

Price: R12,900.00 excl. VAT
Duration: 5 days
Code: WBSVS

Java Web Services

Description

A web service is a software program that makes itself available over the internet for another system to use. Web services use a set of international standards for communication between different systems. This course will teach you how to develop web services using Java. You will learn about the various standards involved, the difference between SOAP and RESTful web services, and how to write client programs that use web services.

Objectives

After you have completed the Java Web Services course, you will be able to:

- Understand the various web service technologies and how to use them.
- Write SOAP and RESTful web services.
- Write Java clients that can use SOAP and RESTful web services.

Intended Audience

You should attend the Java Web Services course if:

- You are a Java programmer and you need to develop web services.
- You are a Java programmer and you need to support web services.
- You are a Java programmer and you want to use web services for internal interfaces.

Prerequisites

Before you attend the Java Web Services course:

- You must have a good knowledge of the Java language. So you should have attended our Java Programming course or already have practical experience programming in Java.
- You should have some knowledge of basic XML. It is not essential, but it will help you to first attend our XML Development course.

Course Contents

Web Services Overview.

- Why use Web Services in enterprise applications.
- Service-Oriented Architecture (SOA).
- Web services as a realisation of SOA.
- Web Service architecture, initiatives and specifications.
- Web Service application programming interfaces (APIs).

Web Services Technologies and APIs.

- SOAP web services as service oriented architectures.
- Binary Attachments using SAAJ, DIME and MTOM.
- Web Service Description Language (WSDL).
- Web Service Interoperability (WS-I).
- Java API for XML-Based Remote Procedure Calls (JAX-RPC).
- Java API for XML Web Services (JAX-WS).
- Java Architecture for XML Binding (JAXB).
- RESTful web services as resource oriented architectures.

- SOAP vs REST.
- Java API for XML RESTful Web Services (JAX-RS).

Overview of XML (EXtensible Markup Language).

- XML syntax and validity.
- XML namespaces.
- Document Type Definition (DTD).
- XML schemas.
- Java API for XML Processing (JAXP).
- Java API for XML Binding (JAXB).

Web Services Endpoints and Clients.

- Exposing Web service endpoints as JSE applications, servlets and EJBs.
- Approaches to creating Web services - the code first vs contract first approaches.
- Java-to-WSDL vs WSDL-to-Java development.
- Accessing web service functionality from clients.

Security and Exceptions.

- Handling exceptions.
- User-defined exceptions.
- Web services security requirements.
- Basic authentication.
- Message-layer security.
- Second generation WS-* initiatives.

RESTful web services.

- Overview of HTTP features - HTTP methods, headers, query strings, status codes.
- HTTP methods and CRUD systems.
- Designing REST URLs.
- Data representations - XML, JSON, text.
- Implementation with various Java APIs - servlet API, JAX-RS, @WebServiceProvider.
- Consuming REST web services.

Best Practices and Design Patterns.

- Best Practices and choosing the correct WS technology.
- Use of JEE design patterns in web services architecture.
- Practical application of JEE design patterns.

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