Ingestion data into Hbase table from RDS table and using a python script

Step 1: Login to EMR cluster.

Command:

ssh -i EMR-cluster-key.pem hadoop@ec2-54-236-6-90.compute-1.amazonaws.com

habse shell # To enter Hbase shell

```
[root@ip-172-31-27-81 ~]# hbase shell
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>
```

Step 2: Create a directory named ‘hdfs-map’ under hdfs ‘/user/

Commands:

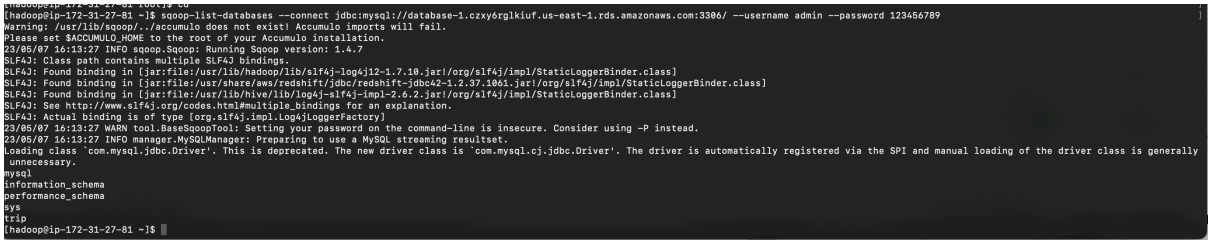
hadoop fs -ls / # to list under / file system
hadoop fs -ls /user # to list under /user
hadoop fs -mkdir /user/hdfs-map # To create directory under /user
hadoop fs -ls /user/hdfs-map # to list under /user/hdfs-map

```
[hadoop@ip-172-31-27-81 root]$
[hadoop@ip-172-31-27-81 root]$
[hadoop@ip-172-31-27-81 root]$ hadoop fs -ls /
Found 4 items
drwxr-xr-x   - hdfs hadoop          0 2023-05-07 15:26 /apps
drwxrwxrwt   - hdfs hadoop          0 2023-05-07 15:26 /tmp
drwxr-xr-x   - hdfs hadoop          0 2023-05-07 15:56 /user
drwxr-xr-x   - hdfs hadoop          0 2023-05-07 15:26 /var
[hadoop@ip-172-31-27-81 root]$
[hadoop@ip-172-31-27-81 root]$ hadoop fs -ls /user
Found 5 items
drwxrwxrwx   - hadoop hadoop          0 2023-05-07 15:26 /user/hadoop
drwxrwxr-x   - hbase  hbase           0 2023-05-07 15:27 /user/hbase
drwxr-xr-x   - mapred mapred          0 2023-05-07 15:26 /user/history
drwxrwxrwx   - hdfs   hadoop          0 2023-05-07 15:26 /user/hive
drwxrwxrwx   - root   hadoop          0 2023-05-07 15:26 /user/root
[hadoop@ip-172-31-27-81 root]$ hadoop fs -mkdir /user/hdfs-map
[hadoop@ip-172-31-27-81 root]$ hadoop fs -ls /user/hdfs-map
[hadoop@ip-172-31-27-81 root]$
```

Step 3: Checking the connectivity between RDS and EMR cluster using jdbc and listing all the databases in RDS.

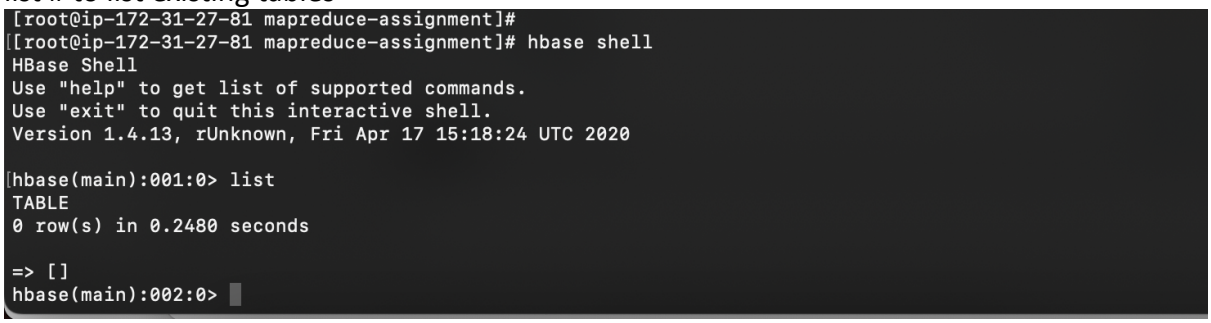
Command:

sqoop-list-databases --connect jdbc:mysql://database-1.czxy6rglkiuf.us-east-1.rds.amazonaws.com:3306/ --username admin --password 123456789



Step 4: Listing the tables in Hbase.

