S Oracle Data Integrator 12c: Advanced Integration and Development Ed 2

Oracle Database

DURATION	MODULES	COURSE CODE
3 Days	**12 Lectures**	**—**

Course Overview

This Oracle Data Integrator course covers advanced techniques for developing data integration applications. You learn how to create an integration environment based on the architecture of Extract Load Transform (EL-T), along with advanced methods and techniques.

What You Will Learn

Introduction

- Lesson Objectives
- Course Objectives
- Target Audience
- Class Introductions
- Agenda of Lessons
- Course Environment
- Course Materials
- Course Practices
- How Can I Learn More?
- Related Training
- Agenda
- Oracle Data Integrator: Features
- Oracle Data Integrator: Features Knowledge Modules
- Conventional Integration Process: ETL
- ETL Versus E-LT
- E-LT Steps
- ODI Architecture
- Summary
- Practice 1-1: Exploring Your Environment

ODI Knowledge Modules: Overview

- Objectives
- Agenda
- Knowledge Modules: Overview
- Global KMs
- KMs in Action
- Code Generation
- Types of Knowledge Modules: Overview
- Agenda
- How to Select Knowledge Modules
- Creating New Knowledge Modules
- Developing Knowledge Modules
- Editing Knowledge Modules
- Knowledge Module Options
- Description of Knowledge Module Tasks
- Details of the Tasks Are Generic
- Setting Knowledge Module Options
- Replacing Existing Knowledge Modules
- Agenda
- Developing Your Own Knowledge Module: General Guidelines
- Substitution API
- Using Substitution Methods
- Various ODI Substitution Methods
- Substitution Methods: Basic Examples
- Using Substitution Methods in Actions
- Working with Object Names
- Working with Lists of Tables, Columns, and Expressions
- Example: Using getTargetColList to Create a Table
- Example: Using getColList in an Insert Values Statement
- Example: Using getSrcTablesList
- Generating the Source Select Statement
- Generating the Source Select Statement: Example
- Working with Datasets
- Obtaining Other Information with the API
- Quiz
- Summary
- Practice 2-1

Developing Knowledge Modules

- Objectives
- Agenda
- Developing KMs: Targeting a Particular Stage of your Integration Process
- Developing KMs: What to Avoid and What to Use
- Developing KMs: Other Recommendations

- Agenda
- Using Java in Knowledge Modules
- Using Code Generation Tags:
- Using Code Generation Tags: Examples
- Code Generation Techniques
- Agenda
- Using ODI Substitution Methods: getJoin()
- Using ODI Substitution Methods: getFilter()
- Using ODI Substitution Methods: getPK()
- Using ODI Substitution Methods in Journalizing Knowledge Modules:
- Examples
- Using ODI Substitution Methods in Reverse-Engineering KM: getModel()
- Agenda
- Troubleshooting KMs
- Quiz
- Summary
- Practice 3-1 Overview: Creating a New Knowledge Module for Data
- Generation

Designing ODI Integration Mappings

- Objectives
- Agenda
- Integration Process: Overview
- Typical Integration Process
- Agenda
- Integration Mappings: Overview
- Basic Integration Process: Sequence of Operations
- Staging Area
- Placing the Staging Area
- Agenda
- Designing Integration Mappings: E-LT- and ETL-Style Mappings
- Designing an ETL-Style Mapping
- Designing an ETL-Style Mapping: Using Multi-Connection IKM
- Designing an ETL-Style Mapping: Using an LKM and a Mono-Connection IKM
- Using an LKM and a Mono-Connection IKM: Steps
- Agenda
- Enforcing ODI Data Quality
- Ways to Handle Erroneous Data
- Error Recycling
- Building a Data Quality Framework
- Quiz
- Summary
- Practice 4-1: Creating an ODI Mapping for XML to Database Transformation with
- ODI Constraint and Error Recycling

Designing Advanced Integration Mappings

- Objectives
- Agenda
- Creating Lookups
- Lookup Goal
- Lookup Match Row Rules: Multiple Match
- Lookup Match Row Rules: No-Match
- Using Set-Based Operators with Integration Mappings
- Example: Flow with Multiple Datasets
- Defining a Dataset
- Using Set-Based Operators with Integration Mappings
- Using Set-Based Operators: Guidelines
- Agenda
- Using Partitioning with ODI Mappings
- Partitioning: Definition in Datastore After Reverse Engineering
- Using Partitioning in a Mapping
- Agenda
- Configuring Reusable Mappings
- Using Reusable Mappings
- Derived Select for Reusable Mappings
- Derived Select for Reusable Mappings: Limitations
- Quiz
- Summary
- Practice 5-1: Configuring Partition Support
- Practice 5-2: Configuring Lookup Match Rules

