Contents at a Glance

1  Understanding SQL
2  Introducing SQL Server
3  Working with SQL Server
4  Retrieving Data
5  Sorting Retrieved Data
6  Filtering Data
7  Advanced Data Filtering
8  Using Wildcard Filtering
9  Creating Calculated Fields
10 Using Data Manipulation Functions
11 Summarizing Data
12 Grouping Data
13 Working with Subqueries
14 Joining Tables
15 Creating Advanced Joins
16 Combining Queries
17 Full Text Searching
18 Inserting Data
19 Updating and Deleting Data
20 Creating and Manipulating Tables
21 Using Views
22 Programming With T-SQL
23 Working with Stored Procedures
24 Using Cursors
25 Using Triggers
26 Managing Transaction Processing
27 Working With XML And JSON
28 Globalization and Localization
29 Managing Security
30 Improving Performance

Appendixes
A  The Example Tables
B  T-SQL Statement Syntax
Accessing the Free Web Edition

Your purchase of this book in any format includes access to the corresponding Web Edition, which provides special features to help you learn:

- The complete text of the book online
- Updates and corrections as they become available

The Web Edition can be viewed on all types of computers and mobile devices with any modern web browser that supports HTML5.

To get access to the Web Edition of *Sams Teach Yourself Microsoft SQL Server T-SQL in 10 Minutes* all you need to do is register this book:

1. Go to www.informit.com/register
2. Sign in or create a new account.
4. Answer the questions as proof of purchase.
5. The Web Edition will appear under the Digital Purchases tab on your Account page. Click the Launch link to access the product.
Contents

Introduction .................................................................................................................. 1
Who Is This Book For? ................................................................................................. 2
Companion Website ..................................................................................................... 2
Conventions Used in This Book .................................................................................... 3

1 Understanding SQL ..................................................................................................... 5
Database Basics ............................................................................................................ 5
What Is SQL? ................................................................................................................ 11
Try It Yourself ............................................................................................................. 12
Summary ...................................................................................................................... 12

2 Introducing SQL Server .............................................................................................. 13
What Is SQL Server? .................................................................................................... 13
Getting Started with SQL Server and T-SQL .............................................................. 16
Preparing for Your Lessons ......................................................................................... 18
Summary ...................................................................................................................... 18

3 Working with SQL Server ........................................................................................... 19
Making the Connection ............................................................................................... 19
Navigating SQL Server Management Studio ............................................................ 20
Creating the Example Tables ....................................................................................... 22
Selecting a Database .................................................................................................... 24
Learning About Databases and Tables ....................................................................... 25
Summary ...................................................................................................................... 29

4 Retrieving Data .......................................................................................................... 31
The SELECT Statement ................................................................................................. 31
Retrieving Individual Columns .................................................................................... 31
Retrieving Multiple Columns ....................................................................................... 33
<table>
<thead>
<tr>
<th>Contents</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Retrieving All Columns</td>
<td>35</td>
</tr>
<tr>
<td>Retrieving Distinct Rows</td>
<td>36</td>
</tr>
<tr>
<td>Limiting Results</td>
<td>37</td>
</tr>
<tr>
<td>Using Fully Qualified Table Names</td>
<td>39</td>
</tr>
<tr>
<td>Summary</td>
<td>40</td>
</tr>
</tbody>
</table>

5 **Sorting Retrieved Data**                                             | 41   |
| Sorting Data                                                            | 41   |
| Sorting by Multiple Columns                                             | 43   |
| Specifying Sort Direction                                               | 44   |
| Summary                                                                 | 47   |

6 **Filtering Data**                                                     | 49   |
| Using the WHERE Clause                                                  | 49   |
| The WHERE Clause Operators                                              | 50   |
| Summary                                                                 | 56   |

7 **Advanced Data Filtering**                                            | 57   |
| Combining WHERE Clauses                                                 | 57   |
| Using the IN Operator                                                   | 61   |
| Using the NOT Operator                                                  | 63   |
| Summary                                                                 | 64   |

8 **Using Wildcard Filtering**                                           | 65   |
| Using the LIKE Operator                                                 | 65   |
| Tips for Using Wildcards                                               | 71   |
| Summary                                                                 | 71   |

9 **Creating Calculated Fields**                                         | 73   |
<p>| Understanding Calculated Fields                                        | 73   |
| Concatenating Fields                                                   | 74   |
| Performing Mathematical Calculations                                   | 79   |
| Summary                                                                 | 81   |</p>
<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Using Data Manipulation Functions</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>Understanding Functions</td>
<td>83</td>
</tr>
<tr>
<td></td>
<td>Using Functions</td>
<td>84</td>
</tr>
<tr>
<td></td>
<td>Summary</td>
<td>95</td>
</tr>
<tr>
<td>11</td>
<td>Summarizing Data</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>Using Aggregate Functions</td>
<td>97</td>
</tr>
<tr>
<td></td>
<td>Aggregates on Distinct Values</td>
<td>105</td>
</tr>
<tr>
<td></td>
<td>Combining Aggregate Functions</td>
<td>106</td>
</tr>
<tr>
<td></td>
<td>Summary</td>
<td>107</td>
</tr>
<tr>
<td>12</td>
<td>Grouping Data</td>
<td>109</td>
</tr>
<tr>
<td></td>
<td>Understanding Data Grouping</td>
<td>109</td>
</tr>
<tr>
<td></td>
<td>Creating Groups</td>
<td>110</td>
</tr>
<tr>
<td></td>
<td>Filtering Groups</td>
<td>111</td>
</tr>
<tr>
<td></td>
<td>Grouping and Sorting</td>
<td>114</td>
</tr>
<tr>
<td></td>
<td><code>SELECT</code> Clause Ordering</td>
<td>117</td>
</tr>
<tr>
<td></td>
<td>Summary</td>
<td>117</td>
</tr>
<tr>
<td>13</td>
<td>Working with Subqueries</td>
<td>119</td>
</tr>
<tr>
<td></td>
<td>Understanding Subqueries</td>
<td>119</td>
</tr>
<tr>
<td></td>
<td>Filtering by Subquery</td>
<td>119</td>
</tr>
<tr>
<td></td>
<td>Using Subqueries as Calculated Fields</td>
<td>123</td>
</tr>
<tr>
<td></td>
<td>Checking for Existence with Subqueries</td>
<td>126</td>
</tr>
<tr>
<td></td>
<td>Summary</td>
<td>128</td>
</tr>
<tr>
<td>14</td>
<td>Joining Tables</td>
<td>129</td>
</tr>
<tr>
<td></td>
<td>Understanding Joins</td>
<td>129</td>
</tr>
<tr>
<td></td>
<td>Creating a Join</td>
<td>132</td>
</tr>
<tr>
<td></td>
<td>Summary</td>
<td>140</td>
</tr>
<tr>
<td>15</td>
<td>Creating Advanced Joins</td>
<td>141</td>
</tr>
<tr>
<td></td>
<td>Using Table Aliases</td>
<td>141</td>
</tr>
<tr>
<td></td>
<td>Using Different Join Types</td>
<td>142</td>
</tr>
</tbody>
</table>
Contents

Using Joins with Aggregate Functions ........................................... 148
Using Joins and Join Conditions .................................................. 150
Summary .................................................................................. 151

16 **Combining Queries** ......................................................... 153
Understanding Combined Queries ............................................... 153
Creating Combined Queries ....................................................... 154
Summary ................................................................................... 159

17 **Full-Text Searching** .......................................................... 161
Understanding Full-Text Searching ............................................. 161
Setting Up Full-Text Searching ................................................... 162
Performing Full-Text Searches ................................................... 166
Summary ................................................................................... 174

18 **Inserting Data** ................................................................. 175
Understanding Data Insertion .................................................... 175
Inserting Complete Rows .......................................................... 176
Inserting Multiple Rows ............................................................ 179
Inserting Retrieved Data ........................................................... 180
Reporting On Inserted Data ....................................................... 183
Summary ................................................................................... 184

19 **Updating and Deleting Data** ............................................. 185
Updating Data ........................................................................... 185
Deleting Data ............................................................................ 187
Reporting On Updated And Deleted Data ................................... 188
Guidelines for Updating and Deleting Data ............................... 189
Summary ................................................................................... 190

20 **Creating and Manipulating Tables** ................................... 191
Creating Tables ........................................................................ 191
Updating Tables ...................................................................... 199
Deleting Tables ...................................................................... 201
Renaming Tables ........................................................... 201
Summary ........................................................................ 202

21 Using Views .................................................................. 203
Understanding Views ...................................................... 203
Using Views ..................................................................... 205
Summary ......................................................................... 212

22 Programming with T-SQL .............................................. 213
Understanding T-SQL Programming .............................. 213
Using Variables .............................................................. 214
Using Conditional Processing ....................................... 221
Grouping Statements ...................................................... 223
Using Looping ................................................................. 225
Summary ......................................................................... 226

23 Working with Stored Procedures .................................. 227
Understanding Stored Procedures ................................. 227
Why Use Stored Procedures ........................................... 228
Using Stored Procedures ................................................. 229
Summary ......................................................................... 238

24 Using Cursors .............................................................. 239
Understanding Cursors .................................................. 239
Working with Cursors .................................................... 239
Summary ......................................................................... 246

25 Using Triggers .............................................................. 247
Understanding Triggers .................................................... 247
Using Triggers ................................................................. 250
Summary ......................................................................... 254
## Contents

26 **Managing Transaction Processing** 255  
  Understanding Transaction Processing ....................................................... 255  
  Controlling Transactions ........................................................................... 257  
  Summary .................................................................................................. 262

27 **Working with XML and JSON** 263  
  Using SQL Server XML Support ................................................................. 263  
  Using SQL Server JSON Support .............................................................. 272  
  Retrieving Data as JSON .......................................................................... 273  
  The JSON Functions ................................................................................. 275  
  Summary .................................................................................................. 275

