

LEARNING MADE EASY



3rd Edition

SQL

ALL-IN-ONE

for
dummies[®]
A Wiley Brand



8
Books
in one!

Allen G. Taylor

Author of all editions of
SQL For Dummies

SQL

ALL-IN-ONE

for
dummies[®]
A Wiley Brand



SQL

ALL-IN-ONE

3rd Edition

by Allen G. Taylor

for
dummies[®]
A Wiley Brand

SQL All-In-One For Dummies®, 3rd Edition

Published by: **John Wiley & Sons, Inc.**, 111 River Street, Hoboken, NJ 07030-5774, www.wiley.com

Copyright © 2019 by John Wiley & Sons, Inc., Hoboken, New Jersey

Published simultaneously in Canada

No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, scanning or otherwise, except as permitted under Sections 107 or 108 of the 1976 United States Copyright Act, without the prior written permission of the Publisher. Requests to the Publisher for permission should be addressed to the Permissions Department, John Wiley & Sons, Inc., 111 River Street, Hoboken, NJ 07030, (201) 748-6011, fax (201) 748-6008, or online at <http://www.wiley.com/go/permissions>.

Trademarks: Wiley, For Dummies, the Dummies Man logo, Dummies.com, Making Everything Easier, and related trade dress are trademarks or registered trademarks of John Wiley & Sons, Inc. and may not be used without written permission. All other trademarks are the property of their respective owners. John Wiley & Sons, Inc. is not associated with any product or vendor mentioned in this book.

LIMIT OF LIABILITY/DISCLAIMER OF WARRANTY: THE PUBLISHER AND THE AUTHOR MAKE NO REPRESENTATIONS OR WARRANTIES WITH RESPECT TO THE ACCURACY OR COMPLETENESS OF THE CONTENTS OF THIS WORK AND SPECIFICALLY DISCLAIM ALL WARRANTIES, INCLUDING WITHOUT LIMITATION WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE. NO WARRANTY MAY BE CREATED OR EXTENDED BY SALES OR PROMOTIONAL MATERIALS. THE ADVICE AND STRATEGIES CONTAINED HEREIN MAY NOT BE SUITABLE FOR EVERY SITUATION. THIS WORK IS SOLD WITH THE UNDERSTANDING THAT THE PUBLISHER IS NOT ENGAGED IN RENDERING LEGAL, ACCOUNTING, OR OTHER PROFESSIONAL SERVICES. IF PROFESSIONAL ASSISTANCE IS REQUIRED, THE SERVICES OF A COMPETENT PROFESSIONAL PERSON SHOULD BE SOUGHT. NEITHER THE PUBLISHER NOR THE AUTHOR SHALL BE LIABLE FOR DAMAGES ARISING HEREFROM. THE FACT THAT AN ORGANIZATION OR WEBSITE IS REFERRED TO IN THIS WORK AS A CITATION AND/OR A POTENTIAL SOURCE OF FURTHER INFORMATION DOES NOT MEAN THAT THE AUTHOR OR THE PUBLISHER ENDORSES THE INFORMATION THE ORGANIZATION OR WEBSITE MAY PROVIDE OR RECOMMENDATIONS IT MAY MAKE. FURTHER, READERS SHOULD BE AWARE THAT INTERNET WEBSITES LISTED IN THIS WORK MAY HAVE CHANGED OR DISAPPEARED BETWEEN WHEN THIS WORK WAS WRITTEN AND WHEN IT IS READ.

For general information on our other products and services, please contact our Customer Care Department within the U.S. at 877-762-2974, outside the U.S. at 317-572-3993, or fax 317-572-4002. For technical support, please visit <https://hub.wiley.com/community/support/dummies>.

Wiley publishes in a variety of print and electronic formats and by print-on-demand. Some material included with standard print versions of this book may not be included in e-books or in print-on-demand. If this book refers to media such as a CD or DVD that is not included in the version you purchased, you may download this material at <http://booksupport.wiley.com>. For more information about Wiley products, visit www.wiley.com.

