

- ```
“tar file tar -xvf mysql-connector-java-8.0.25.tar.gz “
```

- The MySQL Connector directory created in the previous step and then we copy it to the Sqoop library to complete the installation.

“cd mysql-connector-java-8.0.25/ “

“sudo cp mysql-connector-java-8.0.25.jar /usr/lib/sqoop/lib/”

“ Mysql\_secure\_installation “ : To setup MariaDB

```
[hadoop@ip-172-31-7-137 ~]$ cd mysql-connector-java-8.0.25/
[hadoop@ip-172-31-7-137 mysql-connector-java-8.0.25]$ sudo cp mysql-connector-java-8.0.25.jar /usr/lib/sqoop/lib/
[hadoop@ip-172-31-7-137 mysql-connector-java-8.0.25]$ mysql_secure_installation

NOTE: RUNNING ALL PARTS OF THIS SCRIPT IS RECOMMENDED FOR ALL MariaDB
SERVERS IN PRODUCTION USE! PLEASE READ EACH STEP CAREFULLY!

In order to log into MariaDB to secure it, we'll need the current
password for the root user. If you've just installed MariaDB, and
you haven't set the root password yet, the password will be blank,
so you should just press enter here.

Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the MariaDB
root user without the proper authorisation.

Set root password? [Y/n] y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables..
... Success!

By default, a MariaDB installation has an anonymous user, allowing anyone
to log into MariaDB without having to have a user account created for
them. This is intended only for testing, and to make the installation
go a bit smoother. You should remove them before moving into a
production environment.

Remove anonymous users? [Y/n] y
... Success!

Normally, root should only be allowed to connect from 'localhost'. This
ensures that someone cannot guess at the root password from the network.

Disallow root login remotely? [Y/n] n
... skipping.

By default, MariaDB comes with a database named 'test' that anyone can
access. This is also intended only for testing, and should be removed
before moving into a production environment.

Remove test database and access to it? [Y/n] y
- Dropping test database...
... Success!
- Removing privileges on test database...
... Success!
```

## 2: Ingesting data from MySQL RDS to HBase table

Note: --hbase-create-table : creates an HBase table if it does not exist

```
sqoop import \
--connect jdbc:mysql://database-1.cn4wwwocm81j.us-east-1.rds.amazonaws.com/taxi_yellow \
\
--username admin \
--password 7087507642 \
--table trips \
--hbase-table trip_records_hbase_new \
--column-family col1 \
--hbase-create-table \
--hbase-row-key tpep_pickup_datetime,tpep_dropoff_datetime \
--hbase-bulkload \
--split-by payment_type
```

### Code Explanation:

let's break down the SQoop command step by step:

1. **sqoop import**: This is the command to initiate the data import process using SQoop.

2. **--connect**

**jdbc:mysql://database-1.cn4wwwocm81j.us-east-1.rds.amazonaws.com/taxi\_yellow**: This specifies the JDBC connection string to connect to the MySQL database located at the provided endpoint "database-1.cn4wwwocm81j.us-east-1.rds.amazonaws.com" and selects the "taxi\_yellow" database.

3. **--username admin**: Specifies the username to authenticate with the MySQL database, in this case, "admin".

4. **--password 7087507642**: Specifies the password for the provided username, "admin".

5. **--table trips**: Specifies the name of the MySQL table from which data will be imported, in this case, "trips".

6. **--hbase-table trip\_records\_hbase\_new**: Specifies the name of the HBase table where the data will be imported, in this case, "trip\_records\_hbase\_new".
7. **--column-family col1**: Specifies the column family in the HBase table where the data will be stored, in this case, "col1".
8. **--hbase-create-table**: Indicates that the HBase table specified (--hbase-table) should be created if it does not exist.
9. **--hbase-row-key tpep\_pickup\_datetime,tpep\_dropoff\_datetime**: Specifies the columns from the MySQL table that will be used as the row key in the HBase table. In this case, the row key will be composed of values from the columns "tpep\_pickup\_datetime" and "tpep\_dropoff\_datetime".
10. **--hbase-bulkload**: Indicates that the data will be loaded into HBase using bulk loading, which can improve performance for large datasets.
11. **--split-by payment\_type**: Specifies the column used to split the import into multiple parallel tasks. In this case, the import will be split based on the "payment\_type" column.

