

Biju Thomas

OCA

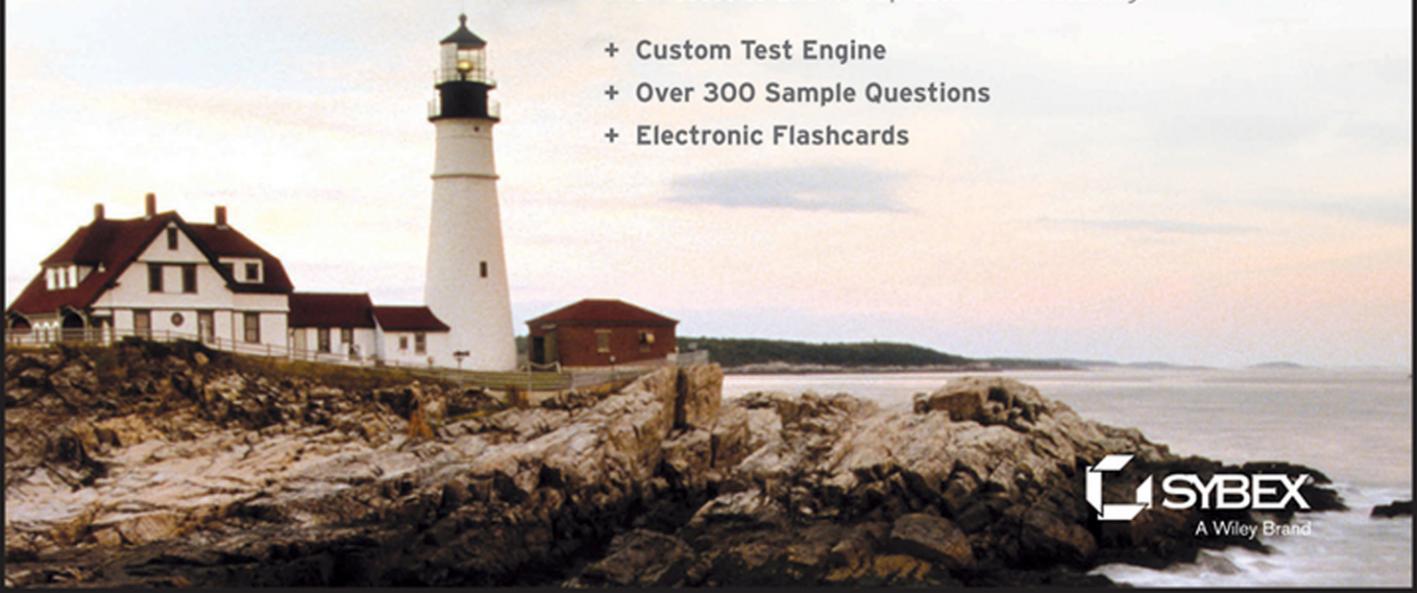
**Oracle Database 12c Administrator
Certified Associate**

STUDY GUIDE

**EXAM 1Z0-061
EXAM 1Z0-062**

Includes Real-World Scenarios, Hands-On Exercises,
and Access to Exam Prep Software Featuring:

- + Custom Test Engine
- + Over 300 Sample Questions
- + Electronic Flashcards



SYBEX
A Wiley Brand

OCA: Oracle® Database 12c Administrator Certified Associate

Study Guide



Biju Thomas

Senior Acquisitions Editor: Jeff Kellum
Development Editor: Lisa Bishop
Technical Editors: Arup Nanda and Syed Jaffar Hussain
Production Editor: Dassi Zeidel
Copy Editor: Kathy Grider-Carlyle
Editorial Manager: Pete Gaughan
Vice President and Executive Group Publisher: Richard Swadley
Associate Publisher: Chris Webb
Media Project Manager 1: Laura Moss-Hollister
Media Associate Producer: Shawn Patrick
Media Quality Assurance: Marilyn Hummel
Book Designer: Judy Fung
Compositor: Craig Woods, Happenstance Type-O-Rama
Proofreader: Kathy Pope
Indexer: Ted Laux
Project Coordinator, Cover: Todd Klemme
Cover Designer: Wiley
Cover Image: © Getty Images Inc./Jeremy Woodhouse
Copyright © 2014 by John Wiley & Sons, Inc., Indianapolis, Indiana
Published simultaneously in Canada
ISBN: 978-1-118-64395-2
ISBN: 978-1-118-76325-4 (ebk)
ISBN: 978-1-118-93133-2 (ebk)

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 either the prior written permission of the Publisher, or authorization through payment of the appropriate per-copy fee to the Copyright Clearance Center, 222 Rosewood Drive, Danvers, MA 01923, (978) 750-8400, fax (978) 646-8600. 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>.

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 Web site 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 Web site may provide or recommendations it may make. Further, readers should be aware that Internet Web sites 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 or to obtain technical support, 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.

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: 2014930413

TRADEMARKS: Wiley and the Sybex logo are trademarks or registered trademarks of John Wiley & Sons, Inc. and/or its affiliates, in the United States and other countries, and may not be used without written permission. Oracle is a registered trademark of Oracle, Inc. 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.

10 9 8 7 6 5 4 3 2 1

Dear Reader,

Thank you for choosing *OCA: Oracle Database 12c Administrator Certified Associate Study Guide*. This book is part of a family of premium-quality Sybex books, all of which are written by outstanding authors who combine practical experience with a gift for teaching.

Sybex was founded in 1976. More than 30 years later, we're still committed to producing consistently exceptional books. With each of our titles, we're working hard to set a new standard for the industry. From the paper we print on, to the authors we work with, our goal is to bring you the best books available.

I hope you see all that reflected in these pages. I'd be very interested to hear your comments and get your feedback on how we're doing. Feel free to let me know what you think about this or any other Sybex book by sending me an email at contactus@sybex.com. If you think you've found a technical error in this book, please visit <http://sybex.custhelp.com>. Customer feedback is critical to our efforts at Sybex.

Best regards,

A handwritten signature in black ink, appearing to read "Chris Webb".

Chris Webb
Associate Publisher
Sybex, an Imprint of Wiley

*To the ladies in my family: my lovely wife, loving mom, and lovable sisters.
Thanks for always being there for me.*

To my children: Joshua and Jeanette. You brighten my day, every day!

Acknowledgments

I thank the great publishing team at Wiley for helping me throughout the writing process. I am grateful to Jeff Kellum, acquisitions editor at Wiley, for initiating this project and having faith in me. I am indebted to Lisa Bishop, developmental editor, for offering valuable suggestions to improve the readability and organization of the book. Thank you both for your patience and hard work throughout this project.

Thank you Kathy Grider-Carlyle, copy editor, for meticulously going through the chapters and making sure there is no ambiguity and that all the pieces tie together. I thank Dassi Zeidel, production editor, for ensuring consistency and accuracy and coordinating the production process. I would like to thank Pete Gaughan, Connor O'Brien, Jenni Housh, Rayna Erlick, and everyone at Wiley who helped in the making of this book.

A technical book is judged by not only who wrote it, but also who reviewed it. I am much honored to have two great Oracle experts as technical reviewers, both Oracle ACE Directors and recipients of the DBA of the Year award. I thank Arup Nanda for reviewing each paragraph of every chapter and for the invaluable comments, tips, and edits. I thank Syed Jaffar Hussain for reading and ensuring the technical accuracy. The input from two Oracle stalwarts certainly improved the content and reliability of the book.

I sincerely thank Gavin Powell for initially signing up as a coauthor to this book. Due to unforeseen events, Gavin could complete only one chapter. I thank Gavin for working with me and helping with the book.

I have many friends to thank at OneNeck IT Solutions for their friendship and encouragement. My heartfelt thanks to Chuck Vermillion (SVP & General Manager) for his support and confidence in me. Thank you Danny Carrizosa for the motivation and your willingness to help out wherever needed. I thank Phil Pearson and the entire team for their trust in my technical abilities. It sure is great working with you all.

Last but not least, no words can express the ceaseless support and encouragement given by my dear wife. Thank you, Shiji. You are the best. I owe my kids several weekends and thank them for their patience and tolerance of so much of my attention being given to this book.

About the Author

Biju Thomas is an Oracle ACE, Oracle Certified Professional, and Certified Oracle Database SQL Expert. Biju has been developing and administering Oracle databases since 1993. He now spends time mentoring DBAs, performance tuning, and designing Oracle solutions. He is Principal Solutions Architect at OneNeck IT Solutions (<http://www.oneneck.com>). He is a frequent presenter at Oracle conferences and a contributor to Oracle technical journals. He blogs at <http://www.bijoos.com/oraclenotes>, and you can follow his tweets @biju_thomas.

About the Contributor

Gavin Powell spent 25 years as an IT professional in positions such as Oracle DBA, performance tuner, data architect, and developer. In the last 10 years, he has branched out into technical writing (many titles in print) in between contract and consulting jobs. He also enjoys music, songwriting, performing, singing, playing too many instruments, home recording/producing, and studying music at Berklee.

About the Technical Reviewers

Arup Nanda has been an Oracle DBA for the last 20 years, working on all aspects of Oracle technology from modeling to performance tuning to disaster recovery. He has written over 500 published articles, presented over 300 sessions in 22 countries, and coauthored 5 books. He is an Oracle ACE Director, Oak Table Network member, editor for *SELECT Journal*, and a board member of Exadata SIG of IOUG. In recognition, Oracle conferred on him the coveted DBA of the Year and Architect of the Year awards in 2003 and 2012, respectively. He lives in Danbury, CT.

Syed Jaffar Hussain has over 21 years IT experience that includes more than 14 years of production Oracle database administration. Oracle has honored him with the prestigious Oracle ACE Director role and named him DBA of the Year for 2011, both for his excellent knowledge and contributions to the Oracle community. He is an Oracle Certified Master (OCM) for Oracle Database 10g, a status granted only after passing extensive challenges in a hands-on environment. He is also an Oracle Database 10g RAC Certified Expert. Syed Jaffar is a well-known Oracle speaker, and he coauthored *Expert Oracle RAC12* and *Oracle 11g R1/R2 Real Application Clusters Essentials*. He blogs regularly at <http://jaffardba.blogspot.com>.

Contents at a Glance

<i>Introduction</i>		<i>xx</i>
<i>Assessment Test</i>		<i>xxxv</i>
Part I	Oracle Database 12c: SQL Fundamentals	1
Chapter 1	Introducing Oracle Database 12c RDBMS	3
Chapter 2	Introducing SQL	35
Chapter 3	Using Single-Row Functions	89
Chapter 4	Using Group Functions	175
Chapter 5	Using Joins and Subqueries	227
Chapter 6	Manipulating Data	281
Chapter 7	Creating Tables and Constraints	319
Part II	Oracle Database 12c: Installation and Administration	381
Chapter 8	Introducing Oracle Database 12c Components and Architecture	383
Chapter 9	Creating and Operating Oracle Database 12c	431
Chapter 10	Understanding Storage and Space Management	511
Chapter 11	Managing Data Concurrency and Undo	569
Chapter 12	Understanding Oracle Network Architecture	603
Chapter 13	Implementing Security and Auditing	677
Chapter 14	Maintaining the Database and Managing Performance	753
Chapter 15	Using Backup and Recovery	825
Chapter 16	Controlling Resources and Jobs	903
Chapter 17	Upgrading to Oracle Database 12c	957
Chapter 18	Using Grid Infrastructure and Data Movement Tools	1005
Appendix A	Answers to Review Questions	1083
Appendix B	About the Additional Study Tools	1119
<i>Index</i>		1123

Contents

<i>Introduction</i>	<i>xx</i>	
<i>Assessment Test</i>	<i>xxxv</i>	
Part I	Oracle Database 12c: SQL Fundamentals	1
Chapter 1	Introducing Oracle Database 12c RDBMS	3
Relational Database Management Systems	4	
Characteristics of a Relational Database	5	
Oracle's Implementation of RDBMS and ORDBMS	8	
The Oracle Database 12c	9	
Oracle Database 12c Implementations	9	
Connecting to Oracle Database	13	
Database Management Tools	14	
Oracle Database 12c in the Cloud	19	
Familiarizing SQL*Plus	19	
Summary	30	
Exam Essentials	31	
Review Questions	32	
Chapter 2	Introducing SQL	35
SQL Fundamentals	36	
Oracle Datatypes	38	
Operators and Literals	42	
Writing Simple Queries	46	
Using the SELECT Statement	46	
Filtering Rows	51	
Sorting Rows	62	
Sorting NULLs	64	
Limiting Rows	67	
Using Expressions	68	
Accepting Values at Runtime	72	
Using Ampersand Substitution Variables	73	
Saving a Variable for a Session	75	
Using Positional Notation for Variables	77	
Summary	78	
Exam Essentials	78	
Review Questions	80	
Chapter 3	Using Single-Row Functions	89
Single-Row Function Fundamentals	90	
Functions for NULL Handling	91	

Using Single-Row Character Functions	95	
Character Function Overview	95	
Character Function Descriptions	96	
Using Single-Row Numeric Functions	108	
Numeric Function Overview	108	
Numeric Function Descriptions	109	
Using Single-Row Date Functions	119	
Date Format Conversion	119	
Date Function Overview	120	
Date Function Descriptions	121	
Using Single-Row Conversion Functions	131	
Conversion Function Overview	131	
Conversion Function Descriptions	133	
Using Other Single-Row Functions	154	
Miscellaneous Function Overview	154	
Miscellaneous Function Descriptions	155	
Summary	167	
Exam Essentials	168	
Review Questions	169	
Chapter 4	Using Group Functions	175
Group Function Fundamentals	176	
Utilizing Aggregate Functions	177	
Grouping Data with GROUP BY	178	
Group Function Overview	182	
Group Function Descriptions: Part 1	184	
Group Function Descriptions: Part 2	194	
Limiting Grouped Data with HAVING	207	
Creating Superaggregates with CUBE and ROLLUP	208	
Nesting Functions	215	
Summary	218	
Exam Essentials	218	
Review Questions	219	
Chapter 5	Using Joins and Subqueries	227
Writing Multiple-Table Queries	228	
Inner Joins	229	
Cartesian Joins	239	
Outer Joins	240	
Other Multiple-Table Queries	245	
Using Set Operators	247	
The UNION Operator	248	
The UNION ALL Operator	249	
The INTERSECT Operator	250	

	The MINUS Operator	250
	Putting It All Together	250
	Using Subqueries	252
	Single-Row Subqueries	253
	Multiple-Row Subqueries	254
	Subquery Returns No Rows	256
	Correlated Subqueries	258
	Scalar Subqueries	259
	Multiple-Column Subqueries	267
	Subqueries in Other DML Statements	268
	Summary	269
	Exam Essentials	270
	Review Questions	271
Chapter 6	Manipulating Data	281
	Using DML Statements	282
	Inserting Rows into a Table	283
	Updating Rows in a Table	289
	Deleting Rows from a Table	294
	Merging Rows	297
	Understanding Transaction Control	299
	Savepoints and Partial Rollbacks	302
	Data Visibility	304
	Summary	306
	Exam Essentials	307
	Review Questions	308
Chapter 7	Creating Tables and Constraints	319
	Database Objects Overview	320
	Schema Objects	322
	Using Sequences	324
	Built-in Datatypes	325
	Character Datatypes	325
	Numeric Datatypes	329
	Date and Time Datatypes	330
	Date Arithmetic	334
	Binary Datatypes	336
	Row ID Datatypes	337
	Creating Tables	338
	Naming Tables and Columns	339
	Specifying Default Values for Columns	341
	Adding Comments	348
	Creating a Table from Another Table	348
	Modifying Tables	350
	Adding Columns	350

	Modifying Columns	353
	Renaming Columns	354
	Dropping Columns	354
	Hiding Columns from Table	356
	Dropping Tables	357
	Renaming Tables	357
	Making Tables Read-Only	358
	Managing Constraints	361
	Creating Constraints	362
	Dropping Constraints	367
	Enabling and Disabling Constraints	367
	Deferring Constraint Checks	369
	Summary	373
	Exam Essentials	374
	Review Questions	375
Part II	Oracle Database 12c: Installation and Administration	381
Chapter 8	Introducing Oracle Database 12c Components and Architecture	383
	Oracle Database Fundamentals	385
	Relational Databases	385
	Oracle Database 12c Objects	386
	Interacting with Oracle Database 12c	388
	Oracle Database 12c Architecture	391
	User and Server Processes	394
	The Oracle Instance	396
	Oracle Storage Structures	410
	Summary	424
	Exam Essentials	425
	Review Questions	426
Chapter 9	Creating and Operating Oracle Database 12c	431
	Oracle Database 12c Software Installation	432
	Planning the Oracle Database 12c Software Install	433
	Using the Oracle Universal Installer	440
	Using DBCA to Create an Oracle 12c Database	447
	Invoking the Database Configuration Assistant	447
	Configuring an Oracle Database Using the DBCA	472
	Deleting an Oracle Database Using the DBCA	472
	Managing Database Templates Using the DBCA	473
	Working with Oracle Database Metadata	475
	Data Dictionary Views	476
	Dynamic Performance Views	477

Managing Initialization-Parameter Files	479	
Locating the Default Parameter File	484	
Modifying Initialization-Parameter Values	485	
Starting Up and Shutting Down an Oracle Instance	490	
Starting Up an Oracle Database 12c Instance	491	
Shutting Down an Oracle Database 12c Instance	495	
Monitoring the Database Alert Log	497	
Summary	503	
Exam Essentials	504	
Review Questions	506	
Chapter 10	Understanding Storage and Space Management	511
Understanding the Physical and Logical Storage	512	
Contents of a Data Block	514	
Managing Tablespaces	516	
Identifying Default Tablespaces	516	
Creating and Maintaining Tablespaces	516	
Obtaining Tablespace Information	531	
Managing Data Files	534	
Performing Operations on Data Files	535	
Using the Oracle Managed Files Feature	539	
Querying Data File Information	544	
Managing Space	547	
Automatic Space Management Features	547	
Monitoring Tablespace Free Space	551	
Using Segment Advisor	554	
Avoiding Out-of-Space Errors	559	
Summary	562	
Exam Essentials	563	
Review Questions	565	
Chapter 11	Managing Data Concurrency and Undo	569
Managing Data Changes Using DML	570	
Understanding “Change”	571	
Differentiating Undo and Redo	572	
Configuring and Monitoring Undo	576	
Managing Data Concurrency	584	
Understanding Locks and Transactions	585	
Maximizing Data Concurrency	587	
Detecting and Resolving Lock Conflicts	590	
Summary	595	
Exam Essentials	595	
Review Questions	597	

Chapter 12	Understanding Oracle Network Architecture	603
	Introducing Network Configurations	604
	Single-Tier Architecture	605
	Two-Tier Architecture	605
	<i>n</i> -Tier Architecture	606
	An Overview of Oracle Net Features	607
	Connectivity	608
	Manageability	608
	Scalability	609
	Security	611
	Accessibility	613
	Configuring Oracle Net on the Server	613
	Understanding the Oracle Listener	614
	Managing Oracle Listeners	618
	Dynamically Registering Services	634
	Oracle Net Logging and Tracing on the Server	635
	Configuring Oracle Net for the Client	637
	Client-Side Names Resolution Options	637
	The Host Naming Method	638
	The Oracle Easy Connect Method	639
	The Local Naming Method	641
	Troubleshooting Client-Side Connection Problems	646
	An Overview of Oracle Shared Server	648
	Dedicated Server vs. Shared Server	648
	Advantages and Disadvantages of Shared Server	650
	Oracle Shared Server Infrastructure	651
	Configuring the Oracle Shared Server	653
	Managing a Shared Server	658
	Understanding Database Resident Connection Pooling	661
	Configuring DRCP	662
	Comparing Connection Architectures	663
	Communicating Between Databases	666
	Introduction to Database Links	666
	Creating Database Links	667
	Summary	669
	Exam Essentials	670
	Review Questions	672
Chapter 13	Implementing Security and Auditing	677
	Creating and Managing User Accounts	678
	Configuring Authentication	679
	Assigning Tablespaces and Quotas	681
	Assigning a Profile and Account Settings	684

	Removing a User from the Database	686
	Managing Default User Accounts	687
Granting and Revoking Privileges		687
	Granting Object Privileges	688
	Granting System Privileges	692
	Role Privileges	701
	Applying the Principle of Least Privilege	705
Controlling Resource Usage by Users		711
	Implementing Password Security Features	714
Auditing Database Activity		718
	Managing Statement Auditing	720
	Managing Privilege Auditing	724
	Managing Object Auditing	725
	Using SQL Developer for Audit Management	727
	Purging the Audit Trail	729
	Managing Fine-Grained Auditing	732
	Implementing Unified Auditing	735
	Summary	745
	Exam Essentials	746
	Review Questions	747
Chapter 14	Maintaining the Database and Managing Performance	753
	Proactive Database Maintenance	754
	Managing Optimizer Statistics	755
	Gathering Performance Statistics	775
	Automatic Database Diagnostic Monitoring	782
	The Advisory Framework	793
	Monitoring Server-Generated Alerts	798
	Understanding Automatic Diagnostic Repository	801
	Managing Performance	806
	Sources of Tuning Information	807
	Tuning Memory	812
	Automatic Shared Memory Management	813
	Automatic SQL Execution Memory Management	815
	Managing Memory Using EM Cloud Control	817
	Summary	819
	Exam Essentials	820
	Review Questions	821
Chapter 15	Using Backup and Recovery	825
	Understanding and Configuring Recovery Components	827
	Understanding Control Files	827
	Understanding Checkpoints	831

Understanding Redo Log Files	832
Understanding Archived Redo Log (ARCHIVELOG) Files	839
Understanding the Fast Recovery Area	844
Performing Backups	848
Understanding Backup Terminology	848
Backing Up the Control File	850
Backing Up the Database	851
Using RMAN to Perform Backups	852
Managing Backups	861
Understanding Types of Database Failures	864
Statement Failures	865
User-Process Failures	865
Network Failures	866
User-Error Failures	866
Instance Failures	867
Media Failures	868
Performing Recovery Operations	868
Understanding Instance Startup	868
Keeping an Instance from Failing	870
Recovering from Instance Failure	870
Tuning Instance Recovery	871
Recovering from User Errors	873
Recovering from the Loss of a Control File	881
Using the Data Recovery Advisor	883
Recovering from the Loss of a Redo Log File	884
Recovering from the Loss of a Non-System-Critical Data File	886
Recovering from the Loss of a System-Critical Data File	892
Table Recovery Using RMAN	893
Summary	894
Exam Essentials	895
Review Questions	897
Chapter 16	
Controlling Resources and Jobs	903
Resource Management with the Resource Manager	904
Functions of the Resource Manager	904
Understanding Resource Manager Components	905
Configuring the Database Resource Manager	911
Accessing and Creating Resource Plans	916
Updating and Deleting Resources	924
Monitoring the Resource Manager	927
Task Automation with Oracle Scheduler	930
Scheduler Architecture	930
Using Oracle Scheduler	934

Relating Tasks with Job Chains	943	
Scheduling Jobs on Remote Systems	947	
Prioritizing Jobs with Oracle Scheduler	948	
Summary	951	
Exam Essentials	951	
Review Questions	952	
Chapter 17	Upgrading to Oracle Database 12c	957
Determining the Database Upgrade Method	958	
Using Direct Upgrade	959	
Migrating to Oracle Database 12 <i>c</i>	961	
Preparing for a Database Upgrade	963	
Running the Pre-Upgrade Information Tool	964	
Preparing the Oracle Home	974	
Reducing Upgrade Downtime	975	
Upgrading the Database	976	
Using Database Upgrade Assistant	976	
Performing a Manual Upgrade	987	
Completing the Post-Upgrade Tasks	991	
Migrating to Unified Auditing	994	
Downgrading to an Earlier Release	995	
Summary	997	
Exam Essentials	997	
Review Questions	999	
Chapter 18	Using Grid Infrastructure and Data Movement Tools	1005
Tools for Moving Data	1006	
Migrating Data Using Data Pump	1008	
Data Pump Architecture	1009	
Using Data Pump Clients	1012	
Using the Data Pump Wizard	1033	
Upgrading an 11g R2 Database to 12 <i>c</i>		
Using Full Transportable Export	1037	
Loading Data with SQL*Loader	1039	
Specifying SQL*Loader Command-Line Parameters	1040	
Specifying Control File Options	1042	
Populating External Tables	1046	
Loading External Tables Using Data Pump	1046	
Loading External Tables Using Loader	1048	
Introducing Grid Infrastructure	1050	
Installing Oracle Grid Infrastructure	1051	
Managing Oracle ASM Storage	1061	
Using Oracle Restart	1068	

Summary	1076
Exam Essentials	1077
Review Questions	1078
Appendix A Answers to Review Questions	1083
Chapter 1: Introducing Oracle Database 12c RDBMS	1084
Chapter 2: Introducing SQL	1085
Chapter 3: Using Single-Row Functions	1087
Chapter 4: Using Group Functions	1089
Chapter 5: Using Joins and Subqueries	1090
Chapter 6: Manipulating Data	1092
Chapter 7: Creating Tables and Constraints	1094
Chapter 8: Introducing Oracle Database 12c Components and Architecture	1096
Chapter 9: Creating and Operating Oracle Database 12c	1098
Chapter 10: Understanding Storage and Space Management	1100
Chapter 11: Managing Data Concurrency and Undo	1102
Chapter 12: Understanding Oracle Network Architecture	1104
Chapter 13: Implementing Security and Auditing	1106
Chapter 14: Maintaining the Database and Managing Performance	1108
Chapter 15: Using Backup and Recovery	1110
Chapter 16: Controlling Resources and Jobs	1112
Chapter 17: Upgrading to Oracle Database 12c	1114
Chapter 18: Using Grid Infrastructure and Data Movement Tools	1116
Appendix B About the Additional Study Tools	1119
Additional Study Tools	1120
Sybex Test Engine	1120
Electronic Flashcards	1120
Bonus Author Materials	1120
PDF of Glossary of Terms	1120
Adobe Reader	1121
System Requirements	1121
Using the Study Tools	1121
Troubleshooting	1121
Customer Care	1122
<i>Index</i>	1123

Table of Exercises

Exercise 1.1	Practicing SQL*Plus File Commands	26
Exercise 9.1	Manually Delete or Remove an Oracle Database	473
Exercise 9.2	Creating an Oracle Database 12c Database.....	501

Introduction

There is high demand for professionals in the information technology (IT) industry, and Oracle certifications are the hottest credential in the database world. You have made the right decision to pursue certification, because being certified in the latest version of Oracle, Oracle Database 12c, will give you a distinct advantage in this highly competitive market.

