

Creation of RDS instance, DB, Table creation in RDS instance and to upload the csv file into the RDS instance. And creation of EMR cluster.

Step 1: Created RDS instance with MySQL and publicly accessible instance with MySQL port allowed security group.

database-1

ModifyActions

Summary

DB Identifier database-1	CPU <div><div></div>13.92%</div>	Status Backing-up	Class db.t3.micro
Role Instance	Current activity 0 Connections	Engine MySQL Community	Region & AZ us-east-1c

Connectivity & securityMonitoringLogs & eventsConfigurationMaintenance & backupsTags

Connectivity & security

Endpoint & port

Endpoint
database-1.cxyzirgikluf.us-east-1.rds.amazonaws.com

Port
3306

Networking

Availability Zone
us-east-1c

VPC
vpc-0246a17e3d5aa4952

Subnet group
default-vpc-0246a17e3d5aa4952

Subnets
subnet-0f7a1ac235890f926
subnet-022829379e6ef75d8
subnet-09709bf643aeb83c6

Security

VPC security groups
default (sg-04d8117f9d5e3e484)
Active

Publicly accessible
Yes

Certificate authority
Info
rds-ca-2019

Certificate authority date
August 22, 2024, 22:38 (UTC+05:30)

Step2: Created EMR cluster with the name map-reduce-program-EMR with root file system size with 40GB

EC2EMR

Amazon EMR > EMR on EC2: Clusters > map-reduce-program-EMR

map-reduce-program-EMR

Updated less than a minute ago

Actions

Summary

Cluster info

Cluster ID
j-2A7DOWF2MG3KB

Cluster configuration
Instance groups

Capacity
1 Primary 0 Core 0 Task

Applications

Amazon EMR version
emr-5.30.1

Installed applications
HBase 1.4.13, Hadoop 2.8.5, Sqoop 1.4.7

Cluster management

Amazon S3 log URI
s3://aws-logs-170637578818-us-east-1/elasticmapreduce/

Persistent application Uls
YARN timeline server

Primary node public DNS
ec2-54-91-5-41.compute-1.amazonaws.com
Connect to the Primary Node using SSH

Status and time

Status
Waiting

Creation time
May 07, 2023, 17:58 (UTC+05:30)

Elapsed time
7 minutes, 41 seconds

PropertiesBootstrap actionsInstancesStepsApplicationsConfigurationsMonitoringEventsTags (0)

Cluster logs

Archive log files to Amazon S3
Turned on

Amazon S3 location
s3://aws-logs-170637578818-us-east-1/elasticmapreduce/

Encryption for logs
Turned off

Cluster termination

Termination option
Manually terminate cluster

Termination protection
Turned off

Step 3: Logged in to EMR cluster and created a directory named 'mapreduce-assignment' under /.

Commands used:

```
ssh -i EMR-cluster-key.pem hadoop@ec2-54-91-5-41.compute-1.amazonaws.com # to login  
to EMR cluster
```

```
sudo su - # To switch user to root
```

`df -h` # to check the sizes of file system to use that file system to download Datasets.

