

Oracle Database 11g Release 2

5th Edition

Oracle PL/SQL

Programming



*Steven Feuerstein
with Bill Pribyl*

Table of Contents

Preface	xxv
---------------	-----

Part I. Programming in PL/SQL

1. Introduction to PL/SQL	3
What Is PL/SQL?	3
The Origins of PL/SQL	4
The Early Years of PL/SQL	4
Improved Application Portability	5
Improved Execution Authority and Transaction Integrity	5
Humble Beginnings, Steady Improvement	6
So This Is PL/SQL	7
Integration with SQL	7
Control and Conditional Logic	8
When Things Go Wrong	9
About PL/SQL Versions	10
Oracle Database 11g New Features	12
Resources for PL/SQL Developers	17
The O'Reilly PL/SQL Series	17
PL/SQL on the Internet	18
Some Words of Advice	19
Don't Be in Such a Hurry!	20
Don't Be Afraid to Ask for Help	21
Take a Creative, Even Radical Approach	22
2. Creating and Running PL/SQL Code	23
Navigating the Database	23
Creating and Editing Source Code	24
SQL*Plus	25
Starting Up SQL*Plus	26
Running a SQL Statement	28

Running a PL/SQL Program	28
Running a Script	30
What Is the “Current Directory”?	31
Other SQL*Plus Tasks	32
Error Handling in SQL*Plus	36
Why You Will Love and Hate SQL*Plus	37
Performing Essential PL/SQL Tasks	38
Creating a Stored Program	38
Executing a Stored Program	41
Showing Stored Programs	42
Managing Grants and Synonyms for Stored Programs	43
Dropping a Stored Program	44
Hiding the Source Code of a Stored Program	44
Editing Environments for PL/SQL	45
Calling PL/SQL from Other Languages	46
C: Using Oracle’s Precompiler (Pro*C)	47
Java: Using JDBC	48
Perl: Using Perl DBI and DBD::Oracle	49
PHP: Using Oracle Extensions	50
PL/SQL Server Pages	51
And Where Else?	52
3. Language Fundamentals	53
PL/SQL Block Structure	53
Anonymous Blocks	54
Named Blocks	56
Nested Blocks	57
Scope	58
Qualify all References to Variables and Columns in SQL Statements	59
Visibility	61
The PL/SQL Character Set	64
Identifiers	66
Reserved Words	68
Whitespace and Keywords	69
Literals	70
NULLs	71
Embedding Single Quotes Inside a Literal String	72
Numeric Literals	73
Boolean Literals	73
The Semicolon Delimiter	74
Comments	75
Single-Line Comment Syntax	75
Multiline Comment Syntax	75

The PRAGMA Keyword	76
Labels	77

Part II. PL/SQL Program Structure

4. Conditional and Sequential Control	81
IF Statements	81
The IF-THEN Combination	82
The IF-THEN-ELSE Combination	84
The IF-THEN-ELSIF Combination	85
Avoiding IF Syntax Gotchas	86
Nested IF Statements	88
Short-Circuit Evaluation	89
CASE Statements and Expressions	90
Simple CASE Statements	91
Searched CASE Statements	93
Nested CASE Statements	95
CASE Expressions	95
The GOTO Statement	97
The NULL Statement	98
Improving Program Readability	99
Using NULL After a Label	99
5. Iterative Processing with Loops	101
Loop Basics	101
Examples of Different Loops	102
Structure of PL/SQL Loops	103
The Simple Loop	104
Terminating a Simple Loop: EXIT and EXIT WHEN	105
Emulating a REPEAT UNTIL Loop	106
The Intentionally Infinite Loop	106
The WHILE Loop	108
The Numeric FOR Loop	109
Rules for Numeric FOR Loops	110
Examples of Numeric FOR Loops	111
Handling Nontrivial Increments	112
The Cursor FOR Loop	112
Example of Cursor FOR Loops	114
Loop Labels	115
The CONTINUE Statement	116
Tips for Iterative Processing	119
Use Understandable Names for Loop Indexes	119

