

TRAINING & REFERENCE

murach's MySQL

**3RD
EDITION**

(Chapter 2)

Thanks for downloading this chapter from [Murach's MySQL \(3rd Edition\)](#). We hope it will show you how easy it is to learn from any Murach book, with its paired-pages presentation, its “how-to” headings, its practical coding examples, and its clear, concise style.

To view the full table of contents for this book, you can go to our [website](#). From there, you can read more about this book, you can find out about any additional downloads that are available, and you can review our other books on related topics.

Thanks for your interest in our books!



MIKE MURACH & ASSOCIATES, INC.

1-800-221-5528 • (559) 440-9071 • Fax: (559) 440-0963

murachbooks@murach.com • www.murach.com

Copyright © 2019 Mike Murach & Associates. All rights reserved.

What developers have said about the previous editions

“If you ever wanted to learn to use MySQL, write SQL queries, create database elements, then this is the book to pick up. Rating: 10 Horseshoes.”

Review by Mohamed Sanoulla, JavaRanch.com

“A great first book into SQL: From all the SQL books I looked over, this has by far the best division of chapters and the best order for learning.”

Posted at an online bookseller

“As a developer with almost 10 years of MySQL experience, I still picked up a lot of new detail on things I thought I knew. Every development shop that works with MySQL should have a copy of this book.”

David Bolton, C/C++/C# Guide, About.com

“This book was the text in my Database Concepts class, and I am so thankful that it was! It provided excellent explanations and examples of database syntax, queries, subqueries, design, etc. I aced the class and decided to take more database classes because of this book.”

Posted at an online bookseller

“This book is not just up-to-date with the latest information on MySQL, but it is extremely easy to read and learn from. It only took me a couple weeks to rip through it!”

Posted at an online bookseller

“I was amazed at how much information was packed into this book. I learned a lot of new MySQL ideas, and I will be using it frequently as a reference.”

Paul Turpin, Southeastern Inter-Relational Database Users Group

How to use MySQL Workbench and other development tools

In the last chapter, you learned about some of the SQL statements that you can use to work with the data in a relational database. Before you learn the details of coding these statements, however, you need to learn how to use MySQL Workbench to enter and execute SQL statements. In addition, you should learn how to use the MySQL Reference manual, and you should at least be familiar with the MySQL Command Line Client.

An introduction to MySQL Workbench	42
The Home page of MySQL Workbench	42
How to open a database connection	44
How to start and stop the database server	46
How to navigate through the database objects	48
How to view and edit the data for a table	50
How to view and edit the column definitions for a table	52
How to use MySQL Workbench to run SQL statements	54
How to enter and execute a SQL statement	54
How to use snippets	56
How to handle syntax errors	58
How to open and save SQL scripts	60
How to enter and execute SQL scripts	62
How to use the MySQL Reference Manual	64
How to view the manual	64
How to look up information	64
How to use the MySQL Command Line Client	66
How to start and stop the MySQL Command Line Client	66
How to use the MySQL Command Line Client to work with a database	68
Perspective	70

An introduction to MySQL Workbench

MySQL Workbench is a free graphical tool that makes it easy to work with MySQL. We recommend using this tool as you work through this book. This chapter shows how to work with version 8.0. However, with some minor variations, the skills presented in this chapter should work for later versions as well.

The Home page of MySQL Workbench

When you start MySQL Workbench, it displays its Home page as shown in figure 2-1. This page is divided into three tabs: Welcome, Models, and Migration.

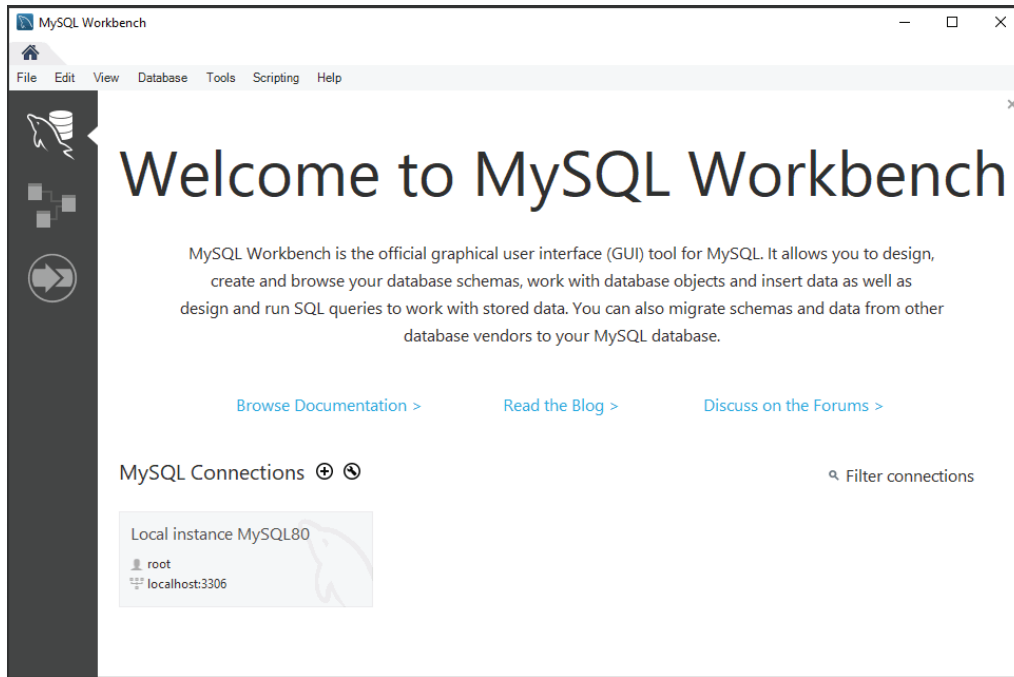
The MySQL Connections section of the Welcome tab contains links that you can use to open a connection to a MySQL server. Then, you can use that connection to code and run SQL statements. By default, this tab contains one connection that allows you to connect as the root user to a MySQL server that's running on the local computer. In this book, this is the only connection you will need. However, if necessary, you can click the ⊕ icon to the right of MySQL Connections to create other connections.

The Welcome tab also contains links to MySQL Workbench documentation, blogs, and forums. This book doesn't show how to use these links, but you may find them useful, especially after you have learned the basic skills for working with MySQL that are described in this book.

The Models tab contains links that let you create a database diagram from a type of data model known as an EER model. You can also use this tab to open existing EER models or to create new ones. Then, you can work with EER diagrams that correspond with these models. To learn more about this, you can read chapter 10.

You can return to the Home page by clicking on the tab with the house icon on it near the top left corner of the Workbench window. In this figure, the Home tab is the only tab that's shown, but you'll see some other tabs in the next few figures.

The Home page of MySQL Workbench



Description

- The Home page of MySQL Workbench is divided into three tabs displayed at the left side of the window: Welcome, Models, and Migration.
- You can use the MySQL Connections section of the Welcome tab to start and stop the database server and to code and run SQL statements.
- You can use the links on the Welcome tab to view the documentation for using MySQL Workbench, view the MySQL Workbench blog, and view and join in the MySQL Workbench forum.
- You can use the Models tab to create and work with EER models.
- You can use the Migration tab to migrate other databases to MySQL and to copy a database from one instance of MySQL to another.
- You can return to the Home page by clicking the tab with the house icon. This tab is always displayed in the top left corner of the Workbench window.

Note

- In some cases, you'll get an "Unsupported Operating System" message when you start MySQL Workbench. This happens, for example, when you start MySQL Workbench 8.0 on Windows 7. If you click the OK button when this message is displayed, MySQL Workbench should work fine. This is a known bug that should be fixed in a future release of Workbench.

Figure 2-1 The Home page of MySQL Workbench

How to open a database connection

Before you can work with a database, you need to connect to the database server. When you start MySQL Workbench, the MySQL Connections section displays a list of saved connections.

By default, MySQL Workbench has one saved connection in this list. This connection is named “Local instance MySQL80”, and it connects as the root user to a MySQL server that’s running on port 3306 of the local host computer. (This assumes that you’re using MySQL version 8.0. If you’re using another version, the number at the end of the connection name will be different.)

