Data Ingestion from RDS to HDFS using Sqoop

After creating an EMR cluster connect to it through putty and then install the MySQL driver in it and then run the below command so that data from RDS is imported to the table SRC_ATM_TRANS in the HDFS.

```
[hadoop@ip-172-31-4-83 mysql-connector-java-8.0.25]$ sudo service mariadb restart
Redirecting to /bin/systemctl restart mariadb.service
[hadoop@ip-172-31-4-83 mysql-connector-java-8.0.25]$ sqoop import \
> --connect jdbc:mysql://upgraddetest.cyaielc9bmnf.us-east-1.rds.amazonaws.com/testdatabase \
> --table SRC_ATM_TRANS \
> --username student --password STUDENT123 \
> --target-dir /user/root/SRC_ATM_TRANS \
> -m 1
```

```
23/06/12 11:15:20 INFO mapreduce.Job: Running job: job_1686567947348_0001
23/06/12 11:15:30 INFO mapreduce.Job: Job job_1686567947348_0001 running in uber mode : false
23/06/12 11:15:30 INFO mapreduce.Job:  map 0% reduce 0%
23/06/12 11:15:59 INFO mapreduce.Job:  map 100% reduce 0%
23/06/12 11:15:59 INFO mapreduce.Job: Job job_1686567947348_0001 completed successfully
23/06/12 11:15:59 INFO mapreduce.Job: Counters: 30
        File System Counters
                FILE: Number of bytes read=0
                FILE: Number of bytes written=189563
                FILE: Number of read operations=0
                FILE: Number of large read operations=0
                FILE: Number of write operations=0
                HDFS: Number of bytes read=87
                HDFS: Number of bytes written=531214815
                HDFS: Number of read operations=4
                HDFS: Number of large read operations=0
                HDFS: Number of write operations=2
        Job Counters
                Launched map tasks=1
                Other local map tasks=1
                Total time spent by all maps in occupied slots (ms)=1270224
                Total time spent by all reduces in occupied slots (ms)=0
                Total time spent by all map tasks (ms)=26463
                Total vcore-milliseconds taken by all map tasks=26463
                Total megabyte-milliseconds taken by all map tasks=40647168
        Map-Reduce Framework
                Map input records=2468572
                Map output records=2468572
                Input split bytes=87
                Spilled Records=0
                Failed Shuffles=0
                Merged Map outputs=0
                GC time elapsed (ms)=213
                CPU time spent (ms)=28920
                Physical memory (bytes) snapshot=610881536
                Virtual memory (bytes) snapshot=3295088640
                Total committed heap usage (bytes)=536870912
        File Input Format Counters
                Bytes Read=0
        File Output Format Counters
                Bytes Written=531214815
23/06/12 11:15:59 INFO mapreduce.ImportJobBase: Transferred 506.6059 MB in 48.8665 seconds (10.3671 MB/sec)
23/06/12 11:15:59 INFO mapreduce.ImportJobBase: Retrieved 2468572 records.
[hadoop@ip-172-31-4-83 mysql-connector-java-8.0.25]$
```

```
[hadoop@ip-172-31-4-83 mysql-connector-java-8.0.25]$ hadoop fs -ls /user/root/SRC_ATM_TRANS
Found 2 items
-rw-r--r--   1 hadoop hadoop          0 2023-06-12 11:15 /user/root/SRC_ATM_TRANS/_SUCCESS
-rw-r--r--   1 hadoop hadoop  531214815 2023-06-12 11:15 /user/root/SRC_ATM_TRANS/part-m-00000
```

After running the above command, we can see that the target directory contains 2 items. The first file is the success file, indicating that the MapReduce job was successful. The second file ‘part-m-00000’ is the

one with all the data we imported. Since we used only one mapper in our import command thus the data is stored in a single file.

```
[hadoop@ip-172-31-4-83 mysql-connector-java-8.0.25]$ hadoop fs -cat /user/root/SRC_ATM_TRANS/part-m-00000
```

By the above command we can see the data present in the part-m-00000 file.