The Proper Way to Say Goodbye	120
Obtaining Information About FOR Loop Execution	121
SQL Statement as Loop	122
6. Exception Handlers	125
Exception-Handling Concepts and Terminology	125
Defining Exceptions	128
Declaring Named Exceptions	128
Associating Exception Names with Error Codes	129
About Named System Exceptions	132
Scope of an Exception	134
Raising Exceptions	135
The RAISE Statement	136
Using RAISE_APPLICATION_ERROR	137
Handling Exceptions	138
Built-in Error Functions	139
Combining Multiple Exceptions in a Single Handler	144
Unhandled Exceptions	145
Propagation of Unhandled Exceptions	145
Continuing Past Exceptions	148
Writing WHEN OTHERS Handling Code	150
Building an Effective Error Management Architecture	152
Decide on Your Error Management Strategy	153
Standardize Handling of Different Types of Exceptions	154
Organize Use of Application-Specific Error Codes	157
Use Standardized Error Management Programs	157
Work with Your Own Exception “Objects”	159
Create Standard Templates for Common Error Handling	162
Making the Most of PL/SQL Error Management	164

Part III. PL/SQL Program Data

7. Working with Program Data	167
Naming Your Program Data	167
Overview of PL/SQL Datatypes	169
Character Data	170
Numbers	170
Dates, Timestamps, and Intervals	172
Booleans	172
Binary Data	172
ROWIDs	173
REF CURSORS	173

Internet Datatypes	174
"Any" Datatypes	174
User-Defined Datatypes	174
Declaring Program Data	175
Declaring a Variable	175
Declaring Constants	176
The NOT NULL Clause	177
Anchored Declarations	177
Anchoring to Cursors and Tables	179
Benefits of Anchored Declarations	180
Anchoring to NOT NULL Datatypes	181
Programmer-Defined Subtypes	182
Conversion Between Datatypes	183
Implicit Data Conversion	183
Explicit Datatype Conversion	185
8. Strings	191
String Datatypes	191
The VARCHAR2 Datatype	192
The CHAR Datatype	193
String Subtypes	194
Working with Strings	195
Specifying String Constants	195
Using Nonprintable Characters	197
Concatenating Strings	198
Dealing with Case	199
Traditional Searching, Extracting, and Replacing	202
Padding	204
Trimming	206
Regular Expression Searching, Extracting, and Replacing	207
Working with Empty Strings	218
Mixing CHAR and VARCHAR2 Values	219
String Function Quick Reference	222
9. Numbers	231
Numeric Datatypes	231
The NUMBER Type	232
The PLS_INTEGER Type	237
The BINARY_INTEGER Type	238
The SIMPLE_INTEGER Type	239
The BINARY_FLOAT and BINARY_DOUBLE Types	241
The SIMPLE_FLOAT and SIMPLE_DOUBLE Types	246
Numeric Subtypes	246

Number Conversions	247
The TO_NUMBER Function	247
The TO_CHAR Function	251
The CAST Function	256
Implicit Conversions	257
Numeric Operators	259
Numeric Functions	260
Rounding and Truncation Functions	260
Trigonometric Functions	261
Numeric Function Quick Reference	261
10. Dates and Timestamps	267
Datetime Datatypes	267
Declaring Datetime Variables	270
Choosing a Datetime Datatype	271
Getting the Current Date and Time	272
Interval Datatypes	274
Declaring INTERVAL Variables	275
When to Use INTERVALs	276
Datetime Conversions	278
From Strings to Datetimes	279
From Datetimes to Strings	281
Working with Time Zones	284
Requiring a Format Mask to Match Exactly	287
Easing Up on Exact Matches	288
Interpreting Two-Digit Years in a Sliding Window	288
Converting Time Zones to Character Strings	290
Padding Output with Fill Mode	291
Date and Timestamp Literals	291
Interval Conversions	292
Converting from Numbers to Intervals	293
Converting Strings to Intervals	294
Formatting Intervals for Display	295
Interval Literals	295
CAST and EXTRACT	297
The CAST Function	297
The EXTRACT Function	299
Datetime Arithmetic	300
Date Arithmetic with Intervals and Datetimes	300
Date Arithmetic with DATE Datatypes	301
Computing the Interval Between Two Datetimes	302
Mixing DATES and TIMESTAMPS	304
Adding and Subtracting Intervals	305

