

## 1

### Introducing SQL

7

What is SQL?	8
Making SQL queries	10
Installing MySQL on Windows	12
Installing MySQL on Linux	14
Adding an ODBC connector	16
Using Microsoft Query tool	18
Summary	20

## 2

### Getting started

21

Introducing databases	22
Exploring databases	23
Creating a database	24
Deleting a database	26
Running SQL scripts	28
Summary	30

## 3

### Creating database tables

31

Exploring database tables	32
Creating a table	34
Deleting a table	36
Table data types	38
Table field modifiers	40
Setting the primary key	42
Altering a table	44
Summary	46

## 4

### Inserting data into tables

47

Inserting complete rows	48
Including a columns list	50
Inserting selected data	52
Updating data	54
Changing specific data	56
Deleting data	58
Summary	60

# 5

## Retrieving data from tables

61

Retrieving a column	62
Retrieving multiple columns	64
Retrieving a row	66
Copying retrieved data	68
Inserting selected fields	70
Summary	72

# 6

## Sorting retrieved data

73

Sorting a column	74
Sorting multiple columns	76
Sorting by column position	78
Setting the sort direction	80
Summary	82

# 7

## Simple data filtering

83

Making comparisons	84
Comparing a single value	86
Comparing a range of values	88
Seeking non-matches	90
Finding null values	92
Summary	94

# 8

## Complex data filtering

95

Comparing multiple values	96
Comparing alternative values	98
Comparing alternative lists	100
Specifying evaluation order	102
Matching strings	104
Matching characters	106
Matching regular expressions	108
Summary	110

# 9

## Generating calculated fields

111

Concatenating fields	112
Trimming padded spaces	114
Adopting aliases	116
Doing arithmetic	118
Summary	120

**10****Manipulating data****121**

Introducing functions	122
Text functions	124
Numeric functions	126
Date and time functions	128
System functions	130
Summary	132

**11****Grouping table data****133**

Finding summary values	134
Counting rows	136
Discovering total values	138
Working with distinct values	140
Creating data groups	142
Filtering grouped data	144
Sorting filtered group data	146
Summary	148

**12****Making complex queries****149**

Using sub-queries	150
Sub-query calculated fields	152
Combining queries	154
Handling duplicate rows	156
Sorting combined results	158
Summary	160

**13****Joining database tables****161**

What are joins?	162
Creating a join	164
Joining multiple tables	166
Creating self joins	168
Creating natural joins	170
Specifying join direction	172
Summary	174

**+****Handy reference****175****Index****187**

# How To Use This Book

The creation of this book has provided me, Mike McGrath, a welcome opportunity to update my previous books on SQL programming with the latest techniques. All the examples I have given in this book demonstrate SQL features using the current MySQL Relational Database Management System that is supported on both Windows and Linux operating systems.

## Conventions in this book

In order to clarify the code listed in the steps given in each example, I have adopted certain colorization conventions. Components of the SQL language itself are colored blue, programmer-specified names are red, numeric and string data values are black, and comments are green, like this:

```
# Insert 5 records into the "top_5_films" table.  
INSERT INTO top_5_films ( position , title , year )  
VALUES ( 1 , "Citizen Kane" , 1941 ) ;
```

Additionally, in order to identify each source code file described in the steps, a colored icon and file name appears in the margin alongside the steps:



query.sql

## Grabbing the source code

You can download a single ZIP archive file containing all the complete example files that appear in this book by following these easy steps:

- 1 Browse to [www.ineasysteps.com](http://www.ineasysteps.com) then navigate to [Free Resources](#) and choose the [Downloads](#) section
- 2 Next, find [SQL in easy steps, 4th edition](#) in the list, then click on the hyperlink entitled [All Code Examples](#) to download the ZIP archive file
- 3 Now, extract the archive contents to any convenient location on your computer

I sincerely hope you enjoy discovering the powerful exciting possibilities of SQL and have as much fun with it as I did in writing this book. *Mike McGrath*

If you don't achieve the result illustrated in any example, simply compare your code to that in the original example files you have downloaded to discover where you went wrong.

