Load Data into MYSQL RDS Instance

Author:	Sri Govind Gutala
Environment:	Amazon Web Services
Date:	September 5, 2016
Document:	MySQL_RDS_Creation_In_AWS.docx

Table of Contents

1. Required Prior Knowledge

You are required to have a basic knowledge on AWS console and its basic components such as IAM, RDS, VPC, creating roles, policies, VPC security groups, etc.

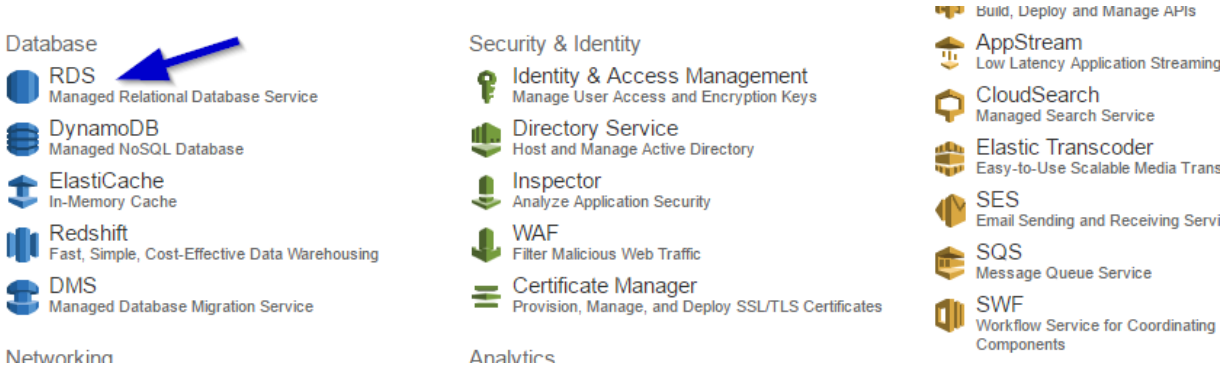
- RDS is a cloud based Relational Database Service delivered by AWS. It is easy to set up, operate and scale.
- In the case of failure, it ensures a safe and quick recovery.
- You can launch a DB instance and get access to a fully featured relational database without worrying about DBA tasks like backups, patch management and security management.
- Using AWS RDS, you can easily achieve high availability and high redundancy.

This document provides information on how to set up your RDS Instance and load data into it. I'm using AWS free tier membership and using micro instances, which are available for free tier. Lets get started!

