TAW10

ABAP Workbench Fundamentals Part I

PARTICIPANT HANDBOOK
INSTRUCTOR-LED TRAINING

Course Version: 16
Course Duration: 5 Day(s)
Material Number: 50136303

Contents

ix Course Overview

1 Unit 1: SAP Systems

2 Lesson: Explaining the Key Capabilities of SAP NetWeaver

11 Lesson: Explaining the Architecture of an SAP System, Application Server (AS) ABAP and Application Server (AS) Java

21 Unit 2: SAP Portfolio Overview

22 Lesson: Identifying the Components of the SAP Applications Portfolio

47 Unit 3: Navigation Basics

48 Lesson: Navigating with SAP Graphical User Interface (GUI) for Windows in AS ABAP Systems

57 Exercise 1: Log On to the System and Run a Transaction

63 Exercise 2: Search Technical Information Using F1

73 Unit 4: SAP User Interfaces

74 Lesson: Accessing AS ABAP Systems with SAP GUI

81 Exercise 3: Search for Transactions in AS ABAP

89 Unit 5: System Core

90 Lesson: Explaining AS ABAP and AS Java System Architecture

101 Lesson: Identifying the Processes of an AS ABAP System

105 Exercise 4: Determine the Processes of SAP NetWeaver AS ABAP

108 Lesson: Processing User Requests in AS ABAP

115 Exercise 5: Monitor the Work Processes in AS ABAP

123 Exercise 6: Monitor Lock Entries

131 Exercise 7: Monitor Updates

135 Lesson: Using Additional Processes of an AS ABAP System

139 Exercise 8: Print an Output in AS ABAP

143 Exercise 9: Schedule a Simple Report in the Background

© Copyright. All rights reserved.

153 Unit 6: Communication and Integration Technologies

154 Lesson: Describing Communication with Remote Function Call-Based (RFC) Technology
161 Exercise 10: Execute a BAPI
167 Lesson: Describing the Basics of Web Services
171 Lesson: Describing OData in SAP Gateway

181 Unit 7: Flow of an ABAP Program

182 Lesson: Describing the Processing of ABAP Programs

199 Unit 8: ABAP Workbench

200 Lesson: Introducing the ABAP Workbench
210 Lesson: Organizing ABAP Development Projects
217 Exercise 11: Create a Package
220 Lesson: Developing ABAP Programs
231 Exercise 12: Develop a Simple ABAP Program
235 Lesson: Finalizing ABAP Development Projects
239 Exercise 13: Create a Transaction

251 Unit 9: Basic ABAP Language Elements

252 Lesson: Defining Elementary Data Objects
264 Lesson: Using Basic ABAP Statements
273 Exercise 14: Use Basic ABAP Statements
280 Lesson: Analyzing Programs with the ABAP Debugger
289 Exercise 15: Debug Statements on Elementary Data Objects

© Copyright. All rights reserved.

299 Unit 10: Modularization Techniques in ABAP

301 Lesson: Explaining Modularization
308 Lesson: Defining and Calling Subroutines
323 Exercise 16: Implement Subroutines
329 Lesson: Calling Function Modules
339 Exercise 17: Call a Function Module
346 Lesson: Creating Function Modules
349 Exercise 18: Create a Function Group
353 Exercise 19: Create a Function Module
360 Lesson: Describing Business Application Programming Interfaces (BAPIs)
366 Lesson: Calling Methods of Global Classes
379 Exercise 20: Use a Global Static Method
386 Lesson: Creating Global Classes and Static Methods
391 Exercise 21: Create a Simple Global Class
393 Exercise 22: Create a Simple Static Method
398 Lesson: Using Local Classes

411 Unit 11: Complex Data Objects

412 Lesson: Using Structured Data Objects
419 Exercise 23: Use Structured Data Objects
425 Lesson: Using Internal Tables
445 Exercise 24: Use Internal Tables

