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

Updated on 09/23/2026

Sign Up

Jakarta EE Training: Developing Enterprise Applications

(formerly Java EE)

5 days (35 hours)

Overview

Jakarta EE is the leading platform for developing robust, secure, and maintainable Java enterprise applications. As the successor to Java EE, it provides a set of standardized specifications for designing web services and managing data, security, transactions, and asynchronous processing.

This Jakarta EE training course will enable you to design and develop enterprise applications by leveraging the platform's key APIs. You will learn how to structure your services using CDI, develop APIs with Jakarta REST, and efficiently manage JSON data exchanges.

You will be able to implement a persistence layer with Jakarta Persistence, ensure transaction consistency with Jakarta Transactions, and integrate asynchronous processing with Jakarta Messaging. You will also learn how to validate data and secure your applications' resources and endpoints.

The course also covers testing, packaging, containerization, and deployment of Jakarta EE applications. Through hands-on workshops, you will progressively develop a structured, testable enterprise application tailored to the constraints of modern environments.

Finally, you will learn how to migrate a Java EE application to Jakarta EE, manage the transition from the javax.* namespace to the jakarta.* namespace, and identify legacy dependencies or components that need to be modernized without requiring a complete rewrite of the application.

By the end of this course, you will be able to design, develop, secure, test, and deploy applications with Jakarta EE 11, as well as develop a

modernization strategy for your existing Java EE applications.

Like all our training courses, this one will introduce you to **the latest stable version** of the technology and its new features.

Objectives

- Understand the architecture and key specifications of Jakarta EE 11
- Structure web components and business services using Jakarta Servlet and CDI
- Develop APIs with Jakarta REST and manage JSON exchanges
- Implement persistence, transactions, and asynchronous messaging
- Validate data and secure Jakarta EE applications
- Test, package, containerize, and deploy enterprise applications
- Migrate a Java EE application to Jakarta EE and manage the transition from javax.* to jakarta.*

Target Audience

- Java developers
- Backend developers
- Software engineers
- Software and technical architects
- Developers responsible for maintaining or modernizing existing Java EE applications

Prerequisites

- Proficiency in Java programming and object-oriented programming
- Knowledge of collections, exceptions, annotations, and major Java APIs
- Basic knowledge of HTTP and how web applications work
- Basic understanding of SQL and relational databases is recommended

Curriculum for our Jakarta EE training course: Developing Enterprise Applications

[Day 1 - Morning]

Introduction to Jakarta EE 11 and its architecture

- Understanding the evolution from Java EE to Jakarta EE and the role of the Eclipse Foundation
- Explore the Jakarta EE 11 architecture and its key specifications
- Understand Jakarta EE-compatible profiles, components, containers, and runtimes

- Structuring an enterprise application into layers: presentation, business services, and data access
- Set up the development environment, build, packaging, and deployment
- Hands-on workshop: Create, package, and deploy your first Jakarta EE application.

[Day 1 - Afternoon]

Developing Web Components and Business Services with Servlets and CDI

- Understanding how the web container works with Jakarta Servlet
- Handling requests, responses, parameters, headers, sessions, filters, and listeners
- Use Jakarta Contexts and Dependency Injection (CDI) to structure business services
- Master scopes, qualifiers, producers, and interceptors
- Decoupling components to improve modularity, maintainability, and testability
- Hands-on workshop: Developing a web layer connected to modular business services using Servlets and CDI.

[Day 2 - Morning]

Create REST APIs and manage JSON exchanges

- Designing web services with Jakarta REST
- Define resources, URIs, HTTP methods, parameters, and response codes
- Generate and consume representations with Jakarta JSON Processing
- Serialize and deserialize Java objects with Jakarta JSON Binding
- Structure DTOs, responses, and exception-handling mechanisms for an API
- Hands-on workshop: Develop a complete REST API that exchanges JSON data with a client application.

[Day 2 - Afternoon]

Validating data and handling application errors

- Understanding how Jakarta Validation works
- Applying validation constraints to business objects, entities, and DTOs
- Create custom constraints and validators tailored to business rules
- Organize validation groups according to different use cases
- Centralize error handling and generate consistent, actionable API responses
- Hands-on workshop: Implementing a validation and error-handling strategy on a Jakarta REST API.