# 1

# Introducing SQL

*Welcome to the exciting world of the Structured Query Language (SQL). This chapter describes how to install a free SQL database server on your system.*

- 8** What is SQL?
- 10** Making MySQL queries
- 12** Installing MySQL on Windows
- 14** Installing MySQL on Linux
- 16** Adding an ODBC connector
- 18** Using Microsoft Query tool
- 20** Summary



A programming language (pronounced either “S-Q-L” or “sequel”) designed to manage database data.

# What is SQL?

The Structured Query Language (SQL) is a language designed specifically for communicating with databases. Today SQL is the industry-wide standard language used by most database systems.

Databases allow collections of data to be stored in an organized manner – in the same way that data can be stored in an organized way inside files within a filing cabinet. Most modern DataBase Management Systems (DBMSs) store data in related tables, so are called Relational DBMS (RDBMS). The data stored inside databases can be examined and manipulated by SQL commands.

SQL commands are known as “queries” and utilize special keywords that can be used both to add data to a database, or extract details of data contained within a database. There are not many keywords so SQL is simple to understand but, despite its apparent simplicity, is a powerful language. Clever use of its language components enable complex sophisticated database operations to be performed.

## The evolution of SQL

The model for the basis of SQL was defined back in 1970 by Dr. E. F. Todd, a researcher for IBM, in a paper entitled “A Relational Model of Data for Large Shared Data Banks”. This article generated a great deal of interest in the feasibility of producing a practical commercial application of such a system.

IBM really began to develop these ideas in 1974 when they started the System/R project which eventually created the Structured English Query Language (SEQUEL). This was rewritten in 1976 to include multi-table and multi-user features and was renamed SQL in 1978. During this time other software companies had begun working on database systems based upon the SQL model. Most notable of these were Oracle, Sybase and Ingres (from the University of California’s Berkeley Ingres project). The first to be released commercially was Oracle in 1979. IBM released improved database products named SQL/DS in 1982 and DB2 in 1983.

Modern versions of Oracle, Sybase, Ingres and DB2 DBMS are available today and are in widespread use around the world.

...cont'd

## Standardization of SQL

In order to clarify the precise nature of SQL, so it could be implemented universally, each aspect of the language was defined in 1989 by the American National Standards Institute (ANSI) in a standard specification known as SQL-89. This was expanded three years later with publication of the SQL-92 specification by a joint committee of ANSI and the International Standards Organization (ISO). A third standard specification, SQL-99, was introduced in 1999 to address issues of advanced SQL syntax and has been subsequently updated with the SQL:2008 standard. Some DBMS vendors have added proprietary features to the ANSI-defined SQL standard. These extended versions even have their own names, such as PL-SQL and Transact-SQL. The examples given in this book mostly use standard ANSI-SQL so they can be applied to any DBMS.

## Forms of SQL query

There are a number of ways that SQL queries may be sent to a database to deposit or extract data:

- Directly input through an integral SQL-client application that is part of the DBMS package – this is the most straightforward method and is used in this book to demonstrate SQL features.
- Input through a third-party SQL-client application – this method communicates with the database via an intermediate software “driver”. On Windows systems these are typically Open DataBase Connectivity (ODBC) data source drivers.
- From a script – often found on web servers to dynamically communicate with a database using a scripting language such as PERL, PHP or Python.
- From an Integrated Development Environment (IDE) – programmers using IDEs, such as Microsoft Visual Basic, can build programs that incorporate SQL queries to a database.

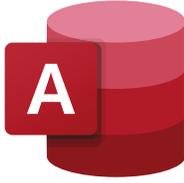
The next two pages outline how to execute SQL queries using a variety of popular software.



“ISO” is not an acronym but is derived from the Greek word “isos” meaning equal – as in “isometric”.