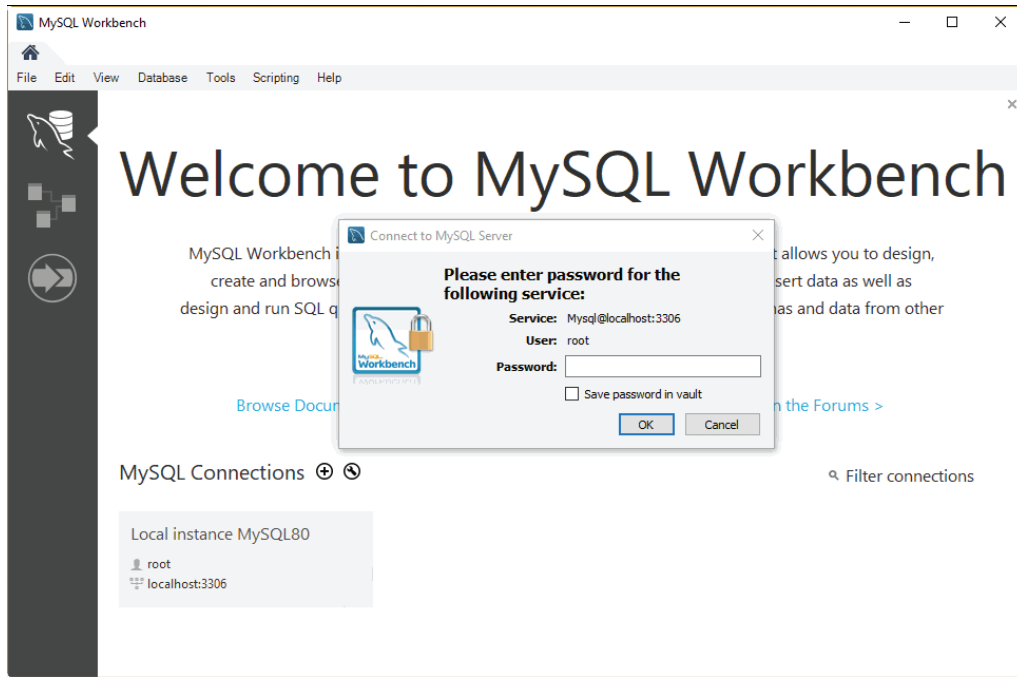
Since this is what you want when you’re first getting started, you typically use this connection to connect to the server. To do that, click the connection and enter the password for the root user if you’re prompted for it. If you installed MySQL Workbench following the directions in appendix A (Windows) or B (macOS), the password for the root user is “sesame80.”

Figure 2-2 shows the dialog box that MySQL Workbench displays to prompt for a password. This dialog box shows that it’s attempting to use the root user to connect to a MySQL server running on port 3306 of the local host. In addition to entering a password in this dialog box, you can select the “Save password in vault” option to save the password so you don’t have to enter it every time you connect to this server. Then, if you ever want to clear the password from the vault, you can right-click the connection, select the Edit Connection item, and click the Clear button.

If you need to connect as another user, or if you need to connect to a MySQL server running on a different computer, you can use MySQL Workbench to edit the connection parameters for a connection. To do that, right-click the connection and select the Edit Connections item. This displays a dialog box that lets you specify the parameters for the connection such as the username, hostname, and port number.

If you want to add a new connection to the Home tab, you can click the ⊕ icon to the right of MySQL Connections, enter a name for the connection, and specify the parameters for the connection. Then, this connection appears in the list of connections, and you can click it to use it.

The dialog box for opening database connections



Description

- To connect as the root user to an instance of MySQL that's running on the local host computer, click the stored connection named "Local instance MySQL80", and enter the password for the root user if prompted.
- To save the password for a connection so you don't have to enter it every time, check the "Save password in vault" option when you're prompted for your password.
- To clear the password from the vault so you are prompted for your password, right-click the connection, select the Edit Connection item, click the Clear button for the password, and click the Close button.
- To edit the connection parameters for a connection, right-click the connection, select the Edit Connection item, enter the connection parameters, and click the Close button. This lets you specify the username, the host address, the port number, and other connection parameters.
- To add a new connection to the Welcome tab of the Home page, click the ⊕ icon to the right of MySQL Connections, enter the connection parameters, and click the OK button. Then, the connection appears in the list of connections.

Figure 2-2 How to open a database connection

How to start and stop the database server

If you installed MySQL on your computer as described in appendix A (Windows) or B (macOS), the *database server* starts automatically when you start your computer. This piece of software is sometimes referred to as the *database service* or *database engine*. It receives SQL statements that are passed to it, processes them, and returns the results.

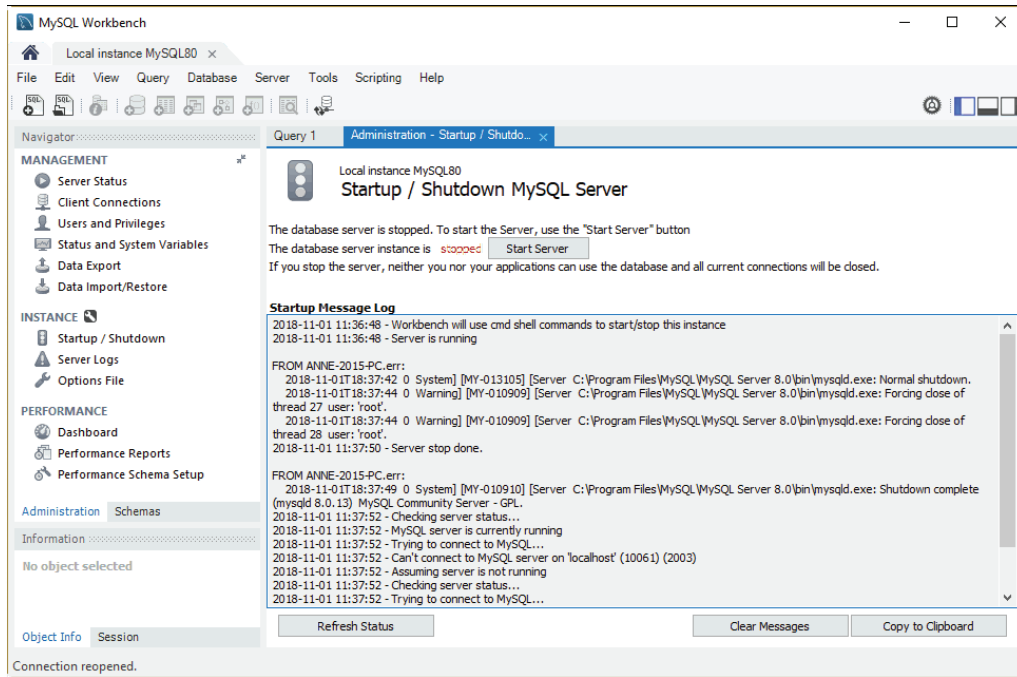
Before you can work with a MySQL database, the database server must be started. To check whether the MySQL database server is running on your computer, you can use the Startup/Shutdown option of MySQL Workbench as shown in figure 2-3. Then, if the server isn't already running, you can start it by clicking on the Start Server button. When you do that, MySQL Workbench displays a message that indicates the status of the MySQL server, and it displays the Stop Server button.

You may also want to stop the database server from time to time. For example, you can stop the server if you aren't going to be using it and you want to free the resources on your computer. Or, you can stop the server if the port that is being used by the MySQL database server conflicts with another program. Then, when you want to work with the database server again, you can start it.

The easiest way to stop the database server is to use the Stop Server button that's available from the Startup/Shutdown option of the Navigator window of MySQL Workbench as described in this figure. When you click this button, MySQL Workbench displays a message when the MySQL server has successfully stopped, and it displays the Start Server button.

When you're running the MySQL database server on your own computer for training purposes, you can stop the database server whenever you want. However, if a database server is running in a production environment, you should make sure that all users are logged off and that no applications are using the database server before you stop it.

The Startup/Shutdown option of MySQL Workbench



How to stop and start the database server

1. Display the Welcome tab of the MySQL Workbench Home page.
2. Click the connection to the local server. This should connect you to the local MySQL server as the root user. If necessary, enter the password for the root user.
3. In the Navigator window, if necessary, click on the Administration tab. Then, select the Startup/Shutdown option from the Instance category.
4. Click the Stop Server button to stop the database server. Or, click the Start Server button to start it.

