

S4D437: Transactional Apps with the ABAP RESTful Programming Model

Course Description

This course provides an in-depth understanding of developing transactional applications using the ABAP RESTful Application Programming (RAP) Model. Participants will learn to define, create, and manage RAP Business Objects, enable draft capabilities, and implement unmanaged and composite business objects for SAP Fiori-based applications. The course emphasizes hands-on development, ensuring learners gain practical experience with RAP concepts, tools, and best practices.

Audience Profile

This course is designed for:

- **Development Consultants**
 - **Developers**
-

Prerequisites

Participants should have:

- Basic knowledge of ABAP programming.
 - Familiarity with SAP Fiori applications and CDS (Core Data Services).
 - Understanding of OData services and basic business process workflows is recommended.
-

Course Objectives

By the end of this course, participants will be able to:

1. Understand the architecture and concepts of the ABAP RESTful Programming Model (RAP).
2. Define and implement RAP Business Objects, including managed, unmanaged, and composite types.
3. Develop and enhance transactional applications with input validation, dynamic feature control, and authority checks.
4. Enable and implement draft handling for stateless applications.
5. Integrate RAP Business Objects with OData services and SAP Fiori elements.

6. Apply best practices for concurrency control, actions, and messaging in RAP.
-

Table of Contents

Unit 1: The ABAP RESTful Programming Model (RAP)

- Lesson 1: Understanding the Concept and Architecture of RAP
 - RAP concept, architecture, and use cases
 - Overview of ABAP development tools
- Lesson 2: Defining an OData UI Service
 - Creating CDS projection views with UI metadata
 - Previewing and using OData UI services

Unit 2: RAP Business Objects (RAP BOs)

- Lesson 1: Defining RAP Business Objects and Their Behavior
 - Creating CDS behavior definitions and projections
- Lesson 2: Using Entity Manipulation Language (EML) to Access RAP Business Objects
 - Purpose, syntax, and usage of EML
 - Working with derived data types
- Lesson 3: Understanding Concurrency Control in RAP
 - Pessimistic and optimistic concurrency control
- Lesson 4: Defining Actions and Messages
 - Implementing actions and exposing them in OData services
 - Defining exception classes and accessing application data
- Lesson 5: Implementing Authority Checks
 - Restricting read access with controls and authority checks

Unit 3: Update and Create in Managed Transactional Apps

- Lesson 1: Enabling Input Fields and Value Help
 - Setting input fields as read-only or mandatory
 - Defining value help
- Lesson 2: Implementing Input Checks with Validations
 - Explaining and implementing validations

- Linking messages to input fields
- Lesson 3: Providing Values with Determinations
 - Implementing determinations and numbering concepts
- Lesson 4: Implementing Dynamic Feature Control
 - Explaining and applying dynamic field, action, and operation control

Unit 4: Draft-Enabled Transactional Apps

- Lesson 1: Understanding the Draft Concept
 - Draft handling for stateless applications
- Lesson 2: Developing Draft-Enabled Applications
 - Transition and state messages
 - Draft behavior implementation

Unit 5: Transactional Apps with Composite Business Object

- Lesson 1: Defining Composite RAP Business Objects
 - Creating compositions in RAP BOs
- Lesson 2: Defining Compositions in OData UI Services
 - Exposing compositions and enabling navigation in Fiori apps
- Lesson 3: Implementing the Behavior for Composite RAP BOs
 - Accessing composite objects using EML

Unit 6: Transactional Apps with Unmanaged Business Object

- Lesson 1: Understanding Data Access in Unmanaged Implementations
 - Defining behavior for unmanaged BOs
- Lesson 2: Implementing Unmanaged Business Objects
 - Implementing data access for unmanaged BOs