2. Create MYSQL RDS Instance on AWS

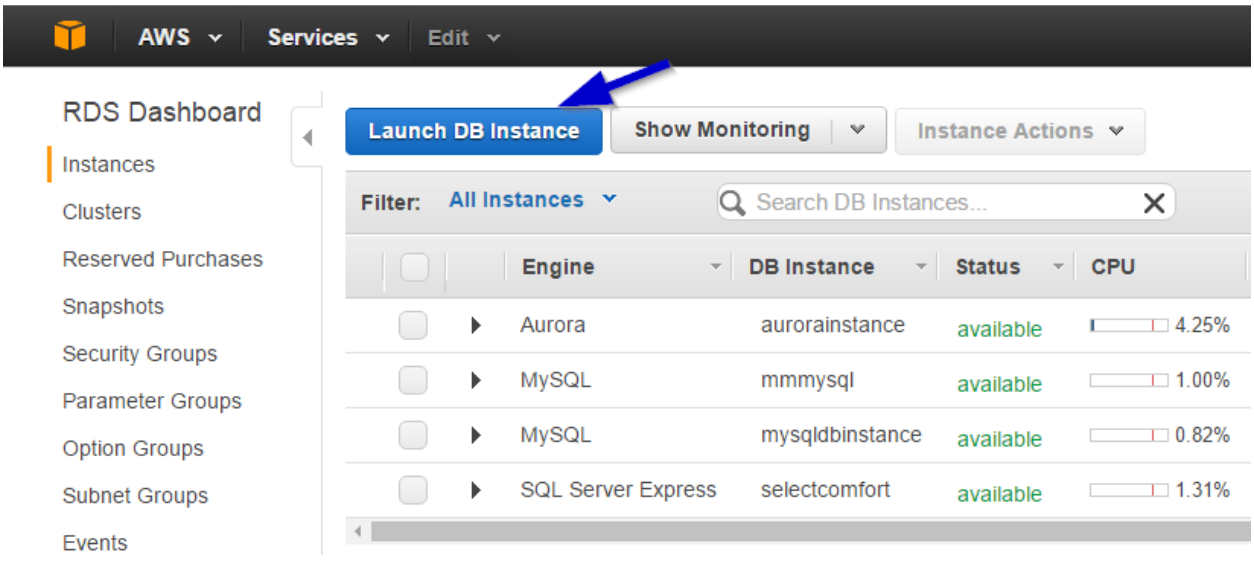
Step 1: Open AWS Console

Login to AWS account with your credentials and select RDS under Database Section

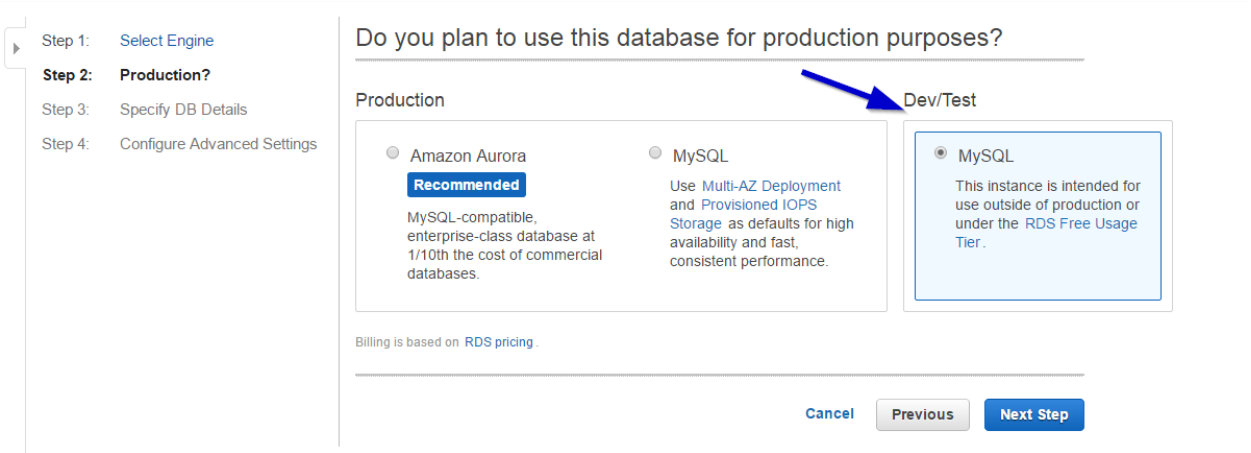


Step 2: Launch DB Instance

Click on **Instances** that is on left side of the page and click **Launch DB Instance**.



