



Oracle Database 12c: PL/SQL II – Intermediate: Develop Program Units



ora12c130-ver2

Learning Resources Available From The Sideris Training Portal

3 Days

General Description

This textbook is intended for those who have already learned the basics of the Oracle PL/SQL database programming language and its syntax, and who are now ready to employ the language in the development of database applications. In particular, the focus of this textbook will be on the use of database-resident stored program units such as procedures, functions, packages and database triggers. New features introduced with the Oracle 12c release of the database are also explained and demonstrated.

Target Audience

The target audience for this textbook is all Oracle professionals. Among the specific groups for whom this textbook will be helpful are:

- Application designers and database developers
- Database administrators
- Web server administrators

About Series

This textbook is part of the Sideris Oracle PL/SQL Database Developer series, which in turn is one of the learning paths from the Sideris Oracle Database 12c: SQL & PL/SQL Programming curriculum.

Prerequisites

Either of the Sideris textbooks Oracle Database 12c: SQL Fundamentals (Levels I & II) or Oracle Database 12c: SQL Complete Library (Levels I, II, & III) are recommended prerequisites for this textbook.

Certification

This textbook considers subjects applicable to certification as both an Oracle PL/SQL Developer Certified Associate (OCA) and an Oracle Advanced PL/SQL Developer Certified Professional (OCP). The topics considered are included within "Exam 1Z0-144: Program With PL/SQL".

Content Summary

Volumes: 2
Pages: 527
Workshops: 11
Exercises: 63

Training Suggestions

This textbook may be used as one module within a Sideris textbook kit entitled Oracle Database 12c: SQL Fundamentals (Levels I & II). When delivered in this format as an intensive instructor-led training (ILT) or live virtual training session at an accelerated pace, the suggested presentation length of the modules included in this kit is 5 days. The sequence in which the modules should be considered is:

- Oracle Database 12c: PL/SQL I – Introduction
- Oracle Database 12c: PL/SQL I – Intermediate: Develop Program Units

© 2015 Sideris Courseware Corporation

831 Beacon Street, Suite 295, Newton, MA 02459

Phone: +1.617.965.9800 ▪ info@sideris.com ▪ www.sideris.com



Objectives

Instructor-Led Training (ILT) / Live Virtual Training Schedule [standalone module]

This textbook may also be used either as a standalone course or as one module assembled with others in a customized training course.

When delivered as a standalone instructor-led training (ILT) or live virtual training session, the suggested length of this module is 3 days. This allows sufficient time to explore all the topics in their entirety and for students to complete all the workshop exercises.

This textbook explains how database-resident program units can be used as part of the overall database application architecture and the benefits realized by doing so. It then builds upon one's knowledge of database-resident program units and applies these to the development of PL/SQL packages. In a production environment most PL/SQL program units should be packaged, and these advanced database programming capabilities along with the benefits of using these are discussed. It concludes with extensive demonstrations on how a particular type of database-resident program unit known as a database trigger can be used as part of an advanced database application design.

Contents

Introducing Database-Resident Program Units

- About Database-Resident Programs
- Physical Storage & Execution
- Types Of Stored Program Units
- Stored Program Unit Advantages
- Modular Design Principles

Creating Stored Procedures & Functions

- Stored Procedures & Functions
- CREATE Procedure / CREATE Function
- Creating Procedures & Functions
- RAISE_SALARY() Procedure
- SALARY_VALID() Function
- The Parameter Specification
- DEFAULT Clause
- SYSTEM & OBJECT Privileges

- Using The Development Tools

Executing Stored Procedures & Functions

- Calling Procedures & Functions
- Unit Testing With EXECUTE
- ANONYMOUS BLOCK Unit Testing
- Specifying A Parameter Notation
- SQL Worksheet Unit Testing
- Calling Functions From SQL

Maintaining Stored Program Units

- Recompiling Programs
- Mass Recompilation Using UTL_RECOMP()
- Dropping Procedures & Functions
- DROP Procedure / Function
- Data Dictionary Metadata



- Using USER_OBJECTS
- Using USER_SOURCE
- Using USER_ERRORS
- Using USER_OBJECT_SIZE
- Using USER_DEPENDENCIES

Managing Dependencies

- DEPENDENCY INTERNALS
- TRACKING DEPENDENCIES
- The DEPENDENCY TRACKING Utility
- SQL Developer Dependency Info
- Dependency Strategy Checklists

Creating & Maintaining Packages

- About Packages
- Creating Packages
- Maintaining Packages
- Performance Considerations

Advanced Package Capabilities

- Definer & Invoker Rights
- White Lists & Accessible By
- Persistent Global Objects
- Defining Initialization Logic
- Object Orientation Support

Advanced Cursor Techniques

- USING CURSOR VARIABLES
- Using SYS_REFCURSOR
- Using CURSOR Expressions

Using System-Supplied Packages

- DBMS_OUTPUT()
- UTL_FILE()
- FOPEN() Example

Database Trigger Concepts

- About Database Triggers
- DML EVENT TRIGGER Sub-Types
- DATABASE TRIGGER Scenario
- TRIGGER Execution Mechanisms
- TRIGGERS Within SQL Worksheet

Creating Database Triggers

- STATEMENT-LEVEL TRIGGERS
- Using RAISE_APPLICATION_ERROR()
- ROW-LEVEL TRIGGERS
- EXAMPLES OF TRIGGERS
- EMPLOYEE_SALARY_CHECK Example
- EMPLOYEE_JOURNAL Example
- BUDGET_EVENT Example
- INSTEAD OF TRIGGERS
- Triggers Within An Application

Maintaining Database Triggers

- CALL Syntax
- Trigger Maintenance Tasks
- SHOW ERRORS Trigger
- DROP Trigger
- ALTER Trigger
- Multiple Triggers For A Table
- Handling Mutating Table Issues



Implementing System Event Triggers

- What Are System Event Triggers?
- Defining The Scope
- Available System Events
- System Event Attributes