Multiplying and Dividing Intervals	306
Using Unconstrained INTERVAL Types	306
Date/Time Function Quick Reference	308
11. Records	311
Records in PL/SQL	311
Benefits of Using Records	312
Declaring Records	314
Programmer-Defined Records	315
Working with Records	318
Comparing Records	325
Trigger Pseudo-Records	326
12. Collections	327
Collections Overview	328
Collections Concepts and Terminology	328
Types of Collections	330
Collection Examples	331
Where You Can Use Collections	335
Choosing a Collection Type	340
Collection Methods (Built-ins)	341
The COUNT Method	343
The DELETE Method	343
The EXISTS Method	345
The EXTEND Method	345
The FIRST and LAST Methods	346
The LIMIT Method	347
The PRIOR and NEXT Methods	348
The TRIM Method	349
Working with Collections	350
Declaring Collection Types	350
Declaring and Initializing Collection Variables	355
Populating Collections with Data	359
Accessing Data Inside a Collection	364
Using String-Indexed Collections	365
Collections of Complex Datatypes	370
Multilevel Collections	374
Working with Collections in SQL	382
Nested Table Multiset Operations	387
Testing Equality and Membership of Nested Tables	389
Checking for Membership of an Element in a Nested Table	390
Performing High-Level Set Operations	390
Handling Duplicates in a Nested Table	392

Maintaining Schema-Level Collections	393
Necessary Privileges	393
Collections and the Data Dictionary	394
13. Miscellaneous Datatypes	395
The BOOLEAN Datatype	395
The RAW Datatype	396
The UROWID and ROWID Datatypes	397
Getting ROWIDs	398
Using ROWIDs	398
The LOB Datatypes	400
Working with LOBs	401
Understanding LOB Locators	403
Empty Versus NULL LOBs	405
Writing into a LOB	407
Reading from a LOB	409
BFILES Are Different	410
SecureFiles Versus BasicFiles	415
Temporary LOBs	417
Native LOB Operations	421
LOB Conversion Functions	425
Predefined Object Types	426
The XMLType Type	426
The URI Types	430
The Any Types	431

Part IV. SQL in PL/SQL

14. DML and Transaction Management	439
DML in PL/SQL	440
A Quick Introduction to DML	440
Cursor Attributes for DML Operations	444
RETURNING Information from DML Statements	445
DML and Exception Handling	446
DML and Records	447
Transaction Management	450
The COMMIT Statement	451
The ROLLBACK Statement	451
The SAVEPOINT Statement	452
The SET TRANSACTION Statement	453
The LOCK TABLE Statement	454
Autonomous Transactions	454

Defining Autonomous Transactions	455
Rules and Restrictions on Autonomous Transactions	456
Transaction Visibility	457
When to Use Autonomous Transactions	458
Building an Autonomous Logging Mechanism	459
15. Data Retrieval	463
Cursor Basics	464
Some Data Retrieval Terms	465
Typical Query Operations	466
Introduction to Cursor Attributes	467
Referencing PL/SQL Variables in a Cursor	470
Choosing Between Explicit and Implicit Cursors	471
Working with Implicit Cursors	471
Implicit Cursor Examples	472
Error Handling with Implicit Cursors	473
Implicit SQL Cursor Attributes	476
Working with Explicit Cursors	477
Declaring Explicit Cursors	479
Opening Explicit Cursors	482
Fetching from Explicit Cursors	483
Column Aliases in Explicit Cursors	484
Closing Explicit Cursors	485
Explicit Cursor Attributes	487
Cursor Parameters	489
SELECT...FOR UPDATE	492
Releasing Locks with COMMIT	494
The WHERE CURRENT OF Clause	495
Cursor Variables and REF CURSORS	496
Why Cursor Variables?	497
Similarities to Static Cursors	498
Declaring REF CURSOR Types	498
Declaring Cursor Variables	499
Opening Cursor Variables	500
Fetching from Cursor Variables	501
Rules for Cursor Variables	504
Passing Cursor Variables as Arguments	507
Cursor Variable Restrictions	509
Cursor Expressions	509
Using Cursor Expressions	510
Restrictions on Cursor Expressions	512