Select MySQL as engine and click on "**Select**" button. Select Database for **Dev/Test** as shown in below figure and click on "**Next Step**".



Step 3: Specify DB Details

Specify the DB Details as shown in below figures and click on **Next Step**.

Figure 1

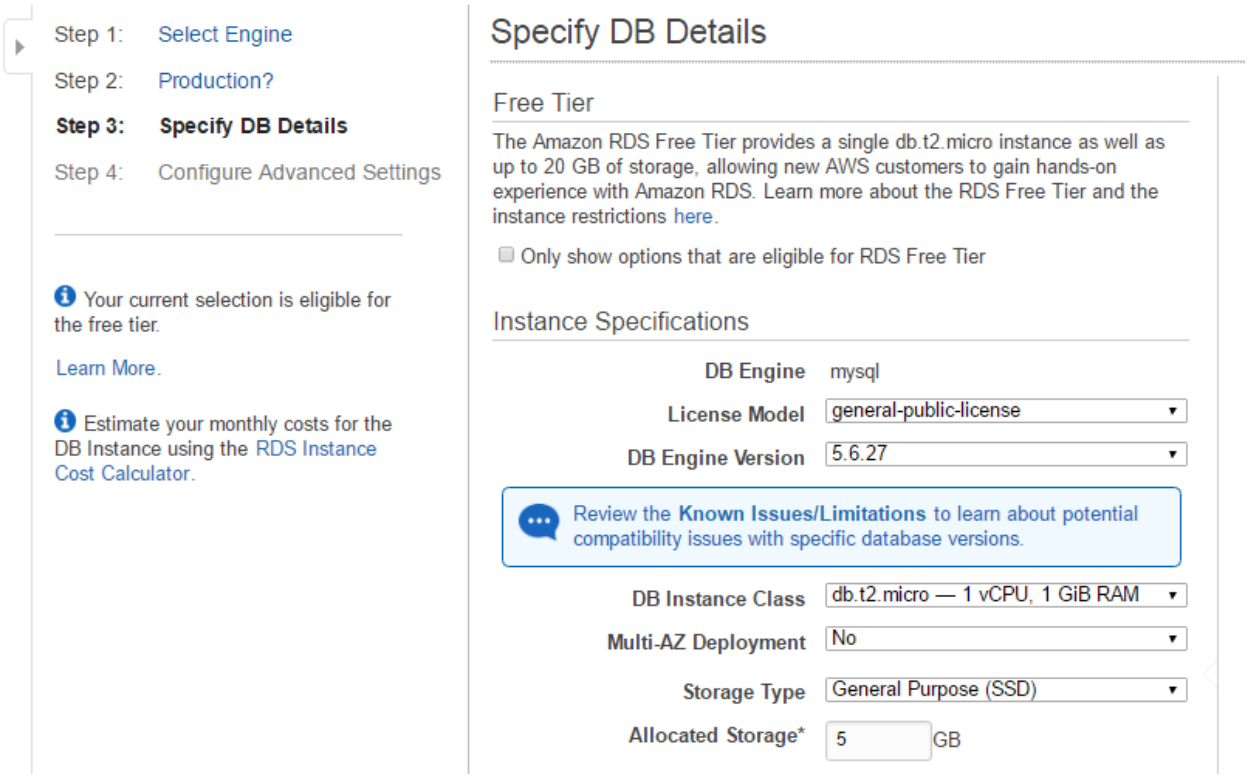
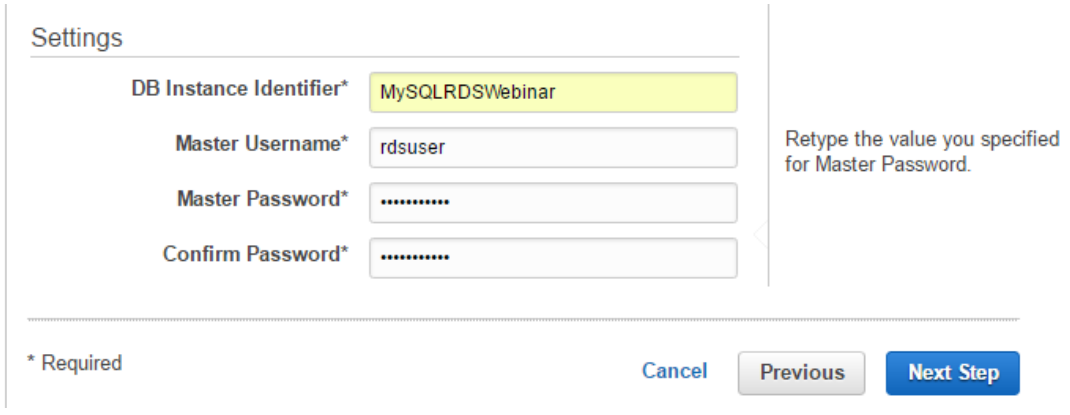
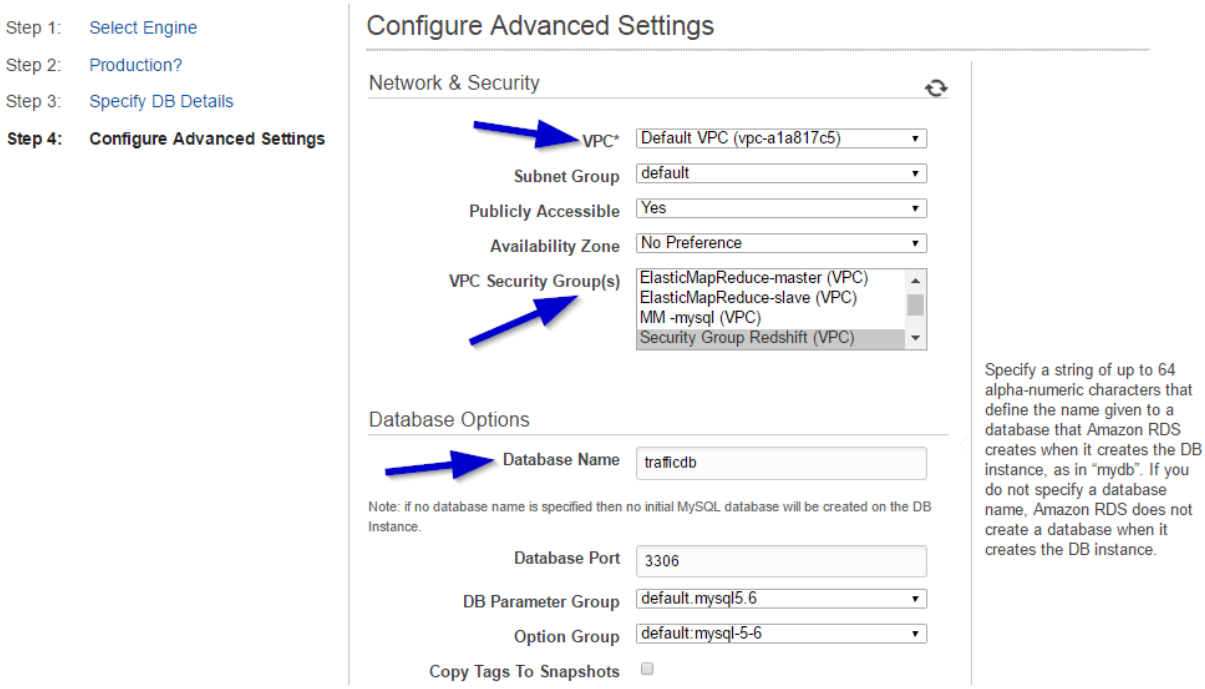