Many readers may already be familiar with Oracle and do not need an introduction to Oracle databases. For those who aren't familiar with the company, Oracle, founded in 1977, sold the first commercial relational database and is now the world's leading database company and second-largest independent software company with revenues of more than \$37 billion, and is headquartered in Redwood City, California.

Oracle databases are the de facto standard for large Internet sites, mission-critical enterprise applications, and cloud solutions. With the acquisition of Sun Microsystems, Oracle offers complete enterprise business solutions with engineered systems capable of running world-class databases and applications. Enterprise Resource Planning (ERP) application suites, data warehouses, and business applications at many large and medium-sized companies rely on Oracle. The demand for DBA resources remains higher than others during weak economic times.

This book is intended to help you on your exciting path toward becoming an Oracle Database 12c Administrator Certified Associate (OCA), which is the first step on the path toward Oracle Certified Professional (OCP) and Oracle Certified Master (OCM) certification. This book covers two of the exams required for the OCA certification. Using this book and a practice database, you can start learning Oracle Database 12c and pass the 1Z0-061 "Oracle Database 12c: SQL Fundamentals" and 1Z0-062 "Oracle Database 12c: Installation and Administration" exams.

Why Become Oracle Certified?

The number one reason to become OCA or OCP certified is to gain more visibility and greater access to the industry's most challenging opportunities. Oracle certification is the best way to demonstrate your knowledge and skills in Oracle database systems. Preparing for the certification exam may be the best time spent on your career because you learn the tasks that are necessary to be successful as a DBA.

Certification is proof of your knowledge and shows that you have the skills required to support Oracle core products. The Oracle certification program can help a company identify proven performers who have demonstrated their skills and who can support the company's investment in Oracle technology. It demonstrates that you have a solid understanding of your job role and the Oracle products used in that role.

The certification tests are scenario-based, which is the most effective way to assess your hands-on expertise and critical problem-solving skills. OCPs are among the best paid in the IT industry. Salary surveys consistently show the OCP certification to yield higher salaries than the other certifications, including Microsoft, Novell, and Cisco.

So, whether you are beginning a career, changing careers, securing your present position, or seeking to refine and promote your position, this book is for you!

Oracle Certifications

Oracle certifications follow a track that is oriented toward a job role. The primary certification tracks are Database, Applications, Java, Enterprise Management, Virtualization, and Operating Systems. Within each track, Oracle has a tiered certification program of OCA and OCP. Only the Database track has OCM. The Database track is clearly for the Database Administrator job role.



For the latest certification information on all of Oracle certification paths, please visit the Oracle website at http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=39&p_org_id=1001&lang=US.

The role of database administrator (DBA) has become a key to success in today's highly complex database systems. The best DBAs work behind the scenes, but are in the spotlight when critical issues arise. They plan, create, maintain, and ensure that the database is available for the business, most importantly, DBAs troubleshoot, diagnose, and resolve issues. They have tools to proactively monitor the database for performance issues and to prevent unscheduled downtime. The DBA's job requires a broad understanding of the architecture of an Oracle database and expertise in solving problems.

Sybex has Oracle certification study guides for the Database track. The following sections will introduce you to the different tiers in the Oracle Database 12c certification track.

Oracle Database 12c Administrator Certified Associate

The Oracle Certified Associate (OCA) credential is the first step toward achieving the Oracle Certified Professional (OCP) certification. OCA shows that you have the fundamental knowledge and skills to support an Oracle Database 12c database. This certification requires you to pass two exams that demonstrate your Oracle basics:

- 1Z0-061: Oracle Database 12c: SQL Fundamentals
- 1Z0-062: Oracle Database 12c: Installation and Administration

If you have already passed any one of the following tests, you need not take the 1Z0-061 exam. You only need to pass the 1Z0-062.

- 1Z0-051: Oracle Database 11g SQL Fundamentals I
- 1Z0-047: Oracle Database SQL Expert

The 1Z0-061 exam can be taken at a testing location or from your home using the Internet. The 1Z0-062 test is offered at a Pearson Vue facility.



To register for the test, visit Pearson Vue at <http://www.pearsonvue.com>.

Oracle Database 12c Administrator Certified Professional

The Oracle Certified Professional credential shows that you have the skill and technical expertise to manage and implement enterprise databases. The OCP tier challenges you to demonstrate your continuing experience and knowledge of Oracle technologies. The OCP test will measure your knowledge in setting up and managing multitenant architecture databases and in backup and recovery. The Oracle Database 12c Administrator Certified Professional certification requires you to have the OCA certification as well as to pass the following exam.

- 1Z0-063: Oracle Database 12c: Advanced Administration

In addition, the OCP candidate must take one instructor-led Oracle university hands-on requirement class.



You should verify the list of approved hands-on courses at the Oracle Education website at http://education.oracle.com/pls/web_prod-plq-dad/db_pages.getpage?page_id=244#5.

Oracle Database 12c Administrator Certified Master

The highest level of certification available in any track is the Oracle Certified Master. The OCM certification credential shows that you have the highest level of expertise in an Oracle product. To become a certified master, you must first achieve OCP status and then complete two advanced instructor-led classes at an Oracle Education facility. You must also pass a hands-on examination at an Oracle Education facility. At the time of writing this book, the Oracle Database 12c Certified Master Exam is not released.

More Information and Resources

You can find most current information about Oracle certification at <http://education.oracle.com/certification>. You may be asked to choose the country of residence before being directed to the site. Follow the links under Certifications to choose the track and learn more.

Choose the Database track to view the different certification versions available. Choose Oracle Database 12c Administrator Certified Associate, and then click on the test to find out more about the test contents, the objectives covered in the test, the passing score, and to register for the test.

The Oracle documentation is available online at <http://tahiti.oracle.com>. Oracle documentation contains a wealth of information, which can be used to supplement what you learn from this book.

Oracle provides training series with step-by-step instructions to perform a variety of Oracle Database 12c tasks. The Oracle by Example (OBE) tutorial can be found at <http://apex.oracle.com/pls/apex/f?p=44785:1>.

Oracle Technology Network (<http://www.oracle.com/technology/index.html>) is a great resource for database administrators and developers. You can read articles; view sample code; access documentation; participate in forums; and most importantly, download Oracle Database 12c, Oracle Enterprise Manager Cloud Control 12c, and other Oracle products.

OCA/OCP Study Guides

The Oracle Database 12c Administration track certification consists of three tests: two for OCA and one for OCP. Sybex offers study guides to help you achieve OCA and OCP certification.

- *OCA: Oracle Database 12c Administrator Certified Associate Study Guide* (9781118643952) – Covers exams 1Z0-061: Oracle Database 12c: SQL Fundamentals and 1Z0-062: Oracle Database 12c: Installation and Administration.
- *OCP: Oracle Database 12c Administrator Certified Professional Study Guide* (9781118644072) – Covers exam 1Z0-063: Oracle Database 12c: Advanced Administration.

These two books are offered in a boxed set as *OCP: Oracle Certified Professional on Oracle Database 12c Certification Kit* (9781118957684).

Oracle Exam Requirements

The Oracle Database 12c Database Certified Associate certification tests your basic SQL skills for the SQL exam and database architecture and administration skills for the DBA exam. The SQL exam tests your knowledge of writing SQL and using the functions available in Oracle Database 12c. The Installation and Administration exam concentrates on the architecture and the basic administration of Oracle 12c databases. The following sections detail the skills needed to pass the SQL Fundamentals and the Installation and Administration exams.

OCA SQL (1Z0-061) Requirements

To pass the Oracle Database 12c SQL Fundamentals exam, you must have the following skills:

- Write SQL SELECT statements that display data from one or more tables
- Join tables using ANSI syntax and Oracle traditional syntax
- Restrict, sort, and aggregate data using single-row, conversion, and group functions
- Write subqueries and queries using SET operators
- Manipulate data: insert, update, delete
- Create and manage tables and other database objects

OCA Installation and Administration (1Z0-062) Requirements

To pass the Oracle Database 12c Installation and Administration exam, you must have the following skills:

- Understand the Oracle server architecture (database and instance)
- Install Oracle Database 12c software and create a database
- Use Database Configuration Assistant and Enterprise Manager tools
- Understand the physical and logical storage of the database, and be able to manage space allocation and growth

- Use the data dictionary views and set database parameters
- Manage and manipulate data, including its storage, loading, and reorganization
- Create and manage tables, constraints, and indexes
- Manage redo logs, archive logs, and automatic undo
- Configure Oracle Net on the server side and client side
- Understand backup and recovery architecture
- Secure the database and audit database usage
- Use advisors to tune and manage the database
- Upgrade the database
- Move data between databases
- Install and manage Grid Infrastructure, Oracle Restart, and Automatic Storage Management

Tips for Taking the OCA Exams

The following tips will help you prepare for and pass each exam.

- Each test consists of between 70 and 95 questions to be completed in 120 to 150 minutes (depending on the exam). Answer the questions you are sure of first, before you run out of time.
- Many questions on the exam have answer choices that at first glance look identical. Read the questions carefully. Do not jump to conclusions. Make sure you clearly understand what each question asks.
- Most questions are based on scenarios. Some of the scenarios contain nonessential information and exhibits. You need to be able to identify what's important and what's not.
- Do not leave any questions unanswered. There is no negative scoring. After selecting an answer, you can mark difficult questions or the ones you are unsure of and come back to them later.
- When answering questions you are not sure about, use a process of elimination to get rid of the obviously incorrect answers first. Doing this greatly improves your odds if you need to make an educated guess.
- If you are not sure of your answer, mark it for review and then look for other questions that may help you eliminate any incorrect answers. At the end of the test, you can review the questions you marked earlier.



You should be familiar with the exam objectives, which are included at the beginning of each chapter. Please check the objective listing on the Oracle Education website for any changes or updates. The detail page for each exam shows the passing score, number of questions, and minutes allocated along with the exam fees and any other requirements.

What Is Covered in This Book

This book covers everything you need to pass the Oracle Database 12c Certified Associate exams. Part I includes the first seven chapters that cover the objectives for the Oracle Database 12c SQL Fundamentals exam. Part II of the book includes the remaining 11 chapters that cover the objectives for the Oracle Database 12c Installation and Administration exam.

Part I: Oracle Database 12c SQL Fundamentals

Chapter 1: Introducing Oracle Database 12c RDBMS introduces you to the features of Oracle Database 12c and the aspects of a relational database.

Chapter 2: Introducing SQL introduces you to writing simple queries using SELECT statements. It also introduces you to filtering and sorting data.

Chapter 3: Using Single-Row Functions discusses the single-row functions and conversion functions available, with details on how and where to use them.

Chapter 4: Using Group Functions explains data aggregations, Oracle's built-in group function, and nesting of functions.

Chapter 5: Using Joins and Subqueries explains how data from multiple tables can be related via joins, subqueries, and by using SET operators.

Chapter 6: Manipulating Data explores how to manipulate data: adding, removing, and updating data. The chapter also covers how transaction control works.

Chapter 7: Creating Tables and Constraints explains how to create and manage tables and constraints. It also discusses the various datatypes available in Oracle Database 12c to store data.

Part II: Oracle Database 12c Installation and Administration

Chapter 8: Introducing Oracle Database 12c Components and Architecture is the first chapter to start if you're reading for the Oracle Database 12c Installation and Administration exam. This chapter introduces you to the Oracle Database 12c database architecture and the relationship between logical and physical storage structures.

Chapter 9: Creating and Operating an Oracle Database 12c explains how you can install the Oracle Database 12c software and create a database. It discusses the initialization parameters, stages of database startup and shutdown, where to find log and trace files, and how to use the data dictionary.

Chapter 10: Understanding Storage and Space Management explores the logical and physical storage of the database. You will learn space management and the various types of tablespaces. This chapter also discusses monitoring space and reclaiming wasted space.

Chapter 11: Managing Data Concurrency and Undo shows you how you can add, update, and remove data from tables, as well as how transactions work. It introduces you to undo data and undo management. This chapter also discusses how Oracle manages locks to ensure data concurrency. Be sure to read Chapter 6 before you read this chapter.

Chapter 12: Understanding Oracle Network Architecture introduces you to the Oracle Net configuration and setup. You will learn to set up network architecture on the server and client.

Chapter 13: Implementing Security and Auditing shows how you can secure your database using privileges, profiles, and roles. You will also learn to audit database usage.

Chapter 14: Maintaining the Database and Managing Performance explores the tools available in Oracle Database 12c to manage the performance of the database. You will learn about optimizer statistics, Automatic Workload Repository, various advisors, and Automatic Memory Management.

Chapter 15: Using Backup and Recovery introduces you to the backup architecture concepts. It discusses the various backup modes and using RMAN. It also delves into the various recovery scenarios and how best to get the data back. It shows you how to use the Data Recovery Advisor, which can help find and recover the database, as well as offer advice.

Chapter 16: Controlling Resources and Jobs shows you how to manage the resources available on the database server using the Resource Manager. You will also learn about the Oracle Scheduler job management system.

Chapter 17: Upgrading to Oracle Database 12c familiarizes you with the process of upgrading a database to Oracle Database 12c. You also learn the various upgrade and migration methods.

Chapter 18: Using Grid Infrastructure and Data Movement Tools introduces you to Data Pump and SQL*Loader, the tools available in Oracle Database 12c to move and load data. This chapter also covers the Grid Infrastructure installation and configuration, including setting up Automatic Storage Management disks.

Each chapter ends with Review Questions that are specifically designed to help you retain the knowledge presented. To really nail down your skills, read and answer each question carefully.

What's Available Online?

The book includes a number of companion study tools, which can be downloaded from www.sybex.com/go/oca12sg. See Appendix B, “About the Additional Study Tools,” for information on how to access and install these tools:

Test Preparation Software The test preparation software prepares you to pass both the 1Z0-061 and 1Z0-062 exams. You'll find all the review and assessment questions from the book *plus* an additional four practice exams (two for each exam) that appear exclusively from the downloadable study tools.

Electronic Flashcards The companion study tools include over 400 flashcards specifically written to hit you hard, so don't get discouraged if you don't ace your way through them at first! They're there to ensure that you're really ready for the exam. And no worries—armed with the review questions, practice exams, and flashcards, you'll be more than prepared when exam day comes!

Author Bonus Material I've included all of the code from the book, as well as three bonus Whitepapers that show you how to install an Oracle 12c database on Windows, how to create non-CDB Oracle Database 12c on Windows, and how to install Oracle Enterprise Manager 12c on Virtual Box.

Glossary A complete glossary of common terms is available at www.sybex.com/go/oca12sg.

How to Use This Book

This book provides a solid foundation for the serious effort of preparing for the Oracle Database 12c Certified Associate exams. To best benefit from the book, use the following study method:

1. Take the assessment tests immediately following this introduction. (The answers are at the end of the tests.) Carefully read the explanations for any questions you get wrong, and note in which chapters the material is covered. This information should help you plan your study strategy.
2. Study each chapter carefully, making sure you fully understand the information and the test objectives listed at the beginning of each chapter. Pay close attention to any chapter related to questions you missed in the assessment test.
3. Complete all hands-on activities in the chapter, referring to the chapter so that you understand the reason for each step you take. It is best to have an Oracle Database 12c database available to try out the examples and the code provided in the book. All the code is also included with the additional study tools.
4. Answer the review questions related to that chapter. Note the review questions that confuse or trick you, and study those sections of the book again.
5. The two bonus exams for each exam are included with the accompanying study tools. They will give you a complete overview of what you can expect to see on the real test.
6. Answer all the flashcard questions included with the study tools.
7. Remember to use the study tools included with this book. The electronic flash cards and the Sybex test engine exam-preparation software has been specifically designed to help you study and pass your exams.



The additional study tools can be downloaded from www.sybex.com/go/oca12sg.

To learn all the material covered in this book, you will need to apply yourself regularly and with discipline. Try to set the same time period every day to study, and select a comfortable and quiet place to do so. If you work hard, you will be surprised at how quickly you will learn this material. You can also find supplemental reading material and Oracle documentation references on my blog that will deepen your knowledge of what you read in this book. All the best!



Prebuilt Oracle Database 12c can be downloaded and set up on VirtualBox. This is convenient, especially if you are new to Oracle. Prebuilt Oracle VMs can be downloaded from <http://www.oracle.com/technetwork/community/developer-vm>. You will have to install and set up Oracle VM VirtualBox to use the prebuilt VMs.

How to Contact the Author

I welcome feedback from you about this book or about books you'd like to see from me in the future. You can reach me by writing to biju.thomas.sybex@gmail.com. For more information about database administration and Oracle Database 12c certification, please visit my blog at www.bijoos.com/oraclenotes. You may follow me on social media through Twitter (@biju_thomas) and Facebook (Oracle Notes www.facebook.com/oraclenotes).

Sybex strives to keep you supplied with the latest tools and information you need for your work. Please check their website at www.sybex.com, where we'll post additional content, errata, and updates that supplement this book if the need arises. Enter **search terms** in the Search box (or type the book's ISBN— 9781118643952), and click Go to get to the book's update page.

Exam Objectives

Part I: 1Z0-061 Oracle Database 12c: SQL Fundamentals Objective Map

- 1 Introduction Chapter 1
 - 1.1 Describe the features of Oracle Database 12c, Chapter 1
 - 1.2 Describe the salient features of Oracle Cloud 12c, Chapter 1
 - 1.3 Explain the theoretical and physical aspects of a relational database, Chapter 1
 - 1.4 Describe Oracle server's implementation of RDBMS and object relational database management system (ORDBMS), Chapter 1
- 2 Retrieving Data Using the SQL SELECT Statement, Chapter 2
 - 2.1 Explain the capabilities of SQL SELECT statements, Chapter 2
 - 2.2 Execute a basic SELECT statement, Chapter 2
- 3 Restricting and Sorting Data, Chapter 2
 - 3.1 Limit the rows that are retrieved by a query, Chapter 2
 - 3.2 Sort the rows that are retrieved by a query, Chapter 2
 - 3.2 Use ampersand substitution to restrict and sort output at runtime, Chapter 2
- 4 Using Single-Row Functions to Customize Output, Chapter 3
 - 4.1 Describe various types of functions available in SQL, Chapter 3
 - 4.2 Use character, number, and date functions in SELECT statements, Chapter 3
- 5 Using Conversion Functions and Conditional Expressions, Chapter 3
 - 5.1 Describe various types of conversion functions that are available in SQL, Chapter 3
 - 5.2 Use the TO_CHAR, TO_NUMBER, and TO_DATE conversion functions, Chapter 3
 - 5.3 Apply conditional expressions in a SELECT statement, Chapter 3
- 6 Reporting Aggregated Data Using the Group Functions, Chapter 4
 - 6.1 Identify the available group functions, Chapter 4
 - 6.2 Describe the use of group functions, Chapter 4
 - 6.3 Group data by using the GROUP BY clause, Chapter 4
 - 6.4 Include or exclude grouped rows by using the HAVING clause, Chapter 4

- 7 Displaying Data from Multiple Tables Using Joins, Chapter 5
 - 7.1 Write SELECT statements to access data from more than one table using equijoins and nonequijoins, Chapter 5
 - 7.2 Join a table to itself by using a self-join, Chapter 5
 - 7.3 View data that generally does not meet a join condition by using OUTER joins, Chapter 5
 - 7.4 Generate a Cartesian product of all rows from two or more tables, Chapter 5
- 8 Using Subqueries to Solve Queries, Chapter 5
 - 8.1 Define subqueries, Chapter 5
 - 8.2 Describe the types of problems that the subqueries can solve, Chapter 5
 - 8.3 Describe the types of subqueries, Chapter 5
 - 8.4 Write single-row and multiple-row subqueries, Chapter 5
 - 8.5 Using the set operators, Chapter 5
 - 8.6 Describe set operators, Chapter 5
 - 8.7 Use a set operator to combine multiple queries into a single query, Chapter 5
 - 8.8 Control the order of rows returned, Chapter 5
- 9 Managing Tables Using DML Statements, Chapter 6
 - 9.1 Truncate data, Chapter 6
 - 9.2 Insert rows into a table, Chapter 6
 - 9.3 Update rows in a table, Chapter 6
 - 9.4 Delete rows from a table, Chapter 6
 - 9.5 Control transaction, Chapter 6
- 10 Introduction to Data Definition Language, Chapter 7
 - 10.1 Categorize the main database objects, Chapter 7
 - 10.2 Explain the table structure, Chapter 7
 - 10.3 Describe the datatypes that are available for columns, Chapter 7
 - 10.4 Create a simple table, Chapter 7
 - 10.5 Explain how constraints are created at the time of table creation, Chapter 7
 - 10.6 Describe how schema objects work, Chapter 7

Part II: IZO-062 Oracle Database 12c Installation and Administration Exam Objectives

- 1.1 Exploring the Oracle Database Architecture, Chapter 8
 - 1.1.1 List the architectural components of Oracle Database, Chapter 8
 - 1.1.2 Explain the memory structures, Chapter 8

- 1.1.3 Describe the background processes, Chapter 8
- 1.1.4 Explain the relationship between logical and physical storage structures, Chapter 8
- 1.2 Oracle Database Management Tools, Chapter 8
 - 1.2.1 Use database management tools, Chapter 8
- 1.3 Oracle Database Instance, Chapter 9
 - 1.3.1 Understand initialization parameter files, Chapter 9
 - 1.3.2 Start up and shut down an Oracle database instance, Chapter 9
 - 1.3.3 View the alert log and access dynamic performance views, Chapter 9
- 2.1 Oracle Software Installation Basics, Chapter 9
 - 2.1.1 Plan for an Oracle database software installation, Chapter 9
 - 2.3 Installing Oracle Database Software, Chapter 9
 - 2.3.1 Install the Oracle database software, Chapter 9
- 2.4 Creating an Oracle Database Using DBCA, Chapter 9
 - 2.4.1 Create a database by using the Database Configuration Assistant (DBCA), Chapter 9
 - 2.4.2 Generate database creation scripts by using DBCA, Chapter 9
 - 2.4.3 Manage database design template by using DBCA, Chapter 9
 - 2.4.4 Configure database options by using DBCA, Chapter 9
- 1.5 Managing Database Storage Structures, Chapter 10
 - 1.5.1 Describe the storage of table row data in blocks, Chapter 10
 - 1.5.2 Create and manage tablespaces, Chapter 10
- 1.7 Managing Space, Chapter 10
 - 1.7.1 Explain how Oracle database server automatically manages space, Chapter 10
 - 1.7.2 Save space by using compression, Chapter 10
 - 1.7.3 Proactively monitor and manage tablespace space usage, Chapter 10
 - 1.7.4 Use the Segment Advisor, Chapter 10
 - 1.7.5 Reclaim wasted space from tables and indexes by using the segment shrink, Chapter 10
 - 1.7.6 Manage resumable space allocations, Chapter 10
- 1.8 Managing Undo Data, Chapter 11
 - 1.8.1 Explain DML and undo data generation, Chapter 11
 - 1.8.2 Monitor and administer undo data, Chapter 11
 - 1.8.3 Describe the difference between undo data and redo data, Chapter 11
 - 1.8.4 Configure undo retention, Chapter 11