16. Dynamic SQL and Dynamic PL/SQL	513
NDS Statements	
The EXECUTE IMMEDIATE Statement	514
The OPEN FOR Statement	514
About the Four Dynamic SQL Methods	517
Binding Variables	523
Argument Modes	525
Duplicate Placeholders	526
Passing NULL Values	527
Working with Objects and Collections	528
Dynamic PL/SQL	529
Build Dynamic PL/SQL Blocks	531
Replace Repetitive Code with Dynamic Blocks	532
Recommendations for NDS	534
Use Invoker Rights for Shared Programs	535
Anticipate and Handle Dynamic Errors	535
Use Binding Rather Than Concatenation	536
Minimize the Dangers of Code Injection	538
When to Use DBMS_SQL	540
Parse Very Long Strings	543
Obtain Information About Query Columns	543
Meet Method 4 Dynamic SQL Requirements	544
Minimize Parsing of Dynamic Cursors	546
Oracle Database 11g New Features	552
DBMS_SQL.TO_REFCURSOR Function	554
DBMS_SQL.TO_CURSOR Function	554
Enhanced Security for DBMS_SQL	556
	558

Part V. PL/SQL Application Construction

17. Procedures, Functions, and Parameters	565
Modular Code	566
Procedures	567
Calling a Procedure	568
The Procedure Header	569
The Procedure Body	570
The END Label	570
The RETURN Statement	571
Functions	571
Structure of a Function	571
The RETURN Datatype	571
The END Label	573
	575

Calling a Function	575
Functions Without Parameters	576
The Function Header	577
The Function Body	577
The RETURN Statement	578
Parameters	579
Defining Parameters	580
Actual and Formal Parameters	581
Parameter Modes	582
Explicit Association of Actual and Formal Parameters in PL/SQL	585
The NOCOPY Parameter Mode Qualifier	589
Default Values	589
Local or Nested Modules	590
Benefits of Local Modularization	591
Scope of Local Modules	594
Sprucing Up Your Code with Local Modules	594
Module Overloading	595
Benefits of Overloading	596
Restrictions on Overloading	599
Overloading with Numeric Types	600
Forward Declarations	601
Advanced Topics	602
Calling Your Function From Inside SQL	602
Table Functions	605
Deterministic Functions	615
Go Forth and Modularize!	616
18. Packages	617
Why Packages?	617
Demonstrating the Power of the Package	618
Some Package-Related Concepts	621
Diagramming Privacy	623
Rules for Building Packages	624
The Package Specification	624
The Package Body	626
Initializing Packages	627
Rules for Calling Packaged Elements	632
Working with Package Data	633
Global Within a Single Oracle Session	633
Global Public Data	634
Packaged Cursors	635
Serializable Packages	639
When to Use Packages	642

Encapsulate Data Access	642
Avoid Hardcoding Literals	645
Improve Usability of Built-in Features	647
Group Together Logically Related Functionality	648
Cache Static Session Data	649
Packages and Object Types	650
19. Triggers	651
DML Triggers	652
DML Trigger Concepts	653
Creating a DML Trigger	655
DML Trigger Example: No Cheating Allowed!	660
Multiple Triggers of the Same Type	665
Who Follows Whom	666
Mutating Table Errors	668
Compound Triggers: Putting It All In One Place	669
DDL Triggers	673
Creating a DDL Trigger	673
Available Events	676
Available Attributes	676
Working with Events and Attributes	678
Dropping the Undroppable	681
The INSTEAD OF CREATE Trigger	682
Database Event Triggers	683
Creating a Database Event Trigger	683
The STARTUP Trigger	685
The SHUTDOWN Trigger	685
The LOGON Trigger	685
The LOGOFF Trigger	686
The SERVERERROR Trigger	686
INSTEAD OF Triggers	690
Creating an INSTEAD OF Trigger	690
The INSTEAD OF INSERT Trigger	692
The INSTEAD OF UPDATE Trigger	694
The INSTEAD OF DELETE Trigger	695
Populating the Tables	695
INSTEAD OF Triggers on Nested Tables	696
AFTER SUSPEND Triggers	697
Setting Up for the AFTER SUSPEND Trigger	698
Looking at the Actual Trigger	700
The ORA_SPACE_ERROR_INFO Function	701
The DBMS_RESUMABLE Package	702
Trapped Multiple Times	703