Using Variables in ODI

- Objectives
- Agenda
- Using Variables in ODI: Overview
- Variable Scope
- Referring to a Variable
- Referring to a Variable: Using “:” Instead of “#”
- Agenda
- Using Variables in Packages
- Using Variables in Packages: Example
- Using Variables in Mappings
- Using Variables in Mappings: Examples
- Using Variables in Object Properties
- Using Variables in Procedures: with Jython
- Using Variables in Procedures: with Options
- Using Variables Within Variables
- Using Variables in the Resource Name of a Datastore
- Using Variables in the Resource Name of a Datastore: Steps

- Using a Variable as a Startup Parameter: Example
- Using ODI Variables in Topology
- Using ODI Variables in Topology: A Server URL
- Using Variables in a Server URL: Example
- Agenda
- Tracking Variables
- Tracking Variables: Notes
- Quiz
- Summary
- Practice 6-1 Overview: Using an ODI Variable as a Startup Parameter

Accelerating Development with Groovy

- Objectives
- Agenda
- ODI Software Development Kit: Overview
- SDK-Supported ODI Operations
- ODI Operations Not Supported by SDK
- Combining Different APIs
- Agenda
- Groovy: Introduction
- Groovy Editor: Introduction
- Executing Script with Groovy Editor
- Agenda
- Performing an SDK Task Using Java with Groovy Editor: Example
- Using Custom Libraries
- Defining Additional Groovy Execution Classpath
- Read Input with the odiInputStream Variable
- Automating Development Tasks: Example 1
- Automating Development Tasks: Example 2a (createModel)
- Automating Development Tasks: Example 2b (createLogicalSchema)
- Automating Development Tasks: Example 3a (Simple Mapping)
- Automating Development Tasks: Example 3b (Mapping and Filter)
- Automating Development Tasks: Example 4
- Quiz
- Summary
- Practice 7-1 Overview: Automating ODI Tasks with Groovy

Working with ASCII Files

- Objectives
- Agenda
- Types of Files in ODI
- Complex Files Concepts
- Sample Complex XML File
- Sample Complex JSON file
- Knowledge Modules for Complex Files

- Requirements for Working with Complex Files
- Agenda
- Setting Up the Topology
- JDBC URL Properties
- JDBC URL: Example
- Complex File Data Server Definition: Example
- Definition of Physical and Logical Schema: Example
- Agenda
- Setting Up an Integration Project and Creating a Complex File Model:
- Example
- Setting Up an Integration Project and Creating a Complex File Model: Reverse
- Engineer
- Designing a Mapping with Complex File Model
- Creating an nXSD File
- nXSD Schema: Example
- Getting to the Native Format Builder Wizard
- Using the Native Format Builder Wizard
- Reading the Source File
- Using the Native Format Builder Wizard
- Quiz
- Summary
- Practice 8-1 Overview: Configuring ODI Topology and ODI Model with Complex
- Files
- Practice 8-2: Configuring Mappings to Support JSON

Integration of ODI in Enterprise Environment and SOA

- Objectives
- Agenda
- Three Types of ODI Agents
- Java EE Agent Deployment Features
- Integration of ODI with Fusion Middleware Control
- Using the ODI Console
- Using the ODI Console: Example
- Agenda
- Types of Web Services
- Data Services: Overview
- Generation of Data Services
- Public Web Services: Overview
- Using Public Web Service OdiInvoke and OdiInvokeWebService
- Installing Public Web Services
- Agenda
- OdiInvokeWebService Tool
- OdiInvokeWebService Tool in Action
- OdiInvokeWebService Tool in a Package
- Agenda

- ODI with SOA Integration Scenarios
- Integration of ODI Within SOA in Action
- Example 1: Using Data Services
- Example 2: Exposing ODI Process as a Web Service
- Creating the BPEL Process in JDeveloper
- Creating ODI Mapping and ODI Package
- Creating ODI Scenario and Editing the BPEL Process
- Deploying the BPEL Process to WLS and Invoking from Enterprise Manager
- Example 3: Calling a Web Service from Within ODI for Processing ODI Errors with
- BPEL Human Workflow
- Processing ODI Errors with Error Hospital
- Processing ODI Errors with BPEL Human Workflow
- Processing ODI Errors with BPEL Human Workflow 1
- Processing ODI Errors with BPEL Human Workflow 2
- Processing ODI Errors with BPEL Human Workflow 3
- Processing ODI Errors with BPEL Human Workflow
- Quiz
- Summary
- Practice 9-1 Overview: Integrating ODI in the Enterprise Environment
- Practice 10-2 Overview: Exposing an ODI Scenario as a Web Service
- Practice 10-3 Overview: Integrating ODI with a BPEL Process Within SOA

