Document 01 – RDS.pdf by Huang-Yin Tso, Arun Santhosh, Vipul Kumar and Gaurav Sharma

First, we can log into MySQL from our local machine:

```
C:\Users\User>cd C:\Program Files\MySQL\MySQL Server 9.0\bin
```

```
C:\Program Files\MySQL\MySQL Server 9.0\bin>mysql -u admin -p -h mysqldb.cuja9ruhz2vz.us-east-1.rds.amazonaws.co
Enter password: ********
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 379
Server version: 8.0.40 Source distribution

Copyright (c) 2000, 2024, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> ^C
mysql>
```

Once we got into RDS by MySQL, we can use MySQL's instructions to create Database (mysqldb) and create table (yellow_tripdata).

```
mysql>CREATE DATABASE mysqldb;

mysql>USE mysqldb;

Database changed

mysql>CREATE TABLE yellow_tripdata (

  VendorID INT,

  tpep_pickup_datetime DATETIME,

  tpep_dropoff_datetime DATETIME,

  passenger_count INT,

  trip_distance FLOAT,

  RatecodeID INT,

  store_and_fwd_flag VARCHAR(1),

  PULocationID INT,

  DOLocationID INT,
```

```
  payment_type INT,
  fare_amount FLOAT,
  extra FLOAT,
  mta_tax FLOAT,
  tip_amount FLOAT,
  tolls_amount FLOAT,
  improvement_surcharge FLOAT,
  total_amount FLOAT
);
```

mysql> SHOW COLUMNS FROM yellow_tripdata;

```
mysql> SHOW COLUMNS FROM yellow_tripdata;
+-----------------------+------------+------+-----+---------+-------+
| Field                 | Type       | Null | Key | Default | Extra |
+-----------------------+------------+------+-----+---------+-------+
| VendorID              | int        | YES  |     | NULL    |       |
| tpep_pickup_datetime  | datetime   | YES  |     | NULL    |       |
| tpep_dropoff_datetime | datetime   | YES  |     | NULL    |       |
| passenger_count       | int        | YES  |     | NULL    |       |
| trip_distance         | float      | YES  |     | NULL    |       |
| RatecodeID            | int        | YES  |     | NULL    |       |
| store_and_fwd_flag    | varchar(1) | YES  |     | NULL    |       |
| PULocationID          | int        | YES  |     | NULL    |       |
| DOLocationID          | int        | YES  |     | NULL    |       |
| payment_type          | int        | YES  |     | NULL    |       |
| fare_amount           | float      | YES  |     | NULL    |       |
| extra                 | float      | YES  |     | NULL    |       |
| mta_tax               | float      | YES  |     | NULL    |       |
| tip_amount            | float      | YES  |     | NULL    |       |
| tolls_amount          | float      | YES  |     | NULL    |       |
| improvement_surcharge | float      | YES  |     | NULL    |       |
| total_amount          | float      | YES  |     | NULL    |       |
+-----------------------+------------+------+-----+---------+-------+
17 rows in set (0.24 sec)
```

mysql>

mysql> exit

Bye

Now it's time to get into our created EMR cluster.