Library of Congress Control Number: 2019934589

ISBN 978-1-119-56961-9 (pbk); ISBN 978-1-119-56960-2 (ebk); ISBN 978-1-119-56959-6 (ebk)

Manufactured in the United States of America

10 9 8 7 6 5 4 3 2 1

Contents at a Glance

Introduction	1
Book 1: SQL Concepts	9
CHAPTER 1: Understanding Relational Databases	11
CHAPTER 2: Modeling a System	31
CHAPTER 3: Getting to Know SQL	55
CHAPTER 4: SQL and the Relational Model	67
CHAPTER 5: Knowing the Major Components of SQL	77
CHAPTER 6: Drilling Down to the SQL Nitty-Gritty	99
Book 2: Relational Database Development	131
CHAPTER 1: System Development Overview	133
CHAPTER 2: Building a Database Model	149
CHAPTER 3: Balancing Performance and Correctness	167
CHAPTER 4: Creating a Database with SQL	199
Book 3: SQL Queries	211
CHAPTER 1: Values, Variables, Functions, and Expressions	213
CHAPTER 2: SELECT Statements and Modifying Clauses	239
CHAPTER 3: Querying Multiple Tables with Subqueries	281
CHAPTER 4: Querying Multiple Tables with Relational Operators	309
CHAPTER 5: Cursors	329
Book 4: Data Security	341
CHAPTER 1: Protecting Against Hardware Failure and External Threats	343
CHAPTER 2: Protecting Against User Errors and Conflicts	373
CHAPTER 3: Assigning Access Privileges	401
CHAPTER 4: Error Handling	413
Book 5: SQL and Programming	429
CHAPTER 1: Database Development Environments	431
CHAPTER 2: Interfacing SQL to a Procedural Language	437
CHAPTER 3: Using SQL in an Application Program	443
CHAPTER 4: Designing a Sample Application	457
CHAPTER 5: Building an Application	477
CHAPTER 6: Understanding SQL's Procedural Capabilities	493
CHAPTER 7: Connecting SQL to a Remote Database	509

Book 6: SQL, XML, and JSON	523
CHAPTER 1: Using XML with SQL	525
CHAPTER 2: Storing XML Data in SQL Tables	553
CHAPTER 3: Retrieving Data from XML Documents	577
CHAPTER 4: Using JSON with SQL	595
Book 7: Database Tuning Overview	609
CHAPTER 1: Tuning the Database	611
CHAPTER 2: Tuning the Environment	623
CHAPTER 3: Finding and Eliminating Bottlenecks	645
Book 8: Appendices	675
APPENDIX A: SQL: 2016 Reserved Words	677
APPENDIX B: Glossary	683
Index	691

Table of Contents

INTRODUCTION	1
About This Book	1
Foolish Assumptions	2
Conventions Used in This Book	3
What You Don't Have to Read	3
How This Book Is Organized	3
Book 1: SQL Concepts	3
Book 2: Relational Database Development	4
Book 3: SQL Queries	4
Book 4: Data Security	4
Book 5: SQL and Programming	5
Book 6: SQL and XML	5
Book 7: Database Tuning Overview	5
Book 8: Appendices	5
Icons Used in This Book	6
Where to Go from Here	6
BOOK 1: SQL CONCEPTS	9
CHAPTER 1: Understanding Relational Databases	11
Understanding Why Today's Databases Are Better than Early Databases	12
Irreducible complexity	12
Managing data with complicated programs	12
Managing data with simple programs	15
Which type of organization is better?	15
Databases, Queries, and Database Applications	16
Making data useful	16
Retrieving the data you want — and only the data you want.	16
Examining Competing Database Models	18
Looking at the historical background of the competing models	18
The hierarchical database model	19
The network database model	23
The relational database model	23
The object-oriented database model	28
The object-relational database model	29
The nonrelational NoSQL model	29
Why the Relational Model Won	29

CHAPTER 2: Modeling a System	31
Capturing the Users' Data Model	31
Identifying and interviewing stakeholders	32
Reconciling conflicting requirements	33
Obtaining stakeholder buy-in	33
Translating the Users' Data Model to a Formal Entity-Relationship Model	34
Entity-Relationship modeling techniques	35
Drawing Entity-Relationship diagrams	40
Understanding advanced ER model concepts	43
A simple example of an ER model	47
A slightly more complex example	48
Problems with complex relationships	52
Simplifying relationships using normalization	53
Translating an ER model into a relational model	53
CHAPTER 3: Getting to Know SQL	55
Where SQL Came From	55
Knowing What SQL Does	56
The ISO/IEC SQL Standard	57
Knowing What SQL Does Not Do	57
Choosing and Using an Available DBMS Implementation	58
Microsoft Access	59
Microsoft SQL Server	64
IBM DB2	64
Oracle Database	64
Sybase SQL Anywhere	65
MySQL	65
PostgreSQL	65
CHAPTER 4: SQL and the Relational Model	67
Sets, Relations, Multisets, and Tables	68
Functional Dependencies	69
Keys	70
Views	71
Users	72
Privileges	72
Schemas	73
Catalogs	74
Connections, Sessions, and Transactions	74
Routines	75
Paths	75