Description

- After you install MySQL, the *database server* usually starts automatically each time you start your computer.
- The database server can also be referred to as the *database service* or the *database engine*.
- If you aren't able to use Workbench to start and stop the database server, you may need to edit your connection so it points to the correct instance of MySQL. To do that, right-click the connection on the Welcome tab of the Home page, select the Edit Connection item, click the System Profile tab, and edit the service name. For MySQL 8.0 on Windows, the service name is typically "MySQL80".

Figure 2-3 How to start and stop the database server

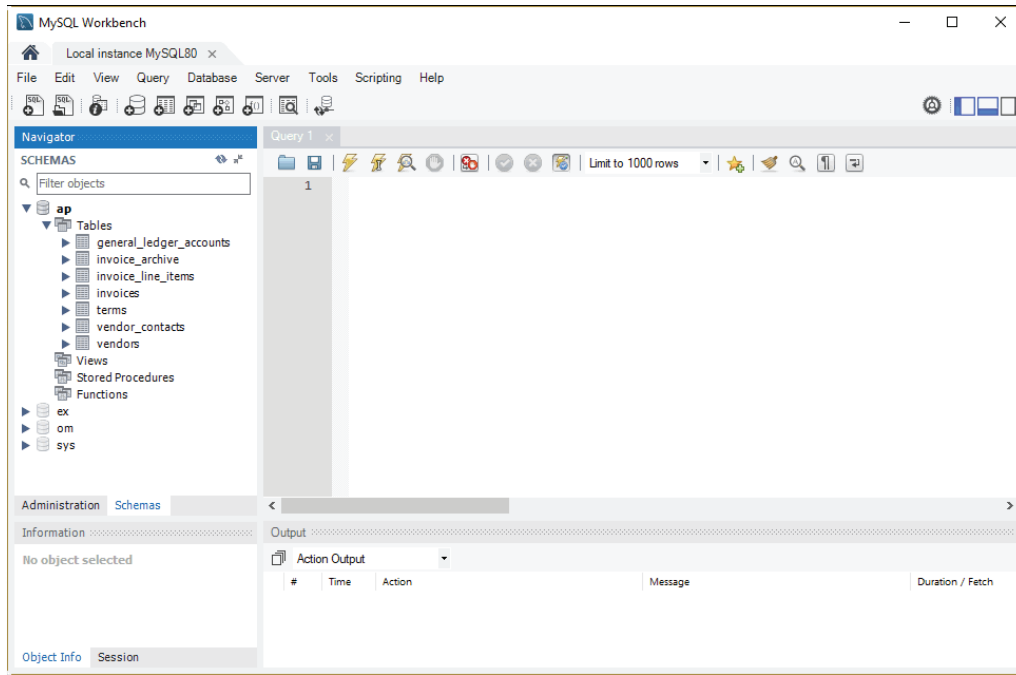
How to navigate through the database objects

After you connect to a database server, you can use the Schemas category of the Navigator window to navigate through the *database objects* in the databases on the server, as shown in figure 2-4. As you can see, these objects include tables, views, stored procedures, and functions. For this chapter, however, you can focus on the tables. Later in this book, you'll learn more about views, stored procedures, and functions.

In this figure, I double-clicked the node for the AP database (*schema*) in the Schemas tab of the Navigator window to select it and view the database objects it contains (tables, views, stored procedures, and functions). Then, I expanded the Tables node to view all of the tables in the AP database.

To work with a node or an object, you can right-click it to display a context-sensitive menu. Then, you can select a command from that menu. For example, you can right-click the node for the AP database to display a list of commands for working with that database.

The tables available for the AP database



Description

- Each database (or *schema*) provides access to the *database objects* that are available. These database objects include tables, views, stored procedures, and functions.
- On some systems, the Navigator window provides Administration and Schemas tabs that you can use to display the Administration and Schemas categories. On other systems, the Navigator window displays the Administration category above the Schemas category.
- To display the databases for the current connection, you can use the Navigator window to view the Schemas category.
- To navigate through the database objects for a database, click the arrows to the left of each of the nodes in the Navigator window to expand or collapse the node.
- To work with a node or an object, right-click the node or object and select a command from the resulting menu.

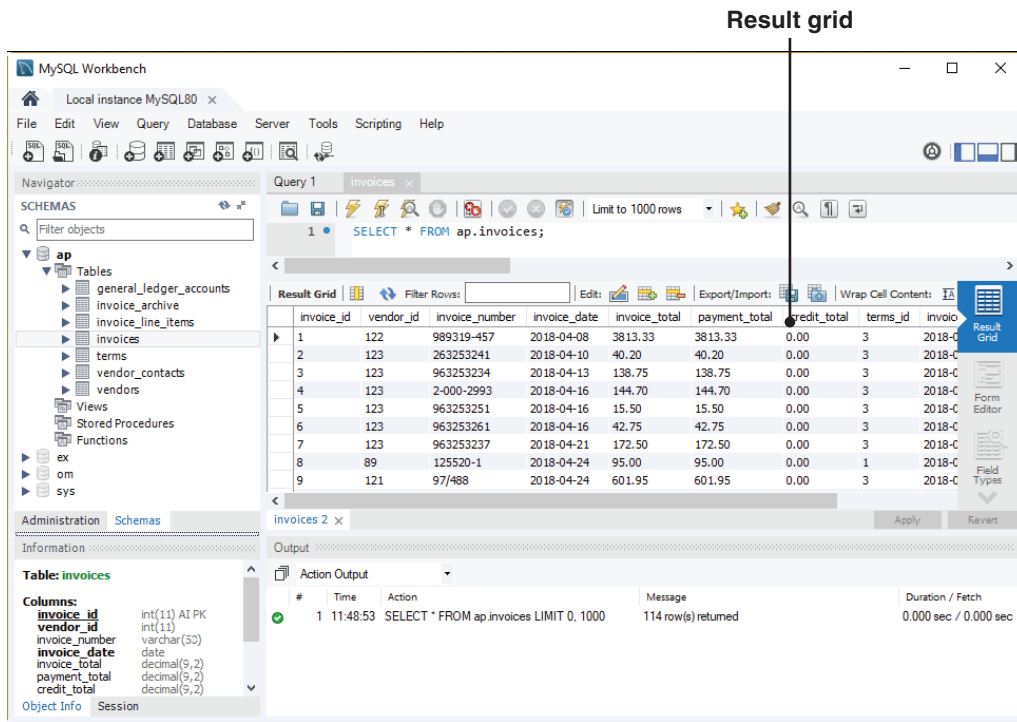
Figure 2-4 How to navigate through the database objects

How to view and edit the data for a table

To view the data for a table, you can right-click the table name and select the `Select Rows - Limit 1000` command. In figure 2-5, for example, I selected this command for the `Invoices` table. This displayed the data for the table in a Result grid. In addition, it displayed information about the `SELECT` statement that was used to retrieve the data in the Output tab.

To insert, edit, and delete the rows in the table, you can use the buttons at the top of the Result grid. Then, to apply the changes to the table, you can click the `Apply` button at the bottom of the Result grid. Or, if you want to cancel the changes, you can click the `Revert` button.

The data for the Invoices table displayed in the Result grid



Description

- To view the data for a table, right-click the table in the Navigator window and select the Select Rows - Limit 1000 command to display it in a Result grid.
- To edit the data for a table, view the data. Then, you can use the buttons at the top of the Result grid to insert, update, and delete rows.
- To apply the changes to the table, click the Apply button at the bottom of the tab. To cancel the changes, click the Revert button.