Command:
list # to list existing tables



Step 5: Importing data from RDS table to EMR Hbase.

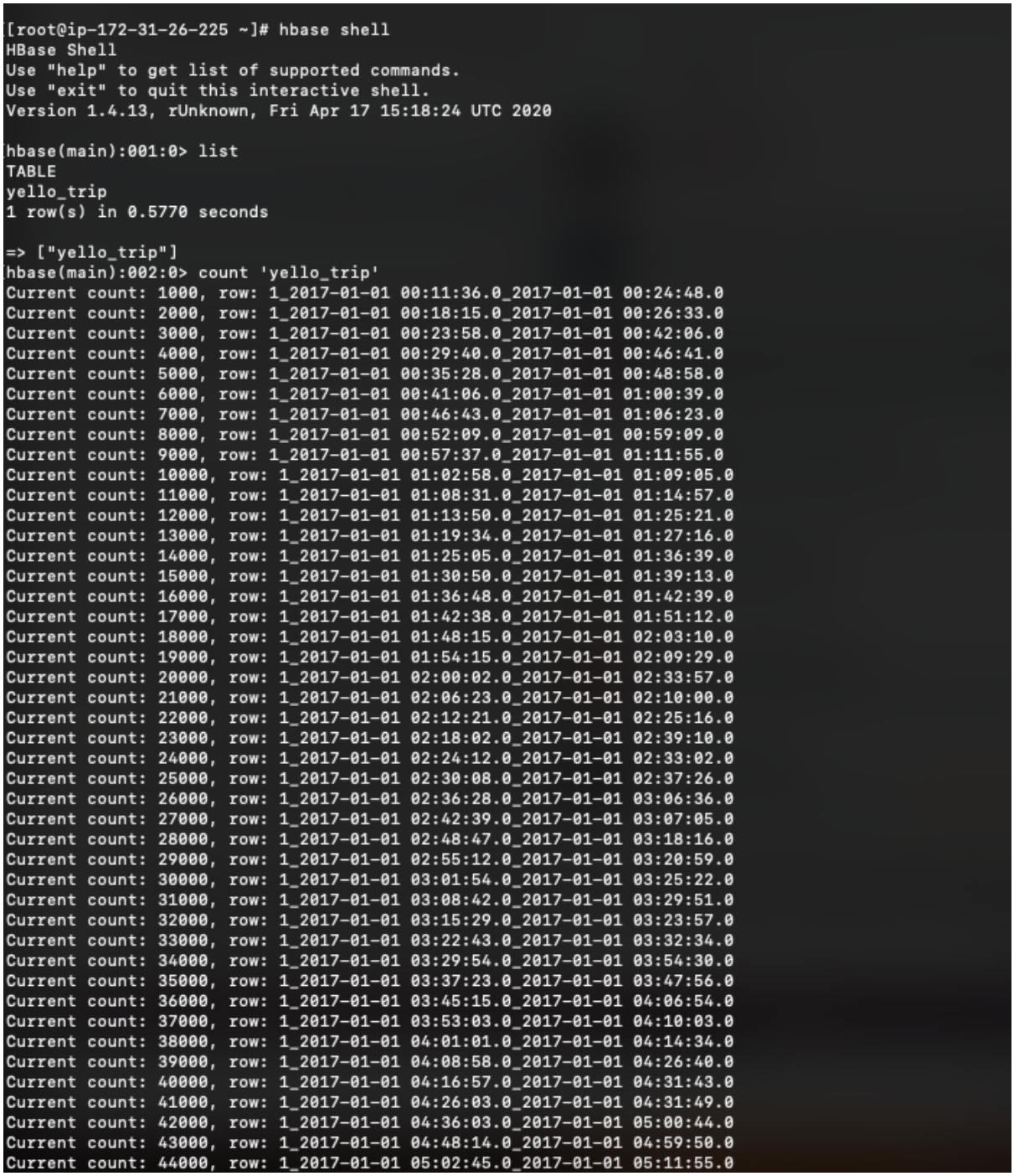
Command:
sqoop import --connect "jdbc:mysql://database-1.czxy6rglkiuf.us-east-1.rds.amazonaws.com:3306/trip" --username admin --password 123456789 --table yello --columns "VendorID,tpep_pickup_datetime,tpep_dropoff_datetime,passenger_count,trip_distance,RatecodeID,store_and_fwd_flag,PULocationID,DOLocationID,payment_type,fare_amount,extra,mta_tax,tip_amount,tolls_amount,improvement_surcharge,total_amount,Airport_fee" --hbase-create-table --hbase-table yello --column-family trip_details --hbase-row-key VendorID,tpep_pickup_datetime,tpep_dropoff_datetime --split-by tpep_dropoff_datetime -m 8

```
[root@ip-172-31-16-199 mysql-connector-java-8.0.25]#
[root@ip-172-31-16-199 mysql-connector-java-8.0.25]#
[root@ip-172-31-16-199 mysql-connector-java-8.0.25]# sqoop import --connect "jdbc:mysql://database-1.czxy6rglkiuf.us-east-1.rds.amazonaws.com:3306/trip" --username admin --password 123456789 --table yello --colum
ns "VendorID,tpep_pickup_datetime,tpep_dropoff_datetime,passenger_count,trip_distance,RatecodeID,store_and_fwd_flag,PULocationID,DOLocationID,payment_type,fare_amount,extra,mta_tax,tip_amount,tolls_amount,improve
ment_surcharge,total_amount,Airport_fee" --hbase-create-table --hbase-table yello --column-family trip_details --hbase-row-key VendorID,tpep_pickup_datetime,tpep_dropoff_datetime --split-by tpep_dropoff_datetime
-m 8
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
23/05/08 06:20:09 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: Found binding in [jar:file:/usr/lib/hive/lib/log4j-slf4j-impl-2.6.2.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]
23/05/08 06:20:09 WARN tool.BaseSqoopTool: Setting your password on the command-line is insecure. Consider using -P instead.
23/05/08 06:20:09 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
23/05/08 06:20:09 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 generall
y unnecessary.
23/05/08 06:20:09 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `yello` AS t LIMIT 1
23/05/08 06:20:09 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `yello` AS t LIMIT 1
23/05/08 06:20:09 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-root/compile/498abdf48ce8e34fc608db326f738b10/yello.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
23/05/08 06:20:13 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-root/compile/498abdf48ce8e34fc608db326f738b10/yello.jar
23/05/08 06:20:13 WARN manager.MySQLManager: It looks like you are importing from mysql.
23/05/08 06:20:13 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
23/05/08 06:20:13 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
23/05/08 06:20:13 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
23/05/08 06:20:13 INFO mapreduce.ImportJobBase: Beginning import of yello
23/05/08 06:20:14 INFO Configuration.deprecation: mapred.jar is deprecated. Instead, use mapreduce.job.jar