This command essentially connects to a MySQL database, selects a specific table, and imports its data into an HBase table with specified configurations.

```

[root@ip-172-31-6-200 mysql-connector-java-8.0.25]# sqoop import --connect jdbc:mysql://database-1.cn4wwocm8lj.us-east-1.rds.amazonaws.com/yellow_taxi \
> --username admin \
> --password 7087507642 \
> --table trip_records \
> --hbase-table trip_records_hbase \
> --column-family colfam1 \
> --hbase-create-table \
> --hbase-row-key tpep_pickup_datetime,tpep_dropoff_datetime \
> --hbase-bulkload \
> --split-by payment_type
Warning: /usr/lib/sqoop/../hcatalog does not exist! HCatalog jobs will fail.
Please set $HCAT_HOME to the root of your HCatalog installation.
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
24/03/03 13:32:56 INFO sqoop.Sqoop: Running Sqoop version: 1.4.7
SLF4J: Class path contains multiple SLF4J bindings.
SLF4J: Found binding in [jar:file:/usr/lib/hadoop/lib/slf4j-log4j12-1.7.10.jar!/org/slf4j/impl/StaticLoggerBinder.class]
SLF4J: Found binding in [jar:file:/usr/share/aws/redshift/jdbc/redshift-jdbc42-1.2.37.1061.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]
24/03/03 13:32:56 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
24/03/03 13:32:56 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
24/03/03 13:32:56 INFO tool.CodeGenTool: Beginning code generation
Loading class 'com.mysql.jdbc.Driver'. This is deprecated. The new driver class is 'com.mysql.cj.jdbc.Driver'. The driver is automatically registered via the SPI and manual loading of the driver class is generally unnecessary.
24/03/03 13:32:57 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM 'trip_records' AS t LIMIT 1
24/03/03 13:32:57 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM 'trip_records' AS t LIMIT 1
24/03/03 13:32:57 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/247509e59f32cb435f3a0ff0bac30e7e/trip_records.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
24/03/03 13:33:00 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-root/compile/247509e59f32cb435f3a0ff0bac30e7e/trip_records.jar
24/03/03 13:33:00 WARN manager.MySQLManager: It looks like you are importing from mysql.
24/03/03 13:33:00 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
24/03/03 13:33:00 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
24/03/03 13:33:00 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
24/03/03 13:33:00 INFO mapreduce.ImportJobBase: Beginning import of trip_records
24/03/03 13:33:00 INFO Configuration.deprecation: mapred.jar is deprecated. Instead, use mapreduce.job.jar
24/03/03 13:33:00 INFO Configuration.deprecation: mapred.map.tasks is deprecated. Instead, use mapreduce.job.maps
24/03/03 13:33:03 WARN mapreduce.TableMapReduceUtil: The addDependencyJars(Configuration, Class<?>...) method has been deprecated since it is easy to use incorrectly. Most users should rely
on addDependencyJars(Job) instead. See HBASE-8386 for more details.
24/03/03 13:33:03 WARN mapreduce.TableMapReduceUtil: The addDependencyJars(Configuration, Class<?>...) method has been deprecated since it is easy to use incorrectly. Most users should rely