Enhancing ODI Security

- Objectives
- Agenda
- Security Concepts: Overview
- Authorizing by Object Instance
- ODI Security Navigator: Overview
- Using Generic and Nongeneric Profiles
- Built-in Profiles
- SUPERVISOR: A Checkbox, Not a Profile
- Agenda
- Security Policy Approach
- Defining Security Policies: Using Generic Profiles
- Defining Security Policies: Using Nongeneric Profiles
- Using Nongeneric Profiles: Granting an Authorization by Object Instance
- Unused Authorizations
- Cleaning Up Unused Authorizations
- Agenda
- ODI Security Integration: Overview
- Implementing External Authentication (OPSS)
- Setting the Authentication Mode
- Configuring External Authentication
- Oracle Unified Directory (OUD)
- Agenda

- Configuring External Authentication with OUD
- 1. Creating a New LDAP User with OUD
- 2a. Editing jps-config-jse.xml to Point to OUD LDAP Server
- 2b. Editing jps-config-jse.xml to Point to OUD LDAP Protocols
- 3. Running odi_credtool to Set Up the Credentials for the Identity Store
- Creating New ODI Master Repository Referencing a User in the External LDAP
- Server
- Creating a New ODI Connection Referencing a User in the External LDAP
- Server
- 6a. Creating New ODI Master Repository Referencing a User in the External LDAP
- Server
- 6b. Creating New ODI Master Repository Referencing a User in the External LDAP
- Server
- Switching the Master Repository Authentication Mode
- Reactivating Users After Switching to Internal Authentication
- External Password Storage
- Setting Up External Password Storage
- Implementing External Password Storage
- Quiz
- Summary
- Practice 10-1 Overview: Implementing ODI External User Authentication

Choosing Integration Strategies: Best Practices

- Objectives
- Agenda
- Integration Strategies
- Agenda
- Strategies with Staging Area on the Target
- Strategies with Staging Area on the Target: Append
- Strategies with Staging Area on the Target: Control Append
- Strategies with Staging Area on the Target: Incremental Update
- Strategies with Staging Area on the Target: Incremental Update:
- Optimization
- Agenda
- Strategies with Staging Area Different from the Target
- Strategies with Staging Area Different from the Target: Server to Server
- Append
- Strategies with Staging Area Different from the Target: Server to File or JMS
- Append
- Agenda
- Using ODI for Bulk Processing
- Using ODI for Bulk Processing: Design Patterns
- Using Cross-Reference (XREF) Table: Overview
- Using the XREF Knowledge Module
- Agenda

- Loading Data Patterns
- Oracle GoldenGate: Overview
- Oracle GoldenGate Solutions: Overview
- Oracle GoldenGate Concepts
- Using ODI and OGG Together
- ODI and OGG Topology: Data Servers
- ODI and OGG Topology: Capture Physical and Logical
- ODI and OGG Topology: Delivery Physical and Logical
- ODI and OGG Topology: CDC
- Initializing CDC Process and Performing Change Operations with OGG
- OGG Command Line: GGSCI
- Agenda
- 1. Use Context-Independent Design
- 2. Use Procedures Only When Needed
- 3. Always Enforce Data Quality
- 4. Handle Error Cases in Packages
- 5. Choose Right Knowledge Module
- 6. Other Best Practices
- Quiz
- Summary
- Practice 11: Integrating ODI 12c and Oracle GoldenGate 12c

Cubes and Dimensions

- Objectives
- Scenario
- Agenda
- Overview of Dimensions
- Product Dimension Structure
- Overview of Hierarchies and Levels
- Dimension Functionality
- Slowly Changing Dimensions
- Example of a SCD
- Defining a Type 2 SCD
- Defining a Type 3 SCD
- Agenda
- Overview of a Cube
- Cube Implementation
- Agenda
- Creating Dimensions
- Dimension Editor: Levels Tab
- Dimension Editor: SCD Level Attributes
- Dimension Editor: Hierarchies Tab
- Loading Data into Dimensions
- Creating Dimensions: Dimension Pattern
- Instructor Demonstration
- Agenda

- Creating Cubes: Cube Editor
- Cube Editor: Details Tab
- Loading Data into Cubes: Smart Components
- Creating Cubes: Cube Pattern
- Instructor Demonstration
- Quiz
- Lesson Summary
- Practice 12-1: Creating Cubes and Dimensions
- Course Summary