

Contents at a Glance

<i>Introduction</i>	1
<i>Part I: Basic PL/SQL Concepts</i>	7
Chapter 1: PL/SQL and Your Database	9
Chapter 2: The PL/SQL Environment	23
<i>Part II: Getting Started with PL/SQL</i>	39
Chapter 3: Laying the Groundwork: PL/SQL Fundamentals	41
Chapter 4: Controlling Program Flow	85
Chapter 5: Handling Exceptions	105
Chapter 6: PL/SQL and SQL Working Together	127
<i>Part III: Standards and Structures</i>	163
Chapter 7: Putting Your Code in the Right Place	165
Chapter 8: Creating Naming Standards	187
Chapter 9: Creating Coding Standards	201
<i>Part IV: PL/SQL Data Manipulations</i>	219
Chapter 10: Basic Datatypes	221
Chapter 11: Advanced Datatypes	253
<i>Part V: Taking PL/SQL to the Next Level</i>	289
Chapter 12: Transaction Control	291
Chapter 13: Dynamic SQL and PL/SQL	313
Chapter 14: PL/SQL Best Practices	335
<i>Part VI: The Part of Tens</i>	355
Chapter 15: Ten PL/SQL Tips	357
Chapter 16: Ten Common Mistakes to Avoid in PL/SQL	377
<i>Index</i>	397

Table of Contents

<i>Introduction</i>	1
About This Book.....	1
Foolish Assumptions	2
How This Book Is Organized.....	2
Part I: Basic PL/SQL Concepts	2
Part II: Getting Started with PL/SQL.....	3
Part III: Standards and Structures	3
Part IV: PL/SQL Data Manipulations	3
Part V: Taking PL/SQL to the Next Level	3
Part VI: The Part of Tens	4
Icons Used in This Book.....	4
Where to Go from Here.....	4
<i>Part 1: Basic PL/SQL Concepts</i>	7
Chapter 1: PL/SQL and Your Database	9
Knowing Just Enough about Relational Databases.....	9
What makes a database “relational”?	10
Understanding basic database terminology.....	12
Introducing database normalization.....	13
What is a DBMS?	16
The Scoop on SQL and PL/SQL.....	16
The purpose of SQL and PL/SQL.....	17
The difference between SQL and PL/SQL.....	18
What’s new in Oracle SQL and PL/SQL?	18
What Is PL/SQL Good For?	19
Using database triggers	19
Scripting with speed	20
Keeping code server-side	20
Programming for Oracle Developer	21
Chapter 2: The PL/SQL Environment	23
Setting Up an Oracle Environment.....	23
Hardware and Software Requirements	25
Accessing the Oracle Technology Network.....	26
Installing the Database	27
Working with Code.....	28
SQL*Plus.....	28
Oracle SQL Developer.....	30
Third-party tools	30



- Establishing a Database Connection31
 - Checking the operating services32
 - Connecting with your username32
 - Unlocking (or locking) accounts33
 - Resetting passwords33
 - Setting up the server to communicate34
- The Basics of PL/SQL Programs34
 - Writing a simple program34
 - Creating stored procedures36
 - Passing parameters to procedures36
- Examining the Sample Data37
 - The Scott/Tiger schema37
 - The Human Resources (HR) and Order Entry (OE) schemas38

Part II: Getting Started with PL/SQL.....39

Chapter 3: Laying the Groundwork: PL/SQL Fundamentals 41

- PL/SQL As a Programming Language41
- Anonymous PL/SQL Blocks.....42
- Introducing the Lexical Set of Elements.....43
 - Identifiers44
 - Delimiters44
 - Literals45
 - Comments45
- Working with Constants and Variables.....46
 - Declaring variables.....46
 - Assigning values to variables48
 - Literals as variable values49
 - Understanding the scope of variables51
- Building Expressions with Operators53
- Running Anonymous Blocks of Code.....56
 - Identifying common mistakes56
 - Spotting compilation errors57
 - Recognizing semicolon-related errors57
- Creating Reusable PL/SQL Code59
 - Wrapping a task into a procedure59
 - Returning values with functions.....61
 - Parameters of subprograms63
- Storing PL/SQL in the Database67
 - Database procedures and functions68
 - Packages69
 - Triggers.....71
 - Interpreting and fixing compilation errors.....73