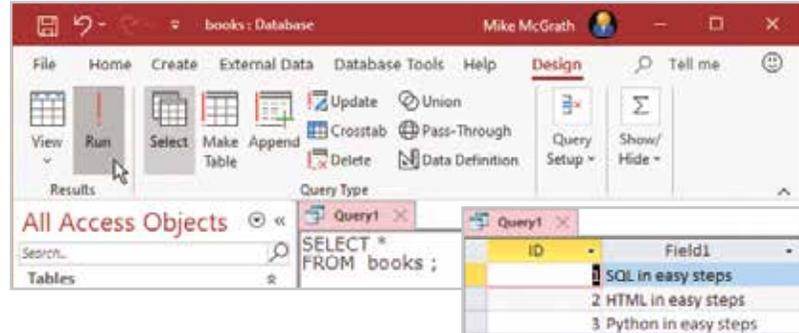
Learning standard ANSI-SQL enables you to interact with every major database that exists.



# Making SQL queries

## Microsoft® Access

Access is the popular database program supplied as part of the Microsoft Office suite. It is popular with Office users on stand-alone PCs and small networks. The SQL View allows you to enter SQL queries to be executed when you click the !Run button.



## Microsoft® SQL Server

The SQL Server DBMS products from Microsoft are popular on computers running the Windows operating system. Microsoft SQL Server Express is a free, lightweight and feature-rich database for data-enabled web and Windows applications. SQL queries can be executed from the SQL Server Management Studio.



## Microsoft® Visual Studio

Visual Studio can be used to create computer programs that make queries against a database via an ODBC Data Source. Simply select the Connect to Database item on the Tools menu to launch the Add Connections dialog, then choose a data source from the “Use user or system data source name” dropdown list. SQL queries can then be made from the program code.

...cont'd

## Oracle®

The Oracle DBMS is popular and widely used in commerce. Oracle Database Express Edition (Oracle Database XE) is a free entry-level, small-footprint database that is simple to administer. SQL queries can be executed from its **SQL>** command prompt.



```
SQL Plus
Connected to:
Oracle Database 18c Express Edition
SQL> SELECT * FROM books;
   ID TITLE
-----
    1 SQL in easy steps
    2 HTML in easy steps
    3 Python in easy steps
SQL> _
```

## IBM® DB2®

The DB2 DBMS is a powerful multi-platform database system. DB2-Express-C is another free full function DB2 data server. SQL queries can be executed from its **db2 =>** command prompt.



```
DB2 CLP
Command Line Processor for DB2 Client 11.5.0.0
db2 => SELECT * FROM books;
ID      TITLE
-----
    1    SQL in easy steps
    2    HTML in easy steps
    3    Python in easy steps

  3 record(s) selected.
db2 => _
```

## MySQL®

The world's most popular open-source database server is the freely available MySQL DBMS product that is supplied with an integral SQL-client from which to execute SQL queries from its **mysql>** command prompt. MySQL is used throughout this book to demonstrate the SQL language. The following pages describe how to install MySQL on both Windows and Linux platforms.





Further guidance on installation of the MySQL Server is available at <http://dev.mysql.com/doc/refman/8.0/en/installing.html>



The MySQL Installer can be launched at any time from the Windows' Start Menu – to change the configuration or to install updates.

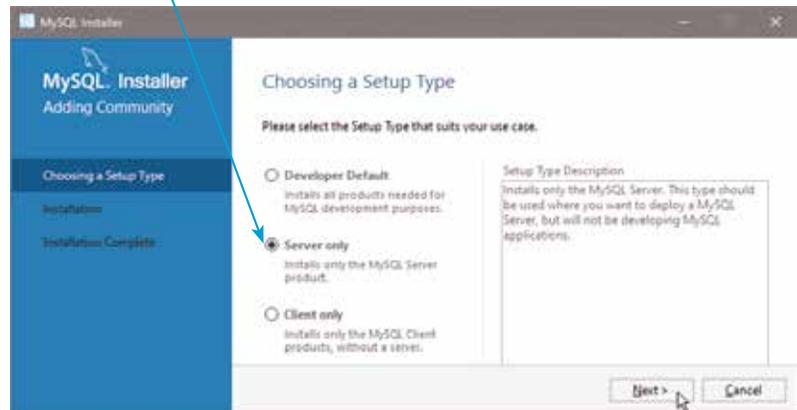
# Installing MySQL on Windows

The MySQL database server, which provides “back-end” storage for data-driven websites, is available for Windows, Mac OS X, and Linux operating systems as a free download.

