Ex: 10	Connect MySQL to Tableau

Aim:

To connect MySQL with Tableau to visualize more dataset right from the given dataset.

Introduction to MySQL

MySQL is an open-source database management system that is distributed, supported and developed by Oracle Incorporation (INC). MySQL is a relational database which stores the data in the form of tables and views. It supports various database objects such as tables, stored procedures, functions, triggers, views, indexes, and even cursors. MySQL database server runs comfortably on any laptop or desktop even with various web applications or servers installed on the system. MySQL database server is designed to support large databases which may contain data of many organizations. MySQL database servers support a wide range of functions and multiple web API's. It performs exceptionally well & securely accesses various databases on the internet. It ensures connectivity with servers and devices at all times.

Introduction to Tableau

Tableau is a powerful business intelligence tool used to turn raw data into an understandable format. It is a tool popularly used to visualize data and can be understood even by a non-technical user. It creates visualizations with the help of dashboards and worksheets, this helps users perform real-time analysis in a very fast and secure manner. It doesn't require any programming skill or technical background to operate it

Methods to connect MySQL to Tableau:

Method 1: Using Tableau's MySQL Connector

Tableau's in-built MySQL connector conveniently establishes a connection with MySQL. You can easily set up a data source and use Tableau to visualize your data to perform a fruitful analysis for your business. This method requires you to install the MySQL ODBC driver.

Method 2: Using ODBC to connect MySQL to Tableau

ODBC drivers can also be used to establish a connection between MySQL & Tableau. The ODBC driver can add a data source such as MySQL and then connect with Tableau.

.

Method 3: Exporting MySQL data as CSV

One unique way to connect MySQL to Tableau is to export data from MySQL in CSV format and then use the Tableau text connector to analyze the data

Method 1 - Using Tableau's MySQL connector

Tableau's in-built MySQL connector conveniently connects MySQL to Tableau. You can easily set up a data source and use Tableau to visualize your data to perform a fruitful analysis for your business. This method requires you to install the MySQL ODBC driver. The ODBC driver allows tableau to interact with data stored in your MySQL database and hence perform analysis easily. The MySQL driver comes installed by default with Tableau v9.3 and above. Users supporting any previous versions of their system can download the driver from the official MySQL site here.

Procedure:

Step 1: Configuring Tableau

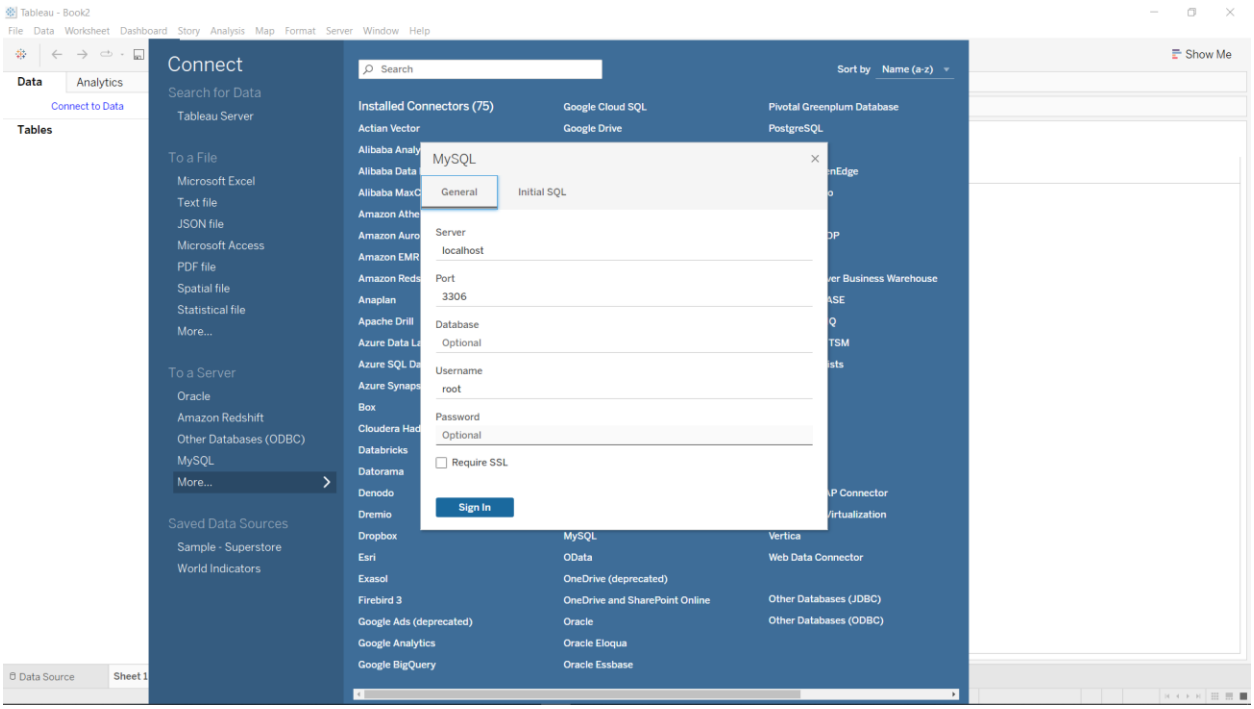
Step 2: Configuring Data Source

Step 3: Performing Analysis Using Tableau

Step 1: Configuring Tableau

Launch Tableau on your workstation and select MySQL from the connect column on the left. This will open a dialogue box where you need to provide the connection details for MySQL. To connect with tableau, you will need to provide information about the server