CHAPTER 5: Knowing the Major Components of SQL	77
Creating a Database with the Data Definition Language	77
The containment hierarchy	78
Creating tables	79
Specifying columns	79
Creating other objects	80
Modifying tables	87
Removing tables and other objects	87
Operating on Data with the Data Manipulation Language (DML)	88
Retrieving data from a database	88
Adding data to a table	89
Updating data in a table	92
Deleting data from a table	95
Updating views doesn't make sense	96
Maintaining Security in the Data Control Language (DCL)	97
Granting access privileges	97
Revoking access privileges	98
Preserving database integrity with transactions	98
CHAPTER 6: Drilling Down to the SQL Nitty-Gritty	99
Executing SQL Statements	99
Interactive SQL	100
Challenges to combining SQL with a host language	101
Embedded SQL	101
Module language	104
Using Reserved Words Correctly	105
SQL's Data Types	105
Exact numerics	106
Approximate numerics	108
Character strings	110
Binary strings	112
Booleans	113
Datetimes	113
Intervals	115
XML type	115
ROW type	116
Collection types	117
REF types	118
User-defined types	119
Data type summary	122
Handling Null Values	123
Applying Constraints	124
Column constraints	125
Table constraints	126
Foreign key constraints	128
Assertions	129

BOOK 2: RELATIONAL DATABASE DEVELOPMENT	131
CHAPTER 1: System Development Overview	133
The Components of a Database System	133
The database	134
The database engine	134
The DBMS front end	135
The database application	135
The user	136
The System Development Life Cycle	136
Definition phase	137
Requirements phase	138
Evaluation phase	140
Design phase	143
Implementation phase	145
Final Documentation and Testing phase	146
Maintenance phase	148
CHAPTER 2: Building a Database Model	149
Finding and Listening to Interested Parties	150
Your immediate supervisor	150
The users	151
The standards organization	151
Upper management	152
Building Consensus	152
Gauging what people want	153
Arriving at a consensus	154
Building a Relational Model	154
Reviewing the three database traditions	155
Knowing what a relation is	156
Functional dependencies	156
Keys	157
Being Aware of the Danger of Anomalies	157
Eliminating anomalies	159
Examining the higher normal forms	162
The Database Integrity versus Performance Tradeoff	164
CHAPTER 3: Balancing Performance and Correctness	167
Designing a Sample Database	168
The ER model for Honest Abe's	168
Converting an ER model into a relational model	170
Normalizing a relational model	170
Handling binary relationships	172
A sample conversion	177

Maintaining Integrity	179
Entity integrity	180
Domain integrity	181
Referential integrity	182
Avoiding Data Corruption	183
Speeding Data Retrievals	185
Hierarchical storage	185
Full table scans	186
Working with Indexes	187
Creating the right indexes	187
Indexes and the ANSI/ISO standard	188
Index costs	188
Query type dictates the best index	188
Data structures used for indexes	190
Indexes, sparse and dense	192
Index clustering	192
Composite indexes	192
Index effect on join performance	193
Table size as an indexing consideration	193
Indexes versus full table scans	194
Reading SQL Server Execution Plans	194
Robust execution plans	194
A sample database	195
CHAPTER 4: Creating a Database with SQL	199
First Things First: Planning Your Database	199
Building Tables	200
Locating table rows with keys	202
Using the CREATE TABLE statement	202
Setting Constraints	204
Column constraints	205
Table constraints	205
Keys and Indexes	205
Ensuring Data Validity with Domains	205
Establishing Relationships between Tables	206
Altering Table Structure	210
Deleting Tables	210
BOOK 3: SQL QUERIES	211
CHAPTER 1: Values, Variables, Functions, and Expressions	213
Entering Data Values	213
Row values have multiple parts	214
Identifying values in a column	214
Literal values don't change	214