Figure 2-5 How to view and edit the data for a table

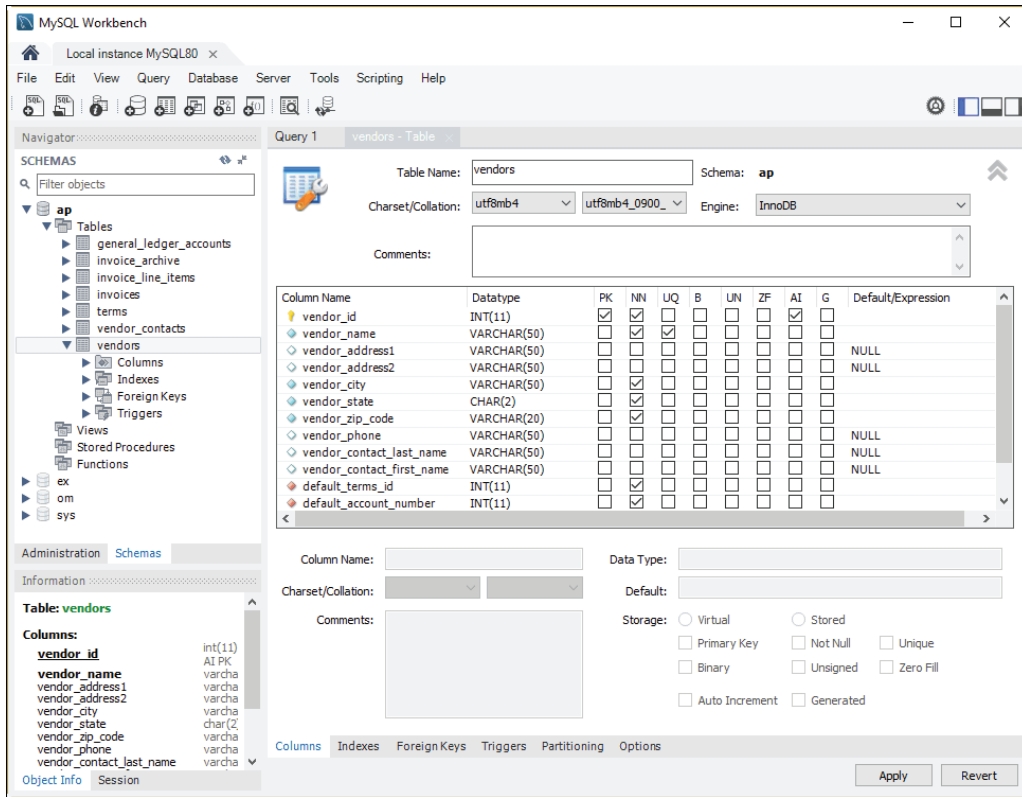
How to view and edit the column definitions for a table

If you want to edit a column definition for a table, you can use the technique described in figure 2-6 to display the column definitions for the table. In this figure, for example, the column definitions for the Vendors table are displayed. At this point, you can view information about each column of the table such as its name and data type.

Once you display the column definitions for a table, you can use the Columns tab to add a column, delete a column, or modify a column. For example, you can add a new column by entering it at the bottom of the list. You can delete a column by right-clicking on it and selecting the Delete command. You can change the name of a column by selecting the column and then clicking on the name and editing it. You can change the data type of a column by selecting the column and then clicking on its data type and selecting another data type from the drop-down list that appears. And so on.

Most of the time, you won't want to use MySQL Workbench to edit the column definitions for a table. Instead, you'll want to edit the scripts that create the database so you can easily recreate the database later. In chapter 11, you'll learn more about creating and modifying the column definitions for a table using both techniques.

The column definitions for the Vendors table



Description

- To view the column definitions for a table, right-click the table name in the Navigator window and select the Alter Table command. Then, select the Columns tab at the bottom of the window that's displayed to view the column definitions for the table.
- To edit the column definitions for a table, view the column definitions. Then, you can use the resulting window to add new columns and modify and delete existing columns.
- For more information about creating and modifying tables, see chapter 11.

Figure 2-6 How to view and edit the column definitions

How to use MySQL Workbench to run SQL statements

Besides letting you review the design of a database, MySQL Workbench is a great tool for entering and running SQL statements.

How to enter and execute a SQL statement

When you first connect to a MySQL server in MySQL Workbench, a SQL Editor tab is automatically opened. Figure 2-7 shows how to use the SQL editor to enter and execute a SQL statement. The easiest way to open a SQL Editor tab is to click the Create New SQL Tab button in the SQL Editor toolbar or press the Ctrl+T keys.

Once you open a SQL tab, you can use standard techniques to enter or edit a SQL statement. As you enter statements, you'll notice that MySQL Workbench automatically applies colors to various elements. For example, it displays keywords in blue. This makes your statements easier to read and understand and can help you identify coding errors.

To execute a single SQL statement like the one in this figure, you can press Ctrl+Enter or click the Execute Current Statement button in the SQL Editor toolbar. If the statement returns data, that data is displayed below the SQL editor in a corresponding Result grid. In this figure, for example, the result set returned by the SELECT statement is displayed. If necessary, you can adjust the height of the Result grid by dragging the bar that separates the SQL Editor tab from the Result grid.

Before you execute a SQL statement, make sure you've selected a database by double-clicking the database in the Navigator window. Otherwise, you'll get an error message like this:

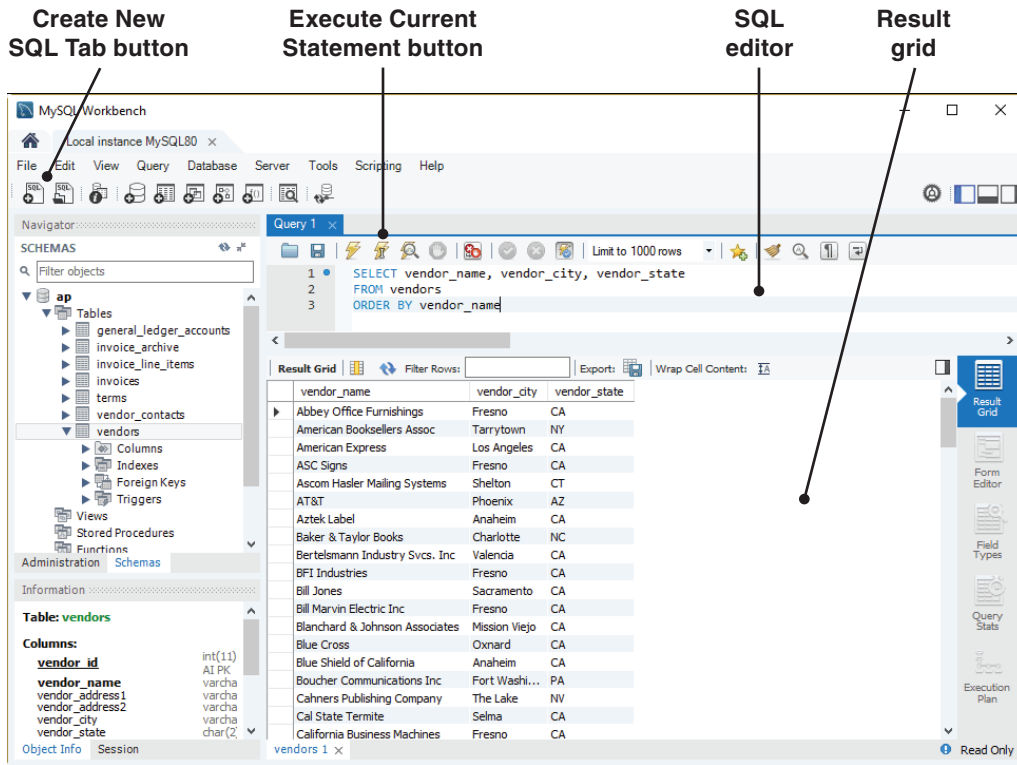
Error Code: 1046. No database selected

Similarly, if you haven't selected the correct database, you'll get an error message that says the table doesn't exist. For example, if the EX database is selected when you attempt to retrieve data from the Vendors table, you'll get an error message like this:

Error Code: 1146. Table 'ex.vendors' doesn't exist

To fix this, you can double-click the AP database to select it.

A SELECT statement and its results



Description

- To open a new SQL tab, press Ctrl+T or click the Create New SQL Tab button (📄) in the SQL Editor toolbar.
- To select the current database, double-click it in the Schemas tab of the Navigator window. This displays the selected database in bold.
- To enter a SQL statement, type it into the SQL editor.
- As you enter the text for a statement, the SQL editor applies color to various elements, such as SQL keywords, to make them easy to identify.
- To execute a SQL statement, press Ctrl+Enter or click the Execute Current Statement button (⚡) in the SQL Editor toolbar. If the statement retrieves data, the data is displayed in a Result grid below the SQL tab.

