbox.bonecp.StatementHandle.execute(StatementHandle.java:254)
	at org.datanucleus.store.rdbms.table.AbstractTable.executeDdlStatement(AbstractTable.java:760)
	at org.datanucleus.store.rdbms.table.TableImpl.createIndices(TableImpl.java:648)
	at org.datanucleus.store.rdbms.table.TableImpl.createConstraints(TableImpl.java:422)
	at org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.performTablesValidation(RDBMSStoreManager.java:3459)
	at org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.addClassTablesAndValidate(RDBMSStoreManager.java:3190)
	at

org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.run(RDBMSStoreManager.java:284
1)
	at
org.datanucleus.store.rdbms.AbstractSchemaTransaction.execute(AbstractSchemaTransaction.java:1
22)
	at
org.datanucleus.store.rdbms.RDBMSStoreManager.addClasses(RDBMSStoreManager.java:1605)
	at org.datanucleus.store.AbstractStoreManager.addClass(AbstractStoreManager.java:954)
	at
org.datanucleus.store.rdbms.RDBMSStoreManager.getDatastoreClass(RDBMSStoreManager.java:6
79)
	at
org.datanucleus.store.rdbms.query.RDBMSQueryUtils.getStatementForCandidates(RDBMSQueryUt
ils.java:408)
	at
org.datanucleus.store.rdbms.query.JDOQLQuery.compileQueryFull(JDOQLQuery.java:947)
	at org.datanucleus.store.rdbms.query.JDOQLQuery.compileInternal(JDOQLQuery.java:370)
	at org.datanucleus.store.query.Query.executeQuery(Query.java:1744)
	at org.datanucleus.store.query.Query.executeWithArray(Query.java:1672)
	at org.datanucleus.store.query.Query.execute(Query.java:1654)
	at org.datanucleus.api.jdo.JDOQuery.execute(JDOQuery.java:221)
	at
org.apache.hadoop.hive.metastore.MetaStoreDirectSql.ensureDbInit(MetaStoreDirectSql.java:185)
	at
org.apache.hadoop.hive.metastore.MetaStoreDirectSql.<init>(MetaStoreDirectSql.java:137)
	at org.apache.hadoop.hive.metastore.ObjectStore.initialize(ObjectStore.java:295)
	at org.apache.hadoop.hive.metastore.ObjectStore.setConf(ObjectStore.java:258)
	at org.apache.hadoop.util.ReflectionUtils.setConf(ReflectionUtils.java:73)
	at org.apache.hadoop.util.ReflectionUtils.newInstance(ReflectionUtils.java:133)
	at org.apache.hadoop.hive.metastore.RawStoreProxy.<init>(RawStoreProxy.java:57)
	at org.apache.hadoop.hive.metastore.RawStoreProxy.getProxy(RawStoreProxy.java:66)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.newRawStore(HiveMetaStore.java:
593)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.getMS(HiveMetaStore.java:571)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.createDefaultDB(HiveMetaStore.ja
va:620)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.init(HiveMetaStore.java:461)
	at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.<init>(RetryingHMSHandler.java:66)
	at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.getProxy(RetryingHMSHandler.java:72)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore.newRetryingHMSHandler(HiveMetaStore.java:57
62)
	at

```
org.apache.hadoop.hive.metastore.HiveMetaStoreClient.<init>(HiveMetaStoreClient.java:199)
	at
org.apache.hadoop.hive.ql.metadata.SessionHiveMetaStoreClient.<init>(SessionHiveMetaStoreClien
t.java:74)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1521)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.<init>(RetryingMetaStoreClient.java:86)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(RetryingMetaStoreClient.java:
132)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(RetryingMetaStoreClient.java:
104)
	at org.apache.hadoop.hive.ql.metadata.Hive.createMetaStoreClient(Hive.java:3005)
	at org.apache.hadoop.hive.ql.metadata.Hive.getMSC(Hive.java:3024)
	at org.apache.hadoop.hive.ql.metadata.Hive.getAllDatabases(Hive.java:1234)
	at org.apache.hadoop.hive.ql.metadata.Hive.reloadFunctions(Hive.java:174)
	at org.apache.hadoop.hive.ql.metadata.Hive.<clinit>(Hive.java:166)
	at org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:503)
	at org.apache.spark.sql.hive.client.HiveClientImpl.<init>(HiveClientImpl.scala:192)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at
org.apache.spark.sql.hive.client.IsolatedClientLoader.createClient(IsolatedClientLoader.scala:264)
	at org.apache.spark.sql.hive.HiveUtils$.newClientForMetadata(HiveUtils.scala:366)
	at org.apache.spark.sql.hive.HiveUtils$.newClientForMetadata(HiveUtils.scala:270)
	at org.apache.spark.sql.hive.HiveExternalCatalog.<init>(HiveExternalCatalog.scala:65)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at org.apache.spark.sql.internal.SharedState$.org$apache$spark$sql$internal$SharedState$
$reflect(SharedState.scala:166)
	at org.apache.spark.sql.internal.SharedState.<init>(SharedState.scala:86)
```

```
	at org.apache.spark.sql.SparkSession$$anonfun$sharedState$1.apply(SparkSession.scala:101)
	at org.apache.spark.sql.SparkSession$$anonfun$sharedState$1.apply(SparkSession.scala:101)
	at scala.Option.getOrElse(Option.scala:121)
	at org.apache.spark.sql.SparkSession.sharedState$lzycompute(SparkSession.scala:101)
	at org.apache.spark.sql.SparkSession.sharedState(SparkSession.scala:100)
	at org.apache.spark.sql.internal.SessionState.<init>(SessionState.scala:157)
	at org.apache.spark.sql.hive.HiveSessionState.<init>(HiveSessionState.scala:32)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at org.apache.spark.sql.SparkSession$.org$apache$spark$sql$SparkSession$
$reflect(SparkSession.scala:978)
	at org.apache.spark.sql.SparkSession.sessionState$lzycompute(SparkSession.scala:110)
	at org.apache.spark.sql.SparkSession.sessionState(SparkSession.scala:109)
	at org.apache.spark.sql.SparkSession$Builder$
$anonfun$getOrCreate$5.apply(SparkSession.scala:878)
	at org.apache.spark.sql.SparkSession$Builder$
$anonfun$getOrCreate$5.apply(SparkSession.scala:878)
	at scala.collection.mutable.HashMap$$anonfun$foreach$1.apply(HashMap.scala:99)
	at scala.collection.mutable.HashMap$$anonfun$foreach$1.apply(HashMap.scala:99)
	at scala.collection.mutable.HashTable$class.foreachEntry(HashTable.scala:230)
	at scala.collection.mutable.HashMap.foreachEntry(HashMap.scala:40)
	at scala.collection.mutable.HashMap.foreach(HashMap.scala:99)
	at org.apache.spark.sql.SparkSession$Builder.getOrCreate(SparkSession.scala:878)
	at org.apache.spark.repl.Main$.createSparkSession(Main.scala:95)
	at $line3.$read$$iw$$iw.<init>(<console>:15)
	at $line3.$read$$iw.<init>(<console>:42)
	at $line3.$read.<init>(<console>:44)
	at $line3.$read$.<init>(<console>:48)
	at $line3.$read$.<clinit>(<console>)
	at $line3.$eval$.$print$lzycompute(<console>:7)
	at $line3.$eval$.$print(<console>:6)
	at $line3.$eval.$print(<console>)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
	at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:498)
	at scala.tools.nsc.interpreter.IMain$ReadEvalPrint.call(IMain.scala:786)
	at scala.tools.nsc.interpreter.IMain$Request.loadAndRun(IMain.scala:1047)
	at scala.tools.nsc.interpreter.IMain$WrappedRequest$
$anonfun$loadAndRunReq$1.apply(IMain.scala:638)
	at scala.tools.nsc.interpreter.IMain$WrappedRequest$
$anonfun$loadAndRunReq$1.apply(IMain.scala:637)
	at scala.reflect.internal.util.ScalaClassLoader$class.asContext(ScalaClassLoader.scala:31)
```

```
	at
scala.reflect.internal.util.AbstractFileClassLoader.asContext(AbstractFileClassLoader.scala:19)
	at scala.tools.nsc.interpreter.IMain$WrappedRequest.loadAndRunReq(IMain.scala:637)
	at scala.tools.nsc.interpreter.IMain.interpret(IMain.scala:569)
	at scala.tools.nsc.interpreter.IMain.interpret(IMain.scala:565)
	at scala.tools.nsc.interpreter.ILoop.interpretStartingWith(ILoop.scala:807)
	at scala.tools.nsc.interpreter.ILoop.command(ILoop.scala:681)
	at scala.tools.nsc.interpreter.ILoop.processLine(ILoop.scala:395)
	at org.apache.spark.repl.SparkILoop$
$anonfun$initializeSpark$1.apply$mcV$sp(SparkILoop.scala:38)
	at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
	at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
	at scala.tools.nsc.interpreter.IMain.beQuietDuring(IMain.scala:214)
	at org.apache.spark.repl.SparkILoop.initializeSpark(SparkILoop.scala:37)
	at org.apache.spark.repl.SparkILoop.loadFiles(SparkILoop.scala:105)
	at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply$mcZ$sp(ILoop.scala:920)
	at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
	at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
	at
scala.reflect.internal.util.ScalaClassLoader$.savingContextLoader(ScalaClassLoader.scala:97)
	at scala.tools.nsc.interpreter.ILoop.process(ILoop.scala:909)
	at org.apache.spark.repl.Main$.doMain(Main.scala:68)
	at org.apache.spark.repl.Main$.main(Main.scala:51)
	at org.apache.spark.repl.Main.main(Main.scala)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
	at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:498)
	at org.apache.spark.deploy.SparkSubmit$.org$apache$spark$deploy$SparkSubmit$
$runMain(SparkSubmit.scala:738)
	at org.apache.spark.deploy.SparkSubmit$.doRunMain$1(SparkSubmit.scala:187)
	at org.apache.spark.deploy.SparkSubmit$.submit(SparkSubmit.scala:212)
	at org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:126)
	at org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala)

18/09/08 04:17:03 WARN DataNucleus.Query: Query for candidates of
org.apache.hadoop.hive.metastore.model.MPartitionColumnStatistics and subclasses resulted in no
possible candidates
Error(s) were found while auto-creating/validating the datastore for classes. The errors are printed in
the log, and are attached to this exception.
org.datanucleus.exceptions.NucleusDataStoreException: Error(s) were found while
auto-creating/validating the datastore for classes. The errors are printed in the log, and are attached to
this exception.
	at
org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.verifyErrors(RDBMSStoreManager.
java:3602)
	at
org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.addClassTablesAndValidate(RDBM
```