`mkdir /mapreduce-assignment #` to create directory under `/` filesystem since it is having enough space as shown in the below image.

```
cd /mapreduce-assignment/ # to change the directory.
```

```

(base) lkadari@lkadari-mac Bhanu-EMR % ssh -i EMR-cluster-key.pem hadoop@ec2-54-91-5-41.compute-1.amazonaws.com
Last login: Sun May 7 12:38:49 2023 from 49.204.178.254

 _ _ | _ _ | _
 _ | ( _ /
 _ _ \ _ _ | _ _ |

Amazon Linux 2 AMI

https://aws.amazon.com/amazon-linux-2/
101 package(s) needed for security, out of 168 available
Run "sudo yum update" to apply all updates.
-bash: warning: setlocale: LC_CTYPE: cannot change locale (UTF-8): No such file or directory

EEEEEEEEEEEEEEEEEEEE MMMMMMMM MMMMMMMM RRRRRRRRRRRRRRRR
E:.....:.....E M:.....:~ M:.....:~ M:.....:~ R:.....:~ R:.....:~
EE:.....:EEEEEEEE:~E M:.....:~ M:.....:~ M:.....:~ R:.....:~ RRRRRRRRR:~R
E:~E:~ EEEEE M:.....:~ M:.....:~ M:.....:~ R:~:~ R:~:~ R:~:~
E:~E:~ M:.....:~ M:~:~ M:~:~ M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
E:.....:EEEEEEEEEEEE M:.....:~ M:~:~ M:~:~ M:~:~ R:~:~ RRRRRRRRR:~R
E:.....:EEEEEEEEEEEE M:.....:~ M:.....:~ M:~:~ M:~:~ R:~:~ RRRRRRRRR:~R
E:~E:~ M:.....:~ M:~:~ M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
E:~E:~ EEEEE M:.....:~ M~M M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
EE:.....:EEEEEEEE:~E M:.....:~ M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
E:.....:.....:~E M:.....:~ M:.....:~ M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
EEEEEEEEEEEEEEEEEEEE MMMMMMMM MMMMMMMM RRRRRRRR RRRRRR

[hadoop@ip-172-31-30-149 ~]$ sudo su -

EEEEEEEEEEEEEEEEEEEE MMMMMMMM MMMMMMMM RRRRRRRRRRRRRRRR
E:.....:.....:~E M:.....:~ M:.....:~ M:.....:~ R:.....:~ R:.....:~
EE:.....:EEEEEEEE:~E M:.....:~ M:.....:~ M:.....:~ R:.....:~ RRRRRRRRR:~R
E:~E:~ EEEEE M:.....:~ M:.....:~ M:.....:~ R:~:~ R:~:~ R:~:~
E:~E:~ M:.....:~ M:~:~ M:~:~ M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
E:.....:EEEEEEEEEEEE M:.....:~ M:~:~ M:~:~ M:~:~ R:~:~ RRRRRRRRR:~R
E:.....:EEEEEEEEEEEE M:.....:~ M:~:~ M:~:~ M:~:~ R:~:~ RRRRRRRRR:~R
E:~E:~ M:.....:~ M:~:~ M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
E:~E:~ EEEEE M:.....:~ M~M M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
EE:.....:EEEEEEEE:~E M:.....:~ M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
E:.....:.....:~E M:.....:~ M:~:~ M:~:~ R:~:~ R:~:~ R:~:~
EEEEEEEEEEEEEEEEEEEE MMMMMMMM MMMMMMMM RRRRRRRR RRRRRR

[root@ip-172-31-30-149 ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        7.9G  0  7.9G  0% /dev
tmpfs           7.9G  0  7.9G  0% /dev/shm
tmpfs           7.9G  1.1M  7.9G  1% /run
tmpfs           7.9G  0  7.9G  0% /sys/fs/cgroup
/dev/xvda1      40G  3.8G  37G  10% /
/dev/xvdb1      5.0G  39M  5.0G  1% /emr
/dev/xvdb2      27G  888M  27G  4% /mnt
/dev/xvdc       32G  67M  32G  1% /mnt1
tmpfs          1.6G  0  1.6G  0% /run/user/0
tmpfs          1.6G  0  1.6G  0% /run/user/995
tmpfs          1.6G  0  1.6G  0% /run/user/988
tmpfs          1.6G  0  1.6G  0% /run/user/989
tmpfs          1.6G  0  1.6G  0% /run/user/993
tmpfs          1.6G  0  1.6G  0% /run/user/994
tmpfs          1.6G  0  1.6G  0% /run/user/992
tmpfs          1.6G  0  1.6G  0% /run/user/996

[root@ip-172-31-30-149 ~]$ mkdir /mapreduce-assignment
[root@ip-172-31-30-149 ~]$ cd /mapreduce-assignment/
[root@ip-172-31-30-149 mapreduce-assignment]# ls
[root@ip-172-31-30-149 mapreduce-assignment]# ll

```

Step 4: Downloading the datasets in to ‘mapreduce-assignment’ directory.