Figure 2-7 How to enter and execute a SQL statement

How to use snippets

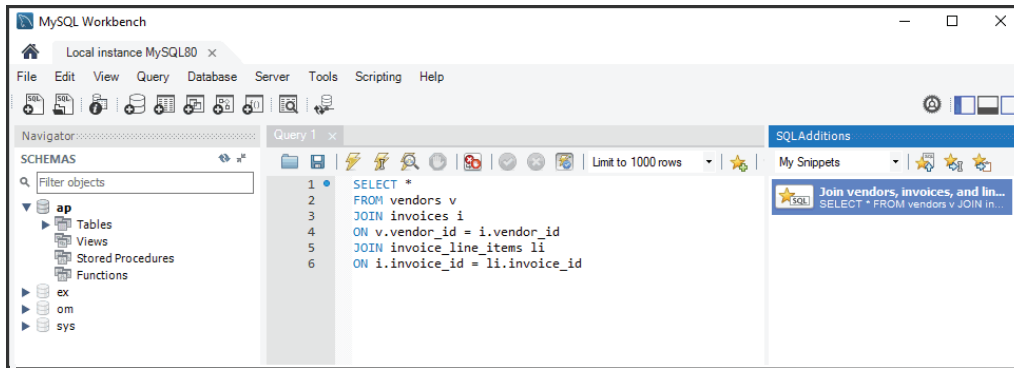
You can think of the *snippets* that come with MySQL Workbench as a library of SQL syntax. This library is divided into statements that you can use to manage a database, define objects in a database, and manipulate the data in a database. You can also create your own snippets that provide custom code. In fact, you're more likely to create your own snippets than you are to use the built-in snippets. That's because the syntax that's provided for the built-in snippets is much more complex than what you typically need.

Figure 2-8 shows how to use snippets. To start, if the SQL Additions window isn't displayed, you can display it by clicking on the rightmost button at the right side of the SQL Editor tab. Then, you can display the snippets tab and use the drop-down list at the top of the tab to select a category of snippets. In this figure, for example, the My Snippets category is displayed. From here, you can select a snippet and then click the Insert Snippet button to enter the snippet into the SQL Editor tab. Finally, you can edit the snippet code so it's appropriate for your SQL statement.

In this figure, the snippet contains code that I wrote for joining the vendors, invoices, and invoice_line_items tables. To create this snippet, I entered it into a SQL Editor tab and then clicked the Add New Snippet button. By saving this statement as a snippet, I can now use it anytime I want to join these three tables instead of having to type it each time.

For now, don't worry if you don't understand the SQL statement presented in this figure. The main point is that you can use the Snippets tab to save and retrieve a variety of SQL code. As you learn more about SQL statements, you'll see how useful this can be.

The SQL Additions tab with a snippet created by a user



Description





- The SQL Additions window contains context help and *snippets*. Snippets contain the syntax for many common SQL statements. You can use the snippets to guide you as you create a SQL statement. You can also create your own snippets and save them for later use.
- The SQL Additions window is displayed to the right of the SQL Editor tab by default. If this window isn't displayed, you can display it by clicking the rightmost button () at the right side of the SQL Editor toolbar. Then, you can click the Snippets tab to display the available snippets.
- The snippets are organized into categories. To display any category of snippets, select the category from the drop-down list at the top of the Snippets tab.
- To enter a snippet into a SQL editor, select the snippet and then click the Insert Snippet button () at the top of the Snippets tab. Then, edit the snippet code so it's appropriate for your SQL statement.
- To replace code in the SQL editor with a snippet, select the code, select the snippet you want to replace it with, and then click the Replace Current Text button ().
- To create your own snippet, enter the code for the snippet into a SQL editor tab. Then, select the category where you want to save the snippet, click the Save Snippet button () in the SQL Editor toolbar, and enter a name for the snippet.
- To delete a snippet, right-click it in the Snippets tab and select the Delete Snippet item.

Figure 2-8 How to use the Snippets tab

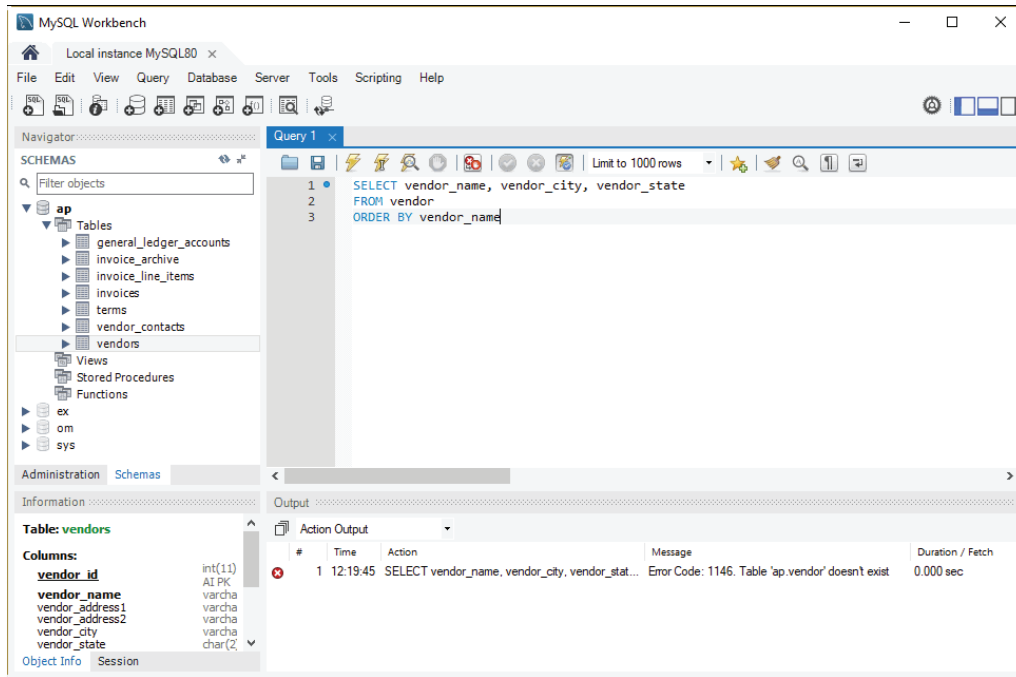
How to handle syntax errors

If an error occurs during the execution of a SQL statement, MySQL Workbench displays a message that includes the error number and a brief description of the error. In figure 2-9, for example, the message displays an error number of 1146 and a brief description that says “Table ap.vendor doesn’t exist.”

In this example, the problem is that the Vendor table doesn’t exist in the database. To fix the problem, you need to edit the SQL statement so the table is Vendors instead of Vendor. Then, you should be able to successfully run the SQL statement.

This figure also lists some other common causes of errors. As you can see, most errors are caused by incorrect syntax. However, it’s also common to get an error if you have selected the wrong database. If, for example, you have selected the EX database and you try to run a statement that refers to tables in the AP database, you will get an error. Regardless of what’s causing the problem, you can usually identify and correct the problem without much trouble. In some cases, though, it may be difficult to figure out the cause of an error. Then, you can usually get more information about the error by searching the Internet or by searching the MySQL Reference Manual as described later in this chapter.

How to handle syntax errors



Common causes of errors

- Having the wrong database selected
- Misspelling the name of a table or column
- Misspelling a keyword
- Omitting the closing quotation mark for a character string

Description

- If an error occurs during the execution of a SQL statement, MySQL Workbench displays a message in the Output tab that includes an error code and a brief description of the error.
- Most errors are caused by incorrect syntax and can be corrected without any additional assistance. Otherwise, you can usually get more information about an error by searching for the error code or description in the MySQL Reference Manual or on the Internet.

Figure 2-9 How to handle syntax errors

How to open and save SQL scripts

In MySQL, a *script* is a file that contains one or more SQL statements. To create a script, you enter the statements you want it to include into a SQL Editor tab. You'll learn more about that in the next figure. Then, you can click the Save button or press Ctrl+S to save the script as described in figure 2-10.

