Document Generated: 01/09/2026

Learning Style: Virtual Classroom

Technology:

Difficulty: Intermediate

Course Duration: 2 Days

Next Course Date: April 13, 2026

Oracle PL/SQL Fundamentals (TTOR12019)

About This Course:

This intensive course trains you in Oracle PL/SQL programming. You will master writing stored procedures, functions, packages, triggers, and working with advanced PL/SQL features. You will practice in hands-on labs against real Oracle

environments. After the course you will be able to design, implement, debug, and deploy PL/SQL code aligned with Oracle best practices.

Course Objectives:

- Structure and execute PL/SQL blocks (anonymous, named)
- Create procedures, functions, packages, and triggers
- Use cursors, explicit and implicit, and bulk operations
- Handle exceptions and design robust error handling
- Leverage advanced features: collections, records, object types
- Manage large data sets with bulk binds, FORALL, dynamic SQL
- Understand PL/SQL compiler optimization, profiling, and performance tuning
- Implement secure, maintainable PL/SQL programs

Audience:

- Developers building Oracle applications
- DBAs supporting PL/SQL environments

Prerequisites:

- SQL knowledge
- Familiarity with Oracle Database concepts

Course Outline:

1) PL/SQL Basics & Block Structure

- Anatomy of a PL/SQL block (declaration, executable, exception)
- Anonymous blocks vs named blocks
- Variables, constants, datatypes
- %TYPE, %ROWTYPE usage

Lab

- Write simple anonymous blocks
- Declare and initialize variables
- Use %TYPE and %ROWTYPE in blocks

2) Control Structures & Loops

- Conditional logic (IF, CASE)
- Loop constructs: LOOP, WHILE, FOR
- EXIT, GOTO, nested loops

Lab

- Build blocks with IF and CASE
- Write loops to process sets of data
- Use EXIT, nested loops for real logic

3) Cursors & Cursor Management

- Implicit vs explicit cursors
- Cursor FOR loops
- Parameterized cursors
- Cursor attributes (FOUND, NOTFOUND, %ROWCOUNT)

Lab

- Create explicit cursors over queries
- Use cursor FOR loops
- Parameterize cursors and fetch attributes

4) Procedures & Functions

- Create procedures and functions
- IN, OUT, IN OUT parameters
- Deterministic vs non-deterministic functions
- Function calls in SQL

Lab

- Build stored procedures and functions
- Pass parameters and return values
- Call functions in SQL and PL/SQL contexts

5) Packages & Modular PL/SQL Design

- Package specification and body
- Public vs private elements
- Initialization blocks
- Overloading and package state

Lab

- Design and build a package
- Encapsulate utility code
- Use overloading and initialization

6) Exception Handling & Logging

- Predefined and user-defined exceptions
- RAISE, PRAGMA EXCEPTION_INIT
- Logging and error propagation
- WHEN OTHERS best practices

Lab

- Include exception blocks in procedures
- Define custom errors and map to codes
- Log errors to tables or files

7) Collections, Records & Data Structures

- PL/SQL collections: nested tables, varrays, associative arrays
- Records and %ROWTYPE
- Multilevel data structures
- Using collections in programs

Lab

- Create and manipulate collections
- Use records to aggregate data
- Combine collections and records

8) Bulk Processing & Performance Techniques

- FORALL, BULK COLLECT
- LIMIT clause
- Bulk binds and reducing context switches
- Dynamic SQL (EXECUTE IMMEDIATE, DBMS_SQL)

Lab

- Rework cursor logic using bulk operations
- Use FORALL and BULK COLLECT
- Write dynamic SQL blocks

9) Advanced PL/SQL Features

- Autonomous transactions
- PRAGMA SERIALLY_REUSABLE, NOCOPY
- Optimizing PL/SQL: PLSQL_OPTIMIZE_LEVEL, profiling
- Security: invoker’s rights vs definer’s rights

Lab

- Build a routine with autonomous transactions
- Test NOCOPY and evaluate output
- Profile PL/SQL units
- Demonstrate privilege models