Checking Out PL/SQL Extras	76
Overloading calls	76
Resolving calls to subprograms	78
Recursion	80
Compiler hints and directives	82
Built-in packages	83
Chapter 4: Controlling Program Flow	85
Creating Condition Statements	85
IF...THEN statements	86
IF...ELSE statements	87
CASE statements.....	89
Comparing with NULL.....	91
Handling conditions	95
Looping the Loop	97
Simple loops.....	97
Nested loops	99
WHILE loop.....	100
FOR loop	102
Chapter 5: Handling Exceptions	105
Understanding Exception Basics	106
Adding Exception Handlers to Your Code	107
Understanding Different Exception Types.....	110
Predefined Exceptions in PL/SQL Code.....	111
Adding User-Defined Exceptions.....	114
Assigning a code to a user-defined exception	115
Including error messages in user-defined exceptions	116
Propagation of Exceptions	118
Seeing propagation of exceptions in action.....	118
Handling exceptions without halting the program	122
Avoiding exceptions raised in declaration part and exception handler	124
Writing Exceptional Exceptions	126
Chapter 6: PL/SQL and SQL Working Together	127
Cursors: What They Are and How to Use Them	128
Returning more than one piece of information	129
Looping through multiple records	132
Placing cursors in nested loops	133
Passing parameters to cursors.....	134
Knowing Where to Declare Cursors.....	137
Defining cursors in the header of the program unit	138
Defining cursors in the local PL/SQL block.....	138
Defining cursors in the package body	139
Defining cursors in the package spec.....	140

Being Explicitly Smart with Implicit Cursors.....	142
Retrieving a single row: The basic syntax.....	142
Handling exceptions in implicit cursors	143
Returning an implicit cursor into a record	143
Accessing Status Info by Using Cursor Variables.....	144
Checking the status of explicit cursors	145
Checking the status of implicit cursors.....	146
Updating Records Fetched from Cursors.....	148
Using a simple UPDATE statement	148
Updating with logical operators.....	148
Taking a Shortcut with CURSOR FOR Loops.....	150
Comparing CURSOR FOR loops to cursors with the LOOP command.....	150
When do CURSOR FOR loops simplify exception handling?.....	152
When CURSOR FOR loops make your life harder	153
Knowing what record is processing.....	155
Referencing Functions in SQL.....	155
Important facts to remember.....	157
Getting good performance with functions	160

Part III: Standards and Structures 163

Chapter 7: Putting Your Code in the Right Place 165

Putting Code in the Database	165
Managing code.....	166
Packaging code in the database	166
Placing triggers on tables	174
Building INSTEAD OF trigger views.....	177
Advantages of putting code in the database	182
Disadvantages of putting code in the database	182
Putting Code in the Application Server (Middle-Tier Approach)	183
Advantages of the middle-tier approach.....	184
Disadvantages of the middle-tier approach.....	184
Placing code in the view layer	185
Where Should You Place the Business Logic?	185

Chapter 8: Creating Naming Standards 187

What's in a Naming Standard?	187
Oracle database influences	188
Java influences.....	188
Modern application development tools and their influences.....	189
Setting Naming Standards for Common Code Elements	189
Basic objects	190
Variables	191

Program units: Procedures, packages, functions, triggers193
 Parameters in code objects194
 Exceptions196
 User-defined datatypes.....196
 Collections.....197
 Filenames.....198
 Making Sure Your Organization Follows Standards199

Chapter 9: Creating Coding Standards201

Why Standards Are Important.....201
 Universal Truths.....202
 Don't hard-code any constant value202
 Don't make your program units too big or too small205
 Put each data element on its own line.....206
 Too many comments are much better than
 too few comments207
 Avoid global variables210
 Indent carefully.....210
 Be careful with capitalization211
 Use generic variable datatype declarations211
 Limit line length.....213
 Use explicit data conversion for dates213
 Use synonyms213
 Developing SQL Code Consistently214
 Using a new line.....214
 Using explicit column lists214
 Prefixing (and suffixing) column names
 from multiple tables215
 Giving columns aliases216
 Using parentheses in complex mathematical
 and logical expressions216
 Using white space intelligently217
 Writing save exception handlers217
 Packaging stored program units.....217

Part IV: PL/SQL Data Manipulations219

Chapter 10: Basic Datatypes221

Introducing the Main Datatype Groups.....221
 Working with Numeric Datatypes222
 Using the NUMBER datatype222
 Boosting performance with BINARY_INTEGER225
 Using BINARY_FLOAT and BINARY_DOUBLE
 for complex calculations226