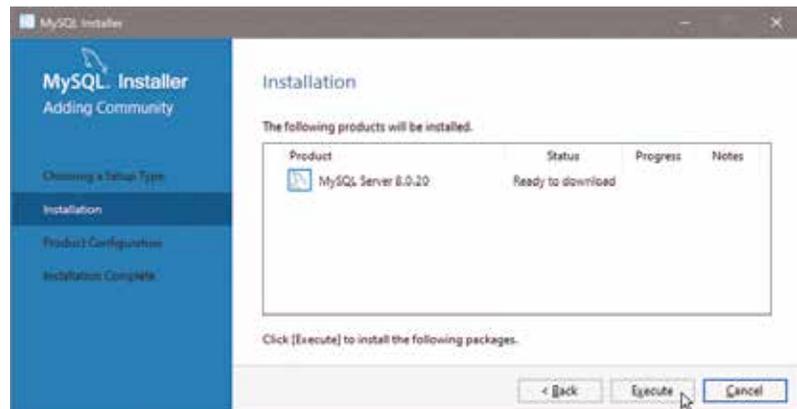
- 1 Download the MySQL Community Server installer for your system from [dev.mysql.com/downloads](http://dev.mysql.com/downloads)



- 2 Run the installer and accept the License terms, then select “Server only” and click **Next** to continue



- 3 Click the **Execute** button to install the MySQL server

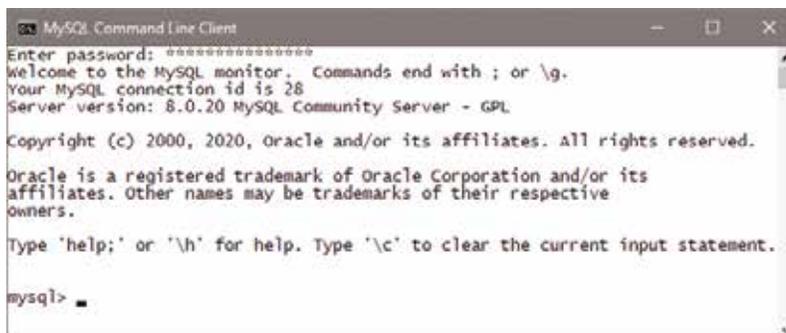


...cont'd

- 4 Click **Next**, then choose the “Standalone MySQL Server” option and click **Next** to continue
- 5 Select “Development Computer” as the server configuration type, then click **Next** to continue
- 6 Select “Use Strong Password Encryption for Authentication”, then click **Next** to continue
- 7 Enter a root user password of your choice twice into the **Password** fields, then click **Next** to continue



- 8 Click **Next** to run as a Windows Service, and **Next** to ignore Plugins, then click the **Execute** button to install the MySQL Server with your selected configuration
- 9 When installation completes launch the **MySQL Command Line Client** from the MySQL group that has been added to the Windows Start menu
- 10 Enter your password to open the MySQL monitor



By default, the MySQL Server uses port 3306. If you are running a firewall you may need to specifically allow the MySQL Server connections via this port. Refer to your firewall documentation for further guidance.



Write down your chosen root user password – you will need it often.



At the `mysql>` prompt type `exit`, `quit`, or `\q` then hit Return to close the MySQL monitor.



