

## Java EE 6: Desarrollo de componentes de negocio con JMS y EJBs

**Duration:** 4 Days

### What you will learn

Esta capacitación Java EE 6: Desarrollo de componentes de negocio con JMS y EJBs le enseña lo necesario para construir funcionalidad backend robusta. Trabaje con instructores expertos de Oracle University utilizando tecnología Enterprise JavaBeans (EJB[™]) version 3.1.

### Learn To:

Implementar funcionalidad a nivel de negocios utilizando tecnología EJB; Armar y desplegar componentes de nivel de negocios de tecnología EJB en un servidor de aplicación.

Integrar una aplicación basada en tecnología EJB utilizando la API Java Messaging Service; Crear e implementar servicios basados en temporizador.

Integrar transacciones y seguridad a una aplicación de empresa.

Describir las mejores prácticas y otros temas avanzados en el desarrollo de componente de negocio con tecnología EJB.

Explorar la experiencia de codificación con tecnología EJB de beans de sesión y beans controlados por mensaje en una aplicación JMS.

Examinar diseño EJB, mejores prácticas, gestión de transacción, seguridad y conceptos fundamentales de mensajería.

### Beneficios que obtendrá

Aprenderá a codificar beans de sesión, beans controlados por mensaje y usar la API JMS. Reconocerá los beneficios de utilizar tecnología EJB con relación a la gestión de transacción, mensajería y seguridad en una aplicación de empresa. También entenderá claramente las mejores prácticas de diseño EJB.

### Java Platform, Enterprise Edition

Este curso es acerca de la tecnología Java Platform, Enterprise Edition 6 (Java EE 6), y utiliza Java EE 6 SDK. Realizará los ejercicios de laboratorio del curso usando NetBeans Integrated Development Environment (IDE). Este entorno de práctica utiliza Oracle WebLogic Server 12c.

### Audience

Developer

### Related Training

#### *Required Prerequisites*

Java SE 7 Programming

*Suggested Prerequisites*

Developing Applications for the Java EE 6 Platform

Java SE 7: Develop Rich Client Applications

**Course Objectives**

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

## Course Topics

### Introducción al curso

- Revisión de objetivos del curso
- Analizar el formato del curso y LVC
- Charla acerca del cronograma de curso de 4 días
- Conocer a los demás estudiantes
- Revisión del programa de estudios Java SE y Java EE

### Introducción a Java EE

- Java Platform, Enterprise Edition(Java EE)
- Arquitectura de aplicaciones Java EE
- Servicios de contenedor Java EE
- Tipos de componente EJB
- Comparación del desarrollo de aplicaciones Java EE con el desarrollo tradicional de aplicaciones de empresa

### Implementación de beans de sesión

- Tipos de beans de sesión
- Bean de sesión con estado
- Bean de sesión sin estado
- Bean de sesión Singleton
- Selección de un tipo de bean de sesión
- Cientes bean de sesión
- Tareas para crear un bean de sesión
- Empaquetado y despliegue

### Acceso a beans de sesión

- Uso de servicios de nomenclatura
- API de JNDI
- Desarrollo de clientes bean de sesión
- Creación de una fachada de sesión

### Conceptos avanzados de bean de sesión

- Contenedores EJB
- Componentes EJB
- Identidad de bean de sesión
- Ciclo de vida de bean de sesión
- Manejadores de evento de ciclo de vida
- Comunicación asíncrona
- Configuración avanzada de bean de sesión

## **Desarrollo de beans de sesión Singleton**

Bean de sesión Singleton

Ciclo de vida de bean de sesión singleton

Concurrencia singleton

## **Uso de contexto e inyecciones de dependencia**

Introducción a beans con nombre CDI

Ámbitos

Calificadores

Alternativas

## **Uso de Java Persistence API**

¿Qué es JPA?

Componentes de arquitectura JPA

Operaciones de entidad

Consultas

## **Desarrollo de aplicaciones Java EE utilizando**

Conceptos de mensajería

Destinos de mensajería

Clientes de mensajería

Mensajes

Creación de una Cola

Productores de mensaje

Browser de cola de mensajes

Creación de un consumidor de cola síncrono

## **Desarrollo de beans controlados por mensaje**

Introducción a los beans controlados por mensaje

Ciclo de vida de un bean controlado por mensaje

Creación de beans JMS controlados por mensaje

Manejadores de eventos de ciclo de vida

Configuración del bean controlado por mensaje

## **Uso de servicios de temporizador**

Descripción de servicios de temporizador

Creación de una notificación de devolución de llamada de temporizador

Procesamiento de una notificación de devolución de llamada de temporizador

Gestión de objetos de temporizador

## **Implementación de métodos y clases de interceptor**

Introducción a los interceptores

Tipos de interceptores

Métodos de interceptor

Clase de interceptor

Interceptores de devolución de llamada de ciclo de vida

## **Implementación de transacciones**

Tarea de demarcación de transacción

Política de transacción

Transacciones administradas por contenedor

Transacciones gestionadas por bean

Transacción en mensajería

## **Implementación de seguridad**

Intervenciones de seguridad

Modelo de seguridad de plataforma Java EE

Autenticación

Autorización

Seguridad programática

Seguridad declarativa

## **Uso de recomendaciones de la tecnología EJB**

Manejo de excepciones

Diseño de aplicaciones Java EE

## **Apéndice A - Introducción a las transacciones**

Análisis de transacciones

Tipos de transacciones

Problemas de concurrencia relacionados con la transacción

Manejo de transacciones distribuidas

Java Transaction API (JTA)

## **Introducción a UML**

Conceptos básicos de UML

Tipos de diagramas UML

Elementos de diagramas UML

Ilustraciones de todos los diagramas UML