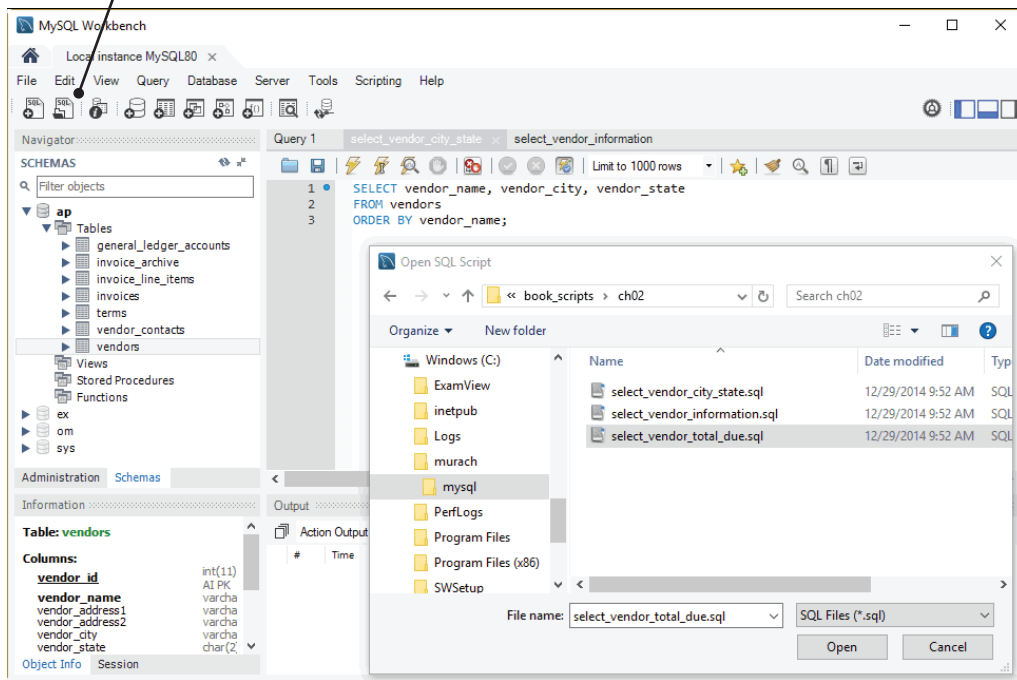
Once you've saved a script, you can open it later. To do that, you can click the Open SQL Script File button in the SQL Editor toolbar, or you can press Ctrl+Shift+O. In this figure, the dialog box that's displayed shows the script files that have been saved for chapter 2. These files are created when you download and install the source code for this book. Note that the names of these files have the .sql extension. (If you're using Windows 10 and the file extensions aren't displayed, you can display them by opening the File Explorer, displaying the View tab, and selecting the "File name extensions" option in the Show/hide group.)

Once you open a script, you can run it as shown in the next figure. You can also use it as the basis for a new SQL script. To do that, just modify it any way you want. Then, you can save it as a new script by pressing the Ctrl+Shift+S keys or selecting the File→Save Script As command.

The screen in this figure shows the tabs for two script files that have been opened. After you open two or more scripts, you can switch between them by clicking on the appropriate tab. Then, you can cut, copy, and paste code from one script to another.

The Open SQL Script dialog box

Open SQL Script
File button



Description

- A *SQL script* is a file that contains one or more SQL statements.
- To open a file that contains a SQL script, click the Open SQL Script File button in the SQL Editor toolbar or press the Ctrl+Shift+O keys. Then, use the Open SQL Script dialog box to locate and open the SQL script.
- When you open a SQL script, MySQL Workbench displays it in its own SQL Editor tab. To switch between open scripts, select the appropriate tab.
- To cut, copy, and paste code from one SQL script to another, use the standard techniques.
- To save a SQL statement to a script file, click the Save button in the SQL Editor toolbar or press Ctrl+S. Then, use the Save SQL Script dialog box to specify a location and name for the file.
- To save a script you've modified to a new file, press the Ctrl+Shift+S keys or select the File→Save Script As command.

Figure 2-10 How to open and save SQL scripts

How to enter and execute SQL scripts

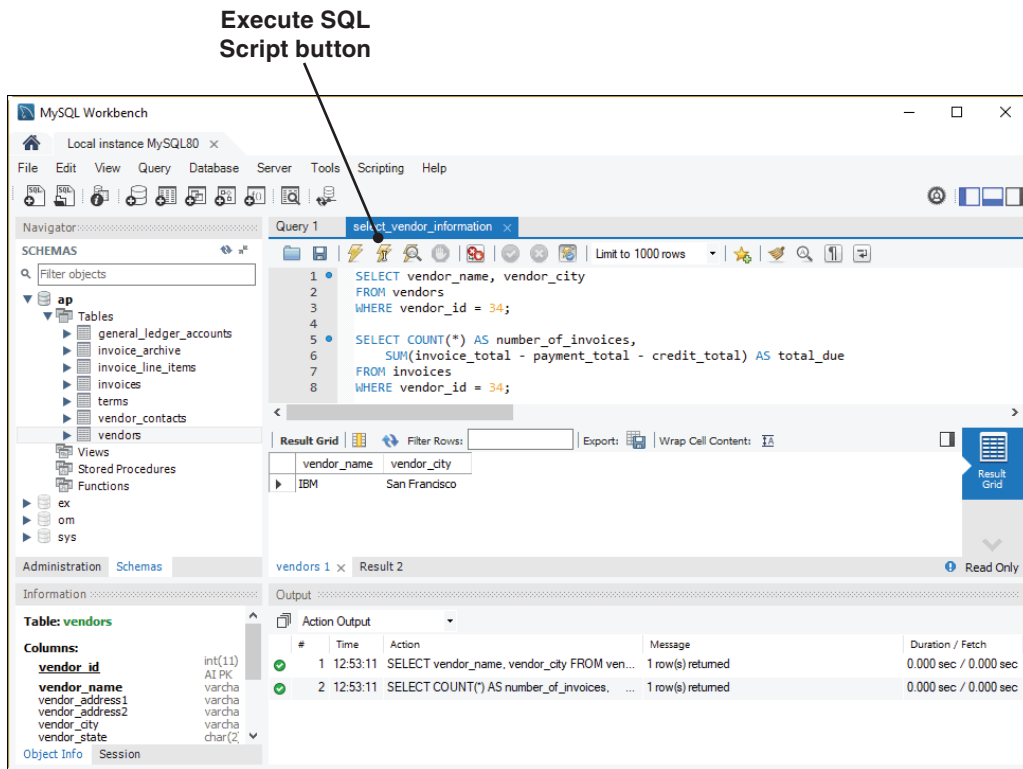
In the last topic, you saw a SQL script that contained a single SQL statement. However, a SQL script typically contains multiple statements. Figure 2-11 shows how to enter and execute scripts like that.

When you code multiple SQL statements within a script, you must code a semicolon at the end of each statement. For example, this figure shows a script that contains two `SELECT` statements. To execute both of these statements, you can press the `Ctrl+Shift+Enter` keys, or you can click the **Execute SQL Script** button in the SQL Editor toolbar. When you do, the results of each script are displayed in a separate Result grid. To switch between Result grids, you can click on the tabs that are displayed below the current Result grid.

If you want to execute a single SQL statement that's stored within a script, you can do that by moving the insertion point into the statement and pressing the `Ctrl+Enter` keys or clicking the **Execute Current Statement** button. Then, if the statement retrieves data, the data is displayed in a single Result grid.

If you need to, you can also execute two or more statements in a script. To do that, you select the statements and then press the `Ctrl+Shift+Enter` keys or click the **Execute SQL Script** button. This is useful if a script contains many statements and you just want to execute some of them.

A SQL script and its results



Description

- When you code a script that contains more than one statement, you must code a semicolon at the end of each statement.
- To run an entire SQL script, press the Ctrl+Shift+Enter keys or click the Execute SQL Script button (⚡) that's located just to the left of the Execute Current Statement button in the SQL Editor toolbar.
- When you run a SQL script, the results of each statement that returns data are displayed in a separate Result grid. To switch between these Result grids, you can click on the tabs that are displayed below the current Result grid.
- To execute one SQL statement within a script, move the insertion point into that statement and press the Ctrl+Enter keys or click the Execute Current Statement button (⚡). If the statement retrieves data, the data is displayed in a Result grid.
- To execute two or more statements within a script, select them in the editor and then press the Ctrl+Shift+Enter keys or click the Execute SQL Script button.