Figure 2

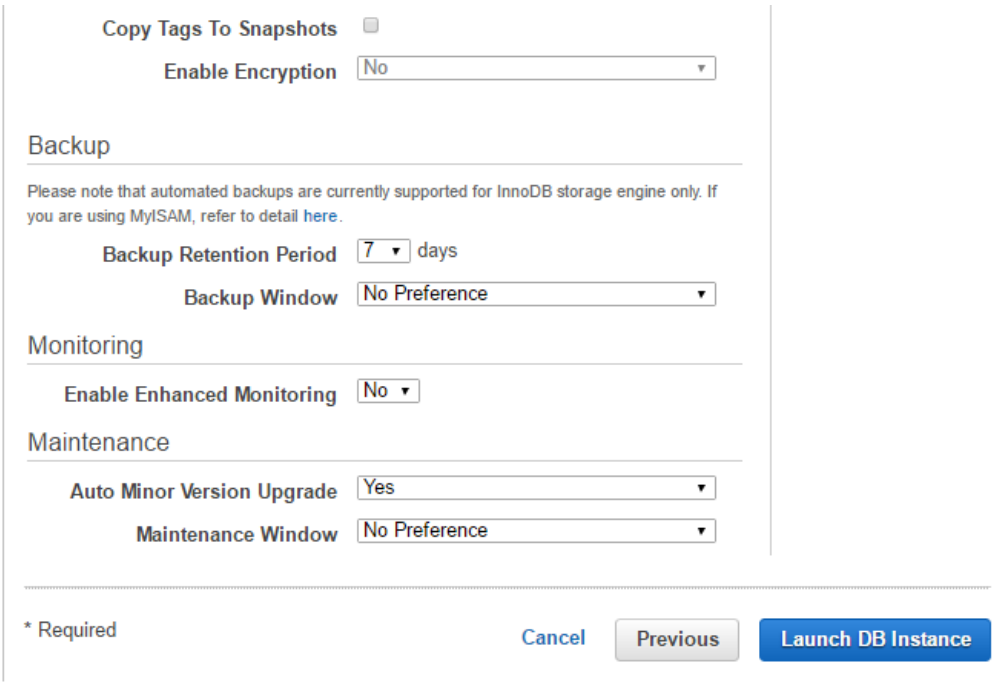


Step 4: Configure Advanced Settings

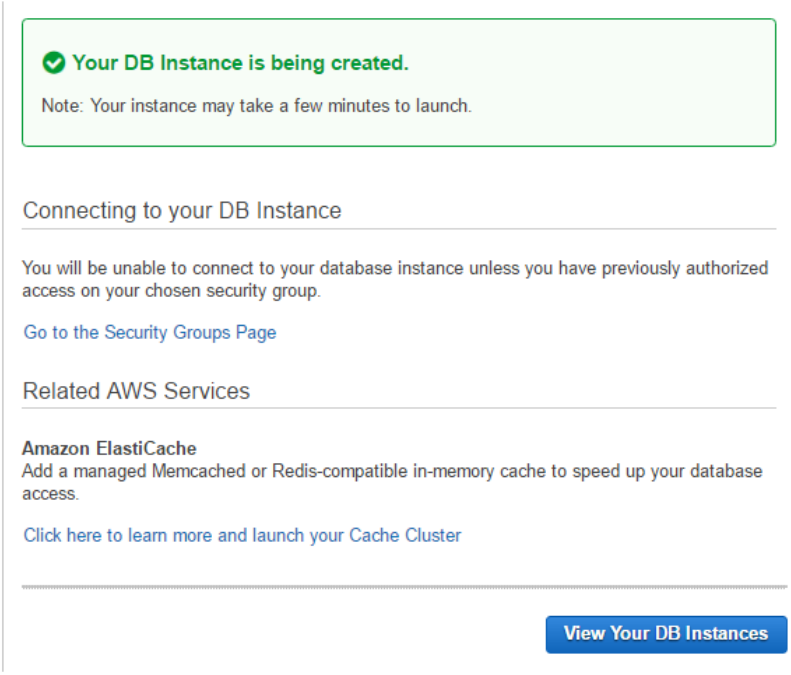
Provide VPC and corresponding VPC security group that you have created already. And select a database name as "**trafficdb**".



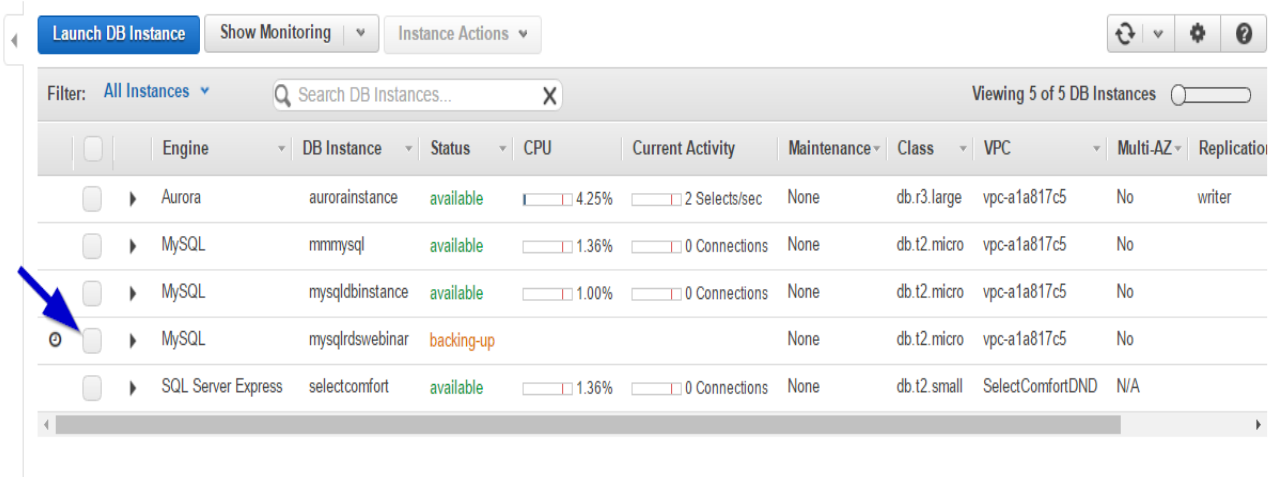
Leave other parameters with default values and click on "**Launch DB Instance**"



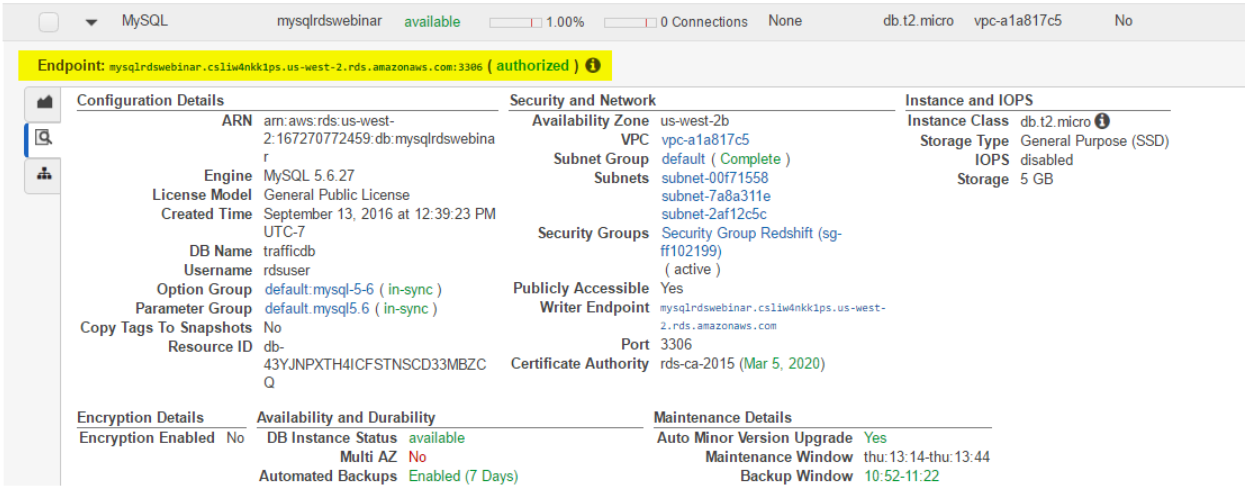
Once Launch DB Instance is clicked, it starts creating DB. Click on View Your DB Instances to view the DB Instance that was created in above step.