[Day 3 - Morning]

Manage data persistence with Jakarta Persistence

- Understanding the Jakarta Persistence Architecture and the Role of the EntityManager
- Creating and mapping persistent entities with annotations
- Configuring relationships between entities and managing their lifecycle
- Using JPQL and query mechanisms to access data
- Master loading strategies and identify key performance issues
- Hands-on workshop: Build a persistence layer and connect a business API to a relational database.

[Day 3 - Afternoon]

Master transactions and asynchronous messaging

- Understand the principles of Jakarta Transactions and ensure the consistency of business operations
- Manage commits, rollbacks, and transactional errors
- Understand the interaction between CDI, persistence, and transactions
- Implement asynchronous communication with Jakarta Messaging
- Create producers and consumers and utilize queue and publish/subscribe models
- Hands-on workshop: Develop a transactional business process that incorporates asynchronous message sending and consumption.

[Day 4 - Morning]

Securing Jakarta EE Applications

- Understand the platform’s security mechanisms and Jakarta Security
- Implement user authentication and authorization
- Protecting resources and endpoints based on application roles
- Understand integration with modern identity management mechanisms
- Securing communications with HTTPS and applying best practices to APIs
- Hands-on workshop: Securing a Jakarta EE API with authentication, authorization, and role-based access control.

[Day 4 - Afternoon]

Testing, Deploying, and Operating Jakarta EE Applications

- Structuring unit tests and integration tests for a Jakarta EE application
- Testing business services, REST APIs, validations, transactions, and data access
- Understanding packaging formats and deploying the application to a compatible Jakarta EE runtime
- Configure the data sources, messaging services, and settings required by the application
- Containerize the application with Docker and utilize logs and diagnostic mechanisms
- Hands-on workshop: Test, package, containerize, and deploy a complete Jakarta EE application, then diagnose its operation.

[Day 5 - Morning]

Designing maintainable enterprise applications

- Organize the web, business, and data layers of an enterprise application
- Apply the principles of decoupling, modularity, and separation of concerns
- Selecting the appropriate Jakarta specifications based on functional and technical requirements
- Structure DTOs, entities, services, CDI components, and REST resources
- Identify common anti-patterns and improve performance and maintainability
- Hands-on workshop: Design the architecture of an application combining Jakarta REST, CDI, Persistence, Transactions, and Messaging.

[Day 5 - Afternoon]

Migrate and modernize a Java EE application to Jakarta EE

- Understand the stages of the evolution from Java EE to Jakarta EE
- Identify the impacts of switching from the javax.* namespace to the jakarta.* namespace
- Audit the dependencies, libraries, frameworks, and APIs of an existing Java EE application
- Adapt the code, configuration, build, and runtime to Jakarta EE
- Gradually modernize legacy components and validate the migration with non-regression tests
- Hands-on workshop: Migrate a Java EE application to Jakarta EE, resolve incompatibilities, and define its modernization roadmap.

Target Audience

This training is intended for both individuals and companies—large or small—seeking to train their teams in a new, advanced IT technology or to acquire specific business knowledge or modern methodologies.

Pre-training Assessment

The pre-training assessment complies with Qualiopi quality standards. Upon final registration, the learner receives a self-assessment questionnaire that allows us to evaluate their estimated proficiency in various types of technologies, as well as their expectations and personal goals for the upcoming training, within the limits imposed by the selected format. This questionnaire also allows us to anticipate certain connection or internal security issues within the company (intra-company or virtual classroom) that could pose challenges for monitoring and ensuring the smooth running of the training session.

Teaching Methods

Hands-On Training: 60% Practical, 40% Theoretical. Training materials distributed in digital format to all participants.

Organization

The course alternates between theoretical input from the instructor, supported by examples and

reflection sessions and group work.

Assessment

At the end of the session, a multiple-choice quiz is administered to verify that participants have successfully acquired the skills.

Certification

A certificate will be issued to each trainee who has completed the entire training program.