

# Oracle Database 10g: Advanced PL/SQL

## What you will learn:

---

This class is applicable to Oracle8i, Oracle9i and Oracle Database 10g users.

In this course, students learn how to use the advanced features of PL/SQL in order to design and tune PL/SQL to interface with the database and other applications in the most efficient manner. Using advanced features of program design, packages, cursors, extended interface methods, and collections, students learn how to write powerful PL/SQL programs. Programming efficiency, use of external C and Java routines, PL/SQL server pages, and fine-grained access are covered.

This course counts towards the [Hands-on course requirement](#) for the Oracle Database 10g Administrator certification.

## Audience:

PL/SQL Developer  
Technical Consultant  
Database Designers

## Prerequisites:

## Required Prerequisites:

[Oracle Database 10g: Program with PL/SQL](#)

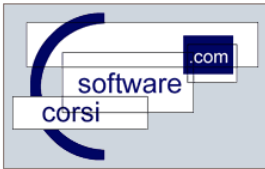
## Suggested Prerequisites:

Understanding of HTML syntax

## Course Objectives:

Design PL/SQL packages and program units that execute efficiently  
Write code to interface with external applications and the operating system  
Create PL/SQL applications that use collections  
Write and tune PL/SQL code effectively to maximize performance  
Implement a virtual private database with fine-grained access control  
Perform code analysis to find program ambiguities, test, trace, and profile PL/SQL code

---



## Course Topics:

### Introduction

Course objectives  
The Oracle complete solution  
Course agenda  
Tables and data used for this course

### PL/SQL Programming Concepts Review

Identify PL/SQL block structure  
Create procedures  
Create functions  
Create packages  
Use cursors  
Handle exceptions  
Understand dependencies  
Identify the Oracle supplied packages

### Design Considerations

List the different guidelines for cursor design  
Describe cursor variables  
Pass cursor variables as program parameters  
Compare cursor variables to static cursors  
Describe the predefined data types  
Create subtypes based on existing types for an application

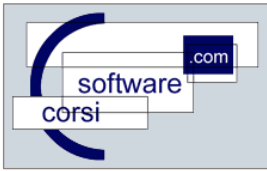
### Collections

Describe and use nested tables  
Describe and use varrays  
Describe and use associative arrays  
Describe and use string indexed collections  
Describe and use nested collections  
Write PL/SQL programs that use collections  
Describe the common collection exceptions and how to code for them  
Compare associative arrays to collections

### Advanced Interface Methods

Execute external C routines from PL/SQL  
Understand the benefits of external routines  
Publish the external C routine in the PL/SQL code  
Execute a PL/SQL routine that calls the external C routine  
Execute Java routines from PL/SQL  
Publish the Java class method by creating the PL/SQL subprogram unit specification that references the Java class method  
Execute the PL/SQL subprogram that invokes the Java class method

### PL/SQL Server Pages



Define embedding PL/SQL code in Web pages (PL/SQL Server Pages)  
Describe the format of a PL/SQL Server Page  
Write the code and content for the PL/SQL Server Page  
Load the PL/SQL Server Page into the database as a stored procedure  
Run a PL/SQL Server Page via a URL  
Debug PL/SQL Server Page problems

## **Fine Grained Access Control**

Understand how fine-grained access control works overall  
Describe the features of fine-grained access control  
Describe an application context  
Set up a logon trigger  
View the results  
Query the dictionary views holding information on fine-grained access

## **Performance and Tuning**

Tune PL/SQL code  
Write smaller executable sections of code  
Compare SQL to PL/SQL on performance  
Understand how bulk binds can improve performance  
Handle exceptions with the FORALL syntax  
Identify data type and constraint issues  
Recognize network issues

## **Analyzing PL/SQL Code**

Use the supplied packages and dictionary views to find coding information  
dbms\_describe supplied package  
Use supplied packages to find error information  
Trace PL/SQL programs using the dbms\_trace supplied package  
Read and interpret the trace information  
Profile PL/SQL programs using the dbms\_profiler supplied package  
Read and interpret the profiler information