The below figure shows that the "**mysqlrdswebinar**" DB Instance is being created. The status changes to "**available**" once database is created completely.



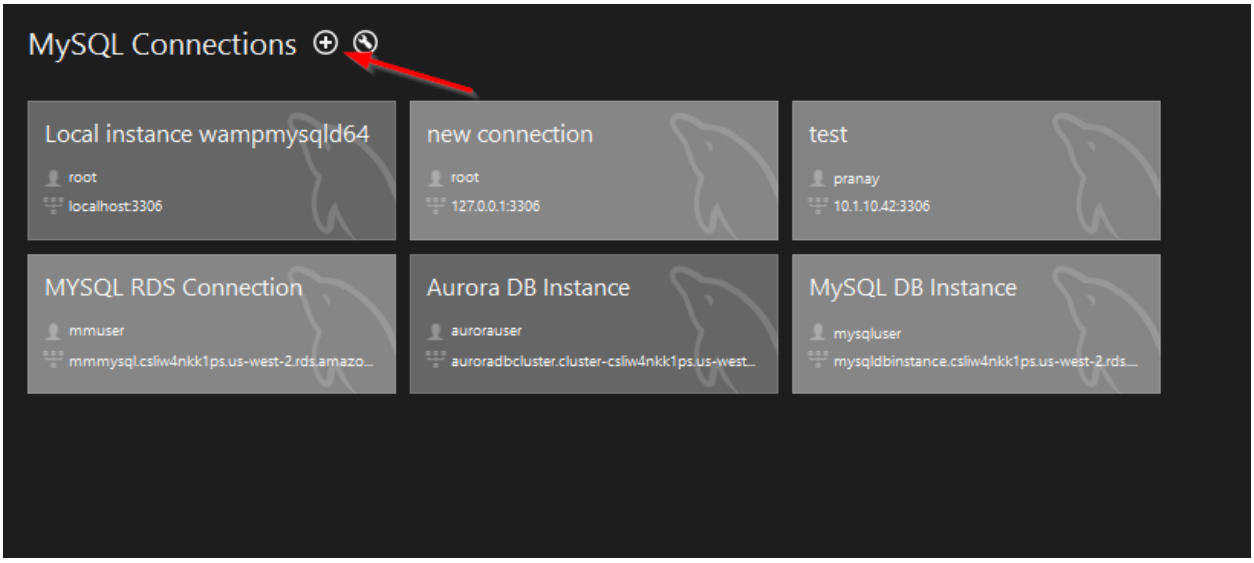
Click on the DB Instance to view connection details (Endpoint). Endpoint is used to connect to MySQL RDS instance through MySQL workbench. In the below figure, one can find database name, username and Endpoint.



3. Connect to MySQL RDS Instance using MySQL Workbench

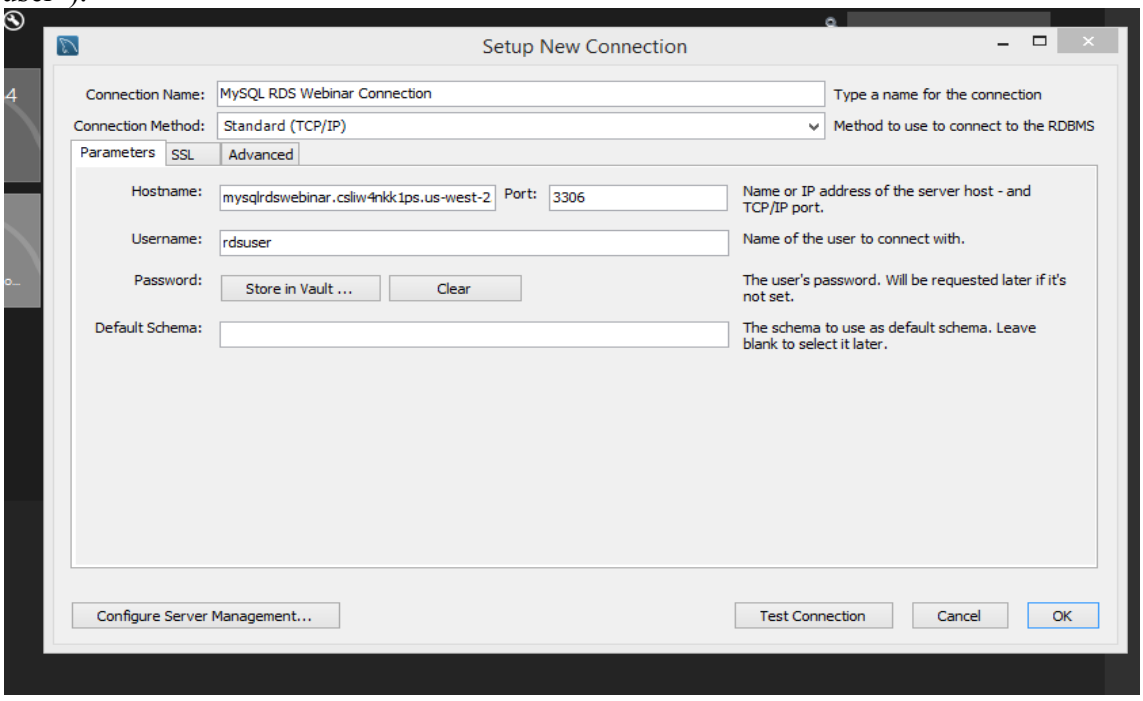
Step 1: Create a connection

Download MySQL workbench from https://dev.mysql.com/downloads/workbench/ .
Open MySQL workbench and click on plus button to Set up a new connection.