SStoreManager.java:3205)
	at
org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.run(RDBMSStoreManager.java:284
1)
	at
org.datanucleus.store.rdbms.AbstractSchemaTransaction.execute(AbstractSchemaTransaction.java:1
22)
	at
org.datanucleus.store.rdbms.RDBMSStoreManager.addClasses(RDBMSStoreManager.java:1605)
	at org.datanucleus.store.AbstractStoreManager.addClass(AbstractStoreManager.java:954)
	at
org.datanucleus.store.rdbms.RDBMSStoreManager.getDatastoreClass(RDBMSStoreManager.java:6
79)
	at
org.datanucleus.store.rdbms.query.RDBMSQueryUtils.getStatementForCandidates(RDBMSQueryUt
ils.java:408)
	at
org.datanucleus.store.rdbms.query.JDOQLQuery.compileQueryFull(JDOQLQuery.java:947)
	at org.datanucleus.store.rdbms.query.JDOQLQuery.compileInternal(JDOQLQuery.java:370)
	at org.datanucleus.store.query.Query.executeQuery(Query.java:1744)
	at org.datanucleus.store.query.Query.executeWithArray(Query.java:1672)
	at org.datanucleus.store.query.Query.execute(Query.java:1654)
	at org.datanucleus.api.jdo.JDOQuery.execute(JDOQuery.java:221)
	at
org.apache.hadoop.hive.metastore.MetaStoreDirectSql.ensureDbInit(MetaStoreDirectSql.java:185)
	at
org.apache.hadoop.hive.metastore.MetaStoreDirectSql.<init>(MetaStoreDirectSql.java:137)
	at org.apache.hadoop.hive.metastore.ObjectStore.initialize(ObjectStore.java:295)
	at org.apache.hadoop.hive.metastore.ObjectStore.setConf(ObjectStore.java:258)
	at org.apache.hadoop.util.ReflectionUtils.setConf(ReflectionUtils.java:73)
	at org.apache.hadoop.util.ReflectionUtils.newInstance(ReflectionUtils.java:133)
	at org.apache.hadoop.hive.metastore.RawStoreProxy.<init>(RawStoreProxy.java:57)
	at org.apache.hadoop.hive.metastore.RawStoreProxy.getProxy(RawStoreProxy.java:66)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.newRawStore(HiveMetaStore.java:
593)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.getMS(HiveMetaStore.java:571)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.createDefaultDB(HiveMetaStore.ja
va:620)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.init(HiveMetaStore.java:461)
	at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.<init>(RetryingHMSHandler.java:66)
	at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.getProxy(RetryingHMSHandler.java:72)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore.newRetryingHMSHandler(HiveMetaStore.java:57

62)
	at
org.apache.hadoop.hive.metastore.HiveMetaStoreClient.<init>(HiveMetaStoreClient.java:199)
	at
org.apache.hadoop.hive.ql.metadata.SessionHiveMetaStoreClient.<init>(SessionHiveMetaStoreClien
t.java:74)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1521)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.<init>(RetryingMetaStoreClient.java:86)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(RetryingMetaStoreClient.java:
132)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(RetryingMetaStoreClient.java:
104)
	at org.apache.hadoop.hive.ql.metadata.Hive.createMetaStoreClient(Hive.java:3005)
	at org.apache.hadoop.hive.ql.metadata.Hive.getMSC(Hive.java:3024)
	at org.apache.hadoop.hive.ql.metadata.Hive.getAllDatabases(Hive.java:1234)
	at org.apache.hadoop.hive.ql.metadata.Hive.reloadFunctions(Hive.java:174)
	at org.apache.hadoop.hive.ql.metadata.Hive.<clinit>(Hive.java:166)
	at org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:503)
	at org.apache.spark.sql.hive.client.HiveClientImpl.<init>(HiveClientImpl.scala:192)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at
org.apache.spark.sql.hive.client.IsolatedClientLoader.createClient(IsolatedClientLoader.scala:264)
	at org.apache.spark.sql.hive.HiveUtils$.newClientForMetadata(HiveUtils.scala:366)
	at org.apache.spark.sql.hive.HiveUtils$.newClientForMetadata(HiveUtils.scala:270)
	at org.apache.spark.sql.hive.HiveExternalCatalog.<init>(HiveExternalCatalog.scala:65)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at org.apache.spark.sql.internal.SharedState$.org$apache$spark$sql$internal$SharedState$

```
$reflect(SharedState.scala:166)
        at org.apache.spark.sql.internal.SharedState.<init>(SharedState.scala:86)
        at org.apache.spark.sql.SparkSession$$anonfun$sharedState$1.apply(SparkSession.scala:101)
        at org.apache.spark.sql.SparkSession$$anonfun$sharedState$1.apply(SparkSession.scala:101)
        at scala.Option.getOrElse(Option.scala:121)
        at org.apache.spark.sql.SparkSession.sharedState$lzycompute(SparkSession.scala:101)
        at org.apache.spark.sql.SparkSession.sharedState(SparkSession.scala:100)
        at org.apache.spark.sql.internal.SessionState.<init>(SessionState.scala:157)
        at org.apache.spark.sql.hive.HiveSessionState.<init>(HiveSessionState.scala:32)
        at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
        at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
        at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
        at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
        at org.apache.spark.sql.SparkSession$.org$apache$spark$sql$SparkSession$
$reflect(SparkSession.scala:978)
        at org.apache.spark.sql.SparkSession.sessionState$lzycompute(SparkSession.scala:110)
        at org.apache.spark.sql.SparkSession.sessionState(SparkSession.scala:109)
        at org.apache.spark.sql.SparkSession$Builder$
$anonfun$getOrCreate$5.apply(SparkSession.scala:878)
        at org.apache.spark.sql.SparkSession$Builder$
$anonfun$getOrCreate$5.apply(SparkSession.scala:878)
        at scala.collection.mutable.HashMap$$anonfun$foreach$1.apply(HashMap.scala:99)
        at scala.collection.mutable.HashMap$$anonfun$foreach$1.apply(HashMap.scala:99)
        at scala.collection.mutable.HashTable$class.foreachEntry(HashTable.scala:230)
        at scala.collection.mutable.HashMap.foreachEntry(HashMap.scala:40)
        at scala.collection.mutable.HashMap.foreach(HashMap.scala:99)
        at org.apache.spark.sql.SparkSession$Builder.getOrCreate(SparkSession.scala:878)
        at org.apache.spark.repl.Main$.createSparkSession(Main.scala:95)
        at $line3.$read$$iw$$iw.<init>(<console>:15)
        at $line3.$read$$iw.<init>(<console>:42)
        at $line3.$read.<init>(<console>:44)
        at $line3.$read$.<init>(<console>:48)
        at $line3.$read$.<clinit>(<console>)
        at $line3.$eval$.$print$lzycompute(<console>:7)
        at $line3.$eval$.$print(<console>:6)
        at $line3.$eval.$print(<console>)
        at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
        at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
        at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
        at java.lang.reflect.Method.invoke(Method.java:498)
        at scala.tools.nsc.interpreter.IMain$ReadEvalPrint.call(IMain.scala:786)
        at scala.tools.nsc.interpreter.IMain$Request.loadAndRun(IMain.scala:1047)
        at scala.tools.nsc.interpreter.IMain$WrappedRequest$
$anonfun$loadAndRunReq$1.apply(IMain.scala:638)
        at scala.tools.nsc.interpreter.IMain$WrappedRequest$
```

```
$anonfun$loadAndRunReq$1.apply(IMain.scala:637)
	at scala.reflect.internal.util.ScalaClassLoader$class.asContext(ScalaClassLoader.scala:31)
	at
scala.reflect.internal.util.AbstractFileClassLoader.asContext(AbstractFileClassLoader.scala:19)
	at scala.tools.nsc.interpreter.IMain$WrappedRequest.loadAndRunReq(IMain.scala:637)
	at scala.tools.nsc.interpreter.IMain.interpret(IMain.scala:569)
	at scala.tools.nsc.interpreter.IMain.interpret(IMain.scala:565)
	at scala.tools.nsc.interpreter.ILoop.interpretStartingWith(ILoop.scala:807)
	at scala.tools.nsc.interpreter.ILoop.command(ILoop.scala:681)
	at scala.tools.nsc.interpreter.ILoop.processLine(ILoop.scala:395)
	at org.apache.spark.repl.SparkILoop$
$anonfun$initializeSpark$1.apply$mcV$sp(SparkILoop.scala:38)
	at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
	at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
	at scala.tools.nsc.interpreter.IMain.beQuietDuring(IMain.scala:214)
	at org.apache.spark.repl.SparkILoop.initializeSpark(SparkILoop.scala:37)
	at org.apache.spark.repl.SparkILoop.loadFiles(SparkILoop.scala:105)
	at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply$mcZ$sp(ILoop.scala:920)
	at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
	at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
	at
scala.reflect.internal.util.ScalaClassLoader$.savingContextLoader(ScalaClassLoader.scala:97)
	at scala.tools.nsc.interpreter.ILoop.process(ILoop.scala:909)
	at org.apache.spark.repl.Main$.doMain(Main.scala:68)
	at org.apache.spark.repl.Main$.main(Main.scala:51)
	at org.apache.spark.repl.Main.main(Main.scala)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
	at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:498)
	at org.apache.spark.deploy.SparkSubmit$.org$apache$spark$deploy$SparkSubmit$
$runMain(SparkSubmit.scala:738)
	at org.apache.spark.deploy.SparkSubmit$.doRunMain$1(SparkSubmit.scala:187)
	at org.apache.spark.deploy.SparkSubmit$.submit(SparkSubmit.scala:212)
	at org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:126)
	at org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala)
Caused by: com.mysql.jdbc.exceptions.jdbc4.MySQLSyntaxErrorException: Specified key was too
long; max key length is 3072 bytes
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at com.mysql.jdbc.Util.handleNewInstance(Util.java:425)
	at com.mysql.jdbc.Util.getInstance(Util.java:408)
	at com.mysql.jdbc.SQLError.createSQLException(SQLError.java:944)
```

```
        at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:3976)
        at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:3912)
        at com.mysql.jdbc.MysqlIO.sendCommand(MysqlIO.java:2530)
        at com.mysql.jdbc.MysqlIO.sqlQueryDirect(MysqlIO.java:2683)
        at com.mysql.jdbc.ConnectionImpl.execSQL(ConnectionImpl.java:2482)
        at com.mysql.jdbc.ConnectionImpl.execSQL(ConnectionImpl.java:2440)
        at com.mysql.jdbc.StatementImpl.executeInternal(StatementImpl.java:845)
        at com.mysql.jdbc.StatementImpl.execute(StatementImpl.java:745)
        at com.jolbox.bonecp.StatementHandle.execute(StatementHandle.java:254)
        at
org.datanucleus.store.rdbms.table.AbstractTable.executeDdlStatement(AbstractTable.java:760)
        at org.datanucleus.store.rdbms.table.TableImpl.createIndices(TableImpl.java:648)
        at org.datanucleus.store.rdbms.table.TableImpl.createConstraints(TableImpl.java:422)
        at
org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.performTablesValidation(RDBMSSt
oreManager.java:3459)
        at
org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.addClassTablesAndValidate(RDBM
SStoreManager.java:3190)
        ... 128 more
Nested Throwables StackTrace:
com.mysql.jdbc.exceptions.jdbc4.MySQLSyntaxErrorException: Specified key was too long; max
key length is 3072 bytes
        at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
        at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
        at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
        at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
        at com.mysql.jdbc.Util.handleNewInstance(Util.java:425)
        at com.mysql.jdbc.Util.getInstance(Util.java:408)
        at com.mysql.jdbc.SQLError.createSQLException(SQLError.java:944)
        at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:3976)
        at com.mysql.jdbc.MysqlIO.checkErrorPacket(MysqlIO.java:3912)
        at com.mysql.jdbc.MysqlIO.sendCommand(MysqlIO.java:2530)
        at com.mysql.jdbc.MysqlIO.sqlQueryDirect(MysqlIO.java:2683)
        at com.mysql.jdbc.ConnectionImpl.execSQL(ConnectionImpl.java:2482)
        at com.mysql.jdbc.ConnectionImpl.execSQL(ConnectionImpl.java:2440)
        at com.mysql.jdbc.StatementImpl.executeInternal(StatementImpl.java:845)
        at com.mysql.jdbc.StatementImpl.execute(StatementImpl.java:745)
        at com.jolbox.bonecp.StatementHandle.execute(StatementHandle.java:254)
        at
org.datanucleus.store.rdbms.table.AbstractTable.executeDdlStatement(AbstractTable.java:760)
        at org.datanucleus.store.rdbms.table.TableImpl.createIndices(TableImpl.java:648)
        at org.datanucleus.store.rdbms.table.TableImpl.createConstraints(TableImpl.java:422)
        at
org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.performTablesValidation(RDBMSSt
oreManager.java:3459)
```