Variables vary	214
Special variables hold specific values	216
Working with Functions	217
Summarizing data with set functions	217
Dissecting data with value functions	220
Using Expressions	229
Numeric value expressions	229
String value expressions	229
Datetime value expressions	230
Interval value expressions	231
Boolean value expressions	232
Array value expressions	232
Conditional value expressions	233
Converting data types with a CAST expression	236
Row value expressions	238
CHAPTER 2: SELECT Statements and Modifying Clauses	239
Finding Needles in Haystacks with the SELECT Statement	239
Modifying Clauses	240
FROM clauses	240
WHERE clauses	241
GROUP BY clauses	259
HAVING clauses	262
ORDER BY clauses	262
Tuning Queries	265
SELECT DISTINCT	265
Temporary tables	268
The ORDER BY clause	272
The HAVING clause	276
The OR logical connective	280
CHAPTER 3: Querying Multiple Tables with Subqueries	281
What Is a Subquery?	281
What Subqueries Do	282
Subqueries that return multiple values	282
Subqueries that return a single value	284
Quantified subqueries return a single value	287
Correlated subqueries	290
Using Subqueries in INSERT, DELETE, and UPDATE Statements	295
Tuning Considerations for Statements Containing Nested Queries	298
Tuning Correlated Subqueries	304

CHAPTER 4:	Querying Multiple Tables with Relational Operators	309
	UNION	310
	UNION ALL	312
	UNION CORRESPONDING	312
	INTERSECT	313
	EXCEPT	315
	JOINS	315
	Cartesian product or cross join	316
	Equi-join	318
	Natural join	320
	Condition join	320
	Column-name join	321
	Inner join	322
	Outer join	323
	ON versus WHERE	327
	Join Conditions and Clustering Indexes	327
CHAPTER 5:	Cursors	329
	Declaring a Cursor	330
	The query expression	331
	Ordering the query result set	331
	Updating table rows	333
	Sensitive versus insensitive cursors	333
	Scrolling a cursor	335
	Holding a cursor	335
	Declaring a result set cursor	335
	Opening a Cursor	336
	Operating on a Single Row	337
	FETCH syntax	338
	Absolute versus relative fetches	338
	Deleting a row	339
	Updating a row	339
	Closing a Cursor	340
	BOOK 4: DATA SECURITY	341
CHAPTER 1:	Protecting Against Hardware Failure and External Threats	343
	What Could Possibly Go Wrong?	344
	Equipment failure	344
	Platform instability	345

Database design flaws	346
Data-entry errors	346
Operator error	347
Taking Advantage of RAID	347
Striping	348
RAID levels	348
Backing Up Your System	351
Preparation for the worst	352
Full or incremental backup	352
Frequency	353
Backup maintenance	353
Coping with Internet Threats	354
Viruses	354
Trojan horses	356
Worms	356
Denial-of-service attacks	357
SQL injection attacks	357
Phishing scams	370
Zombie spambots	370
Installing Layers of Protection	371
Network-layer firewalls	371
Application-layer firewalls	371
Antivirus software	371
Vulnerabilities, exploits, and patches	372
Education	372
Alertness	372
CHAPTER 2: Protecting Against User Errors and Conflicts	373
Reducing Data-Entry Errors	374
Data types: The first line of defense	374
Constraints: The second line of defense	374
Sharp-eyed humans: The third line of defense	375
Coping with Errors in Database Design	375
Handling Programming Errors	376
Solving Concurrent-Operation Conflicts	376
Passing the ACID Test: Atomicity, Consistency, Isolation, and Durability	378
Operating with Transactions	379
Using the SET TRANSACTION statement	379
Starting a transaction	380
Committing a transaction	383
Rolling back a transaction	383
Implementing deferrable constraints	386

Getting Familiar with Locking	391
Two-phase locking	391
Granularity	392
Deadlock	392
Tuning Locks	393
Measuring performance with throughput	394
Eliminating unneeded locks	394
Shortening transactions	394
Weakening isolation levels (ver-r-ry carefully)	395
Controlling lock granularity	396
Scheduling DDL statements correctly	396
Partitioning insertions	396
Cooling hot spots	397
Tuning the deadlock interval	397
Enforcing Serializability with Timestamps	397
Tuning the Recovery System	400
CHAPTER 3: Assigning Access Privileges	401
Working with the SQL Data Control Language	401
Identifying Authorized Users	402
Understanding user identifiers	402
Getting familiar with roles	402
Classifying Users	404
Granting Privileges	404
Looking at data	405
Deleting data	406
Adding data	406
Changing data	406
Referencing data in another table	406
Using certain database facilities	408
Responding to an event	408
Defining new data types	409
Executing an SQL statement	409
Doing it all	409
Passing on the power	409
Revoking Privileges	410
Granting Roles	411
Revoking Roles	412
CHAPTER 4: Error Handling	413
Identifying Error Conditions	414
Getting to Know SQLSTATE	414
Handling Conditions	416
Handler declarations	417
Handler actions and handler effects	417
Conditions that aren't handled	419