- 1.9 Managing Data Concurrency, Chapter 11
 - 1.9.1. Describe the locking mechanism and how Oracle manages data concurrency, Chapter 11
 - 1.9.2 Monitor and resolve locking conflicts, Chapter 11
- 1.4 Configuring the Oracle Network Environment, Chapter 12
 - 1.4.1 Configure Oracle Net services, Chapter 12
 - 1.4.2 Use tools for configuring and managing the Oracle network, Chapter 12
 - 1.4.3 Configure client-side network, Chapter 12
 - 1.4.4 Understand database resident connection polling, Chapter 12
 - 1.4.5 Configure communication between databases, Chapter 12
- 1.10 Implementing Oracle Database Auditing, Chapter 13
 - 1.10.1 Explain DBA responsibilities for security and auditing, Chapter 13
 - 1.10.2 Enable standard database auditing and Unified Auditing, Chapter 13
- 1.6 Administering User Security, Chapter 13
 - 1.6.1 Create and manage database user accounts, Chapter 13
 - 1.6.2 Grant and revoke privileges, Chapter 13
 - 1.6.3 Create and manage roles, Chapter 13
 - 1.6.4 Create and manage profiles, Chapter 13
- 1.16. Performing Database Maintenance, Chapter 14
 - 1.16.1 Manage the Automatic Workload Repository (AWR), Chapter 14
 - 1.16.2 Use the Automatic Database Diagnostic Monitor (ADDM), Chapter 14
 - 1.16.3 Describe and use the advisory framework, Chapter 14
 - 1.16.4 Set alert thresholds, Chapter 14
 - 1.16.5 Use automated tasks, Chapter 14
- 1.17 Managing Performance, Chapter 14
 - 1.17.1 Use Enterprise Manager to monitor performance, Chapter 14
 - 1.17.2 Use Automatic Memory Management, Chapter 14
 - 1.17.3 Use the Memory Advisor to size memory buffers, Chapter 14
- 1.18 Managing Performance SQL Tuning, Chapter 14
 - 1.18.1 Manage optimizer statistics, Chapter 14
 - 1.18.2 Use the SQL Tuning Advisor, Chapter 14
 - 1.18.3 Use the SQL Access Advisor to tune a workload, Chapter 14
- 1.19 Managing Resources Using Database Resource Manager, Chapter 16
 - 1.19.1 Configure the Database Resource Manager, Chapter 16
 - 1.19.2 Access and create resource plans, Chapter 16
 - 1.19.3 Monitor the Resource Manager, Chapter 16

- 1.20 Automating Tasks by Using Oracle Scheduler, Chapter 16
 - 1.20.1 Use Oracle Scheduler to simplify management tasks, Chapter 16
 - 1.20.2 Use job chains to perform a series of related tasks, Chapter 16
 - 1.20.3 Use Scheduler jobs on remote systems, Chapter 16
 - 1.20.4 Use advanced Scheduler features to prioritize jobs, Chapter 16
- 1.11 Backup and Recovery Concepts, Chapter 15
 - 1.11.1 Identify the importance of checkpoints, redo log files, and archive log files, Chapter 15
- 1.12 Backup and Recovery Configuration, Chapter 15
 - 1.12.1 Configure the fast recovery area, Chapter 15
 - 1.12.1 Configure ARCHIVELOG mode, Chapter 15
- 1.13 Performing Database Backups, Chapter 15
 - 1.13.1 Create consistent database backups, Chapter 15
 - 1.13.2 Back up your database without shutting it down, Chapter 15
 - 1.13.3 Create incremental backups, Chapter 15
 - 1.13.4 Automate database backups, Chapter 15
 - 1.13.5 Manage backups, Chapter 15
- 1.14 Performing Database Recovery, Chapter 15
 - 1.14.1 Determine the need for performing recovery, Chapter 15
 - 1.14.2 Use Recovery Manager (RMAN) and the Data Recovery Advisor to perform recovery of the control file, redo log file, and data file, Chapter 15
- 2.6 Upgrading Database Software, Chapter 17
 - 2.6.1 Describe upgrade methods, Chapter 17
 - 2.6.2 Describe data migration methods, Chapter 17
 - 2.6.3 Describe the upgrade process, Chapter 17
- 2.7 Preparing to Upgrade to Oracle Database 12c, Chapter 17
 - 2.7.1 Describe upgrade requirements when certain features or options are used in Oracle database, Chapter 17
 - 2.7.2 Use the Pre-Upgrade Information tool before performing an upgrade, Chapter 17
 - 2.7.3 Prepare the new Oracle home prior to performing an upgrade, Chapter 17
- 2.8 Upgrading to Oracle Database 12c, Chapter 17
 - 2.8.1 Upgrade the database to Oracle Database 12c by using the Database Upgrade Assistant (DBUA), Chapter 17
 - 2.8.2 Perform a manual upgrade to Oracle Database 12c by using scripts and tools, Chapter 17

- 2.9 Performing Post-Upgrade Tasks, Chapter 17
 - 2.9.1 Migrate to Unified Auditing, Chapter 17
 - 2.9.2 Perform post-upgrade tasks, Chapter 17
- 1.15 Moving Data, Chapter 18
 - 1.15.1 Describe ways to move data, Chapter 18
 - 1.15.2 Use SQL*Loader to load data from a non-Oracle database, Chapter 18
 - 1.15.3 Use external tables to move data via platform-independent files, Chapter 18
 - 1.15.4 Use Data Pump Export and Import to move data between Oracle databases, Chapter 18
- 2.10 Migrating Data by Using Oracle Data Pump, Chapter 18
 - 2.10.1 Migrate data by using Oracle Data Pump, Chapter 18
- 2.2 Installing Oracle Grid Infrastructure for a Standalone Server, Chapter 18
 - 2.2.1 Configure storage for Oracle Automatic Storage Management (ASM), Chapter 18
 - 2.2.2 Install Oracle Grid Infrastructure for a standalone server, Chapter 18
- 2.5 Using Oracle Restart, Chapter 18
 - 2.5.1 Use Oracle Restart to manage components, Chapter 18

Assessment Test

Exam 1Z0-061: SQL Fundamentals

1. Which operator will be evaluated first in the following SELECT statement?

```
SELECT (2+3*4/2-5) FROM dual;
```

- A. +
- B. *
- C. /
- D. -

2. John wants to remove the values present in column SALARY in the EMPLOYEES table for all employees who belong to DEPARTMENT_ID 90. Which SQL would accomplish the task?

- A. DELETE FROM EMPLOYEES (SALARY) WHERE DEPARTMENT_ID = 90;
- B. INSERT INTO EMPLOYEES (SALARY) VALUES (NULL) WHERE DEPARTMENT_ID = 90;
- C. UPDATE EMPLOYEES SET SALARY = NULL WHERE DEPARTMENT_ID = 90;
- D. MERGE EMPLOYEES SET SALARY IS NULL WHERE DEPARTMENT_ID = 90;

3. Which function can possibly return a non-NULL value when one of the arguments is NULL?

- A. NULLIF
- B. LENGTH
- C. CONCAT
- D. INSTR
- E. TAN

4. The following statement will raise an exception on which line?

```
select dept_name, avg(all salary)
       ,count(*) "number of employees"
  from emp , dept
 where emp.deptno = dept.dept_no
       and count(*) > 5
 group by dept_name
 order by 2 desc;
```

- A. select dept_name, avg(all salary), count(*) "number of employees"
- B. where emp.deptno = dept.dept_no
- C. and count(*) > 5

- D. group by dept_name
E. order by 2 desc;
5. Review the code segment.
1. INSERT INTO salaries VALUES (101, 23400, SYSDATE);
 2. UPDATE salaries
 3. SET salary = salary * 1.1
 4. AND effective_dt = SYSDATE
 5. WHERE empno = 333;

Which line has an error?

- A. 2
- B. 4
- C. 5
- D. No error

6. Review the following SQL and choose the most appropriate option.

```
SELECT job_id, COUNT(*)  
FROM employees  
GROUP BY department_id;
```

- A. The statement will show the number of jobs in each department.
 - B. The statement will show the number of employees in each department.
 - C. The statement will generate an error.
 - D. The statement will work if the GROUP BY clause is removed.
7. Which datatype stores data outside the Oracle database?
- A. UROWID
 - B. BFILE
 - C. BLOB
 - D. NCLOB
 - E. EXTERNAL
8. The DEPT table has the following data.

```
SQL> SELECT * FROM dept;  
DEPTNO DNAME          LOC  
-----  
      10 ACCOUNTING    NEW YORK  
      20 RESEARCH       DALLAS  
      30 SALES          CHICAGO  
      40 OPERATIONS     BOSTON
```

Consider this INSERT statement:

```
INSERT INTO (SELECT * FROM dept WHERE deptno = 10)
VALUES (50, 'MARKETING', 'FORT WORTH');
```

Choose the best answer.

- A. The INSERT statement is invalid; a valid table name is missing.
 - B. 50 is not a valid DEPTNO value, since the subquery limits DEPTNO to 10.
 - C. The statement will work without error.
 - D. A subquery and a VALUES clause cannot appear together.
9. Which two of the following queries is valid syntax that would return all rows from the EMPLOYEES and DEPARTMENTS tables, even if there are no corresponding/related rows in the other table?
- A. `SELECT last_name, first_name, department_name FROM employees e FULL JOIN departments d ON e.department_id = d.department_id;`
 - B. `SELECT last_name, first_name, department_name FROM employees e OUTER JOIN departments d ON e.department_id = d.department_id;`
 - C. `SELECT e.last_name, e.first_name, d.department_name FROM employees e LEFT OUTER JOIN departments d ON e.department_id = d.department_id RIGHT OUTER JOIN employees f ON f.department_id = d.department_id;`
 - D. `SELECT e.last_name, e.first_name, d.department_name FROM employees e CROSS JOIN departments d ON e.department_id = d.department_id;`
 - E. `SELECT last_name, first_name, department_name FROM employees FULL OUTER JOIN departments USING (department_id);`
10. Which of the following statements could use an index on the columns PRODUCT_ID and WAREHOUSE_ID of the OE.INVENTORIES table?
- A. `select count(distinct warehouse_id) from oe.inventories;`
 - B. `select product_id, quantity_on_hand from oe.inventories where product_id = 100;`
 - C. `insert into oe.inventories values (5,100,32);`
 - D. None of these statements could use the index.
11. The following statements are executed:

```
create sequence my_seq;
select my_seq.nextval from dual;
select my_seq.nextval from dual;
rollback;
select my_seq.nextval from dual;
```

What will be selected when the last statement is executed?

- A.** 0
 - B.** 1
 - C.** 2
 - D.** 3
 - E.** NULL
- 12.** Which of the following statements are true? (Choose two.)
- A.** Primary key constraints allow NULL values in the columns.
 - B.** Unique key constraints allow NULL values in the columns.
 - C.** Primary key constraints do not allow NULL values in the columns.
 - D.** A non-unique index cannot be used to enforce primary key constraints.
- 13.** The current time in Dubai is “04-APR-2013 08:50:00” and the time in Dallas is “03-APR-2013 23:50:00”. A user from Dubai is connected to a session in the database located on a server in Dallas. What will be the result of his query?
- SELECT TO_CHAR(SYSDATE, 'DD-MON-YYYY HH24:MI:SS') FROM dual;
- A.** 04-APR-2013 08:50:00
 - B.** 03-APR-2013 8 23:50:00
 - C.** 03-APR-2013 2324:50:00
 - D.** None of the above
- 14.** The FIRED_EMPLOYEES table has the following structure.

```
EMPLOYEE_ID NUMBER (4)
FIRE_DATE DATE
```

How many rows will be counted from the last SQL statement in the code segment?

```
SELECT COUNT(*) FROM FIRED_EMPLOYEES;
COUNT(*)
-----
105

INSERT INTO FIRED_EMPLOYEES VALUES (104, TRUNC(SYSDATE));
SAVEPOINT A;
INSERT INTO FIRED_EMPLOYEES VALUES (106, TRUNC(SYSDATE));
SAVEPOINT B;
INSERT INTO FIRED_EMPLOYEES VALUES (108, TRUNC(SYSDATE));
ROLLBACK TO A;
```

```
INSERT INTO FIRED_EMPLOYEES VALUES (104, TRUNC(SYSDATE));
COMMIT;
SELECT COUNT(*) FROM FIRED_EMPLOYEES;
```

- A. 109
B. 106
C. 105
D. 107
15. At a minimum, how many join conditions should there be to avoid a Cartesian join if there are three tables in the FROM clause?
- A. 1
B. 2
C. 3
D. There is no minimum.
16. Why does the following statement fail?
- ```
CREATE TABLE FRUITS-N-VEGETABLES
(NAME VARCHAR2 (40));
```
- A. The table should have more than one column in its definition.  
B. NAME is a reserved word, which cannot be used as a column name.  
C. The table name is invalid.  
D. Column length cannot exceed 30 characters.
17. Which two statements are true about NULL values?
- A. You cannot search for a NULL value in a column using the WHERE clause.  
B. If a NULL value is returned in the subquery or if NULL is included in the list when using a NOT IN operator, no rows will be returned.  
C. Only = and != operators can be used to search for NULL values in a column.  
D. In an ascending order sort, NULL values appear at the bottom of the result set.  
E. Concatenating a NULL value to a non-NULL string results in a NULL.
18. Table *CUSTOMERS* has a column named *CUST\_ZIP* which could be NULL. Which of the following functions include the NULL rows in its result?
- A. COUNT (CUST\_ZIP)  
B. SUM (CUST\_ZIP)  
C. AVG (DISTINCT CUST\_ZIP)  
D. None of the above

- 19.** Using the following EMP table, you need to increase everyone's salary by 5 percent of their combined salary and bonus. Which of the following statements will achieve the desired results?

| Column Name     | emp_id   | name     | salary | bonus  |
|-----------------|----------|----------|--------|--------|
| Key Type        | pk       | pk       |        |        |
| NULLs/Unique    | NN       | NN       | NN     |        |
| <b>FK Table</b> |          |          |        |        |
| Datatype        | VARCHAR2 | VARCHAR2 | NUMBER | NUMBER |
| Length          | 9        | 50       | 11,2   | 11,2   |

- A.** UPDATE emp SET salary = (salary + bonus)\*1.05;
- B.** UPDATE emp SET salary = salary\*1.05 + bonus\*1.05;
- C.** UPDATE emp SET salary = salary + (salary + bonus)\*0.05;
- D.** A, B, and C will achieve the desired results.
- E.** None of these statements will achieve the desired results.

- 20.** Which option is not available in Oracle when modifying tables?

- A.** Add new columns
- B.** Rename existing column
- C.** Drop existing column
- D.** None of the above

- 21.** The following data is from the EMPLOYEES table:

| DEPARTMENT_ID | EMPNO | FIRST_NAME |
|---------------|-------|------------|
| 30            | 119   | Karen      |
| 50            | 124   | Kevin      |
| 50            | 135   | Ki         |
| 80            | 146   | Karen      |
|               | 178   | Kimberely  |
| 50            | 188   | Kelly      |
| 50            | 197   | Kevin      |

Which row (empno) will be returned last when the following query is executed?

```
select department_id, employee_id empno, first_name
from employees
```

order by 1, 2

- A. 188
  - B. 178
  - C. 146
  - D. 119
- 22.** INTERVAL datatypes store a period of time. Which components are included in the INTERVAL DAY TO SECOND datatype column? (Choose all that apply.)
- A. Years
  - B. Quarters
  - C. Months
  - D. Days
  - E. Hours
  - F. Minutes
  - G. Seconds
  - H. Fractional seconds
- 23.** Which components are not part of the easy connect connection string? (Choose two.)
- A. hostname
  - B. service name
  - C. database sid
  - D. port number
  - E. network protocol
- 24.** The table CUSTOMERS has the following data:

| ID   | NAME     | ZIP   | UPD_DATE  |
|------|----------|-------|-----------|
| L921 | LEEZA    | 75252 | 01-JAN-00 |
| B023 | WILLIAMS | 15215 |           |
| K783 | KATHY    | 75252 | 15-FEB-00 |
| B445 | BENJAMIN | 76021 | 15-FEB-00 |
| D334 | DENNIS   | 12443 |           |

You issue the following command to alter the table:

1. ALTER TABLE CUSTOMERS
2. MODIFY
3. (UPD\_DATE DEFAULT SYSDATE NOT NULL,
4. ZIP NOT NULL);

Which line of code will cause an error?

- A.** Line 2
  - B.** Line 3
  - C.** Line 4
  - D.** There will be no error.
- 25.** In ANSI SQL, a self-join can be represented by using which of the following? (Choose the best answer.)
- A.** NATURAL JOIN clause
  - B.** CROSS JOIN clause
  - C.** JOIN ... USING clause
  - D.** JOIN ... ON clause
  - E.** All of the above
- 26.** What will be result of `trunc(2916.16, -1)`?
- A.** 2916.2
  - B.** 290
  - C.** 2916.1
  - D.** 2900
  - E.** 2910
- 27.** The table ADDRESSES is created using the following syntax. How many indexes will be created automatically when this table is created?

```
CREATE TABLE ADDRESSES (
 NAME VARCHAR2 (40) PRIMARY KEY,
 STREET VARCHAR2 (40),
 CITY VARCHAR2 (40),
 STATE CHAR (2) REFERENCES STATE (ST_CODE),
 ZIP NUMBER (5) NOT NULL,
 PHONE VARCHAR2 (12) UNIQUE);
```

- A.** 0
  - B.** 1
  - C.** 2
  - D.** 3
- 28.** Which line of the following code has an error?

```
SELECT *
FROM emp
```

```
WHERE comm = NULL
ORDER BY ename;
```

- A. SELECT \*
  - B. FROM emp
  - C. WHERE comm = NULL
  - D. There is no error in this statement.
29. How do you represent the following business rule in an ER diagram? “A customer may have one or more orders; an order must belong to one and only one customer.”
- A. Single solid line.
  - B. Single line that is solid at one end and dotted at another end.
  - C. Single solid line with a crowfoot at one end.
  - D. The business rule cannot be represented in the ER diagram.
30. What order does Oracle use in resolving a table or view referenced in a SQL statement?
- A. Table/view within user’s schema, public synonym, private synonym
  - B. Table/view within user’s schema, private synonym, public synonym
  - C. Public synonym, table/view within user’s schema, private synonym
  - D. Private synonym, public synonym, table/view within user’s schema
31. Which two options are not true when you execute a COMMIT statement?
- A. All locks created by DML statements are released in the session.
  - B. All savepoints created are erased in the session.
  - C. Queries started before the COMMIT in other sessions will show the current changes after COMMIT.
  - D. All undo information written from the DML statements are erased.
32. Which two operators are used to add more joining conditions in a multiple-table query?
- A. NOT
  - B. OR
  - C. AND
  - D. Comma (,)
33. What is wrong with the following SQL?

```
SELECT department_id, MAX(COUNT(*))
FROM employees
GROUP BY department_id;
```

- A. Aggregate functions cannot be nested.
  - B. GROUP BY clause should not be included when using nested aggregate functions.
  - C. The department\_id column in the SELECT clause should not be used when using nested aggregate functions.
  - D. When using COUNT function cannot be nested.
- 34.** Which types of constraints can be created on a view?
- A. Check, NOT NULL
  - B. Primary key, foreign key, unique key
  - C. Check, NOT NULL, primary key, foreign key, unique key
  - D. No constraints can be created on a view.
- 35.** Which two declarations define the maximum length of a CHAR datatype column in bytes?
- A. CHAR (20)
  - B. CHAR (20) BYTE
  - C. CHAR (20 BYTE)
  - D. BYTE (20 CHAR)
  - E. CHAR BYTE (20)
- 36.** Which SELECT statement clauses can be used to limit the rows returned (say, you want to display the rows 6 through 15). (Choose two.)
- A. WHERE
  - B. OFFSET
  - C. FETCH
  - D. FILTER
- 37.** You query the database with the following:

```
SELECT PRODUCT_ID FROM PRODUCTS
WHERE PRODUCT_ID LIKE '%S_J_C' ESCAPE '\';
```

Choose the PRODUCT\_ID strings from the options that will satisfy the query. (Choose two.)

- A. BTS\_J\_C
- B. SJC
- C. SKJKC
- D. S\_J\_C

**38.** The EMPLOYEE table is defined as follows:

```
EMP_NAME VARCHAR2(40)
HIRE_DATE DATE
SALARY NUMBER (14,2)
```

Which query is most appropriate to use if you need to find the employees who were hired before 01-Jan-1998 and have a salary above 5,000 or below 1,000?

- A. SELECT emp\_name FROM employee WHERE hire\_date > TO\_DATE('01011998','MMDDYYYY') AND SALARY < 1000 OR > 5000;
- B. SELECT emp\_name FROM employee WHERE hire\_date < TO\_DATE('01011998','MMDDYYYY') AND SALARY < 1000 OR SALARY > 5000;
- C. SELECT emp\_name FROM employee WHERE hire\_date < TO\_DATE('01011998','MMDDYYYY') AND (SALARY < 1000 OR SALARY > 5000);
- D. SELECT emp\_name FROM employee WHERE hire\_date < TO\_DATE('01011998','MMDDYYYY') AND SALARY BETWEEN 1000 AND 5000;

**39.** What happens when you issue the following command? (Choose all correct answers.)

```
TRUNCATE TABLE SCOTT.EMPLOYEE;
```

- A. All the rows in the table EMPLOYEE owned by SCOTT are removed.
  - B. The storage space used by the table EMPLOYEE is released (except the initial extent).
  - C. If foreign key constraints are defined to this table using the ON DELETE CASCADE clause, the rows from the child tables are also removed.
  - D. The indexes on the table are dropped.
  - E. You cannot truncate a table if triggers are defined on the table.
- 40.** Which two statements will drop the primary key defined on table EMP. The primary key name is PK\_EMP.
- A. ALTER TABLE EMP DROP PRIMARY KEY;
  - B. DROP CONSTRAINT PK\_EMP;
  - C. ALTER TABLE EMP DROP CONSTRAINT PK\_EMP;
  - D. ALTER CONSTRAINT PK\_EMP DROP CASCADE;
  - E. DROP CONSTRAINT PK\_EMP ON EMP;

## Exam 1Z0-062: Installation and Administration

1. Which database version cannot be upgraded directly to Oracle Database 12c?
  - A. 11.2.0.2
  - B. 11.2.0.1
  - C. 11.1.0.7
  - D. 10.2.0.5
  - E. All of the above
2. The following steps might be related to relocating a data file belonging to the USERS tablespace. Pick the steps that are required for relocating the data file from /disk1 to /disk2.
  - A. Copy the file '/disk1/users01.dbf' to '/disk2/users01.dbf' using an OS command.
  - B. ALTER DATABASE MOVE DATAFILE '/disk1/users01.dbf' to '/disk2/users01.dbf'
  - C. ALTER DATABASE RENAME FILE '/disk1/users01.dbf' to '/disk2/users01.dbf'
  - D. ALTER TABLESPACE USERS OFFLINE
  - E. ALTER TABLESPACE USERS ONLINE
3. Which of the following is not considered part of an Oracle database?
  - A. Data files
  - B. Redo logs
  - C. The pfile and spfile
  - D. Control files
4. The highest level at which a user can request a lock is the \_\_\_\_\_ level.
  - A. Schema
  - B. Table
  - C. Row
  - D. Block
5. To grant the SELECT privilege on the table HR.CUSTOMERS to all users in the database, which statement would you use?
  - A. GRANT SELECT ON HR.CUSTOMERS TO ALL USERS;
  - B. GRANT SELECT ON HR.CUSTOMERS TO ALL;
  - C. GRANT SELECT ON HR.CUSTOMERS TO ANONYMOUS;
  - D. GRANT SELECT ON HR.CUSTOMERS TO PUBLIC;

6. Which query can be used to find and categorize all chained jobs (only chained jobs), and includes any remote steps as applied to those chained jobs, as executed from the current database? (Choose the best answers, more than one if appropriate.)
- A. select table\_name from dictionary where table\_name like 'DBA\_%SCHED%' or table\_name like 'DBA\_%REMOTE%';
  - B. SELECT TABLE\_NAME FROM DICTIONARY WHERE TABLE\_NAME LIKE 'DBA\_%CHAIN%' OR TABLE\_NAME LIKE 'DBA\_%REMOTE%';
  - C. select table\_name from dictionary where table\_name like 'DBA\_%CHAIN%' or table\_name like 'DBA\_%REMOTE%';
  - D. select table\_name from dictionary where table\_name like 'DBA\_%SCHED%' or table\_name like 'DBA\_%CHAIN%';
  - E. None of the above
7. The Automatic Workload Repository (AWR) is primarily populated with performance statistics by which Oracle Database 12c background process?
- A. MMNL
  - B. QMN1
  - C. MMON
  - D. MMAN
8. The initialization parameter RESUMABLE\_TIMEOUT is set to 600. Which other statements are the minimum required to enable a resumable session?
- A. No other setup is required if the RESUMABLE\_TIMEOUT is a nonzero value.
  - B. ALTER SESSION ENABLE RESUMABLE
  - C. ALTER SESSION ENABLE RESUMABLE TIMEOUT 0
  - D. ALTER SESSION BEGIN RESUMABLE SESSION
9. Which utility is used to install Oracle Database 12c?
- A. DBCA
  - B. OUI
  - C. runInstaller
  - D. Oracle Database 12c is installed using the URL to [www.oracle.com](http://www.oracle.com).
10. You have just made changes to the `listener.ora` file for the listener called `listener1` using Oracle Net Manager. Which of the following commands or combinations of commands would you use to put the changes into effect with the least amount of client disruption?
- A. lsnrctl stop listener1 followed by lsnrctl start listener1
  - B. lsrnctl restart listener1
  - C. lsnrctl reload listener1
  - D. lsnrctl cycle services

