
OracleAS Portal 10g: Build Portlets with PL/SQL

Duration 2 Days

What you will learn:

This class uses OracleAS 10g (Release 9.0.4.0.0). OracleAS Portal provides an environment to create enterprise portals. An enterprise portal is a corporate gateway to information access and management. In this course, students learn to create, deploy, and test portlets, the building blocks of a portal page. Students learn to build portlets running in an Oracle database by applying the OracleAS 10g Portal Developer Kit (PDK) PL/SQL application programming interface (API). Students also learn to make portlets customizable to enable end user personalization as well as how to secure portlets. Students learn to enhance portlet performance by implementing caching and they learn to build multi-lingual portlets.

Audience:

- PL/SQL Developer
- Portal Developer

Prerequisites:

Required Prerequisites:

- Knowledge of PL/SQL Web Toolkit
- Knowledge of HTML
- [Oracle9i: Program with PL/SQL](#)

Suggested Prerequisites:

- [OracleAS Portal 10g: Build Corporate Portals](#)

Course Objectives:

- Trap errors in the portlet code and display meaningful error messages
- Control access to portlets by using the OracleAS Security services
- Enhance portlet performance by using various types of portlet caching
- Build portlets that render their content in the user's preferred language
- Enable end user customization of portlets
- Identify the key features of OracleAS Portal 10g
- List the OracleAS Portal events and the provider methods that respond to the portal events

- Develop database providers and PL/SQL portlets by using the Portal Developer Kit
- Enable portlets to read from and pass parameters to other portlets
- Make portlets capable of keeping track of subsequent user requests (session handling)

Course Topics:

Organizing and Coding Database Providers

- Identifying OracleAS Portal events
- Registering and deregistering a provider
- Specifying portlet information
- Adding a portlet to a portal page
- Rendering a portlet
- Removing a portlet from a portal page
- Creating and managing database providers

Introduction To OracleAS Portal

- Enterprise portal solution
- What is OracleAS Portal?
- What is a portlet?
- What is a portlet provider?
- Challenges of the portlet developer
- When are programmatic portlets the best development choice?
- Provider development options
- What is OracleAS Portal Developer Kit?

Organizing and Coding PL/SQL Portlets

- Organizing portlets in PL/SQL packages
- Identifying the portlet show modes
- Coding portlet show modes
- Integrating page styles into portlets
- Creating dialog boxes
- Generating provider and portlet PL/SQL packages

Passing Parameters to Portlets

- Parameter types
- Passing and retrieving private portlet parameters
- Specifying multiple parameter values
- Retrieving all portlet parameters
- Passing and retrieving public portlet parameters

Accessing Portal Session Information

- Session storage
- Accessing session objects
- Loading session objects
- Saving and dropping session objects
- Session object security and visibility
- Session context service and API

Customizing Portlets

- Organizing portlet preferences
- Accessing preference storage
- Creating and deleting preference paths and preference names
- Maintaining preference values
- Implementing portlet customization
- Preference storage utilities

Handling Errors

- The error stack
- Adding error messages to the error stack
- Managing the error stack
- Displaying error messages in full screen or inline
- Raising exceptions
- Implementing warnings in PL/SQL portlets

Providing Portlet Security

- Oracle Single Sign-On authentication model
- Implementing lightweight users
- Providing secure access to OracleAS Portal PL/SQL resources
- Managing portal users and groups
- Portal privileges and Access Control List
- Managing application privileges
- Implementing security services in PL/SQL portlets

Writing Multilingual Portlets

- Storing multilingual messages in the NLS string storage
- Language and territory mappings
- Maintaining NLS strings
- Setting preferred language for a portal session
- Implementing NLS strings in portlet messages

Enhancing Portlet Performance Using Caching

-
- Caching goals
 - Caching PL/SQL portlets
 - Expiry, validation, and invalidation-based portlet caching concepts
 - Implementing expiry-based caching
 - Implementing validation-based caching
 - Implementing invalidation-based caching