Dealing with Execution Exceptions: The WHENEVER Clause	419
Getting More Information: The Diagnostics Area	420
The diagnostics header area	421
The diagnostics detail area	422
Examining an Example Constraint Violation	424
Adding Constraints to an Existing Table	426
Interpreting SQLSTATE Information	426
Handling Exceptions	427
BOOK 5: SQL AND PROGRAMMING	429
CHAPTER 1: Database Development Environments	431
Microsoft Access	431
The Jet engine	432
DAO	432
ADO	432
ODBC	433
OLE DB	433
Files with the .mdb extension	433
The Access Database Engine	433
Microsoft SQL Server	433
IBM Db2	434
Oracle 18c	434
SQL Anywhere	435
PostgreSQL	435
MySQL	435
CHAPTER 2: Interfacing SQL to a Procedural Language	437
Building an Application with SQL and a Procedural Language	437
Access and VBA	438
SQL Server and the .NET languages	439
MySQL and C++.NET or C#	440
MySQL and C	440
MySQL and Perl	441
MySQL and PHP	441
MySQL and Java	441
Oracle SQL and Java	441
Db2 and Java	442
CHAPTER 3: Using SQL in an Application Program	443
Comparing SQL with Procedural Languages	444
Classic procedural languages	444
Object-oriented procedural languages	445
Nonprocedural languages	445

Difficulties in Combining SQL with a Procedural Language	446
Challenges of using SQL with a classical procedural language	446
Challenges of using SQL with an object-oriented procedural language	447
Embedding SQL in an Application	448
Embedding SQL in an Oracle Pro*C application	449
Embedding SQL in a Java application	451
Using SQL in a Perl application	452
Embedding SQL in a PHP application	452
Using SQL with a Visual Basic .NET application	452
Using SQL with other .NET languages	453
Using SQL Modules with an Application	453
Module declarations	454
Module procedures	455
Modules in Oracle	456
CHAPTER 4: Designing a Sample Application	457
Understanding the Client's Problem	458
Approaching the Problem	458
Interviewing the stakeholders	458
Drafting a detailed statement of requirements	459
Following up with a proposal	459
Determining the Deliverables	460
Finding out what's needed now and later	460
Planning for organization growth	461
Nailing down project scope	462
Building an Entity-Relationship Model	463
Determining what the entities are	464
Relating the entities to one another	464
Transforming the Model	467
Eliminating any many-to-many relationships	467
Normalizing the ER model	470
Creating Tables	471
Changing Table Structure	475
Removing Tables	475
Designing the User Interface	475
CHAPTER 5: Building an Application	477
Designing from the Top Down	477
Determining what the application should include	478
Designing the user interface	478
Connecting the user interface to the database	479

	Coding from the Bottom Up	481
	Preparing to build the application	481
	Creating the application's building blocks	489
	Gluing everything together	490
	Testing, Testing, Testing	490
	Fixing the bugs	491
	Turning naive users loose	491
	Bringing on the hackers	491
	Fixing the newly found bugs	491
	Retesting everything one last time	492
CHAPTER 6:	Understanding SQL's Procedural Capabilities	493
	Embedding SQL Statements in Your Code	494
	Introducing Compound Statements	494
	Atomicity	495
	Variables	496
	Cursors	496
	Assignment	497
	Following the Flow of Control Statements	497
	IF . . . THEN . . . ELSE . . . END IF	497
	CASE . . . END CASE	498
	LOOP . . . END LOOP	499
	LEAVE	500
	WHILE . . . DO . . . END WHILE	500
	REPEAT . . . UNTIL . . . END REPEAT	501
	FOR . . . DO . . . END FOR	501
	ITERATE	502
	Using Stored Procedures	502
	Working with Triggers	503
	Trigger events	505
	Trigger action time	505
	Triggered actions	505
	Triggered SQL statement	506
	Using Stored Functions	506
	Passing Out Privileges	507
	Using Stored Modules	508
CHAPTER 7:	Connecting SQL to a Remote Database	509
	Native Drivers	510
	ODBC and Its Major Components	511
	Application	512
	Driver manager	513
	Drivers	514
	Data sources	515

What Happens When the Application Makes a Request	515
Using handles to identify objects	516
Following the six stages of an ODBC operation	517

