

Oracle 11g INTERMEDIATE PL/SQL

Duration: 3 days

Compile PL/SQL Programs, Update, Insert, and Delete data, Define and Execute Testing Procedures and Functions, and gain an understanding of Advanced PL/SQL Features.

Course Description:

You will receive hands-on instruction in the intermediate and advanced features of Oracle11g PL/SQL procedural language for SQL. You will cover how to control data sharing and locking, debug and use error reporting procedures use Oracle supplied packages, and develop database triggers, stored procedures and functions for future reuse.

You will cover...

- The PL/SQL Environment
- PL/SQL Program Structure
- Native Compilation
- Update, Insert and Delete Statements
- Error Functions / Debugging
- Defining, Executing and Testing Functions And Procedures
- Creating Package Specifications And Bodies
- Creating Triggers Of All Types
- Using Oracle Supplied Packages
- Understanding Advanced PL/SQL Features

Who Needs to Attend:

Application Developers and Database Administrators who need a comprehensive understanding of Oracle 10g PL/SQL language.

Prerequisites:

Oracle 11g SQL Basics & SQL*PLUS or Equivalent experience, and Oracle 11g PL/SQL Introduction or Equivalent Experience.

Course Labs:

1. Creating Procedures
2. Creating Functions
3. Creating Packages
4. Creating Triggers
5. Embedded Functions and Procedures
6. Creating Autonomous Transactions
7. Encrypting Source Code
8. Using the UTL_FILE Package
9. Using the DBMS_ALERT Package
10. Creating Object Types
11. Creating and Manipulating Object Tables
12. Working With Collections
13. Collections And Bulk Binding
14. DBMS_SQL / Native Dynamic SQL
15. Working With Large Objects
16. Load, Publish And Run Java

Course Content:

Review Of Introduction to PL/SQL

- Features and benefits of PL/SQL
- Relationship of PL/SQL to SQL
- PL/SQL development tools
- Native Compilation
- PL/SQL System/Session Params
- PL/SQL anonymous block
- Variable declarations
- PL/SQL types and Records
- Declaring variable datatypes dynamically
- Modifying database data (DML)
- Transaction control statements
- Declaring explicit cursors
- Implicit cursor attributes
- Exception Handling
- Creating Procedures, Functions, Packages, and DML Triggers
- Debugging With SQL Developer
- Conditional Compilation
- Warnings

Advanced Cursors

- Cursor parameters
- Parameter defaults
- Taking advantage of a weak cursor variable
- OPEN FOR, FETCH and CLOSE
- Using the FOR UPDATE clause
- Using PL/SQL collections and nested collections

Advanced Packages

- Initializing variables
- Module Overloading
- Recursion
- Purity levels
- Using the "Persistent State" to advantage
- One Time Only Procedures
- Forward Declarations
- Using Persistent State
- Code Encapsulation
- Constant and Exception Standardization

Advanced Triggers

- Trigger limitations
- Mutating and Constraining Tables
- Using CALL and client triggers
- DDL / SERVERERROR triggers
- LOGON/LOGOFF, SUSPEND, STARTUP/SHUTDOWN triggers
- Schema vs. Database triggers
- Using alternative events and levels
- INSTEAD OF triggers on views

PL/SQL Composite Datatypes and Collections

- PL/SQL records
- Associative Arrays (INDEX BY)
- Nested Tables
- VARRAYs
- Built-in type methods
- Arrays of composite types
- Using PL/SQL record variables
- PL/SQL collections

Bulk-Bind Data Loading Using PL/SQL

- Defining bulk binds
- BULK COLLECT / FORALL
- Error handling with bulk binds – SAVE EXCEPTIONS

Using Oracle Supplied Packages

- DBMS_OUTPUT Package
- UTL_FILE Package (file i/o)
- DBMS_ALERT Package
- DBMS_PIPE Package
- DBMS_JOB Package
- DBMS_SCHEDULER Package
- DBMS_STATS Package
- DBMS_UTILITY Package
- UTL_SMTP Package
- UTL_MAIL Package
- DBMS_SQL Package

Writing Native Dynamic SQL

- EXECUTE IMMEDIATE
- Benefits Of NDS

PL/SQL Wrapper

- PL/SQL wrapper (source code encryption)

Understanding Dependencies

- Defining dependencies
- Local vs. Remote
- Viewing dependencies
- Effect of breaking dependency chain

Large Object Management in PL/SQL

- Differences between LONG/LONG RAW and LOBs
- Creating and using BFILEs
- Creating tables with LOBs
- LOBs and PL/SQL
- DBMS_LOB capabilities
- Temporary LOBs

Objects

- Basic Objects
- Object Inheritance

Java

- Java in PL/SQL
- Sample encapsulation