Commands to download the Datasets in to ‘/mapreduce-assignment’ directory.

```
wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-01.csv
```

```
wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-02.csv
```

```
wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-03.csv
```

```
wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-04.csv
```

```
wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-05.csv
```

```
wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-06.csv
```

```
[root@ip-172-31-38-149 mapreduce-assignment]# wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-01.csv
--2023-05-07 12:47:26-- 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)... 54.231.193.89, 54.231.228.153, 3.5.26.243, ...
Connecting to nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)[54.231.193.89]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 914829548 (872M) [text/csv]
Saving to: 'yellow_tripdata_2017-01.csv'

100M[=====] 914,829,548 47.2MB/s in 18s

2023-05-07 12:47:43 (47.3 MB/s) - 'yellow_tripdata_2017-01.csv' saved [914829548/914829548]

[root@ip-172-31-38-149 mapreduce-assignment]# wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-02.csv
--2023-05-07 12:47:43-- https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-02.csv
Resolving nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)... 52.216.12.12, 52.216.185.67, 52.216.111.91, ...
Connecting to nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)[52.216.12.12]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 863487858 (823M) [text/csv]
Saving to: 'yellow_tripdata_2017-02.csv'

100M[=====] 863,487,858 44.7MB/s in 18s

2023-05-07 12:48:01 (45.5 MB/s) - 'yellow_tripdata_2017-02.csv' saved [863487858/863487858]

[root@ip-172-31-38-149 mapreduce-assignment]# wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-03.csv
--2023-05-07 12:48:02-- https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-03.csv
Resolving nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)... 52.217.43.186, 52.217.283.177, 52.217.226.185, ...
Connecting to nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)[52.217.43.186]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 969889825 (928M) [text/csv]
Saving to: 'yellow_tripdata_2017-03.csv'

100M[=====] 969,889,825 43.3MB/s in 21s

2023-05-07 12:48:23 (43.6 MB/s) - 'yellow_tripdata_2017-03.csv' saved [969889825/969889825]

[root@ip-172-31-38-149 mapreduce-assignment]# wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-04.csv
--2023-05-07 12:48:23-- https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-04.csv
Resolving nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)... 54.231.229.217, 3.5.21.116, 3.5.25.90, ...
Connecting to nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)[54.231.229.217]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 946349441 (903M) [text/csv]
Saving to: 'yellow_tripdata_2017-04.csv'

100M[=====] 946,349,441 53.3MB/s in 28s

2023-05-07 12:48:43 (45.2 MB/s) - 'yellow_tripdata_2017-04.csv' saved [946349441/946349441]

[root@ip-172-31-38-149 mapreduce-assignment]# wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-05.csv
--2023-05-07 12:48:43-- https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-05.csv
Resolving nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)... 52.217.111.149, 3.5.28.192, 3.5.28.192, ...
Connecting to nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)[52.217.111.149]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 951965526 (908M) [text/csv]
Saving to: 'yellow_tripdata_2017-05.csv'

100M[=====] 951,965,526 58.1MB/s in 16s

2023-05-07 12:49:02 (47.1 MB/s) - 'yellow_tripdata_2017-05.csv' saved [951965526/951965526]

[root@ip-172-31-38-149 mapreduce-assignment]# wget https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-06.csv
--2023-05-07 12:49:15-- https://nyc-tlc-upgrad.s3.amazonaws.com/yellow_tripdata_2017-06.csv
Resolving nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)... 54.231.288.281, 3.5.29.19, 52.216.51.235, ...
Connecting to nyc-tlc-upgrad.s3.amazonaws.com (nyc-tlc-upgrad.s3.amazonaws.com)[54.231.288.281]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 918286488 (878M) [text/csv]
Saving to: 'yellow_tripdata_2017-06.csv'

100M[=====] 918,286,488 50.1MB/s in 18s

2023-05-07 12:49:33 (47.1 MB/s) - 'yellow_tripdata_2017-06.csv' saved [918286488/918286488]
```

Command:

Is -lrt # command used to list all the files in directory in a sorted manner.

```
[root@ip-172-31-38-149 mapreduce-assignment]# ls -lrt
total 542548
-rw-r--r-- 1 root root 914829548 Nov 25 15:05 yellow_tripdata_2017-01.csv
-rw-r--r-- 1 root root 863487858 Nov 25 15:07 yellow_tripdata_2017-02.csv
-rw-r--r-- 1 root root 969889825 Nov 25 15:07 yellow_tripdata_2017-03.csv
-rw-r--r-- 1 root root 946349441 Nov 25 15:08 yellow_tripdata_2017-04.csv
-rw-r--r-- 1 root root 951965526 Nov 25 15:08 yellow_tripdata_2017-05.csv
-rw-r--r-- 1 root root 918286488 Nov 25 15:09 yellow_tripdata_2017-06.csv
[root@ip-172-31-38-149 mapreduce-assignment]# pwd
/mapreduce-assignment
[root@ip-172-31-38-149 mapreduce-assignment]#
```

Step 5: Connecting RDS instance from EMR cluster and MySQLWorkbench

Commands:

```
mysql -h database-1.czxy6rglkiuf.us-east-1.rds.amazonaws.com -P 3306 -u admin -p # to connect RDS instance from EMR cluster.
```

```
show databases; # to see existing databases.
```