455 Unit 12: Data Modeling and Data Retrieval

457 Lesson: Explaining Data Models
467 Exercise 25: Analyze Transparent Tables in the Data Dictionary
473 Lesson: Retrieving Single Database Records
481 Exercise 26: Implement Single Record Access
487 Lesson: Retrieving Multiple Database Records
489 Exercise 27: Implement a SELECT Loop
497 Exercise 28: Implement an Array Fetch (Optional Exercise)
503 Lesson: Describing Other Aspects of Database Access
514 Lesson: Implementing Authorization Checks
521 Exercise 29: Implement an Authorization Check

533 Unit 13: Classic ABAP Reports

534 Lesson: Implementing ABAP Lists
539 Lesson: Implementing Selection Screens
547 Exercise 30: Create a Selection Screen and a Classic ABAP List
554 Lesson: Implementing Events of ABAP Reports
561 Exercise 31: Implement Events of ABAP Reports

© Copyright. All rights reserved.

571 Unit 14: **Program Analysis Tools**

572 Lesson: Improving the Quality of ABAP Code with the Code Inspector

581 Unit 15: **Program Calls and Memory Management**

582 Lesson: Calling Programs Synchronously

591 Exercise 32: Submit Report

598 Lesson: Describing the ABAP Runtime and Memory Management

623 Exercise 33: Call Transaction and Use SAP Memory

639 Unit 16: **ABAP Open SQL**

640 Lesson: Processing and Aggregating Datasets on the Database

645 Exercise 34: Request Ordered and Condensed Datasets from the Database

657 Exercise 35: Perform Calculations on the Database

662 Lesson: Implementing Complex WHERE Conditions and Special INTO Clauses

670 Lesson: Selecting Data from Multiple Database Tables

677 Exercise 36: Implement a Join of Multiple Database Tables

© Copyright. All rights reserved.

TAW10

ABAP Workbench Fundamentals Part II

PARTICIPANT HANDBOOK
INSTRUCTOR-LED TRAINING

Course Version: 16
Course Duration: 5 Day(s)
Material Number: 50136303

Contents

ix Course Overview

1 Unit 1: Introduction to the ABAP Dictionary

2 Lesson: Describing the ABAP Dictionary

11 Unit 2: Data Types in the ABAP Dictionary

12 Lesson: Creating Domains and Data Elements
19 Exercise 1: Create Basic Data Types
26 Lesson: Creating Flat Structures
31 Exercise 2: Create Simple and Nested Structures
37 Lesson: Creating Table Types and Deep Structures
41 Exercise 3: Create Table Types
47 Exercise 4: Create Deep Structures
52 Lesson: Creating Type Groups

61 Unit 3: Database Tables

62 Lesson: Creating Transparent Tables
75 Exercise 5: Create Tables in the ABAP Dictionary
85 Lesson: Defining Cluster Tables and Pooled Tables

97 Unit 4: Performance During Table Access

98 Lesson: Creating Database Table Indexes
103 Exercise 6: Create Table Indexes
107 Lesson: Setting Up Table Buffering
121 Exercise 7: Set Up Table Buffering

133 Unit 5: Input Checks

134 Lesson: Creating Fixed Values
137 Exercise 8: Create Fixed Values
140 Lesson: Defining Foreign Keys to Perform Input Checks
149 Exercise 9: Perform Input Checks to Ensure Data Consistency
160 Lesson: Creating Text Tables
163 Exercise 10: Create a Text Table

© Copyright. All rights reserved.