on addDependencyJars(Job) instead. See HBASE-8386 for more details.
24/03/03 13:33:03 INFO zlib.ZlibFactory: Successfully loaded & initialized native-zlib library
24/03/03 13:33:03 INFO compress.CodecPool: Got brand-new compressor [.deflate]
24/03/03 13:33:03 INFO client.RMProxy: Connecting to ResourceManager at ip-172-31-6-200.ec2.internal/172.31.6.200:8032
24/03/03 13:33:08 INFO db.DBInputFormat: Using read committed transaction isolation
24/03/03 13:33:08 INFO db.DataDrivenDBInputFormat: BoundingValsQuery: SELECT MIN('payment_type'), MAX('payment_type') FROM 'trip_records'
24/03/03 13:34:00 INFO db.IntegerSplitter: Split size: 1; Num splits: 4 from: 1 to: 5
24/03/03 13:34:00 INFO mapreduce.JobSubmitter: number of splits:5

24/03/03 13:33:00 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-root/compile/247509e59f32cb435f3a0ff0bac30e7e/trip_records.jar
24/03/03 13:33:00 WARN manager.MySQLManager: It looks like you are importing from mysql.
24/03/03 13:33:00 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
24/03/03 13:33:00 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
24/03/03 13:33:00 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
24/03/03 13:33:00 INFO mapreduce.ImportJobBase: Beginning import of trip_records
24/03/03 13:33:00 INFO Configuration.deprecation: mapred.jar is deprecated. Instead, use mapreduce.job.jar
24/03/03 13:33:00 INFO Configuration.deprecation: mapred.map.tasks is deprecated. Instead, use mapreduce.job.maps
24/03/03 13:33:03 WARN mapreduce.TableMapReduceUtil: The addDependencyJars(Configuration, Class<?>...) method has been deprecated since it is easy to use incorrectly. Most users should rely
on addDependencyJars(Job) instead. See HBASE-8386 for more details.
24/03/03 13:33:03 WARN mapreduce.TableMapReduceUtil: The addDependencyJars(Configuration, Class<?>...) method has been deprecated since it is easy to use incorrectly. Most users should rely
on addDependencyJars(Job) instead. See HBASE-8386 for more details.
24/03/03 13:33:03 INFO zlib.ZlibFactory: Successfully loaded & initialized native-zlib library
24/03/03 13:33:03 INFO compress.CodecPool: Got brand-new compressor [.deflate]
24/03/03 13:33:03 INFO client.RMProxy: Connecting to ResourceManager at ip-172-31-6-200.ec2.internal/172.31.6.200:8032
24/03/03 13:33:08 INFO db.DBInputFormat: Using read committed transaction isolation
24/03/03 13:33:08 INFO db.DataDrivenDBInputFormat: BoundingValsQuery: SELECT MIN('payment_type'), MAX('payment_type') FROM 'trip_records'
24/03/03 13:34:00 INFO db.IntegerSplitter: Split size: 1; Num splits: 4 from: 1 to: 5
24/03/03 13:34:00 INFO mapreduce.JobSubmitter: number of splits:5
24/03/03 13:34:00 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1709466620659_0002
24/03/03 13:34:00 INFO Impl.YarnClientImpl: Submitted application application_1709466620659_0002
24/03/03 13:34:00 INFO mapreduce.Job: The url to track the job: http://ip-172-31-6-200.ec2.internal:20888/proxy/application_1709466620659_0002/
24/03/03 13:34:00 INFO mapreduce.Job: Running job: job_1709466620659_0002
24/03/03 13:34:09 INFO mapreduce.Job: Job job_1709466620659_0002 running in uber mode : false
24/03/03 13:34:09 INFO mapreduce.Job: map 0% reduce 0%