23/05/08 06:20:14 INFO Configuration.deprecation: mapred.map.tasks is deprecated. Instead, use mapreduce.job.maps
23/05/08 06:20:17 INFO mapreduce.HBaseImportJob: Creating missing HBase table yello
23/05/08 06:20:20 WARN mapreduce.TableMapReduceUtil: The addDependencyJars(Configuration, Class<?>...) method has been deprecated since it is easy to use incorrectly. Most users should rely on addDependencyJars(J
ob) instead. See HBASE-8386 for more details.
23/05/08 06:20:20 INFO client.RMProxy: Connecting to ResourceManager at ip-172-31-16-199.ec2.internal/172.31.16.199:8032
23/05/08 06:20:32 INFO db.DBInputFormat: Using read commited transaction isolation
23/05/08 06:20:32 INFO db.DataDrivenDBInputFormat: BoundingValsQuery: SELECT MIN(`tpep_dropoff_datetime`), MAX(`tpep_dropoff_datetime`) FROM `yello`
23/05/08 06:21:13 INFO db.IntegerSplitter: Split size: 877977625; Num splits: 8 from: 1481388013000 to: 1488411834000
23/05/08 06:21:13 INFO mapreduce.JobSubmitter: number of splits:8
23/05/08 06:21:13 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1683526383858_0001
23/05/08 06:21:14 INFO impl.YarnClientImpl: Submitted application application_1683526383858_0001
23/05/08 06:21:14 INFO mapreduce.Job: The url to track the job: http://ip-172-31-16-199.ec2.internal:20888/proxy/application_1683526383858_0001/
23/05/08 06:21:14 INFO mapreduce.Job: Running job: job_1683526383858_0001
23/05/08 06:21:24 INFO mapreduce.Job: Job job_1683526383858_0001 running in uber mode : false
23/05/08 06:21:24 INFO mapreduce.Job:  map 0% reduce 0%
23/05/08 06:22:29 INFO mapreduce.Job:  map 13% reduce 0%
23/05/08 06:22:31 INFO mapreduce.Job:  map 25% reduce 0%

23/05/08 06:40:35 INFO mapreduce.Job:  map 38% reduce 0%
```

```
Warning: /usr/lib/sqoop/../accumulo does not exist! Accumulo imports will fail.
Please set $ACCUMULO_HOME to the root of your Accumulo installation.
23/05/08 16:23:20 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: Found binding in [jar:file:/usr/lib/hive/lib/log4j-slf4j-impl-2.6.2.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]
Enter password:
23/05/08 16:23:26 INFO manager.MySQLManager: Preparing to use a MySQL streaming resultset.
23/05/08 16:23:26 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.
23/05/08 16:23:27 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `yello` AS t LIMIT 1
23/05/08 16:23:27 INFO manager.SqlManager: Executing SQL statement: SELECT t.* FROM `yello` AS t LIMIT 1
23/05/08 16:23:27 INFO orm.CompilationManager: HADOOP_MAPRED_HOME is /usr/lib/hadoop-mapreduce
Note: /tmp/sqoop-hadoop/compile/795c4cb577bddff6d97ecabe53772890/yello.java uses or overrides a deprecated API.
Note: Recompile with -Xlint:deprecation for details.
23/05/08 16:23:30 INFO orm.CompilationManager: Writing jar file: /tmp/sqoop-hadoop/compile/795c4cb577bddff6d97ecabe53772890/yello.jar
23/05/08 16:23:30 WARN manager.MySQLManager: It looks like you are importing from mysql.