Connection Name
Amazon-RDS

Host: **ip-10-1-3-134**
Socket: **/tmp/mysql.sock**
Port: **3306**
Version: **8.0.32 (Source distribution)**
Compiled for: **Linux (x86_64)**
Configuration File: **unknown**
Running Since: **Sun May 7 17:45:42 2023 (0:39)**

Available Server Features

Performance Schema: ☐ Off
Thread Pool: ☐ n/a
Memcached Plugin: ☐ n/a
Semisync Replication Plugin: ☐ n/a
SSL Availability: ☒ On

PAM Authentication:
Password Validation:
Audit Log:
Firewall:
Firewall Trace:

Server Directories

Base Directory: **/rdsdbbin/mysql-8.0.32.R3/**
Data Directory: **/rdsdbdata/db/**
Disk Space in Data Dir: **unable to retrieve**
InnoDB Data Directory: **/rdsdbdata/db/innodb**
Plugins Directory: **/rdsdbbin/mysql-8.0.32.R3/lib/plugin/**
Tmp Directory: **/rdsdbdata/tmp**
Error Log: ☒ On **/rdsdbdata/log/error/mysql-error.log**
General Log: ☐ Off
Slow Query Log: ☐ Off

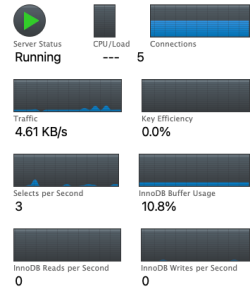
Replica

this server is not a replica in a replication setup

Authentication

SHA256 Password Private Key: **private_key.pem**
SHA256 Password Public Key: **public_key.pem**

SSL



The screenshot shows the Amazon RDS Administration Console. The left sidebar contains navigation options: MANAGEMENT (Server Status, Client Connections, Users and Privileges, Status and System Variables, Data Export, Data Import/Restore), INSTANCE (Startup / Shutdown, Server Logs, Options File), and PERFORMANCE (Dashboard, Performance Reports, Performance Schema Setup). The main pane shows the 'Server Status' page for the instance 'Amazon-RDS'. The 'Query 1' tab is active, displaying the query 'show databases;'. The 'Result Grid' shows the output of the query, listing the databases: information_schema, mysql, performance_schema, and sys. The 'Action Output' table shows the execution details: 1 row returned, 4 rows returned, and a duration of 0.215 sec / 0.000022...

```
[root@ip-172-31-38-149 mapreduce-assignment]# mysql -h database-1.czyxgk1kuf.us-east-1.rds.amazonaws.com -P 3306 -u admin -p
Enter password:
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MySQL connection id is 96
Server version: 8.0.32 Source distribution

Copyright (c) 2000, 2015, Oracle, MariaDB Corporation Ab and others.

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

MySQL [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
+-----+
4 rows in set (0.00 sec)

MySQL [(none)]>
```

Step 6: In RDS instance, creating Database named 'trip' and the table named 'yello' with all the columns as same as in the given dataset under 'trip' database.

Commands:

```
create database trip; # to create database named 'trip' in RDS
```

```
use trip; # To enter trip database.
```

Below command is used to create table named 'yello' under 'trip' with the column names as same as in the datasets.

```
create table yello
(
VendorID int,
tpep_pickup_datetime datetime,
tpep_dropoff_datetime datetime,
passenger_count int,
trip_distance float,
RatecodeID int,
store_and_fwd_flag varchar(255),
PULocationID varchar(255),
DOLocationID varchar(255),
payment_type int,
fare_amount float,
extra float,
mta_tax float,
tip_amount float,
tolls_amount float,
improvement_surcharge float,
total_amount float,
Airport_fee float
);
```

```
show tables; # to check the existing tables.
```

```

mysql [(none)]>
MySQL [(none)]> create database trip;
Query OK, 1 row affected (0.01 sec)

MySQL [(none)]> show databases;
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| performance_schema |
| sys |
| trip |
+-----+
6 rows in set (0.00 sec)

MySQL [(none)]> use trip;
Database changed
MySQL [trip]> create table yello
-> (
-> VendorID int,
-> tpep_pickup_datetime datetime,
-> tpep_dropoff_datetime datetime,
-> passenger_count int,
-> trip_distance float,
-> RatecodeID int,
-> store_and_fwd_flag varchar(255),
-> PUlocationID varchar(255),
-> DOlocationID varchar(255),
-> payment_type int,
-> fare_amount float,
-> extra float,
-> mta_tax float,
-> tip_amount float,
-> tolls_amount float,
-> improvement_surcharge float,
-> total_amount float,
-> Airport_fee float
-> );
Query OK, 0 rows affected (0.03 sec)

MySQL [trip]> show tables;
+-----+
| Tables_in_trip |
+-----+
| yello |
+-----+
1 row in set (0.00 sec)

MySQL [trip]>

