

1 Declaring Variables

- Objectives 1-2
- PL/SQL Block Structure 1-3
- Executing Statements and PL/SQL Blocks 1-4
- Block Types 1-5
- Program Constructs 1-6
- Use of Variables 1-7
- Handling Variables in PL/SQL 1-8
- Types of Variables 1-9
- Using iSQL*Plus Variables Within PL/SQL Blocks 1-10
- Types of Variables 1-11
- Declaring PL/SQL Variables 1-12
- Guidelines for Declaring PL/SQL Variables 1-13
- Naming Rules 1-14
- Variable Initialization and Keywords 1-15
- Scalar Data Types 1-17
- Base Scalar Data Types 1-18
- Scalar Variable Declarations 1-22
- The %TYPE Attribute 1-23
- Declaring Variables with the %TYPE Attribute 1-24
- Declaring Boolean Variables 1-25
- Composite Data Types 1-26
- LOB Data Type Variables 1-27
- Bind Variables 1-28
- Using Bind Variables 1-30
- Referencing Non-PL/SQL Variables 1-31
- DBMS_OUTPUT.PUT_LINE 1-32
- Summary 1-33
- Practice 1 Overview 1-35

2 Writing Executable Statements

- Objectives 2-2
- PL/SQL Block Syntax and Guidelines 2-3
- Identifiers 2-5
- PL/SQL Block Syntax and Guidelines 2-6
- Commenting Code 2-7
- SQL Functions in PL/SQL 2-8
- SQL Functions in PL/SQL: Examples 2-9
- Data type Conversion 2-10
- Nested Blocks and Variable Scope 2-12
- Identifier Scope 2-14
- Qualify an Identifier 2-15
- Determining Variable Scope 2-16
- Operators in PL/SQL 2-17
- Programming Guidelines 2-19
- Indenting Code 2-20
- Summary 2-21
- Practice 2 Overview 2-22

3 Interacting with the Oracle Server

- Objectives 3-2

- SQL Statements in PL/SQL 3-3
- SELECT Statements in PL/SQL 3-4
- Retrieving Data in PL/SQL 3-7
- Naming Conventions 3-9
- Manipulating Data Using PL/SQL 3-10
- Inserting Data 3-11
- Updating Data 3-12
- Deleting Data 3-13
- Merging Rows 3-13
- Naming Conventions 3-16
- SQL Cursor 3-18
- SQL Cursor Attributes 3-19
- Transaction Control Statements 3-21
- Summary 3-22
- Practice 3 Overview 3-24

4 Writing Control Structures

- Objectives 4-2
- Controlling PL/SQL Flow of Execution 4-3
- IF Statements 4-4
- Simple IF Statements 4-5
- Compound IF Statements 4-6
- IF-THEN-ELSE Statement Execution Flow 4-7
- IF-THEN-ELSE Statements 4-8
- CASE Expressions 4-12
- CASE Expressions: Example 4-13
- Handling Nulls 4-15
- Logic Tables 4-16
- Boolean Conditions 4-17
- Iterative Control: LOOP Statements 4-18
- Basic Loops 4-19
- WHILE Loops 4-21
- FOR Loops 4-23
- Guidelines While Using Loops 4-26
- Nested Loops and Labels 4-27
- Summary 4-29
- Practice 4 Overview 4-30

5 Working with Composite Data Types

- Objectives 5-2
- Composite Data Types 5-3
- PL/SQL Records 5-4
- Creating a PL/SQL Record 5-5
- PL/SQL Record Structure 5-7
- The %ROWTYPE Attribute 5-8
- Advantages of Using %ROWTYPE 5-10
- The %ROWTYPE Attribute 5-11
- INDEX BY Tables 5-13
- Creating an INDEX by Table 5-14
- INDEX BY Table Structure 5-15
- Creating an INDEX BY Table 5-16

Using INDEX BY Table Methods 5-17
INDEX BY Table of Records 5-18
Example of PL/SQL Table of Records 5-19
Summary 5-20
Practice 5 Overview 5-21