173 Unit 6: Dictionary Object Dependencies

174 Lesson: Differentiating Between Active and Inactive Dictionary Objects

179 Exercise 11: Differentiate Between Active and Inactive Dictionary Objects

183 Lesson: Identifying Dependencies with ABAP Dictionary Objects

187 Exercise 12: Identify Dependencies with ABAP Dictionary Objects

195 Unit 7: Table Changes

196 Lesson: Performing a Table Conversion

205 Exercise 13: Perform a Table Conversion

210 Lesson: Enhancing Tables Using Append Structures

217 Exercise 14: Add Enhancements to a Table Using an Append Structure

225 Unit 8: Views and Maintenance Views

226 Lesson: Creating Database Views

239 Exercise 15: Create a Database View

244 Lesson: Creating Maintenance Views

253 Exercise 16: Create a Maintenance View

258 Lesson: Creating View Clusters

261 Exercise 17: Create a View Cluster

273 Unit 9: Search Helps

274 Lesson: Creating Search Helps

291 Exercise 18: Create Search Helps

295 Lesson: Applying Advanced Search Help Techniques

299 Exercise 19: Create Collective Search Helps

303 Exercise 20: Create Append Search Helps

313 Unit 10: Selection Screens

314 Lesson: Implementing a Selection Screen

327 Exercise 21: Implement a Selection Screen

334 Lesson: Implementing Multiple Selection Screens

341 Exercise 22: Define Tabstrips on a Selection Screen

347 Lesson: Implementing Input Checks and Creating Variants

353 Exercise 23: Implement Input Checks and Create Variants

© Copyright. All rights reserved.

361 Unit 11: **Introduction to Screen Programming**

363 Lesson: Explaining the User Dialog Programming Model
371 Exercise 24: Create a Dialog Program
375 Lesson: Introducing Screen Programming
381 Lesson: Creating Screens and Screen Elements
389 Exercise 25: Create a Screen
399 Lesson: Modifying Screens at Runtime
405 Lesson: Designing Screen Sequence
411 Lesson: Calling a Dialog Box Dynamically
413 Exercise 26: Call a Dialog Box

425 Unit 12: **The Program Interface**

426 Lesson: Explaining User Interfaces
432 Lesson: Setting a GUI Title and a GUI Status
441 Exercise 27: Create a GUI Status

453 Unit 13: **Simple Screen Elements**

454 Lesson: Creating Screen Elements for Output
462 Lesson: Creating Input/Output Fields
468 Lesson: Defining Checkboxes and Radio Button Groups
473 Exercise 28: Create a Radio Button Group
479 Lesson: Creating Pushbuttons

489 Unit 14: **Screen Error Handling**

490 Lesson: Handling Errors Using Dialog Messages and Field Input Checks
503 Lesson: Handling Errors Using Navigation and Input Help
509 Exercise 29: Check Input Values

523 Unit 15: **Subscreens**

524 Lesson: Creating Subscreens
535 Exercise 30: Embed Subscreens

549 Unit 16: **Tabstrip Controls**

550 Lesson: Creating Tabstrip Controls
557 Lesson: Programming Tabstrip Controls
563 Exercise 31: Create Tabstrip Controls

© Copyright. All rights reserved.

TAW12

ABAP Workbench Concepts part I

PARTICIPANT HANDBOOK
INSTRUCTOR-LED TRAINING

Course Version: 16
Course Duration: 5 Day(s)
Material Number: 50136093

Contents

ix Course Overview

1 Unit 1: Introduction to Object-Oriented Programming

2 Lesson: Explaining the Object-Oriented Programming Model
14 Lesson: Analyzing and Designing with Unified Modeling Language (UML)
27 Exercise 1: Create UML Diagrams

37 Unit 2: Fundamental Object-Oriented Syntax

39 Lesson: Creating Local Classes
53 Exercise 2: Create Local Classes
60 Lesson: Creating Objects
67 Exercise 3: Create Objects
71 Lesson: Accessing Methods and Attributes
79 Exercise 4: Call Methods
85 Lesson: Implementing Constructors in Local Classes
89 Exercise 5: Create and Use Constructors
97 Lesson: Implementing Class Constructors in Local Classes
101 Exercise 6: Create and Use Static Constructors