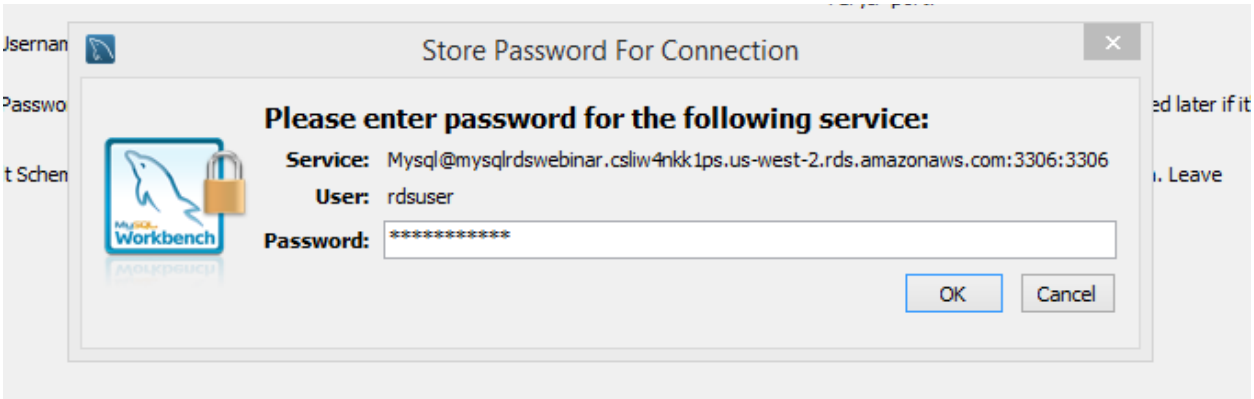


Step 2: Provide Connection details

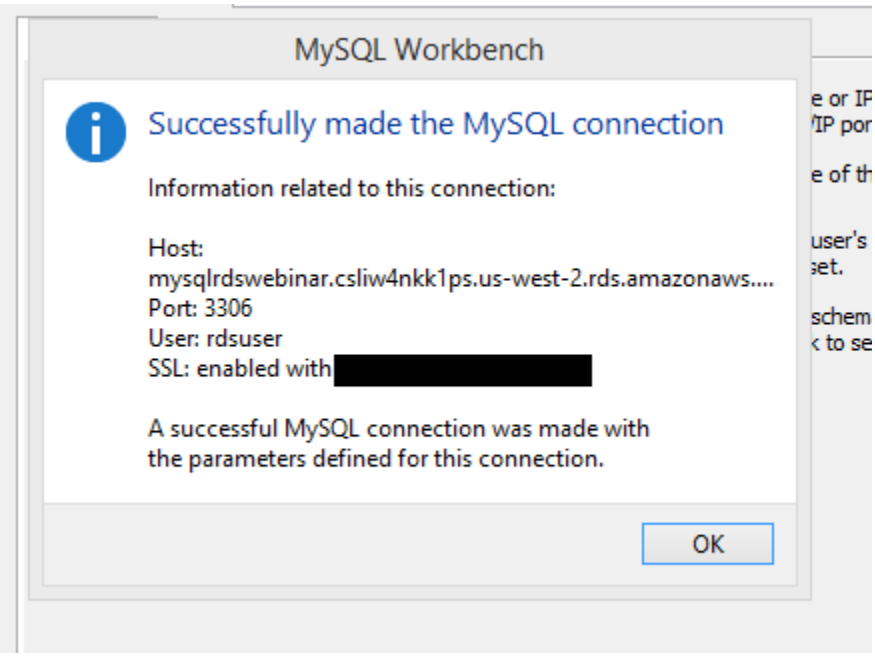
Specify endpoint for Hostname (remove port number "**:3306"** at the end of endpoint), Port number as 3306, username and password as provided while creating DB Instance (here it is "rdsuser").



Click on **Store in Vault** for password. It will prompt to enter the password. Specify the password and click on **OK.**