28 **Globalization and Localization** 277  
  Understanding Character Sets and Collation Sequences ............................ 277  
  Working with Collation Sequences ............................................................ 278  
  Managing Case Sensitivity ......................................................................... 281  
  Working with Unicode .............................................................................. 283  
  Summary .................................................................................................. 285

29 **Managing Security** 287  
  Understanding Access Control .................................................................... 287  
  Managing Users ....................................................................................... 289  
  Managing Access Rights ............................................................................ 291  
  Summary .................................................................................................. 293

30 **Improving Performance** 295  
  Improving Performance .............................................................................. 295  
  Summary .................................................................................................. 297

A **The Example Tables** 299  
  Understanding the Example Tables ............................................................. 299
About the Author

Ben Forta is Adobe Systems’ Senior Director of Education Initiatives and has more than three decades of experience in the computer industry in product development, support, training, and product marketing. He is the author of the best-selling Sams Teach Yourself SQL in 10 Minutes, spinoff titles (like this one), the ColdFusion Web Application Construction Kit series, Sams Teach Yourself Regular Expressions in 10 Minutes, as well as books on Flash, Java, Windows, and other subjects. He has extensive experience in database design and development, has implemented databases for several highly successful commercial software programs and websites, and is a frequent lecturer and columnist on Internet and database technologies. Ben lives in Oak Park, Michigan, with his wife Marcy and their children. Ben welcomes your e-mail at ben@forta.com and invites you to visit his website at http://forta.com/ and to follow him on Twitter at @benforta.
We Want to Hear from You!

As the reader of this book, you are our most important critic and commentator. We value your opinion and want to know what we’re doing right, what we could do better, what areas you’d like to see us publish in, and any other words of wisdom you’re willing to pass our way.

You can email or write directly to let us know what you did or didn’t like about this book—as well as what we can do to make our books stronger.

*Please note that we cannot help you with technical problems related to the topic of this book, and that due to the high volume of mail we receive, we might not be able to reply to every message.*

When you write, please be sure to include this book’s title and author as well as your name and phone or email address. We will carefully review your comments and share them with the author and editors who worked on the book.

E-mail: feedback@samspublishing.com

Mail: Reader Feedback
Sams Publishing
800 East 96th Street
Indianapolis, IN 46240 USA
Introduction

SQL Server has become one of the most popular database management systems in the world. From small development projects to some of the best-known and most prestigious sites on the Web, SQL Server has proven itself to be a solid, reliable, fast, and trusted solution to all sorts of data-storage needs.

This book is based on my bestselling book *Sams Teach Yourself SQL in 10 Minutes*, which has become the most-used SQL tutorial in the world, with an emphasis on teaching what you really need to know—methodically, systematically, and simply. But as popular and as successful as that book is, it does have some limitations:

- In covering all the major DBMSs, coverage of DBMS-specific features and functionality had to be kept to a minimum.

- To simplify the SQL taught, the lowest common denominator had to be found—SQL statements that would (as much as possible) work with all major DBMSs. This requirement necessitated that better DBMS-specific solutions not be covered.

- Although basic SQL tends to be rather portable between DBMSs, more advanced SQL most definitely is not. As such, that book could not cover advanced topics, such as triggers, cursors, stored procedures, access control, transactions, and more in any real detail.

And that is where this book comes in. *Sams Teach Yourself Microsoft SQL Server T-SQL in 10 Minutes* builds on the proven tutorials and structure of *Sams Teach Yourself SQL in 10 Minutes*, without getting bogged down with anything but Transact-SQL (T-SQL, for short). It starts with simple data retrieval and works on to more complex topics, including the use of joins, subqueries, full text–based searches, functions and stored procedures, cursors, triggers, table constraints, XML and JSON, and much more.
You’ll learn what you need to know methodically, systematically, and simply—in highly focused lessons designed to make you immediately and effortlessly productive.

NOTE: *Written for SQL Server 2012 through 2016*

This book was written with SQL Server 2012 through 2016 in mind, and covers features and technologies new to these versions of the software.

So turn to Lesson 1, “Understanding SQL,” and get to work. You’ll be taking advantage of all SQL Server has to offer in no time at all.

**Who Is This Book For?**

This book is for you if…

- You are new to SQL.
- You are just getting started with SQL Server and want to hit the ground running.
- You want to quickly learn how to get the most out of SQL Server and T-SQL.
- You want to learn how to use T-SQL in your own application development.
- You want to be productive quickly and easily using SQL Server without having to call someone for help.

**Companion Website**

This book has a companion website online at http://forta.com/books/0672337924/. Visit the site to access the following:

- Table creation and population scripts used to create the sample tables used throughout this book
- The online support forum
Online errata (should one be required)

Other books that may be of interest to you

Conventions Used in This Book

This book uses different typefaces to differentiate between code and regular English, and also to help you identify important concepts.

Text that you type and text that should appear on your screen is presented in monospace type. It looks like this to mimic the way text looks on your screen.

Placeholders for variables and expressions appear in monospace italic font. You should replace the placeholder with the specific value it represents.

This arrow (➥) at the beginning of a line of code means that a single line of code is too long to fit on the printed page. Continue typing all the characters after the ➥ as though they were part of the preceding line.

NOTE: A note presents interesting pieces of information related to the surrounding discussion.

TIP: A tip offers advice or teaches an easier way to do something.

CAUTION: A caution advises you about potential problems and helps you steer clear of disaster.

NEW TERMS: New Term icons provide clear definitions of new, essential terms.
Input ▼

Input identifies code that you can type in yourself. It usually appears next to a listing.

Output ▼

Output highlights the output produced by running T-SQL code. It usually appears after a listing.

Analysis ▼

Analysis alerts you to the author’s line-by-line analysis of input or output.
In this lesson, you learn how to connect and log into SQL Server, how to issue SQL Server statements, and how to obtain information about databases and tables. You also create the example tables we’ll be using throughout this book.

Making the Connection

Now that you have a SQL Server DBMS and client software to use with it, it would be worthwhile to briefly discuss connecting to the database.

SQL Server, like all client/server DBMSs, requires that you log into the DBMS before you’re able to issue commands. SQL Server can authenticate users and logins using its own user list or using the Windows user list (the logins used to start using Windows). As such, depending on how SQL Server is configured, it may log you in automatically using whatever login you used for Windows itself, or it may prompt you for a login name and password.

When you first installed SQL Server, you may have been prompted for an administrative login (often named sa for system administrator) and a password. If you are using your own local server and are simply experimenting with SQL Server, using this login is fine. In the real world, however, the administrative login is closely protected because access to it grants full rights to create tables, drop entire databases, change logins and passwords, and more.
To connect to SQL Server, you need the following pieces of information:

- The hostname (the name of the computer). This is localhost or your own computer name if you’re connecting to a local SQL Server.
- A valid username (if Windows authentication is not being used).
- The user password (if required).

**TIP: Simplified Local Logins**

If you are using a locally installed SQL Server and SQL Server Management Studio, you should be able to just click the Connect button when prompted to Connect.

If you’re using one of the client applications discussed in the previous lesson, a dialog box will be displayed to prompt you for this information.

**NOTE: Using Other Clients**

If you are using a client other than Microsoft SQL Server Management Studio you still need to provide this information in order to connect to SQL Server.

After you are connected, you have access to whatever databases and tables your login name has access to. (Logins, access control, and security are revisited in Lesson 29, “Managing Security.”)

**Navigating SQL Server Management Studio**

Because we’ll be spending so much time in Microsoft SQL Server Management Studio, a quick tour of the tool is worthwhile:

- At the top of the screen is a toolbar containing lots of buttons. One of those buttons is labeled New Query, and when you click
Navigating SQL Server Management Studio

on it you open a window in which you can type and execute SQL. You’ll be using this button and window extensively.

► The left column is the Object Explorer used to browse databases, tables, and more. You can right-click on objects to see their details, edit them, and perform other tasks too. Click on the + next to Databases to see the databases that have been already created.

► On the left of the toolbar, right above the Object Explorer, is a drop-down box (the first time you open Microsoft SQL Server Management Studio this box will likely contain the word master). This drop-down box is used to select the database you want to work with (as you will recall from Lesson 1, a single DBMS can work with lots of databases). We’ll create our own database shortly, which will then appear in this drop-down box.

► On the right side of the screen is a Properties panel. If it is not visible, don’t worry, it’ll open when there are properties to be displayed.

► Not visible initially is the Results panel. This appears under the Query window and displays query results when there are results to show.

There are many ways to use Microsoft SQL Server Management Studio, but here are the basic steps needed to enter and execute SQL statements:

► As just noted, the New Query button at the top-left of the screen opens a window in which SQL statements are entered.

► As T-SQL statements are typed, Microsoft SQL Server Management Studio automatically color-codes the statements and text (this is an invaluable troubleshooting tool because it lets you quickly spot typos or missing quotes and so on).

► To execute (run) a statement, click the Execute button (the one with the red exclamation point on it). You can also press F5 or Ctrl+E to execute a statement.
LESSON 3: Working with SQL Server

▲ To verify that a SQL statement is syntactically correct (without executing it), click the Parse button (the one with the blue checkmark on it).

▲ Microsoft SQL Server Management Studio displays statement results at the bottom of the screen. Results may be displayed in a grid (the default behavior), as plain text, or saved to a file. You can switch between these modes by clicking the appropriate toolbar buttons.

▲ In addition to displaying statement results, Microsoft SQL Server Management Studio also displays status messages (the number of rows returned, for example) in a second tab labeled Messages.

▲ To obtain help, click the statement you need help with and press F1.

Creating the Example Tables

In order to follow along with the examples, you need the example database tables. Everything you need to get up and running can be found on this book’s web page at http://www.forta.com/books/0672337924/.