6 Writing Explicit Cursors

Objectives 6-2
About Cursors 6-3
Explicit Cursor Functions 6-4
Controlling Explicit Cursors 6-5
Declaring the Cursor 6-7
Opening the Cursor 6-9
Fetching Data from the Cursor 6-10
Closing the Cursor 6-12
Explicit Cursor Attributes 6-13
The %ISOPEN Attribute 6-14
Controlling Multiple Fetches 6-15
The %NOTFOUND and %ROWCOUNT Attributes 6-16
Example 6-18
Cursors and Records 6-19
Cursor FOR Loops 6-20
Cursor FOR Loops Using Subqueries 6-22
Summary 6-24
Practice 6 Overview 6-25

7 Advanced Explicit Cursor Concepts

Objectives 7-2
Cursors with Parameters 7-3
The FOR UPDATE Clause 7-5
The WHERE CURRENT OF Clause 7-7
Cursors with Subqueries 7-9
Summary 7-10
Practice 7 Overview 7-11

8 Handling Exceptions

Objectives 8-2
Handling Exceptions with PL/SQL 8-3
Handling Exceptions 8-4
Exception Types 8-5
Trapping Exceptions 8-6
Trapping Exceptions Guidelines 8-7
Trapping Predefined Oracle Server Errors 8-8
Predefined Exceptions 8-11
Trapping Nonpredefined Oracle Server Errors 8-12
Nonpredefined Error 8-13
Functions for Trapping Exceptions 8-14
Trapping User-Defined Exceptions 8-16
User-Defined Exception 8-17
Calling Environments 8-18
Propagating Exceptions 8-19

RAISE_APPLICATION_ERROR Procedure 8-20
RAISE_APPLICATION_ERROR 8-22
Summary 8-23
Practice 8 Overview 8-23

9 Creating Procedures

Objectives 9-2
PL/SQL Program Constructs 9-4
Overview of Subprograms 9-5
Block Structure for Anonymous PL/SQL Blocks 9-6
Block Structure for PL/SQL Subprograms 9-7
PL/SQL Subprograms 9-8
Developing Subprograms by Using iSQL*Plus 9-9
What Is a Procedure? 9-11
Syntax for Creating Procedures 9-12
Developing Procedures 9-13
Formal Versus Actual Parameters 9-14
Procedural Parameter Modes 9-15
Creating Procedures with Parameters 9-16
IN Parameters: Example 9-17
OUT Parameters: Example 9-18
Viewing OUT Parameters 9-20
IN OUT Parameters 9-21
Viewing IN OUT Parameters 9-22
Methods for Passing Parameters 9-23
DEFAULT Option for Parameters 9-24
Examples of Passing Parameters 9-25
Declaring Subprograms 9-26
Invoking a Procedure from an Anonymous PL/SQL Block 9-27
Invoking a Procedure from Another Procedure 9-28
Handled Exceptions 9-29
Unhandled Exceptions 9-31
Removing Procedures 9-33
Benefits of Subprograms 9-34
Summary 9-35
Practice 9 Overview 9-37

10 Creating Functions

Objectives 10-2
Overview of Stored Functions 10-3
Syntax for Creating Functions 10-4
Creating a Function 10-5
Creating a Stored Function by Using iSQL*Plus 10-6
Creating a Stored Function by Using iSQL*Plus: Example 10-7
Executing Functions 10-8
Executing Functions: Example 10-9
Advantages of User-Defined Functions in SQL Expressions 10-10
Invoking Functions in SQL Expressions: Example 10-11
Locations to Call User-Defined Functions 10-12
Restrictions on Calling Functions from SQL Expressions 10-13
Restrictions on Calling from SQL 10-15

Removing Functions 10-16
Procedure or Function? 10-17
Comparing Procedures and Functions 10-18
Benefits of Stored Procedures and Functions 10-19
Summary 10-20
Practice 10 Overview 10-21

11 Managing Subprograms

Objectives 11-2
Required Privileges 11-3
Granting Access to Data 11-4
Using Invoker's-Rights 11-5
Managing Stored PL/SQL Objects 11-6
USER_OBJECTS 11-7
List All Procedures and Functions 11-8
USER_SOURCE Data Dictionary View 11-9
List the Code of Procedures and Functions 11-10
USER_ERRORS 11-11
Detecting Compilation Errors: Example 11-12
List Compilation Errors by Using USER_ERRORS 11-13
List Compilation Errors by Using SHOW ERRORS 11-14
DESCRIBE in iSQL*Plus 11-15
Debugging PL/SQL Program Units 11-16
Summary 11-17
Practice 11 Overview 11-19