at
org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.addClassTablesAndValidate(RDBMSStoreManager.java:3190)
at
org.datanucleus.store.rdbms.RDBMSStoreManager$ClassAdder.run(RDBMSStoreManager.java:2841)
at
org.datanucleus.store.rdbms.AbstractSchemaTransaction.execute(AbstractSchemaTransaction.java:122)
at
org.datanucleus.store.rdbms.RDBMSStoreManager.addClasses(RDBMSStoreManager.java:1605)
at org.datanucleus.store.AbstractStoreManager.addClass(AbstractStoreManager.java:954)
at
org.datanucleus.store.rdbms.RDBMSStoreManager.getDatastoreClass(RDBMSStoreManager.java:679)
at
org.datanucleus.store.rdbms.query.RDBMSQueryUtils.getStatementForCandidates(RDBMSQueryUtils.java:408)
at
org.datanucleus.store.rdbms.query.JDOQLQuery.compileQueryFull(JDOQLQuery.java:947)
at org.datanucleus.store.rdbms.query.JDOQLQuery.compileInternal(JDOQLQuery.java:370)
at org.datanucleus.store.query.Query.executeQuery(Query.java:1744)
at org.datanucleus.store.query.Query.executeWithArray(Query.java:1672)
at org.datanucleus.store.query.Query.execute(Query.java:1654)
at org.datanucleus.api.jdo.JDOQuery.execute(JDOQuery.java:221)
at
org.apache.hadoop.hive.metastore.MetaStoreDirectSql.ensureDbInit(MetaStoreDirectSql.java:185)
at
org.apache.hadoop.hive.metastore.MetaStoreDirectSql.<init>(MetaStoreDirectSql.java:137)
at org.apache.hadoop.hive.metastore.ObjectStore.initialize(ObjectStore.java:295)
at org.apache.hadoop.hive.metastore.ObjectStore.setConf(ObjectStore.java:258)
at org.apache.hadoop.util.ReflectionUtils.setConf(ReflectionUtils.java:73)
at org.apache.hadoop.util.ReflectionUtils.newInstance(ReflectionUtils.java:133)
at org.apache.hadoop.hive.metastore.RawStoreProxy.<init>(RawStoreProxy.java:57)
at org.apache.hadoop.hive.metastore.RawStoreProxy.getProxy(RawStoreProxy.java:66)
at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.newRawStore(HiveMetaStore.java:593)
at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.getMS(HiveMetaStore.java:571)
at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.createDefaultDB(HiveMetaStore.java:620)
at
org.apache.hadoop.hive.metastore.HiveMetaStore$HMSHandler.init(HiveMetaStore.java:461)
at
org.apache.hadoop.hive.metastore.RetryingHMSHandler.<init>(RetryingHMSHandler.java:66)
at

org.apache.hadoop.hive.metastore.RetryingHMSHandler.getProxy(RetryingHMSHandler.java:72)
	at
org.apache.hadoop.hive.metastore.HiveMetaStore.newRetryingHMSHandler(HiveMetaStore.java:57
62)
	at
org.apache.hadoop.hive.metastore.HiveMetaStoreClient.<init>(HiveMetaStoreClient.java:199)
	at
org.apache.hadoop.hive.ql.metadata.SessionHiveMetaStoreClient.<init>(SessionHiveMetaStoreClien
t.java:74)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at org.apache.hadoop.hive.metastore.MetaStoreUtils.newInstance(MetaStoreUtils.java:1521)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.<init>(RetryingMetaStoreClient.java:86)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(RetryingMetaStoreClient.java:
132)
	at
org.apache.hadoop.hive.metastore.RetryingMetaStoreClient.getProxy(RetryingMetaStoreClient.java:
104)
	at org.apache.hadoop.hive.ql.metadata.Hive.createMetaStoreClient(Hive.java:3005)
	at org.apache.hadoop.hive.ql.metadata.Hive.getMSC(Hive.java:3024)
	at org.apache.hadoop.hive.ql.metadata.Hive.getAllDatabases(Hive.java:1234)
	at org.apache.hadoop.hive.ql.metadata.Hive.reloadFunctions(Hive.java:174)
	at org.apache.hadoop.hive.ql.metadata.Hive.<clinit>(Hive.java:166)
	at org.apache.hadoop.hive.ql.session.SessionState.start(SessionState.java:503)
	at org.apache.spark.sql.hive.client.HiveClientImpl.<init>(HiveClientImpl.scala:192)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at
org.apache.spark.sql.hive.client.IsolatedClientLoader.createClient(IsolatedClientLoader.scala:264)
	at org.apache.spark.sql.hive.HiveUtils$.newClientForMetadata(HiveUtils.scala:366)
	at org.apache.spark.sql.hive.HiveUtils$.newClientForMetadata(HiveUtils.scala:270)
	at org.apache.spark.sql.hive.HiveExternalCatalog.<init>(HiveExternalCatalog.scala:65)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at

sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at org.apache.spark.sql.internal.SharedState$.org$apache$spark$sql$internal$SharedState$
$reflect(SharedState.scala:166)
	at org.apache.spark.sql.internal.SharedState.<init>(SharedState.scala:86)
	at org.apache.spark.sql.SparkSession$$anonfun$sharedState$1.apply(SparkSession.scala:101)
	at org.apache.spark.sql.SparkSession$$anonfun$sharedState$1.apply(SparkSession.scala:101)
	at scala.Option.getOrElse(Option.scala:121)
	at org.apache.spark.sql.SparkSession.sharedState$lzycompute(SparkSession.scala:101)
	at org.apache.spark.sql.SparkSession.sharedState(SparkSession.scala:100)
	at org.apache.spark.sql.internal.SessionState.<init>(SessionState.scala:157)
	at org.apache.spark.sql.hive.HiveSessionState.<init>(HiveSessionState.scala:32)
	at sun.reflect.NativeConstructorAccessorImpl.newInstance0(Native Method)
	at
sun.reflect.NativeConstructorAccessorImpl.newInstance(NativeConstructorAccessorImpl.java:62)
	at
sun.reflect.DelegatingConstructorAccessorImpl.newInstance(DelegatingConstructorAccessorImpl.ja
va:45)
	at java.lang.reflect.Constructor.newInstance(Constructor.java:423)
	at org.apache.spark.sql.SparkSession$.org$apache$spark$sql$SparkSession$
$reflect(SparkSession.scala:978)
	at org.apache.spark.sql.SparkSession.sessionState$lzycompute(SparkSession.scala:110)
	at org.apache.spark.sql.SparkSession.sessionState(SparkSession.scala:109)
	at org.apache.spark.sql.SparkSession$Builder$
$anonfun$getOrCreate$5.apply(SparkSession.scala:878)
	at org.apache.spark.sql.SparkSession$Builder$
$anonfun$getOrCreate$5.apply(SparkSession.scala:878)
	at scala.collection.mutable.HashMap$$anonfun$foreach$1.apply(HashMap.scala:99)
	at scala.collection.mutable.HashMap$$anonfun$foreach$1.apply(HashMap.scala:99)
	at scala.collection.mutable.HashTable$class.foreachEntry(HashTable.scala:230)
	at scala.collection.mutable.HashMap.foreachEntry(HashMap.scala:40)
	at scala.collection.mutable.HashMap.foreach(HashMap.scala:99)
	at org.apache.spark.sql.SparkSession$Builder.getOrCreate(SparkSession.scala:878)
	at org.apache.spark.repl.Main$.createSparkSession(Main.scala:95)
	at $line3.$read$$iw$$iw.<init>(<console>:15)
	at $line3.$read$$iw.<init>(<console>:42)
	at $line3.$read.<init>(<console>:44)
	at $line3.$read$.<init>(<console>:48)
	at $line3.$read$.<clinit>(<console>)
	at $line3.$eval$.$print$lzycompute(<console>:7)
	at $line3.$eval$.$print(<console>:6)
	at $line3.$eval.$print(<console>)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)

```
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
	at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:498)
	at scala.tools.nsc.interpreter.IMain$ReadEvalPrint.call(IMain.scala:786)
	at scala.tools.nsc.interpreter.IMain$Request.loadAndRun(IMain.scala:1047)
	at scala.tools.nsc.interpreter.IMain$WrappedRequest$
$anonfun$loadAndRunReq$1.apply(IMain.scala:638)
	at scala.tools.nsc.interpreter.IMain$WrappedRequest$
$anonfun$loadAndRunReq$1.apply(IMain.scala:637)
	at scala.reflect.internal.util.ScalaClassLoader$class.asContext(ScalaClassLoader.scala:31)
	at
scala.reflect.internal.util.AbstractFileClassLoader.asContext(AbstractFileClassLoader.scala:19)
	at scala.tools.nsc.interpreter.IMain$WrappedRequest.loadAndRunReq(IMain.scala:637)
	at scala.tools.nsc.interpreter.IMain.interpret(IMain.scala:569)
	at scala.tools.nsc.interpreter.IMain.interpret(IMain.scala:565)
	at scala.tools.nsc.interpreter.ILoop.interpretStartingWith(ILoop.scala:807)
	at scala.tools.nsc.interpreter.ILoop.command(ILoop.scala:681)
	at scala.tools.nsc.interpreter.ILoop.processLine(ILoop.scala:395)
	at org.apache.spark.repl.SparkILoop$
$anonfun$initializeSpark$1.apply$mcV$sp(SparkILoop.scala:38)
	at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
	at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
	at scala.tools.nsc.interpreter.IMain.beQuietDuring(IMain.scala:214)
	at org.apache.spark.repl.SparkILoop.initializeSpark(SparkILoop.scala:37)
	at org.apache.spark.repl.SparkILoop.loadFiles(SparkILoop.scala:105)
	at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply$mcZ$sp(ILoop.scala:920)
	at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
	at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
	at
scala.reflect.internal.util.ScalaClassLoader$.savingContextLoader(ScalaClassLoader.scala:97)
	at scala.tools.nsc.interpreter.ILoop.process(ILoop.scala:909)
	at org.apache.spark.repl.Main$.doMain(Main.scala:68)
	at org.apache.spark.repl.Main$.main(Main.scala:51)
	at org.apache.spark.repl.Main.main(Main.scala)
	at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
	at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
	at
sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
	at java.lang.reflect.Method.invoke(Method.java:498)
	at org.apache.spark.deploy.SparkSubmit$.org$apache$spark$deploy$SparkSubmit$
$runMain(SparkSubmit.scala:738)
	at org.apache.spark.deploy.SparkSubmit$.doRunMain$1(SparkSubmit.scala:187)
	at org.apache.spark.deploy.SparkSubmit$.submit(SparkSubmit.scala:212)
	at org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:126)
	at org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala)
18/09/08 04:17:08 WARN metastore.ObjectStore: Failed to get database global_temp, returning
```