23/05/08 16:23:30 WARN manager.MySQLManager: This transfer can be faster! Use the --direct
23/05/08 16:23:30 WARN manager.MySQLManager: option to exercise a MySQL-specific fast path.
23/05/08 16:23:30 INFO manager.MySQLManager: Setting zero DATETIME behavior to convertToNull (mysql)
23/05/08 16:23:30 INFO mapreduce.ImportJobBase: Beginning import of yello
23/05/08 16:23:30 INFO Configuration.deprecation: mapred.jar is deprecated. Instead, use mapreduce.job.jar
23/05/08 16:23:31 INFO Configuration.deprecation: mapred.map.tasks is deprecated. Instead, use mapreduce.job.maps
23/05/08 16:23:31 INFO client.RMProxy: Connecting to ResourceManager at ip-172-31-19-244.ec2.internal/172.31.19.244:8032
23/05/08 16:23:37 INFO db.DBInputFormat: Using read commited transaction isolation
23/05/08 16:23:37 INFO db.DataDrivenDBInputFormat: BoundingValsQuery: SELECT MIN(`tpep_dropoff_datetime`), MAX(`tpep_dropoff_datetime`) FROM `yello`
23/05/08 16:24:16 INFO db.IntegerSplitter: Split size: 877977625; Num splits: 8 from: 1481388013000 to: 1488411834000
23/05/08 16:24:16 INFO mapreduce.JobSubmitter: number of splits:8
23/05/08 16:24:16 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1683561592377_0003
23/05/08 16:24:16 INFO impl.YarnClientImpl: Submitted application application_1683561592377_0003
23/05/08 16:24:16 INFO mapreduce.Job: The url to track the job: http://ip-172-31-19-244.ec2.internal:20888/proxy/application_1683561592377_0003/
23/05/08 16:24:16 INFO mapreduce.Job: Running job: job_1683561592377_0003
23/05/08 16:24:25 INFO mapreduce.Job: Job job_1683561592377_0003 running in uber mode : false
23/05/08 16:24:25 INFO mapreduce.Job:  map 0% reduce 0%
23/05/08 16:26:16 INFO mapreduce.Job:  map 13% reduce 0%
23/05/08 16:26:17 INFO mapreduce.Job:  map 25% reduce 0%
23/05/08 16:26:45 INFO mapreduce.Job:  map 38% reduce 0%
23/05/08 16:26:47 INFO mapreduce.Job:  map 50% reduce 0%
23/05/08 16:28:38 INFO mapreduce.Job:  map 63% reduce 0%
23/05/08 16:28:44 INFO mapreduce.Job:  map 75% reduce 0%
23/05/08 16:29:25 INFO mapreduce.Job:  map 88% reduce 0%
23/05/08 16:29:28 INFO mapreduce.Job:  map 100% reduce 0%
23/05/08 16:29:29 INFO mapreduce.Job: Job job_1683561592377_0003 completed successfully
23/05/08 16:29:29 INFO mapreduce.Job: Counters: 31
        File System Counters
                FILE: Number of bytes read=0
                FILE: Number of bytes written=1517248
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=1461
                HDFS: Number of bytes written=1890799616
                HDFS: Number of read operations=32
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=16
        Job Counters
                Killed map tasks=3
                Launched map tasks=11
                Other local map tasks=11
                Total time spent by all maps in occupied slots (ms)=56411568
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=1175241
                Total vcore-milliseconds taken by all map tasks=1175241
```