- 11.** If a job is not running and a STOP\_JOB procedure is executed, what will happen? (Choose the best answer.)
- A.** An error will not occur.
  - B.** The job will stop running.
  - C.** The job will be dropped altogether.
  - D.** All of the above.
  - E.** None of the above.
- 12.** Where does Oracle Database 12c record all the changes made to the database that can be used for recovery operations?
- A.** Control files
  - B.** Redo log files
  - C.** Alert log file
  - D.** Parameter file
- 13.** What is accomplished when you issue the following statement?
- ```
ALTER USER JOHN DEFAULT ROLE ALL;
```
- A.** John is assigned all the roles created in the database.
 - B.** Any existing roles remain the same, but any future roles created will be enabled.
 - C.** All of John's roles are enabled except the roles with passwords.
 - D.** All of John's roles are enabled including the roles with passwords.
- 14.** Which activity is a must do before upgrading a database to 12c using the manual upgrade method?
- A.** Run dbms_stats.gather_dictionary_stats.
 - B.** Run \$ORACLE_HOME/rdbms/admin/emremove.sql.
 - C.** Run dbms_stats.gather_database_stats.
 - D.** Run preupgrd.sql script.
- 15.** What would you do to reduce the time required to start the instance after a database crash?
- A.** Multiplex the redo log files
 - B.** Increase the size of the redo log files
 - C.** Set FAST_START_MTTR_TARGET parameter to 0
 - D.** All of the above
 - E.** None of the above

- 16.** When you are configuring Oracle Shared Server, which initialization parameter would you likely need to modify?
- A.** DB_CACHE_SIZE
 - B.** DB_BLOCK_BUFFERS
 - C.** LARGE_POOL_SIZE
 - D.** BUFFER_SIZE
 - E.** None of the above
- 17.** Which options are available in DBCA to configure recovery-related operations?
- A.** Data Guard
 - B.** Standby Database
 - C.** Fast Recovery Area
 - D.** Archiving
- 18.** When you started the Oracle Database 12c database, you got an ORA-01157: cannot identify data file ... error. After invoking RMAN, which command would you first use before performing the REPAIR FAILURE?
- A.** RECOVER FAILURE
 - B.** ADVISE FAILURE
 - C.** LIST FAILURE
 - D.** CHANGE FAILURE
- 19.** Which of the following commands is most likely to generate an error message?
(Choose two.)
- A.** ALTER SYSTEM SET UNDO_MANAGEMENT=AUTO SCOPE=MEMORY;
 - B.** ALTER SYSTEM SET UNDO_MANAGEMENT=AUTO SCOPE=SPFILE;
 - C.** ALTER SYSTEM SET UNDO_MANAGEMENT=MANUAL SCOPE=MEMORY;
 - D.** ALTER SYSTEM SET UNDO_MANAGEMENT=MANUAL SCOPE=SPFILE;
 - E.** ALTER SYSTEM SET UNDO_TABLESPACE=RBS1 SCOPE=BOTH;
- 20.** You performed a SHUTDOWN ABORT on the database. What happens when you issue the STARTUP command?
- A.** Startup will fail, because you have not completed the instance recovery.
 - B.** Oracle automatically performs recovery. All committed changes are written to data files.
 - C.** During instance recovery, you have the option to selectively commit uncommitted transactions.
 - D.** After the database starts, you have to manually clean out uncommitted transactions from the transaction table.

21. Where does the following procedure allow jobs to be created?

```
BEGIN  
  DBMS_SCHEDULER.CREATE_GROUP(group_name=>'things'  
    ,group_type=>'DB_DEST' ,member=>'LOCAL');  
END;  
/
```

- A.** On a remote server as a database destination
 - B.** On a remote server using a hostname and IP address
 - C.** On a local database server only
 - D.** On a local server using a hostname and an IP address
 - E.** None of the above
- 22.** Which of the following statements is not always true? (Choose two.)
- A.** Every database should have at least two tablespaces.
 - B.** Every database should have at least two data files.
 - C.** Every database should have at least three multiplexed redo logs.
 - D.** Every database should have at least three control files.
- 23.** How can you prevent someone from using an all-alphabet password?
- A.** Set the initialization parameter PASSWORD_COMPLEXITY to ALPHANUM.
 - B.** Alter that user's profile setting PASSWORD_COMPLEXITY to ALPHNANUM.
 - C.** Alter the user's profile to use a password-verify function that performs comparisons to validate the password.
 - D.** There is no mechanism that lets you prevent an all-alphabet password.
- 24.** Which metadata view can be used to find Oracle Scheduler priorities?
- A.** DBA_CHAINS
 - B.** DBA_SCHEDULER_CHAINS
 - C.** DBA_SCHEDULER_JOBS
 - D.** DBA_SCHEDULER_DESTS
 - E.** None of the above
- 25.** Which component of the SGA has the dictionary cache?
- A.** Buffer cache
 - B.** Library cache
 - C.** Shared pool

- D. Program global area
 - E. Large pool
 - F. Result cache
26. Which of the following advisors is used to determine if the database read-consistency mechanisms are properly configured?
- A. Undo Management Advisor
 - B. SQL Access Advisor
 - C. SQL Tuning Advisor
 - D. Memory Advisor
27. Which storage parameter is used to make sure that each extent is a multiple of the value specified?
- A. MINEXTENTS
 - B. INITIAL
 - C. MINIMUM EXTENT
 - D. MAXEXTENTS
28. What is (are) the name of the audit trail view(s) where audit records from Unified Auditing is (are) stored, when database auditing, RMAN, and Data Pump auditing are enabled?
- A. UNIFIED_AUDIT_TRAIL
 - B. UNIFIED_AUDIT_TRAIL, COMPONENT_AUDIT_TRAIL
 - C. SYS.AUD\$
 - D. AUDIT_TRAIL_DB, AUDIT_TRAIL_DP, AUDIT_TRAIL_RMAN
29. Which of the following is the utility that you can use to test the network connections across TCP/IP?
- A. trcasst
 - B. lsnrctl
 - C. namesctl
 - D. ping
 - E. None of the above
30. Undo data in an undo tablespace is not used for which of the following purposes?
- A. Providing users with read-consistent queries
 - B. Rolling forward after an instance failure
 - C. Flashback queries
 - D. Recovering from a failed transaction
 - E. Restoring original data when a ROLLBACK is issued

- 31.** In which file could you tell SQL*Loader to use the direct path option?
- A.** Log file
 - B.** Parfile
 - C.** Bad file
 - D.** Data file
- 32.** Which of the following is false about shared servers?
- A.** Shared servers can process requests from many users.
 - B.** Shared servers receive their requests directly from dispatchers.
 - C.** Shared servers place completed requests on a dispatcher response queue.
 - D.** The SHARED_SERVERS parameter configures the number of shared servers to start at instance startup.
- 33.** Identify the statement that is not true about checkpoints.
- A.** Instance recovery is complete when the data from the last checkpoint up to the latest SCN in the control file has been written to the data files.
 - B.** A checkpoint keeps track of what has already been written to the data files.
 - C.** The redo log group writes must occur before a commit complete is returned to the user.
 - D.** The distance between the checkpoint position in the redo log file and the end of the redo log group can never be more than 90 percent of the size of the largest redo log group.
 - E.** How much the checkpoint lags behind the SCN is controlled by both the size of the redo log groups and by setting the parameter FAST_START_MTTR_TARGET.
- 34.** Which parameter is used to enable the Automatic Memory Management feature of the Oracle database?
- A.** MEMORY_MANAGEMENT
 - B.** MEMORY_TARGET
 - C.** SGA_TARGET
 - D.** MEMORY_SIZE
- 35.** When performing a Data Pump Import using impdp, which of the following options is not a valid value to the TABLE_EXISTS_ACTION parameter?
- A.** SKIP
 - B.** APPEND
 - C.** TRUNCATE
 - D.** RECREATE

- 36.** Two tablespaces are critical to the database. The loss of a data file in these tablespaces requires an instance shutdown to recover. Which are these tablespaces?
- A.** TEMP
 - B.** SYSTEM
 - C.** UNDO
 - D.** SYSAUX
- 37.** Which initialization parameter determines the location of the alert log file?
- A.** DIAGNOSTIC_DEST
 - B.** BACKGROUND_DUMP_DEST
 - C.** ALERT_LOG_DEST
 - D.** USER_DUMP_DEST
- 38.** Which statement regarding a global temporary table and redo generation is true?
- A.** Global temporary tables generate the same amount of redo as persistent tables.
 - B.** Global temporary tables generate redo from undo when a DML operation is performed on them.
 - C.** By setting the TEMP_UNDO_ENABLED to TRUE, you turn off the undo generation for DML on temporary tables.
 - D.** All of the above are true.
- 39.** Which database management tools are automatically installed and configured when Oracle Database 12c database is configured? (Choose two.)
- A.** OEM Database Express
 - B.** OEM Cloud Control
 - C.** SQL*Plus
 - D.** SQL Developer
- 40.** Which parameter is used to set up the directory for Oracle to create data files, if the DATAFILE clause does not specify a filename when you're creating or altering tablespaces?
- A.** DB_FILE_CREATE_DEST
 - B.** DB_CREATE_FILE_DEST
 - C.** DB_8K_CACHE_SIZE
 - D.** USER_DUMP_DEST
 - E.** DB_CREATE_ONLINE_LOG_DEST_1

- 41.** In Oracle Database 12c, which script is recommended to run for upgrading a database from 11.1.0.7?
- A.** catupgrd.sql
 - B.** utlu121i.sql
 - C.** catctl.pl
 - D.** utlu121s.sql
- 42.** How do you enable Database Resident Connection Pooling (DRCP)?
- A.** Using listener.ora configuration.
 - B.** Using Oracle Net Manager configuration.
 - C.** Using a PL/SQL package.
 - D.** Depends on the application. PHP, HTTP, and SQL*Net applications have different configurations.
- 43.** Which initialization parameter determines the location of the alert log file?
- A.** LOG_ARCHIVE_DEST
 - B.** USER_DUMP_DEST
 - C.** BACKGROUND_DUMP_DEST
 - D.** DIAGNOSTIC_DEST
- 44.** Which product do you install to configure an Oracle Automatic Storage Management instance?
- A.** Oracle Database 12c database
 - B.** Oracle Database 12c Grid Infrastructure
 - C.** Oracle Database 12c Automatic Storage Management
 - D.** Oracle Restart

Answers to Assessment Test

Exam 1Z0-061: SQL Fundamentals

1. B. In the arithmetic operators, unary operators are evaluated first, then multiplication and division, and finally addition and subtraction. The expression is evaluated from left to right. See Chapter 2 for more information.
2. C. Except for option C, all other SQL statements are not valid syntactically. To change a value in a column for existing rows, you need to use the UPDATE statement. The DELETE statement deletes the entire row. The INSERT statement adds a new row. The MERGE statement performs kind of an upsert—it inserts if the row does not exist, and updates if the row exists. For more information, read Chapter 6.
3. C. CONCAT will return a non-NUL if only one parameter is NULL. Both CONCAT parameters would need to be NULL for CONCAT to return NULL. The NULLIF function returns NULL if the two parameters are equal. The LENGTH of a NULL is NULL. INSTR will return NULL if NULL is passed in, and the tangent (TAN) of a NULL is NULL. See Chapter 3 to learn more.
4. C. Group functions cannot appear in the WHERE clause. Read Chapter 4 to learn more about group functions.
5. B. When using multiple columns to update in a single UPDATE statement, the column assignments in the SET clause must be separated by a comma, not an AND operator. See Chapter 6 for more information.
6. C. Since job_id is used in the SELECT clause, it must be used in the GROUP BY clause also. For more information, see Chapter 4.
7. B. UROWID datatype is used to store ROWID data. The BFILE datatype stores only the locator to an external file in the database; the actual data is stored as an operating system file. BLOB, NCLOB, and CLOB are other Large Object data types in Oracle Database 12c. EXTERNAL is not a valid datatype. Read Chapter 7 for more information.
8. C. The statement will work without error. Option B would be correct if you used the WITH CHECK OPTION clause in the subquery. See Chapter 5 for more information about subqueries.
9. A, E. An outer join on both tables can be achieved using the FULL OUTER JOIN syntax. The join condition can be specified by using the ON clause to specify the columns explicitly or using the USING clause to specify columns with common column names. Options B and D would result in errors. In option B, the join type is not specified; OUTER is an optional keyword. In option D, CROSS JOIN is used to get a Cartesian result, and Oracle does not expect a join condition. See Chapter 5 for more information.

10. A, B. The index contains all the information needed to satisfy the query in option A, and a full-index scan would be faster than a full-table scan. A subset of indexes columns is specified in the WHERE clause of option B, hence Oracle database can use the index. Read Chapter 7 to learn more about using an index.
11. D. The CREATE SEQUENCE statement will create an increasing sequence that will start with 1, increment by 1, and be unaffected by the rollback. A rollback will never stuff values back into a sequence. Read Chapter 7 for more information.
12. B, C. Primary key and unique key constraints can be enforced using non-unique indexes. Unique constraints allow NULL values in the columns, but primary keys do not. See Chapter 7 for more information.
13. B. The SYSDATE function returns the date and time on the server where the database instance is started. CURRENT_DATE returns local date and time. See Chapter 3 for more information.
14. D. The first INSERT statement and last INSERT statement will be saved in the database. The ROLLBACK TO A statement will undo the second and third inserts. For more information, see Chapter 6.
15. B. There should be at least $n-1$ join conditions when joining n tables to avoid a Cartesian join. See Chapter 5 for more information.
16. C. The table and column names can include only three special characters: #, \$, and _. No other characters are allowed in the table name. You can have letters and numbers in the table name. Read Chapter 7 for more information.
17. B, D. You can use the IS NULL or IS NOT NULL operator to search for NULLs or non-NULLs in a column. Since NULLs are sorted higher, they appear at the bottom of the result set in an ascending order sort. See Chapter 2 for more information.
18. D. COUNT (<column_name>) does not include the NULL values, whereas COUNT (*) includes the NULL values. No other aggregate function takes NULL into consideration. See Chapter 4 for more information.
19. E. These statements don't account for possible NULL values in the BONUS column. Read Chapter 3 for more information.
20. D. Using the ALTER TABLE statement, you can add new columns, rename existing columns, and drop existing columns. To learn more about managing tables, read chapter 7.
21. B. Since DEPARTMENT_ID is NULL for employee 178, NULL will be sorted after the non-NULL values when doing an ascending order sort. Since we did not specify the sort order or NULLS FIRST clause, the defaults are ASC and NULLS LAST. Read Chapter 2 for more information.
22. D, E, F, G. The INTERVAL DAY TO SECOND datatype is used to store an interval between two datetime components. Read Chapter 7 for more information.

- 23.** C, E. The easy connect syntax is @<host>:<port>/<service_name>. For databases where the sid is the same as the database unique name, the service name is the same as the sid; thus you can say option C is also part of the easy connect string. However, in reality it is the service name. Easy connect only supports TCP protocol. See Chapter 1 for more information.
- 24.** B. When altering an existing column to add a NOT NULL constraint, no rows in the table should have NULL values. In the example, there are two rows with NULL values. Read Chapter 7 for more information.
- 25.** D. NATURAL JOIN and JOIN ... USING clauses will not allow alias names to be used. Since a self-join is getting data from the same table, you must include alias names and qualify column names. See Chapter 5 for more information.
- 26.** E. The TRUNC function used with a negative second argument will truncate to the left of the decimal. See Chapter 3 for more information.
- 27.** C. Oracle creates unique indexes for each unique key and primary key defined in the table. The table ADDRESSES has one unique key and a primary key. Indexes will not be created for NOT NULL or foreign key constraints. See Chapter 7 for more information.
- 28.** D. Although there is no error in this statement, the statement will not return the desired result. When a NULL is compared, you cannot use the = or != operators; you must use the IS NULL or IS NOT NULL operator. See Chapter 2 for more information.
- 29.** C. The solid line represents that the order must belong to a customer, and the crow-foot represents that a customer can have more than one order. See Chapter 1 for more information.
- 30.** B. Private synonyms override public synonyms, and tables or views owned by the user always resolve first. See Chapter 7 for more information.
- 31.** C, D. When a COMMIT is executed, all locks are released, savepoints are erased, and queries started before the COMMIT will constitute a read-consistent view using the undo information. See Chapter 6 for more information.
- 32.** B, C. The operators OR and AND are used to add more joining conditions to the query. NOT is a negation operator, and a comma is used to separate column names and table names. For more information, see Chapter 5.
- 33.** C. Since we are finding the aggregate of aggregate, non-aggregate columns should not be used in the SELECT clause. See Chapter 4 for more information.
- 34.** B. You can create primary key, foreign key, and unique key constraints on a view. The constraints on views are not enforced by Oracle. To enforce a constraint, it must be defined on a table. See Chapter 7 for more information.

- 35.** A, C. The maximum lengths of CHAR and VARCHAR2 columns can be defined in characters or bytes. BYTE is the default; the default can be changed by setting the database parameter NLS_LENGTH_SEMANTICS. See Chapter 7 for more information.
- 36.** B, C. The row limiting feature uses OFFSET and FETCH clauses to filter the rows. To get to rows 6 through 15 here, you may use the clause OFFSET 5 ROWS FETCH NEXT 10 ROWS. See Chapter 2 for more information.
- 37.** D. The substitution character % may be substituted for zero or many characters. The substitution character _ does not have any effect in this query because an escape character precedes it, so it is treated as a literal. See Chapter 2 for more information.
- 38.** C. You have two main conditions in the question: one on the hire_date and the other on the salary. So an AND operator should be used. In the second part, you have two options: the salary can be either more than 5000 or less than 1000, so the second part should be enclosed in parentheses; use an OR operator. Option B is similar to option C except for the parentheses, but the difference changes the meaning completely. Option B would select the employees who were hired before 01-Jan-1998 or have a salary above 5,000 or have a salary below 1,000. Read Chapter 2 for more information.
- 39.** A, B. The TRUNCATE command is used to remove all the rows from a table or cluster. By default, this command releases all the storage space used by the table and resets the table's high-water mark to zero. None of the indexes, constraints, or triggers on the table are dropped or disabled. If there are valid foreign key constraints defined to this table, you must disable all of them before truncating the table. Read Chapter 6 for more information.
- 40.** A, C. Since there can be only one primary key per table, the syntax in option A works. Any constraint (except NOT NULL) can be dropped using the syntax in option C. See Chapter 7 for more information.

Exam 1Z0-062: Installation and Administration

- 1.** B. Database Upgrade Assistant and a manual upgrade both support an upgrade from 10.2.0.5, 11.1.0.7, 11.2.0.2, and higher-version databases. For 11.2.0.1, you must upgrade to 11.2.0.2 or higher before upgrading to 12.1.0. See Chapter 17 for more information.
- 2.** B. To rename or relocate a data file in Oracle Database 12c, you need to use only the statement specified in option B. To rename a data file in pre-12c databases, you need to take the tablespace offline so that Oracle does not try to update the data file while you are renaming. Using OS commands, copy the data file to the new location and using the ALTER DATABASE RENAME FILE command or the ALTER TABLESPACE RENAME FILE command, rename the file in the database's control file. To rename the file in the database, the new file should exist. Bring the tablespace online for normal database operation. For more information, read Chapter 10.
- 3.** C. Although pfiles and spfiles are physical files used to configure the Oracle instance, they are not considered part of the database. For more information, see Chapter 8.

4. B. The highest level at which a user can request a lock is the table level; the only other lock level available to a user is a row-level lock. Users cannot lock at the block or schema level. Read Chapter 11 for more information.
5. D. PUBLIC is the group or class of database users where all existing and future database users belong. Read Chapter 13 for more information.
6. B, C. B and C are both the best answers. They are identical because queries in Oracle are case-insensitive. For more information, read Chapter 16.
7. C. The Manageability Monitor (MMON) process gathers performance statistics from the SGA (System Global Area) and stores them in the AWR. MMNL (Manageability Monitor Light) also does some AWR-related statistics gathering, but not to the extent that MMON does. QMN1 is the process that monitors Oracle advanced queuing features. MMAN (Memory Manager) is the process that dynamically manages the sizes of each SGA component when directed to make changes by the ADDM (Automatic Database Diagnostic Monitoring). See Chapter 14 for more information.
8. B. RESUMABLE_TIMEOUT set at the initialization parameter is used only as the default timeout value for a resumable session. The resumable session must be enabled using option B. Option C disables the resumable session. Option D is invalid. Read Chapter 10 for more information.
9. B. Use the Oracle Universal Installer (OUI) to install and configure the Oracle Database 12c software. The OUI is a Java-based application that provides the same installation look and feel no matter which operating system is being used for the installation. runInstaller is the executable that invokes OUI on Linux/Unix systems. See Chapter 9 for more information.
10. C. Although you can use choice A to stop and start the listener, doing so temporarily disrupts clients attempting to connect to the database. Choice D is fine if you are starting and stopping the default listener called LISTENER, but we are using a nondefault listener. Choice B is not valid because RESTART is not a valid command-line argument for lsnrctl. Therefore, the best method is C, to use the lsnrctl reload listener1 command to load the new set of values for the listener without disrupting connection service to the databases the listener is servicing. For more information, see Chapter 12.
11. E. An error will occur because the job is not running, given that a job that is not executing cannot be stopped. The error produced will be ORA-27366, including other following errors as the error trap is pushed up the stack back to the executable (something like SQL*Plus), which executed the STOP_JOB procedure. See Chapter 16 for more information.
12. B. Redo log files record all the changes made to the Oracle database, whether the change is committed or not. See Chapter 15 for more information.
13. D. Default roles are enabled when a user connects to the database, even if the roles are password protected. Read Chapter 13 for more information.

14. D. Although options A and B are recommended practices for upgrade performance, option D is a must. The Pre-Upgrade Information tool must be run, and all errors must be taken care of before upgrading the database. Option C is not required for the upgrade process. For more information, read Chapter 17.
15. E. To tune the instance recovery time, configure the FAST_START_MTTR_TARGET parameter to a nonzero value. The default is 300 seconds. A lower value will reduce the instance recovery time, but may cause frequent checkpoints. A value of 0 turns off MTTR tuning. Read Chapter 15 for more information.
16. C. Oracle Shared Server requires a shift of memory away from individual session processes to the SGA. More information has to be kept in the SGA (in the UGA) within the shared pool. A large pool is configured and is responsible for most of the SGA space allocation. The cache size and block buffers settings do not affect Oracle Shared Server. See Chapter 12 for more information.
17. C, D. Recovery options are optional when you create a database. Specify Fast Recovery Area and Enable Archiving are the options available. Read Chapter 9 for more information.
18. B. The REPAIR FAILURE command works only after an ADVISE FAILURE. Option A is invalid. LIST FAILURE displays the failures. CHANGE FAILURE can be used to lower or raise the priority of a failure. For more information, see Chapter 15.
19. A, C. You cannot dynamically change the parameter UNDO_MANAGEMENT after the instance has started. You can, however, change the UNDO_TABLESPACE parameter to switch to another undo tablespace while the instance is up and running. See Chapter 11 for more information.
20. B. Oracle automatically performs instance recovery after a database crash or SHUTDOWN ABORT. All uncommitted changes are rolled back, and committed changes are written to data files during instance recovery. Read Chapter 9 for more information.
21. C. DB_DEST implies a database instance destination identified by a TNS name. LOCAL implies the current machine only. See Chapter 16 for more information.
22. C, D. Every database must have at least two redo log files, which may or may not be multiplexed. Every database must have one control file. It is a good idea to have more than one control file for redundancy. Because SYSTEM and SYSAUX are mandatory tablespaces, there will be at least two data files. Read Chapter 8 for more information.
23. C. There are no standard password complexity settings in either the initialization parameters or profiles. A password-verify function can validate new passwords against any rules that you can code in PL/SQL, including regular expression comparisons. For more information, read Chapter 13.
24. B. DBA_CHAINS does not exist. DBA_SCHEDULER_CHAINS only shows chained groups of jobs. DBA_SCHEDULER_DESTS shows job execution destinations. The DBA_SCHEDULER_JOBS metadata view has a JOB_PRIORITY column. For more information, read Chapter 16.