Handling numeric datatypes in built-in functions	228
Keeping Track of Date and Time	229
Selecting the info you want from DATE	229
Using TIMESTAMP	234
Using TIMESTAMP WITH TIME ZONE	235
Storing elapsed time with INTERVAL	236
Working with dates and built-in functions	237
Storing Logical Values with BOOLEAN	241
Processing Characters and Strings	242
Limiting variable length with CHAR versus VARCHAR2	242
Useful character built-in functions	244
Extending your options with regular expressions	250
Chapter 11: Advanced Datatypes	253
Handling Large Objects in the Database	253
Using internal large objects (CLOB, BLOB)	254
Creating pointers with external large objects	254
Working with Large Objects	255
Populating BFILE	255
Loading data to the CLOB by using BFILE	256
Loading a page to a BLOB	257
Performing basic string operations on CLOBs	258
Keeping Code Consistent with User-Defined Subtypes	259
Defining Your Own Datatypes	260
Records	261
Object types	267
Grouping Sets of Data into Collections	271
Using VARRAYs	272
Nesting variable data sets in tables	275
Associative arrays (index-by tables)	280
Speeding Up Data Collection with Bulk Operations	283
Using the BULK COLLECT command	284
Adding a limit to BULK COLLECT	286
 Part V: Taking PL/SQL to the Next Level	 289
Chapter 12: Transaction Control	291
Using Transactions to Maintain Data Consistency	292
Committing or Rolling Back Changes to the Database	293
Firing explicit and implicit commits	294
Rolling back changes	294
Knowing when to use implicit rollbacks	297
Resolving deadlocks	298

Autonomous Transactions.....298
 Setting up the syntax for an autonomous transaction299
 Handling auditing and security with autonomous transactions.....300
 Autonomous transaction features.....302
 Applying autonomous transactions to other real-world situations308

Chapter 13: Dynamic SQL and PL/SQL 313

Taking Advantage of Dynamic SQL313
 A Simple EXECUTE IMMEDIATE314
 Building SQL on the Fly316
 Improve performance with bind variables.....317
 Return information using cursor variables324
 Building DDL on the Fly.....325
 Using Quoted Strings with Dynamic SQL.....327
 Working with Advanced Datatypes.....328
 Using BULK COLLECT with dynamic SQL329
 Dynamic OPEN...FOR330

Chapter 14: PL/SQL Best Practices 335

Why Are Best Practices Important?.....335
 Laying the Groundwork for Good Coding336
 Understanding the big picture.....336
 Communicating effectively.....337
 Creating a code specification.....337
 Writing Code with Best Practices in Mind338
 Stub out your code.....338
 Check the architecture as you go.....339
 Prove code works with test cases340
 Use code libraries.....340
 Keep the code maintainable341
 Don't forget about performance.....341
 Compile as you go341
 Debug the timesaving way342
 Testing Your Code343
 What are you testing?345
 Creating a testing architecture345
 Performance and load testing.....345
 Tuning performance346
 "Good enough is best"348
 Coding the Agile Way349
 Working together in Agile teams349
 Programming in pairs350
 Delivering code quickly350
 Test first.....351

Keeping Up-to-Date with Oracle	352
Conventional wisdom isn't always right	352
Buy books	353
Go to conferences.....	353
Join your local Oracle user group.....	354
Use online resources.....	354
Part VI: The Part of Tens	355
Chapter 15: Ten PL/SQL Tips	357
Use SELECT INTO Correctly	357
Don't Assume Column Order in DML Statements	360
Use the NOCOPY Command	362
Be Careful of Function Calls That Impact Performance	364
Beware of Implicit Commits.....	366
Use Only One RETURN Statement per Function	369
Save Debugging Time with WHEN OTHERS	370
Know When to Pass Parameters or Use Global Variables.....	372
Use Implicit Cursors When Appropriate.....	373
Remember That Dates Are Numbers.....	375
Chapter 16: Ten Common Mistakes to Avoid in PL/SQL	377
Catching an Exception with WHEN OTHERS THEN NULL	377
Forgetting to Handle NULL Values	379
Creating Unintended Boolean Expressions	380
Forgetting to Close an Explicit Cursor	382
Starting Endless Loops	384
Reinventing the Wheel.....	386
Converting Datatypes Implicitly	388
Cutting and Pasting Code.....	391
Ignoring Code Readability.....	393
Assuming Code Doesn't Need Comments.....	395
Index.....	397