The web page contains two SQL script files that you may download. These files are plain text files that contain SQL statement that create and populate the example tables we’ll be using. Download both files:

▲ create.sql contains the T-SQL statements to create the six database tables (including the statements to define all primary keys and foreign key constraints).

▲ populate.sql contains the SQL INSERT statements used to populate these tables.

NOTE: For SQL Server Only

The SQL statements in the downloadable .sql files are very DBMS-specific and are designed to be used only with Microsoft SQL Server. The scripts have been tested extensively with SQL Server 2012 and later, and have not been tested with earlier versions of SQL Server.
After you have downloaded the scripts, you can use them to create and populate the tables needed to follow along with the lessons in this book. Here are the steps to follow:

1. To be safe, we’ll use a dedicated database to learn T-SQL, this way you won’t be able to mistakenly access or change other data. In Microsoft SQL Server Management Studio, right click on Databases in the Object Explorer and select New Database ... to display the New Database dialog. Specify learnsql as your new database name (or pick any name of your choice, but do not use any existing database name, just to be on the safe side). You can ignore all other options in the dialog box and just click OK to create the database.

2. Now that you have a database, you can create the example tables. Click New Query to open a query window. Make sure the new database is selected from the list of databases from the drop-down list (or you’ll create tables in the wrong database).

3. First you need to execute the create.sql script. You may simply copy and paste the entire contents of the file into the query window, or you can use the File menu options to open create.sql directly. When you have the SQL code in the Query Window, click the Execute button to run the script and create the tables.

4. Repeat the previous step using the populate.sql file to populate the new tables. Again, make sure you have the right database selected.

And with that you should be good to go!
Selecting a Database

As you’ve just seen, when you first connect to SQL Server, a default database is opened for you. This is usually a database named `master` (which as a rule you should never play with). Before you perform any database operations, you need to select the appropriate database. You did this using the drop-down box in the toolbar, but you can also select the database using the SQL `USE` keyword.

**NEW TERM: Keyword**
A reserved word that is part of the T-SQL language. Never name a table or column using a keyword. Appendix E, “T-SQL Reserved Words,” lists the SQL Server keywords.

For example, to use the `learnsql` database, you would enter the following (in a query window) and execute it:

```sql
USE learnsql;
```

**TIP: SQL Server Management Studio Is Smart**
SQL Server Management Studio helps you write SQL code. In addition to the color-coding, which you undoubtedly noticed, it also tries to give you options so that you write less code (and make fewer mistakes). As such, when you typed the `USE` statement it popped up an inline database list for selection. You’ll see lots of this type of help and behavior when using the tool.

**Output ▼**

Command(s) completed successfully.

**Analysis ▼**

The `USE` statement does not return any results, but you will see a message confirming that the command was successfully completed.
Learning About Databases and Tables

Remember, you must always `USE` a database before you can access any data in it.

**Learning About Databases and Tables**

But what if you don’t know the names of the available databases? And for that matter, how do the client applications obtain the list of available databases that are displayed in the drop-down list?

Information about databases, tables, columns, users, privileges, and more, are stored within databases and tables themselves (yes, SQL Server uses SQL Server to store this information). These internal tables are all in the *master* database (which is why you don’t want to tamper with it), and they are generally not accessed directly. Instead, SQL Server includes a suite of prewritten stored procedures that can be used to obtain this information (information that SQL Server then extracts from those internal tables).

**NOTE: Stored Procedures**

Stored procedures will be covered in Lesson 23, “Working with Stored Procedures.” For now, it suffices to say that stored procedures are SQL statements that are saved in SQL Server and can be executed as needed.
Look at the following example:

**Input ▼**

```
sp_databases;
```

**Output ▼**

<table>
<thead>
<tr>
<th>DATABASE_NAME</th>
<th>DATABASE_SIZE</th>
<th>REMARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>coldfusion</td>
<td>9096</td>
<td>NULL</td>
</tr>
<tr>
<td>learnsql</td>
<td>3072</td>
<td>NULL</td>
</tr>
<tr>
<td>forta</td>
<td>2048</td>
<td>NULL</td>
</tr>
<tr>
<td>master</td>
<td>4608</td>
<td>NULL</td>
</tr>
<tr>
<td>model</td>
<td>1728</td>
<td>NULL</td>
</tr>
<tr>
<td>msdb</td>
<td>5824</td>
<td>NULL</td>
</tr>
<tr>
<td>tempdb</td>
<td>8704</td>
<td>NULL</td>
</tr>
</tbody>
</table>

**Analysis ▼**

`sp_databases;` returns a list of available databases. Included in this list might be databases used by SQL Server internally (such as `master` and `tempdb` in this example). Of course, your own list of databases might not look like those shown above.

To obtain a list of tables within a database, make sure you are using the right database, and then use `sp_tables;`, as seen here:

**Input ▼**

```
sp_tables;
```

**Analysis ▼**

`sp_tables;` returns a list of available tables in the currently selected database, and not just your tables; it also includes all sorts of system tables and other entries (possibly hundreds of entries).

To obtain a list of tables (just tables, not views, and not system tables and so on), you can use this statement:

**Input ▼**

```
sp_tables NULL, dbo, learnsql, "'TABLE'";
```
Learning About Databases and Tables

Output ▼

<table>
<thead>
<tr>
<th>TABLE_QUALIFIER</th>
<th>TABLE_OWNER</th>
<th>TABLE_NAME</th>
<th>TABLE_TYPE</th>
<th>REMARKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>crashcourse</td>
<td>dbo</td>
<td>customers</td>
<td>TABLE</td>
<td>NULL</td>
</tr>
<tr>
<td>crashcourse</td>
<td>dbo</td>
<td>orderitems</td>
<td>TABLE</td>
<td>NULL</td>
</tr>
<tr>
<td>crashcourse</td>
<td>dbo</td>
<td>orders</td>
<td>TABLE</td>
<td>NULL</td>
</tr>
<tr>
<td>crashcourse</td>
<td>dbo</td>
<td>products</td>
<td>TABLE</td>
<td>NULL</td>
</tr>
<tr>
<td>crashcourse</td>
<td>dbo</td>
<td>vendors</td>
<td>TABLE</td>
<td>NULL</td>
</tr>
<tr>
<td>crashcourse</td>
<td>dbo</td>
<td>productnotes</td>
<td>TABLE</td>
<td>NULL</td>
</tr>
<tr>
<td>crashcourse</td>
<td>dbo</td>
<td>sysdiagrams</td>
<td>TABLE</td>
<td>NULL</td>
</tr>
</tbody>
</table>

Analysis ▼

Here, `sp_tables` accepts a series of parameters telling it which database to use, as well as what specifically to list (`'TABLE'` as opposed to `'VIEW'` or `'SYSTEM TABLE'`).

`sp_columns` can be used to display a table’s columns:

Input ▼

```
sp_columns customers;
```

NOTE: Shortened for Brevity

`sp_columns` returns lots of data. In the output that follows, I truncated the display because the full output would have been far wider than the pages in this book, likely requiring many lines for each row.

Output ▼

<table>
<thead>
<tr>
<th>TABLE_QUALIFIER</th>
<th>TABLE_OWNER</th>
<th>TABLE_NAME</th>
<th>COLUMN_NAME</th>
<th>DATA_TYPE</th>
<th>TYPE_NAME</th>
</tr>
</thead>
<tbody>
<tr>
<td>learnsql</td>
<td>dbo</td>
<td>customers</td>
<td>cust_id</td>
<td>4</td>
<td>int identity</td>
</tr>
<tr>
<td>learnsql</td>
<td>dbo</td>
<td>customers</td>
<td>cust_name</td>
<td>-8</td>
<td>nchar</td>
</tr>
<tr>
<td>learnsql</td>
<td>dbo</td>
<td>customers</td>
<td>cust_address</td>
<td>-8</td>
<td>nchar</td>
</tr>
<tr>
<td>learnsql</td>
<td>dbo</td>
<td>customers</td>
<td>cust_city</td>
<td>-8</td>
<td>nchar</td>
</tr>
<tr>
<td>learnsql</td>
<td>dbo</td>
<td>customers</td>
<td>cust_state</td>
<td>-8</td>
<td>nchar</td>
</tr>
<tr>
<td>learnsql</td>
<td>dbo</td>
<td>customers</td>
<td>cust_zip</td>
<td>-8</td>
<td>nchar</td>
</tr>
<tr>
<td>learnsql</td>
<td>dbo</td>
<td>customers</td>
<td>cust_country</td>
<td>-8</td>
<td>nchar</td>
</tr>
<tr>
<td>learnsql</td>
<td>dbo</td>
<td>customers</td>
<td>cust_contact</td>
<td>-8</td>
<td>nchar</td>
</tr>
<tr>
<td>learnsql</td>
<td>dbo</td>
<td>customers</td>
<td>cust_email</td>
<td>-8</td>
<td>nchar</td>
</tr>
</tbody>
</table>
Analysis ▼

sp_columns requires that a table name be specified (customers in this example), and returns a row for each field, containing the field name, its datatype, whether NULL is allowed, key information, default value, and much more.

NOTE: What Is Identity?

Column cust_id is an identity column. Some table columns need unique values (for example, order numbers, employee IDs, or, as in the example just shown, customer IDs). Rather than have to assign unique values manually each time a row is added (and having to keep track of what value was last used), SQL Server can automatically assign the next available number for you each time a row is added to a table. This functionality is known as identity. If it is needed, it must be part of the table definition used when the table is created using the CREATE statement. We’ll look at CREATE in Lesson 20, “Creating and Manipulating Tables.”

Lots of other stored procedures are supported, too, including:

- sp_server_info—Used to display extensive server status information
- sp_spaceused—Used to display the amount of space used (and unused) by a database
- sp_statistics—Used to display usage statistics pertaining to database tables
- sp_helpuser—Used to display available user accounts
- sp_helplogins—Used to display user logins and what they have rights to

It is worthwhile to note that client applications use these same stored procedures you’ve seen here. Applications that display interactive lists of databases and tables, that allow for the interactive creation and editing of tables, that facilitate data entry and editing, or that allow for user account
and rights management, and more, all accomplish what they do using the same stored procedures that you can execute directly yourself.