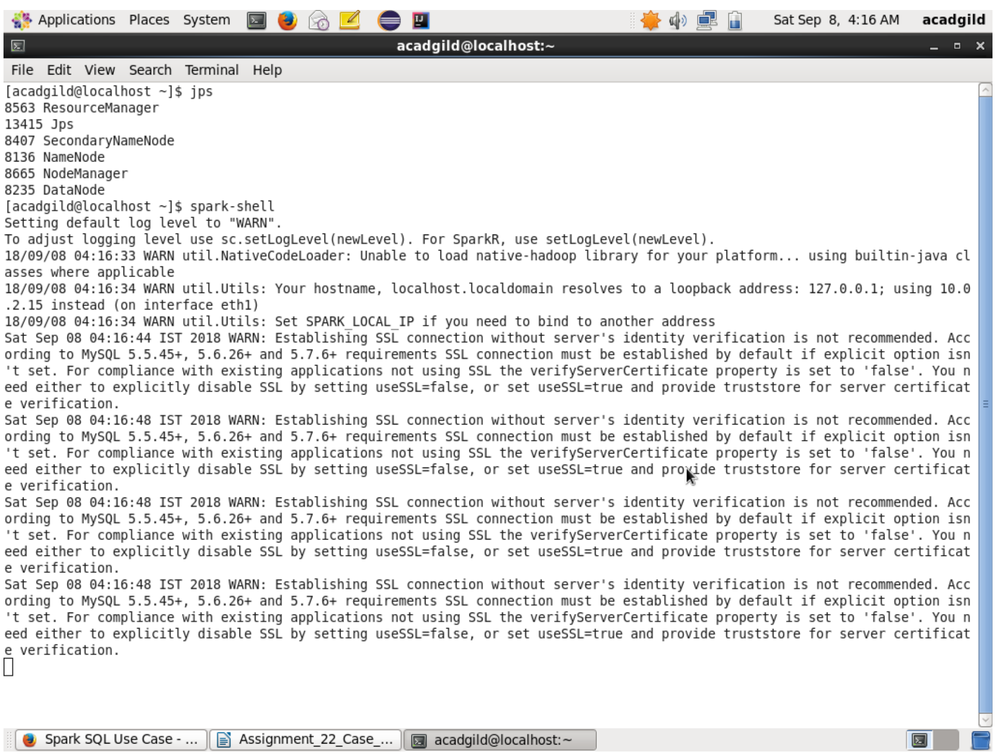
NoSuchObjectException
Spark context Web UI available at http://10.0.2.15:4040
Spark context available as 'sc' (master = local[*], app id = local-1536360396992).
Spark session available as 'spark'.
Welcome to

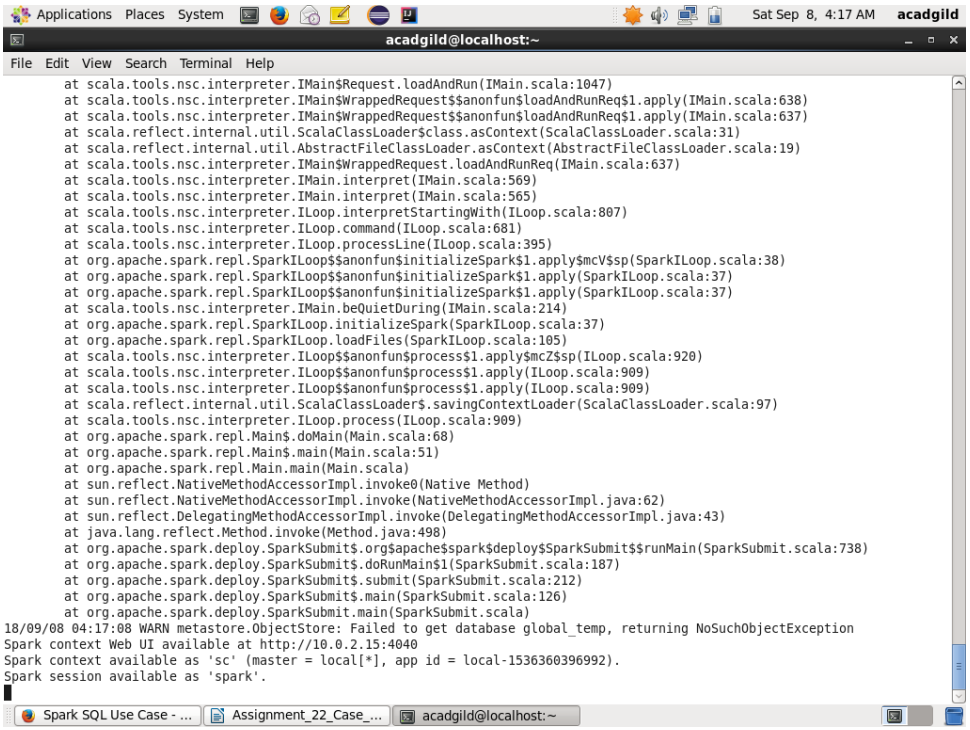
```
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.1.0
      /_/
```

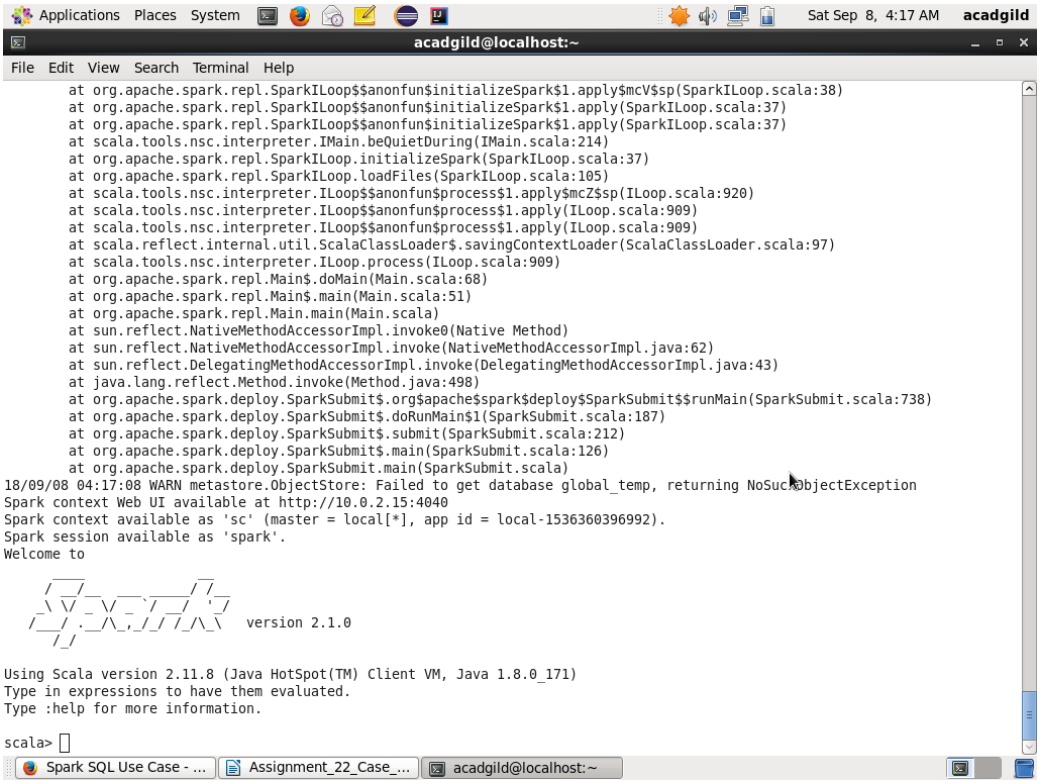
Using Scala version 2.11.8 (Java HotSpot(TM) Client VM, Java 1.8.0_171)
Type in expressions to have them evaluated.
Type :help for more information.

scala>

Output :




Applications Places System
Sat Sep 8, 4:17 AM acadgild
acadgild@localhost:~
File Edit View Search Terminal Help
at scala.tools.nsc.interpreter.IMain$Request.loadAndRun(IMain.scala:1047)
at scala.tools.nsc.interpreter.IMain$WrappedRequest$$anonfun$loadAndRunReq$1.apply(IMain.scala:638)
at scala.tools.nsc.interpreter.IMain$WrappedRequest$$anonfun$loadAndRunReq$1.apply(IMain.scala:637)
at scala.reflect.internal.util.ScalaClassLoader$class.asContext(ScalaClassLoader.scala:31)
at scala.reflect.internal.util.AbstractFileClassLoader.asContext(AbstractFileClassLoader.scala:19)
at scala.tools.nsc.interpreter.IMain$WrappedRequest.loadAndRunReq(IMain.scala:637)
at scala.tools.nsc.interpreter.IMain.interpret(IMain.scala:569)
at scala.tools.nsc.interpreter.IMain.interpret(IMain.scala:565)
at scala.tools.nsc.interpreter.ILoop.interpretStartingWith(ILoop.scala:807)
at scala.tools.nsc.interpreter.ILoop.command(ILoop.scala:681)
at scala.tools.nsc.interpreter.ILoop.processLine(ILoop.scala:395)
at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.applymcVsp(SparkILoop.scala:38)
at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
at scala.tools.nsc.interpreter.IMain.beQuietDuring(IMain.scala:214)
at org.apache.spark.repl.SparkILoop.initializeSpark(SparkILoop.scala:37)
at org.apache.spark.repl.SparkILoop.loadFiles(SparkILoop.scala:105)
at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.applymcZsp(ILoop.scala:920)
at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
at scala.reflect.internal.util.ScalaClassLoader$.savingContextLoader(ScalaClassLoader.scala:97)
at scala.tools.nsc.interpreter.ILoop.process(ILoop.scala:909)
at org.apache.spark.repl.Main$.doMain(Main.scala:68)
at org.apache.spark.repl.Main$.main(Main.scala:51)
at org.apache.spark.repl.Main.main(Main.scala)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at org.apache.spark.deploy.SparkSubmit$.org$apache$spark$deploy$SparkSubmit$$runMain(SparkSubmit.scala:738)
at org.apache.spark.deploy.SparkSubmit$.doRunMain$1(SparkSubmit.scala:187)
at org.apache.spark.deploy.SparkSubmit$.submit(SparkSubmit.scala:212)
at org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:126)
at org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala)
18/09/08 04:17:08 WARN metastore.ObjectStore: Failed to get database global_temp, returning NoSuchObjectException
Spark context Web UI available at http://10.0.2.15:4040
Spark context available as 'sc' (master = local[*], app id = local-1536360396992).
Spark session available as 'spark'.
Spark SQL Use Case - ...
Assignment_22_Case_...
acadgild@localhost:~