12 Creating Packages

Objectives 12-2
Overview of Packages 12-3
Components of a Package 12-4
Referencing Package Objects 12-5
Developing a Package 12-6
Creating the Package Specification 12-8
Declaring Public Constructs 12-9
Creating a Package Specification: Example 12-10
Creating the Package Body 12-11
Public and Private Constructs 12-12
Creating a Package Body: Example 12-13
Invoking Package Constructs 12-15
Declaring a Bodiless Package 12-17
Referencing a Public Variable from a Stand-alone Procedure 12-18
Removing Packages 12-19
Guidelines for Developing Packages 12-20
Advantages of Packages 12-21
Summary 12-23
Practice 12 Overview 12-26

13 More Package Concepts

Objectives 13-2
Overloading 13-3
Overloading: Example 13-4

- Using Forward Declarations 13-7
- Creating a One-Time-Only Procedure 13-9
- Restrictions on Package Functions Used in SQL 13-10
- User Defined Package: taxes_pack 13-11
- Invoking a User Defined Package Function from a SQL Statement 13-12
- Persistent State of Package Variables: Example 13-13
- Persistent State of Package Variables 13-14
- Controlling the Persistent State of a Package Cursor 13-15
- Executing PACK_CUR 13-17
- PL/SQL Tables and Records in Packages 13-18
- Summary 13-19
- Practice 13 Overview 13-20

14 Oracle Supplied Packages

- Objectives 14-2
- Using Supplied Packages 14-3
- Using Native Dynamic SQL 14-4
- Execution Flow 14-5
- Using the DBMS_SQL Package 14-6
- Using DBMS_SQL 14-8
- Using the EXECUTE IMMEDIATE Statement 14-9
- Dynamic SQL Using EXECUTE IMMEDIATE 14-11
- Using the DBMS_DDL Package 14-12
- Using DBMS_JOB for Scheduling 14-13
- DBMS_JOB Subprograms 14-14
- Submitting Jobs 14-15
- Changing Job Characteristics 14-17
- Running, Removing, and Breaking Jobs 14-18
- Viewing Information on Submitted Jobs 14-19
- Using the DBMS_OUTPUT Package 14-20
- Interacting with Operating System Files 14-21
- What Is the UTL_FILE Package? 14-22
- File Processing Using UTL_FILE 14-23
- UTL_FILE Procedures and Functions 14-24
- Exceptions Specific to the UTL_FILE Package 14-25
- The FOPEN and IS_OPEN Functions 14-26
- Using UTL_FILE 14-27
- UTL_HTTP Package 14-29
- Using the UTL_HTTP Package 14-30
- Using the UTL_TCP Package 14-31
- Oracle-Supplied Packages 14-32
- Summary 14-33
- Practice 14 Overview 14-34

15 Manipulating Large Objects

- Objectives 15-2
- What Is a LOB? 15-3
- Contrasting LONG and LOB Data Types 15-4
- Anatomy of a LOB 15-5
- Internal LOBs 15-6
- Managing Internal LOBs 15-7

- What Are BFILES? 15-8
- Securing BFILES 15-9
- A New Database Object: DIRECTORY 15-10
- Guidelines for Creating DIRECTORY Objects 15-11
- Managing BFILES 15-12
- Preparing to Use BFILES 15-13
- The BFILENAME Function 15-14
- Loading BFILES 15-15
- Migrating from LONG to LOB 15-17
- The DBMS_LOB Package 15-19
- DBMS_LOB.READ and DBMS_LOB.WRITE 15-22
- Adding LOB Columns to a Table 15-23
- Populating LOB Columns 15-24
- Updating LOBs by Using SQL 15-26
- Updating LOBs by Using DBMS_LOB in PL/SQL 15-27
- Selecting CLOB Values by Using SQL 15-28
- Selecting CLOB Values, Using DBMS_LOB 15-29
- Selecting CLOB Values in PL/SQL 15-30
- Removing LOBs 15-31
- Temporary LOBs 15-32
- Creating a Temporary LOB 15-33
- Summary 15-34
- Practice 15 Overview 15-36