c:\> ssh -i "C:\Users\User\Documents\Data Science\MapReducing Programming Assignment\mysqldb1.pem" hadoop@ec2-54-227-201-183.compute-1.amazonaws.com

```
c:\>ssh -i "C:\Users\User\Documents\Data Science\MapReducing Programming Assignment\mysqldb1.pem" hadoop@ec2-54-211-193-99.compute-1.amazonaws.com
The authenticity of host 'ec2-54-211-193-99.compute-1.amazonaws.com (54.211.193.99)' can't be established.
ED25519 key fingerprint is SHA256:OwRDu2aMFSFpBDUJ3UsOzhbGNEBtlODEMFMjAKMZ3x4.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'ec2-54-211-193-99.compute-1.amazonaws.com' (ED25519) to the list of known hosts.
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'

EEEEEEEEEEEEEEEEEEEE MMMMMMMM           MMMMMMMM RRRRRRRRRRRRRR
E::::::::::::::::::E M::::::M        M::::::M R::::::::::::::R
EE::::::EEEEEEEEE:::E M::::::M       M::::::M R::::::RRRRRR:::::R
  E::::E       EEEEE M::::::M     M::::::M RR::::R      R::::R
  E::::E             M::::::M::M   M::M::::::M   R::::R      R::::R
  E::::::EEEEEEEEEE   M::::::M M::M M::M M::::::M   R::::RRRRRR::::::R
  E::::::::::::::::E  M::::::M  M::M::M  M::::::M   R::::::::::::RR
  E::::::EEEEEEEEEE   M::::::M   M::::M   M::::::M   R::::RRRRRR::::R
  E::::E             M::::::M    M::M    M::::::M   R::::R      R::::R
  E::::E       EEEEE M::::::M     MMM     M::::::M   R::::R      R::::R
EE::::::EEEEEEEE:::E M::::::M             M::::::M   R::::R      R::::R
E::::::::::::::::::E M::::::M             M::::::M RR::::R      R::::R
EEEEEEEEEEEEEEEEEEEE MMMMMMMM             MMMMMMMM RRRRRRR      RRRRRR

[hadoop@ip-172-31-43-192 ~]$
```

Download dataset yellow_tripdata_2017-01.csv – 04.csv from NYC-TLC

[hadoop@ip-172-31-34-21 ~]$ wget -P /home/hadoop/ https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-01.csv

--2025-02-04 20:48:12-- https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-01.csv

Resolving nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)... 52.217.231.209, 3.5.27.73, 16.182.68.97, ...

Connecting to nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)|52.217.231.209|:443... connected.

HTTP request sent, awaiting response... 200 OK

Length: 914029540 (872M) [text/csv]

Saving to: ‘/home/hadoop/yellow_tripdata_2017-01.csv’

yellow_tripdata_2017-01.csv 100%[==>] 871.69M 49.6MB/s in 18s

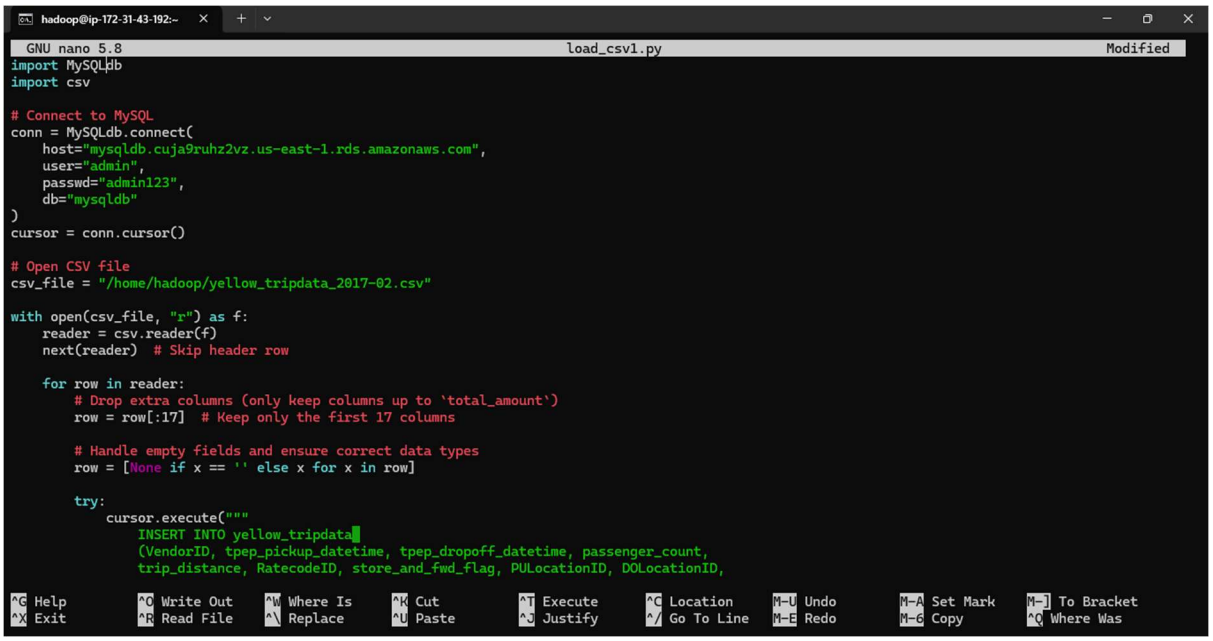
2025-02-04 20:48:30 (48.8 MB/s) - ‘/home/hadoop/yellow_tripdata_2017-01.csv’ saved [914029540/914029540]