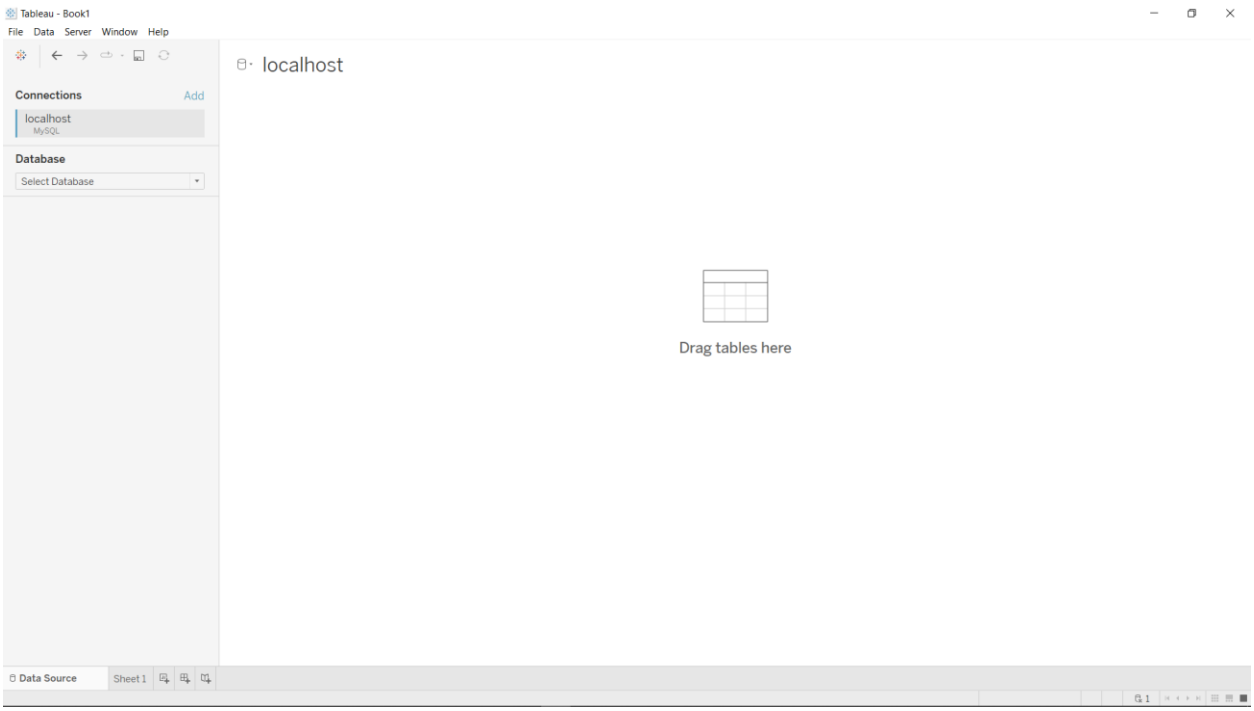
which hosts your database. If you want to connect to a contained database, you can also specify the name of the database.



Click on Sign in to establish a connection. This will enable a connection without SSL. To establish an SSL enabled connection, click the Require SSL checkbox before you sign in. If you want to run a specific SQL command every time a new connection is established, you can use the Initial SQL option. This will open a dialogue box, where you can specify your desired SQL query.

Step 2: Configuring Data Source

The data source page loads up after configuring the Tableau connector and successfully signing in. This is how the page looks like:



Select the data source name option and give a unique name to the database you are using. It's considered a good practice to have a unique name as it makes it much easier for users to identify the database from which data is being fetched.

To select the desired schema, you can use the schema drop-down list from the column on the left. You can also perform a text-based search to find the desired option. Now similarly find and select the desired table and drag it onto the canvas. This is how you can connect MySQL with Tableau. Now click on the sheets tab to begin the analysis. Custom SQL features can be used to focus on specific SQL statements, rather than querying the entire database. Click on the Custom SQL option from the panel on the left. A new dialogue box will now open up, where you can provide the query you want to execute.

Step 3: Performing Analysis Using Tableau

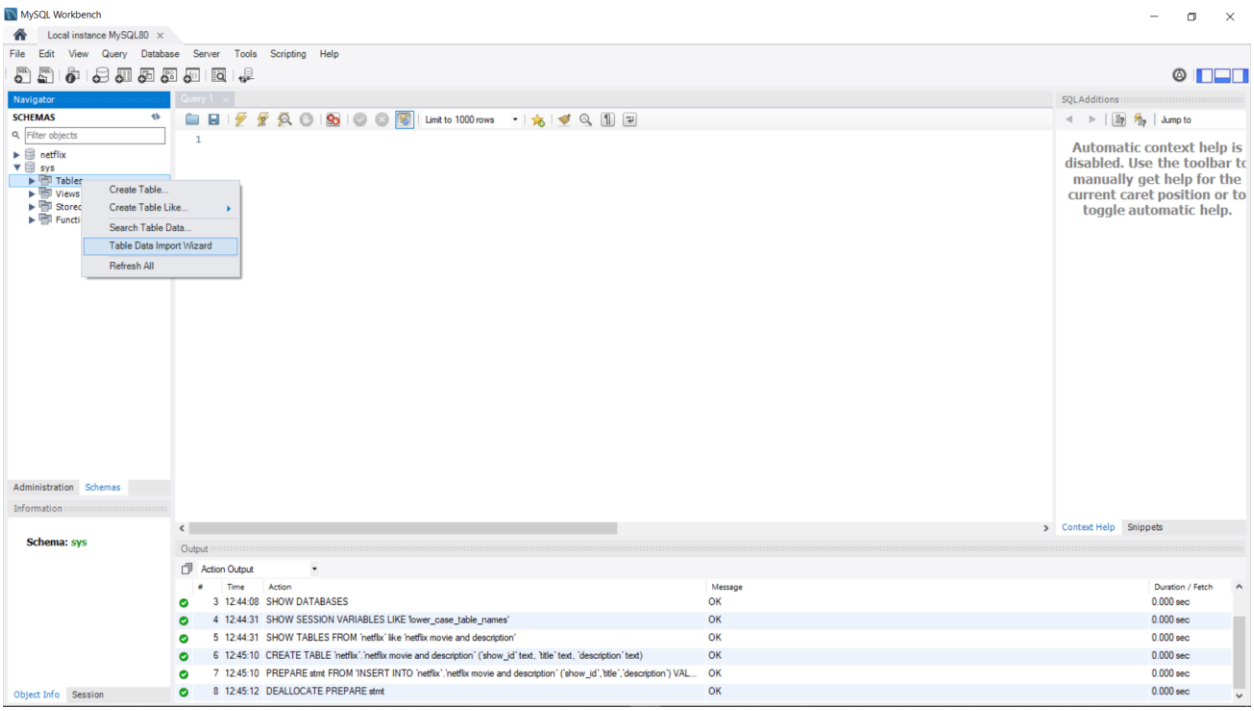
To visualize the data, click on the data option from the desktop menu bar and then select a table from the list of data sources (usually found at the bottom of the pop-up).