Applications Places System
Sat Sep 8, 4:17 AM acadgild
acadgild@localhost:~
File Edit View Search Terminal Help
at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.applymcVsp(SparkILoop.scala:38)
at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
at org.apache.spark.repl.SparkILoop$$anonfun$initializeSpark$1.apply(SparkILoop.scala:37)
at scala.tools.nsc.interpreter.IMain.beQuietDuring(IMain.scala:214)
at org.apache.spark.repl.SparkILoop.initializeSpark(SparkILoop.scala:37)
at org.apache.spark.repl.SparkILoop.loadFiles(SparkILoop.scala:105)
at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.applymcZsp(ILoop.scala:920)
at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
at scala.tools.nsc.interpreter.ILoop$$anonfun$process$1.apply(ILoop.scala:909)
at scala.reflect.internal.util.ScalaClassLoader$.savingContextLoader(ScalaClassLoader.scala:97)
at scala.tools.nsc.interpreter.ILoop.process(ILoop.scala:909)
at org.apache.spark.repl.Main$.doMain(Main.scala:68)
at org.apache.spark.repl.Main$.main(Main.scala:51)
at org.apache.spark.repl.Main.main(Main.scala)
at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
at java.lang.reflect.Method.invoke(Method.java:498)
at org.apache.spark.deploy.SparkSubmit$.org$apache$spark$deploy$SparkSubmit$$runMain(SparkSubmit.scala:738)
at org.apache.spark.deploy.SparkSubmit$.doRunMain$1(SparkSubmit.scala:187)
at org.apache.spark.deploy.SparkSubmit$.submit(SparkSubmit.scala:212)
at org.apache.spark.deploy.SparkSubmit$.main(SparkSubmit.scala:126)
at org.apache.spark.deploy.SparkSubmit.main(SparkSubmit.scala)
18/09/08 04:17:08 WARN metastore.ObjectStore: Failed to get database global_temp, returning NoSuchObjectException
Spark context Web UI available at http://10.0.2.15:4040
Spark context available as 'sc' (master = local[*], app id = local-1536360396992).
Spark session available as 'spark'.
Welcome to
version 2.1.0
Using Scala version 2.11.8 (Java HotSpot(TM) Client VM, Java 1.8.0_171)
Type in expressions to have them evaluated.
Type :help for more information.
scala>
Spark SQL Use Case - ...
Assignment_22_Case_...
acadgild@localhost:~

Solution:

Objective 1 :

- **Load HVAC.csv file into temporary table**

Explaination :

1. To create RDD from the data set in HVAC.csv file present locally.

```
val data = sc.textFile("file:///home/acadgild/Desktop/HVAC.csv")
```

2. To remove header from the RDD present in dataset.

```
val header = data.first()

val data1 = data.filter(row => row != header)
```

3. Writing a case class for holding the schema of the dataset.

```
case class
hvac_cls(Date:String,Time:String,TargetTemp:Int,ActualTemp:Int,System:Int,SystemAge:Int,
BuildingId:Int)
```

4. Splitting each row of the dataset with the delimiter ‘as’ . Then, mapping the columns to our case class. finally, we are converting it into a data frame.

```
val hvac = data1.map(x=>x.split(",")).map(x =>
hvac_cls(x(0),x(1),x(2).toInt,x(3).toInt,x(4).toInt,x(5).toInt,x(6).toInt)).toDF
```

5. Creating a table HVAC for our dataframe.

hvac.registerTempTable("HVAC")

Terminal Execution :

scala> val data = sc.textFile("file:///home/acadgild/Desktop/HVAC.csv")

data: org.apache.spark.rdd.RDD[String] = file:///home/acadgild/Desktop/HVAC.csv MapPartitionsRDD[3] at textFile at <console>:24

scala> val header = data.first()

header: String = Date,Time,TargetTemp,ActualTemp,System,SystemAge,BuildingID

scala> val data1 = data.filter(row => row != header)

data1: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[6] at filter at <console>:28

scala> case class hvac_cls(Date:String,Time:String,TargetTemp:Int,ActualTemp:Int,System:Int,SystemAge:Int,BuildingId:Int)

defined class hvac_cls

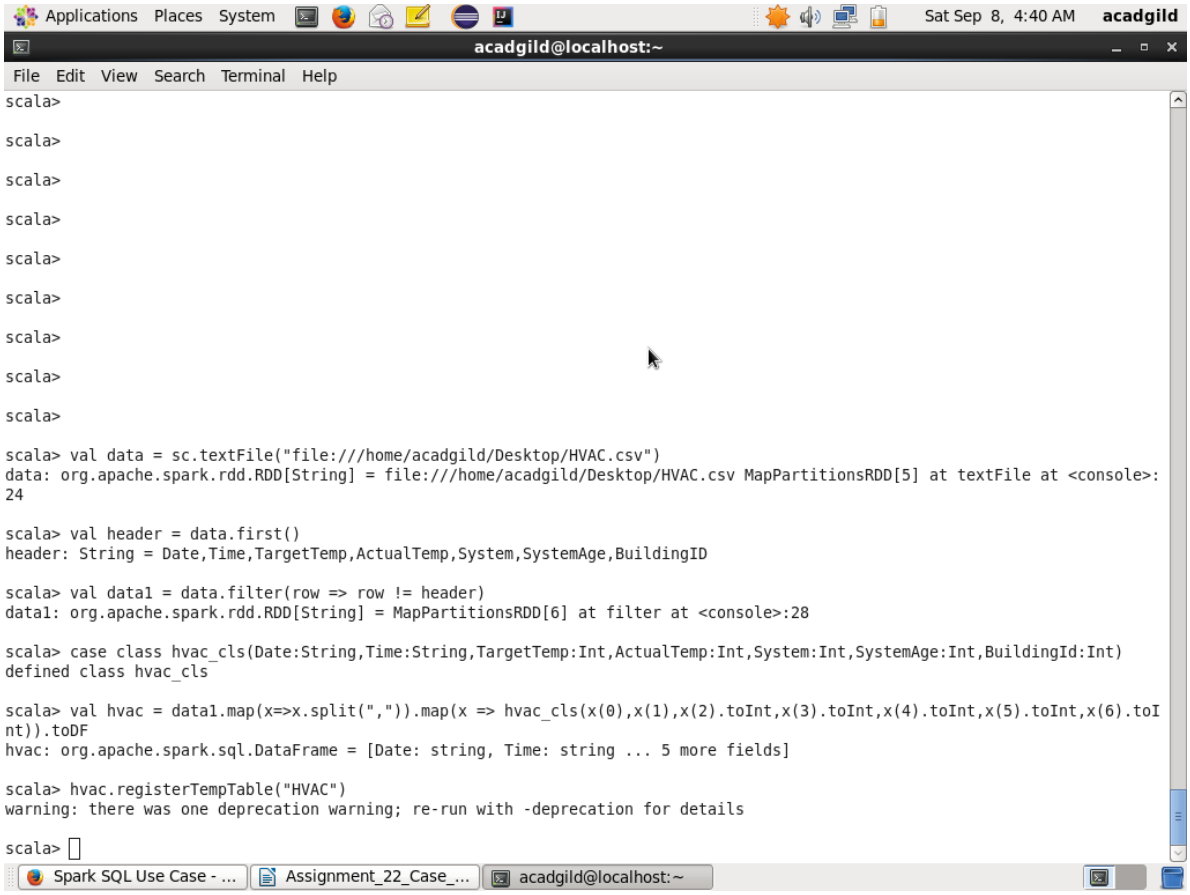
scala> val hvac = data1.map(x=>x.split(",")).map(x => hvac_cls(x(0),x(1),x(2).toInt,x(3).toInt,x(4).toInt,x(5).toInt,x(6).toInt)).toDF

hvac: org.apache.spark.sql.DataFrame = [Date: string, Time: string ... 5 more fields]

scala> hvac.registerTempTable("HVAC")

warning: there was one deprecation warning; re-run with -deprecation for details

Output :


Applications Places System
Sat Sep 8, 4:40 AM acadgild
acadgild@localhost:~
File Edit View Search Terminal Help
scala>
scala>
scala>
scala>
scala>
scala>
scala>
scala>
scala>
scala> val data = sc.textFile("file:///home/acadgild/Desktop/HVAC.csv")
data: org.apache.spark.rdd.RDD[String] = file:///home/acadgild/Desktop/HVAC.csv MapPartitionsRDD[5] at textFile at <console>:
24
scala> val header = data.first()
header: String = Date,Time,TargetTemp,ActualTemp,System,SystemAge,BuildingID
scala> val data1 = data.filter(row => row != header)
data1: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[6] at filter at <console>:28
scala> case class hvac_cls(Date:String,Time:String,TargetTemp:Int,ActualTemp:Int,System:Int,SystemAge:Int,BuildingId:Int)
defined class hvac_cls
scala> val hvac = data1.map(x=>x.split(",")).map(x => hvac_cls(x(0),x(1),x(2).toInt,x(3).toInt,x(4).toInt,x(5).toInt,x(6).toI
nt)).toDF
hvac: org.apache.spark.sql.DataFrame = [Date: string, Time: string ... 5 more fields]
scala> hvac.registerTempTable("HVAC")
warning: there was one deprecation warning; re-run with -deprecation for details
scala>
Spark SQL Use Case - ...
Assignment_22_Case_...
acadgild@localhost:~

- **Add a new column, tempchange - set to 1, if there is a change of greater than +/-5 between actual and target temperature**

Explaination :

1. Perform a SQL query on the table to creates one new column tempchange. This 'tempchange ' column will set to 1 if there is a temperature change of either +5 or -5 between the actual_temperature and the target_temperature

```
val hvac1 = sqlContext.sql("select *,IF((targettemp - actualtemp) > 5, '1',
IF((targettemp - actualtemp) < -5, '1', 0)) AS tempchange from HVAC")
```