- 25.** C. The shared pool has three components: library cache, result cache, and dictionary cache. Read Chapter 8 for more information.
- 26.** A. You can use the Undo Management Advisor to monitor and manage the undo segments to ensure maximum levels of read consistency and minimize occurrences of ORA-01555: Snapshot Too Old error messages. For more information, see Chapter 14.
- 27.** C. The MINIMUM_EXTENT parameter is used to make sure each extent is a multiple of the value specified. This parameter is useful to reduce fragmentation in the tablespace. Read Chapter 10 for more information.
- 28.** A. Irrespective of the components audited, all audit trail information is queried from UNIFIED_AUDIT_TRAIL. Audit records are stored in a table under the AUDSYS table. See Chapter 13 for more information.
- 29.** D. Protocols come with tools that allow you to test network connectivity. One such utility for TCP/IP is ping. The user supplies either an IP address or a hostname to the ping utility. It then searches the network for this address. If it finds one, it displays information on data that is sent and received and on how quickly it found this address. The other choices are Oracle-supplied utilities. Read Chapter 12 for more information.
- 30.** B. The online redo log files are used to roll forward after an instance failure; undo data is used to roll back any uncommitted transactions. See Chapter 11 for more information.
- 31.** B. The log and bad files are written to (not read from) SQL*Loader, and the data file contains only data. The direct=y option, which tells SQL*Loader to use the direct path option, can appear on the command line or in the parfile. See Chapter 18 for more information.
- 32.** B. Shared servers can process requests from many users. The completed requests are placed into the dispatchers' response queues. The servers are configured with the SERVERS parameter. However, shared servers do not receive requests directly from dispatchers. The requests are taken from the request queue. Read Chapter 12 for more information.
- 33.** D. The distance between the checkpoint position in the redo log file and the end of the redo log group can never be more than 90 percent of the size of the smallest redo log group. See Chapter 15 for more information.
- 34.** B. A nonzero value for the MEMORY_TARGET parameter enables the Automatic Memory Management. SGA_TARGET enables Automatic Shared Memory Management. Automatic Memory Management tunes both SGA and PGA components of the memory. For more information, read Chapter 14.
- 35.** D. REPLACE is the valid value; it drops the existing table and creates the table using the definition from the dump file. SKIP leaves the table untouched. APPEND inserts rows to the existing table. TRUNCATE leaves the structure but removes all existing rows before inserting rows. Read Chapter 18 to learn more.

- 36.** B, C. Only the SYSTEM and UNDO tablespaces require the instance to be shut down when their data files need to be recovered. See Chapter 15 for more information.
- 37.** A. Oracle Database 12c uses the Automatic Diagnostic Repository to maintain the alert log and other diagnostic information. The BACKGROUND_DUMP_DEST parameter is derived from the DIAGNOSTIC_DEST. See Chapter 9 for more information.
- 38.** B. Option A is not correct because operations on temporary tablespaces do not generate redo, and a temporary table is created in a temporary tablespace. When DML is performed on a temporary table, the undo is written by default to the active undo tablespace, which generates redo. By setting the parameter TEMP_UNDO_ENABLED, you can direct the undo writing to a temporary tablespace instead of an undo tablespace, thus eliminating redo generation. Option C is wrong. When TEMP_UNDO_ENABLED is true, the undo gets generated as usual; however, the undo gets generated in the temp tablespace, so the redo is not generated. See Chapter 11 for more information.
- 39.** A, C. OEM Database Express is automatically configured when you create a database using DBCA. SQL*Plus is installed along with Oracle RDBMS software tools. SQLDeveloper and OEM Cloud Control must be downloaded and installed separately. Read Chapter 8 for more information.
- 40.** B. DB_CREATE_FILE_DEST specifies the directory to use to create data files and temp files. This directory is also used for control files and redo log files if the DB_CREATE_ONLINE_LOG_DEST_1 parameter is not set. See Chapter 10 for more information.
- 41.** C. In the previous database releases, the upgrade script was named catupgrd.sql. In Oracle Database 12c, the upgrade script is also catupgrd.sql, but it must be invoked using the catctl.pl perl script. This script runs the upgrade in parallel by default, thereby reducing the upgrade time. The utlu121i.sql script is replaced with the preupgrd.sql script and is used for pre-upgrade information. The utlu121s.sql script is a post-upgrade status tool. For more information, see Chapter 17.
- 42.** C. DRCP is enabled and disabled by using the database package DBMS_CONNECTION_POOL. The procedure START_POOL starts the server pool and STOP_POOL stops it. The CONFIGURE_POOL procedure can be used to configure various parameters. Read Chapter 12 for more information.
- 43.** D. DIAGNOSTIC_DEST determines the location of the alert log file and trace files. See Chapter 14 for more information.
- 44.** B. Oracle Grid Infrastructure includes Oracle ASM and Oracle Restart. ASM disks are managed and controlled by a special type of instance known as the ASM instance. The ASM instance does not have any data files or database associated to it; it has only the memory structures and processes. See Chapter 18 for more information.

Oracle Database 12c: SQL Fundamentals

PART

I



Chapter 1

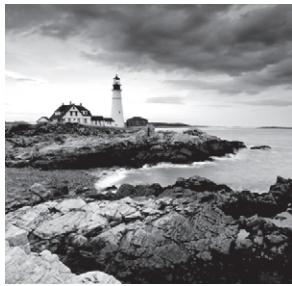


Introducing Oracle Database 12c RDBMS

ORACLE DATABASE 12c: SQL FUNDAMENTALS EXAM OBJECTIVES COVERED IN THIS CHAPTER:

✓ Introduction

- Describe the features of Oracle Database 12c.
- Describe the salient features of Oracle Cloud 12c.
- Explain the theoretical and physical aspects of a relational database.
- Describe Oracle server's implementation of RDBMS and object relational database management system (ORDBMS).



Organizations and individuals collect and use a variety of information (data). A database collects data, stores and organizes data, and retrieves related data used by a business. Oracle is the world's most widely used database management system. With the release of its Database 12c, Oracle has enhanced the capabilities of its feature-rich database to include cloud architecture. The *c* in 12c stands for cloud computing. From Oracle version 8 onward, Oracle includes the core emphasis of the release along with the version number in its name. Versions 8 and 9 are called *i* to indicate Internet computing; versions 10 and 11 are called *g* for grid computing.

With the cloud enablement, Oracle Database 12c lets you manage many databases as one, thereby reducing overhead and valuable resource consumption.

This chapter will introduce you to the Oracle Database 12c high-level components and how the Oracle database is organized. You will also learn about the relational and object capabilities of the database, and the tools available for database administrators (DBAs) to retrieve information and manage the database.



Exam objectives are subject to change at any time without prior notice and at Oracle's sole discretion. Please visit Oracle's Training and Certification website at <http://www.oracle.com/education/certification> for the most current exam objectives.

Relational Database Management Systems

A *database management system* (DBMS) controls the storage, organization, and retrieval of data. In a DBMS, the kernel code is the software piece that manages the storage and memory component of the database. There is metadata in the DBMS that keeps track of all the components of the database, also known as the dictionary. The code or language used to retrieve data from the database is known as SQL, which stands for *Structured Query Language*.

Over the years, database management systems have evolved from hierarchical to network to relational database management systems (RDBMS). A *relational database management system* is an organized model of subjects and characteristics that have relationships among the subjects. A well-designed relational database provides volumes of information about a business or process. RDBMS is the most widely used database system, and the object

structures are related. We see relationships everywhere in our daily lives: parents and children, team and players, doctor and patient, to name a few. The main advantages of RDBMS include the way it stores and retrieves information and how the data integrity is maintained. RDBMS structures are easy to understand and build. These structures are logically represented using the *entity-relationship (ER) model*. The exam will have one or two questions on the ER diagram and/or the RDBMS concept. You may already be familiar with the RDBMS concepts and ER diagrams; a brief refresher is included here.

Characteristics of a Relational Database

Relational databases have the following three major characteristics that constitute a well-defined RDBMS:

- **Structures** are objects that store or access data from the database. Tables, views, and indexes are examples of structures in Oracle.
- **Operations** are the actions that are used to define the structures or to manipulate data between the structures. SELECT statements and CREATE statements are examples of operations in Oracle.
- **Integrity rules** govern what kinds of actions are allowed on data and the database structure. These rules protect the data and the structure of the database. The primary keys and foreign keys are examples of integrity rules in Oracle.

Logical Model

In the design phase of the system development cycle, a logical model of the database is created. A logical model of an RDBMS is typically a block diagram of entities and relationships, referred to as an *entity-relationship (ER) model* or ER diagram.

An ER model has *entity*, *relationship*, and *attributes*. An ER model is visual, showing the structure, characteristics, and interactions within and around the data being modeled.

Entities and Attributes An entity in a logical model is much like a noun in grammar—a person, place, or thing. The characteristics of an entity are known as its attributes. Attributes are detailed information about an entity that serves to qualify, identify, classify, or quantify it. For example: ABC Inc. has many offices in the United States; each office has many departments, and each department may have many employees. Placing the organization of ABC Inc. in terms of the ER model, you could identify OFFICE, DEPARTMENT, and EMPLOYEE as entities. Each entity will also have its own characteristics. For instance, when you say “office,” you might want to know the address and city where the office is located, the state, and how many employees work there. Similarly, you might want to know the department name, its manager, the employee’s name, date of birth, hiring date, and salary grade. You might also like to know the employee’s spouse’s name. See Figure 1.1.

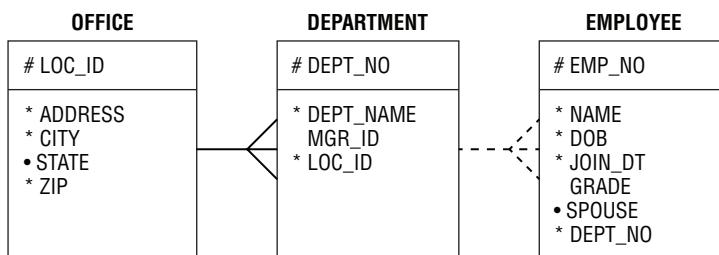
There are optional and mandatory attributes. In Figure 1.1, the spouse’s name, along with the employee information, is optional; whereas the employee name, the department he/she belongs to, hire date, and date of birth are mandatory in Figure 1.2. An asterisk along with the attribute name indicates that it is mandatory. The optional attribute may be indicated with a small o.

FIGURE 1.1 Entities and attributes

Office	Department	Employee
Address City State ZIP	Department Name Manager	Name DOB Join Date Grade Spouse

Relationships and Unique Identifiers In the example of ABC Inc., the relationship between the entities is described as “each office has many departments,” “one department belongs to only one office,” “each department has many employees,” and “one employee can belong to only one department.” If there is an office in one city, there should be at least one department. So it is mandatory to have at least one occurrence of department for each location. There may be many departments in one location. In the ER model, a solid line represents a mandatory relationship, and a crowfoot represents the “many.” But in some departments, there may not be any employees at all. Optional occurrence is represented by a dotted line.

You should be able to identify each occurrence of an entity uniquely. Now what happens if there are two employees with the same name? How do you distinguish them? For office location, the city and state uniquely identify each office; for department, the department name identifies it uniquely. For employee, you can introduce a unique identifier (UID) called employee number. Figure 1.2 is a refined version of Figure 1.1, and it shows the entities, attributes, relationships, optional and mandatory attributes, and UIDs. UID is represented in the diagram using a pound (#) symbol.

FIGURE 1.2 An entity-relationship (ER) model

Three types of relationships can be defined between the entities. (Figure 1.3):

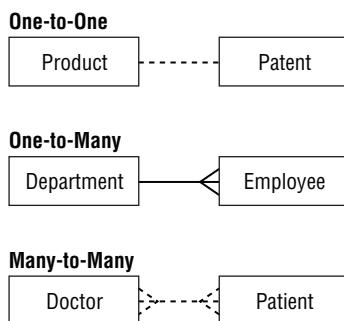
One-to-One A one-to-one relationship is one in which each occurrence of one entity is represented by a single occurrence in another entity. For example, product and patent—one product might have a patent, and one patent corresponds to only one product.

One-to-Many A one-to-many relationship is one in which an occurrence of one entity can be represented by many occurrences in another entity. For example, department and

employees—one department has one or more employees, and an employee belongs to only one department.

Many-to-Many A many-to-many relationship is one in which an occurrence from one entity can be represented by one or more occurrences in another entity, and an occurrence from the second entity may be represented by one or many occurrences in the first entity. Many-to-many relationships should not exist in RDBMS because they cannot be properly joined to represent a single row correctly. To solve this, create another entity that has an one-to-many relationship with the first entity and an one-to-many relationship with the second entity. For example, doctor and patient—a patient can visit many doctors, and a doctor can have many patients.

FIGURE 1.3 Types of relationships



The logical model also provides information known as *access paths*. They are the common ways you usually query the database in order to retrieve information. For example, you would always query the employee records with the Dept_No or Emp_No. Think of the access paths as an index to the data; they help us locate data just as the index of a book helps us quickly find the information we need.

When you have established the relationships between entities, it's time to normalize the design. *Normalization* is the process of eliminating redundant information from the entities until you can uniquely identify each occurrence of the entity. This may not always be practical due to performance and implementation issues. In such cases, you can denormalize to some extent.

Physical Model

The physical model is created by taking the logical model and creating a database and database objects to represent the entities and relationships. In the physical model, each entity becomes a *table* and attributes of the entity become *columns* of the table. The relationship between the entities is part of one or more *constraints* between the tables. Physical implementations might force you to combine, separate, or create completely new entities in order to best realize the logical model. The unique identifiers of an entity become the *primary key* of the table. Stored procedures, functions, and triggers may be created to enforce business rules.



In RDBMS, the physical database storage is independent of the logical model.

Oracle's Implementation of RDBMS and ORDBMS

A database server is the key to information management. An Oracle database satisfies all three major characteristics of the relational model. Oracle lets you define tables, columns, column characteristics such as datatype, length, whether the values are mandatory, and default values. Defining *foreign key* ensures the *referential integrity* of the data. You can define *primary keys* and indexes on the data. The primary key of a relational table uniquely identifies each record in the table; it may consist of a single attribute (column) or multiple attributes in combination. A foreign key is a column (or collection of columns) in one table that uniquely identifies a row of another table, defining the relationship between the tables.

Records in a database table can be seen as instances of the entity. Each occurrence of an entity is differentiated by the values of the attributes. Oracle stores these records as *rows* of the table and the attributes as *columns* in each row. In the most generic form, a database table can be seen as a single spreadsheet with unlimited numbers of columns and rows. The columns are not defined until the user names them and gives them a datatype. Oracle extends the concept of spreadsheets by defining relationships between multiple spreadsheets, constraints on columns, and providing mechanisms for multiple users to access the same database table(s) at the same time.

The data access path is implemented in Oracle using indexes. Indexing allows us to predefine to the relational database system the most common access paths that will be used. These indexes decrease the time required to search for data in a table using a number of algorithms such as B-tree, bitmap, etc.

Oracle implements the RDBMS characteristics using the following set of structures:

- Tables are used for data storage.
- Views and synonyms are created for data access.
- Indexes are used to speed up data retrieval.
- Primary keys, foreign keys, and unique keys are called *constraints* and are created to enforce data integrity.
- Triggers are created to satisfy the business rules.
- Roles and privileges are used for security.
- Procedures, functions, and packages are used to code the application.

Oracle, since version 8i, is also an Object Relational DBMS. An RDBMS that implements object-oriented features such as user-defined types, inheritance, and polymorphism is called ORDBMS. It lets you create user-defined object types in the relational database system. Object types are structures that consist of built-in or user-defined data types. For example, Address can be defined as an object type and can be referenced in tables.

Here's an example where STREET_TYPE is defined as:

```
STREET_TYPE
STREET_NUMBER NUMBER (6)
STREET_NAME1 VARCHAR2 (40)
STREET_SUFFIX VARCHAR2 (10)
APARTMENT_NO VARCHAR2 (5)
```

Here's an example where ADDRESS_TYPE is an object type defined using another object type as:

```
ADDRESS_TYPE
STREET      STREET_TYPE
CITY        VARCHAR2 (30)
STATE       CHAR (2)
ZIP         NUMBER (5)
```

In this example for CUSTOMER_TABLE, the object CUST_ADDR is a type.

```
CUSTOMER_TABLE
CUST_NAME   VARCHAR2 (40)
CUST_ADDR   ADDRESS_TYPE
CUST_PHONE  VARCHAR2 (12)
CUST_FAX    VARCHAR2 (12)
```

Now that the ADDRESS_TYPE is defined, it can be used in any number of tables, where ADDRESS needs to be stored. This is a small example to show you how objects can be reused and how the functionality of the RDBMS can be extended to include built-in complex business rules.

The Oracle Database 12c

An Oracle Database 12c server is a feature-rich RDBMS that extends its capabilities beyond any other RDBMS in the market, with object relational and cloud capabilities. In this section, we will discuss the capabilities and features of Oracle Database 12c.

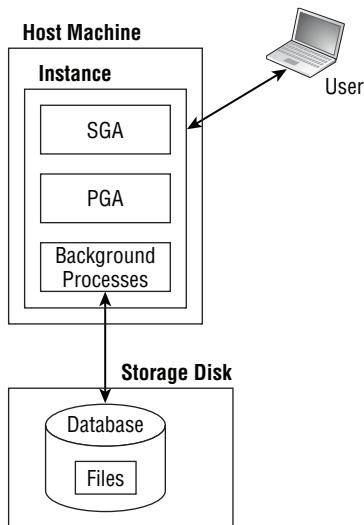
Oracle Database 12c Implementations

Let's start with the architecture of the database server at a very high level. Detailed architecture and components are discussed in various chapters in Part II of this book.

The physical structure of an Oracle Database 12c server consists of two major components: the *database* and the *instance*. The database is a set of physical files saved on the disk that store information. The instance is a set of memory structures and processes that uses the physical components to manipulate and retrieve data.

Figure 1.4 shows the database architecture. The host machine is where the Oracle instance is running. It has the memory structures and processes. The storage array, or disk, is where the database resides.

FIGURE 1.4 An Oracle database server



In the architecture shown in Figure 1.4, one instance communicates with one database. The host machine is where users and applications connect and interact. If the machine goes down for some reason, the database will be unavailable. Oracle alleviates this issue by introducing an architecture named the *Real Application Clusters* (RAC).

Figure 1.5 shows RAC architecture. In this architecture, more than one instance communicates to a single database. Oracle RAC takes reliability a step further by removing the database server as a single point of failure. If an instance fails, the remaining instances in the RAC pool remain open and active; and connections from failed instances can be failed-over to active instances. The RAC load balancer directs the user connection request to the appropriate instance.

With RAC, high availability and CPU/memory capacity available to the database is increased. Oracle manages the connection load balancing and failover automatically.

Many organizations have several hundreds or thousands of Oracle databases. Imagine if the policy were to have one instance per server, then you would have as many servers as the number of instances to manage. If you have a high-capacity server or if the database resource requirements are minimal, you can have more than one instance on the same host machine. Figure 1.6 shows an architecture in which more than one database is hosted on the same machine.

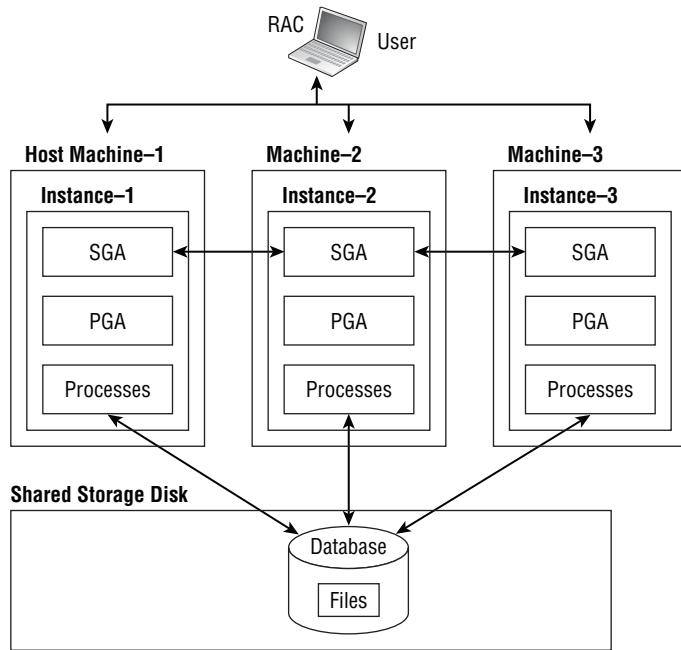
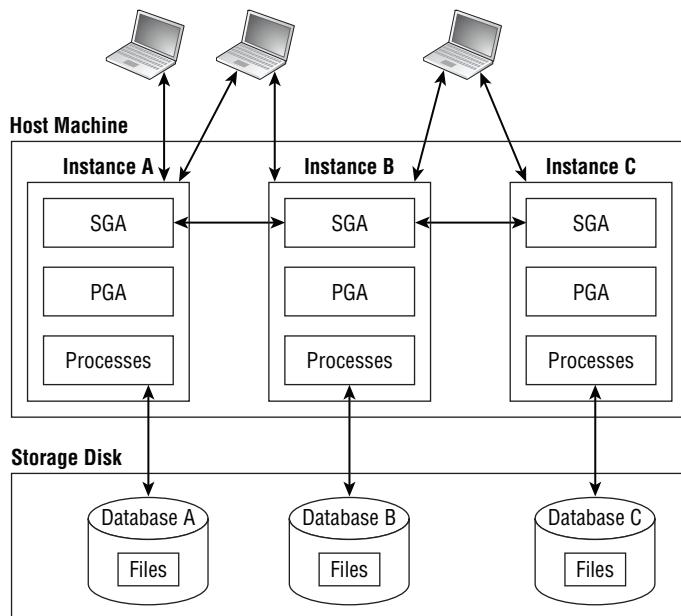
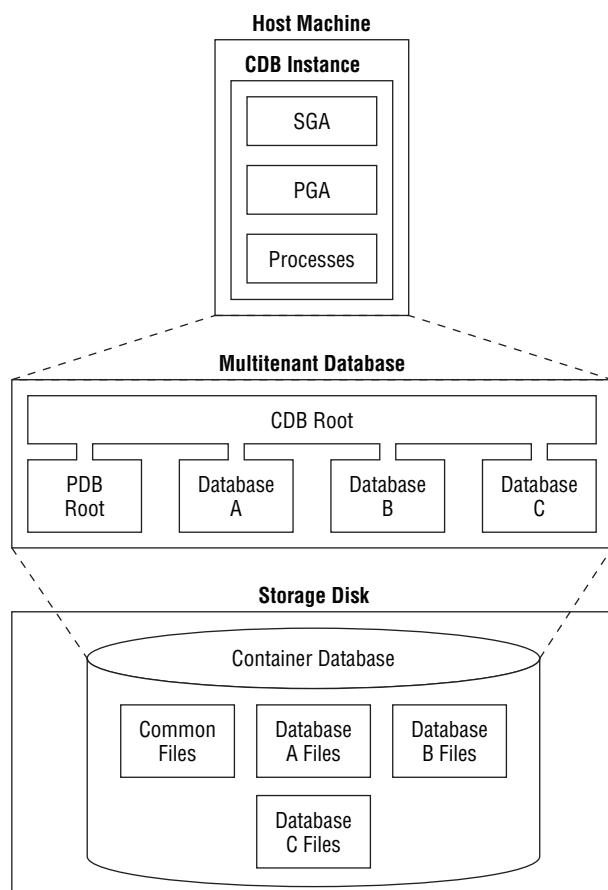
FIGURE 1.5 An Oracle database server — RAC**FIGURE 1.6** Multiple Oracle databases on same machine

Figure 1.6 clearly shows that even though you consolidated multiple database servers to one host, you still have the same overhead of managing the database. You need instance memory structures, processes, and management activities such as backup for each instance or database.

With Oracle 12c, a new architecture feature is introduced known as the *multitenant architecture*. The multitenant architecture enables an Oracle database to function as a multitenant container database (CDB) that includes zero, one, or many *pluggable databases* (PDBs). All databases created prior to Oracle Database 12c are non-CDB; a pluggable database appears as a non-CDB to the application, so existing code and application need not be changed when you move to Oracle Database 12c.

The PDBs belonging to a CDB share the database overhead such as redo, undo, and memory. Oracle RDBMS is responsible for keeping the pluggable databases separate, private for the application, and secure. The instance and SGA are assigned to a container database. Figure 1.7 shows the *multitenant database* architecture. The databases that are part of the CDB are known as pluggable databases.