Here we are selecting a table called netflix to visualize the data. Now from the menu bar, select dashboard and create a new dashboard. This will generate a simple bar graph for your data.

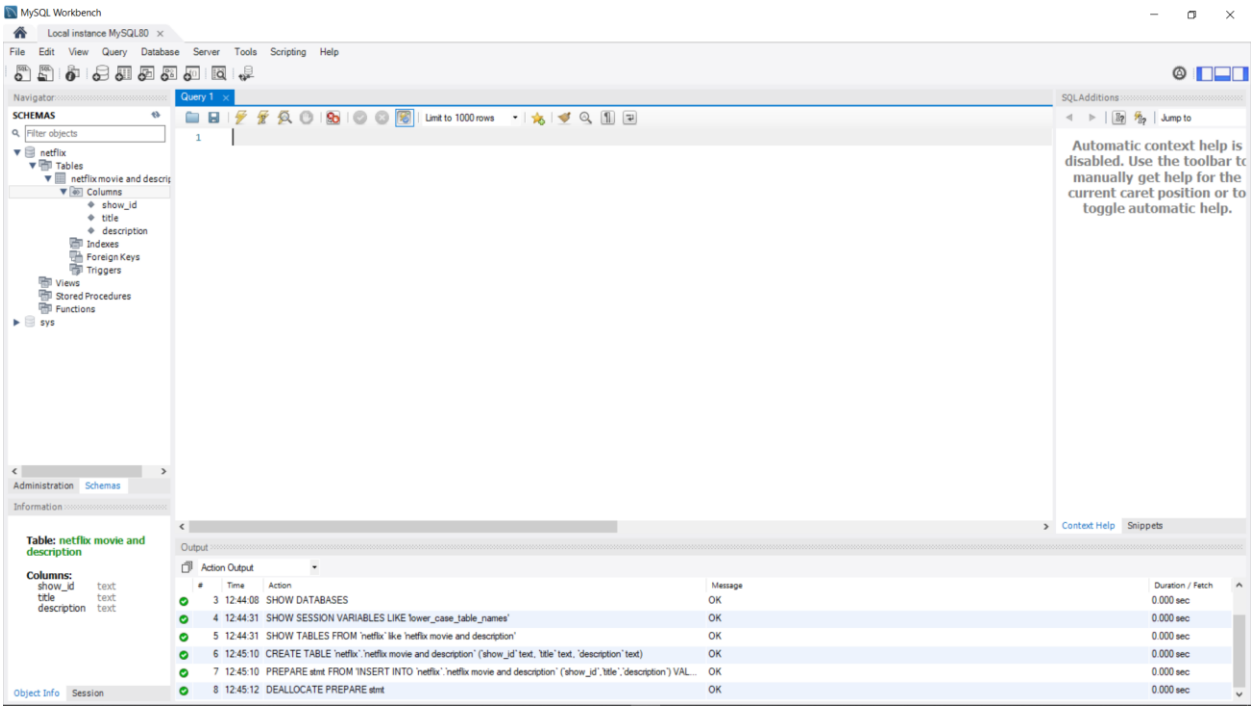
Connect to a SQL Query with MySQL Workbench:

MySQL Workbench is a unified visual tool for database architects, developers, and DBAs. MySQL Workbench provides data modeling, SQL development, and comprehensive administration tools for server configuration, user administration, backup, and much more.