Figure 2-11 How to enter and execute SQL scripts

How to use the MySQL Reference Manual

Figure 2-12 shows how to use another useful tool for working with the MySQL database: the *MySQL Reference Manual*. In most cases, you'll use a web browser to view this manual directly from the Internet. That way, you can be sure that the information is always up-to-date. However, you can also download this manual and save it on your hard drive. Either way, you can use the MySQL Reference Manual to quickly look up detailed technical information about the MySQL database, including information about SQL statements and functions.

How to view the manual

You can view the MySQL Reference Manual by using a web browser to go to the web address shown at the top of this figure. Here, the Reference Manual for version 8.0 of MySQL is displayed. However, you can easily select another version by selecting it from the drop-down list at the right side of the page.

How to look up information

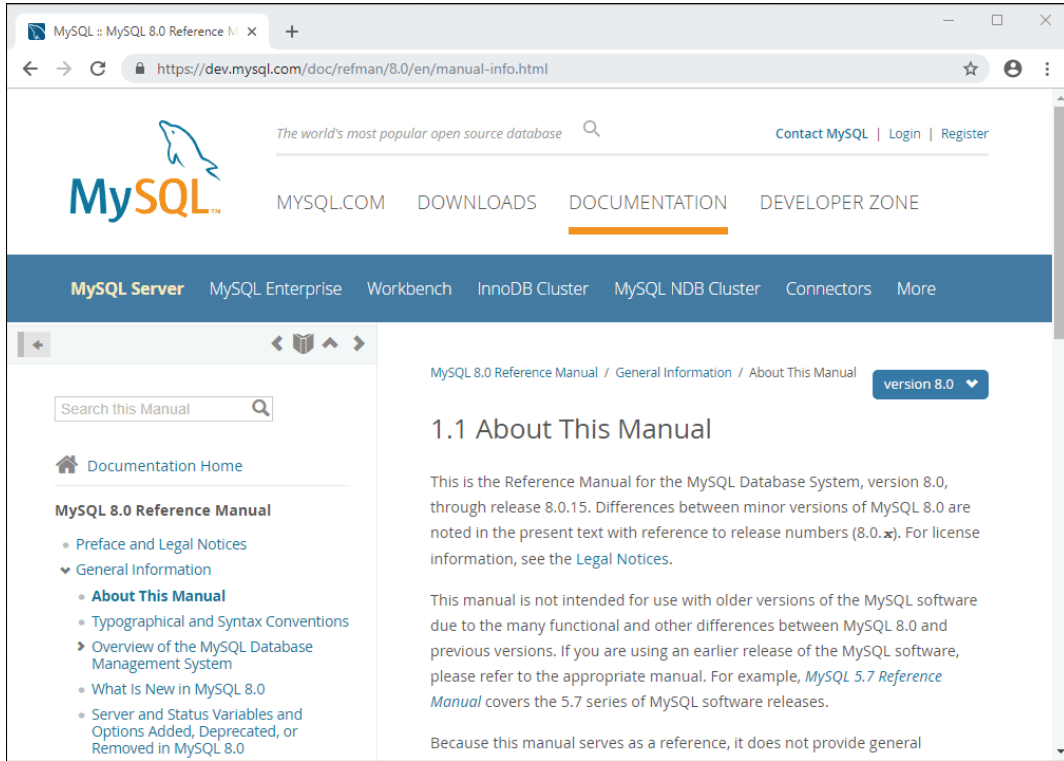
Once you've navigated to the correct version of the MySQL Reference Manual, it's easy to look up information. To do that, you can use the links in the left sidebar to drill down to the information that you're looking for. When you find the topic you want, you can click it to display it in the main part of the window. Then, if you want to navigate back up the hierarchy of information, you can use the breadcrumb links across the top of the page. In this figure, for example, you can click the "MySQL 8.0 Reference Manual" link to return to the Home page for the manual. Or, you can click the "General Information" link to navigate to that page.

Another easy way to look up information is to search for a specific word or phrase. To do that, type the word or phrase in the "Search this Manual" text box located at the top of the sidebar and click the Search icon or press the Enter key. Then, you can click the links in the search results to view information about the search terms.

The web address for the MySQL 8.0 Reference Manual

<https://dev.mysql.com/doc/refman/8.0/en/>

A web page from the MySQL Reference Manual



Description

- To view the *MySQL Reference Manual*, go to the MySQL website and select the correct version of the manual. The web address for MySQL 8.0 is shown above.
- To view a chapter, click the link for the chapter in the table of contents on the right side of the page.
- To return to the Home page for the manual, click the Start icon (📖) for the manual that's displayed at the top of the left sidebar.
- To search for a particular word or phrase, type the word or phrase in the “Search this Manual” text box in the left sidebar and click the Search icon or press the Enter key. Then, you can scroll through the results and click links to get more information.
- You can also download the MySQL Reference Manual. However, it typically makes sense to use it online.

Figure 2-12 How to use the MySQL Reference Manual

How to use the MySQL Command Line Client

Before MySQL Workbench was available, programmers used a command-line tool known as the *MySQL Command Line Client* to connect to a MySQL server and work with it. This tool is also known as the *MySQL command line*. Although you may never need this tool, you should at least be aware that it exists. This tool comes with MySQL, and it can be useful if MySQL Workbench isn't installed on the system that you're using.

How to start and stop the MySQL Command Line Client

Figure 2-13 shows how to start and stop the MySQL Command Line Client in Windows. Although this figure shows the Command Prompt window that's available from Windows, you can use the MySQL Command Line Client on other operating systems too. In particular, on macOS, you can use the Terminal window to start the MySQL Command Line Client.

When you use Windows, there's an easy way to start the MySQL Command Line Client if you want to log in as the root user for the database server that's running on the local computer. To do that, you just select the MySQL Command Line Client command from the Start menu. Then, MySQL will prompt you for a password. If you enter the password correctly, you will be logged on to the database server as the root user.

In some cases, you'll need to use a command line to start the MySQL Command Line Client instead of using the Start menu. For example, you may need to do that if you want to log into a database that's running on a different computer, if you want to log in as a user other than the root user, or if you're using another operating system such as macOS. In those cases, you can open a command line and change the directory to the bin directory for the MySQL installation. Then, you can execute the `mysql` command and supply the parameters that are needed to connect to the database server.

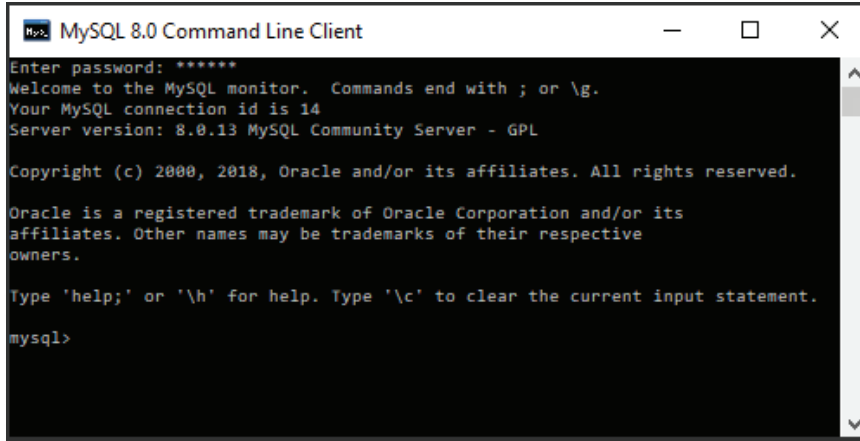
If the MySQL server is located on a remote computer, you can specify `-h`, followed by the host name of the computer, and `-u`, followed by a valid username. In addition, you specify `-p` so MySQL prompts you for a valid password. Although it can take some experimentation to get these connection parameters right, you only need to figure this out once.

Once you enter a valid password for the specified username, the MySQL Command Line Client displays a welcome message and a command line that looks like this:

```
mysql>
```

From this prompt, you can enter any statement that works with MySQL. When you're done, you can exit the MySQL Command Line Client by entering "exit" or "quit" followed by a semicolon.