**Summary**

In this lesson, you learned how to connect and log into SQL Server, how to select databases using `USE`, and how to introspect SQL databases, tables, and internals using stored procedures. Armed with this knowledge, you can now dig into the all-important `SELECT` statement.
This page intentionally left blank
Symbols

* (asterisk)
  multiplication operator, 80
  wildcard character, 35–36, 101
@ (at symbol), 214, 233
[ ] (brackets), 69–70, 275
/ (division) operator, 80
= (equality) operator, 51
> (greater than) operator, 51
>= (greater than or equal to) operator, 51
< (less than) operator, 51, 52
<= (less than or equal to) operator, 51
<> (nonequality) operator, 51, 53
!= (nonequality) operator, 51, 53
!> (not greater than) operator, 51
!< (not less than) operator, 51
( ) (parentheses), 61
% (percent sign)
  modulo operator, 80
  wildcard character, 66–68
+ (plus sign)
  addition operator, 80
  concatenation operator, 74–77
; (semicolon), 33, 179–180
'
(single quotes), 53, 169, 220
- (subtraction) operator, 80
_ (underscore), 68–69

A

Abs() function, 94
ABSOLUTE keyword, 243
absolute value, returning, 94
access control
  access rights, 291–293
  explained, 287–288
  logins, renaming, 291
  user accounts
    creating, 290
    deleting, 290
    enabling/disabling, 290
  user management, 289
access rights, 291–293
accessing SQL Server, 16–17
accounts (user)
  creating, 290
  deleting, 290
  enabling/disabling, 290
addition operator (+), 80
administrative login, 19
AFTER triggers, 249
aggregate functions
  Avg(), 98–99
  combining, 106–107
  Count(), 100–101, 106
  distinct values, 105
  explained, 97–98
  with joins, 148–150
  Max(), 101–102, 106
  Min(), 102–103
  Sum(), 103–104
aliases
  calculated field aliases, 77–78
  naming, 107
  table aliases, 141–142
all columns, retrieving, 35–36
ALL keyword, 105
ALTER FULLTEXT statement, 165–166
ALTER LOGIN statement, 290–291
ALTER TABLE statement, 199–200, 280, 291
ALTER TRIGGER statement, 249
altering tables, 199–200
AND operator, 57–58, 170
application level, filtering data at, 50
ASC keyword, 46–47
ascending order, sorting data in, 44–46
assigning
  search term weights, 173
  values to variables, 215–216
asterisk (*)
  multiplication operator, 80
  wildcard character, 35–36, 101
AUTO keyword, 266
autocommits, 259, 261
averages, returning, 98–99
Avg() function, 98–99
B
batch statements, 219
BEGIN statement, 223–224
BETWEEN operator, 51, 54
Boolean operators. See operators
brackets ([ ]), 69–70, 275
BREAK statement, 226
C
calculated fields
  aliases, 77–78
  concatenating, 74–77
  explained, 73–74
  mathematical calculations, 79–80
  subqueries as, 123–126
  views, 210–211
Cartesian products, 133–137
case sensitivity
  collation sequences, 281–283
data sorting, 46
searches, 67
statements, 33
Cast() function, 270
catalogs, full-text
  creating, 163
dropping, 165–166
  managing, 165–166
character sets
  definition of, 277
  Unicode, 283–285
CharIndex() function, 85
columns

checking
for existance, 126–128
for nonmatches, 53
for NULL values, 54–55
for range of values, 54
against single values, 51–52
FROM clause, 133
clauses. See also keywords; statements
ELSE, 222
FOR JSON AUTO, 273
FOR XML, 264–268
FROM, 133
FULL OUTER JOIN, 148
GROUP BY, 110–111, 115–116
HAVING, 111–114
INNER JOIN, 137–138
JSON PATH, 274
ORDER BY
example, 115–116
sorting by multiple columns, 43–44
sorting by single column, 41–43
order of, 117
OUTER JOIN, 146–147
OUTPUT
DELETE statement, 188–189
INSERT statement, 183–184
UPDATE statement, 188–189
WHERE
AND operator, 57–58
brackets ([ ]) wildcard, 69–70
comparison operators. See comparison operators
joins, 133–137
LIKE operator, 65–66
nonmatches, checking for, 53
NULL values, checking for, 54–55
percent sign (%) wildcard, 66–68
range of values, checking for, 54
single values, checking against, 51–52
subqueries, 119–128
syntax, 49–50
underscore (_) wildcard, 68–69
views, 209
client server software, 14–15
client utilities, 15–16
CLOSE statement, 241–242
closing cursors, 241–242
COLLATE keyword, 281–283
collations
case sensitivity, 281–283
collation sequences, 278–280
definition of, 277
importance of, 278
columns. See also fields
aliases, 77–78
average value of, 98–99
deleting
deleted data, reporting on, 188–189
guidelines, 188–189
syntax, 187–188
explained, 7–8
fully qualified column names, 133
identity columns, 28
maximum value, 101–102
minimum value, 102–103
omitting from INSERT operations, 179
returning list of, 27–28
sorting by
  * multiple columns, 43–44
  * single columns, 41–43
totals, 103–104
updating
  * guidelines, 188–189
  * syntax, 185–187
updated data, reporting on, 188–189
combined queries
  * creating, 154–157
  * duplicate rows, including/eliminating, 157–158
  * explained, 153
  * sorting results of, 158–159
combining aggregate functions, 106–107
comments, 220
COMMIT statement, 259
commits
  * autocommits, 259, 261
  * explicit commits, 259
comparison operators
  * BETWEEN operator, 54
  * equality (=) operator, 51
  * IS NULL operator, 54–55
  * less than (<) operator, 52
  * nonequality (<> operator, 53
  * nonequality (!=) operator, 53
  * table of, 49–50
complete rows, inserting, 176–179
concatenating fields, 74–77
concatenation (+) operator, 74–77
conditional processing
  * BEGIN statement, 223–224
  * END statement, 223–224
  * IF statement, 221–223
  * WHILE statement, 225–226
configuring full-text searching
  * full-text catalogs, 163–166
  * full-text indexes, 164–166
  * full-text search support, 162–163
connecting to SQL Server, 19–20
CONTAINS keyword, 168–172
CONTAINSTABLE() function, 172–173
contents of variables, viewing, 216–217
CONTINUE statement, 226
controlling access. See access control
converting
  * numeric values to strings, 85
  * strings to lowercase, 85
  * strings to uppercase, 85
  * variables to strings, 217
correlated subqueries, 125–126
Cos() function, 94
cosine, returning, 94
Count() function
  * DISTINCT keyword, 100–101
  * explained, 100–101
  * as subqueries, 124–125
counting rows, 100–101
CREATE FULLTEXT CATALOG statement, 163
CREATE FULLTEXT INDEX statement, 164
CREATE LOGIN statement, 290
CREATE PROCEDURE statement, 230–231
CREATE TABLE statement
  collation sequences, 280
DEFAULT keyword, 197–198
default values, 197–198
  example, 269
IDENTITY keyword, 195–197
NULL values, 193–194
primary keys, 194–195
syntax, 192–193
CREATE TRIGGER statement, 248
CREATE VIEW statement, 205
create.sql file, 22
  creating
  calculated fields
    aliases, 77–78
    concatenating, 74–77
    explained, 73–74
    mathematical calculations, 79–80
combined queries, 154–157
cursors, 240–241
full-text catalogs, 163
full-text indexes, 164–165
groups, 110–111
joins, 132–133
sample tables, 22–23
stored procedures, 230–231
tables
  basic table creation, 192–193
DEFAULT keyword, 197–198
default values, 197–198
  explained, 191
IDENTITY keyword, 195–197
NULL values, 193–194
primary keys, 194–195
triggers, 248
user accounts, 290
views, 205
cross joins, 136
cursors
  closing, 241–242
  creating, 240–241
  deleting, 240–241
  explained, 239–240
  fetched data, 242–245
  opening, 241–242
  with WHILE loops, 225
cursor data, fetching, 242–245
deleting
  deleted data, reporting on, 188–189
  guidelines, 188–189
  syntax, 187–188
filtering
  at application level, 50
  comparison operators, 49–50
  filter conditions, 51–55
  groups, 111–114
  IS NULL operator, 54–55
  multiple WHERE clauses, 57–64
  by subqueries, 119–123
  views, 209
  WHERE clause syntax, 49–50
  wildcard filtering, 65–70
formatting
  client versus server formatting, 74
  overview, 35
  queries, 122
  retrieved data, 207–208
  statements, 193

groups
  creating, 110–111
  explained, 109–110
  filtering, 111–114
  grouping versus sorting, 114–116

inserting
  complete rows, 176–179
  explained, 175
  list of changes, returning, 183–184
  multiple rows, 179–180
  retrieved data, 180–183

JSON (JavaScript Object Notation) data
  functions, 275
  retrieving data as JSON, 273–274

retrieving
  all columns, 35–36
  distinct rows, 36–37
  fully qualified table names, 39–40
  individual columns, 31–33
  multiple columns, 33–34
  overview, 31
  results, limiting, 37–39

sorting
  ascending order, 46–47
  case sensitivity, 46
  descending order, 44–46

  dictionary sort order, 46
  grouping versus sorting, 114–116
  by multiple columns, 43–44
  by single column, 41–43

  summarizing with aggregate functions
    Avg(), 98–99
    Count(), 100–101, 106
    distinct values, 105
    explained, 97–98
    Max(), 101–102, 106
    Min(), 101–102
    multiple aggregate functions, 106–107
    Sum(), 103–104

updating
  guidelines, 188–189
  syntax, 185–187
  updated data, reporting on, 188–189