1. Create a new Schema by opening the tables dropdown and right click to import data from the external source.

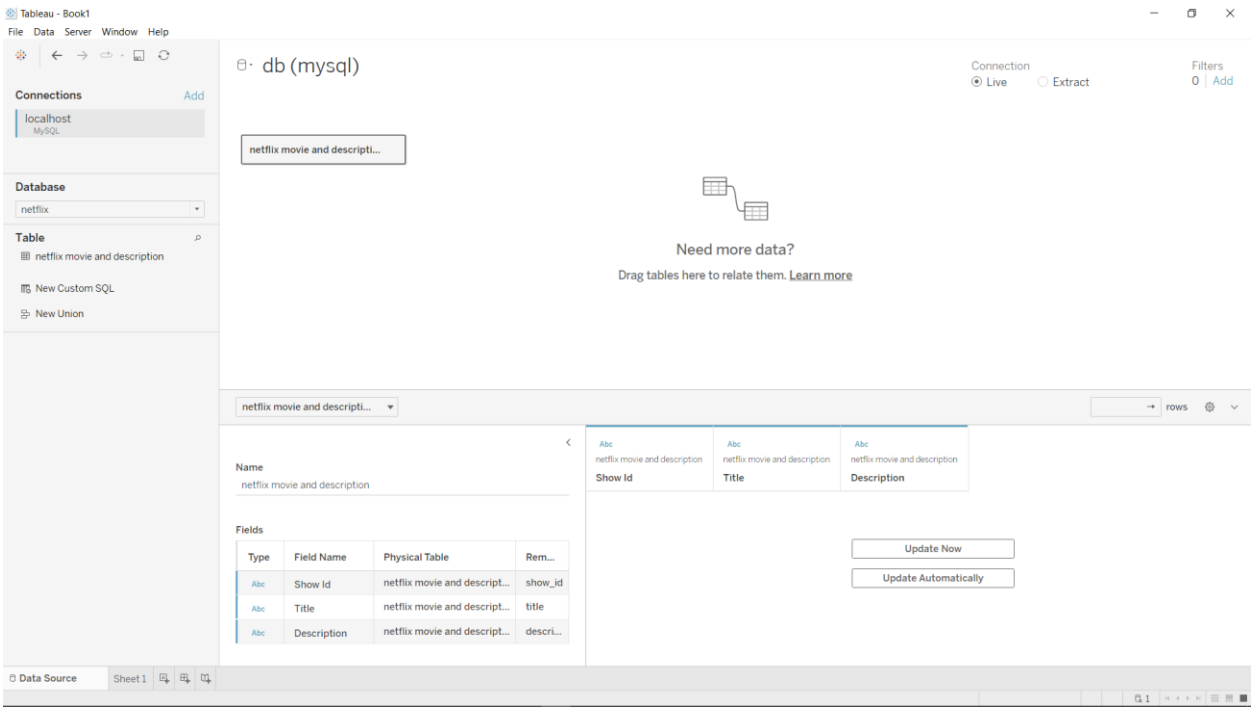


2. After importing the data refresh the database to see the new table which we have created by clicking the Refresh button by right-clicking the database and selecting refresh all.



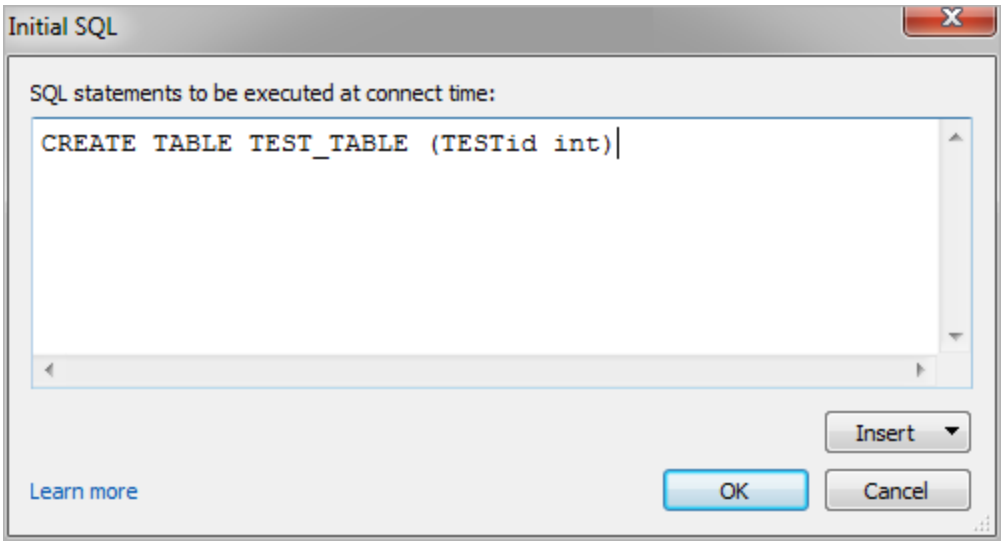
3. When finished, go to tableau software, after opening the MySQL page select the table and click OK.

When you click OK, the query runs and the custom SQL query table appears in the logical layer of the canvas. Only relevant fields from the custom SQL query are displayed in the data grid on the Data Source page.



Run Initial SQL

When connecting to some databases, you can specify an initial SQL command that will run when a connection is made to the database, for example, when you open the workbook, refresh an extract, sign in to Tableau Server, or publish to Tableau Server. Initial SQL is not run when your refresh your view. Note that this initial SQL is different than a custom SQL connection. A custom SQL connection defines a relation (or table) to issue queries against. For more information, see Connect to a Custom SQL Query.



You can use this command to

- Set up temporary tables to use during the session.
- Set up a custom data environment.
- You have the option to add an initial SQL command in the Server Connection dialog box or on the Data Source page.

To use initial SQL

In the Server Connection dialog box, click Initial SQL. Or, on the Data Source page, select Data > Initial SQL or Data > Query Banding and Initial SQL depending on the database you connect to.

Enter the SQL command into the Initial SQL dialog box. You can use the Insert drop-down menu to pass parameters to your data source. Your software license may restrict you from using initial SQL with your connection. If you publish to Tableau Server, the server must be configured to allow Initial SQL statements. By default, the server software is configured to allow these statements to run when the workbook is loaded in a web browser.

Parameters in an initial SQL statement

- You can pass parameters to your data source in an initial SQL statement. There are several reasons why this is useful:

- You can configure impersonation using the TableauServerUser or TableauServerUserFull parameters.
- If your data source supports it, you can set up row-level security (for example, for Oracle VPD or SAP Sybase ASE) to make sure that users see only the data that they are authorized to see.
- You can provide more details in logging, for example, the Tableau version or the workbook name.