To Fix or Not to Fix?	704
Maintaining Triggers	705
Disabling, Enabling, and Dropping Triggers	705
Creating Disabled Triggers	706
Viewing Triggers	706
Checking the Validity of Triggers	707
20.8 Managing PL/SQL Code	709
Managing Code in the Database	710
Overview of Data Dictionary Views	711
Display Information About Stored Objects	712
Display and Search Source Code	713
Use Program Size to Determine Pinning Requirements	715
Obtain Properties of Stored Code	715
Analyze and Modify Trigger State Through Views	716
Analyze Argument Information	717
Analyze Identifier Usage (Oracle Database 11g's PL/Scope)	719
Managing Dependencies and Recompiling Code	721
Analyzing Dependencies with Data Dictionary Views	722
Fine-Grained Dependency (Oracle Database 11g)	726
Remote Dependencies	727
Limitations of Oracle's Remote Invocation Model	730
Recompiling Invalid Program Units	731
Compile-Time Warnings	735
A Quick Example	735
Enabling Compile-Time Warnings	736
Some Handy Warnings	738
Testing PL/SQL Programs	746
Typical, Tawdry Testing Techniques	747
General Advice for Testing PL/SQL Code	751
Automated Testing Options for PL/SQL	752
Testing with utPLSQL	753
Testing with Quest Code Tester for Oracle	755
Tracing PL/SQL Execution	756
DBMS_APPLICATION_INFO	759
Quest Error Manager Tracing	761
The DBMS_TRACE Facility	763
Debugging PL/SQL Programs	766
The Wrong Way to Debug	767
Debugging Tips and Strategies	769
Protecting Stored Code	774
Restrictions on and Limitations of Wrapping	774
Using the Wrap Executable	775

704	Dynamic Wrapping with DBMS_DDL	775
705	Guidelines for Working with Wrapped Code	776
706	Introduction to Edition-Based Redefinition (Oracle Database 11g Release 2)	777
707		
21.	Optimizing PL/SQL Performance	781
	Tools to Assist in Optimization	783
709	Analyzing Memory Usage	783
710	Identifying Bottlenecks in PL/SQL Code	783
711	Calculating Elapsed Time	788
712	Choosing the Fastest Program	790
713	Avoiding Infinite Loops	792
714	Performance-Related Warnings	793
715	The Optimizing Compiler	793
716	Insights on How the Optimizer Works	795
717	Runtime Optimization of Fetch Loops	798
718	Data Caching Techniques	799
719	Package-Based Caching	800
720	Deterministic Function Caching	805
721	Function Result Cache (Oracle Database 11g)	807
722	Caching Summary	819
723	Bulk Processing for Multirow SQL	820
724	High Speed Querying with BULK COLLECT	821
725	High Speed DML with FORALL	828
726	Improving Performance With Pipelined Table Functions	838
727	Replacing Row-Based Inserts with Pipelined Function-Based Loads	839
728	Tuning Merge Operations with Pipelined Functions	846
729	Asynchronous Data Unloading with Parallel Pipelined Functions	848
730	Performance Implications of Partitioning and Streaming Clauses in Parallel Pipelined Functions	851
731	Pipelined Functions and the Cost-Based Optimizer	853
732	Tuning Complex Data Loads with Pipelined Functions	859
733	A Final Word on Pipelined Functions	866
734	Specialized Optimization Techniques	866
735	Using the NOCOPY Parameter Mode Hint	867
736	Using the Right Datatype	870
737	Stepping Back for the Big Picture on Performance	871
738		
22.	I/O and PL/SQL	873
	Displaying Information	873
739	Enabling DBMS_OUTPUT	874
740	Write Lines to the Buffer	874
741	Read the Contents of the Buffer	875

Reading and Writing Files	876
The UTL_FILE_DIR Parameter	877
Work with Oracle Directories	879
Open Files	880
Is the File Already Open?	882
Close Files	882
Read from Files	883
Write to Files	885
Copy Files	888
Delete Files	889
Rename and Move Files	890
Retrieve File Attributes	890
Sending Email	891
Oracle Prerequisites	893
Configuring Network Security	893
Send a Short (32,767 or Less) Plaintext Message	894
Include “Friendly” Names in Email Addresses	896
Send a Plaintext Message of Arbitrary Length	897
Send a Message with a Short (< 32,767) Attachment	898
Send a Small File (< 32767) as an Attachment	900
Attach a File of Arbitrary Size	900
Working with Web-Based Data (HTTP)	903
Retrieve a Web Page in “Pieces”	903
Retrieve a Web Page into a LOB	905
Authenticate Using HTTP Username/Password	906
Retrieve an SSL-Encrypted Web Page (Via HTTPS)	906
Submit Data to a Web Page via GET or POST	908
Disable Cookies or Make Cookies Persistent	912
Retrieve Data from an FTP Server	912
Use a Proxy Server	913
Other Types of I/O Available in PL/SQL	913
Database Pipes, Queues, and Alerts	914
TCP Sockets	914
Oracle’s Built-in Web Server	914