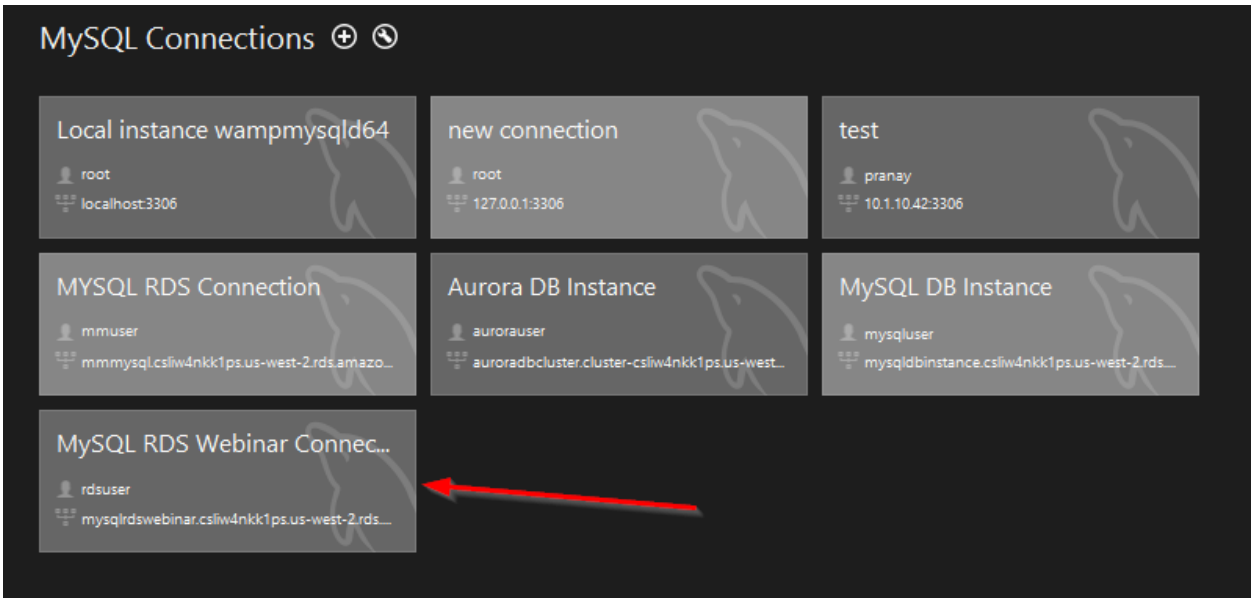


If the connection is successful, it shows the below message.

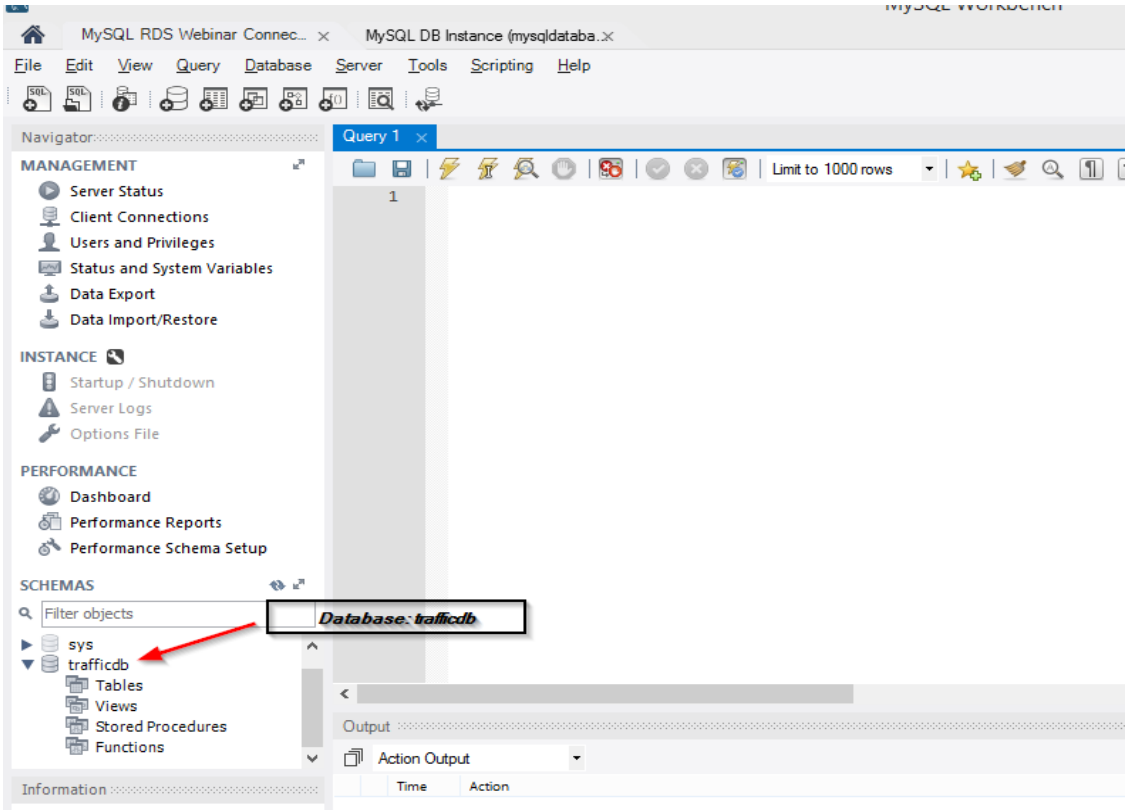


Step 3: Open MySQL RDS Instance

Open on DB Connection that was created in above step.

