



Java EE 6: Develop Business Components with JMS & EJBs

- **Formato do curso:** Presencial
- **Preço:** 1800€
- **Nível:** Avançado
- **Duração:** 24 horas

This Java EE 6: Develop Business Components with JMS & EJBs training teaches you the knowledge required to build robust back-end functionality. Work with expert Oracle University instructors using Enterprise JavaBeans (EJB[™]) version 3.1 technology.

Learn To:

- Implement business-tier functionality using EJB technology.
- Assemble and deploy EJB technology business-tier components on an application server.
- Integrate an EJB technology-based application using the Java Messaging Service API.
- Create and implement timer-based services.
- Integrate transactions and security into an enterprise application.
- Describe best practices and other advanced issues in business component development with EJB technology.
- Explore the EJB technology coding experience of session beans and message driven-beans in a JMS application.
- Examine EJB design, best practices, transaction management, messaging fundamentals and security.

Benefits to You

You will learn to code session beans, message driven beans and use the JMS API. You will recognize the benefits of using EJB technology with respect to transaction management, messaging, and security in an enterprise application. You will also get a clear understanding of the EJB design best practices.

Java Platform, Enterprise Edition

This course features the Java Platform, Enterprise Edition 6 (Java EE 6) technology, and uses the Java EE 6 SDK. You'll perform the course lab exercises using the NetBeans Integrated Development Environment (IDE). This hands-on lab environment uses Oracle WebLogic Server 12c.

Destinatários

- Java Developer
- Java EE Developer

Pré-requisitos

Pré-requisitos obrigatórios:

- Developing Applications with Java EE 6 on WebLogic Server 12c
- Java SE 7 Programming

Pré-requisitos aconselháveis:

- Java SE 7: Develop Rich Client Applications
 - Developing Applications for the Java EE 6 Platform
-

Objetivos

- Implement interceptor classes and methods
 - Implement transactions
 - Implement exception handling for EJB technology
 - Create a timer using the Timer Service
 - Handle timer notification within an EJB component
 - Implement security for Java EE technology
 - Evaluate best practices for EJB technology
 - Develop a Java EE Application
 - Create message-driven beans
 - Examine the Java Platform, Enterprise Edition (Java EE)
 - Implement Enterprise JavaBeans (EJB 3.1) session beans
 - Use dependency injection with CDI
 - Use JPA for persistence
 - Develop Java EE technology applications using messaging
-

Programa

Introducing the Course

- Reviewing course objectives
- Discussing course format and LVC
- Discussing 4 day course schedule
- Getting acquainted with other students
- Reviewing the Java SE and Java EE Curriculum

Introducing Java EE

- Java Platform, Enterprise Edition(Java EE)
- Java EE application architecture

- Java EE container services
- EJB component types
- Comparison of Java EE application development with traditional enterprise application development

Implementing Session Beans

- Types of session beans
- Stateful Session beans
- Stateless Session beans
- Singleton Session beans
- Choosing a Session bean type
- Session Bean clients
- Tasks of creating a Session bean
- Packaging and deployment

Accessing Session Beans

- Using Naming Services
- JNDI API
- Developing Session Bean Clients
- Creating a Session Facade

Advanced Session Bean Concepts

- EJB Containers
- EJB Components
- Session Bean Identity
- Session Bean Life Cycle
- Lifecycle Event Handlers
- Asynchronous Communication
- Advanced Session Bean Configuration

Developing Singleton Session Bean

- Singleton Session Bean
- Singleton Session Bean Life Cycle
- Singleton Concurrency

Using Context and Dependency Injections

- Introduction to CDI Named Beans
- Scopes
- Qualifiers
- Alternatives

Using Java Persistence API

- What is JPA?

- Components of JPA architecture
- Entity operations
- Queries

Developing Java EE Applications Using

- Messaging Concepts
- Messaging Destinations
- Messaging Clients
- Messages
- Creating a Queue
- Message Producer
- Queue Message Browser
- Creating a Synchronous Queue Consumer

Developing Message-Driven Beans

- Introducing Message-Driven Beans
- Life Cycle of a Message-Driven Bean
- Creating JMS Message-Driven Beans
- Life Cycle Event Handlers
- Configuring the Message-Driven Bean

Using Timer Services

- Describing timer services
- Creating a timer callback notification
- Processing a timer callback notification
- Managing timer objects

Implementing Interceptor Classes and Methods

- Introducing Interceptors
- Types of Interceptors
- Interceptor methods
- Interceptor class
- Lifecycle call back interceptors

Implementing Transactions

- Transaction demarcation task,
- Transaction policy
- Container managed transactions
- Bean managed transactions
- Transaction in messaging

Implementing Security

- Security Interventions
- Java EE Platform Security Model
- Authentication
- Authorization
- Programmatic Security
- Declarative Security

Using EJB Technology Best Practices

- Exception Handling
- Java EE Application Design

Appendix A - Introducing Transactions

- Examining Transactions
- Types of Transactions
- Transaction-Related Concurrency Issues
- Handling Distributed Transactions
- Java Transaction API (JTA)

Introducing UML

- UML Basics
- Types UML Diagrams
- Elements of UML diagrams
- Illustrations of all UML diagrams