



## **BUILD APPLICATION WITH ADF FRAMEWORK**

### **Course Overview**

#### **Topic**

#### **Introduction to Fusion and ADF**

- Describing Fusion architecture
- Explaining how ADF fits into the Fusion architecture
- Describing the ADF technology stack (MVC)

#### **Querying and Persisting Data**

- Using view objects
- Using entity objects to persist data
- Synchronizing entity objects with database table changes
- Creating associations
- Creating updateable view objects
- Creating master-detail relationships
- Refactoring

#### **Programmatically Customizing Data Services**

- Generating Java classes
- Programmatically modifying the behavior of entity objects
- Programmatically modifying the behavior of view objects
- Adding service methods to an application module
- Using client APIs

#### **Understanding UI Technologies**

- Describing the use of Web browsers and HTML
- Describing the function of Servlets and JSPs
- Defining JavaServer Faces
- Explaining the JSF component architecture and JSF component types
- Explaining the purpose of backing beans and managed beans
- Describing the JSF life cycle
- Explaining how ADF Faces augments the JSF life cycle

#### **Adding Functionality to Pages**

- Internationalizing the user interface
- Using component facets
- Displaying tabular data in tables
- Displaying hierarchical data in trees
- Displaying text or media with icons and images
- Defining search forms and display results
- Displaying data graphically

## **Ensuring Reusability**

- Designing for reuse
- Using task flow templates
- Creating and using page templates
- Creating and using declarative components
- Creating and using page fragments
- Deciding which type of reusable component to use

## **Implementing Transactional Capabilities**

- Handling transactions with ADF BC
- Using task flows to control transactions
- Sharing data controls
- Handling transaction exceptions
- Defining response to the Back button

## **Getting Started with JDeveloper**

- Listing JDeveloper benefits for application development
- Using the features of the JDeveloper IDE
- Defining IDE preferences
- Creating applications, projects, and connections in JDeveloper

## **Exposing Data**

- Creating application modules
- Using master-detail view objects in application modules
- Managing Business Components transactions
- Abstracting business services with ADF Model

## **Validating User Input**

- Understanding validation options: Database, Data Model, or UI
- Triggering validation execution
- Handling validation errors
- Using Groovy expressions in validation

- Using programmatic validation

## **Binding UI Components to Data**

- Creating a JSF page
- Adding UI components to a page
- Describing the ADF Model layer
- Using a Page Definition file
- Examining data binding objects and metadata files
- Binding existing components to data
- Running and testing the page

## **Implementing Navigation on Pages**

- Using ADF Faces navigation components
- Using buttons and links
- Using menus for navigation
- Using breadcrumbs
- Using a train component

## **Passing Values Between UI Elements**

- Defining the data model to reduce the need to pass values
- Using a managed bean to hold values
- Using page parameters
- Using task flow parameters
- Passing values from containing pages to regions

## **Implementing Security in ADF BC Applications**

- Exploring ADF Application security options
- Understanding ADF security framework
- Enabling users to access resources
- Implementing a Login page
- Understanding ADF controller authorization
- Using Expression Language to extend security capabilities

## **Building a Data Model with ADF Business Components**

- Introducing ADF Business Components
- Creating Business Components from tables
- Testing the data model

## **Declaratively Customizing Data Services**

- Internationalizing the data model
- Editing business components
- Modifying default behavior of entity objects
- Changing the locking behavior of an application module

## **Troubleshooting ADF BC Applications**

- Troubleshooting the business service
- Troubleshooting the UI
- Using logging and diagnostics
- Using the JDeveloper debugger

## **Planning the User Interface**

- Describing the Model-View-Controller design pattern
- Differentiating between bounded and unbounded task flows
- Creating and converting task flows
- Defining control flows
- Defining global navigation
- Creating routers for conditional navigation
- Calling methods and other task flows
- Implementing validation in the user interface

## **Achieving the Required Layout**

- Using complex layout components
- Explaining how to use ADF Faces skins
- Using dynamic page layout

## **Responding to Application Events**

- Using managed beans
- Coordinating JSF and ADF lifecycles
- Using phase and event listeners
- Using action listeners and methods
- Understanding additional AJAX events