FIGURE 1.7 Oracle Database 12c — Multitenant architecture with pluggable databases



Pluggable databases in the multitenant architecture, as the name suggests, can be unplugged and plugged to another CDB easily. With the pluggable databases and multitenant architecture, Oracle Database 12c offers the following benefits:

- **Increased server utilization:** Because the overhead associated with each database is now shared among all databases, you can consolidate more databases with the same resources.
- **Cost reduction:** By consolidating hardware and sharing database memory and files, the cost of hardware is less.
- **Application transparency:** Although the architecture changed, each PDB acts and works as a traditional pre-12c Oracle database. There is no need to change application code or architecture to start using Oracle Database 12c.
- **Manage many databases as one:** Administrative activities such as patching and upgrade are performed on the container database so they do not need to be repeated for each database in the CDB. This drastically reduces the administrative time required.
- **Less backup configuration:** With multiple databases consolidated into one server, you still have to back up each database separately. With container and pluggable databases, you only need to back up one multitenant container database.
- **Easier provisioning:** In the container database architecture, it is very easy to clone and provision pluggable databases.
- **Less time to upgrade:** When you upgrade the container database, all the pluggable databases are automatically upgraded.
- **Move databases:** It is also possible to move a pluggable database from one container database to another. This is especially useful if you plan to upgrade all but a few databases to the next release. Before upgrading the container database, you may move a few pluggable databases to another container database, or move pluggable databases to another CDB of higher release.
- **Separation of duties:** Database administrators can be defined as a CDB administrator (common user with administrative privileges on all CDB and PDBs) or a PDB administrator (local user in PDB with administrative privilege only on the PDB).

In the next section, you will learn about users and schema in the database, which is the basis for connecting to the database.

Connecting to Oracle Database

Before you can connect to an Oracle database, you must create a user. When you create a new Oracle database, several default users are automatically created. (The preferred method for creating a database is to use the Database Configuration Assistant [DBCA], which is discussed in Chapter 9, “Creating and Operating Oracle Database 12c.”)

SYS is the data dictionary or metadata dictionary owner for the database. Using this account for day-to-day operations is not recommended. You connect to SYS to start and stop database and other key administrative tasks. SYSTEM is a powerful administrative user in the database. Initially, you use this user connection to create database users and other

administrators in the database. Along with `SYS` and `SYSTEM`, several other database users are also created based on the options you choose during the installation and on the components installed in the database.

In a container database, the users are either common or local. Common users are visible on the container as well as in all pluggable databases; they have the same username and password across all pluggable databases and in the container database. The schema for common user is still local to each pluggable database and the container database.

A *schema* is a collection of database objects owned by a user account in the database. The objects in the schema may be related to support a business application. A schema and user have a one-to-one relationship in a database. A schema is created as a user in the database, but when the user owns database objects, it is called a schema. Schema objects are discussed in Chapter 7, “Creating Tables and Constraints.”

When an object is created under someone’s schema, the user has full privilege on the object by default. Using Oracle’s roles and privileges, a schema owner or administrator can grant privilege on his or her object (such as a table) to another user in the same database. This is known as object-level privilege. For certain users, you may want to grant privileges on all the objects in the database; this is accomplished by using the system privileges. Privileges and system security are discussed in Chapter 13, “Implementing Security and Auditing.”

The next section will introduce you to the tools available to manage and administer Oracle Database 12c.

Database Management Tools

Oracle Database 12c comes with multiple feature-rich tools to help administrators manage and monitor the database and all of its components. In this section, you will review the tools that are used for everyday administration of Oracle database. Let’s start with the tool that is equally beneficial to DBAs, developers, and power users.

SQL Developer

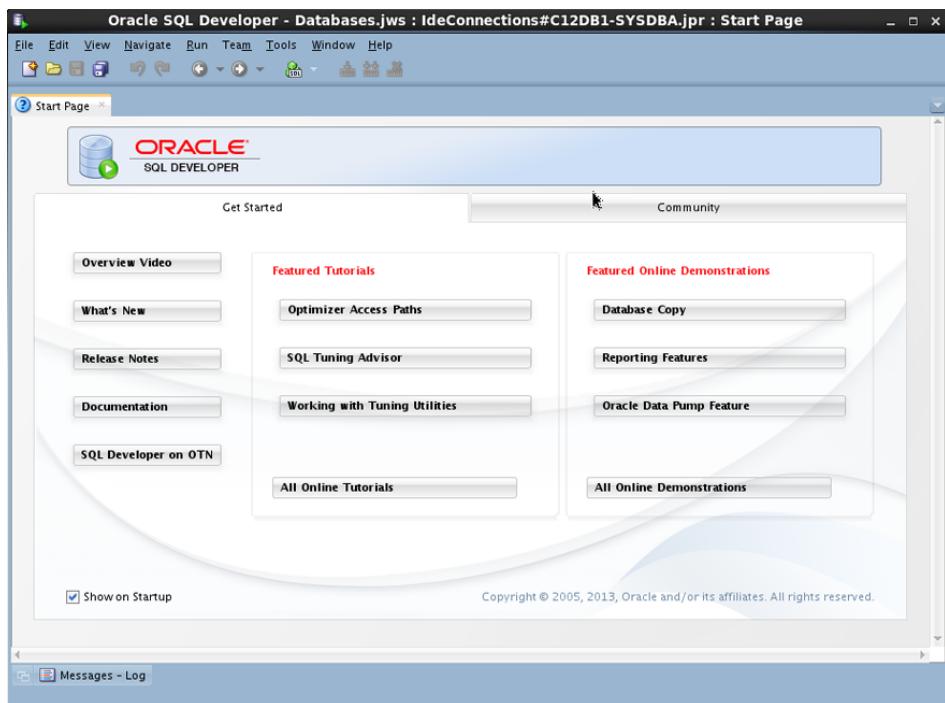
SQL Developer is a graphical tool used to perform everyday activities in the Oracle database. It has several predefined menu functions; therefore, there is no need to remember the syntax or SQL command to perform basic functions. In addition to Oracle (database versions higher than 9i Release 2), SQL Developer can also connect to Microsoft Access, Microsoft SQL Server, MySQL, IBM DB2, and Sybase Adaptive Server databases to view data and metadata.

SQL Developer is installed along with Oracle Database 12c on Windows platforms. For Linux, you must download and install it outside of an Oracle Database 12c installation. The distribution usually included in the database software distribution might not be the current version. It is better to always download the latest version and install SQL Developer from <http://www.oracle.com/technetwork/developer-tools/sql-developer/downloads>. As instructed on the download page, you will also need Java JRE installed for SQL Developer to work.

Once it is installed, you can invoke SQL Developer from `/usr/local/bin` directory on Linux or from the Oracle Installation program group on Windows. Figure 1.8 shows the

initial SQL Developer screen. To get started and learn more about SQL Developer, click the choices available in the screen.

FIGURE 1.8 The initial SQL Developer screen

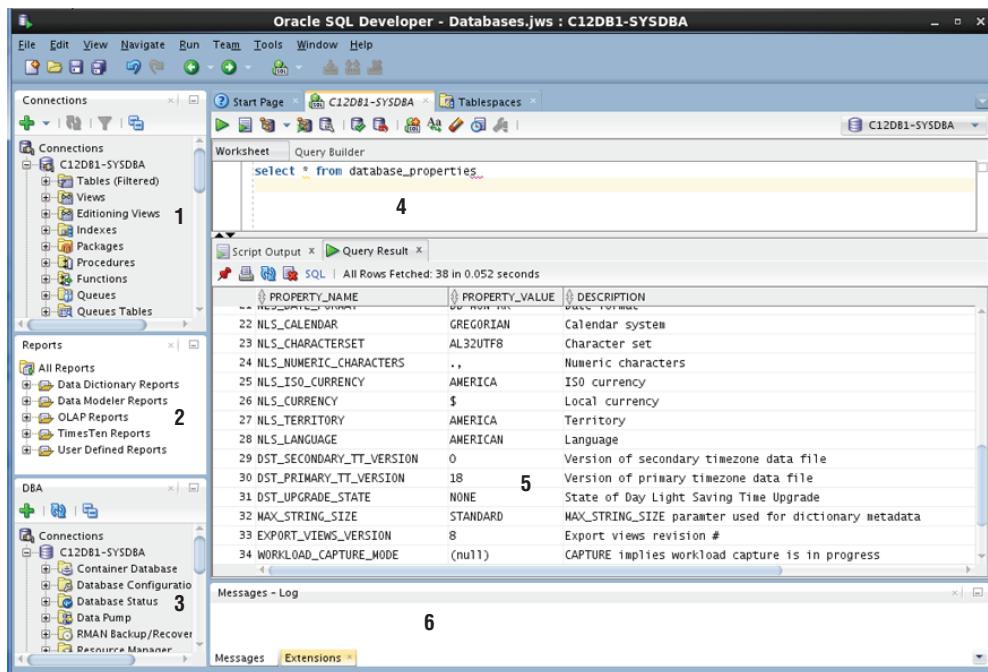


The Connections navigation pane is the most commonly used pane, which navigates through the database objects providing you with information and options to modify the objects. In Figure 1.9, you can see the various sections and navigations in SQL Developer. The section marked 1 shows various database connections and the objects in those databases. Section 2 provides you with a set of predefined reports. If you know how to write SQL, you can define your own reports as well. Section 3 shows the DBA navigation screen, showing you various DBA tasks.

Section 4 is the SQL worksheet. You use this section to interact directly with the database using SQL language. Output from the SQL commands is listed in section 5; both query output and script output are visible. Section 6 is the logging pane. This is useful when you are debugging code.



For simplicity and ease of results capture, the SQL statements used in this book are using SQL*Plus. All of the SQL commands can also run using the SQL Worksheet in SQL Developer.

FIGURE 1.9 SQL Developer navigation windows

SQL*Plus

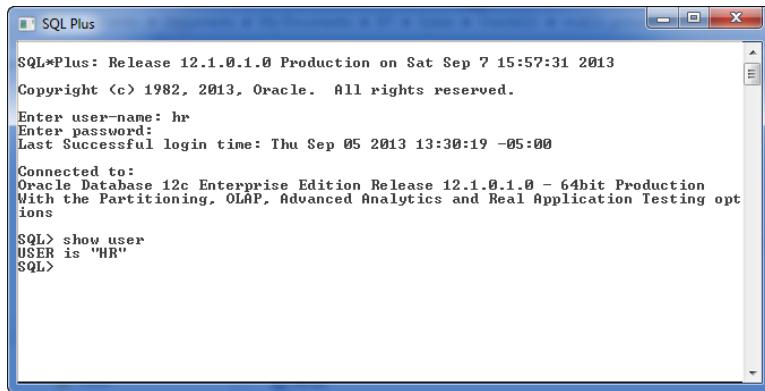
SQL*Plus is Oracle's command-line interface to the Oracle database. You run SQL commands to query the database or to manage the database. SQL*Plus is packaged with the Oracle software and can be installed using the client software installation routine on any machine. This tool is automatically installed when you install the Oracle Database 12c server software.

On Unix/Linux platforms, you can invoke SQL*Plus using the `sqlplus` executable found in the `$ORACLE_HOME/bin` directory. On Windows, SQL*Plus is under the Oracle Home Group menu. On Windows and Unix/Linux platforms, when you start SQL*Plus, you will be prompted for a username and password, as shown in Figure 1.10.

Once you are in SQL*Plus, you can connect to another database or change your connection by using the `CONNECT` command, with this syntax:

```
CONNECT <username>/<password>@<connect_string>
```

The slash separates the username and password. The connect string following @ is the database alias name known as the net service name. If you omit the password, you will be prompted to enter it. If you omit the connect string, SQL*Plus tries to connect you to the local database defined in the `ORACLE_SID` variable (or the net service name defined by `TWO_TASK` variable). You will not be prompted for the <connect_string>. Include <connect_string> along with <username> as in `bthomas@myowndb`.

FIGURE 1.10 A SQL*Plus screen

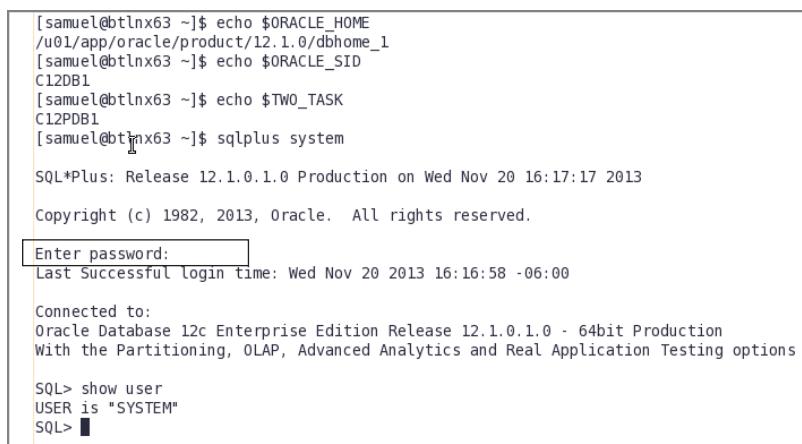
You may replace the connect string with a construct called the easy connect. The syntax is `[//]Host[:Port]/<service_name>`. For example, to connect to database service named C12DB1 on machine BTLNX63, where the listener is running in port 1521, use

```
sqlplus system@"btlnx63:1521/C12DB1"
```

You can invoke and connect to SQL*Plus using the `sqlplus` command, with this syntax:

```
sqlplus <username>/<password>@<connectstring>
```

If you invoke the tool with just `sqlplus`, you will be prompted for a username and password, as in Figure 1.10. If you invoke SQL*Plus with a username, you will be prompted for a password. See Figure 1.11 for an example.

FIGURE 1.11 A SQL*Plus screen with a password prompt

Once you are connected to SQL*Plus, you will get the SQL> prompt. This is the default prompt, which can be changed using the SET SQLPROMPT command. Type the command you want to execute at this prompt. With SQL*Plus, you can enter, edit, and execute SQL statements; perform database administration; and execute statements interactively by accepting user input. You can also format query results and perform calculations.

To exit from SQL*Plus, use the EXIT command. On platforms where a return code is used, you can provide a return code while exiting. You can also use the QUIT command to complete the session. EXIT and QUIT are synonymous.



The command `sqlplus -help` displays a help screen to show the various options available when starting SQL*Plus. Multiple administrative connections such as SYSDBA, SYSOPER, SYSBACKUP, SYSASM, SYSKM, and SYSDG are also available. They are discussed in this book in various chapters.

Oracle Enterprise Manager Database Express 12c

Oracle Enterprise Manager (OEM) Database Express 12c is a web-based tool that can be configured using default by Database Configuration Assistant (DBCA, the tool you use to create and configure databases) when you create an Oracle database. OEM Database Express by default uses port 5500, hence is invoked using URL `https://database_host_machine:5500/em`.

OEM Database Express is designed to manage only one database, and is intended for database administrators. When you invoke Database Express, you will be prompted for a username and password to connect to the database. You should provide a user account that has administrative privileges.

For the SQL section of this book, you will not be using this tool, so we will not discuss it in more detail here. You can read more information about OEM Database Express in Chapter 3, “Getting Started with Database Administration,” of “Oracle Database 2 Day DBA 12c Release 1 (Part E17643-12),” found at <http://docs.oracle.com/>.

Oracle Enterprise Manager Cloud Control 12c

Oracle Enterprise Manager Cloud Control 12c (OEM 12c) is Oracle’s integrated enterprise management administrative tool, providing complete cloud management solutions. With OEM 12c, you can manage multiple databases and all products under the Oracle stack. It is a complete cloud lifecycle management answer to quickly set up, administer, and support enterprise clouds and Oracle environments from applications to storage.

OEM 12c is not part of the Oracle Database 12c software install; it must be downloaded and installed separately. To read more about OEM 12c, please check out “Oracle Enterprise Manager Cloud Control Introduction 12c Release 3 (Part E25353-14)” at <http://docs.oracle.com/>.

Oracle Database 12c in the Cloud

Cloud architecture emphasizes sharing resources and maximizing the effectiveness of shared resources. Cloud resources are shared not only by multiple users, but are also capable of reallocation based on demand. Cloud computing allows organizations to provision resources and applications rapidly, with improved manageability and less administrative overhead.

By enabling customers to efficiently use their information technology infrastructure, Oracle Database 12c was designed for the cloud. The following are the benefits of having Oracle Database 12c in the cloud architecture:

- It consolidates multiple Oracle databases into multitenant container databases.
- With multitenant architecture, DBAs can manage multiple databases as one database for many administrative tasks on the database. DBAs need to perform fewer patches and upgrades and will not need to configure many backups.
- It automatically optimizes database storage and performance based on usage.
- Oracle Database 12c supports smart compression and storage tier. The heat map feature tracks data usage information; administrators can create appropriate policies to automatically move and compress data based on age and activity of data.
- Oracle RAC supports deployment of database instances across a pool of servers, helping to avoid downtime caused by unplanned server outages.
- With Oracle Enterprise Manager Cloud Control 12c, the provisioning and cloning of databases are simplified.

Oracle Database 12c helps customers reduce IT complexity and cost through private database cloud deployments by consolidation. Cloud computing offers an opportunity for IT organizations to be more responsive to changes in application workloads and business demands.

Because the test is on SQL and the tool used throughout the book for executing SQL is SQL*Plus, the next section will discuss some fundamentals of SQL*Plus.

Becoming Familiar with SQL*Plus

SQL*Plus, widely used by DBAs and developers to interact with a database, is a powerful tool from Oracle. Using SQL*Plus, you can execute all SQL statements and PL/SQL programs, format results from queries, and administer the database.

Earlier in this chapter, you learned how to connect to the database using SQL*Plus. In this section, you will learn about entering SQL commands, understanding the difference between SQL commands and SQL*Plus commands, editing the SQL*Plus buffer, and running commands in a script.

Entering SQL Statements

A SQL statement can spread across multiple lines, and the commands are not case sensitive. The previously executed SQL statement will always be available in the *SQL buffer*.

The buffer can be edited or saved to a file. You can terminate a SQL statement in any of the following ways:

- End with a semicolon (;): The statement is completed and executed.
- Enter a slash (/) on a new line by itself: The statement in the buffer is executed.
- Enter a blank line: The statement is saved in the buffer.

You can use the RUN command instead of a slash to execute a statement in the buffer. The SQL prompt returns when the statement has completed execution. You can enter your next command at the prompt.



Only SQL statements and PL/SQL blocks are stored in the SQL buffer; SQL*Plus commands are not stored in the buffer.

Entering SQL*Plus Commands

SQL*Plus has its own commands to perform specific tasks on the database, as well as to format the query results. Unlike SQL statements, which are terminated with a semicolon or a blank line, SQL*Plus commands are entered on a single line. Pressing Enter executes the SQL*Plus command.

When you log in to the SQL*Plus session, you get the SQL prompt. By default, the prompt is SQL>. You can change this prompt using the SET SQLPROMPT SQL*Plus command. When you continue a SQL command to the next line, a line number appears at the beginning of the line. As shown here, when you type SELECT USERNAME in the first line (the SQL prompt line) and press Enter, line number 2 appears where you continue the SQL command FROM DBA_USERS.

```
SQL> SELECT USERNAME  
      2 FROM DBA_USERS
```

SQL statements can span multiple lines. If you want to continue a SQL*Plus command onto the next line, you must end the current line with a hyphen (-), which indicates command continuation. When a command continuation character is entered, SQL*Plus will not show the line number next, but instead displays the greater than symbol (>). This is in contrast to SQL statements, which can be continued to the next line without a continuation operator. For example, the following SQL statement gives an error, because SQL*Plus treats the hyphen operator (-) as a continuation character instead of a minus operator:

```
SQL> SELECT 800 -  
> 400 FROM dual;  
SELECT 800 400 FROM dual  
*  
ERROR at line 1:  
ORA-00923: FROM keyword not found where expected  
SQL>
```

You need to put the hyphen in the next line for the query to succeed:

```
SQL> SELECT 800  
2 - 400 FROM dual;
```

```
800-400  
-----  
400  
SQL>
```

Getting Structural Information with the DESCRIBE Command

You can use the DESCRIBE command to obtain information about the database objects. Using DESCRIBE on a table or view shows the columns, its datatypes, and whether each column can be NULL. Using DESCRIBE on a stored program, such as procedure or function, shows the parameters that need to be passed in/out, their datatype, and whether there is a default value. You can abbreviate this command to the first four characters or more—DESC, DESCR, and DESCRIB are all valid.

If you're connected to the HR schema and need to see the tables and views in this schema, use the following query:

```
SQL> SELECT * FROM tab;
```

TNAME	TABTYPE	CLUSTERID
COUNTRIES	TABLE	
DEPARTMENTS	TABLE	
EMPLOYEES	TABLE	
EMP_DETAILS_VIEW	VIEW	
JOB_HISTORY	TABLE	
JOBS	TABLE	
LOCATIONS	TABLE	
REGIONS	TABLE	

8 rows selected.

To see the columns or definition of the EMPLOYEES table, execute:

```
SQL> DESCRIBE employees
```

Name	Null?	Type
EMPLOYEE_ID	NOT NULL	NUMBER(6)
FIRST_NAME		VARCHAR2(20)

LAST_NAME	NOT NULL VARCHAR2(25)
EMAIL	NOT NULL VARCHAR2(25)
PHONE_NUMBER	VARCHAR2(20)
HIRE_DATE	NOT NULL DATE
JOB_ID	NOT NULL VARCHAR2(10)
SALARY	NUMBER(8,2)
COMMISSION_PCT	NUMBER(2,2)
MANAGER_ID	NUMBER(6)
DEPARTMENT_ID	NUMBER(4)

If there are invisible columns in the table, they are not displayed by the DESCRIBE command unless you use SET COLINVISIBLE ON.



Invisible columns are newly introduced in Oracle Database 12c, where a column in the table can be hidden from the application. Invisible columns help to remove a column from the table quickly without actually dropping the column. Invisible columns are discussed in Chapter 7.

Editing the SQL Buffer

The most recent SQL statement executed or entered is stored in the SQL buffer of SQL*Plus. You can run the command in this buffer again by simply typing a slash or using the RUN command.

SQL*Plus provides a set of commands to edit the buffer. Suppose you want to add another column or add an ORDER BY condition to the statement in the buffer. You do not need to type the entire SQL statement again. Instead, just edit the existing statement in the buffer.

One way to edit the SQL*Plus buffer is to use the EDIT command to write the buffer to an operating-system file named afiedt.buf (this is the default filename, which can be changed) and then use a system editor to make changes.



You can use your favorite text editor by defining it in SQL*Plus. For example, to make Notepad your favorite editor, just issue the command DEFINE _EDITOR = NOTE PAD.

To view the editor defined, just execute DEFINE _EDITOR as shown here.

```
SQL> define _editor
DEFINE _EDITOR      = "Notepad" (CHAR)
Provide the entire path if the program is not available in the search path.
```

Another way to edit the buffer is to use the SQL*Plus editing commands. You can make changes, delete lines, add text, and list the buffer contents using the commands described in the following sections. Most editing commands operate on the current line. You can change

the current line simply by typing the line number. All commands can be abbreviated, except DEL (which is already abbreviated).

LIST

The LIST command lists the contents of the buffer. The asterisk indicates the current line. The abbreviated command for LIST is L.

```
SQL> L
 1  SELECT empno, ename
 2* FROM emp
SQL> LIST LAST
 2* FROM emp
SQL>
```

The command LIST *n* displays line *n*, and LIST * displays the current line. The command LIST *m n* displays lines from *m* through *n*. If you substitute * for *m* or *n*, it implies from or to the current line. The command LIST LAST displays the last line.

APPEND

The APPEND *text* command adds text to the end of a line. The abbreviated command is A.

```
SQL> A WHERE empno <> 7926
 2* FROM emp WHERE empno <> 7926
SQL>
```

CHANGE

The CHANGE */old/new* command changes an old entry to a new entry. The abbreviated command is C. If you omit *new*, *old* will be deleted.

```
SQL> C /<>/=
 2* FROM emp WHERE empno = 7926
SQL> C /7926
 2* FROM emp WHERE empno =
SQL>
```

The ellipses (...) can be used as wildcard characters. The following example changes everything in the line from “fro” to the new value.

```
SQL> l
 1* select name from v$instance
SQL> c/fro.../from v$database
 1* select name from v$database
SQL>
```

The next example shows the substitution of a string in the middle of the line using ellipses.

```
SQL> l
 1* select owner from dba_tables where table_name like 'HR%'
SQL> c/dba...table/dba_views where view
 1* select owner from dba_views where views where table_name like 'HR%'
SQL>
```

INPUT

The **INPUT** *text* command adds a line of text. Its abbreviation is **I**. If *text* is omitted, you can add as many lines as you want.

```
SQL> I
 3 7777 AND
 4 empno = 4354
 5
SQL> I ORDER BY 1
SQL> L
 1 SELECT empno, ename
 2 FROM emp WHERE empno =
 3 7777 AND
 4 empno = 4354
 5* ORDER BY 1
SQL>
```

DEL

The **DEL** command used alone or with * deletes the current line. The **DEL** *m n* command deletes lines from *m* through *n*. If you substitute * for *m* or *n*, it implies the current line. The command **DEL LAST** deletes the last line.

```
SQL> 3
 3* 7777 AND
SQL> DEL
SQL> L
 1 SELECT empno, ename
 2 FROM emp WHERE empno =
 3 empno = 4354
 4* ORDER BY 1
SQL> DEL 3 *
```

```
SQL> L
  1  SELECT empno, ename
  2* FROM emp WHERE empno =
SQL>
```

CLEAR BUFFER

The CLEAR BUFFER command (abbreviated CL BUFF) clears the buffer. This deletes all lines from the buffer.

```
SQL> L
  1  SELECT empno, ename
  2* FROM emp WHERE empno =
SQL> CL BUFF
buffer cleared
SQL> L
No lines in SQL buffer.
SQL>
```

Using Script Files

SQL*Plus provides commands to save the SQL buffer to a file, as well as to run SQL statements from a file. SQL statements saved in a file are called a *script file*.

