

## Oracle Database: Develop PL/SQL Program Units

### خلاصه :

این دوره برای برنامه نویسی در پایگاه داده اوراکل طراحی شده است ، پس از طی این دوره شما قادر خواهید بود توسط زبان PL/SQL در پایگاه داده اوراکل زیر برنامه هایی از قبیل procedure ، function ، trigger ، package و غیره را ایجاد کنید

مدت دوره: ۳۰ ساعت

پیش نیاز: Oracle SQL & PL/SQL

### مخاطبین:

- برنامه نویسان اوراکل که از پایگاه داده اوراکل جهت پیاده سازی کسب و کار برنامه های خود استفاده میکنند
- برنامه نویسان زبانهای گوناگون (java ، .net و غیره ) که جهت نگهدارای اطلاعات برنامه های خود از پایگاه داده اوراکل استفاده میکنند
- طراحان و برنامه نویسان انبار داده (Data Warehouse) که برای پیاده سازی ETL های خود از پایگاه داده اوراکل استفاده میکنند
- کارفرمایانی که از پایگاه داده اوراکل استفاده کرده و نیاز به کسب دانش بیشتر جهت استخراج و کار با داده های جمع آوری شده توسط برنامه های کاربردی دارند

### اهداف دوره:

در انتهای این دوره دانشجویان قادر خواهند بود:

- رویه و تابع ایجاد کرده و روشهای مختلف استفاده از آنها را بشناسند
- انواع پکیج ها را طراحی و ایجاد کنند
- با انواع پکیج های از پیش ساخته اوراکل آشنا شده و از آنها در نوشتن برنامه های خود استفاده کنند
- از روشهای overload رویه ها و توابع برای انعطاف پذیری بیشتر برنامه های کاربردی استفاده کنند
- برای پوشش کسب و کارهای برنامه های کاربردی trigger تعریف کنند
- با روش های گوناگون پیاده سازی SQL های پویا آشنا شوند
- با کامپایلر PL/SQL آشنا شده و با تنظیم پارامترهای مربوطه کارایی آن را افزایش دهند
- وابستگی های موجود در پایگاه داده را شناخته و مدیریت کنند

### سرفصل دوره:

#### Creating Stored Procedures

- Describe PL/SQL blocks and subprograms
- Describe the uses and benefits of procedures
- Create, call, and remove procedures
- Use formal and actual parameters
- Identify the available parameter-passing modes
- Pass parameters using the positional, named, or combination techniques
- Handle exceptions in procedures
- View the procedure information

#### Creating Functions and Debugging Subprograms

- Creating Stored Functions
- The Difference Between Procedures and Functions
- Developing Functions
- Creating and Executing and Removing Functions
- Identifying the Advantages of Using Stored Functions in SQL Statements
- Using User-Defined Functions in SQL Statements

- □ Using a PL/SQL Function in the SQL WITH Clause
- □ Restrictions When Calling Functions from SQL statements

## Creating Packages

- □ Using PL/SQL Packages
- □ The Components of a PL/SQL Package
- □ The Visibility of a Package's Components
- □ Developing a PL/SQL Package
- □ Creating the Package Specification and Package Body
- □ Invoking the Package Constructs
- □ Creating and Using Bodiless Packages
- □ Removing a Package

## Working With Packages

- □ Overloading Subprograms
- □ Using Forward Declarations to Solve Illegal Procedure Reference
- □ Initializing Packages
- □ Using Package Functions in SQL and Restrictions
- □ Controlling Side Effects of PL/SQL Subprograms
- □ Persistent State of Packages
- □ Persistent State of Package Variables and Cursors
- □ Using PL/SQL Tables of Records in Packages

## Using Oracle-Supplied Packages in Application Development

- □ Using Oracle-Supplied Packages
- □ Examples of Some of the Oracle-Supplied Packages
- □ How Does the DBMS\_OUTPUT Package Work?
- □ Using the UTL\_FILE Package to Interact With Operating System Files
- □ Using the UTL\_MAIL Package

## Using Dynamic SQL

- □ The Execution Flow of SQL
- □ Working With Dynamic SQL
- □ When Do You Need Dynamic SQL?
- □ Using Native Dynamic SQL (NDS)
- □ Declaring Cursor Variables
- □ Executing a PL/SQL Block Dynamically
- □ Using Native Dynamic SQL to Compile PL/SQL Code

## Design Considerations for PL/SQL Code

- □ Standardize constants with a constant package
- □ Standardize exceptions with an exception package
- □ Write PL/SQL code that uses local subprograms
- □ Grant Roles to PL/SQL Packages and Standalone Stored Subprograms
- □ Use the NOCOPY compiler hint to pass parameters by reference
- □ Use the PARALLEL ENABLE hint for optimization
- □ Use the AUTONOMOUS TRANSACTION pragma to run independent transactions within a single transaction
- □ Describe the differences between invoker rights and definer rights

## Creating Compound, DDL, and Event Database Triggers

- □ Describe compound triggers
- □ Describe mutating tables
- □ Create triggers on DDL statements
- □ Create triggers on system events
- □ Display information about triggers

## Using PL/SQL compiler

- □ Using the PL/SQL Compiler Using the Initialization Parameters for PL/SQL Compilation
- □ Using the PL/SQL Compile Time Warnings
- □ Viewing the Current Setting of PLSQL\_WARNINGS
- Viewing the Compiler Warnings: Using SQL Developer, SQL\*Plus, or the Data Dictionary Views
- □ Guidelines for Using PLSQL\_WARNINGS

## Managing Dependencies

- □ Describe dependent and referenced objects
- □ Track procedural dependencies with dictionary views
- □ Predict the effect of changing a database object upon stored procedures and functions
- □ Manage local and remote procedural dependencies

منبع درسی:

اسلاید و منابع معرفی شده توسط شرکت اوراگل