```

```
24/03/03 14:15:48 INFO mapreduce.Job: Job job_1709466620659_0002 completed successfully
24/03/03 14:15:48 INFO mapreduce.Job: Counters: 50
```

#### File System Counters

```
FILE: Number of bytes read=13215104659
FILE: Number of bytes written=18164974513
FILE: Number of read operations=0
FILE: Number of large read operations=0
FILE: Number of write operations=0
HDFS: Number of bytes read=591
HDFS: Number of bytes written=25190684400
HDFS: Number of read operations=19
HDFS: Number of large read operations=0
HDFS: Number of write operations=5
```

#### Job Counters

```
Killed map tasks=1
Launched map tasks=5
Launched reduce tasks=1
Other local map tasks=5
Total time spent by all maps in occupied slots (ms)=85053504
Total time spent by all reduces in occupied slots (ms)=138253536
Total time spent by all map tasks (ms)=1771948
Total time spent by all reduce tasks (ms)=1440141
Total vcore-milliseconds taken by all map tasks=1771948
Total vcore-milliseconds taken by all reduce tasks=1440141
Total megabyte-milliseconds taken by all map tasks=2721712128
Total megabyte-milliseconds taken by all reduce tasks=4424113152
```

#### Map-Reduce Framework

```
Map input records=18880595
Map output records=302089520
Map output bytes=42389675891
Map output materialized bytes=4948503259
Input split bytes=591
Combine input records=0
Combine output records=0
Reduce input groups=18842048
Reduce shuffle bytes=4948503259
Reduce input records=302089520
Reduce output records=301472768
Spilled Records=1108364028
Shuffled Maps =5
Failed Shuffles=0
Merged Map outputs=5
GC time elapsed (ms)=18760
CPU time spent (ms)=3244240
Physical memory (bytes) snapshot=4895903744
Virtual memory (bytes) snapshot=21252292608
Total committed heap usage (bytes)=4115136512
```

#### Shuffle Errors

```
BAD_ID=0
CONNECTION=0
```

#### Map-Reduce Framework

```
Map input records=18880595
Map output records=302089520
Map output bytes=42389675891
Map output materialized bytes=4948503259
Input split bytes=591
Combine input records=0
Combine output records=0
Reduce input groups=18842048
Reduce shuffle bytes=4948503259
Reduce input records=302089520
Reduce output records=301472768
Spilled Records=1108364028
Shuffled Maps =5
Failed Shuffles=0
Merged Map outputs=5
GC time elapsed (ms)=18760
CPU time spent (ms)=3244240
Physical memory (bytes) snapshot=4895903744
Virtual memory (bytes) snapshot=21252292608
Total committed heap usage (bytes)=4115136512
```

#### Shuffle Errors

```
BAD_ID=0
CONNECTION=0
IO_ERROR=0
WRONG_LENGTH=0
WRONG_MAP=0
WRONG_REDUCE=0
```

#### File Input Format Counters

```
Bytes Read=0
```

#### File Output Format Counters

```
Bytes Written=25190684400
```

```
4/03/03 14:15:48 INFO mapreduce.ImportJobBase: Transferred 23.4607 GB in 2,564.7358 seconds (9.3669 MB/sec)
```

```
4/03/03 14:15:48 INFO mapreduce.ImportJobBase: Retrieved 302089520 records.
```

```
4/03/03 14:15:48 WARN mapreduce.LoadIncrementalHFiles: managed connection cannot be used for bulkload. Creating unmanaged connection.
```

```
4/03/03 14:15:48 WARN mapreduce.LoadIncrementalHFiles: Skipping non-directory hdfs://ip-172-31-6-200.ec2.internal:8020/user/root/trip_records/_SUCCESS
```

```
4/03/03 14:15:48 INFO impl.MetricsConfig: loaded properties from hadoop-metrics2-hbase.properties
```

```
4/03/03 14:15:48 INFO impl.MetricsSystemImpl: Scheduled Metric snapshot period at 10 second(s).
```

```
4/03/03 14:15:48 INFO impl.MetricsSystemImpl: Hbase metric system started
```

```
4/03/03 14:15:48 WARN mapreduce.LoadIncrementalHFiles: Trying to bulk load hfile hdfs://ip-172-31-6-200.ec2.internal:8020/user/root/trip_records/colfam1/35f12e128f6c4ea9e0ealde0812ad35 wi
```

```
h size: 11170136743 bytes can be problematic as it may lead to oversplitting.
```

```
4/03/03 14:15:48 WARN mapreduce.LoadIncrementalHFiles: Trying to bulk load hfile hdfs://ip-172-31-6-200.ec2.internal:8020/user/root/trip_records/colfam1/59ded78b40124fdca4376f30435c6643 wi
```

```
h size: 11170130983 bytes can be problematic as it may lead to oversplitting.
```

```
4/03/03 14:15:49 INFO Configuration.deprecation: hbase.offheapcache.minblocksize is deprecated. Instead, use hbase.blockcache.minblocksize
```