# Installing MySQL on Linux

MySQL for Linux platforms can be freely downloaded from [www.mysql.com](http://www.mysql.com). The recommended method of installation is to use the Redhat Package Manager (RPM), which is included with virtually all versions of Linux, or use the Package Installer that is included with the Linux Mint distro.

- 1 Open a web browser and visit the MySQL APT Repository at [dev.mysql.com/downloads/repo/apt](http://dev.mysql.com/downloads/repo/apt)
- 2 Log in with an Oracle account, then download the latest MySQL configuration file



- 3 Next, install the file on your system



You can download the `mysql-apt-config` file to your Downloads folder, then double-click its file icon to open the Package Installer app.



...cont'd

- 4 Open a Terminal window and update the repository package lists by entering this command  
**sudo apt update**
- 5 Install the MySQL server package with this command  
**sudo apt install mysql-server**
- 6 When asked to confirm the installation, press the **Y** key, then hit **Enter** to continue

When the installation has completed, you should now be able to connect to the MySQL Server in a Terminal by stating the user (root), and by creating a root user password.

- 7 Enter this command in a Terminal window  
**mysql -u root -p**
- 8 When asked, enter a root user password of your choice – you should then be welcomed to the MySQL monitor

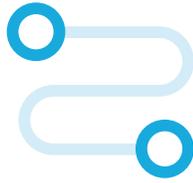
```
mike@linux-pc: ~  
File Edit View Search Terminal Help  
mike@linux-pc:~$ mysql -u root -p  
Enter password:  
Welcome to the MySQL monitor.  Commands end with ; or \g.  
Your MySQL connection id is 8  
Server version: 8.0.20 MySQL Community Server - GPL  
  
Copyright (c) 2000, 2020, Oracle and/or its affiliates. All rights reserved.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement  
.  
mysql> █
```



The MySQL root user password is case-sensitive – be sure to use the correct case to connect to the MySQL server.



At the **mysql>** prompt, type **exit**, **quit**, or **\q** then hit Return to close the MySQL monitor.



# Adding an ODBC connector

Usually, third-party client applications can only connect to the MySQL server if an appropriate Open DataBase Connectivity (ODBC) connector is installed on the system.

There are a whole range of ODBC connectors freely available for the MySQL server. These and other features can be added to your MySQL installation using the MySQL Installer app.

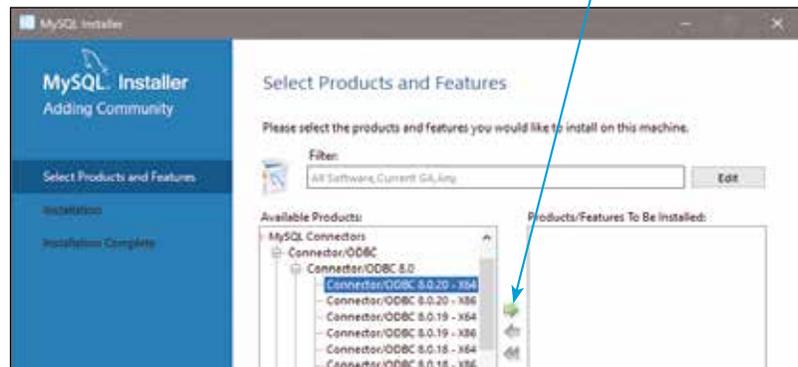
- 1 Launch the **MySQL Installer** app from the MySQL group on the Windows Start menu



- 2 Next, click the **Add** button in the Installer dialog to open a “Select Products and Features” window