```

```

MySQL [trip]> show tables;
+-----+
| Tables_in_trip |
+-----+
| yello |
+-----+
1 row in set (0.00 sec)

MySQL [trip]>
MySQL [trip]> select * from yello;
Empty set (0.00 sec)

MySQL [trip]> select count(1) from yello;
+-----+
| count(1) |
+-----+
| 0 |
+-----+
1 row in set (0.00 sec)

MySQL [trip]>

```

Step 7(a): In RDS instance, Loading 'yellow_tripdata_2017-01.csv' and 'yellow_tripdata_2017-02.csv' datasets [As per the task 1] into the existing table 'yello' under 'trip' database in MySQL.

Commands:

Below commands used to upload the data from EMR cluster to RDS instance.

```

LOAD DATA LOCAL INFILE '/mapreduce-assignment/yellow_tripdata_2017-01.csv'
INTO TABLE yello
FIELDS TERMINATED BY ','

```

```
LINES TERMINATED BY '\n'
IGNORE 1 LINES;
```

```
LOAD DATA LOCAL INFILE '/mapreduce-assignment/yellow_tripdata_2017-02.csv'
INTO TABLE yello
FIELDS TERMINATED BY ','
LINES TERMINATED BY '\n'
IGNORE 1 LINES;
```

```
MySQL [trip]>
MySQL [trip]>
MySQL [trip]> LOAD DATA LOCAL INFILE '/mapreduce-assignment/yellow_tripdata_2017-01.csv'
-> INTO TABLE yello
-> FIELDS TERMINATED BY ','
-> LINES TERMINATED BY '\n'
-> IGNORE 1 LINES;
Query OK, 9710820 rows affected, 65635 warnings (1 min 59.38 sec)
Records: 9710820 Deleted: 0 Skipped: 0 Warnings: 9710820

MySQL [trip]> LOAD DATA LOCAL INFILE '/mapreduce-assignment/yellow_tripdata_2017-02.csv'
-> INTO TABLE yello
-> FIELDS TERMINATED BY ','
-> LINES TERMINATED BY '\n'
-> IGNORE 1 LINES;
Query OK, 9169775 rows affected, 65635 warnings (2 min 5.72 sec)
Records: 9169775 Deleted: 0 Skipped: 0 Warnings: 9169775

MySQL [trip]> select count(1) from yello;
+-----+
| count(1) |
+-----+
| 18880595 |
+-----+
1 row in set (46.60 sec)
```

Step 7(b): Running count on 'yello' table and viewing the content of the data with limit of 10 in CLI. since it is a huge records.

Command:

select count(1) from yello; # to check the row count in 'yello' table.