XML (Extensible Markup Language) data
  retrieving, 264–268
  searching for, 271–272
  storing, 268–270
  support for, 263
  XML schemas, 271

Database Engine Tuning Advisor, 296–297
Database Management System (DBMS), 6. See also SQL Server
database servers, 14
DatabasePropertyEX() function, 279
databases. See also tables
  definition of, 6
  everyday uses of, 5
  master databases, 25
DROP VIEW statement

returning list of, 26
scaling, 131
selecting, 24–25
SYS, 289
datatatypes
date time, 91
explained, 7–8
NCHAR, 283
NTEXT, 283
NVARCHAR, 283
parameter datatypes, 233
date manipulation functions, 87–94
date time datatype, 91
DateAdd() function, 87
DateDiff() function, 87, 91–93
DateName() function, 87
DatePart() function, 87–90, 221
Day() function, 87, 92–93
DBMS (Database Management System), 6. See also SQL Server
DEALLOCATE statement, 240–241
debugger, 218
declaring variables, 214–215
DEFAULT keyword, 197–198
default values
assigning to variables, 215
specifying, 197–198
DELETE statement
DELETE FROM, 188
DELETE triggers, 252
deleted data, reporting on, 188–189
guidelines, 188–189
syntax, 187–188
DELETE triggers, 252
deleting

data
deleted data, reporting on, 188–189
guidelines, 188–189
syntax, 187–188
tables, 201
user accounts, 290

derived columns (aliases), 78
DESC keyword, 44–46
descending order, sorting data in, 44–46
dialogs, New Database, 23
dictionary sort order, 46
different tables, combining, 159
DISABLE TRIGGER statement, 250
disabling
INSERT statement, 175
triggers, 249–250
user accounts, 290
DISTINCT keyword, 36–37, 105–106
distinct values, aggregates on, 105
division operator (/), 80
documentation, 297
downloading SQL Server, 17
DROP FULLTEXT statement, 165–166
DROP LOGIN statement, 290
DROP PROCEDURE statement, 231–232
DROP TABLE statement, 201
DROP TRIGGER statement, 249
DROP VIEW statement, 205
dropping
- catalogs/indexes, 165–166
- logins, 290
- stored procedures, 231–232
- tables, 201
- triggers, 249
- views, 205
duplicate rows in unions, including/eliminating, 157–158

E

ELEMENTS keyword, 266–267
eliminating duplicate rows in unions, 157–158
ELSE clause, 222
empty strings, 194
ENABLE TRIGGER statement, 250
enabling
  - full-text searching, 162–163
  - triggers, 249–250
  - user accounts, 290
encodings, 277
END statement, 223–224
equality (=) operator, 51
equijoins, 137
evaluation, order of, 59–61, 80
EXEC statement, 230
EXECUTE statement, 230
executing stored procedures, 230
exist function, 271
existence, checking for, 126–128
EXISTS keyword, 126–128
Exp() function, 94
explicit commits, 259
EXPLICIT keyword, 267–268
exponential values, returning, 94

F

FETCH statement, 242–245
@@FETCH_STATUS function, 244
fetching data, 242–245
fields. See also columns
calculated fields
  - aliases, 77–78
  - concatenating, 74–77
  - explained, 73–74
  - mathematical calculations, 79–80
  - subqueries as, 123–126
  - views, 210–211
identity fields, 176–177
files
  - create.sql, 22
  - populate.sql, 22
filter conditions
  - comparison operators, 49–50
    - AND operator, 57–58
    - IN operator, 61–63
    - NOT operator, 63–64
    - OR operator, 58–59
    - order of evaluation, 59–61
  - nonmatches, 53
  - NULL values, 54–55
  - range of values, 54
  - single values, 51–52
WHERE clause, 49–50
wildcard filtering
  - brackets ([ ]), 69–70
  - case sensitivity, 67
  - LIKE operator, 65–66
  - NULL values, 68
percent sign (%) wildcard, 66–68
tips for, 71
underscore (_) wildcard, 68–69

filtering data
at application level, 50
comparison operators, 49–50
groups, 111–114
IS NULL operator, 54–55
multiple WHERE clauses
NOT operator, 63–64
AND operator, 57–58
OR operator, 58–59
IN operator, 61–63
order of evaluation, 59–61
nonmatches, 53
NULL values, 54–55
range of values, 54
single value, 51–52
by subqueries, 119–123
views, 209
WHERE clause syntax, 49–50
wildcard filtering
brackets ([ ]) wildcard, 69–70
case sensitivity, 67
LIKE operator, 65–66
NULL values, 68
percent sign (%) wildcard, 66–68
tips for, 71
trailing spaces, 68
underscore (_) wildcard, 68–69
FIRST keyword, 243
fn_helpcollations() function, 278, 279–280
foreign keys, 11, 130
formatting
client versus server formatting, 74
overview, 35
queries, 122
retrieved data, 207–208
statements, 193
FORMSOF() function, 172
FOR XML clause, 264–268
four-digit year format, 90
FREETEXT keyword, 166–168
FULL OUTER JOIN clause, 148
full-text catalogs
creating, 163
managing, 165–166
full-text indexes
creating, 164–165
managing, 165–166
full-text searching
enabling, 162–163
explained, 161–162
full-text catalogs
creating, 163
managing, 165–166
full-text indexes
creating, 164–165
managing, 165–166
performance, 161
performing with CONTAINS, 168–172
performing with FREETEXT, 166–168
search results, ranking, 172–173
FulltextCatalogProperty()
function, 166
FULLTEXTTABLE() function, 172–173
fully qualified column names, 133
fully qualified table names, 39–40
functions

Abs(), 94
Avg(), 98–99
Cast(), 270
CharIndex(), 85
CONTAINSTABLE(), 172–173
Cos(), 94
Count()

DISTINCT keyword, 106
explained, 100–101
as subqueries, 124–125
DatabasePropertyEX(), 279
DateAdd(), 87
DateDiff(), 87, 91–93
DateName(), 87
DatePart(), 87–90, 221
Day(), 87, 92–93
exist, 271
Exp(), 94
explained, 83
@@FETCH_STATUS, 244
fn_helpcollations(), 278,
279–280
FORMSOF(), 172
FulltextCatalogProperty(), 166
FULLTEXTTABLE(), 172–173
GetDate(), 87, 198, 221
@@IDENTITY system
function, 197
ISABOUT(), 173
ISJSON(), 275
JSON_MODIFY(), 275
JSON_QUERY(), 275
JSON_VALUE(), 275
Left(), 85
Len(), 85
Lower(), 85
LTrim(), 77, 85
Max()

DISTINCT keyword, 106
explained, 101–102
order numbers,
determining, 196
Min(), 102–103
modify, 271
Month(), 87, 92–93
nodes, 271
Pi(), 94
portability, 83
query, 271
Rand(), 94
Replace(), 85
Right(), 85
Round(), 94
RTrim(), 76–77, 85
ServerProperty(), 279
Sin(), 94
Soundex(), 85–86
Sqrt(), 94
Square(), 94
Str(), 85
SubString(), 85
Sum(), 103–104
Tan(), 94
testing, 80
types of

aggregate functions,
97–107
date and time manipulation
functions, 87–94
numeric manipulation
functions, 94
overview, 84
text manipulation functions,
84–86
Upper(), 84–85
value, 271
Year(), 87, 92–93

GetDate() function, 198, 221
globalization
  character sets
definition of, 277
  Unicode, 283–285
collations
  case sensitivity, 281–283
collation sequences,
    278–280
definition of, 277
  importance of, 278
encodings, 277
GRANT statement, 291–292
granting access rights, 291–292
great than (>) operator, 51
greater than or equal to (>=) operator, 51
GROUP BY clause (SELECT statement), 110–111, 115–116
grouping statements, 223–224
groups
  creating, 110–111
  explained, 109–110
  filtering, 111–114
  grouping versus sorting, 114–116

HAVING clause (SELECT statement), 111–114
hostname, 20

identity, 28
identity fields, 176–177
IDENTITY keyword, 195–197
@@IDENTITY system function, 197
IF statement, 221–223
IMPLICIT_TRANSACTIONS setting, 261
improving performance, 295–297
including duplicate rows in unions, 157–158
incrementally building queries, 128
indexes
  full-text indexes
    creating, 164–165
    managing, 165–166
  updating, 165
individual columns, retrieving, 31–33
INNER JOIN clause, 137–138
inner joins, 137
IN operator, 61–63
INSERT statement
  disabling, 175
  INSERT SELECT, 180–183
  INSERT triggers, 251–252
  inserting complete rows, 176–179
  inserting multiple rows, 179–180
  inserting retrieved data, 180–183
  INTO keyword, 179
  omitting columns from, 179
  OUTPUT clause, 183–184
  overview, 175
  returning list of changes, 183–184
  Unicode text, 284–285
  VALUES keyword, 177–179
INSERT triggers, 251–252
inserting data
  complete rows, 176–179
  explained, 175
  multiple rows, 179–180
retrieved data, 180–183
returning, 183–184
installing SQL Server, 18
INSTEAD OF triggers, 249
integrity, referential, 131
intelligent stored procedures, 235–238
IS NULL operator, 51, 54–55
ISABOUT() function, 173
ISJSON() function, 275

J

JavaScript Object Notation. See
JSON (JavaScript Object Notation)
joins
  advantages of, 131
  with aggregate functions, 148–150
  Cartesian products, 133–137
creating, 132–133
cross joins, 136
definition of, 129
equijoins, 137
guidelines, 150
inner joins, 137
multiple table joins, 138–140
natural joins, 145
outer joins, 145–148
performance considerations, 139
referential integrity, 131
self joins, 142–144
simplifying with views, 206–207
WHERE clause, 133–137
JSON (JavaScript Object Notation)
functions, 275
retrieving data as JSON, 273–274
FOR JSON AUTO clause, 273
JSON PATH clause, 274
JSON_MODIFY() function, 275
JSON_QUERY() function, 275
JSON_VALUE() function, 275

