

**Price:** R14,800.00 excl. VAT  
**Duration:** 5 days  
**Delivery:** Virtual classroom or  
on-site training

## JEE Development

*Available on request for group bookings only*

### Description

The JEE Development course will give you an overview of how the JEE technologies are organised and how they fit together. Java Enterprise Edition (JEE; now called Jakarta EE) is a collection of technologies for the Java platform that is designed to support large, distributed software systems in a corporate environment. You will learn how to use Java Servlets and Java Server Pages (JSP) to build dynamic web applications. You will also learn how to use Enterprise JavaBeans (EJBs) to build Java systems with reusable components.

### Objectives

After you have completed the JEE Development course, you will be able to:

- Understand the JEE architecture and choose appropriate JEE technology.
- Develop applications using the JEE platform.
- Write Java Server Pages and servlets, and deploy them on a application server.
- Write an Enterprise JavaBean (EJB) and deploy it on an application server.
- Expose the EJB as a SOAP web service and access it from a JSE application.

### Intended Audience

You should attend the JEE Development course if:

- You are a Java programmer and you need to develop systems using the JEE architecture.
- You are an architect, project manager or systems analyst and you need to understand the JEE architecture.
- You need a detailed overview of JEE to understand the role of the various technologies.

If you need an in-depth understanding of specific technologies within JEE, then you should think about attending one of our more specialised courses:

- Enterprise JavaBeans course.
- Java Web Services course.

### Prerequisites

Before you attend the JEE Development course:

- You must have attended our Java Programming course or already be comfortable with the fundamentals of the Java programming language.

### Course Contents

#### **JEE Technologies**

- Servlets and Java Server Pages (JSP).
- Enterprise JavaBeans (EJB).

- Java Naming and Directory Interface (JNDI).
- Java Database Connectivity (JDBC).
- Java Persistence API (JPA).
- Java Transaction API (JTA) and Transaction Service (JTS).
- Remote Method Invocation (RMI) and RMI-IIOP.
- SOAP and REST web services.
- Java Message Service (JMS).
- JEE Connector Architecture (JCA).
- JavaMail.

### Java Servlets and JSPs

- The role and use of servlets and JSPs.
- Servlet lifecycle.
- Request, response, cookie and session objects.
- JSP scriptlets, declarations, expressions, directives.
- Custom tags and the JSP Standard Tag Library (JSTL).
- MVC architecture.
- Filters and event listeners.
- Expression Language (EL).
- Web applications and deployment descriptors.

### Enterprise JavaBeans

- Comparison between EJB2 and EJB3.
- Annotation-driven development.
- EJB interfaces and components.
- EJB lifecycle.
- Stateless, stateful and singleton session beans.
- Message driven beans (MDB).
- EJB3 entity beans and the JPA.
- Programmatic vs declarative transactions.
- Interceptors and decorators.

### Messaging with JMS

- Message driven beans and message queue (MQ) servers.
- Message types and usages.
- Point-to-point versus publish and subscribe messaging.

### Web Services in JEE

- Service Oriented Architecture with SOAP.
- Resource Oriented Architecture with REST.

- SOAP web services with JAX-WS.
- REST web services with JAX-RS.
- Best Practices.

#### **JNDI, RMI and CORBA**

- Referencing remote objects with JNDI.
- Serialization.
- RMI and RMI-IIOP.
- CORBA and Java IDL.

#### **Additional JEE Topics**

- JEE Core Design Patterns.
- Best Practices.
- Spring Framework as an alternative to JEE.

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