You can work with script files as follows:

- To save the SQL buffer to an operating-system file, use the command `SAVE filename`. If you do not provide an extension, the saved file will have the extension `.sql`.
- By default, the `SAVE` command will not overwrite an existing file. If you want to overwrite an existing file, you need to use the keyword `REPLACE`.
- To add the buffer to the end of an existing file, use the `SAVE filename APPEND` command.
- You can edit the saved file using the `EDIT filename` command.
- You can bring the contents of a *script file* to the SQL buffer using the `GET filename` command.
- If you want to run a script file, use the command `START filename`. You can also run a script file using `@filename`.
- An `@@filename` used inside a script file looks for the filename in the directory where the parent *script file* is saved and executes it.

Exercise 1.1 will familiarize you with the script file commands, as well as the other topics covered so far.

EXERCISE 1.1**Practicing SQL*Plus File Commands**

In this exercise, you will learn how to edit the SQL*Plus buffer using various buffer edit commands.

1. Enter the following SQL code; the third line is a blank line so that the SQL code is saved in the buffer:

```
SQL> SELECT employee_id, first_name, last_name
  2  FROM    employees
  3
SQL>
```

2. List the SQL buffer:

```
SQL> L
  1  SELECT employee_id, first_name, last_name
  2* FROM    employees
SQL>
```

3. Save the buffer to a file named myfile; the default extension will be .sql:

```
SQL> SAVE myfile
Created file MYFILE.sql
SQL>
```

4. Choose to edit the file:

```
SQL> EDIT myfile
SQL>
```

5. Add WHERE EMPLOYEE_ID = 106 as the third line to the SQL statement.

6. List the buffer:

```
SQL> LIST
  1  SELECT employee_id, first_name, last_name
  2* FROM    employees
SQL>
```

The buffer listed is still the old buffer. The edited changes are not reflected because you edited the file MYFILE, which is not yet loaded to the buffer.

EXERCISE 1.1 (*continued*)

7. Bring the file contents to the buffer:

```
SQL> GET myfile
  1  SELECT employee_id, first_name, last_name
  2  FROM   employees
  3* WHERE employee_id = 106
SQL>
```

8. List the buffer to verify its contents:

```
SQL> LI
  1  SELECT employee_id, first_name, last_name
  2  FROM   employees
  3* WHERE employee_id = 106
SQL>
```

9. Change the employee number from 106 to 110:

```
SQL> C/106/110
  3* WHERE employee_id = 110
SQL>
```

10. Save the buffer again to the same file:

```
SQL> SAVE myfile
SP2-0540: File "MYFILE.sql" already exists.
Use "SAVE filename[.ext] REPLACE".
SQL>
```

An error is returned, because SAVE will not overwrite the file by default.

11. Save the file using the REPLACE keyword:

```
SQL> SAVE myfile REPLACE
Wrote file MYFILE.sql
SQL>
```

12. Execute the file:

```
SQL> START myfile
```

EXERCISE 1.1 (*continued*)

```
EMPLOYEE_ID FIRST_NAME          LAST_NAME
----- -----
    110  John                  Chen
SQL>
```

13. Change the employee number from 110 to 106, and append this SQL code to the file; then execute it using @:

```
SQL> C/110/106
 3* WHERE employee_id = 106
SQL> SAVE myfile APPEND
Appended file to MYFILE.sql
SQL> @MYFILE
EMPLOYEE_ID FIRST_NAME          LAST_NAME
----- -----
    110  John                  Chen

EMPLOYEE_ID FIRST_NAME          LAST_NAME
----- -----
    106  Valli                 Pataballa
SQL>
```

Saving Query Results to a File

You can use the SPOOL *filename* command to save the query results to a file. By default, the SPOOL command creates a .lst file extension. SPOOL overwrites an existing file by default. If you include the APPEND option—as in SPOOL *filename* APPEND—the results are added to an existing file. A new file will be created if the file does not exist already.

SPOOL OFF stops writing the output to the file. SPOOL OUT stops the writing of output and sends the output file to the printer. SPOOL with no clauses lists the current spooling status.

Adding Comments to a Script File

Comments in the script file can improve readability and make the code more understandable. You can enter comments in SQL*Plus using the REMARKS (abbreviated REM) command. Lines in the script file beginning with the keyword REM are comments and are not executed. You can also enter a comment between /* and */. Comments can also be entered following -- (double hyphen); all characters following -- in the line are treated as comments by Oracle.

While a script file with comments is being executed, the remarks entered using the REMARKS command are not displayed on the screen, but the comments within /* and */ are displayed on the screen with the prefix DOC> when there is more than one line between /* and */. You can turn this off by using SET DOCUMENT OFF.

Now that you understand the concepts of RDBMS and how Oracle Database 12c helps organizations achieve the cloud architecture, let's move on to the core of the Oracle Database 12c SQL Fundamentals OCA exam in the coming chapters. Before moving on to Chapter 2, "Introducing SQL," please make sure you have an Oracle Database 12c to practice on and try out the examples.

You may perform a quick default install of the database after downloading the software from OTN (www.technet.oracle.com).



Real World Scenario

Install Oracle Database 12c for SQL Practice

To be able to practice the examples provided in this book and to familiarize yourself with Oracle Database 12c SQL, an Oracle Database 12c database must be available to you. If you do not have such a database, you can follow these instructions to install software and create databases on a Windows machine.

Download and Install Software

You may download Oracle Database 12c software from Oracle Technology Network (OTN) or from Oracle Cloud Delivery service (edelivery.oracle.com). After downloading the software, you can invoke the setup.exe to install software. For detailed instructions on downloading and installing Oracle software, refer to www.bijoos.com/certify/db12csw.pdf. You can also refer to Chapter 9 to install database software.

Create Oracle Database

Databases are created using the Database Configuration Assistant tool. You can choose the Create Database With Default Configuration option to create a database quickly. For detailed instructions on creating a database, refer to www.bijoos.com/certify/db12c_ndb.pdf. You can also refer to Chapter 9 to create a database.

Create Sample Schema

The sample schema provided by Oracle includes HR, OE, PM, SH, and IX. For the majority of the SQL used in the book, the HR schema is used. If you did not install the sample schema during database creation, you can do so using the following procedure.

When you install Oracle software, you can choose the Create Database With Default Configuration option, but this will not include the sample schemas. The account SYS is the Oracle dictionary owner, and SYSTEM is a database administrator (DBA) account. Initially, the sample schema user accounts are locked. You need to log in to the database using SQL*Plus as the SYSTEM user and then unlock the account using the ALTER USER statement. To unlock the HR schema, use `ALTER USER hr IDENTIFIED BY hrpassword ACCOUNT UNLOCK;`. Now you can log in to the database using the hr user with the password hrpassword. Remember, the password is case sensitive by default.

To install the sample schemas in an existing Oracle Database 12c, follow the instructions in the Oracle document “Oracle Database Sample Schemas 12c Release 1 (12.1) Part E15979-04” at <http://docs.oracle.com/>. Chapter 2 of this document provides instructions on how to install the sample schemas using Database Configuration Assistant (DCA) as well as on running scripts. The same chapter also gives you steps to reinitialize the sample schema data.

The manual installation of HR and OE sample data on Linux-based Oracle Database 12c databases can be quickly summarized as:

Change the directory to \$ORACLE_HOME/demo/schema/human_resources.

Connect to database using SQL*Plus as SYSDBA (sqlplus sys@mydb as sysdba).

Run the schema and objects creation script (@hr_main.sql).

Change the directory to \$ORACLE_HOME/demo/schema/order_entry.

Connect to the database using SQL*Plus as SYSDBA (sqlplus sys@mydb as sysdba).

Run the schema and objects creation script (@oe_main.sql).

Summary

This chapter reviewed the concepts of relational database systems and Object RDBMS. You also learned how Oracle implements the RDBMS and relational theory into the Oracle database. The entity-relationship diagram is a modeling tool used in the beginning stages of application development.

You also learned about the high-level architecture and various implementations of Oracle, such as single database, RAC database, and container database.

Oracle Database 12c is cloud enabled. The multitenant architecture of the database helps to consolidate multiple Oracle databases (pluggable databases) into a single container database.

Various tools are available for the DBA to connect to the Oracle database and administer it. SQL*Plus is Oracle’s SQL command-line interface tool. SQL Developer is a graphical tool, with ease of navigation and predefined tasks. You also saw an overview of SQL*Plus in this chapter, including how to connect to the database using SQL*Plus and basic editing commands.

SQL*Plus supports all SQL statements and has its own formatting and enhancement commands. Using this tool, you can produce interactive SQL statements and formatted reports. SQL*Plus is the command-line interface to the database widely used by DBAs. SQL*Plus has its own buffer where SQL statements are buffered. You can edit the buffer using SQL*Plus editing commands. The DESCRIBE command is used to get information on a table, view, function, or procedure. Multiple SQL and SQL*Plus commands can be stored in a file and can be executed as a unit. Such files are called script files.

Exam Essentials

Know RDBMS Concepts. Review the RDBMS concepts. Understand entities and relationships.

Understand what structures make Object RDBMS. Learn how Oracle implements the object relational database management system.

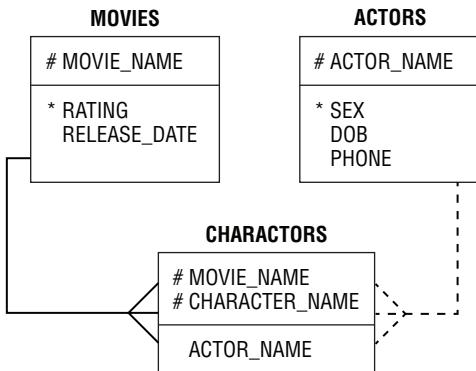
Know the tools. Have an understanding of what tools are available for database management in Oracle and their purposes.

Learn the various architectures Oracle Database 12c can implement. Oracle database can be installed as a single instance single database, multiple instance RAC database, or multi-tenant container database.

Identify Oracle Database 12c cloud features. Know the features of Oracle Database 12c that make cloud implementation easier.

Review Questions

1. Look at the diagram. What kind of relationship exists between MOVIES and CHARACTERS?



- A. Each movie may have one or more characters.
 - B. Each movie must have one or more characters.
 - C. Many movies may have many characters.
 - D. One movie can have only one character.
2. When the physical model is being designed from the logical model, which element may be attributed as a table from the ER diagram?
- A. Relationship
 - B. Attribute
 - C. Unique identifier
 - D. Entity
3. Which statement about the object type is true?
- A. They are structures that consist of built-in or user-defined data types.
 - B. They are structures that consist of only built-in data types.
 - C. They are structures that consist of only user-defined data types.
 - D. Only one column in a table can be object type.
4. Which of the following is not a benefit of Oracle Database 12c?
- A. Manage multiple databases as one
 - B. Fast provisioning of cloned databases
 - C. Plug and unplug databases
 - D. Patch each pluggable database separately

5. Which one of the following Oracle SQL*Plus command lines is not valid?
 - A. sqlplus <username>
 - B. sqlplus @<connect_string>
 - C. sqlplus <username>@<connect_string>
 - D. sqlplus
6. Which database tools are parts of Oracle Database 12c? Choose two.
 - A. Oracle Enterprise Manager Cloud Control 12c
 - B. Oracle Enterprise Manager Database Express 12c
 - C. SQL Developer
 - D. TOAD (Tool for Oracle Application Developers)
7. In the physical implementation of RDBMS, which database object is used to represent unique identifiers?
 - A. Any constraint
 - B. Index
 - C. Primary key
 - D. Foreign key
8. SQL Developer is a tool primarily for whom?
 - A. Database administrators
 - B. End users
 - C. Application developers
 - D. All of the above
9. Which architecture in the Oracle Database 12c implementation guards against unplanned machine downtime?
 - A. Multitenancy Container Database
 - B. Real Application Clusters
 - C. Consolidate multiple databases and instances to one server
 - D. None of the above
10. Which connection method to the Oracle database is known as the easy connect?
 - A. <username>@<connect_string>
 - B. <username>@<host>:<port>/<service_name>
 - C. Both A and B
 - D. Neither A or B

Chapter 2



Introducing SQL

ORACLE DATABASE 12c: SQL FUNDAMENTALS EXAM OBJECTIVES COVERED IN THIS CHAPTER:

✓ Retrieving Data Using the SQL SELECT Statement

- Explain the capabilities of SQL SELECT statements.
- Execute a basic SELECT statement.

✓ Restricting and Sorting Data

- Limit the rows that are retrieved by a query.
- Sort the rows that are retrieved by a query.
- Use ampersand substitution to restrict and sort output at runtime.



Oracle Database 12*c* is a very powerful and feature-rich relational database management system (RDBMS). SQL has been adopted by most RDBMSs for the retrieval and management of data, schema creation, and access control. The American National Standards Institute (ANSI) has been refining standards for the SQL language for more than 25 years. Oracle, like many other companies, has taken the ANSI standard of SQL and extended it to include much additional functionality.

SQL is the basic language used to manipulate and retrieve data from Oracle Database 12*c*. SQL is a nonprocedural language, meaning it does not have programmatic constructs such as loop structures. PL/SQL is Oracle's procedural extension of SQL, and SQLJ allows embedded SQL operations in Java code. The scope of the Oracle Database 12*c* SQL Fundamentals test includes only SQL.

In this chapter, we will discuss Oracle SQL fundamentals such as the various types of SQL statements, introduce SQL*Plus and a few SQL*Plus commands, and discuss SELECT statements.

You will learn how to write basic SQL statements to retrieve data from tables. This will include coverage of SQL SELECT statements, which are used to query data from the database-storage structures, such as tables and views. You will also learn how to limit the information retrieved and to display the results in a specific order.

SQL Fundamentals

SQL is the standard language used to query and modify data as well as manage databases. SQL is the common language used by programmers, database administrators, and users to access and manipulate data as well as to administer databases. To get started with SQL in this chapter, we will use the sample HR schema supplied with Oracle Database 12*c*.

SQL statements are like plain English but with specific syntax. SQL is a simple, yet powerful language used to create, access, and manipulate data and structures in the database. SQL statements can be categorized as listed in Table 2.1.

TABLE 2.1 SQL Statement Categories

SQL Category	Description
Data Manipulation Language (DML)	Used to access, insert, modify, or delete data in the existing structures of the database. DML statements include those used to query information (SELECT), add new rows (INSERT), modify existing rows (UPDATE), delete existing rows (DELETE), perform a conditional update or insert operation (MERGE), see an execution plan of SQL (EXPLAIN PLAN), and lock a table to restrict access (LOCK TABLE). Including the SELECT statement in the DML group is debatable within the SQL community, because SELECT does not modify data.
Data Definition Language (DDL)	Used to define, alter, or drop database objects and their privileges. DDL statements include those used to create, modify, drop, or rename objects (CREATE, ALTER, DROP, RENAME), remove all rows from a database object without dropping the structure (TRUNCATE), manage access privileges (GRANT, REVOKE), audit database use (AUDIT, NOAUDIT), and add a description about an object to the dictionary (COMMENT).
Transaction Control	Used to group a set of DML statements as a single transaction. Using these statements, you can save the changes (COMMIT) or discard the changes (ROLLBACK) made by DML statements. Also included in the transaction-control statements are statements to set a point or marker in the transaction for possible rollback (SAVEPOINT) and to define the properties for the transaction (SET TRANSACTION).
Session Control	Used to control the properties of a user session. (A session is the point from which you are connected to the database until you disconnect.) Session-control statements include those to control the session properties (ALTER SESSION) and to enable/disable roles (SET ROLE).
System Control	Used to manage the properties of the database. There is only one statement in this category (ALTER SYSTEM).

Table 2.1 provides an overview of all the statements that will be covered in this book. Do not worry if you do not understand certain terms, such as *role*, *session*, *privilege*, and so on. We will cover all the statements in the coming chapters and include many examples. In this chapter, we will begin by writing simple statements to query the database (SELECT statements), but we'll go over some fundamentals first.

This section provided an overview of SQL*Plus, the tool you will be using to enter and execute SQL statements in Oracle Database 12c. In the next sections, we will discuss some of the Oracle Database 12c SQL fundamentals before showing you how to write your first SQL query (a SELECT statement).

Oracle Datatypes

The basic structure of data storage in Oracle Database 12c is a table. A table can be considered as a spreadsheet with columns and rows. Data is stored in the table as rows. Each column in the table has storage characteristics such as the type of data contained in the column. Oracle has several built-in datatypes to store different kinds of data. In this section, we will go over the built-in datatypes available in Oracle Database 12c. Chapter 7, “Creating Tables and Constraints,” includes a detailed discussion of datatypes and how to create and maintain tables.

When you create a table to store data in the database, you need to specify a datatype for all the columns you define in that table. Oracle has many datatypes to suit application requirements. Oracle Database 12c also supports ANSI and DB2 datatypes. The Oracle built-in datatypes can be broadly classified as shown in Table 2.2.

TABLE 2.2 Oracle Built-in Datatypes

Category	Datatypes
Character	CHAR, NCHAR, VARCHAR2, NVARCHAR2
Number	NUMBER, FLOAT, BINARY_FLOAT, BINARY_DOUBLE
Long and raw	LONG, LONG RAW, RAW
Date and time	DATE, TIMESTAMP, TIMESTAMP WITH TIME ZONE, TIMESTAMP WITH LOCAL TIME ZONE, INTERVAL YEAR TO MONTH, INTERVAL DAY TO SECOND
Large object	CLOB, NCLOB, BCLOB, BFILE
Row ID	ROWID, UROWID

In the following sections, we will discuss only a few of the built-in datatypes to get you started with SQL. We discuss all the datatypes and their usage in detail in Chapter 7.

CHAR(<size>) and VARCHAR2(<size>)

The *CHAR* datatype is a fixed-length alphanumeric string, which has a maximum length in bytes (to specify length in characters, use the *CHAR* keyword inside parentheses along with a size; see Chapter 7). Data stored in *CHAR* columns is space-padded to fill the maximum length. Its size can range from a minimum of 1 byte to a maximum of 2,000 bytes. The default size is 1.

When you create a column using the *CHAR* datatype, the database will ensure that all data placed in this column has the defined length. If the data is shorter than the defined length, it is space-padded on the right to the specified length. If the data inserted is longer than the column length, an error is raised.

The *VARCHAR2* datatype is a variable-length alphanumeric string, which has a maximum length in bytes (to specify the length in characters, use the *CHAR* keyword inside parentheses along with a size; see Chapter 7). *VARCHAR2* columns require only the amount of space needed to store the data and can store up to 32KB (32,767 bytes). There is no default size for the *VARCHAR2* datatype. An empty *VARCHAR2(2000)* column takes up as much room in the database as an empty *VARCHAR2(1)* column.

VARCHAR2 columns by default allow only 4,000 bytes or characters. To be able to store more than 4,000 bytes or characters, the *MAX_STRING_SIZE* database parameter needs to be set to *EXTENDED*. The default value of this parameter is *STANDARD*, which allows only 4,000 bytes in a *VARCHAR2* column.



The default size of a *CHAR* datatype is 1. For a *VARCHAR2* datatype, you must always specify the size.

The *VARCHAR2* and *CHAR* datatypes have different comparison rules for trailing spaces. With the *CHAR* datatype, trailing spaces are ignored. With the *VARCHAR2* datatype, trailing spaces are not ignored, and they sort higher than no trailing spaces. Here's an example:

```
CHAR datatype: 'Yo' = 'Yo      '
VARCHAR2 datatype: 'Yo' < 'Yo      '
```

NUMBER (<p>, <s>)

The *NUMBER* datatype stores numbers with a precision of *<p>* digits and a scale of *<s>* digits. The precision and scale values are optional. Numeric datatypes are used to store negative and positive integers, fixed-point numbers, and floating-point numbers. The precision can be between 1 and 38, and the scale has a range between -84 and 127. If the precision and scale are omitted, Oracle assumes the maximum of the range for both values.

You can have precision and scale digits in the integer part. The scale rounds the value after the decimal point to *<s>* digits. For example, if you define a column as *NUMBER(5,2)*, the range of values you can store in this column is from -999.99 to 999.99—that is, 5 – 2 = 3 for the integer part, and the decimal part is rounded to two digits. Even if you do not include the decimal part for the value inserted, the maximum number you can store in a *NUMBER(5,2)* definition is 999.

Oracle will round numbers inserted into numeric columns with a scale smaller than the inserted number. For example, if a column were defined as NUMBER(4,2) and you specified a value of 12.125 to go into that column, the resulting number would be rounded to 12.13 before it was inserted into the column. If the value exceeds the precision, however, an Oracle error is returned. You cannot insert 123.1 into a column defined as NUMBER(4,2). Specifying the scale and precision does not force all inserted values to be a fixed length.

If the scale is negative, the number is rounded to the left of the decimal. Basically, a negative scale forces $<s>$ number of zeros just to the left of the decimal.

If you specify a scale that is greater than the precision value, the precision defines the maximum number of digits to the right of the decimal point after the zeros. For example, if a column is defined as NUMBER(3,5), the range of values you can store is from -0.00999 to 0.00999—that is, it requires two zeros ($<s>-<p>$) after the decimal point and rounds the decimal part to three digits ($<p>$) after zeros. Table 2.3 shows several examples of how numeric data is stored with various definitions.

TABLE 2.3 Precision and Scale Examples

Value	Datatype	Stored Value	Explanation
123.2564	NUMBER	123.2564	The range and precision are set to the maximum, so the datatype can store any value.
1234.9876	NUMBER(6,2)	1234.99	Because the scale is only 2, the decimal part of the value is rounded to two digits.
12345.12345	NUMBER(6,2)	Error	The range of the integer part is only from -9999 to 9999.
123456	NUMBER(6,2)	Error	The precision is larger than specified; the range is only from -9999 to 9999.
1234.9876	NUMBER(6)	1235	The decimal part is rounded to the next integer.
123456.1	NUMBER(6)	123456	The decimal part is rounded.
12345.345	NUMBER(5,-2)	12300	The negative scale rounds the number $<s>$ digits left to the decimal point. -2 rounds to hundreds.
1234567	NUMBER(5,-2)	1234600	Rounded to the nearest hundred.
12345678	NUMBER(5,-2)	Error	Outside the range; can have only five digits, excluding the two zeros representing hundreds, for a total of seven digits: $(s - (-p)) = s + p = 5 + 2 = 7$.

Value	Datatype	Stored Value	Explanation
123456789	NUMBER(5,-4)	123460000	Rounded to the nearest 10,000.
1234567890	NUMBER(5,-4)	Error	Outside the range; can have only five digits, excluding the four trailing zeros.
12345.58	NUMBER(*, 1)	12345.6	The use of * in the precision specifies the default limit (38).
0.1	NUMBER(4,5)	Error	Requires a zero after the decimal point ($5 - 4 = 1$).
0.01234567	NUMBER(4,5)	0.01235	Rounded to four digits after the decimal point and zero.
0.09999	NUMBER(4,5)	0.09999	Stored as it is; only four digits after the decimal point and zero.
0.099996	NUMBER(4,5)	Error	Rounding this value to four digits after the decimal and zero results in 0.1, which is outside the range.

DATE

The *DATE* datatype is used to store date and time information. This datatype can be converted to other forms for viewing, but it has a number of special functions and properties that make date manipulation and calculations simple. The time component of the *DATE* datatype has a resolution of one second—no less. The *DATE* datatype occupies a storage space of 7 bytes. The following information is contained within each *DATE* datatype:

- Century
- Year
- Month
- Day
- Hour
- Minute
- Second

DATE datatype does not have fractional seconds or time zone. Date values are inserted or updated in the database by converting either a numeric value or a character value into a *DATE* datatype using the function *TO_DATE*. Oracle defaults the format to display the date as *DD-MON-YY* (the format is determined by parameter *NLS_DATE_FORMAT*). This format shows that the default date must begin with a two-digit day, followed by a three-character abbreviation for the month, followed by a two-digit year. If you specify the date without

including a time component, the time is defaulted to midnight, or 00:00:00 in military time. The SYSDATE function returns the current system date and time from the database server to which you're currently connected.

TIMESTAMP [<precision>]

The *TIMESTAMP* datatype stores date and time information with fractional precision for seconds. The only difference between the DATE and TIMESTAMP datatypes is the ability to store fractional seconds up to a precision of nine digits. The default precision is 6 and can range from 0 to 9. Similar to the SYSDATE function, the SYSTIMESTAMP function returns the current system date and time, with fractional precision for seconds.