## Checking in hBase:

Commands :

|                       |                           |
|-----------------------|---------------------------|
| 1: To get into hbase  | hbase shell               |
| 2: Tables in hbase:   | List                      |
| 3: Count data copied: | count 'trip_records_hbase |

```
hadoop@ip-172-31-6-200:~$
 at java.lang.reflect.Method.invoke(Method.java:498)
 at org.jruby.javasupport.JavaMethod.invokeDirectWithExceptionHandling(JavaMethod.java:450)
 at org.jruby.javasupport.JavaMethod.invokeStaticDirect(JavaMethod.java:362)
 at org.jruby.java.invokers.StaticMethodInvoker.call(StaticMethodInvoker.java:58)
 at org.jruby.runtime.callsite.CachingCallSite.cacheAndCall(CachingCallSite.java:312)
 at org.jruby.runtime.callsite.CachingCallSite.call(CachingCallSite.java:169)
 at usr.lib.hbase.bin.hirb._file_ (/usr/lib/hbase/bin/hirb.rb:129)
 at usr.lib.hbase.bin.hirb.load(/usr/lib/hbase/bin/hirb.rb)
 at org.jruby.Ruby.runScript(Ruby.java:697)
 at org.jruby.Ruby.runScript(Ruby.java:690)
 at org.jruby.Ruby.runNormally(Ruby.java:597)
 at org.jruby.Ruby.runFromMain(Ruby.java:446)
 at org.jruby.Main.doRunFromMain(Main.java:369)
 at org.jruby.Main.internalRun(Main.java:258)
 at org.jruby.Main.run(Main.java:224)
 at org.jruby.Main.run(Main.java:208)
 at org.jruby.Main.main(Main.java:188)
log4j:ERROR Either File or DatePattern options are not set for appender [ORFAS].
HBase Shell
Use "help" to get list of supported commands.
Use "exit" to quit this interactive shell.
Version 1.4.13, rUnknown, Fri Apr 17 15:18:24 UTC 2020

hbase(main):001:0> list
TABLE
trip_records_hbase
1 row(s) in 0.2450 seconds

=> ["trip_records_hbase"]
hbase(main):002:0> describe 'trip_records_hbase'
Table trip_records_hbase is ENABLED
trip_records_hbase
COLUMN FAMILIES DESCRIPTION
(NAME => 'colfam1', BLOOMFILTER => 'ROW', VERSIONS => '1', IN MEMORY => 'false',
KEEP DELETED CELLS => 'FALSE', DATA BLOCK ENCODING => 'NONE', TTL => 'FOREVER',
COMPRESSION => 'NONE', MIN VERSIONS => '0', BLOCKCACHE => 'true', BLOCKSIZE =>
'65536', REPLICATION SCOPE => '0')
1 row(s) in 0.1110 seconds

hbase(main):003:0> count 'trip_records_hbase'
Current count: 1000, row: 2017-01-01 00:06:50_2017-01-01 00:14:24
Current count: 2000, row: 2017-01-01 00:10:49_2017-01-01 00:32:23
Current count: 3000, row: 2017-01-01 00:14:09_2017-01-01 00:15:47
Current count: 4000, row: 2017-01-01 00:17:05_2017-01-01 00:33:41
Current count: 5000, row: 2017-01-01 00:19:44_2017-01-01 00:29:27
Current count: 6000, row: 2017-01-01 00:22:20_2017-01-01 00:35:07
Current count: 7000, row: 2017-01-01 00:24:57_2017-01-01 00:27:01
```