Terminal Execution :

```
scala> val sqlContext = new org.apache.spark.sql.SQLContext(sc)
warning: there was one deprecation warning; re-run with -deprecation for details
sqlContext: org.apache.spark.sql.SQLContext = org.apache.spark.sql.SQLContext@1377df8

scala> val hvac1 = sqlContext.sql("select *,IF((targettemp - actualtemp) > 5, '1', IF((targettemp - actualtemp) < -5, '1', 0)) AS tempchange from HVAC")
hvac1: org.apache.spark.sql.DataFrame = [Date: string, Time: string ... 6 more fields]
scala> hvac1.registerTempTable("HVAC1")
warning: there was one deprecation warning; re-run with -deprecation for details
scala> hvac1.show
+-------+--------+----------+----------+------+---------+----------+----------+
|   Date|    Time|TargetTemp|ActualTemp|System|SystemAge|BuildingId|tempchange|
+-------+--------+----------+----------+------+---------+----------+----------+
| 6/1/13| 0:00:01|        66|        58|    13|       20|         4|         1|
```

| 6/2/13| 1:00:01| 69| 68| 3| 20| 17| 0|

| 6/3/13| 2:00:01| 70| 73| 17| 20| 18| 0|

| 6/4/13| 3:00:01| 67| 63| 2| 23| 15| 0|

| 6/5/13| 4:00:01| 68| 74| 16| 9| 3| 1|

| 6/6/13| 5:00:01| 67| 56| 13| 28| 4| 1|

| 6/7/13| 6:00:01| 70| 58| 12| 24| 2| 1|

| 6/8/13| 7:00:01| 70| 73| 20| 26| 16| 0|

| 6/9/13| 8:00:01| 66| 69| 16| 9| 9| 0|

|6/10/13| 9:00:01| 65| 57| 6| 5| 12| 1|

|6/11/13|10:00:01| 67| 70| 10| 17| 15| 0|

|6/12/13|11:00:01| 69| 62| 2| 11| 7| 1|

|6/13/13|12:00:01| 69| 73| 14| 2| 15| 0|

|6/14/13|13:00:01| 65| 61| 3| 2| 6| 0|

|6/15/13|14:00:01| 67| 59| 19| 22| 20| 1|

|6/16/13|15:00:01| 65| 56| 19| 11| 8| 1|

|6/17/13|16:00:01| 67| 57| 15| 7| 6| 1|

|6/18/13|17:00:01| 66| 57| 12| 5| 13| 1|

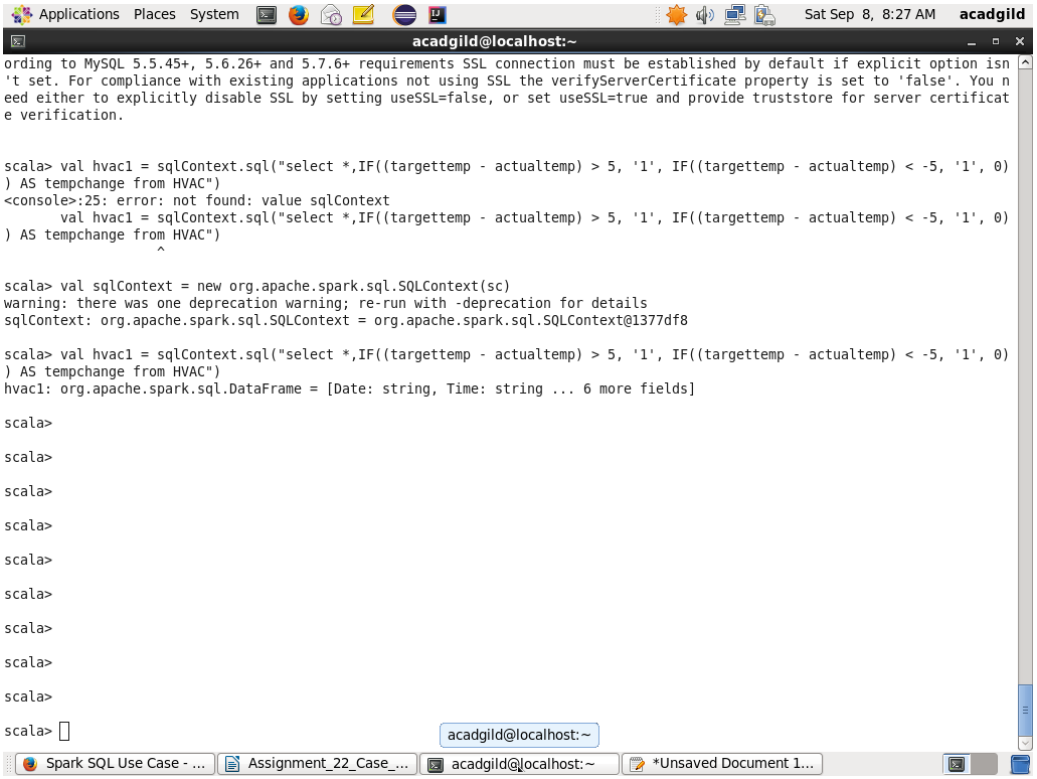
|6/19/13|18:00:01| 69| 58| 8| 22| 4| 1|

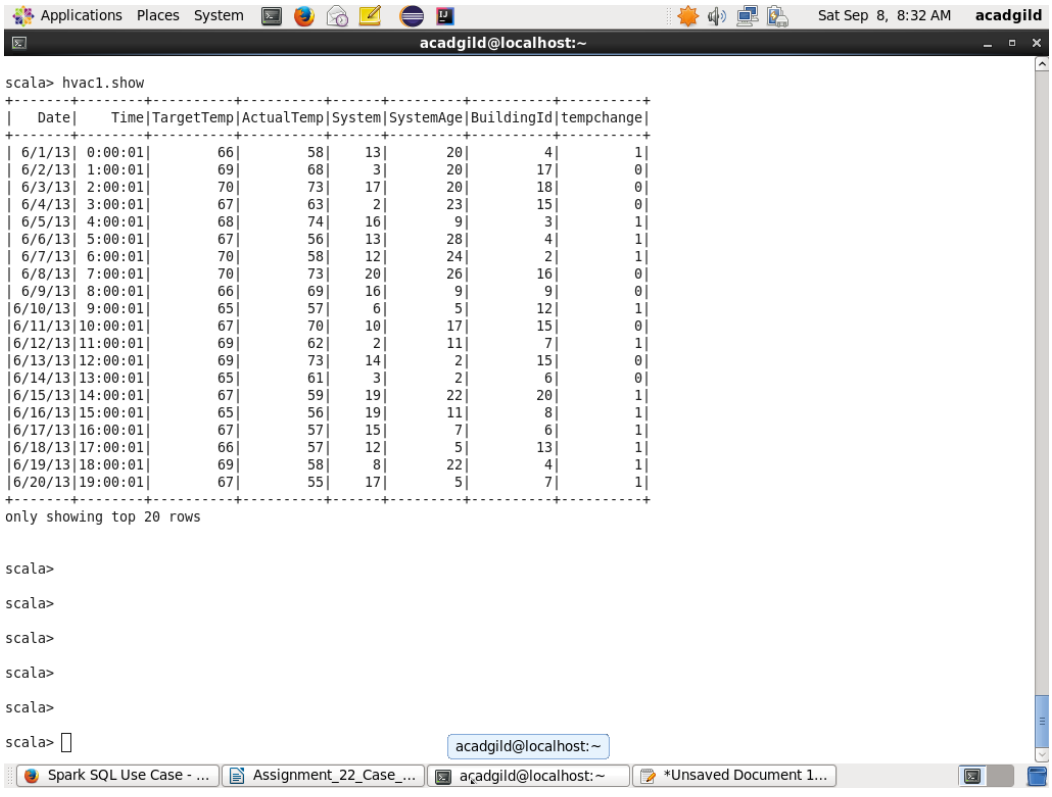
|6/20/13|19:00:01| 67| 55| 17| 5| 7| 1|

+-------+--------+----------+----------+------+---------+----------+----------+

only showing top 20 rows

Output :


Applications Places System
Sat Sep 8, 8:27 AM acadgild
acadgild@localhost:~
ording to MySQL 5.5.45+, 5.6.26+ and 5.7.6+ requirements SSL connection must be established by default if explicit option isn
't set. For compliance with existing applications not using SSL the verifyServerCertificate property is set to 'false'. You n
eed either to explicitly disable SSL by setting useSSL=false, or set useSSL=true and provide truststore for server certificat
e verification.
scala> val hvac1 = sqlContext.sql("select *,IF((targettemp - actualtemp) > 5, '1', IF((targettemp - actualtemp) < -5, '1', 0)
) AS tempchange from HVAC")
<console>:25: error: not found: value sqlContext
val hvac1 = sqlContext.sql("select *,IF((targettemp - actualtemp) > 5, '1', IF((targettemp - actualtemp) < -5, '1', 0)
) AS tempchange from HVAC")
^
scala> val sqlContext = new org.apache.spark.sql.SQLContext(sc)
warning: there was one deprecation warning; re-run with -deprecation for details
sqlContext: org.apache.spark.sql.SQLContext = org.apache.spark.sql.SQLContext@1377df8
scala> val hvac1 = sqlContext.sql("select *,IF((targettemp - actualtemp) > 5, '1', IF((targettemp - actualtemp) < -5, '1', 0)
) AS tempchange from HVAC")
hvac1: org.apache.spark.sql.DataFrame = [Date: string, Time: string ... 6 more fields]
scala>
scala>
scala>
scala>
scala>
scala>
scala>
scala>
scala>
acadgild@localhost:~
Spark SQL Use Case - ...
Assignment_22_Case_...
acadgild@localhost:~
*Unsaved Document 1...


Applications Places System
Sat Sep 8, 8:32 AM acadgild
acadgild@localhost:~
scala> hvac1.show
+-------+--------+----------+----------+------+---------+----------+----------+
| Date| Time|TargetTemp|ActualTemp|System|SystemAge|BuildingId|tempchange|
+-------+--------+----------+----------+------+---------+----------+----------+
6/1/13	0:00:01	66	58	13	20	4	1
6/2/13	1:00:01	69	68	3	20	17	0
6/3/13	2:00:01	70	73	17	20	18	0
6/4/13	3:00:01	67	63	2	23	15	0
6/5/13	4:00:01	68	74	16	9	3	1
6/6/13	5:00:01	67	56	13	28	4	1
6/7/13	6:00:01	70	58	12	24	2	1
6/8/13	7:00:01	70	73	20	26	16	0
6/9/13	8:00:01	66	69	16	9	9	0
6/10/13	9:00:01	65	57	6	5	12	1
6/11/13	10:00:01	67	70	10	17	15	0
6/12/13	11:00:01	69	62	2	11	7	1
6/13/13	12:00:01	69	73	14	2	15	0
6/14/13	13:00:01	65	61	3	2	6	0
6/15/13	14:00:01	67	59	19	22	20	1
6/16/13	15:00:01	65	56	19	11	8	1
6/17/13	16:00:01	67	57	15	7	6	1
6/18/13	17:00:01	66	57	12	5	13	1
6/19/13	18:00:01	69	58	8	22	4	1
6/20/13	19:00:01	67	55	17	5	7	1
+-------+--------+----------+----------+------+---------+----------+----------+
only showing top 20 rows
scala>
scala>
scala>
scala>
scala>
scala>
acadgild@localhost:~
Spark SQL Use Case - ...
Assignment_22_Case_...
acadgild@localhost:~
*Unsaved Document 1...

