+33 (0)1 48 10 09 46
formation@ambient-it.net
140 Avenue Jean Lolive 93500 Pantin - France

Updated on 22/07/2025

Sign up

Progress ABL Web/API training

2 days (14 hours)

Presentation

Master the design and exposure of high-performance Web APIs with Progress ABL thanks to this comprehensive, structured and practice-oriented training course. Designed for developers, architects and integrators, it will enable you to create, secure and deploy robust REST services in the OpenEdge/PASOE environment.

You'll start with a clear introduction to the Progress OpenEdge ecosystem and the logic of the ABL language, exploring its procedural and object-oriented paradigms, syntax, buffer and transaction management, as well as best practices for structuring projects.

The course will then take you through the creation of REST APIs with WebHandler, handling HTTP requests, JSON serialization, connecting to Progress databases, and securing access via authentication or OAuth 2.0.

You'll also learn how to consume external Web services from ABL (REST or SOAP), handle HTTP calls, process responses, and integrate these exchanges into your application workflows.

Like all our training courses, this one is based on the latest stable version of Progress ABL.

Objectives

- Understand the architecture of the Progress OpenEdge environment, its relational data model and the fundamentals of the ABL language.
- Know how to design, develop and structure procedural ABL programs, applying the best best practices in terms of readability, modularity and maintenance
- master data access and manipulation via ABL instructions (FIND, FOR EACH, CREATE...), buffers, transactions and temp-tables
- Be able to create functional user interfaces with AppBuilder, integrating graphical components and event-driven logic

- Apply best practices in error handling, debugging, code organization and performance optimization in a professional application context.

Target audience

- Developers
- Analysts

Prerequisites

- Knowledge of relational databases

Progress ABL Intro training program

Introduction to Progress ABL and the OpenEdge environment

- General architecture
- Basic syntax
- Supported paradigms: procedural, event-driven, object-oriented
- OpenEdge Studio vs OpenEdge Developer Kit
- Configuring the development database
- Directories, files, conventions
- .p, .cls, .w files and their roles

ABL language basics

- Variable declarations
- IF, CASE, DO, REPEAT, FOR EACH
- Table access via FOR EACH, FIND, BUFFER
- Transactions : DO TRANSACTION, UNDO, RETRY
- Internal vs. external procedures
- Defining and calling functions
- Class declaration
- Inheritance, interfaces and polymorphism
- CATCH/THROW vs UNDO/RETURN
- Logging, debugging, stack trace

ABL and the Web: modern architectures

- REST, SOAP, JSON services
- Client/server vs. n-tier
- Key features of OpenEdge AppServer / PASOE
- Differences between Classic AppServer and PASOE
- Introduction to WebHandler and WebSpeed

- Use cases and limitations
- Modern alternatives

Creating REST APIs with ABL and PASOE

- Handler structure: GET/POST/PUT/DELETE methods
- Routing with annotations or configuration
- Access to headers, query params, body JSON
- Building HTTP responses
- Interfacing with Progress tables
- JSON serialization of results
- Basic authentication and OAuth 2.0
- JWT token verification

Consumption of external Web services

- Creating and sending HTTP requests
- Response management, JSON/XML parsing
- Using Web Services Proxy Generator
- Mapping XML types to ABL
- Handling network errors
- Logs, HTTP codes, error parsing

ABL Web/API application security

- PASOE sessions and login
- OAuth 2.0 with external identity providers
- TLS/SSL configuration for PASOE
- Use of self-signed or CA certificates
- User/permission mapping
- Restricting access to API resources

Deployment and monitoring

- Assembling .r and rest/webapps files
- Deployment in PASOE via AdminServer
- Application and server logging
- OpenEdge profiler and monitoring tools
- ABL query optimization
- PASOE threading and configuration
- Build scripting with PCT (Progress Compilation Tools)
- Automation with Jenkins, Git, Artifactory

Companies involved

This training course is aimed at both individuals and companies, large or small, wishing to train their teams in a new advanced computer technology, or to acquire specific business knowledge or modern methods.

acquire specific business knowledge or modern methods.

Positioning on entry to training

Positioning at the start of training complies with Qualiopi quality criteria. As soon as registration is finalized, the learner receives a self-assessment questionnaire which enables us to assess his or her estimated level of proficiency in different types of technology, as well as his or her expectations and personal objectives for the forthcoming course, within the limits imposed by the selected format. This questionnaire also enables us to anticipate any connection or security difficulties within the company (intra-company or virtual classroom) which could be problematic for the follow-up and smooth running of the training session.

Teaching methods

Practical training: 60% hands-on, 40% theory. Training material distributed in digital format to all participants.

Organization

The course alternates theoretical input from the trainer, supported by examples, with brainstorming sessions and group work.

Validation

At the end of the session, a multiple-choice questionnaire is used to check that skills have been correctly acquired.

Certification

A certificate will be awarded to each trainee who completes the training course.