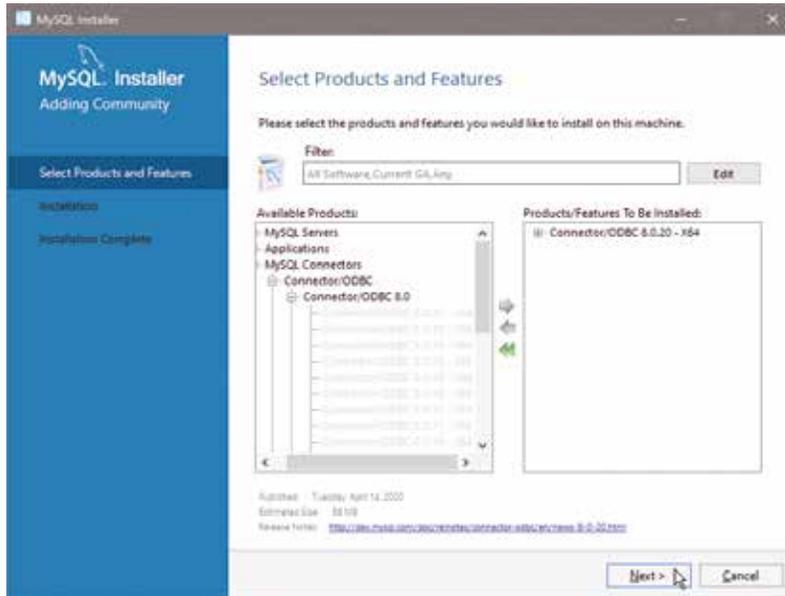


- 3 Expand the **MySQL Connectors** categories, then choose an ODBC connector and click the arrow button to select that connector for installation



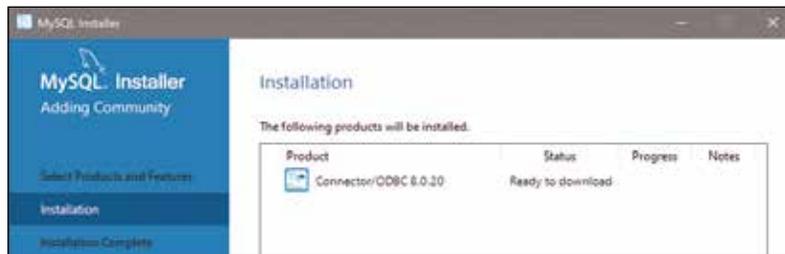
Choose the connector that matches your MySQL server version (here, it's 8.0.19) and your system architecture (e.g. X64).

...cont'd



Your system needs to be connected to the internet to add features to your MySQL installation.

- 4 Click **Next** to continue, then click the **Execute** button on the next screen to install the ODBC connector



- 5 When installation completes, click the **Finish** button to see the connector is added to your MySQL installation



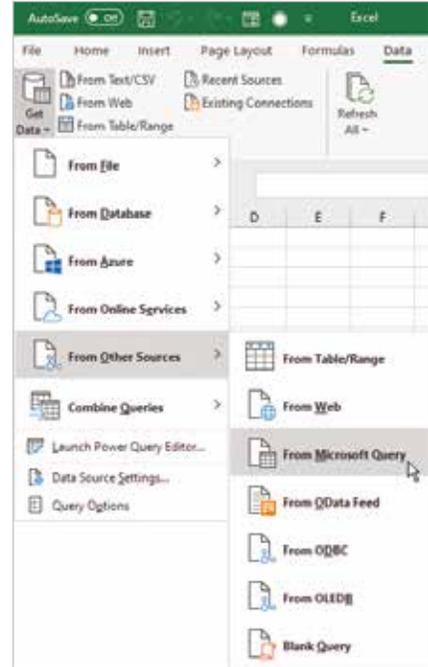


# Using Microsoft Query tool

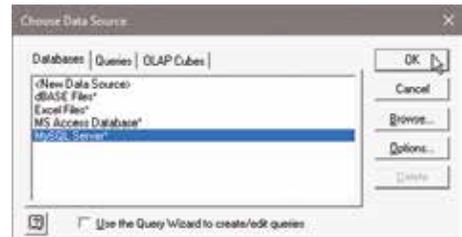
The Microsoft Query tool can be used to make SQL queries to a database. It is installed on many Windows systems without the user even knowing, because it is included with other Microsoft products such as the Excel spreadsheet program in Office.