K

keys
  foreign keys, 11, 130
primary keys
    creating, 194–195
definition of, 9–11, 130
    and NULL values, 195
FROM keyword, 31–33
AS keyword, 78
IN keyword, 155
INTO keyword, 179
keywords. See also clauses;
statements
  ABSOLUTE, 243
  ALL, 105
  AS, 78
  ASC, 46–47
  AUTO, 266
  COLLATE, 281–283
  CONTAINS, 168–172
  DEFAULT, 197–198
  DESC, 44–46
  DISTINCT, 36–37, 105–106
ELEMENTS, 266–267
EXISTS, 126–128
EXPLICIT, 267–268
FIRST, 243
FREETEXT, 166–168
FROM
  individual columns, retrieving, 31–33
  multiple columns, retrieving, 33–34
IDENTITY, 195–197
IN, 155
INTO, 179
LAST, 243
LEFT, 146–147
NEAR, 170–171
NEXT, 243
NULL
  aggregate functions and, 100–103
  checking for, 54–55
  CREATE TABLE statement, 193–194
  null value elimination warning, 150
  UPDATE statement, 187
  wildcards and, 68
OUTPUT
  DELETE statement, 188–189
  INSERT statement, 183–184
  UPDATE statement, 188–189
PERCENT, 38–39
PRIOR, 243
RAW, 266–267
RIGHT, 146–147
TOP, 37–39
UNION, 154–157
VALUES, 177–179

LAST keyword, 243
Left() function, 85
LEFT keyword, 146–147
left outer joins, 147
Len() function, 85
less than (<) operator, 51, 52
less than or equal to (<=) operator, 51
LIKE operator, 65–66
limiting results, 37–39
listing
columns, 27–28
databases, 26
inserted data, 183–184	
tables, 26–27
triggers, 250
local servers, 17
localization
  character sets
definition of, 277
  Unicode, 283–285
collations
  case sensitivity, 281–283
collation sequences, 278–280
definition of, 277
  importance of, 278
encodings, 277
locating SQL Server Management Studio, 18
logging in, 19

logins
  dropping, 290
  renaming, 291
  sa (system administrator), 288

loops, WHILE, 225–226

Lower() function, 85

lowercase, converting strings to, 85

LTrim() function, 77, 85

M

maintaining referential integrity, 131

Management Studio. See SQL Server Management Studio

managing
  full-text catalogs, 165–166
  full-text indexes, 165–166
  users, 289

master databases, 25

mathematical calculations, 79–80

mathematical operators, 80

Max() function
  DISTINCT keyword, 106
  explained, 101–102
  order numbers, determining, 196
  maximum value, returning, 101–102

Microsoft SQL Server Management Studio. See SQL Server Management Studio

Min() function, 102–103

minimum value, returning, 102–103

modify function, 271

Month() function, 87, 92–93

multiple columns
  retrieving, 33–34
  sorting by, 43–44

multiple rows, inserting, 179–180

multiple tables, joining, 138–140

multiple WHERE clauses
  AND operator, 57–58
  IN operator, 61–63
  NOT operator, 63–64
  OR operator, 58–59
  order of evaluation, 59–61

multiplication (*) operator, 80

multistatement triggers, 252

N

N prefix, 285

names
  alias names, 107
  fully qualified column names, 133
  savepoint names, 260

natural joins, 145

navigating SQL Server Management Studio, 20–22

NCHAR datatype, 283

NEAR operator, 170–171

New Database dialog, 23

New Query button, 20–21

NEXT keyword, 243

nodes function, 271

non-ANSI outer joins, 147–148

nonequality (< >) operator, 51, 53

nonequality (! =) operator, 51, 53

nonmatches, checking for, 53

nonnumeric data, aggregate functions and
  Max(), 102
  Min(), 103

not greater than (! >) operator, 51
overriding IDENTITY keyword

not less than (!<) operator, 51
NOT operator, 63–64
NTEXT datatype, 283
null value elimination warning, 150
NULL values
aggregate functions and
\[ Avg() \], 100
\[ Count() \], 101
\[ Max() \], 102
\[ Min() \], 103
\[ Sum() \], 104
checking for, 54–55
compared to empty strings, 194
CREATE TABLE statement, 193–194
null value elimination warning, 150
primary keys and, 195
UPDATE statement, 187
wildcards and, 68
numeric manipulation functions, 94
numeric values, converting to strings, 85
NVARCHAR datatype, 283

Object Explorer, 21
obtaining SQL Server, 17
omitting columns from INSERT operations, 179
OPEN statement, 241–242
opening cursors, 241–242
operators. See also keywords
AND, 57–58, 170
comparison operators

BETWEEN operator, 54
equality operator (=), 51
IS NULL operator, 54–55
less than operator (<), 52
nonequality operator (<>), 53
nonequality operator (!=), 53
table of, 49–50
concatenation (+) operator, 74–77
CONTAINS, 168–172
FREETEXT, 166–168
IN, 61–63
LIKE, 65–66
mathematical operators, 80
NEAR, 170–171
NOT, 63–64
OR, 58–59, 222–223
order of evaluation, 59–61
optimizing performance, 295–297
order of clauses, 117
of evaluation, 59–61, 80
ORDER BY clause
example, 115–116
sorting by multiple columns, 43–44
sorting by single column, 41–43
OR operator, 58–59, 222–223
OUTER JOIN clause, 146–147
outer joins, 145–148
OUTPUT clause
DELETE statement, 188–189
INSERT statement, 183–184
UPDATE statement, 188–189
overriding IDENTITY keyword, 196
parameters of stored procedures, 232–235
parentheses, 61
PERCENT keyword, 38–39
percent sign (%)
    modulo operator, 80
    wildcard character, 66–68
performance
    full-text searching, 161
    improving, 295–297
pi, returning, 94
Pi() function, 94
plus sign (+)
    addition operator, 80
    concatenation operator, 74–77
populate.sql file, 22
portability, 83
precedence, order of, 80
predicates. See also keywords; operators
    AND, 170
    CONTAINS, 168–172
definition of, 66
    FREETEXT, 166–168
    IN, 61–63
    LIKE, 65–66
    NEAR, 170–171
    NOT, 63–64
    OR, 58–59, 222–223
primary keys
    creating, 194–195
definition of, 9–11, 130
    and NULL values, 195
PRINT statement, 217
PRIOR keyword, 243
procedures. See stored procedures
processing
    conditional processing
        BEGIN statement, 223–224
        END statement, 223–224
        IF statement, 221–223
        WHILE statement, 225–226
transaction processing
    autocommits, 259, 261
    COMMIT statement, 259
    explained, 255–257
    ROLLBACK statement, 258–259
    savepoints, 260–261
programming
    comments, 220
conditional processing
    BEGIN statement, 223–224
    END statement, 223–224
    IF statement, 221–223
    WHILE statement, 225–226
explained, 213
variables
    @ (at symbol), 214
    assigning values to, 215–216
    converting to strings, 217
    declaring, 214–215
definition of, 214
    rules and restrictions, 214
    in statements, 218–220
    viewing contents of, 216–217
Properties panel, 21
queries

aggregate functions
- $\text{Avg}()$, 98–99
- combining, 106–107
- $\text{Count}()$, 100–101, 106
- distinct values, 105
- explained, 97–98
- with joins, 148–150
- $\text{Max}()$, 101–102, 106
- $\text{Min}()$, 102–103
- $\text{Sum}()$, 103–104
AS keyword, 78
building incrementally, 128
combined queries
- creating, 154–157
- duplicate rows, including/eliminating, 157–158
- explained, 153
- sorting results of, 158–159
concatenation (+) operator, 74–77
cursors
- closing, 241–242
- creating, 240–241
- deleting, 240–241
- explained, 239–240
- fetched data, 242–245
- opening, 241–242
formatting, 122
GROUP BY clause, 110–111, 115–116
HAVING clause, 111–114
joins
- advantages of, 131
- with aggregate functions, 148–150
Cartesian products, 133–137
creating, 132–133
cross joins, 136
definition of, 129
equi joins, 137
guidelines, 150
inner joins, 137
multiple table joins, 138–140
natural joins, 145
outer joins, 145–148
performance considerations, 139
referential integrity, 131
self joins, 142–144
WHERE clause, 133–137
ORDER BY clause, 115–116
sequence of clauses, 117
subqueries
- as calculated fields, 123–126
- checking for existence with, 126–128
correlated subqueries, 125–126
definition of, 119
EXISTS keyword, 126–128
filtering by, 119–123
WHERE clause
- $\text{IN}$ operator, 61–63
- $\text{NOT}$ operator, 63–64
- $\text{OR}$ operator, 58–59
- order of evaluation, 59–61
wildcard filtering
- brackets ([ ]) wildcard, 69–70
case sensitivity, 67
LIKE operator, 65–66
NULL values, 68
percent sign (%) wildcard, 66–68
tips for, 71
trailing spaces, 68
underscore (_ ) wildcard, 68–69
query function, 271
quotation marks, 220
quotes, single (‘), 53, 169, 220