Now we move files to HDFS

[hadoop@ip-172-31-34-21 ~]$ hdfs dfs -put /home/hadoop/yellow_tripdata_2017-01.csv /user/hadoop/taxi_data/

Now we can open a nano file

[hadoop@ip-172-31-34-21 ~]$ nano load_csv.py



We load this code so python can help us load csv file into My SQL

import MySQLdb

import csv

Connect to MySQL

conn = MySQLdb.connect(

host="mysqldb.cuja9ruhz2vz.us-east-1.rds.amazonaws.com",

user="admin",

passwd="admin123",

```
    db="mysqldb"
)
cursor = conn.cursor()

# Open CSV file
csv_file = "/home/hadoop/yellow_tripdata_2017-02.csv"

with open(csv_file, "r") as f:
    reader = csv.reader(f)
    next(reader)  # Skip header row

    for row in reader:
        # Drop extra columns (only keep columns up to `total_amount`)
        row = row[:17]  # Keep only the first 17 columns

        # Handle empty fields and ensure correct data types
        row = [None if x == '' else x for x in row]

        try:
            cursor.execute("""
                INSERT INTO yellow_tripdata
                (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)
                VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s, %s)
            """, row)
        except Exception as e:
```

```
        print("Error inserting row:", row)
        print("Error Message:", e)


conn.commit()
cursor.close()
conn.close()
```

Run python script to load the csv file 01 - 04

[hadoop@ip-172-31-34-21 ~]$ python3 load_csv1.py

```
[hadoop@ip-172-31-40-210 ~]$ hdfs dfs -put /home/hadoop/yellow_tripdata_2017-01.csv /user/hadoop/taxi_data/
put: `/user/hadoop/taxi_data/': No such file or directory: `hdfs://ip-172-31-40-210.ec2.internal:8020/user/hadoop/taxi_data'
[hadoop@ip-172-31-40-210 ~]$ hdfs dfs -mkdir -p /user/hadoop/taxi_data
[hadoop@ip-172-31-40-210 ~]$ hdfs dfs -ls /user/hadoop/taxi_data
[hadoop@ip-172-31-40-210 ~]$ hdfs dfs -put /home/hadoop/yellow_tripdata_2017-01.csv /user/hadoop/taxi_data/
[hadoop@ip-172-31-40-210 ~]$ nano load_csv.py
[hadoop@ip-172-31-40-210 ~]$ python3 load_csv.py
Finished loading data from /home/hadoop/yellow_tripdata_2017-01.csv
```

After that we can get into MySQL and see if it's been loaded

```
[hadoop@ip-172-31-34-21 ~]$ python3 load_csv3.py

Finished loading data from /home/hadoop/yellow_tripdata_2017-04.csv
[hadoop@ip-172-31-34-21 ~]$
[hadoop@ip-172-31-34-21 ~]$ mysql -h mysqldb.cuja9ruhz2vz.us-east-1.rds.amazonaws.com -u admin -p
Enter password:
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MySQL connection id is 1016
Server version: 8.0.40 Source distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MySQL [(none)]> USE mysqldb;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
MySQL [mysqldb]> SELECT COUNT(*) FROM yellow_tripdata;


+----------+
| COUNT(*) |
+----------+
| 39223171 |
+----------+
1 row in set (40 min 4.949 sec)
```