BOOK 6: SQL, XML, AND JSON 523

CHAPTER 1: Using XML with SQL	525
Introducing XML	526
Knowing the Parts of an XML Document	527
XML declaration	527
Elements	528
Attributes	529
Entity references	530
Numeric character references	531
Using XML Schema	531
Relating SQL to XML	532
Using the XML Data Type	533
When to use the XML type	533
When not to use the XML type	534
Mapping SQL to XML	535
Mapping character sets to XML	535
Mapping identifiers to XML	535
Mapping data types to XML	536
Mapping nonpredefined data types to XML	537
Mapping tables to XML	542
Handling null values	542
Creating an XML schema for an SQL table	543
Operating on XML Data with SQL Functions	544
XMLELEMENT	545
XMLFOREST	545
XMLCONCAT	546
XMLAGG	546
XMLCOMMENT	547
XMLPARSE	547
XMLPI	548
XMLQUERY	548
XMLCAST	549
Working with XML Predicates	549
DOCUMENT	549
CONTENT	550
XMLEXISTS	550
VALID	550

CHAPTER 2: Storing XML Data in SQL Tables	553
Inserting XML Data into an SQL Pseudotable	553
Creating a Table to Hold XML Data	555
Updating XML Documents	556
Discovering Oracle's Tools for Updating XML Data in a Table	557
APPENDCHILDXML	557
INSERTCHILDXML	558
INSERTXMLBEFORE	559
DELETXML	560
UPDATEXML	561
Introducing Microsoft's Tools for Updating XML Data in a Table	562
Inserting data into a table using OPENXML	562
Using updategrams to map data into database tables	563
Using an updategram namespace and keywords	563
Specifying a mapping schema	565
CHAPTER 3: Retrieving Data from XML Documents	577
XQuery	578
Where XQuery came from	578
What XQuery requires	579
XQuery functionality	579
Usage scenarios	580
FLWOR Expressions	584
The for clause	586
The let clause	587
The where clause	588
The order by clause	589
The return clause	589
XQuery versus SQL	590
Comparing XQuery's FLWOR expression with SQL's SELECT expression	591
Relating XQuery data types to SQL data types	591
CHAPTER 4: Using JSON with SQL	595
Using JSON with SQL	595
The SQL/JSON Data Model	596
SQL/JSON items	596
SQL/JSON sequences	597
Parsing JSON	598
Serializing JSON	598
SQL/JSON Functions	598
Query functions	598
Constructor functions	604
IS JSON predicate	606
JSON nulls and SQL nulls	607
SQL/JSON Path Language	607

BOOK 7: DATABASE TUNING OVERVIEW	609
CHAPTER 1: Tuning the Database	611
Analyzing the Workload	612
Considering the Physical Design	613
Choosing the Right Indexes	614
Avoiding unnecessary indexes	614
Choosing a column to index	615
Using multicolumn indexes	616
Clustering indexes	616
Choosing an index type	618
Weighing the cost of index maintenance	618
Using composite indexes	619
Tuning Indexes	619
Tuning Queries	620
Tuning Transactions	621
Separating User Interactions from Transactions	622
Minimizing Traffic between Application and Server	622
Precompiling Frequently Used Queries	622
CHAPTER 2: Tuning the Environment	623
Surviving Failures with Minimum Data Loss	624
What happens to transactions when no failure occurs?	624
What happens when a failure occurs and a transaction is still active?	625
Tuning the Recovery System	625
Volatile and nonvolatile memory	625
Memory system hierarchy	627
Putting logs and transactions on different disks	628
Tuning write operations	630
Performing database dumps	631
Setting checkpoints	632
Optimizing batch transactions	634
Tuning the Operating System	634
Scheduling threads	634
Determining database buffer size	638
Tuning the page usage factor	639
Maximizing the Hardware You Have	639
Optimizing the placement of code and data on hard disks	639
Tuning the page replacement algorithm	640
Tuning the disk controller cache	640
Adding Hardware	641
Faster processor	642
More RAM	642
Faster hard disks	642

More hard disks	642
Solid State Disk (SSD)	643
RAID arrays	643
Working in Multiprocessor Environments	643
CHAPTER 3: Finding and Eliminating Bottlenecks	645
Pinpointing the Problem	645
Slow query	646
Slow update	646
Determining the Possible Causes of Trouble	647
Problems with indexes	647
Pitfalls in communication	649
Determining whether hardware is robust enough and configured properly	650
Implementing General Principles: A First Step Toward Improving Performance	651
Avoid direct user interaction	651
Examine the application/database interaction	651
Don't ask for columns that you don't need	652
Don't use cursors unless you absolutely have to	652
Precompiled queries	653
Tracking Down Bottlenecks	653
Isolating performance problems	653
Performing a top-down analysis	653
Partitioning	656
Locating hotspots	656
Analyzing Query Efficiency	657
Using query analyzers	657
Finding problem queries	667
Managing Resources Wisely	671
The disk subsystem	671
The database buffer manager	672
The logging subsystem	673
The locking subsystem	673
BOOK 8: APPENDICES	675
APPENDIX A: SQL: 2016 Reserved Words	677
APPENDIX B: Glossary	683
INDEX	691