select * from yello limit 10; # To display the first 10 records from the 'yello' table.

```
MySQL [trip]>
MySQL [trip]> select count(1) from yello;
+-----+
| count(1) |
+-----+
| 18880595 |
+-----+
1 row in set (37.62 sec)

MySQL [trip]> select * from yello limit 10;
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 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_amo |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 0 | 1 | 2017-01-01 00:32:05 | 2017-01-01 00:37:48 | 0 | 1 | 1.2 | 1 | N | 140 | 236 | 2 | 6.5 | 0.5 | 0.5 | |
| 0 | 1 | 2017-01-01 00:43:25 | 2017-01-01 00:47:42 | 0 | 2 | 0.7 | 1 | N | 237 | 140 | 2 | 5 | 0.5 | 0.5 |
| 0 | 1 | 2017-01-01 00:49:10 | 2017-01-01 00:53:53 | 0 | 2 | 0.8 | 1 | N | 140 | 237 | 2 | 5.5 | 0.5 | 0.5 |
| 0 | 1 | 2017-01-01 00:36:42 | 2017-01-01 00:41:09 | 0 | 1 | 1.1 | 1 | N | 41 | 42 | 2 | 6 | 0.5 | 0.5 |
| 0 | 1 | 2017-01-01 00:07:41 | 2017-01-01 00:18:16 | 0 | 1 | 3 | 1 | N | 48 | 263 | 2 | 11 | 0.5 | 0.5 |
| 0 | 1 | 2017-01-01 00:20:52 | 2017-01-01 00:24:59 | 0 | 2 | 0.7 | 1 | N | 236 | 262 | 2 | 5 | 0.5 | 0.5 |
| 0 | 1 | 2017-01-01 00:33:49 | 2017-01-01 00:42:38 | 0 | 2 | 1.6 | 1 | N | 236 | 238 | 1 | 8 | 0.5 | 0.5 | 1 |
| .85 | 1 | 2017-01-01 00:48:22 | 2017-01-01 00:52:55 | 0 | 2 | 0.6 | 1 | N | 238 | 239 | 1 | 5 | 0.5 | 0.5 | 1 |
| .25 | 1 | 2017-01-01 00:57:12 | 2017-01-01 01:06:28 | 0 | 2 | 1 | 1 | N | 239 | 48 | 1 | 7.5 | 0.5 | 0.5 | 1 |
| .75 | 1 | 2017-01-01 00:10:25 | 2017-01-01 00:29:00 | 0 | 1 | 1 | 1 | N | 246 | 48 | 2 | 12 | 0.5 | 0.5 |
| 0 | 0 | 0.3 | 13.3 | 0 | 0 | 0.3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
10 rows in set (0.01 sec)

MySQL [trip]> ||
```

Step 7(c): Viewing the records in MySQLWorkbench.

Amazon-RDS

Administration - Server Status

Schemas

Query 1

Limit to 1000 rows

Context Help

Snippets

SCHMAS

Filter objects

> sys

1 show databases;

2

3 use trip;

4

5 select * from yello;

100%

215

Result Grid

Filter Rows

Export

Fetch rows

Report Grid

Form Editor

MySQL

	VendorID	trip_pickup_datetime	trip_dropoff_datetime	passenger_count	trip_distance	RatecodeID	store_and_fwd_flag	PULocationID	DOLocationID	payment_type	fare_amount	extra	mta_tax	tip_amount	toils_amount	improvement_surchar	total_amount
> 1	2017-01-01 00:32:05	2017-01-01 00:37:48	1	1.2	1	N		140	226	2	6.5	0.5	0.5	0	0	0.3	7.8
1	2017-01-01 00:43:25	2017-01-01 00:47:42	2	0.7	1	N		237	140	2	5	0.5	0.5	0	0	0.3	6.3
1	2017-01-01 00:49:10	2017-01-01 00:53:53	2	0.8	1	N		140	237	2	5.5	0.5	0.5	0	0	0.3	6.8
1	2017-01-01 00:36:42	2017-01-01 00:41:09	1	1.1	1	N		41	42	2	6	0.5	0.5	0	0	0.3	7.3
1	2017-01-01 00:07:41	2017-01-01 00:18:16	1	3	1	N		48	263	2	11	0.5	0.5	0	0	0.3	12.3
1	2017-01-01 00:20:52	2017-01-01 00:24:59	2	0.7	1	N		236	262	2	5	0.5	0.5	0	0	0.3	6.3
1	2017-01-01 00:33:49	2017-01-01 00:42:38	2	1.6	1	N		236	238	1	8	0.5	0.5	1.85	0	0.3	11.15
1	2017-01-01 00:49:22	2017-01-01 00:52:15	2	0.6	1	N		238	239	1	5	0.5	0.5	1.25	0	0.3	7.55
1	2017-01-01 00:57:12	2017-01-01 01:06:28	2	1	1	N		239	48	1	7.5	0.5	0.5	1.75	0	0.3	10.55
1	2017-01-01 00:10:25	2017-01-01 00:29:06	1	1	1	N		246	48	2	12	0.5	0.5	0	0	0.3	13.3

Object Info

Session

No object selected

Action Output

Time

Action

Response

Duration / Fetch Time

1	18:26:40	show databases	4 row(s) returned	0.215 sec / 0.000022...
2	18:58:39	show databases	5 row(s) returned	0.283 sec / 0.00003...
3	18:59:04	use trip	0 row(s) affected	0.239 sec
4	18:59:04	select * from yello LIMIT 0, 1000	1000 row(s) returned	0.477 sec / 0.216 sec

Query Completed