The following parameters are supported in an initial SQL statement:

Parameter	Description	Example of returned value
TableauServerUser	The user name of the current server user. Use when setting up impersonation on the server. Returns an empty string if the user is not signed in to Tableau Server.	jsmith
TableauServerUserFull	The user name and domain of the current server user. Use when setting up impersonation on the server. Returns an empty string if the user is not signed in to Tableau Server.	domain.lan\jsmith
TableauApp	The name of the Tableau application.	Tableau Desktop Professional Tableau Server
TableauVersion	The version of the Tableau application.	9.3
WorkbookName	The name of the Tableau workbook. Use only in workbooks with an embedded data source.	Financial-Analysis

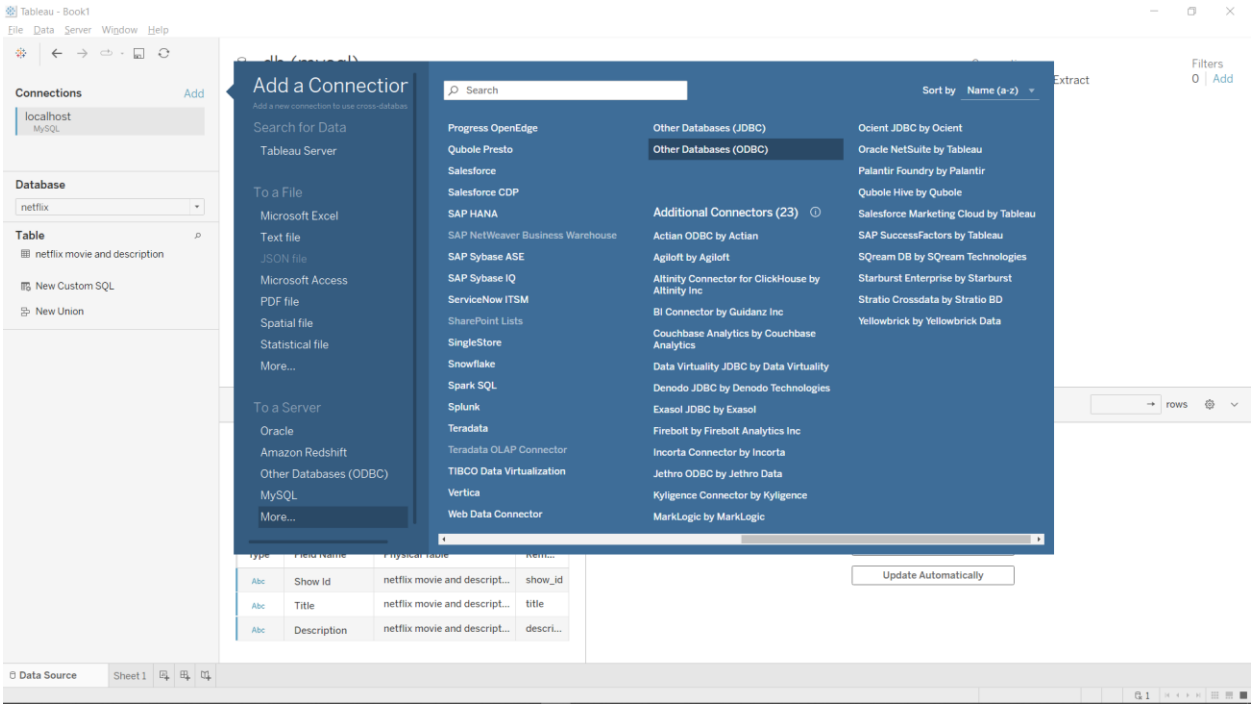
Method 2: Using ODBC to connect MySQL to Tableau

ODBC drivers can also be used to connect MySQL to Tableau. The ODBC driver can add a data source such as MySQL and then connect with Tableau.

Procedure:

Step 1: Installing MySQL ODBC Driver and DSN Setup

Step 2: Connecting to Tableau

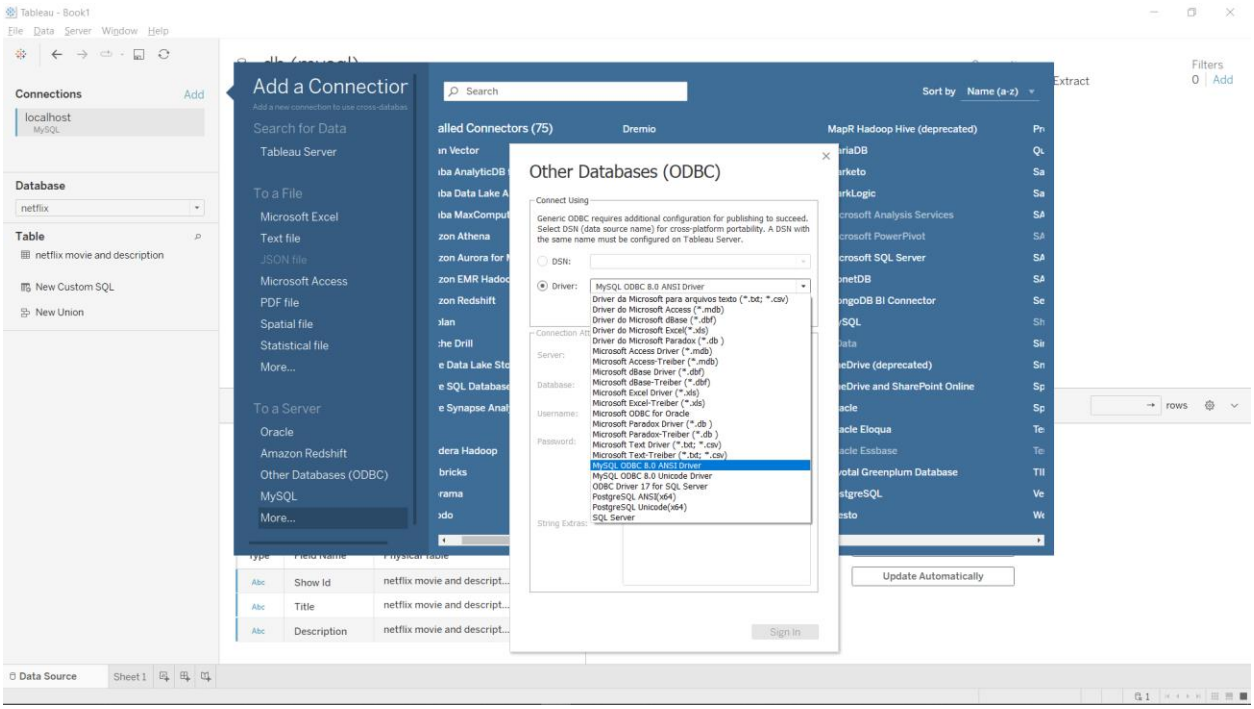


Step 1: Installing MySQL ODBC Driver and DSN Setup

Once the ODBC driver is installed on your system, launch it to start the DSN set up process. Choose the correct 64 or 32-bit version and select the System DSN tab.

MySQL to Tableau: ODBC Connector,

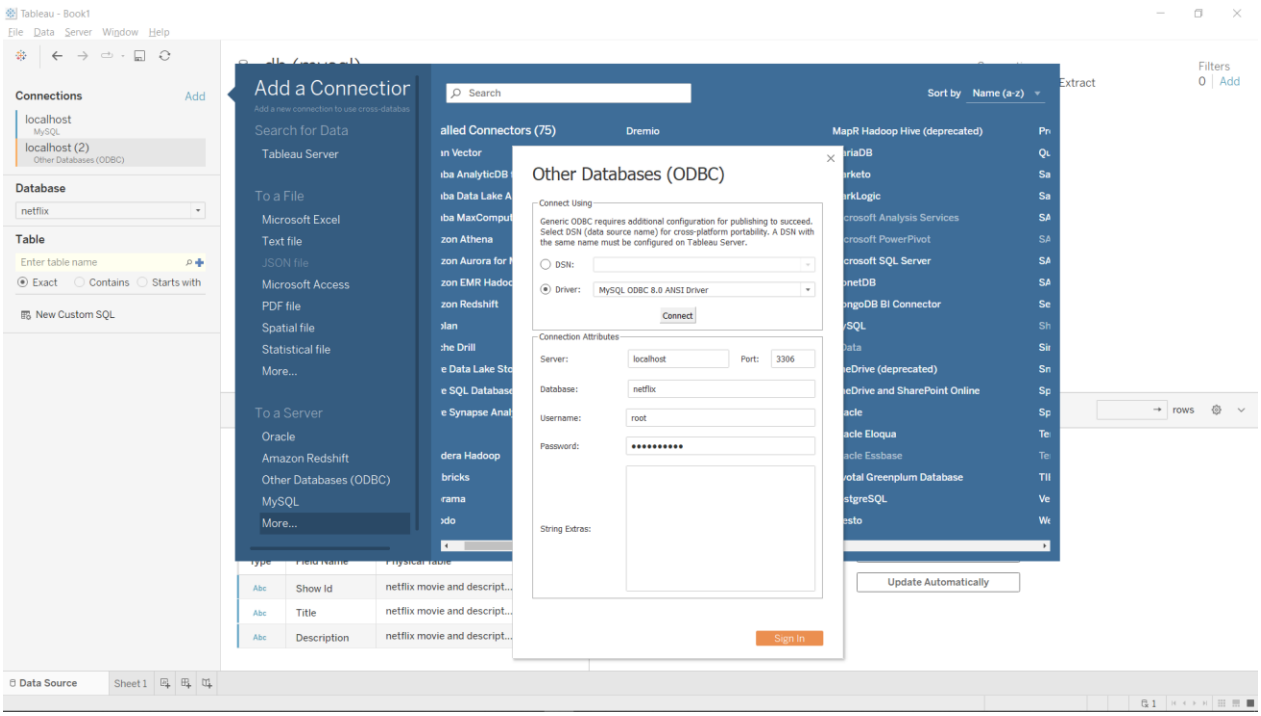
Click on the add button and select the MySQL ODBC driver. You can either select the MySQL ODBC 8.0 ANSI driver or the MySQL ODBC 8.0 UNICODE driver.



The ANSI driver delivers the best performance but has a limited character set. For the entire character set, use the UNICODE option.

MySQL to Tableau: Selecting MySQL Unicode ODBC Driver.

The MySQL DSN setup box will now appear, where you need to fill in details regarding the connection settings and authentication:



MySQL to Tableau: Configuring MySQL Connection.

Fill in the necessary fields:

Data Source: Give the name of the data source you want to connect to.

TCP/IP Server: Give the network name or IP address of the MySQL server.

Port: MySQL uses port 3306 by default, however, you can fill in the correct port number as per your MySQL cluster.

Database: Give the name of your database.