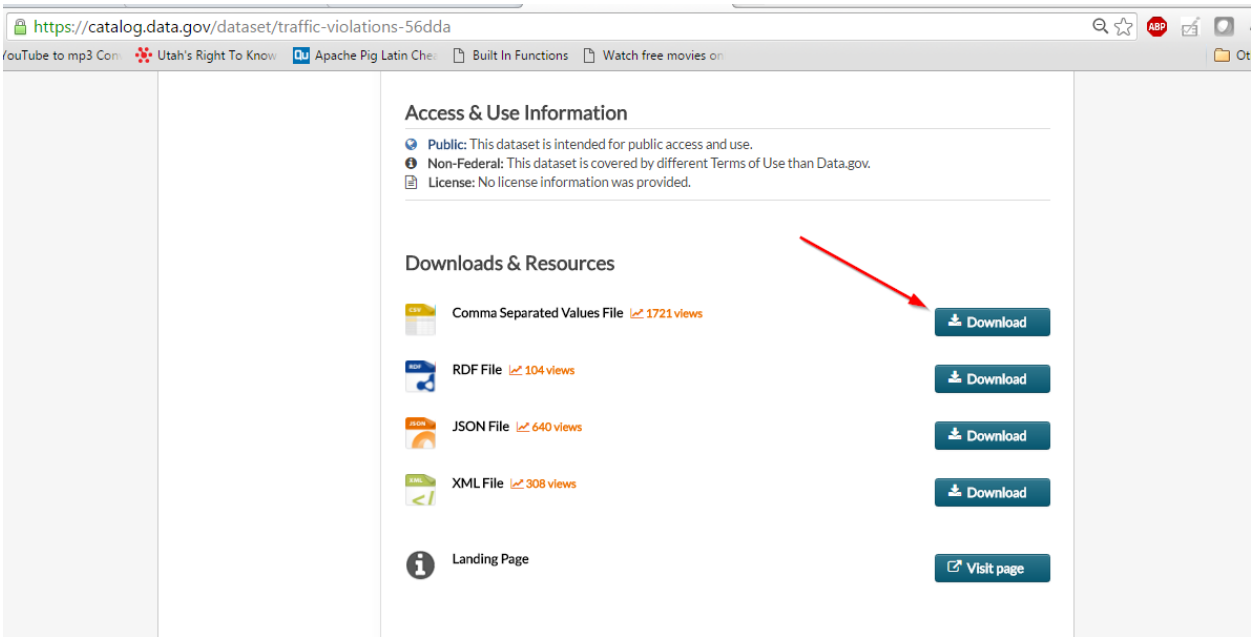


It opens a SQL Editor to execute sql queries.



Step 4: Download Dataset

Click on https://catalog.data.gov/dataset/traffic-violations-56dda and click on Download for CSV dataset.

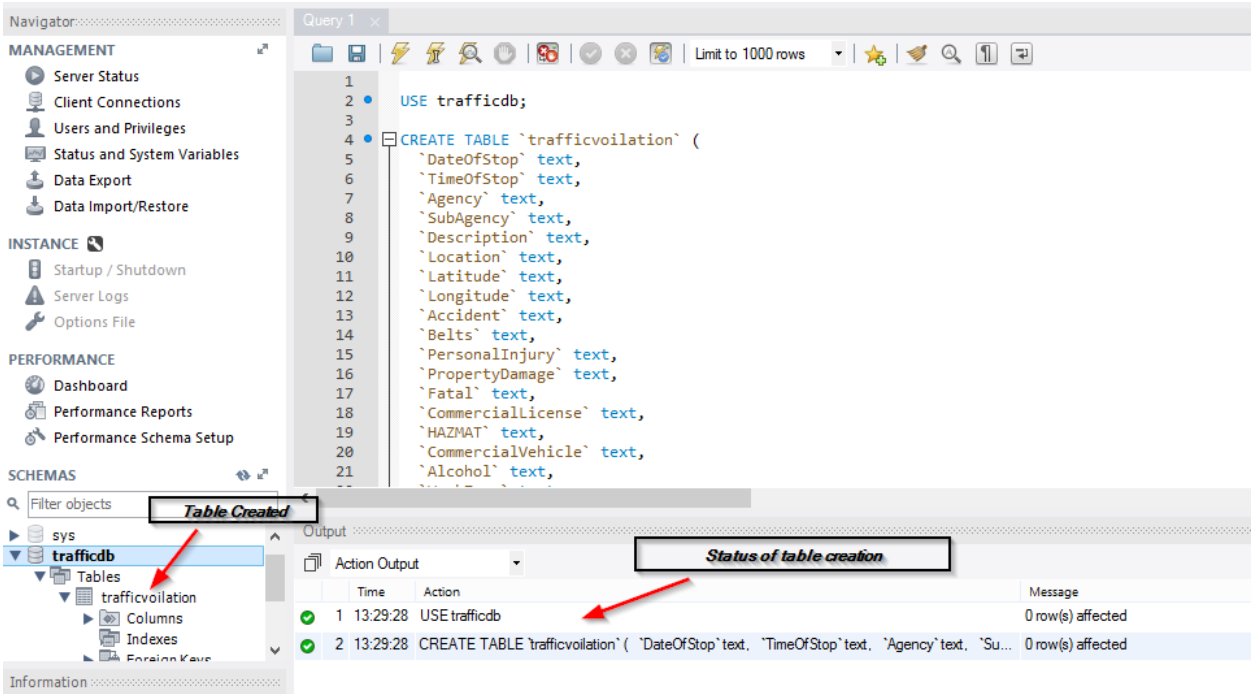


Step 5: Table Creation

Create table with below command on MySQL Workbench.

```
USE trafficdb;

CREATE TABLE `trafficvoilation` (
  `DateOfStop` text,
  `TimeOfStop` text,
  `Agency` text,
  `SubAgency` text,
  `Description` text,
  `Location` text,
  `Latitude` text,
  `Longitude` text,
  `Accident` text,
  `Belts` text,
  `PersonalInjury` text,
  `PropertyDamage` text,
  `Fatal` text,
  `CommercialLicense` text,
  `HAZMAT` text,
  `CommercialVehicle` text,
  `Alcohol` text,
  `WorkZone` text,
  `State` text,
  `VehicleType` text,
  `Year` int(11) DEFAULT NULL,
  `Make` text,
  `Model` text,
  `Color` text,
  `ViolationType` text,
  `Charge` text,
  `Article` text,
  `ContributedToAccident` text,
  `Race` text,
  `Gender` text,
  `DriverCity` text,
  `DriverState` text,
  `DLState` text,
  `ArrestType` text,
  `Geolocation` text
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```



Step 6: Load data into table

Execute below commands to load the dataset into table created in Step 5.

SET autocommit=0;
SET unique_checks=0;
SET foreign-key_checks=0;
SET session character_set_database=latin1;

LOAD DATA LOCAL INFILE ‘C:\\Users\\sri\\Downloads\\Traffics_Violations.csv’
INTO TABLE trafficviolation
FIELDS TERMINATED BY ‘,’ ENCLOSED BY “”
LINES TERMINATED BY ‘\n’
IGNORE 1 LINES;

SET unique_checks=1;
SET foreign_key_checks=1;
COMMIT;

It takes about 1 to 2 minutes to load 300 MB traffic violation file.

Step 7: Verify data in table

Execute any query to view the data loaded in table.

```
Select * from trafficvoilation limit 10;
```

The result of query can be seen in below image.