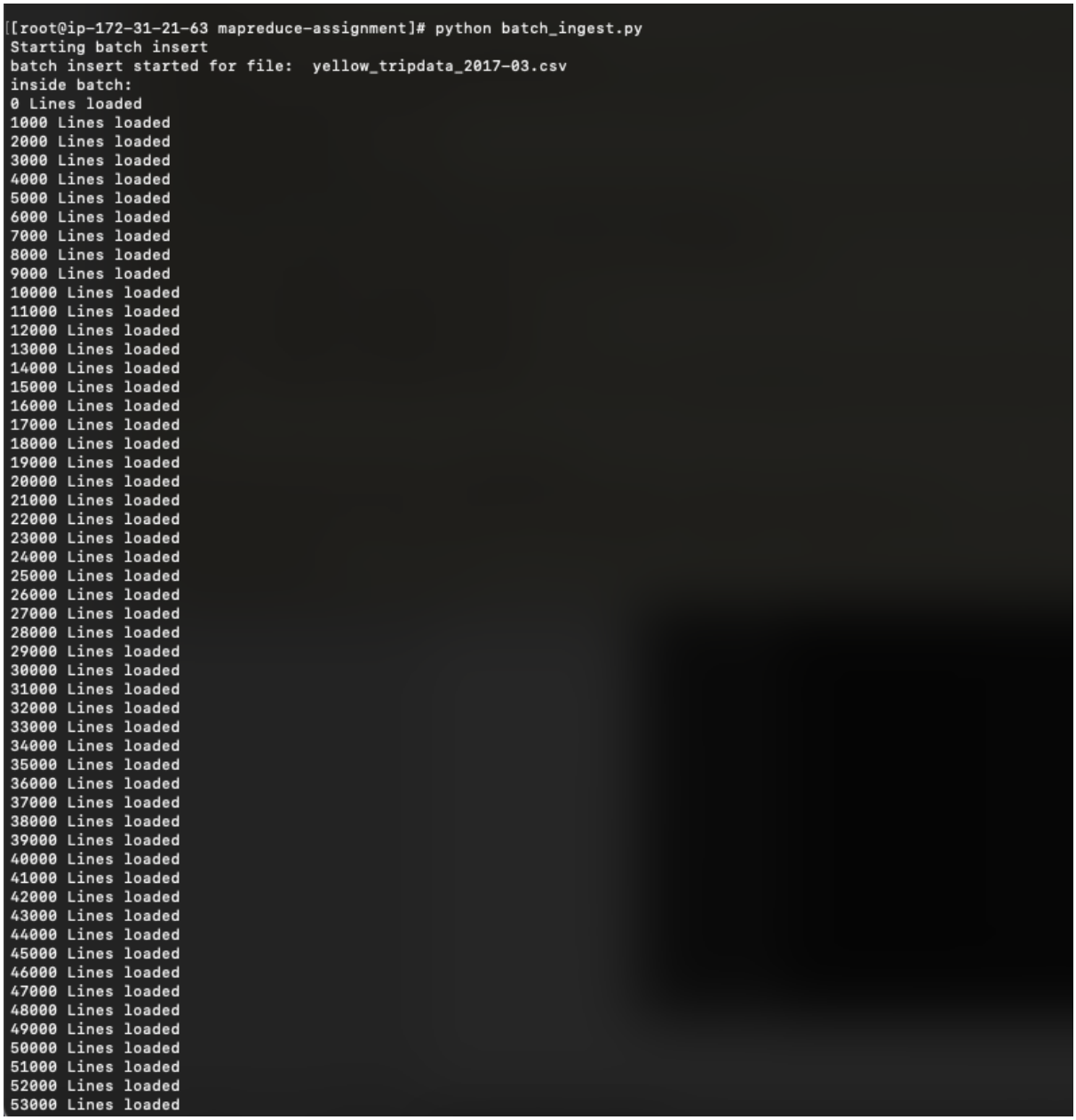
```
23/05/08 16:24:16 INFO impl.YarnClientImpl: Submitted application application_1683561592377_0003
23/05/08 16:24:16 INFO mapreduce.Job: The url to track the job: http://ip-172-31-19-244.ec2.internal:20888/proxy/application_1683561592377_0003/
23/05/08 16:24:16 INFO mapreduce.Job: Running job: job_1683561592377_0003
23/05/08 16:24:25 INFO mapreduce.Job: Job job_1683561592377_0003 running in uber mode : false
23/05/08 16:24:25 INFO mapreduce.Job:  map 0% reduce 0%
23/05/08 16:26:16 INFO mapreduce.Job:  map 13% reduce 0%
23/05/08 16:26:17 INFO mapreduce.Job:  map 25% reduce 0%
23/05/08 16:26:45 INFO mapreduce.Job:  map 38% reduce 0%
23/05/08 16:26:47 INFO mapreduce.Job:  map 50% reduce 0%
23/05/08 16:28:38 INFO mapreduce.Job:  map 63% reduce 0%
23/05/08 16:28:44 INFO mapreduce.Job:  map 75% reduce 0%
23/05/08 16:29:25 INFO mapreduce.Job:  map 88% reduce 0%
23/05/08 16:29:28 INFO mapreduce.Job:  map 100% reduce 0%
23/05/08 16:29:29 INFO mapreduce.Job: Job job_1683561592377_0003 completed successfully
23/05/08 16:29:29 INFO mapreduce.Job: Counters: 31
        File System Counters
                FILE: Number of bytes read=0
                FILE: Number of bytes written=1517248
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=1461
                HDFS: Number of bytes written=1890799616
                HDFS: Number of read operations=32
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=16
        Job Counters
                Killed map tasks=3
                Launched map tasks=11
                Other local map tasks=11
                Total time spent by all maps in occupied slots (ms)=56411568
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=1175241
                Total vcore-milliseconds taken by all map tasks=1175241
                Total megabyte-milliseconds taken by all map tasks=1805170176
        Map-Reduce Framework
                Map input records=18880595
                Map output records=18880595
                Input split bytes=1461
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=2940
                CPU time spent (ms)=241700
                Physical memory (bytes) snapshot=3915235328
                Virtual memory (bytes) snapshot=26396659712
                Total committed heap usage (bytes)=3311403008
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=1890799616
23/05/08 16:29:29 INFO mapreduce.ImportJobBase: Transferred 1.7609 GB in 357.9347 seconds (5.0378 MB/sec)
23/05/08 16:29:29 INFO mapreduce.ImportJobBase: Retrieved 18880595 records.
[hadoop@ip-172-31-19-244 ~]$
```



