Oracle Database: SQL and PL/SQL Fundamentals

Course Number 80198 – 40 Hours

Overview

This Oracle Database: SQL and PL/SQL Fundamentals training delivers the fundamentals of SQL and PL/SQL along with the benefits of the programming languages using Oracle Database technology. You'll explore the concepts of relational databases.

Learn To:

- Write queries against single and multiple tables, manipulate data in tables and create database objects.
- Use single row functions to customize output.
- Invoke conversion functions and conditional expressions.
- Use group functions to report aggregated data.
- Create PL/SQL blocks of application code that can be shared by multiple forms, reports and data management applications.
- Develop anonymous PL/SQL blocks, stored procedures and functions.
- Declare identifiers and trap exceptions.
- Use DML statements to manage data.
- Use DDL statements to manage database objects.
- Declare PL/SQL Variables.
- Conditionally control code flow (loops, control structures).
- Describe stored procedures and functions.
- Retrieve row and column data from tables.

Benefits to You

Ensure fast, reliable, secure and easy to manage performance. Optimize database workloads, lower IT costs and deliver a higher quality of service by enabling consolidation onto database clouds.

On Completion, Delegates will be able to

- Run data manipulation statements (DML) to update data in the Oracle Database.
- Design PL/SQL anonymous block that execute efficiently.
- Describe the features and syntax of PL/SQL.
- Handle runtime errors.
- Describe stored procedures and functions.
- Use PL/SQL programming constructs and conditionally control code flow (loops, control structures, and explicit cursors).
- Use cursors to process rows.
- Identify the major structural components of the Oracle Database 11g.

- Retrieve row and column data from tables with the SELECT statement.
- Create reports of sorted and restricted data.
- Employ SQL functions to generate and retrieve customized data.
- Display data from multiple tables using the ANSI SQL 99 JOIN syntax.
- Create reports of aggregated data.
- Run data definition language (DDL) statements to create and manage schema objects.

Who Should Attend

- Application Developers
- Forms Developer
- Functional Implementer
- PL/SQL Developer
- Portal Developer
- Reports Developer
- Technical Consultant

Prerequisites

- Familiarity with data processing concepts and techniques
- Familiarity with programming concepts

Course Contents

Introduction

- Overview of Oracle Database 12c and related products
- Overview of relational database management concepts and terminologies
- Introduction to SQL and its development environments
- The HR schema and the tables used in this course
- Oracle Database documentation and additional resources

Retrieve Data using the SQL SELECT Statement

- List the capabilities of SQL SELECT statements
- Generate a report of data from the output of a basic SELECT statement
- Use arithmetic expressions and NULL values in the SELECT statement
- Invoke Column aliases
- Concatenation operator, literal character strings, alternative quote operator, and the DISTINCT keyword
- Display the table structure using the DESCRIBE command

Restricted and Sorted Data

- Write queries with a WHERE clause to limit the output retrieved
- Describe the comparison operators and logical operators
- Describe the rules of precedence for comparison and logical operators
- Usage of character string literals in the WHERE clause
- Write queries with an ORDER BY clause
- Sort the output in descending and ascending order
- Substitution Variables

Usage of Single-Row Functions to Customize Output

- List the differences between single row and multiple row functions
- Manipulate strings using character functions
- Manipulate numbers with the ROUND, TRUNC, and MOD functions
- Perform arithmetic with date data
- Manipulate dates with the DATE functions

Conversion Functions and Conditional Expressions

- Describe implicit and explicit data type conversion
- Describe the TO_CHAR, TO_NUMBER, and TO_DATE conversion functions
- Nesting multiple functions
- Apply the NVL, NULLIF, and COALESCE functions to data
- Usage of conditional IF THEN ELSE logic in a SELECT statement

Aggregated Data Using the Group Functions

- Usage of the aggregation functions in SELECT statements to produce meaningful reports
- Describe the AVG, SUM, MIN, and MAX function
- How to handle Null Values in a group function?
- Divide the data in groups by using the GROUP BY clause
- Exclude groups of date by using the HAVING clause