117 Unit 3: Inheritance and Casting

118 Lesson: Implementing Inheritance
129 Exercise 7: Implement Inheritance
140 Lesson: Implementing Upcasts Using Inheritance
143 Exercise 8: Implement Upcasts
148 Lesson: Implementing Polymorphism Using Inheritance
153 Exercise 9: Implement Polymorphism Using Inheritance
160 Lesson: Implementing Downcasts Using Inheritance
163 Exercise 10: Implement Downcasts

177 Unit 4: Interfaces and Casting

178 Lesson: Defining and Implementing Local Interfaces
185 Exercise 11: Define and Implement a Local Interface
195 Lesson: Implementing Polymorphism Using Interfaces
199 Exercise 12: Implement Polymorphism Using Interfaces
206 Lesson: Integrating Class Models Using Interfaces
209 Exercise 13: Integrate Class Models Using Interfaces

© Copyright. All rights reserved.

229 Unit 5: **Object-Oriented Events**

230 Lesson: Implementing Events in Local Classes
239 Exercise 14: Implement Events in Local Classes
252 Lesson: Implementing Events in Local Interfaces
253 Exercise 15: Implement Handling of Interface Events

267 Unit 6: **Object-Oriented Repository Objects**

268 Lesson: Creating Global Classes
279 Exercise 16: Implement a Global Class
289 Lesson: Defining and Implementing Global Interfaces
295 Exercise 17: Import and Implement a Global Interface
306 Lesson: Implementing Inheritance in Global Classes

315 Unit 7: **ABAP Object-Oriented Examples**

316 Lesson: Using the ABAP List Viewer (ALV)
323 Exercise 18: Implement the ALV Grid Control
331 Exercise 19: Implement a Dialog Box with the ALV Grid Control
340 Lesson: Describing Business Add-Ins (BAdIs)

351 Unit 8: **Class-Based Exceptions**

352 Lesson: Explaining Class-Based Exceptions
359 Lesson: Defining and Raising Exceptions
369 Exercise 20: Implement Class-Based Exceptions
384 Lesson: Implementing Advanced Exception Handling Techniques
393 Exercise 21: Map Exceptions to Each Other

411 Unit 9: **Object-Oriented Design Patterns**

412 Lesson: Implementing Advanced Object-Oriented Techniques
423 Lesson: Implementing the Singleton Pattern
427 Exercise 22: Implement the Singleton Pattern
436 Lesson: Implementing Factory Classes Using Friendship
439 Exercise 23: Implement a Factory Class Using Friendship

455 Unit 10: **Program Calls and Memory Management**

456 Lesson: Using Shared Objects
479 Exercise 24: Use Shared Objects

© Copyright. All rights reserved.

493 Unit 11: Dynamic Programming

494 Lesson: Using Generic Data Types

501 Exercise 25: Use Dynamically Typed Field Symbols in Dynamic SQL Statements

511 Exercise 26: Access Structure Components Dynamically

522 Lesson: Describing Data Types, Data Objects, and Objects at Runtime

535 Exercise 27: Provide Column Headers Using RTTI

547 Lesson: Creating Data Types, Data Objects, and Objects at Runtime

555 Exercise 28: Create Data Objects at Runtime

565 Exercise 29: Optional: Create Data Types at Runtime with RTTC

© Copyright. All rights reserved.

TAW12

ABAP Workbench Concepts part II

PARTICIPANT HANDBOOK
INSTRUCTOR-LED TRAINING

Course Version: 16
Course Duration: 5 Day(s)
Material Number: 50136093

Contents

xi Course Overview

1 Unit 1: Adjustment of SAP Standard Software

2 Lesson: Adjusting SAP Standard Software
8 Lesson: Describing Classic Enhancement Types

23 Unit 2: Enhancement of Dictionary Elements

24 Lesson: Enhancing Tables Using Append Structures and Customizing Includes
29 Exercise 1: Enhance a Table Using an Append Structure
34 Lesson: Enhancing Texts of Data Elements

