

Java Web Services, REST & Microservices

Course 6928 – 32 Hours

Overview

Starting with EAI, followed by SOA, the story ends with Services. Appearance of mobile and IoT as the next generation of internet clients motivated organizations to publish their services on the web. Today, most business services deployed on any platform (domains, private & public clouds) are available on the internet. Any new backend feature is followed by integration tier implemented as a Web-Service .

This course deep dives into the architecture used in last decade and evolved into the lightweight, fast and effective ways and methods used today. The course relates to both XML but mostly REST web services and how to implement it via JAX-WS & JAX-RS. The course introduces Full-Stack development using Angular as a ClientMVC framework. The course also explores Microservices discussing the patterns and how to correctly design and use Java Web APIs and web containers for implementing successful Microservices.

Who Should Attend

- Java developers that maintains SOAP based web-services
- Java developers that want to create REST based web-services
- Developers that want to implement Microservices with Java

Prerequisites

- Experience in JavaSE
- Familiarity with client side technologies (HTML, JavaScript)

Course Contents

Architecture

- 2,3,N tier model
- RPC
- XML, JSON & Binding (JAXB)
- MVC Model 2
- SPA
- Future internet clients

Intro to web development

- Intro to HTTP
- Java web module components
- Servlets API – intro
- Basic configuration
- Understanding HTTPRequest, response & sessions

Web-Services – JAX-WS

- Architecture
- Intro to WSDL
- Intro to SOAP
- Building services with JAX-WS
- Building clients with JAX-WS

REST – JAX-RS

- Why REST?
- Going stateless
- HTTP for RPC
- RESTful methodology
- Creating REST APIs with JAX-RS
- Unit Testing with REST client API
- Managing sessions
- Applying declarative security

Microservices

- Why & what is?
- Creating microservices with JAX-RS
- Benefits and drawbacks
- Challenges
- Best practices
- Patterns
 - Decomposition patterns
 - Storage patterns
 - Discovery & integration patterns
 - Invocation patterns
 - Deployments
 - Cross-cutting concerns patterns
 - Testing patterns
 - Circuit Breaker
 - Access Token
 - Observability patterns
 - UI patterns

Intro to clientMVC – Angular Framework

- Intro to client MVC
- Understand the need in Java full stack
- Intro to Angular framework