Introduction

SQL is the internationally recognized standard language for dealing with data in relational databases. Developed by IBM, SQL became an international standard in 1986. The standard was updated in 1989, 1992, 1999, 2003, 2008, 2011, and 2016. It continues to evolve and gain capability. Database vendors continually update their products to incorporate the new features of the ISO/IEC standard. (For the curious out there, ISO is the International Organization for Standardization, and IEC is the International Electrotechnical Commission.)

SQL isn't a general-purpose language, such as C++ or Java. Instead, it's strictly designed to deal with data in relational databases. With SQL, you can carry out all the following tasks:

- » Create a database, including all tables and relationships.
- » Fill database tables with data.
- » Change the data in database tables.
- » Delete data from database tables.
- » Retrieve specific information from database tables.
- » Grant and revoke access to database tables.
- » Protect database tables from corruption due to access conflicts or user mistakes.

About This Book

This book isn't just about SQL; it's also about how SQL fits into the process of creating and maintaining databases and database applications. In this book, I cover how SQL fits into the larger world of application development and how it handles data coming in from other computers, which may be on the other side of the world or even in interplanetary space.

Here are some of the things you can do with this book:

- » Create a model of a proposed system and then translate that model into a database.
- » Find out about the capabilities and limitations of SQL.
- » Discover how to develop reliable and maintainable database systems.
- » Create databases.
- » Speed database queries.
- » Protect databases from hardware failures, software bugs, and Internet attacks.
- » Control access to sensitive information.
- » Write effective database applications.
- » Deal with data from a variety of nontraditional data sources by using XML.

Foolish Assumptions

I know that this is a *For Dummies* book, but I don't really expect that you're a dummy. In fact, I assume that you're a very smart person. After all, you decided to read this book, which is a sign of high intelligence indeed. Therefore, I assume that you may want to do a few things, such as re-create some of the examples in the book. You may even want to enter some SQL code and execute it. To do that, you need at the very least an SQL editor and more likely also a database management system (DBMS) of some sort. Many choices are available, both proprietary and open source. I mention several of these products at various places throughout the book but don't recommend any one in particular. Any product that complies with the ISO/IEC international SQL standard should be fine.

Take claims of ISO/IEC compliance with a grain of salt, however. No DBMS available today is 100 percent compliant with the ISO/IEC SQL standard. For that reason, some of the code examples I give in this book may not work in the particular SQL implementation that you're using. The code samples I use in this book are consistent with the international standard rather than with the syntax of any particular implementation unless I specifically state that the code is for a particular implementation.

Conventions Used in This Book

By *conventions*, I simply mean a set of rules I've employed in this book to present information to you consistently. When you see a term *italicized*, look for its definition, which I've included so that you know what things mean in the context of SQL. Website addresses and email addresses appear in `monofont` so that they stand out from regular text. Many aspects of the SQL language — such as statements, data types, constraints, and keywords — also appear in `monofont`. Code appears in its own font, set off from the rest of the text, like this:

```
CREATE SCHEMA RETAIL1 ;
```

What You Don't Have to Read

I've structured this book modularly — that is, it's designed so that you can easily find just the information you need — so you don't have to read whatever doesn't pertain to your task at hand. Here and there throughout the book, I include sidebars containing interesting information that isn't necessarily integral to the discussion at hand; feel free to skip them. You also don't have to read text marked with the Technical Stuff icons, which parses out über-techy tidbits (which may or may not be your cup of tea).

How This Book Is Organized

SQL All-in-One Desk Reference For Dummies, 3rd Edition is split into eight minibooks. You don't have to read the book sequentially; you don't have to look at every minibook; you don't have to review each chapter; and you don't even have to read all the sections of any particular chapter. (You can if you want to, however; it's a good read.) The table of contents and index can help you quickly find whatever information you need. In this section, I briefly describe what each minibook contains.