Objective 2 :

- **Load building.csv file into temporary table**

Explaination :

1. To create RDD from the data set in building.csv file present locally.

```
val data2 = sc.textFile("file:///home/acadgild/Desktop/building.csv")
```

2. To remove header from the RDD present in dataset.

```
val header1 = data2.first()
val data3 = data2.filter(row => row != header1)
```

3. Writing a case class for holding the schema of the dataset.

```
case class
building(buildid:Int,buildmgr:String,buildAge:Int,hvacproduct:String,Country:String)
```

4. Splitting each row of the dataset with the delimiter ‘as’ . Then, mapping the columns to our case class. finally, we are converting it into a data frame.

```
val build = data3.map(x=> x.split(",")).map(x =>
building(x(0).toInt,x(1),x(2).toInt,x(3),x(4))).toDF
```

5. Creating a table HVAC for our dataframe.

```
build.registerTempTable("building")
```

Terminal Execution :

```
scala> val data2 = sc.textFile("file:///home/acadgild/Desktop/building.csv")

data2: org.apache.spark.rdd.RDD[String] = file:///home/acadgild/Desktop/building.csv
MapPartitionsRDD[20] at textFile at <console>:26


scala> val header1 = data2.first()

header1: String = BuildingID,BuildingMgr,BuildingAge,HVACproduct,Country


scala> val data3 = data2.filter(row => row != header1)

data3: org.apache.spark.rdd.RDD[String] = MapPartitionsRDD[21] at filter at <console>:30


scala> data3.foreach(println)

1,M1,25,AC1000,USA

2,M2,27,FN39TG,France

3,M3,28,JDNS77,Brazil

4,M4,17,GG1919,Finland

5,M5,3,ACMAX22,Hong Kong

6,M6,9,AC1000,Singapore

7,M7,13,FN39TG,South Africa

8,M8,25,JDNS77,Australia

9,M9,11,GG1919,Mexico

10,M10,23,ACMAX22,China

11,M11,14,AC1000,Belgium

12,M12,26,FN39TG,Finland

13,M13,25,JDNS77,Saudi Arabia

14,M14,17,GG1919,Germany

15,M15,19,ACMAX22,Israel
```

16,M16,23,AC1000,Turkey

17,M17,11,FN39TG,Egypt

18,M18,25,JDNS77,Indonesia

19,M19,14,GG1919,Canada

20,M20,19,ACMAX22,Argentina

```
scala> case class
building(buildid:Int,buildmgr:String,buildAge:Int,hvacproduct:String,Country:String)

defined class building

scala> val build = data3.map(x=> x.split(",")).map(x =>
building(x(0).toInt,x(1),x(2).toInt,x(3),x(4))).toDF

build: org.apache.spark.sql.DataFrame = [buildid: int, buildmgr: string ... 3 more fields]

scala> build.show

+-------+--------+--------+-----------+------------+

|buildid|buildmgr|buildAge|hvacproduct|     Country|

+-------+--------+--------+-----------+------------+

|      1|      M1|      25|     AC1000|         USA|

|      2|      M2|      27|     FN39TG|      France|

|      3|      M3|      28|     JDNS77|      Brazil|

|      4|      M4|      17|     GG1919|     Finland|

|      5|      M5|       3|    ACMAX22|   Hong Kong|

|      6|      M6|       9|     AC1000|   Singapore|

|      7|      M7|      13|     FN39TG|South Africa|

|      8|      M8|      25|     JDNS77|   Australia|

|      9|      M9|      11|     GG1919|      Mexico|

|     10|     M10|      23|    ACMAX22|       China|

|     11|     M11|      14|     AC1000|     Belgium|

|     12|     M12|      26|     FN39TG|     Finland|
```

| 13| M13| 25| JDNS77|Saudi Arabia|

| 14| M14| 17| GG1919| Germany|

| 15| M15| 19| ACMAX22| Israel|

| 16| M16| 23| AC1000| Turkey|

| 17| M17| 11| FN39TG| Egypt|

| 18| M18| 25| JDNS77| Indonesia|

| 19| M19| 14| GG1919| Canada|

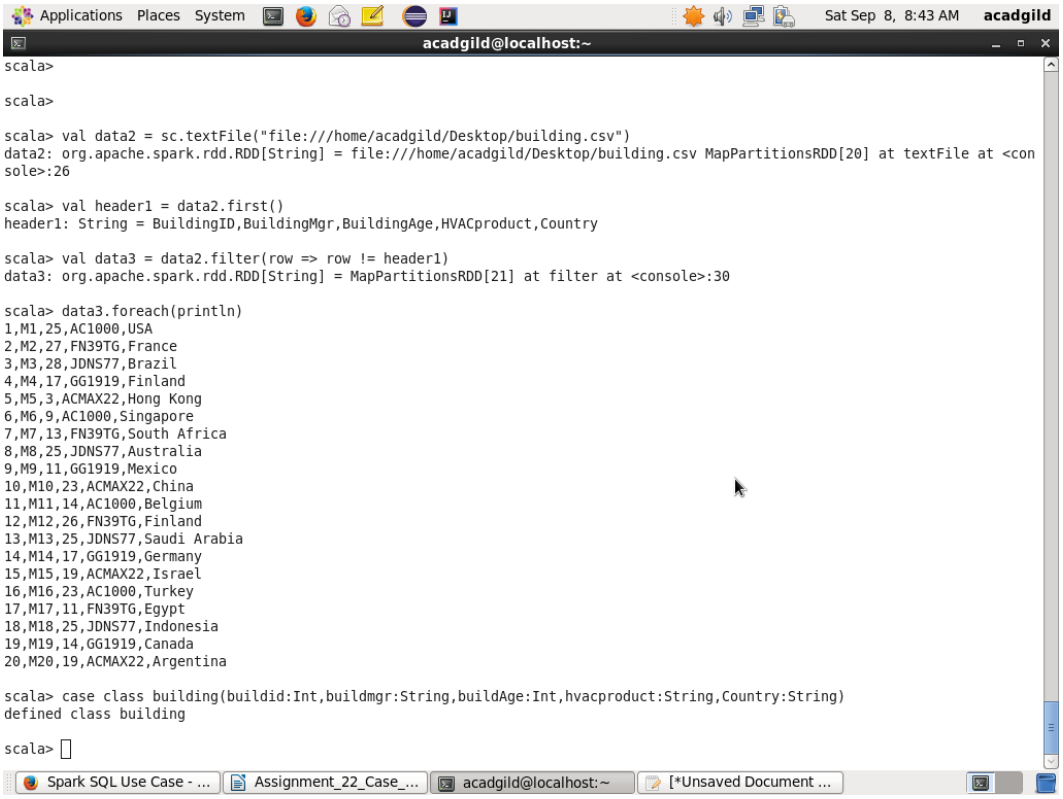
| 20| M20| 19| ACMAX22| Argentina|

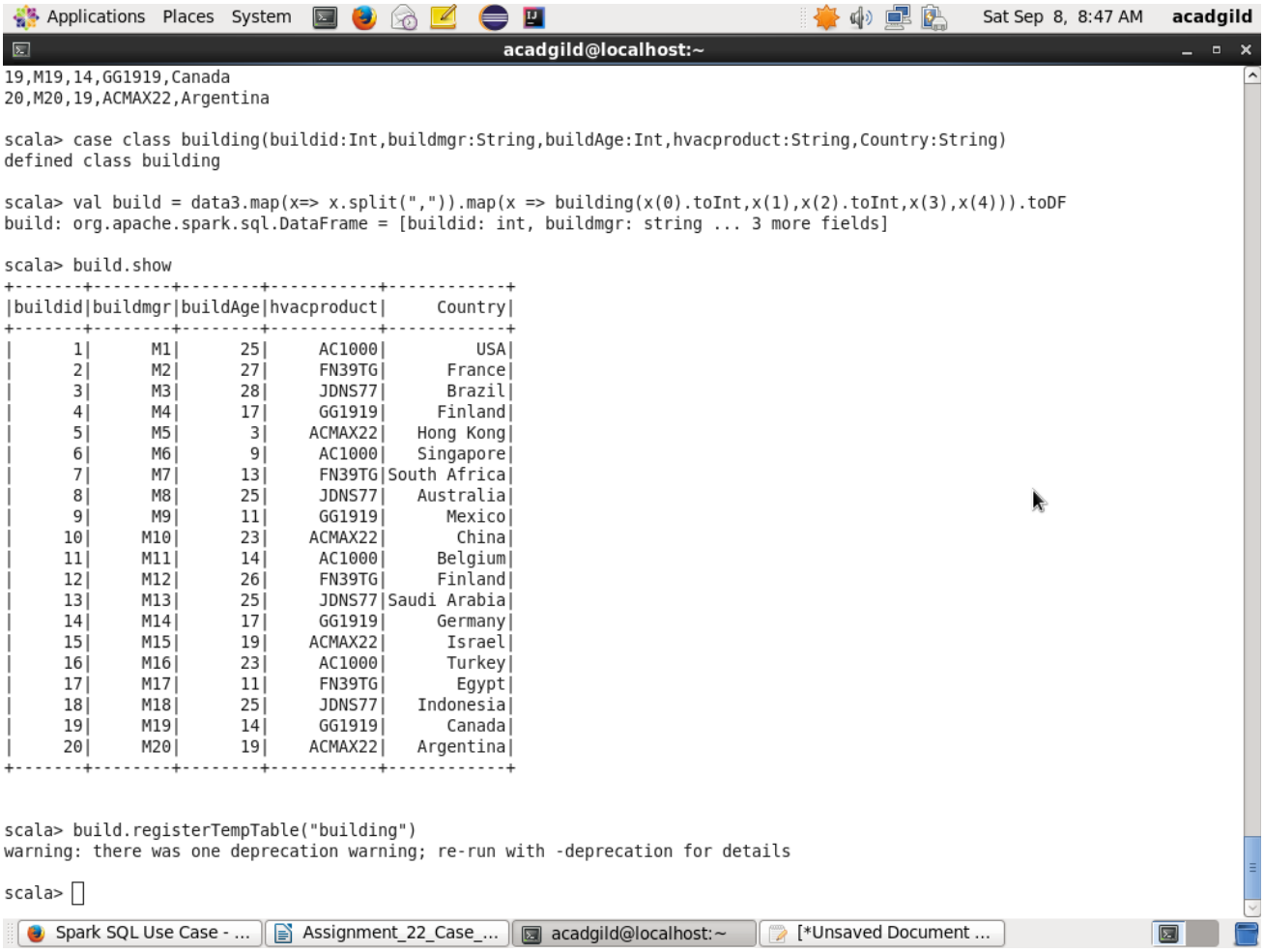
+-------+--------+--------+-----------+------------+

scala> build.registerTempTable("building")

warning: there was one deprecation warning; re-run with -deprecation for details

Output :




Applications Places System
Sat Sep 8, 8:47 AM acadgild
acadgild@localhost:~
19,M19,14,GG1919,Canada
20,M20,19,ACMAX22,Argentina
scala> case class building(buildid:Int,buildmgr:String,buildAge:Int,hvacproduct:String,Country:String)
defined class building
scala> val build = data3.map(x=> x.split(",")).map(x => building(x(0).toInt,x(1),x(2).toInt,x(3),x(4))).toDF
build: org.apache.spark.sql.DataFrame = [buildid: int, buildmgr: string ... 3 more fields]
scala> build.show
+-------+--------+--------+-----------+------------+
|buildid|buildmgr|buildAge|hvacproduct| Country|
+-------+--------+--------+-----------+------------+
1	M1	25	AC1000	USA
2	M2	27	FN39TG	France
3	M3	28	JDNS77	Brazil
4	M4	17	GG1919	Finland
5	M5	3	ACMAX22	Hong Kong
6	M6	9	AC1000	Singapore
7	M7	13	FN39TG	South Africa
8	M8	25	JDNS77	Australia
9	M9	11	GG1919	Mexico
10	M10	23	ACMAX22	China
11	M11	14	AC1000	Belgium
12	M12	26	FN39TG	Finland
13	M13	25	JDNS77	Saudi Arabia
14	M14	17	GG1919	Germany
15	M15	19	ACMAX22	Israel
16	M16	23	AC1000	Turkey
17	M17	11	FN39TG	Egypt
18	M18	25	JDNS77	Indonesia
19	M19	14	GG1919	Canada
20	M20	19	ACMAX22	Argentina
+-------+--------+--------+-----------+------------+
scala> build.registerTempTable("building")
warning: there was one deprecation warning; re-run with -deprecation for details
scala>
Spark SQL Use Case - ...
Assignment_22_Case_...
acadgild@localhost:~
[*Unsaved Document ...

