

**Price:** R8,600.00 excl. VAT  
**Duration:** 5 days  
**Code:** JEJB3

# Enterprise JavaBeans 3 Development

## Description

The Enterprise JavaBeans Development course focuses on JEE development using Enterprise JavaBeans version 3. The JEE architecture is revised. The types, uses and problems of EJBs are discussed. JEE patterns and best practices are presented.

## Objectives

Delegates who complete the Enterprise JavaBeans Development course will be able to:

- Understand the various JEE platform technologies.
- Write entity EJBs and deploy them on an application server.
- Write and deploy session EJBs.
- Write and deploy message-driven beans to receive JMS messages.

## Intended Audience

Experienced Java programmers and developers who need to develop scalable and robust enterprise applications using EJBs and the JEE environment. The Enterprise JavaBeans Development course is aimed at companies and individuals who are already doing JEE development and need training in the specific EJB aspects of the JEE architecture.

## Prerequisites

The Java Programming course and at least 1 year of Java programming experience. The JEE overview course is recommended.

## Course Contents

*The lecturer reserves the right to modify the contents of the course to suit the needs of the delegates.*

**JEE Architectural Overview** • JEE application servers. • Servlets and JavaServer Pages (JSP). • Enterprise JavaBeans (EJB). • Java Transaction API (JTA) Services (JTS). • RMI and RMI-IIOP. • Java Naming and Directory Interface (JNDI). • JavaMail. • Java Message Service (JMS). • Java Database Connectivity (JDBC). • Java APIs for XML Processing (JAXP) and binding (JAXB). • SOA and Web Services. • JAX-RPC and JAX-WS. • JEE Connector Architecture (JCA). • Java Authentication and Authorization Service (JAAS). • Java Persistence API (JPA).

**EJB Overview** • EJBs as the core of a JEE application. • EJB2 versus EJB3. • Ease of use and design simplification of EJB3. • OO design in EJB3 vs component design in EJB2. • Use of plain Java objects (POJOs) and interfaces (POJI). • Annotations vs deployment descriptors.

**EJB Types and Usage.** • Entity beans coded as POJOs. • Stateless and stateful session beans. • Exposing session beans as Web services. • Message driven beans. • Timers and the timer service. • EJB lifecycle, callbacks and listeners. • Interceptors. • JNDI Enterprise Naming Context (ENC). • Packaging and deploying EJBs. • Portability issues.

**Entity Beans and Persistence** • Overview of Java Persistence API. • Entity managers and persistence units. • Managed vs unmanaged entities. • Mapping persistent objects. • Entity relationships and inheritance. • EJB Query Language (QL).

**Transactions** • Overview of Java Transaction API (JTA and JTS). • The ACID principle. • Isolation and database locking. • Programmatic vs declarative transactions. • Exceptions and transactions.

**Messaging with JMS** • Point to point vs Publish and Subscribe messaging. • Queuing mechanisms. • Connection Factories, Connections, Sessions, Destinations. • Message types.

**Patterns and Best Practices** • EJB design in the real world. • JEE Design Patterns. • Best Practices in JEE development.