Book 1: SQL Concepts

SQL is a language specifically and solely designed to create, operate on, and manage relational databases. I start with a description of databases and how relational databases differ from other kinds. Then I move on to modeling business and other kinds of tasks in relational terms. Next, I cover how SQL relates to relational

databases, provide a detailed description of the components of SQL, and explain how to use those components. I also describe the types of data that SQL deals with, as well as constraints that restrict the data that can be entered into a database.

Book 2: Relational Database Development

Many database development projects, like other software development projects, start in the middle rather than at the beginning, as they should. This fact is responsible for the notorious tendency of software development projects to run behind schedule and over budget. Many self-taught database developers don't even realize that they're starting in the middle; they think they're doing everything right. This minibook introduces the System Development Life Cycle (SDLC), which shows what the true beginning of a software development project is, as well as the middle and the end.

The key to developing an effective database that does what you want is creating an accurate model of the system you're abstracting in your database. I describe modeling in this minibook, as well as the delicate trade-off between performance and reliability. The actual SQL code used to create a database rounds out the discussion.

Book 3: SQL Queries

Queries sit at the core of any database system. The whole reason for storing data in databases is to retrieve the information you want from those databases later. SQL is, above all, a query language. Its specialty is enabling you to extract from a database exactly the information you want without cluttering what you retrieve with a lot of stuff you don't want.

This minibook starts with a description of values, variables, expressions, and functions. Then I provide detailed coverage of the powerful tools SQL gives you to zero in on the information you want, even if that information is scattered across multiple tables.

Book 4: Data Security

Your data is one of your most valuable assets. Acknowledging that fact, I discuss ways to protect it from a diverse array of threats. One threat is outright loss due to hardware failures. Another threat is attack by hackers wielding malicious viruses and worms. In this minibook, I discuss how you can protect yourself from such threats, whether they're random or purposeful.

I also deal extensively with other sources of error, such as the entry of bad data or the harmful interactions of simultaneous users. Finally, I cover how to control access to sensitive data and how to handle errors gracefully when they occur — as they inevitably will.

Book 5: SQL and Programming

SQL's primary use is as a component of an application program that operates on a database. Because SQL is a data language, not a general-purpose programming language, SQL statements must be integrated somehow with the commands of a language such as Visual Basic, Java, C++, or C#. This book outlines the process with the help of a fictitious sample application, taking it from the beginning — when the need for a new application is perceived — to the release of the finished application. Throughout the example, I emphasize best practices.

Book 6: SQL and XML

XML is the language used to transport data between dissimilar data stores. The 2005 extensions to the SQL:2003 standard greatly expanded SQL's capacity to handle XML data. This minibook covers the basics of XML and how it relates to SQL. I describe SQL functions that are specifically designed to operate on data in XML format, as well as the operations of storing and retrieving data in XML format.

Book 7: Database Tuning Overview

Depending on how they're structured, databases can respond efficiently to requests for information or perform very poorly. Often, the performance of a database degrades over time as its structure and the data in it change or as typical types of retrievals change. This minibook describes the parts of a database that are amenable to tuning and optimization. It also gives a procedure for tracking down bottlenecks that are choking the performance of the entire system.

Book 8: Appendices

Appendix A lists words that have a special meaning in SQL:2016. You can't use these words as the names of tables, columns, views, or anything other than what they were meant to be used for. If you receive a strange error message for an SQL statement that you entered, check whether you inadvertently used a reserved word inappropriately.

Appendix B is a glossary that provides brief definitions of many of the terms used in this book, as well as many others that relate to SQL and databases, whether they're used in this book or not.

Icons Used in This Book

For *Dummies* books are known for those helpful icons that point you in the direction of really great information. This section briefly describes the icons used in this book.



TIP

The Tip icon points out helpful information that's likely to make your job easier.



REMEMBER

This icon marks a generally interesting and useful fact — something that you may want to remember for later use.



WARNING

The Warning icon highlights lurking danger. When you see this icon, pay attention, and proceed with caution.



TECHNICAL
STUFF

This icon denotes techie stuff nearby. If you're not feeling very techie, you can skip this info.

Where to Go from Here

Book 1 is the place to go if you're just getting started with databases. It explains why databases are useful and describes the different types. It focuses on the relational model and describes SQL's structure and features.

Book 2 goes into detail on how to build a database that's reliable as well as responsive. Unreliable databases are much too easy to create, and this minibook tells you how to avoid the pitfalls that lie in wait for the unwary.

Go directly to Book 3 if your database already exists and you just want to know how to use SQL to pull from it the information you want.