Microsoft Query is a third-party SQL client that requires an ODBC connector to connect to the database, such as the MySQL ODBC connector installed on pages 16-17.

- 1 Launch the **Excel** app, then open a blank workbook
- 2 Select the **Data** tab on the ribbon
- 3 Now, click **Get Data** to open a dropdown menu of possible sources
- 4 Select **From Other Sources, From Microsoft Query** to open a “Choose Data Source” dialog



- 5 In the “Choose Data Source” dialog, select the **MySQL Server** item, then click **OK** to see an “Add Tables” dialog appear and a disabled “Microsoft Query” window

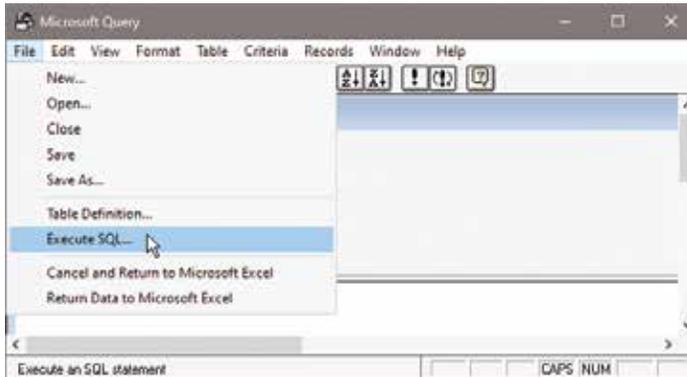


- 6 Close the “Add Tables” dialog to enable the “Microsoft Query” window



Uncheck the “Use Query Wizard to create/edit queries” option in the Choose Data Source dialog if it is checked.

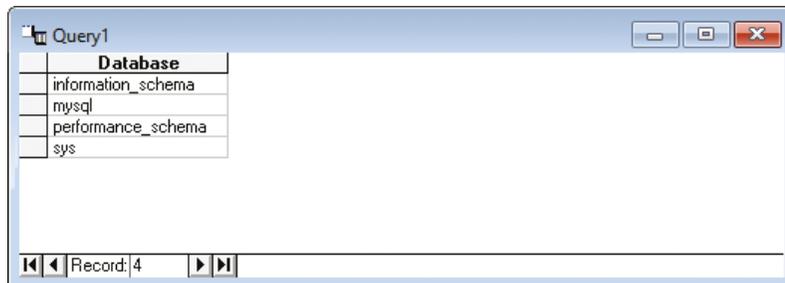
...cont'd



- 7 In the “Microsoft Query” window, click **File, Execute SQL...** to open an “Execute SQL” dialog
- 8 In the “Execute SQL” dialog type **SHOW DATABASES ;** into the “SQL statement” field



- 9 Click the **Execute** button to see all current MySQL databases appear listed in a “Query” dialog



The MySQL databases shown here are those installed by default on installation – you will learn how to add your own databases in Chapter 2.

# Summary

- The Structured Query Language (SQL) is the standard language for communicating with databases.
- SQL is standardized by the American National Standards Institute (ANSI) and the International Standards Organization (ISO).
- Data can be stored in a database or retrieved from a database by making an SQL query.
- Oracle Database XE, IBM DB2-Express-C and the MySQL Community Edition are all free database applications.
- SQL queries can be made from an integral client application, such as at the `mysql>` prompt in the MySQL monitor.
- Third-party applications can be used to make SQL queries via an intermediate connector using Open DataBase Connectivity (ODBC).
- Server-side scripts can make SQL queries to provide dynamic content to the user.
- Computer programs can be written to make SQL queries to a database in an Integrated Development Environment (IDE) – such as Microsoft’s Visual Studio IDE.
- MySQL is the world’s most popular open-source DataBase Management System (DBMS), recognized for its speed and reliability.
- There are versions of MySQL for all popular operating systems – including Windows, Linux, Solaris, and OS2.
- The Microsoft Query tool can execute SQL queries against the MySQL DBMS via an ODBC connector.