16 Creating Database Triggers

- Objectives 16-2
- Types of Triggers 16-3
- Guidelines for Designing Triggers 16-4
- Database Trigger: Example 16-5
- Creating DML Triggers 16-6
- DML Trigger Components 16-7
- Firing Sequence 16-11
- Syntax for Creating DML Statement Triggers 16-13
- Creating DML Statement Triggers 16-14
- Testing SECURE_EMP 16-15
- Using Conditional Predicates 16-16
- Creating a DML Row Trigger 16-17
- Creating DML Row Triggers 16-18
- Using OLD and NEW Qualifiers 16-19
- Using OLD and NEW Qualifiers: Example Using Audit_Emp_Table 16-20
- Restricting a Row Trigger 16-21
- INSTEAD OF Trigger 16-22
- Creating an INSTEAD OF Trigger 16-23
- Differentiating between Database Triggers and Stored Procedures 16-27
- Differentiating between Database Triggers and Form Builder Triggers 16-28
- Managing Triggers 16-29
- DROP TRIGGER Syntax 16-30
- Trigger Test Cases 16-31
- Trigger Execution Model and Constraint Checking 16-32
- Trigger Execution Model and Constraint Checking: Example 16-33
- A Sample Demonstration for Triggers Using Package Constructs 16-34

After Row and After Statement Triggers 16-35
Demonstration: VAR_PACK Package Specification 16-36
Demonstration: Using the AUDIC_EMP Procedure 16-38
Summary 16-39
Practice 16 Overview 16-40

17 More Trigger Concepts

Objectives 17-2
Creating Database Triggers 17-3
Creating Triggers on DDL Statements 17-4
Creating Triggers on System Events 17-5
LOGON and LOGOFF Trigger Example 17-6
CALL Statement 17-7
Reading Data from a Mutating Table 17-8
Mutating Table: Example 17-9
Implementing Triggers 17-11
Controlling Security within the Server 17-12
Controlling Security with a Database Trigger 17-13
Using the Server Facility to Audit Data Operations 17-14
Auditing by Using a Trigger 17-15
Enforcing Data Integrity within the Server 17-16
Protecting Data Integrity with a Trigger 17-17
Enforcing Referential Integrity within the Server 17-18
Protecting Referential Integrity with a Trigger 17-19
Replicating a Table within the Server 17-20
Replicating a Table with a Trigger 17-21
Computing Derived Data within the Server 17-22
Computing Derived Values with a Trigger 17-23
Logging Events with a Trigger 17-24
Benefits of Database Triggers 17-26
Managing Triggers 17-27
Viewing Trigger Information 17-28
Using USER_TRIGGERS 17-29
Listing the Code of Triggers 17-30
Summary 17-31
Practice 17 Overview 17-32

18 Managing Dependencies

Objectives 18-2
Understanding Dependencies 18-3
Dependencies 18-4
Local Dependencies 18-5
A Scenario of Local Dependencies 18-6
Displaying Direct Dependencies by Using USER_DEPENDENCIES 18-7
Displaying Direct and Indirect Dependencies 18-8
Displaying Dependencies 18-9
Another Scenario of Local Dependencies 18-10
A Scenario of Local Naming Dependencies 18-11
Understanding Remote Dependencies 18-12
Concepts of Remote Dependencies 18-13
REMOTE_DEPENDENCIES_MODE Parameter 18-14

Remote Dependencies and Time stamp Mode 18-15
Remote Procedure B Compiles at 8:00 a.m. 18-16
Local Procedure A Compiles at 9:00 a.m. 18-17
Execute Procedure A 18-18
Remote Procedure B Recompiled at 11:00 a.m. 18-19
Execute Procudre A 18-20
Signature Mode 18-21
Recompiling a PL/SQL Program Unit 18-22
Unsuccessful Recompilation 18-23
Successful Recompilation 18-24
Recompilation of Procedures 18-25
Packages and Dependencies 18-26
Summary 18-28
Practice 18 Overview 18-29