The MySQL Command Line Client displayed by Windows



How to start the MySQL Command Line Client (Windows only)

Start→All Programs→MySQL→MySQL Server 8.0→MySQL 8.0 Command Line Client

How to start the MySQL Command Line Client from the command line

For Windows

```
cd \Program Files\MySQL\MySQL Server 8.0\bin
mysql -u root -p
```

For macOS

```
cd /usr/local/mysql/bin
./mysql -u root -p
```

How the mysql command works

The syntax

```
mysql -h hostname -u username -p
```

Examples

```
mysql -u ap_tester -p
mysql -h localhost -u root -p
mysql -h murach.com -u ap_tester -p
```

How to exit from the MySQL Command Line Client

```
mysql>exit;
```

Description

- MySQL provides a command-line client program called the *MySQL Command Line Client* that lets you enter SQL statements that work with MySQL databases. This program is also known as the *MySQL command line*.
- For Windows, use a Command Prompt window to start the MySQL Command Line Client.
- For macOS, use a Terminal window to start the MySQL Command Line Client.
- To stop the MySQL Command Line Client, enter “exit” or “quit” at the command line, followed by a semicolon.
- MySQL 8.0 also includes a Unicode version of the command-line client program. For more information on this program, you can refer to section 4.5.1.6.2 of the reference manual.

Figure 2-13 How to start and stop the MySQL Command Line Client

How to use the MySQL Command Line Client to work with a database

Once the MySQL Command Line Client is connected to a database server, you can use it to run SQL statements that work with the databases that are available from that server. When you enter a statement, you must end it with a semicolon. Otherwise, the `mysql` command line displays a second line when you press the Enter key like this:

```
mysql> show databases
->
```

This shows that the MySQL Command Line Client is waiting for you to finish your statement. To finish a statement and execute it, you just type a semicolon and press the Enter key.

Figure 2-14 shows how to execute three SQL statements. Here, I entered all three of these statements in lowercase letters. That's because SQL isn't case-sensitive, and lowercase letters are easier to type.

To list the names of the databases stored on a server, you use the `SHOW DATABASES` statement as illustrated by the first example. Here, the “ap”, “ex”, and “om” databases are the databases that are created when you install our downloadable databases as described in appendixes A and B. The “information_schema”, “performance_schema”, and “mysql” databases are internal databases that are used by the MySQL server. And the “sys” database is a database that comes with MySQL and can be used to interpret data in the “performance_schema” database.

To select the database that you want to work with, you can enter a `USE` statement as illustrated by the second example. Here, the `AP` database is selected, and the message after this statement says “Database changed” to indicate that the statement was successful. After you select a database, the commands and statements that you enter will work with that database.

To retrieve data from the database, you use a `SELECT` statement as illustrated by the third example. Here, the `vendor_name` column from the `Vendors` table is displayed. Note, however, that the result set is limited to only the first five rows. When you successfully execute a `SELECT` statement, the MySQL Command Line Client displays a message giving the number of rows that are included in the result set and the amount of time it took to run the query. In this case, it took less than 1/100 of a second to run the query.

How to list the names of all databases managed by the server

```
mysql> show databases;
+-----+
| Database |
+-----+
| ap       |
| ex       |
| information_schema |
| mysql    |
| om       |
| performance_schema |
| sys      |
+-----+
7 rows in set (0.00 sec)
```

How to select a database for use

```
mysql> use ap;
Database changed
```

How to select data from a database

```
mysql> select vendor_name from vendors limit 5;
+-----+
| vendor_name |
+-----+
| Abbey Office Furnishings |
| American Booksellers Assoc |
| American Express |
| ASC Signs |
| Ascom Hasler Mailing Systems |
+-----+
5 rows in set (0.00 sec)
```

Description

- You can use the MySQL Command Line Client to work with any of the databases running on the database server. To do that, you can use any SQL statement that works with a MySQL database.
- To execute a SQL statement, type the statement on the command line, followed by a semicolon. Then, press the Enter key.
- To show a list of all available databases, you can use the `SHOW DATABASES` statement.
- To select the database that you want to work with, you can use the `USE` statement.
- SQL statements aren't case-sensitive. As a result, when using the MySQL Command Line Client, most programmers enter their statements in lowercase letters because they're easier to type.

Figure 2-14 How to use the MySQL Command Line Client to work with a database

Perspective

In this chapter, you learned how to use MySQL Workbench to start and stop a MySQL server and to enter and execute SQL statements. With that as background, you're ready to go on to the next chapter, where you'll start learning the details of coding your own SQL statements.

Terms

MySQL Workbench
database server
database service
database engine
database object
schema
snippet
SQL script
MySQL Reference manual
MySQL Command Line Client

Before you start the exercises...

Before you start the exercises for this chapter, you need to install the MySQL Server and MySQL Workbench. In addition, you need to download and install the source files for this book, and you need to create the databases and tables for this book. The procedures for doing all of these tasks are provided in appendix A (Windows) and B (macOS).

Exercises

In these exercises, you'll use MySQL Workbench to review the tables in the AP database. In addition, you'll use MySQL Workbench to enter SQL statements and run them against these tables.

Make sure the MySQL server is running

1. Start MySQL Workbench and open a connection for the root user.
2. Check whether the MySQL server is running. If it isn't, start it. When you're done, close the Startup/Shutdown tab.

Use MySQL Workbench to review the Accounts Payable (AP) database

3. In the Navigator window, expand the node for the AP database so you can see all of the database objects it contains.
4. View the data for the Vendors and Invoices tables.

5. Navigate through the database objects and view the column definitions for at least the Vendors and Invoices tables.

Use MySQL Workbench to enter and run SQL statements

6. Double-click the AP database to select it. When you do that, MySQL Workbench should display the database in bold.
7. Open a SQL Editor tab. Then, enter and run this SQL statement:

```
SELECT vendor_name FROM vendors
```
8. Delete the *e* at the end of vendor_name and run the statement again. Note the error number and the description of the error.
9. Open another SQL Editor tab. Then, enter and run this statement:

```
SELECT COUNT(*) AS number_of_invoices,  
       SUM(invoice_total) AS grand_invoice_total  
FROM invoices
```

Use MySQL Workbench to open and run scripts

10. Open the select_vendor_city_state script that's in the c:\murach\mysql\scripts\ch02 directory. Note that this script contains just one SQL statement. Then, run the statement.
11. Open the select_vendor_total_due script that's in the ch02 directory. Note that this opens another SQL Editor tab.
12. Open the select_vendor_information script that's in the ch02 directory. Notice that this script contains two SQL statements that end with semicolons (scroll down if you need to).
13. Press the Ctrl+Shift+Enter keys or click the Execute SQL Script button to run both of the statements in this script. Note that this displays the results in two Result grids. Make sure to view the results of both SELECT statements.
14. Move the insertion point into the first statement and press Ctrl+Enter to run just that statement.
15. Move the insertion point into the second statement and press Ctrl+Enter to run just that statement.
16. Exit from MySQL Workbench.

How to build your MySQL skills

The easiest way is to let [Murach's MySQL \(3rd Edition\)](#) be your guide! So if you've enjoyed this chapter, I hope you'll get your own copy of the book today. You can use it to:

- Teach yourself how to code SQL statements to retrieve and maintain the data in a MySQL database
- Design and create your own MySQL databases using EER diagrams and SQL statements
- Take advantage of procedural coding routines that are stored with a MySQL database to maintain data integrity and increase your own productivity
- Handle basic DBA tasks like monitoring, configuring, securing, and backing up a database
- Pick up new skills whenever you want to or need to by focusing on material that's new to you
- Look up coding details or refresh your memory on forgotten details when you're in the middle of developing a MySQL application
- Loan to your colleagues who are always asking you questions about MySQL programming



Mike Murach, Publisher

To get your copy, you can order online at www.murach.com or call us at 1-800-221-5528 (toll-free in the U.S. and Canada). And remember, when you order directly from us, this book comes with my personal guarantee:

100% Guarantee

You must be satisfied. Each book you buy directly from us must outperform any competing book or course you've ever tried, or send it back within 30 days for a full refund...no questions asked.

Thanks for your interest in Murach books!

A handwritten signature in black ink that reads "Mike".