

Oracle Database 10g: Develop PL/SQL Program Units

Duration: 3 Days

What you will learn

This course is designed for developers with basic PL/SQL and SQL language skills. In class students learn to develop, execute and manage PL\SQL stored program units like procedures, functions, packages and database triggers. Student also learn to manage object dependencies and recompilation of invalid objects. This course also describes the characteristics and ways of manipulation of large objects. Students are introduced to the utilization of some of the Oracle-supplied packages. This course counts towards the Hands-on course requirement for the Oracle Database 10g Administrator Certification. Only instructor-led inclass or instructor-led online formats of this course will meet the Certification Hands-on Requirement. Self Study CD-Rom and Knowledge Center courses are excellent study and reference tools but DO NOT meet the Hands-on Requirement for certification.

Learn To: Create and manage Procedures and Functions
Manage Dependencies and Large Objects
Create Packages and maintain them
Create and apply Triggers
Utilizing Oracle-Supplied Packages in Application Development
Understand and influence the PL/SQL Compiler

Audience

Application Developers
Database Administrators
Database Designers
Forms Developer
PL/SQL Developer

Prerequisites

Required Prerequisites

Oracle Database 10g: PL/SQL Fundamentals

Oracle Database 10g: Introduction to SQL

Course Objectives

Write dynamic SQL for more coding flexibility
Design PL/SQL code for predefined data types, local sub-programs, additional programs and standardized constants and
Use the compiler warnings infrastructure
Manipulate large objects
Create triggers to solve business challenges
Manage dependencies between PL/SQL sub-programs
Schedule PL/SQL jobs to run independently
Create stored procedures and functions
Design PL/SQL packages to group and contain related constructs
Create overloaded package sub-programs for more flexibility
Categorize the Oracle supplied PL/SQL packages
Use the Oracle supplied PL/SQL packages to generate screen output, file output, web output, and mail output

Course Topics

Introduction

Modularize program development with PL/SQL blocks

Describe the Human Resources (HR) data set

Show the PL/SQL environment (PLSQL block / PLSQL engine / Oracle server and statement executor)

Show the PL/SQL environment (PLSQL block / PLSQL engine / Oracle server and statement executor)

Code PL/SQL in the SQL*Plus environment

Code PL/SQL in the JDeveloper environment

Create and invoke simple procedures and functions

Creating Stored Procedures

Describe the block structure for PL/SQL stored procedures

Invoke a stored procedure/function from different tools

Call a stored procedure with host variables from iSQL*Plus, Forms, Java, C, etc

Invoke a stored procedure from an anonymous block or another stored procedure

List the CREATE OR REPLACE PROCEDURE syntax

Identify the development steps for creating a stored procedure

Use the SHOW ERRORS command

View source code in the USER_SOURCE dictionary view

Creating Stored Functions

Describe stored functions

List the CREATE OR REPLACE FUNCTION syntax

Identify the steps to create a stored function

Execute a stored function

Identify the advantages of using stored functions in SQL statements

Identify the restrictions of calling functions from SQL statements

Remove a function

Creating Packages

List the advantages of packages

Describe packages

Show the components of a package Diagram the visibility of constructs within a package

Develop a package

Create the package specification

Declare public constructs

Create the package body

Using More Package Concepts

List the benefits of overloading

Show overloading example

Use forward declarations in packages

Create a one-time only procedure (package code initialization)

List the restrictions on package functions used in SQL

Encapsulate code in a package demonstration

Invoke a user-defined package function from a SQL statement

Utilize the persistent state of package variables

Utilizing Oracle Supplied Packages in Application Development

List the various uses for the Oracle supplied packages

Reuse pre-packaged code to complete various tasks from developer to DBA purposes

Use the DESCRIBE command to view the package specifications and overloading

Explain how DBMS_OUTPUT works (in conjunction with SET SERVEROUTPUT ON)

Interact with operating system files with UTL_MAIL

Describe file processing with UTL_FILE

Review UTL_FILE routines and exceptions

Use UTL_FILE to generate a report to a file

Dynamic SQL and Metadata

Describe using native dynamic SQL

List the execution flow of SQL

Show the syntax for the EXECUTE IMMEDIATE statement for native dynamic SQL

Create a procedure to generate native dynamic SQL using EXECUTE IMMEDIATE to delete rows from a table

Describe the DBMS_SQL package

Provide an example of DBMS_SQL

List the advantages of using Native Dynamic SQL Over the DBMS_SQL package

Design Considerations for PL/SQL Code

Standardize constants with a constant package

Standardize exceptions with an exception handling package

Introduce local sub-programs

Use local sub-programs

Track run time errors with an exception package

Describe the NOCOPY compiler hint

Use the NOCOPY compiler hint

Explain the effects of NOCOPY

Managing Dependencies

Define dependent and referenced objects

Diagram dependencies with code, views, procedures, and tables

Manage local dependencies between a procedure, view, and a table

Analyze a scenario of local dependencies

Display direct dependencies using the USER_DEPENDENCIES view

Run the UTL_DTREE.SQL script to create objects that enable you to view direct and indirect dependencies

Predict the effects of changes on dependent objects

Manipulating Large Objects

Describe a LOB object

Diagram the anatomy of a LOB

Manage and list the features on internal LOBs

Describe, manage, and secure BFILEs

Create and use the DIRECTORY object to access and use BFILEs

Prepare BFILEs for usage

Use the BFILENAME function to load BFILEs

Describe the DBMS_LOB package

Creating Triggers

Describe the different types of triggers and how they execute

List the benefits and guidelines of using database triggers

Show how triggers are executed with a basic database trigger example

Show syntax and create DML triggers, and list the DML trigger components

Explain the firing sequence of triggers

Create a DML statement and row level triggers

Use the OLD and NEW qualifiers to reference column values

Use conditional predicates with triggers

Applications for Triggers

Describe the different types of triggers and how they execute

List the benefits and guidelines of using database triggers

Show a LOGON and LOGOFF trigger example

Use the CALL statement to call a stored procedure from a trigger

Define a mutating table

Show a mutating table code example

Describe business application scenarios for implementing with triggers

Describe the privileges required to manage triggers

Understanding and Influencing the PL/SQL Compiler

Describe the features of the PL/SQL compiler in Oracle Database 10g

List the features of native compilation

Identify the 3 parameters used to influence compilation (PLSQL_CODE_TYPE, PLSQL_DEBUG, PLSQL_OPTIMIZE_LE

Show how to set the parameters

Describe the dictionary view used to see how code is compiled (USER_PLSQL_OBJECTS)

Change the parameter settings, recompile code, and view the results

Describe the compiler warning infrastructure in Oracle Database 10g

List the steps used in setting compiler warning levels