45 Unit 3: Customer Exits

46 Lesson: Describing Enhancement Projects
56 Lesson: Enhancing Programs Using Program Exits
69 Exercise 2: Implement a Customer Exit – Program Exit
74 Lesson: Enhancing Menus Using Menu Exits
79 Exercise 3: Implement a Customer Exit – Menu Exit
84 Lesson: Enhancing Screens Using Screen Exits
93 Exercise 4: Implement a Customer Exit – Screen Exit

103 Unit 4: Classic Business Add-Ins

104 Lesson: Describing Classic Business Add-Ins (BAdIs)
110 Lesson: Enhancing Programs Using Classic BAdIs
119 Exercise 5: Implement a BAdI Program Exit
123 Exercise 6: Implement a BAdI Menu Exit
128 Lesson: Explaining Additional Details of Business Add-Ins (BAdIs)

135 Unit 5: New Business Add-Ins

136 Lesson: Describing the Change from Classic BAdIs to New BAdIs
140 Lesson: Using New Business Add-In (BAdI) Program Exits
147 Exercise 7: Implement New BAdI Program Exits

153 Unit 6: Explicit Enhancement Options

154 Lesson: Describing the Enhancement Framework
160 Lesson: Using Explicit Enhancements
169 Exercise 8: Use Explicit Enhancements

© Copyright. All rights reserved.

179 Unit 7: Implicit Enhancements

180 Lesson: Explaining Implicit Enhancements
183 Lesson: Implementing Implicit Enhancements in Function Modules
185 Exercise 9: Use Implicit Enhancements of an SAP Function Module
191 Lesson: Implementing Implicit Enhancements in Global SAP Classes
195 Exercise 10: Enhance a Global SAP Class
201 Lesson: Using Other Implicit Enhancements
205 Exercise 11: Enhance SAP Structures and SAP Subroutines

215 Unit 8: Modifications of the SAP Standard Application

216 Lesson: Modifying SAP Standard Software
227 Lesson: Modifying Applications using the Modification Assistant
237 Exercise 12: Implement Modifications
244 Lesson: Implementing User Exits
249 Lesson: Adjusting Modifications
255 Exercise 13: Perform Modification Adjustment with Transaction SPDD
259 Exercise 14: Perform Modification Adjustment with Transaction SPAU

267 Unit 9: Introduction to Web Dynpro

268 Lesson: Outlining the Benefits of Web Dynpro
273 Lesson: Displaying Web Dynpro Component Architecture
279 Exercise 15: Create a Simple Web Dynpro
285 Lesson: Navigating Between Views
287 Exercise 16: Set Up Navigation Between Views
295 Lesson: Creating View Assemblies
299 Exercise 17: Create View Assemblies
305 Lesson: Identifying Web Dynpro Entities and Relationships

317 Unit 10: Web Dynpro Controllers

318 Lesson: Outlining Controller Types and Entities

329 Unit 11: Web Dynpro Context

330 Lesson: Creating Context Nodes
338 Lesson: Setting Context Node Properties
345 Lesson: Sharing Data Between Controllers with Context Mapping
349 Exercise 18: Define, Map, and Bind Data to the Context

© Copyright. All rights reserved.

357 Unit 12: Web Dynpro User Interface

358 Lesson: Defining the View Layout
375 Exercise 19: Arrange UI Elements with Layout Managers
381 Lesson: Controlling UI Element Behavior with Data Binding
389 Lesson: Using Composite UI Elements
401 Exercise 20: Display Tables with the Service Wizard

415 Unit 13: Controller and Context Programming

416 Lesson: Outlining Controller Methods and Attributes
428 Lesson: Accessing the Context at Runtime with Controller Methods
439 Exercise 21: Access the Context at Runtime
445 Lesson: Adding New Elements to a Context Node
453 Exercise 22: Display Mass Data Using Tables
461 Lesson: Implementing Supply Functions
465 Exercise 23: Implement Supply Functions

© Copyright. All rights reserved.