R
Rand() function, 94
range of values, checking for, 54
ranking search results, 172–173
RAW keyword, 266–267
referential integrity, maintaining, 131
reformatting retrieved data, 207–208
relational tables
aliases, 141–142
explained, 129–131
formatting, 130
joins
advantages of, 131
with aggregate functions, 148–150
Cartesian products, 133–137
creating, 132–133
cross joins, 136
definition of, 129
equijoins, 137
guidelines, 150
inner joins, 137
multiple table joins, 138–140
natural joins, 145
outer joins, 145–148
performance considerations, 139
referential integrity, 131
self joins, 142–144
WHERE clause, 133–137
primary keys, 130
removing
access rights, 292–293
views, 205
renaming
logins, 291
tables, 201
Replace() function, 85
reporting
on deleted data, 188–189
on inserted data, 183–184
on updated data, 188–189
restrictions (views), 205
results, limiting, 37–39
Results panel, 21
retrieved data
inserting, 180–183
reformatting, 207–208
retrieving data
columns
all columns, 35–36
individual columns, 31–33
multiple columns, 33–34
JSON (JavaScript Object Notation) data, 273–274
overview, 31
rows, 36–37
XML (Extensible Markup Language) data, 264–268
reusable views, 207
REVOKE statement, 292–293
revoking access rights, 292–293
Right() function, 85
RIGHT keyword, 146–147
right outer joins, 147
ROLLBACK statement, 258–259
rolling back statements, 258–259
Round() function, 94
rounding numbers, 94
rows
counting, 100–101
deleting
  deleted data, reporting on, 188–189
guidelines, 188–189
syntax, 187–188
duplicate rows in unions, including/eliminating, 157–158
explained, 9–10
inserting
  complete rows, 176–179
  list of changes, returning, 183–184
  multiple rows, 179–180
  retrieved data, 180–183
retrieving, 36–37
updating
guidelines, 188–189
syntax, 185–187
updated data, reporting on, 188–189
RTrim() function, 76–77, 85
rules (views), 205
running stored procedures, 230
savepoints, 260–261
scaling databases, 131
schemas (XML), 271
search criteria
  comparison operators, 49–50
  multiple WHERE clauses
    AND operator, 57–58
    IN operator, 61–63
    NOT operator, 63–64
    OR operator, 58–59
    order of evaluation, 59–61
  nonmatches, 53
  NULL values, 54–55
  range of values, 54
  single values, 51–52
WHERE clause, 49–50
wildcard filtering
  brackets ([]) wildcard, 69–70
  case sensitivity, 67
  LIKE operator, 65–66
  NULL values, 68
  percent sign (%) wildcard, 66–68
  tips for, 71
  underscore (_) wildcard, 68–69
search results, ranking, 172–173
search term weights, assigning, 173
searching
  full-text searching
    enabling, 162–163
    explained, 161–162
    full-text catalogs, 163–166
    full-text indexes, 163–166
    performance, 161
    performing with
      CONTAINS, 168–172
sa (system administrator) login, 288
SAVE TRANSACTION statement, 260–261
searching

performing with FREET-EXT, 166–168

search results, ranking, 172–173

search term weights, assigning, 173

SELECT

aggregate functions, 97–107

ASC keyword, 46–47

AS keyword, 78

asterisk (*) wildcard, 35–36

clause ordering, 117

comparison operators, 49–50

concatenation (+) operator, 74–77

DESC keyword, 44–46

DISTINCT keyword, 36–37

EXISTS keyword, 126–128

formatting, 122

FROM keyword, 31–34

fully qualified table names, 39–40

GROUP BY clause, 110–111, 115–116

HAVING clause, 111–114

LIKE operator, 65–66

multiple columns, retrieving, 33–34

ORDER BY clause, 41–44, 115–116

overview, 31

PERCENT keyword, 38–39

TOP keyword, 37–39

WHERE clause. See WHERE clause

distinct values, 105

subqueries

as calculated fields, 123–126

checking for existance with, 126–128

correlated subqueries, 125–126

definition of, 119

filtering by, 119–123

wildcard filtering

* (asterisk), 61–62

brackets ([ ] ) wildcard, 69–70

case sensitivity, 67

LIKE operator, 65–66

percent sign (%) wildcard, 66–68

tips for, 71

trailing spaces, 68

underscore (_ ) wildcard, 68–69

for XML (Extensible Markup Language) data, 271–272

security

access rights, 291–293

explained, 287–288

logins, renaming, 291

user accounts

creating, 290

deleting, 290

enabling/disabling, 290

user management, 289

SELECT statement

aggregate functions

Avg(), 98–99

combining, 106–107

Count(), 100–101, 106

distinct values, 105
correlated subqueries, 125–126
definition of, 119
filtering by, 119–123
TOP keyword, 37–39
UNION keyword, 154–157
variables in, 219–220
views
advantages of, 204–205
with calculated fields, 210–211
creating, 205
explained, 203–204
filtering data with, 209
reformatting retrieved data with, 207–208
removing, 205
reusable views, 207
rules and restrictions, 205
simplifying joins with, 206–207
updating, 205, 211–212
WHERE clause
AND operator, 57–58
comparison operators. See comparison operators
IN operator, 61–63
joins and, 133–137
nonmatches, checking for, 53
NOT operator, 63–64
NULL values, checking for, 54–55
OR operator, 58–59
order of evaluation, 59–61
range of values, checking for, 54
single values, checking against, 51–52
syntax, 49–50
wildcard filtering
brackets ([ ]) wildcard, 69–70
case sensitivity, 67
LIKE operator, 65–66
NULL values, 68
percent sign (%) wildcard, 66–68
tips for, 71
trailing spaces, 68
underscore (_) wildcard, 68–69
selecting databases, 24–25
self joins, 142–144
semicolon (;), 33, 179–180
sequences
of clauses, 117
collation sequences, 278–280
of evaluation, 59–61, 80
ServerProperty() function, 279
SET IDENTITY_INSERT statement, 177
SET statement, 186, 215–216
setting up
access rights, 291–292
full-text searching
full-text catalogs, 163–166
full-text indexes, 164–166
full-text search support, 162–163
Sin() function, 94
sines, returning, 94
single quotes (’), 53, 169, 220
single values, checking against, 51–52
sorting combined query results, 158–159
sorting data
  in ascending order, 46–47
  case sensitivity and, 46
  in descending order, 44–46
  dictionary sort order, 46
  grouping versus sorting, 114–116
  by multiple columns, 43–44
  by single column, 41–43
Soundex() function, 85–86
sp_columns stored procedure, 27–28
sp_databases stored procedure, 26
sp_fulltext_database stored
  procedure, 162–163
sp_helplogs stored procedure, 28, 289
sp_helplogins stored procedure, 28, 289
SP_HELPTRIGGER stored
  procedure, 250
sp_helpuser stored procedure, 28
sp_rename stored procedure, 201
sp_server_info stored procedure, 28
sp_spaceused stored procedure, 28
sp_statistics stored procedure, 28
sp_tables stored procedure, 26–27
spaces
  trailing spaces, 68
  trimming, 76–77
SQL (Structured Query Language)
  advantages of, 11
  definition of, 11
SQL Server
  advantages of, 13
  client utilities, 15–16, 18
  connecting to, 19–20
  Database Engine Tuning
    Advisor, 296–297
documentation, 297
installing, 18
JSON (JavaScript Object
  Notation) data
    functions, 275
    retrieving data as JSON, 273–274
obtaining, 17
server access, 16–17
server software, 14–15
SQL Server Express, 17
SQL Server Management Studio
  locating, 18
  navigating, 20–22
  overview, 15–16
versions, 15
XML (Extensible Markup
  Language) data
    retrieving, 264–268
    searching for, 271–272
    storing, 268–270
    support for, 263
SQL Server Express, 17
SQL Server Management Studio
  locating, 18
  navigating, 20–22
  overview, 15–16
Sqrt() function, 94
square brackets ([ ]), 275
Square() function, 94
square roots, returning, 94
squares, returning, 94
statements
ALTER FULLTEXT, 165–166
ALTER LOGIN, 290–291
ALTER TABLE, 199–200, 280, 291
ALTER TRIGGER, 249
autocommits, 259, 261
batches, 219
BEGIN, 223–224
BREAK, 226
case conventions, 33
CLOSE, 241–242
COMMIT, 259
CONTINUE, 226
CREATE FULLTEXT CATALOG, 163
CREATE FULLTEXT INDEX, 164
CREATE LOGIN, 290
CREATE PROCEDURE, 230–231
CREATE TABLE
  collation sequences, 280
  DEFAULT keyword, 197–198
default values, 197–198
  example, 269
  IDENTITY keyword, 195–197
  NULL values, 193–194
  primary keys, 194–195
  syntax, 192–193
CREATE TRIGGER, 248
CREATE VIEW, 205
DEALLOCATE, 240–241
DELETE
  DELETE FROM, 188
  DELETE triggers, 252
deleted data, reporting on, 188–189
guidelines, 188–189
syntax, 187–190
DISABLE TRIGGER, 250
DROP FULLTEXT, 165–166
DROP LOGIN, 290
DROP PROCEDURE, 231–232
DROP TABLE, 201
DROP TRIGGER, 249
DROP VIEW, 205
ENABLE TRIGGER, 250
END, 223–224
EXEC, 230
EXECUTE, 230
explicit commits, 259
FETCH, 242–245
formatting, 193
GRANT, 291–292
grouping, 223–224
IF, 221–223
INSERT
  disabling, 175
  INSERT SELECT, 180–183
  INSERT triggers, 251–252
  inserting complete rows, 176–179
  inserting multiple rows, 179–180
  inserting retrieved data, 180–183
  INTO keyword, 179
  omitting columns from, 179
  OUTPUT clause, 183–184
  overview, 175
  returning list of changes, 183–184
  Unicode text, 284–285
  VALUES keyword, 177–179
OPEN, 241–242
PRINT, 217
REVOKE, 292–293
ROLLBACK, 258–259
rolling back, 258–259
SAVE TRANSACTION, 260–261
SELECT
aggregate functions, 97–107
ASC keyword, 46–47
AS keyword, 78
asterisk (*) wildcard, 35–36
batch processing, 218–219
clause ordering, 117
COLLATE keyword, 281–283
combined queries, 153–159
comparison operators, 49–50
concatenation (+) operator, 74–77
DESC keyword, 44–46
DISTINCT keyword, 36–37
EXISTS keyword, 126–128
FOR JSON AUTO clause, 273, 274
formatting, 122
FOR XML clause, 264–268
FROM clause, 133
FROM keyword, 31–34
FULL OUTER JOIN clause, 148
fully qualified table names, 39–40
GROUP BY clause, 110–111, 115–116
HAVING clause, 111–114
INNER JOIN clause, 137–138
INSERT SELECT, 180–183
joins. See joins
LIKE operator, 65–66
multiple columns, retrieving, 33–34
ORDER BY clause, 41–44, 115–116
OUTER JOIN clause, 146–147
overview, 31
PERCENT keyword, 38–39
retrieving variable values with, 216
TOP keyword, 37–39
UNION keyword, 154–157
variables in, 219–220
views, 203–212
WHERE clause. See WHERE clause
wildcard filtering, 65–71
SET, 186, 215–216
SET IDENTITY_INSERT, 177
subqueries
as calculated fields, 123–126
checking for existence with, 126–128
correlated subqueries, 125–126
declaration of, 119
filtering by, 119–123
triggers
AFTER triggers, 249
creating, 248
DELETE triggers, 252
disabling, 249–250
dropping, 249  
enabling, 249–250  
explained, 247  
guidelines, 253–254  
INSERT triggers, 251–252  
INSTEAD OF triggers, 249  
multistatement triggers, 252  
trigger assignments, determining, 250  
UPDATE triggers, 253  
updating, 249