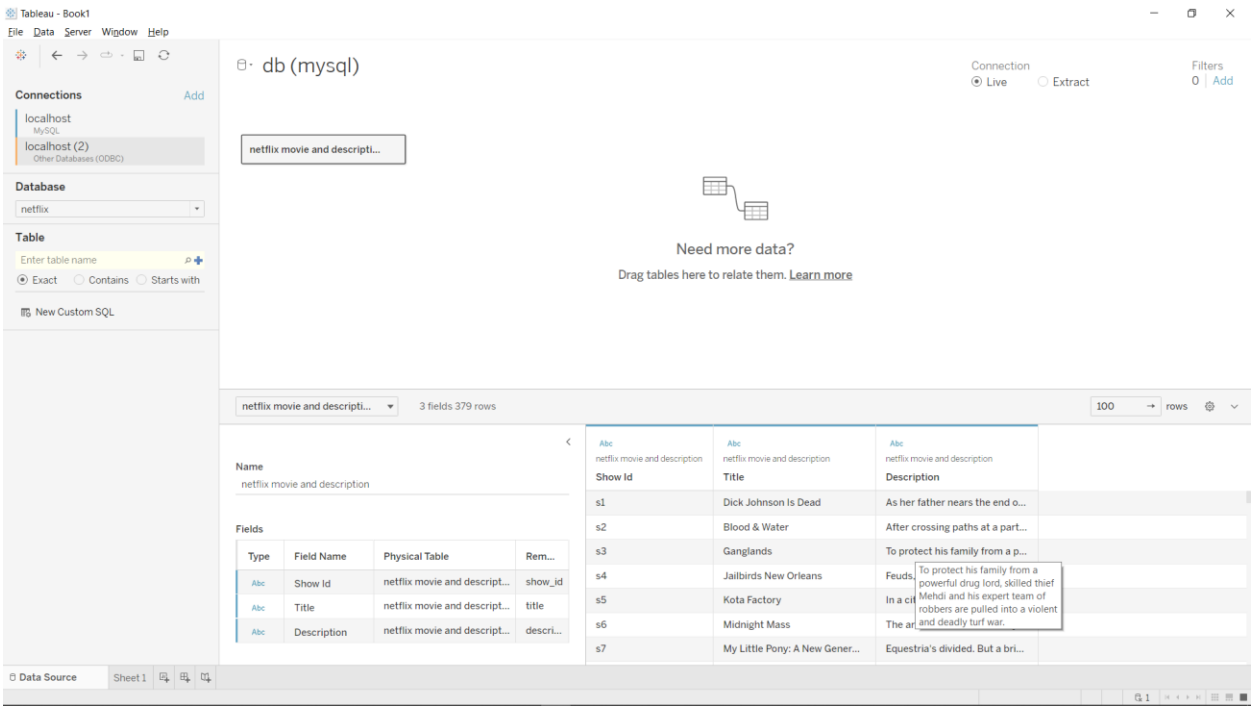
User: Enter the username in the correct format to connect with the MySQL database.

Passwords: Enter your password here.

Adding Database name & Password.

To test the connection, click on the test button. If the connection is successful, a new dialogue box will open up as follows:

Successful connection.



Step 2: Connecting to Tableau

Launch Tableau on your workstation and select other databases (ODBC) from the connect column on the left. This will open a dialogue box where you need to provide the connection details.

MySQL to Tableau: Selecting the ODBC Connector.

In the ODBC dialogue box, select the MySQL DSN from a drop-down list and click on ok.

Selecting the ODBC connector.

Select the data source name option and give a unique name to the database you are using. It's considered a good practice to have a unique name as it makes it much easier for users to identify the database from which data is being fetched.

Connecting MySQL to Tableau.

To select the desired schema, you can use the schema drop-down list from the column on the left. You can also perform a text-based search to find the desired option. Now similarly find and select the desired table and drag it onto the canvas.

This is how you can use ODBC to connect MySQL to Tableau to run the powerful analysis.

Method 3: Exporting MySQL data as CSV

One unique way to connect MySQL to Tableau is to export data from MySQL in CSV format and then use the Tableau text connector to analyze the data.

This can be done using 2 simple steps:

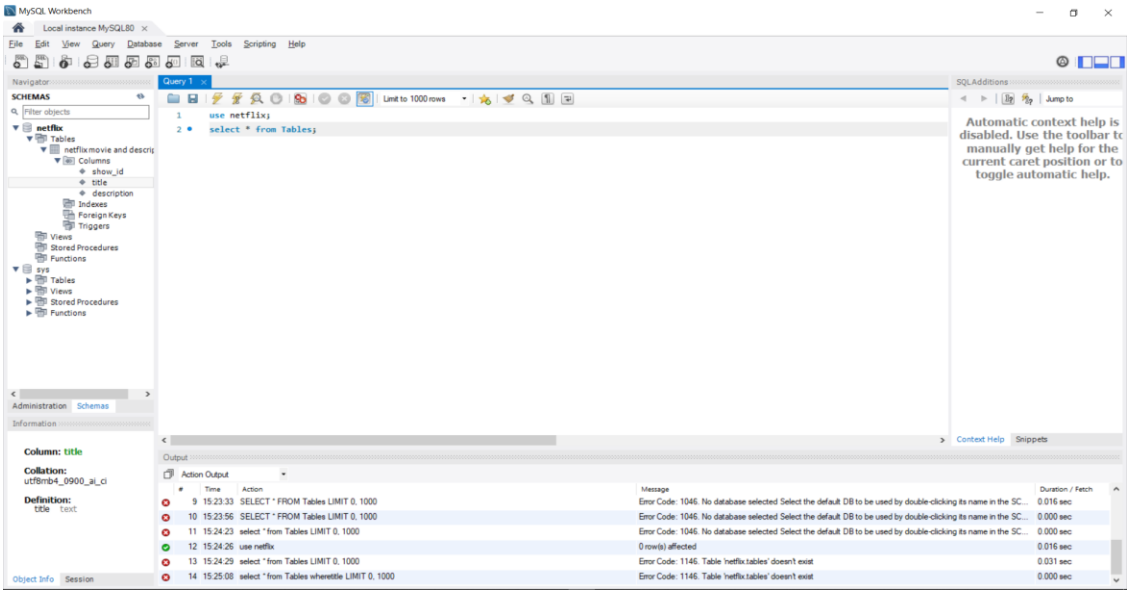
Step 1: Export data in CSV using MySQL workbench