Objective 3 :

- **Figure out the number of times, temperature has changed by 5 degrees or more for each country:**

Explaination :

1. Join the two datasets using the buildingId:

```
val build1 = sqlContext.sql("select h.*, b.country, b.hvacproduct from building b join hvac1 h on buildid = buildingid")
```

2. Select the tempchange column and the country column, to find the maximum temperature change in the areas.

```
val test = build1.map(x => (new Integer(x(7).toString),x(8).toString))
```

3. Filter the rows which have a change in temperature, which is identified by value of tempchange column as 1.

```
val test1 = test.filter(x=> {if(x._1==1) true else false})
```

4. Take the country and add 1 to know how many times the temperature in that building has changed. Here, reduceByKey operation is applied on the data to count the number of times temperature has been changed and finally, sorting and displaying the result in descending order.

```
val test2 = test1.map(x=>(x._2,1)).groupBy("_1").count().sort(desc("count")).show
```

Terminal Execution :

```
scala> val build1 = sqlContext.sql("select h.*, b.country, b.hvacproduct from building b join hvac1 h on buildid = buildingid")
build1: org.apache.spark.sql.DataFrame = [Date: string, Time: string ... 8 more fields]
scala> val test = build1.map(x => (new Integer(x(7).toString),x(8).toString))
test: org.apache.spark.sql.Dataset[(Integer, String)] = [_1: int, _2: string]

scala> val test1 = test.filter(x=> {if(x._1==1) true else false})
test1: org.apache.spark.sql.Dataset[(Integer, String)] = [_1: int, _2: string]
scala> val test2 = test1.map(x=>(x._2,1)).groupBy("_1").count().sort(desc("count")).show
+------------+-----+
|          _1|count|
+------------+-----+
|     Finland|  473|
|      France|  251|
|   Hong Kong|  248|
|      Turkey|  243|
|   Indonesia|  243|
|       China|  241|
|South Africa|  237|
|       Egypt|  236|
|Saudi Arabia|  233|
|      Canada|  232|
|      Israel|  232|
|   Argentina|  230|
|   Singapore|  230|
|      Mexico|  228|
```

| Brazil| 226|

| Australia| 225|

| USA| 213|

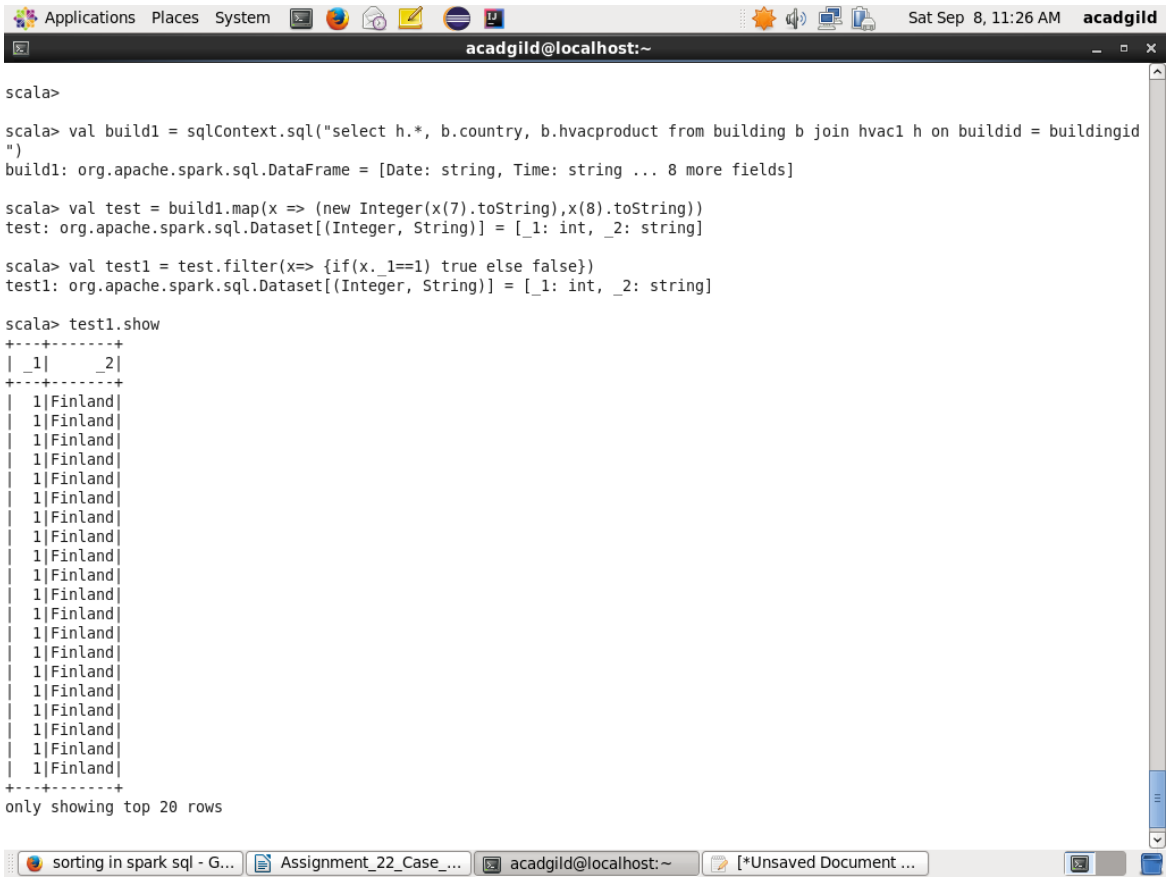
| Belgium| 199|

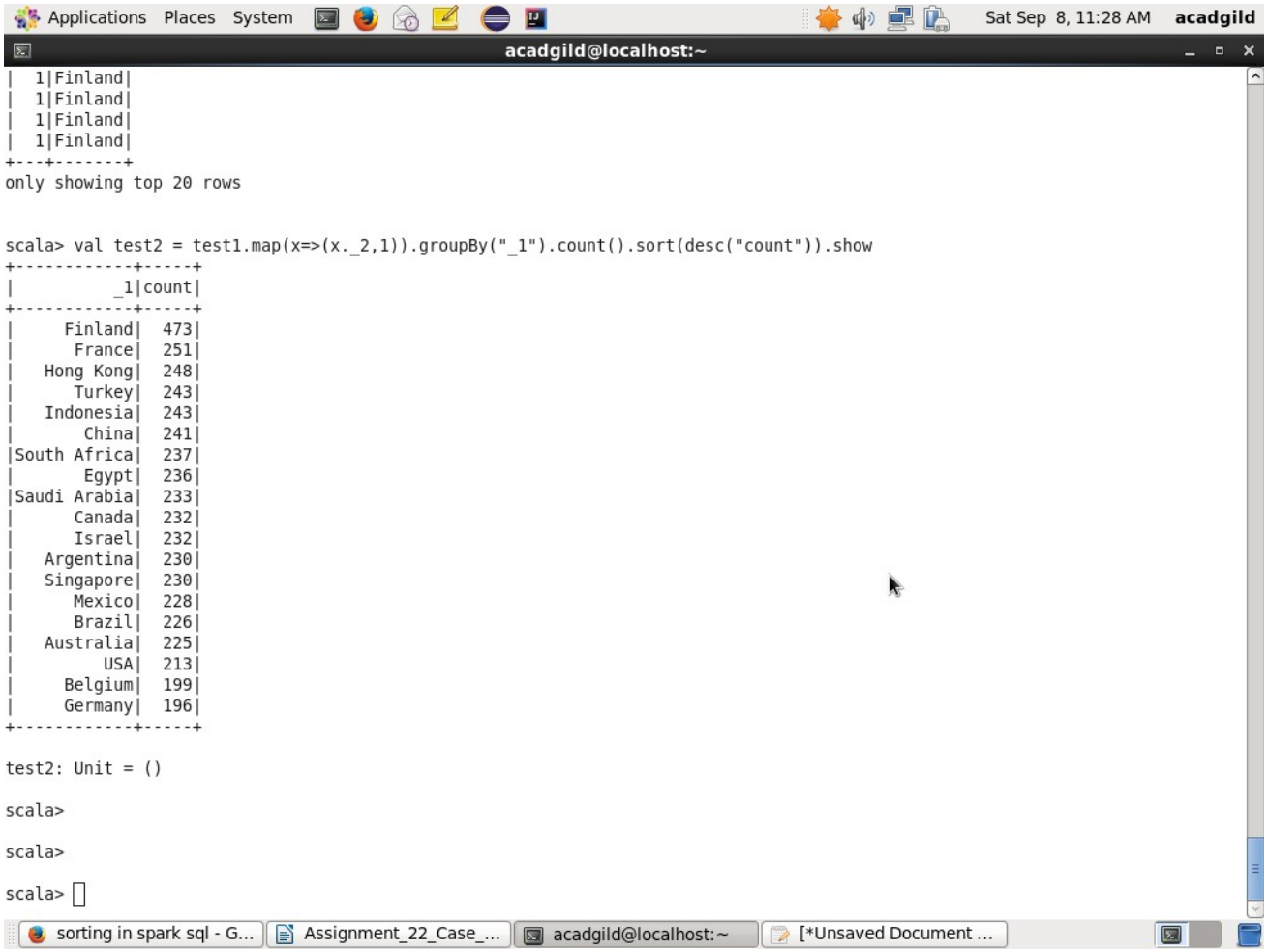
| Germany| 196|

+----------+-----+

test2: Unit = ()

Output :





Result of the Analysis:

From the above output, we can say that temperature in Finland is changing more frequently followed by France and Hong Kong.

Nice To Know :

If there is a continuous stream of data collected from the sensors, we can automate this analysis using Spark Streaming to know the temperature changes in the real-time, so that we can take accurate measures to reduce the temperature changes.