Part VI. Advanced PL/SQL Topics

23. Application Security and PL/SQL	919
Security Overview	919
Encryption	921
Key Length	922
Algorithms	923

	Padding and Chaining	924
	The DBMS_CRYPTO Package	925
	Encrypting Data	926
	Encrypting LOBs	929
	SecureFiles	930
	Decrypting Data	930
	Performing Key Generation	932
	Performing Key Management	932
	Cryptographic Hashing	938
	Using Message Authentication Codes	940
	Using Transparent Data Encryption (TDE)	941
	Transparent Tablespace Encryption	944
	Row-Level Security	945
	Why Learn About RLS?	947
	A Simple RLS Example	949
	Using Dynamic Policies	953
	Using Column-Sensitive RLS	957
	RLS Debugging	960
	Application Contexts	964
	Using Application Contexts	965
	Security in Contexts	966
	Contexts as Predicates in RLS	967
	Identifying Non-Database Users	970
	Fine-Grained Auditing	972
	Why Learn About FGA?	973
	A Simple FGA Example	974
	Access How Many Columns?	976
	Checking the Audit Trail	977
	Using Bind Variables	978
	Using Handler Modules	979
24.	PL/SQL Architecture	981
	Who (or What) is DIANA?	981
	How Does Oracle Execute PL/SQL Code?	982
	An Example	983
	Compiler Limits	986
	The Default Packages of PL/SQL	987
	Execution Authority Models	990
	The Definer Rights Model	990
	The Invoker Rights Model	995
	Combining Rights Models	997
	Conditional Compilation	998
	Examples of Conditional Compilation	999

1078	The INQUIRY Directive	1000
1078	The \$IF Directive	1004
1079	The \$ERROR Directive	1005
1084	Synchronizing Code with Packaged Constants	1006
1082	Program-Specific Settings with Inquiry Directives	1006
1092	Working with Postprocessed Code	1008
1092	PL/SQL and Database Instance Memory	1009
1102	PGA, UGA, and CGA	1010
1106	Cursors, Memory, and More	1011
1110	Tips on Reducing Memory Use	1013
1112	What to Do if You Run Out of Memory	1024
1116	Native Compilation	1027
1118	When to Run Interpreted Mode	1027
1121	When to Go Native	1028
1122	Native Compilation and Database Release	1028
1123	What You Need to Know	1029
1125		
1127	25. Globalization and Localization in PL/SQL	1031
1127	Overview and Terminology	1033
1128	Unicode Primer	1034
1130	National Character Set Datatypes	1036
	Character Encoding	1036
1131	Globalization Support Parameters	1037
	Unicode Functions	1038
1132	Character Semantics	1045
	String Sort Order	1049
1133	Binary Sort	1050
	Monolingual Sort	1050
1134	Multilingual Sort	1053
	Multilingual Information Retrieval	1054
	IR and PL/SQL	1056
	Date/Time	1059
	Timestamp Datatypes	1059
	Date/Time Formatting	1060
	Currency Conversion	1064
	Globalization Development Kit for PL/SQL	1066
	UTL_118N Utility Package	1066
	UTL_LMS Error-Handling Package	1069
	GDK Implementation Options	1070
	26. Object-Oriented Aspects of PL/SQL	1073
	Introduction to Oracle's Object Features	1073
	Object Types by Example	1075

Creating a Base Type	1076
Creating a Subtype	1078
Methods	1079
Invoking Supertype Methods in Oracle Database 11g	1084
Storing, Retrieving, and Using Persistent Objects	1085
Evolution and Creation	1093
Back to Pointers?	1095
Generic Data: The ANY Types	1102
I Can Do It Myself	1106
Comparing Objects	1110
Object Views	1115
A Sample Relational System	1116
Object View with a Collection Attribute	1118
Object Subview	1121
Object View with Inverse Relationship	1122
INSTEAD OF Triggers	1123
Differences Between Object Views and Object Tables	1125
Maintaining Object Types and Object Views	1127
Data Dictionary	1127
Privileges	1128
Concluding Thoughts from a (Mostly) Relational Developer	1130
A. Regular Expression Metacharacters and Function Parameters	1133
B. Number Format Models	1139
C. Date Format Models	1143
Index	1149