Step 2: Using Tableau's Text connector

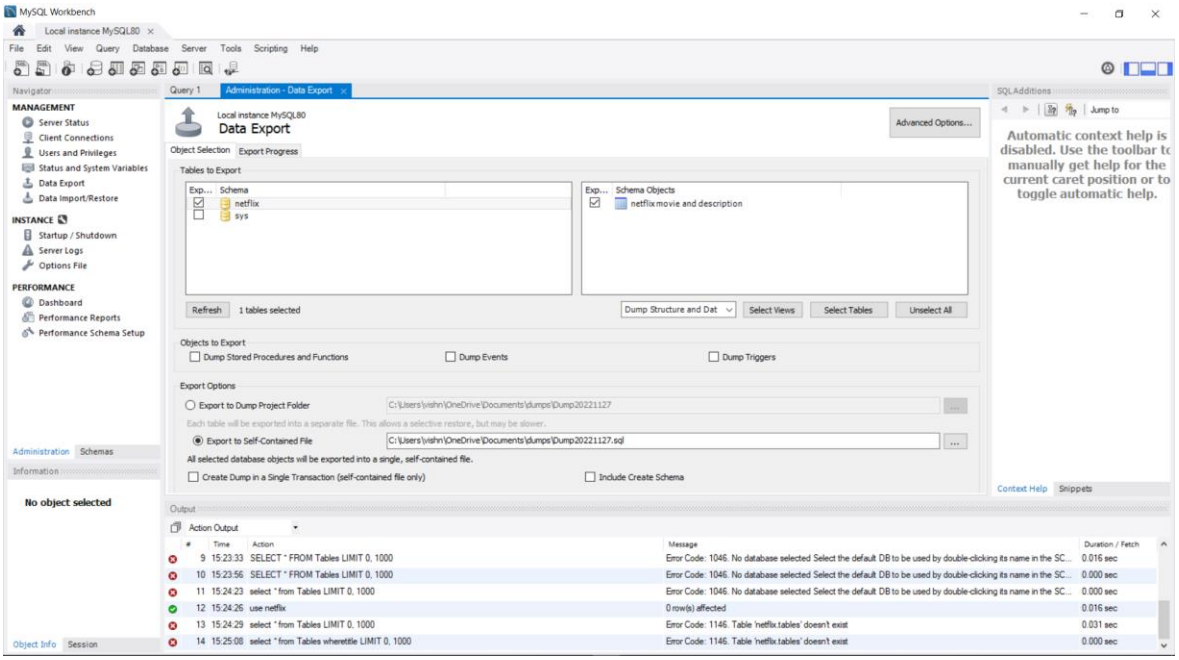
Step 1: Export data as CSV using MySQL workbench

Launch MySQL workbench on your system and open the desired project. MySQL allows exporting data in the desired format using the table data export wizard.

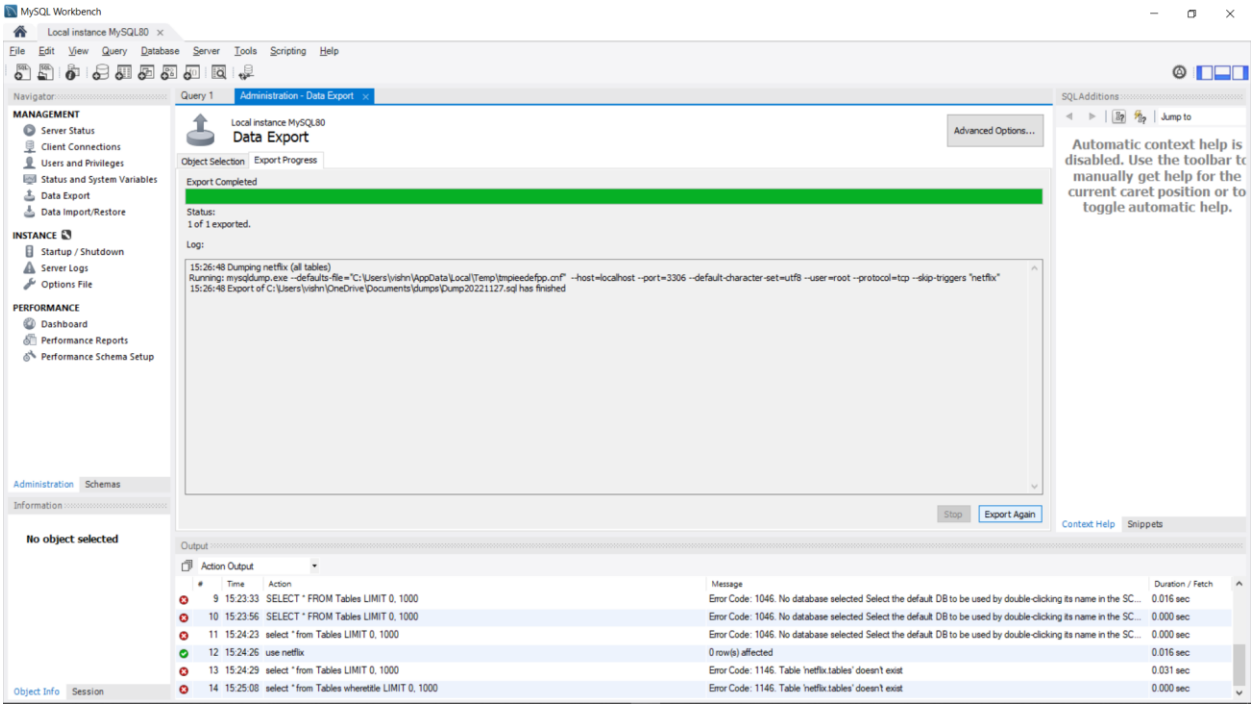
Select the table you want to export using the navigator option and right-click on it. Select the table data export wizard option from the list of available options to begin the process.



A new dialogue box will now open up, representing the table and the columns that you want to export. Select the objects you want to export



In the export section, click start export to export the files



Result:

The data from the external dataset is successfully integrated with MySQL and Tableau to provide easier approach to use data faster from SQL queries in Tableau. The output is shown and MySQL is successfully integrated with Tableau