

Price: R1,800.00 excl VAT
Duration: 1 day
Code: J2EEV

JEE Technology Overview

Description

The JEE Technology overview course is for companies and individuals who are entering the JEE field for the first time and require an overview of the JEE architecture to understand the use of the appropriate technologies. Topics include an overview of JEE technology and architectures, including Java servlets and Java Server Pages, Enterprise JavaBeans and others.

Companies and individuals who are already doing JEE development and require training in specific aspects of the JEE architecture are advised to attend the more specialised Java Servlets and JSP course and the Enterprise JavaBeans Development course.

This course is run in parallel with the JEE Development course.

Objectives

Delegates who attend the JEE Technology Overview will be able to:

- Understand the concepts of distributed computing and component-based architectures.
- Understand the JEE architecture and choose appropriate JEE technology.
- Understand the differences between the JEE and .NET platforms.
- Understand the role of web services in distributed computing.

Intended Audience

The JEE overview is suitable for project and programme managers, system architects, developers and programmers who require an understanding of the JEE environment.

Prerequisites

None

Course Contents

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

Distributed Computing Overview • Monolithic software development. • Client-server model. • Multi-tier development. • Origins of and need for distributed computing. • Component-based architectures and JEE containers. • Web Services.

JEE API Overview • Servlets and Java Server Pages (JSP). • Enterprise JavaBeans (EJB). • Transaction APIs: JTA and JTS. • Java Database Connectivity (JDBC). • Java Naming and Directory Interface (JNDI). • Remote Method Invocation (RMI) and RMI-IIOP. • Java Authentication and Authorization Service (JAAS). • JavaMail API. • Java Message Service (JMS). • JEE Connector Architecture (JCA). • Java APIs for XML Processing (JAXP) and binding (JAXB). • Java Persistence API (JPA). • SOA and Web Services. • Comparison between JEE and .NET architectures. • JEE design patterns and best practices in JEE development.

Java Servlets and Java Server Pages (JSP) • Static vs dynamic pages. • HTTP requests, responses, cookies and sessions. • Usage of servlets and JSP pages. • JSP Standard Tag Library (JSTL). • JavaBeans and the MVC architecture. • Web development frameworks.

The role of EJBs • EJBs as the core of a JEE application. • Entity beans as core business data. • Session beans as business processes. • Message-driven beans as decoupled business processes. • The use of web services.

EJB types, uses and versions • Components of an EJB. • Stateless and stateful session beans. • Bean-managed persistence (BMP) and container-managed persistence (CMP). • Message driven beans and message queue (MQ) servers. • EJB2 versus EJB3. • JPA as the heart of EJB3 persistence.