Step 6: Importing data into hbase table from yellow_tripdata_2017-03.csv and yellow_tripdata_2017-04.csv using python batch file.

Command:

python batch_ingest.py


[[root@ip-172-31-21-63 mapreduce-assignment]# python batch_ingest.py
Starting batch insert
batch insert started for file: yellow_tripdata_2017-03.csv
inside batch:
0 Lines loaded
1000 Lines loaded
2000 Lines loaded
3000 Lines loaded
4000 Lines loaded
5000 Lines loaded
6000 Lines loaded
7000 Lines loaded
8000 Lines loaded
9000 Lines loaded
10000 Lines loaded
11000 Lines loaded
12000 Lines loaded
13000 Lines loaded
14000 Lines loaded
15000 Lines loaded
16000 Lines loaded
17000 Lines loaded
18000 Lines loaded
19000 Lines loaded
20000 Lines loaded
21000 Lines loaded
22000 Lines loaded
23000 Lines loaded
24000 Lines loaded
25000 Lines loaded
26000 Lines loaded
27000 Lines loaded
28000 Lines loaded
29000 Lines loaded
30000 Lines loaded
31000 Lines loaded
32000 Lines loaded
33000 Lines loaded
34000 Lines loaded
35000 Lines loaded
36000 Lines loaded
37000 Lines loaded
38000 Lines loaded
39000 Lines loaded
40000 Lines loaded
41000 Lines loaded
42000 Lines loaded
43000 Lines loaded
44000 Lines loaded
45000 Lines loaded
46000 Lines loaded
47000 Lines loaded
48000 Lines loaded
49000 Lines loaded
50000 Lines loaded
51000 Lines loaded
52000 Lines loaded
53000 Lines loaded