Current count: 3422000, row: 2017-01-12 15:22:23 2017-01-12 15:39:31  
Current count: 3423000, row: 2017-01-12 15:26:01 2017-01-12 16:10:22  
Current count: 3424000, row: 2017-01-12 15:29:42 2017-01-12 15:32:58  
Current count: 3425000, row: 2017-01-12 15:33:30 2017-01-12 15:33:53  
Current count: 3426000, row: 2017-01-12 15:36:53 2017-01-12 15:48:29  
Current count: 3427000, row: 2017-01-12 15:40:28 2017-01-12 15:47:57  
Current count: 3428000, row: 2017-01-12 15:44:04 2017-01-12 16:48:00  
Current count: 3429000, row: 2017-01-12 15:47:41 2017-01-12 16:06:58  
Current count: 3430000, row: 2017-01-12 15:51:15 2017-01-12 16:00:46  
Current count: 3431000, row: 2017-01-12 15:55:01 2017-01-12 16:15:40  
Current count: 3432000, row: 2017-01-12 15:58:52 2017-01-12 16:05:10  
Current count: 3433000, row: 2017-01-12 16:02:30 2017-01-12 16:09:34  
Current count: 3434000, row: 2017-01-12 16:06:02 2017-01-12 16:12:21  
Current count: 3435000, row: 2017-01-12 16:09:48 2017-01-12 16:21:23  
Current count: 3436000, row: 2017-01-12 16:13:19 2017-01-12 16:31:49  
Current count: 3437000, row: 2017-01-12 16:16:54 2017-01-12 16:28:45  
Current count: 3438000, row: 2017-01-12 16:20:51 2017-01-12 16:38:21  
Current count: 3439000, row: 2017-01-12 16:25:05 2017-01-12 16:27:41  
Current count: 3440000, row: 2017-01-12 16:29:15 2017-01-12 16:33:17  
Current count: 3441000, row: 2017-01-12 16:33:14 2017-01-12 17:07:50  
Current count: 3442000, row: 2017-01-12 16:37:11 2017-01-12 16:51:58  
Current count: 3443000, row: 2017-01-12 16:41:11 2017-01-12 16:52:52  
Current count: 3444000, row: 2017-01-12 16:44:49 2017-01-12 17:02:07  
Current count: 3445000, row: 2017-01-12 16:48:45 2017-01-12 16:53:41  
Current count: 3446000, row: 2017-01-12 16:52:50 2017-01-12 16:57:11  
Current count: 3447000, row: 2017-01-12 16:56:51 2017-01-12 17:31:51  
Current count: 3448000, row: 2017-01-12 17:00:41 2017-01-12 17:11:14  
Current count: 3449000, row: 2017-01-12 17:04:28 2017-01-12 17:14:49  
Current count: 3450000, row: 2017-01-12 17:08:15 2017-01-12 17:11:59  
Current count: 3451000, row: 2017-01-12 17:11:49 2017-01-12 17:38:05  
Current count: 3452000, row: 2017-01-12 17:15:21 2017-01-12 17:20:37  
Current count: 3453000, row: 2017-01-12 17:18:45 2017-01-12 17:37:57  
Current count: 3454000, row: 2017-01-12 17:22:25 2017-01-12 17:30:08  
Current count: 3455000, row: 2017-01-12 17:25:52 2017-01-12 17:41:40  
Current count: 3456000, row: 2017-01-12 17:29:17 2017-01-12 17:40:01  
Current count: 3457000, row: 2017-01-12 17:32:49 2017-01-12 17:40:38  
Current count: 3458000, row: 2017-01-12 17:35:58 2017-01-12 17:40:12  
Current count: 3459000, row: 2017-01-12 17:39:05 2017-01-12 17:46:34  
Current count: 3460000, row: 2017-01-12 17:42:18 2017-01-12 18:03:30  
Current count: 3461000, row: 2017-01-12 17:45:27 2017-01-12 18:14:01  
Current count: 3462000, row: 2017-01-12 17:48:37 2017-01-12 18:02:07  
Current count: 3463000, row: 2017-01-12 17:51:52 2017-01-12 17:55:30  
Current count: 3464000, row: 2017-01-12 17:54:59 2017-01-12 18:04:14  
Current count: 3465000, row: 2017-01-12 17:58:05 2017-01-12 18:00:52  
Current count: 3466000, row: 2017-01-12 18:01:11 2017-01-12 18:06:13  
Current count: 3467000, row: 2017-01-12 18:04:18 2017-01-12 18:52:29  
Current count: 3468000, row: 2017-01-12 18:07:13 2017-01-12 18:37:29  
Current count: 3469000, row: 2017-01-12 18:10:05 2017-01-12 18:14:10  
Current count: 3470000, row: 2017-01-12 18:12:58 2017-01-12 18:35:21  
Current count: 3471000, row: 2017-01-12 18:15:50 2017-01-12 18:25:39