UPDATE  
guidelines, 188–189  
syntax, 185–187  
UPDATE triggers, 253  
updated data, reporting on, 188–189

USE, 24–25  
variables in, 218–220  
WHILE, 225–226  
white space in, 33

stored procedures  
advantages of, 228–229  
creating, 230–231  
definition of, 25  
disadvantages of, 229  
dropping, 231–232  
executing, 230  
explained, 227–228  
intelligent stored procedures, 235–238  
parameters, 232–235  
sp_columns, 27–28  
sp_databases, 26  
sp_fulltext_database, 162–163  
sp_helplogsins, 28, 289  
SP_HELPTRIGGER, 250  
sp_helpuser, 28  
sp_rename, 201  
sp_server_info, 28  
sp_spaceused, 28  
sp_statistics, 28  
sp_tables, 26–27

storing XML (Extensible Markup Language) data, 268–270  
Str() function, 85  
strings  
converting variables to, 217  
empty strings, 194  
text manipulation functions, 84–86

Structured Query Language. See SQL (Structured Query Language)

Subqueries  
as calculated fields, 123–126  
checking for existence with, 126–128  
correlated subqueries, 125–126  
definition of, 119  
EXISTS keyword, 126–128  
filtering by, 119–123

SubString() function, 85  
subtraction (-) operator, 80  
Sum() function, 103–104  
summarizing data  
Avg() function, 98–99  
Count() function  
DISTINCT keyword, 106  
explained, 100–101  
explained, 97–98  
Max() function  
DISTINCT keyword, 106  
explained, 101–102  
Min() function, 102–103  
Sum() function, 103–104
at symbol (@), 214, 233
SYS database, 289
system administrator (sa) login, 288
system administrators, 19

T

tables
  aliases
    explained, 141–142
    naming, 107
  columns
    aliases, 77–78
    average value of, returning, 98–99
    deleting, 187–190
    explained, 7–8
    fully qualified column names, 133
    identity columns, 28
    maximum value, returning, 101–102
    minimum value, returning, 102–103
    omitting from INSERT operations, 179
    returning list of, 27–28
    sorting by, 41–44
    totals, returning, 103–104
    updating, 185–187
  creating
    basic table creation, 192–193
    DEFAULT keyword, 197–198
    default values, 197–198
    explained, 191
    IDENTITY keyword, 195–197

  NULL values, 193–194
  primary keys, 194–195
data. See data
datatypes, 7–8
definition of, 6–7
deleting, 201
different tables, combining, 159
dropping, 201
foreign keys, 11, 130
fully qualified table names, 39–40
joins
  advantages of, 131
  with aggregate functions, 148–150
  Cartesian products, 133–137
  creating, 132–133
  cross joins, 136
  definition of, 129
  equijoins, 137
  guidelines, 150
  inner joins, 137
  multiple table joins, 138–140
  natural joins, 145
  outer joins, 145–148
  performance considerations, 139
  referential integrity, 131
  self joins, 142–144
  simplifying with views, 206–207
  WHERE clause, 133–137
primary keys
  creating, 194–195
  definition of, 9–11, 130
  and NULL values, 195
relational tables, 129–131
renaming, 201
returning list of, 26–27
rows
  counting, 100–101
deleting, 187–188
duplicate rows in unions,
  including/eliminating,
  157–158
explained, 9–10
inserting, 176–183
updating, 185–187
sample tables, creating, 22–23
  updating, 199–200
views
  advantages of, 204–205
  with calculated fields,
  210–211
creating, 205
explained, 203–204
filtering data with, 209
reformatting retrieved data
  with, 207–208
removing, 205
reusable views, 207
rules and restrictions, 205
simplifying joins with,
  206–207
updating, 205, 211–212
Tan() function, 94
tangents, returning, 94
testing calculations, 80
text
  aggregate functions and
    Max(), 102
    Min(), 103
  full-text searching
    enabling, 162–163
    explained, 161–162
full-text catalogs, 163–166
full-text indexes, 164–166
performance, 161
performing with
  CONTAINS, 168–172
  FREETEXT, 166–168
search results, ranking,
  172–173
text manipulation functions, 84
time manipulation functions, 87–94
TOP keyword, 37–39
totals (column), returning, 103–104
trailing spaces, 68
transaction processing
  autocommits, 259, 261
  COMMIT statement, 259
  explained, 255–257
  ROLLBACK statement, 258–259
  savepoints, 260–261
triggers
  AFTER triggers, 249
  creating, 248
  DELETE triggers, 252
disabling, 249–250
dropping, 249
enabling, 249–250
explained, 247
guidelines, 253–254
INSERT triggers, 251–252
INSTEAD OF triggers, 249
multistatement triggers, 252
trigger assignments,
determining, 250
UPDATE triggers, 253
updating, 249
trigonometric functions, 94
trimming spaces, 76–77
tuning performance, 295–297

U
underscore (_), 68–69
undoing statements, 258–259
Unicode, 283–285
UNION keyword, 154–157
UPDATE statement
guidelines, 188–189
syntax, 185–187
UPDATE triggers, 253
updated data, reporting on, 188–189
UPDATE triggers, 253
updated data, reporting on, 188–189
updating
data
  guidelines, 188–189
  syntax, 185–187
  updated data, reporting on, 188–189
indexes, 165
tables, 199–200
triggers, 249
views, 205, 211–212
Upper() function, 84–85
uppercase, converting strings to, 85
USE statement, 24–25
user accounts
  creating, 290
  deleting, 290
  enabling/disabling, 290
users
  access rights, 291–293
  logins, renaming, 291
managing, 289
user accounts
  creating, 290
  deleting, 290
  enabling/disabling, 290
utilities, Database Engine Tuning Advisor, 296–297

V
value function, 271
values
  assigning to variables, 215–216
NULL values
  aggregate functions and.
  See aggregate functions
  checking for, 54–55
  compared to empty strings, 194
  CREATE TABLE statement, 193–194
  null value elimination
  warning, 150
  primary keys and, 195
  wildcards and, 68, 187
VALUES keyword, 177–179
variables
  @ (at symbol), 214
  assigning values to, 215–216
  converting to strings, 217
  declaring, 214–215
  definition of, 214
  rules and restrictions, 214
  in statements, 218–220
  viewing contents of, 216–217
versions of SQL Server, 15
viewing variable contents, 216–217
views

advantages of, 204–205
with calculated fields, 210–211
creating, 205
dropping, 205
explained, 203–204
filtering data with, 209
reformatting retrieved data with, 207–208
removing, 205
reusable views, 207
rules and restrictions, 205
simplifying joins with, 206–207
updating, 205, 211–212

W

warnings, null value elimination warning, 150
weight values (search), 173
WHERE clause
AND operator, 57–58
brackets ([ ]) wildcard, 69–70
comparison operators
  BETWEEN operator, 54
  equality operator (=), 51
  IS NULL operator, 54–55
  less than operator (<), 52
  nonequality operator (<>,), 53
  nonequality operator (<>), 53
  table of, 49–50
IN operator, 61–63
joins, 133–137
LIKE operator, 65–66
nonmatches, checking for, 53
NOT operator, 63–64
NULL values, checking for, 54–55
OR operator, 58–59
order of evaluation, 59–61
percent sign (%) wildcard, 66–68
range of values, checking for, 54
single values, checking against, 51–52
subqueries
  as calculated fields, 123–126
  checking for existence with, 126–128
correlated subqueries, 125–126
definition of, 119
EXISTS keyword, 126–128
  filtering by, 119–123
syntax, 49–50
underscore (_ ) wildcard, 68–69
views, 209
WHILE statement, 225–226
white space, 33
wildcard filtering
  * (asterisk), 61–62
  brackets ([ ]) wildcard, 69–70
case sensitivity, 67
LIKE operator, 65–66
percent sign (%) wildcard, 66–68
tips for, 71
trailing spaces, 68
underscore (_ ) wildcard, 68–69
### X-Y-Z

<table>
<thead>
<tr>
<th>XML (Extensible Markup Language)</th>
<th>Year() function, 87, 92–93</th>
</tr>
</thead>
<tbody>
<tr>
<td>data</td>
<td></td>
</tr>
<tr>
<td>storing, 268–270</td>
<td></td>
</tr>
<tr>
<td>support for, 263</td>
<td></td>
</tr>
<tr>
<td>XML schemas, 271</td>
<td></td>
</tr>
<tr>
<td>XPath, 268, 272</td>
<td></td>
</tr>
<tr>
<td>XQuery, 269, 272</td>
<td></td>
</tr>
<tr>
<td>retrieving, 264–268</td>
<td></td>
</tr>
<tr>
<td>searching for, 271–272</td>
<td></td>
</tr>
</tbody>
</table>