Operators and Literals

An *operator* is a manipulator that is applied to a data item in order to return a result. Special characters represent different operations in Oracle (+ represents addition, for example). Operators are commonly used in all programming environments, and you should already be familiar with the following operators, which may be classified into two types:

Unary Operator A unary operator has only one operand. It has the format *<operator><operand>*. Examples are +2 and -5.

Binary Operator A binary operator has two operands. It has the format *<operand1><operator><operand2>*. You can insert spaces between the operand and operator to improve readability. Examples are $5 + 4$ and 7×5 .

We'll now discuss the various types of operators available in Oracle.

Arithmetic Operators

Arithmetic operators operate on numeric values. Table 2.4 shows the various arithmetic operators in Oracle and how to use them.

TABLE 2.4 Arithmetic Operators

Operator	Purpose	Example
+ -	Unary operators: Used to represent positive or negative data item. For positive items, the + is optional.	-234.44
+	Addition: Used to add two data items or expressions.	2+4
-	Subtraction: Used to find the difference between two data items or expressions.	20.4-2
*	Multiplication: Used to multiply two data items or expressions.	5*10
/	Division: Used to divide a data item or expression with another.	8.4/2



Do not use two hyphens (--) to represent double negation; use a space or parentheses in between, as in `-(-20)`. Two hyphens represent the beginning of a comment in SQL.

Concatenation Operator

The *concatenation operator* is used to concatenate or join two character (text) strings. The result of concatenation is another character string. Concatenating a zero-length string (' ') or a NULL with another string results in a string, not a NULL (NULL in Oracle Database 12c represents unknown or missing data). Two vertical bars (||) are used as the concatenation operator.

Here are two examples:

```
'Oracle12c' || 'Database' results in 'Oracle12cDatabase'.  
'Oracle12c ' || 'Database' results in 'Oracle12c Database'.
```

Operator Precedence

If multiple operators are used in the same expression, Oracle evaluates them in the *order of precedence* set in the database engine. Table 2.5 lists the precedence. Operators with higher precedence are evaluated before operators with lower precedence. Operators with the same precedence are evaluated from left to right.

TABLE 2.5 SQL Operator Precedence

Precedence	Operator	Purpose
1	- +	Unary operators, negation
2	* /	Multiplication, division
3	+ -	Addition, subtraction, concatenation

Using parentheses changes the order of precedence. The innermost parenthesis is evaluated first. In the expression $1+2*3$, the result is 7, because 2×3 is evaluated first and the result is added to 1. In the expression $(1+2)*3$, $1+2$ is evaluated first, and the result is multiplied by 3, giving 9.

Literals

A *literal* is a value that represents a fixed value (constant). There are four types of literals:

- Text (or character)
- Numeric (integer and number)

- Datetime
- Interval

You can use literals within many of the SQL functions, expressions, and conditions.

Text Literals

A *text literal* must be enclosed in single quotation marks. Any character between the quotation marks is considered part of the text value. Oracle treats all text literals as though they were CHAR datatypes for comparison (blank padded). The maximum length of a text literal is 4,000 bytes if the database parameter MAX_STRING_SIZE is STANDARD. Single quotation marks can be included in the literal text value by preceding it with another single quotation mark. Here are some examples of text literals:

```
'The Quick Brown Fox'  
'That man''s suit is black'  
'And I quote: "This will never do." '  
'12-SEP-2011'
```

Alternatively, you can use Q or q quoting, which provides a range of delimiters. The syntax for using the Q/q quoting with a quote-delimiter text literal is as follows:

```
[Q|q]' <quote_delimiter> <text literal> <quote_delimiter>'
```

<quote_delimiter> is any character except a space, tab, or carriage return. The quote delimiter can be a single quotation mark, but make sure inside the text literal a single quotation mark is not immediately followed by another single quotation mark. If the opening quote delimiter is [or { or < or (, then the closing quote must be the corresponding] or } or > or). For all other quote delimiters, the opening quote delimiter must be the same as the closing quote delimiter. Here are some examples of text literals using the alternative quoting mechanism:

```
q'<The Quick Brown Fox>'  
Q'#The Quick Brown Fox#'  
q'{That man's suit is black}'  
Q'(And I quote: "This will never do." )'  
Q'"And I quote: "This will never do." "'  
q'[12-SEP-2001]'
```

Numeric Literals

Integer literals can be any number of numerals, excluding a decimal separator and up to 38 digits long. Here are two examples:

- 24
- -456

Number literals and *floating-point literals* can include scientific notation, as well as digits and the decimal separator. E or e represents a number in scientific notation; the exponent can be in the range of -130 to 125. If the literal is followed by an F or f, it is treated as a BINARY_FLOAT datatype. If the literal is followed by a D or d, it is treated as a BINARY_DOUBLE datatype. Here are some examples:

- 24.0
- -345.65
- 23E-10
- 1.5f
- -34.567D
- -4d
- -4.0E+0

Datetime Literals

You can specify a date value as a string literal using the *datetime literals*. The most common methods to represent the datetime values are to use the conversion function TO_DATE or TO_TIMESTAMP with the appropriate format mask. To complete this discussion of literals, we will discuss the datetime literals briefly.

The DATE literal uses the keyword DATE followed by the date value in single quotes, and the value must be specified in YYYY-MM-DD format with no time component. The time component will be defaulted to midnight (00:00:00). The following are examples of the DATE literal:

```
DATE '2008-03-24'  
DATE '1999-12-31'
```

Similar to the TIMESTAMP datatype, the TIMESTAMP literal can be used to specify the year, month, date, hour, minute, second, and fractional second. You can also include time-zone data along with the TIMESTAMP literal. The time zone information can be specified using the UTC offset or using the time-zone region name. The literal must be in the format YYYY-MM-DD HH24:MI:SS TZ.

Here are some examples of the TIMESTAMP literal:

```
TIMESTAMP '2008-03-24 03:25:34.123'  
TIMESTAMP '2008-03-24 03:25:34.123 -7:00'  
TIMESTAMP '2008-03-24 03:25:34.123 US/Central'  
TIMESTAMP '2008-03-24 03:25:34.123 US/Central CDT'
```

Interval Literals

Interval literals specify a period of time in terms of years and months or in terms of days and seconds. These literals correspond to the Oracle datatypes INTERVAL YEAR TO MONTH and INTERVAL DAY TO SECOND. We'll discuss these datatypes in more detail in Chapter 7.

Writing Simple Queries

A *query* is a request for information from the database tables. Queries do not modify data; they read data from database tables and views. Simple queries are those that retrieve data from a single table or view. A table is used to store data and is stored in rows and columns. The basis of a query is the SELECT statement. The SELECT statement can be used to get data from a single table or from multiple tables. Queries using multiple tables are discussed in later chapters.

Using the SELECT Statement

The SELECT statement is the most commonly used statement in SQL. It allows you to retrieve information already stored in the database. The statement begins with the keyword SELECT, followed by the names of the columns with data you want to query. You can select information either from all the columns (denoted by *) or from name-specific columns in the SELECT clause to retrieve data. When * is used in the column projection, it lists the columns in the same order as they are defined in the table definition. The FROM clause provides the name of the table, view, or materialized view to use in the query. These objects are discussed in detail in later chapters. For simplicity, we will use tables for the rest of this chapter.

The simple SELECT statement at a high level is of the form:

```
SELECT column_list
FROM table
WHERE filtering_conditions
ORDER BY column_list
OFFSET n_rows
FETCH row_limiting_conditions
```

The SELECT clause lists the columns from the table you are interested in reading. The FROM clause identifies the table name. The SELECT and FROM clauses are mandatory, and the rest of the clauses are optional. The FETCH clause is used to limit rows; the OFFSET clause is used to skip rows; the WHERE clause is used for filtering; and the ORDER BY clause is for sorting. We will discuss all of these clauses in detail in the coming sections of this chapter.

Let's use the JOBS table defined in the HR schema of Oracle Database 12c. You can use the SQL*Plus tool to connect to the database as discussed earlier in the chapter. The JOBS table definition is provided in Table 2.6.

TABLE 2.6 JOBS Table Definition

Column Name	Datatype	Length
JOB_ID	VARCHAR2	10
JOB_TITLE	VARCHAR2	35

Column Name	Datatype	Length
MIN_SALARY	NUMBER	6,0
MAX_SALARY	NUMBER	6,0

The simple form of a SELECT statement to retrieve all the columns and rows from the JOBS table is as follows (only part of output result set is shown here):

```
SQL> SELECT * FROM jobs;
```

JOB_ID	JOB_TITLE	MIN_SALARY	MAX_SALARY
AD_PRES	President	20080	40000
AD_VP	Administration Vice President	15000	30000
AD_ASST	Administration Assistant	3000	6000
FI_MGR	Finance Manager	8200	16000
FI_ACCOUNT	Accountant	4200	9000
<hr/>			
IT_PROG	Programmer	4000	10000
MK_MAN	Marketing Manager	9000	15000
MK_REP	Marketing Representative	4000	9000
HR_REP	Human Resources Representative	4000	9000
PR_REP	Public Relations Representative	4500	10500

19 rows selected.



The keywords, column names, and table names are not case sensitive. Only literals enclosed in single quotation marks are case sensitive in Oracle.

How do you list only the job title and minimum salary from this table? If you know the column names and the table name, writing the query is simple. Here, the column names are JOB_TITLE and MIN_SALARY, and the table name is JOBS. Execute the query by ending the query with a semicolon. In SQL*Plus, you can execute the query by entering a slash on a line by itself or by using the RUN command.

```
SQL> SELECT job_title, min_salary FROM jobs;
```

JOB_TITLE	MIN_SALARY
President	20080
Administration Vice President	15000

Administration Assistant	3000
Finance Manager	8200
Accountant	4200
Accounting Manager	8200
Public Accountant	4200
...	
Programmer	4000
Marketing Manager	9000
Marketing Representative	4000
Human Resources Representative	4000
Public Relations Representative	4500

19 rows selected.

Notice that the numeric column (`MIN_SALARY`) is aligned to the right, and the character column (`JOB_TITLE`) is aligned to the left. If you want the column heading `MIN_SALARY` to be more meaningful, you can provide a *column alias* to appear in the query results.

Column Alias Names

The column alias name is defined next to the column name with a space or by using the keyword AS. If you want a space in the column alias name, you must enclose it in double quotation marks. The case is preserved only when the alias name is enclosed in double quotation marks; otherwise, the display will be uppercase. The following example demonstrates using an alias name for the column heading in the previous query:

```
SELECT job_title AS Title, min_salary AS "Minimum Salary"  
FROM jobs;
```

TITLE	Minimum Salary
President	20080
Administration Vice President	15000
Administration Assistant	3000
Finance Manager	8200
Accountant	4200
Accounting Manager	8200
...	
Programmer	4000
Marketing Manager	9000
Marketing Representative	4000
Human Resources Representative	4000

```
Public Relations Representative      4500
```

19 rows selected.

In this listing, the column alias name Title appears in all capital letters because it was not enclosed in double quotation marks.



The asterisk (*) is used to select all columns in the table. This is useful when you do not know the column names or when you are too lazy to type all the column names. The invisible columns and pseudo columns are not included in *.

Ensuring Uniqueness

The DISTINCT keyword (or UNIQUE keyword) following SELECT ensures that the resulting rows are unique. Uniqueness is verified against the complete row, not the first column. If you need to find the unique departments in the EMPLOYEES table, issue this query:

```
SELECT DISTINCT department_id  
FROM employees;
```

```
DEPARTMENT_ID  
-----
```

```
100
```

```
30
```

```
90
```

```
20
```

```
70
```

```
110
```

```
50
```

```
80
```

```
40
```

```
60
```

```
10
```

12 rows selected.

To demonstrate that uniqueness is enforced across the row, let's do one more query using the SELECT DISTINCT clause. Notice DEPARTMENT_ID repeating for each JOB_ID value in the following example:

```
SELECT DISTINCT department_id, job_id  
FROM employees;
```

```
DEPARTMENT_ID JOB_ID
-----
110 AC_ACCOUNT
90 AD_VP
50 ST_CLERK
80 SA_REP
110 AC_MGR
...
10 AD_ASST
20 MK_REP
40 HR_REP
30 PU_MAN
```

20 rows selected.



SELECT * FROM TAB; shows all the tables and views in your schema. Don't be alarmed if you see a table name similar to BIN\$PJY23QpwQfu0zPN9uaXw+w==\$0. These are tables that belong to the Recycle Bin (or dropped tables). The tasks of creating tables and managing tables are discussed in Chapter 7.

The DUAL Table

The DUAL table is a special table available to all users in the database. It has one column and one row. Oracle optimized the database so that it no longer performs physical or logical input/output on the DUAL table. The DUAL table is mostly used to select system variables or to evaluate an expression. Here are a few examples. The first query is to show the contents of the DUAL table.

```
SQL> SELECT * FROM dual;
```

```
D
-
X
```

```
SQL> SELECT SYSDATE, USER FROM dual;
```

```
SYSDATE      USER
-----
13-JUL-13    HR
```

```
SQL> SELECT 'I''m ' || user || ' Today is ' || SYSDATE  
2  FROM dual;
```

```
'I''M'||USER||'TODAYIS'||SYSDATE
```

```
-----  
I'm HR Today is 13-JUL-13
```

```
SQL>
```



SYSDATE and USER are built-in functions that provide information about the environment. These functions are discussed in Chapter 3, “Using Single-Row Functions.”

Filtering Rows

You can use the WHERE clause in the SELECT statement to restrict the number of rows processed. Any logical conditions of the WHERE clause use the comparison operators. Rows are returned or operated upon where the data satisfies the logical condition(s) of the WHERE clause. You can use column names or expressions in the WHERE clause, but not column alias names. The WHERE clause follows the FROM clause in the SELECT statement.

How do you list the employees who work for department 90? The following example shows how to limit the query to only the records belonging to department 90 by using a WHERE clause:

```
SELECT first_name || ' ' || last_name "Name", department_id  
FROM   employees  
WHERE  department_id = 90;
```

Name	DEPARTMENT_ID
Steven King	90
Neena Kochhar	90
Lex De Haan	90



You need not include the column names in the SELECT clause to use them in the WHERE clause.

You can use various operators in Oracle Database 12c in the WHERE clause to limit the number of rows.

Comparison Operators

Comparison operators compare two values or expressions and give a Boolean result of TRUE, FALSE, or NULL. The comparison operators include those that test for equality, inequality, less than, greater than, and value comparisons.

= (Equality)

The = operator tests for equality. The test evaluates to TRUE if the values or results of an expression on both sides of the operator are equal. The following shows an example of finding all the employees belonging to department 90—that is, the department ID is equal to 90:

```
SELECT first_name || ' ' || last_name "Name", department_id  
FROM   employees  
WHERE  department_id = 90;
```

Name	DEPARTMENT_ID
Steven King	90
Neena Kochhar	90
Lex De Haan	90

!=, <>, or ^= (Inequality)

You can use any one of these three operators to test for inequality. The test evaluates to TRUE if the values on both sides of the operator do not match. The following shows an example of querying employees whose commission is not the default 35 percent—that is, to find all employees with a commission percent that is not 35:

```
SELECT first_name || ' ' || last_name "Name", commission_pct  
FROM   employees  
WHERE  commission_pct != .35;
```

Name	COMMISSION_PCT
John Russell	.4
Karen Partners	.3
Alberto Errazuriz	.3
Gerald Cambrault	.3
...	
Jack Livingston	.2
Kimberely Grant	.15
Charles Johnson	.1

32 rows selected.

< (Less Than)

The < operator evaluates to TRUE if the left side (expression or value) of the operator is less than the right side of the operator. The following shows an example of employees with a commission percent less than 15:

```
SELECT first_name || ' ' || last_name "Name", commission_pct
FROM   employees
WHERE  commission_pct < .15;
```

Name	COMMISSION_PCT
Mattea Marvins	.1
David Lee	.1
Sundar Ande	.1
Amit Banda	.1
Sundita Kumar	.1
Charles Johnson	.1

6 rows selected.

> (Greater Than)

The > operator evaluates to TRUE if the left side (expression or value) of the operator is greater than the right side of the operator. The following shows an example of employees with a commission percent over 35:

```
SELECT first_name || ' ' || last_name "Name", commission_pct
FROM   employees
WHERE  commission_pct > .35;
```

Name	COMMISSION_PCT
John Russell	.4

<= (Less Than or Equal To)

The <= operator evaluates to TRUE if the left side (expression or value) of the operator is less than or equal to the right side of the operator. The following shows an example similar to the one for less than, but here we include all employees who have a commission percent less than 15 or equal to 15:

```
SELECT first_name || ' ' || last_name "Name", commission_pct
FROM   employees
WHERE  commission_pct <= .15;
```

Name	COMMISSION_PCT
Oliver Tuvault	.15
Danielle Greene	.15
Mattea Marvins	.1
David Lee	.1
Sundar Ande	.1
Amit Banda	.1
William Smith	.15
Elizabeth Bates	.15
Sundita Kumar	.1
Kimberely Grant	.15
Charles Johnson	.1

11 rows selected.

>= (Greater Than or Equal To)

The `>=` operator evaluates to TRUE if the left side (expression or value) of the operator is greater than or equal to the right side of the operator. The following shows an example of employees with commission percent at or over 35:

```
SELECT first_name || ' ' || last_name "Name", commission_pct
FROM   employees
WHERE  commission_pct >= .35;
```

Name	COMMISSION_PCT
John Russell	.4
Janette King	.35
Patrick Sully	.35
Allan McEwen	.35

ANY or SOME

You can use the ANY or SOME (both are the same and can be used interchangeably) operator to compare a value to each value in a list or subquery. The ANY and SOME operators always must be preceded by one of the following comparison operators: `=`, `!=`, `<`, `>`, `<=`, or `>=`. Consider the following SQL with ANY operator:

```
SELECT first_name || ' ' || last_name "Name", department_id
FROM   employees
WHERE  department_id <= ANY (10, 15, 20, 25);
```

Name	DEPARTMENT_ID
Jennifer Whalen	10
Michael Hartstein	20
Pat Fay	20

Oracle in fact expands the ANY condition to department_id <= 10 OR department_id <= 15 OR department_id <= 20 OR department_id <= 25. The behavior of ANY with each comparison operator is

$X = \text{ANY } (\text{list})$: Evaluates to TRUE if value of X matches at least one value in the list.

$X \neq \text{ANY } (\text{list})$: Evaluates to TRUE if value of X does not match one or more value in the list.

$X > \text{ANY } (\text{list})$: Evaluates to TRUE if value of X is higher than the smallest value in the list.

$X < \text{ANY } (\text{list})$: Evaluates to TRUE if value of X is lower than the biggest value in the list.

$X \geq \text{ANY } (\text{list})$: Evaluates to TRUE if value of X is higher than or equal to the smallest value in the list.

$X \leq \text{ANY } (\text{list})$: Evaluates to TRUE if value of X is lower than or equal to the biggest value in the list.

ALL

You can use the ALL operator to compare a value to every value in a list or subquery. The ALL operator must always be preceded by one of the following comparison operators: =, !=, <, >, <=, or >=. The following shows an example of using the ALL operator:

```
SELECT first_name || ' ' || last_name "Name", department_id
FROM   employees
WHERE  department_id >= ALL (80, 90, 100);
```

Name	DEPARTMENT_ID
Nancy Greenberg	100
Daniel Faviet	100
John Chen	100
Ismael Sciarra	100
Jose Manuel Urman	100
Luis Popp	100
Shelley Higgins	110
William Gietz	110

8 rows selected.

The SQL code is evaluated by Oracle as `department_id >= 80 AND department_id >= 90 AND department_id >= 100`. The behavior of ALL with each comparison operator is

`X = ALL (list)`: Evaluates to TRUE if value of X matches all of the values in the list.

`X != ALL (list)`: Evaluates to TRUE if value of X does not match any value in the list.

`X > ALL (list)`: Evaluates to TRUE if value of X is higher than the biggest value in the list.

`X < ALL (list)`: Evaluates to TRUE if value of X is lower than the smallest value in the list.

`X >= ALL (list)`: Evaluates to TRUE if value of X is higher than or equal to the highest value in the list.

`X <= ALL (list)`: Evaluates to TRUE if value of X is lower than or equal to the smallest value in the list.

The ANY, SOME, and ALL operators make more sense when used with subqueries rather than a list of known values. Subqueries are discussed in Chapter 5, “Using Joins and Subqueries.”



For all the comparison operators discussed, if one side of the operator is NULL, the result is NULL. At least one question with the ANY, ALL operator will be on the OCA test, so make sure you understand these operators thoroughly.

Logical Operators

Logical operators are used to combine the results of two comparison conditions (compound conditions) to produce a single result or to reverse the result of a single comparison. NOT, AND, and OR are the logical operators. When a logical operator is applied to NULL, the result is UNKNOWN. UNKNOWN acts similarly to FALSE; the only difference is that NOT FALSE is TRUE, whereas NOT UNKNOWN is also UNKNOWN.

NOT

You can use the NOT operator to reverse the result. It evaluates to TRUE if the operand is FALSE, and it evaluates to FALSE if the operand is TRUE. NOT returns NULL if the operand is NULL. The following shows an example of employees that do not belong to department number greater or equal to 30—that is, it basically evaluates to department number less than 30:

```
SELECT first_name, department_id  
FROM   employees  
WHERE  not (department_id >= 30);
```

FIRST_NAME	DEPARTMENT_ID
Jennifer	10

Michael	20
Pat	20

AND

The AND operator evaluates to TRUE if both operands are TRUE. It evaluates to FALSE if either operand is FALSE. Otherwise, it returns NULL. The following shows an example of employees with the last name Smith and a salary over \$7,500. Both conditions must evaluate to TRUE:

```
SELECT first_name, salary
FROM   employees
WHERE  last_name = 'Smith'
AND    salary     > 7500;
```

FIRST_NAME	SALARY
Lindsey	8000

OR

The OR operator evaluates to TRUE if either operand is TRUE. It evaluates to FALSE if both operands are FALSE. Otherwise, it returns NULL. The following shows an example of employees whose last name is Smith or whose first name is Kelly. Either one condition must evaluate to TRUE:

```
SELECT first_name, last_name
FROM   employees
WHERE  first_name = 'Kelly'
OR    last_name   = 'Smith';
```

FIRST_NAME	LAST_NAME
Kelly	Chung
Lindsey	Smith
William	Smith

Logical Operator Truth Tables

The following tables are the truth tables for the three logical operators.

Table 2.7 is a truth table for the AND operator.

TABLE 2.7 AND Truth Table

AND	TRUE	FALSE	UNKNOWN
TRUE	TRUE	FALSE	UNKNOWN
FALSE	FALSE	FALSE	FALSE
UNKNOWN	UNKNOWN	FALSE	UNKNOWN

Table 2.8 is the truth table for the OR operator.

TABLE 2.8 OR Truth Table

OR	TRUE	FALSE	UNKNOWN
TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	UNKNOWN
UNKNOWN	TRUE	UNKNOWN	UNKNOWN

Table 2.9 is the truth table for the NOT operator.

TABLE 2.9 NOT Truth Table

NOT	
TRUE	FALSE
FALSE	TRUE
UNKNOWN	UNKNOWN

Other Operators

In the following sections, we will discuss the operators that can be used in the WHERE clause of the SQL statement that were not discussed earlier.

IN and NOT IN

You can use the IN and NOT IN operators to test a membership condition. IN is equivalent to the =ANY operator, which evaluates to TRUE if the value exists in the list or the result set

from a subquery. The NOT IN operator is equivalent to the !=ALL operator, which evaluates to TRUE if the value does not exist in the list or the result set from a subquery. The following examples demonstrate how to use these two operators:

```
SELECT first_name, last_name, department_id  
FROM employees  
WHERE department_id IN (10, 20, 90);
```

FIRST_NAME	LAST_NAME	DEPARTMENT_ID
Jennifer	Whalen	10
Michael	Hartstein	20
Pat	Fay	20
Steven	King	90
Neena	Kochhar	90
Lex	De Haan	90

6 rows selected.

FIRST_NAME	LAST_NAME	DEPARTMENT_ID
Hermann	Baer	70
Pat	Fay	20
Michael	Hartstein	20

SQL>



When using the NOT IN operator, if any value in the list or the result returned from the subquery is NULL, the NOT IN condition is evaluated to FALSE. For example, last_name not in ('Smith', 'Thomas', NULL) evaluates to last_name != 'Smith' AND last_name != 'Thomas' AND last_name != NULL. Any comparison on a NULL value results in NULL. So the previous condition does not return any row even though there may be some rows with LAST_NAME as Smith or Thomas.

BETWEEN

You can use the BETWEEN operator to test a range. BETWEEN *A* AND *B* evaluates to TRUE if the value is greater than or equal to *A* and less than or equal to *B*. If NOT is used, the result is the reverse. The following example lists all the employees whose salary is between \$5,000 and \$6,000:

```
SELECT first_name, last_name, salary  
FROM employees
```