Display Data From Multiple Tables

- Write SELECT statements to access data from more than one table
- Join Tables Using SQL:1999 Syntax
- View data that does not meet a join condition by using outer joins
- Join a table to itself by using a self join
- Create Cross Joins

Usage of Subqueries to Solve Queries

- Use a Subquery to Solve a Problem
- Single-Row Subqueries
- Group Functions in a Subquery
- Multiple-Row Subqueries
- Use the ANY and ALL Operator in Multiple-Row Subqueries
- Use the EXISTS Operator

SET Operators

- Describe the SET operators
- Use a SET operator to combine multiple queries into a single query
- Describe the UNION, UNION ALL, INTERSECT, and MINUS Operators
- Use the ORDER BY Clause in Set Operations

Data Manipulation

- Add New Rows to a Table
- Change the Data in a Table
- Use the DELETE and TRUNCATE Statements
- How to save and discard changes with the COMMIT and ROLLBACK statements
- Implement Read Consistency
- Describe the FOR UPDATE Clause

DDL Statements to Create and Manage Tables

- Categorize Database Objects
- Create Tables
- Describe the data types
- Understand Constraints
- Create a table using a subquery
- How to alter a table?
- How to drop a table?

Other Schema Objects

- Create, modify, and retrieve data from a view
- Perform Data manipulation language (DML) operations on a view
- How to drop a view?
- Create, use, and modify a sequence
- Create and drop indexes
- Create and drop synonyms

Introduction to PL/SQL

- PL/SQL Overview
- List the benefits of PL/SQL Subprograms
- Overview of the Types of PL/SQL blocks
- Create a Simple Anonymous Block
- Generate the Output from a PL/SQL Block

PL/SQL Identifiers

- List the different Types of Identifiers in a PL/SQL subprogram
- Usage of the Declarative Section to Define Identifiers
- Use of variables to store data
- Scalar Data Types
- %TYPE Attribute
- Bind Variables
- Sequences in PL/SQL Expressions

Write Executable Statements

- Basic PL/SQL Block Syntax Guidelines
- How to comment code?
- SQL Functions in PL/SQL
- Data Type Conversion
- Nested Blocks
- Operators in PL/SQL

Interaction with the Oracle Server

- SELECT Statements in PL/SQL to Retrieve data
- Data Manipulation in the Server Using PL/SQL
- The SQL Cursor concept
- Learn to use SQL Cursor Attributes to Obtain Feedback on DML
- How to save and discard transactions?

Control Structures

- Conditional processing Using IF Statements
- Conditional processing Using CASE Statements
- Simple Loop Statement
- While Loop Statement
- For Loop Statement
- The Continue Statement

Usage of Composite Data Types

- PL/SQL Records
- The %ROWTYPE Attribute
- Insert and Update with PL/SQL Records
- Associative Arrays (INDEX BY Tables)
- INDEX BY Table Methods
- INDEX BY Table of Records

Explicit Cursors

- Understand Explicit Cursors
- Declare the Cursor
- How to open the Cursor?
- Fetching data from the Cursor
- How to close the Cursor?
- Cursor FOR loop
- Explicit Cursor Attributes
- FOR UPDATE Clause and WHERE CURRENT Clause

Exception Handling

- What are Exceptions?
- Handle Exceptions with PL/SQL
- Trap Predefined Oracle Server Errors
- Trap Non-Predefined Oracle Server Errors
- Trap User-Defined Exceptions
- Propagate Exceptions
- RAISE_APPLICATION_ERROR Procedure

Stored Procedures and Functions

- What are Stored Procedures and Functions?
- Differentiate between anonymous blocks and subprograms
- Create a Simple Procedure
- Create a Simple Procedure with IN parameter
- Create a Simple Function
- Execute a Simple Procedure
- Execute a Simple Function