Java Web Services – Course Outline

# Duration

2 days

# Objectives

* Design and develop SOAP based web services using JAX-WS
  + Understand WSDL, Schema definition using code first approach (server side)
  + Understand proxy-stubs, XML binding using XML first approach (client side)
* Design and develop RESTful web services using Spring MVC
  + Understand concepts of REST
  + Perform CRUD operations using Spring MVC over REST API
  + Handle security and multi-part file upload using RESTful services

# Pre-requisite

All attendees must be experienced Java developers with a basic understanding of web application development.

# Outline

## Day 1 – Session 1 - Java web services (JAX-WS)

* Overview

Why Web Services?

Service-Oriented Architecture

Simple Object Access Protocol (SOAP)

Web Service Description Language (WSDL)

Universal Description, Discovery and Integration (UDDI)

* The Simple Object Access Protocol (SOAP)

Messaging Model

Namespaces

SOAP over HTTP

The SOAP Envelope

The Message Header

The Message Body

* The Java API for XML-Based Web Services

How It Works - Build Time and Runtime

The Service Endpoint Interface

Working from WSDL

Working from Java

RPC and Document Styles

One-Way Messaging

Binary Protocols

* Web Services Description Language

Web Services as Component-Based Software

The Need for an IDL

Web Services Description Language

WSDL Information Model

The Abstract Model - Service Semantics

Message Description

Messaging Styles

The Concrete Model - Ports, Services, Locations

Extending WSDL - Bindings

Service Description

* The Java API for XML Binding

The Need for Data Binding

XML Schema

Two Paths

JAXB Compilation

Mapping Schema Types to Java

Java-to-XML Mapping Using Annotations

Marshaling and Unmarshaling

Working with JAXB Object Models

In-Memory Validation

## Day 1 – Session 2 –RESTful Web services and JAX-RS

* Introduction to REST
  + REST and the Rebirth of HTTP
  + RESTful Architectural Principles
* Designing RESTful Services
  + The Object Model
  + Model the URIs
  + Defining the Data Format
  + Assigning HTTP Methods
* First JAX-RS Service
  + Developing a JAX-RS RESTful Service
  + Deploying Our Service
* HTTP Method and URI Matching
  + Binding HTTP Methods
  + @Path
  + Subresource Locators
* JAX-RS Injection
  + The Basics
  + @PathParam
  + @MatrixParam
  + @QueryParam
  + @FormParam
  + @HeaderParam
  + @CookieParam
  + Common Functionality
* JAX-RS Content Handlers
  + Built-in Content Marshalling
  + JAXB
  + Custom Marshalling
* Response Codes, Complex Responses, and Exception Handling
  + Default Response Codes
  + Complex Responses
  + Exception Handling

## Day 2 – Session 1 –RESTful Web services using Spring MVC

* Overview of Spring MVC
  + Spring MVC Configuration
  + Role of DispatcherServlet
  + Controller initialization and mapping
  + Overview of View and Model
  + Data binding in View
* Spring RESTful API
  + Exposing the JAX-WS API implemented earlier via Spring RESTful API
  + Annotations and REST mapping
  + Overview of Content handler
  + Using JSON for request and response

## Day 2 – Session 2 –Security and Attachments via Spring RESTful Web services

* Overview of Spring security
  + User authentication
  + Authorization using roles and URL mapping
* Securing Spring RESTful services
  + Configuring Spring security filters for RESTful services
  + Storing and accessing the user session
  + Token based security (OAuth2)
* Sending binary attachments to RESTful services
  + Configuring Multipart handler
  + Reading content from MultipartEntity

# Hardware & Network Requirements

* All participants to have individual desktops with at least dual core or higher CPU with minimum 2GB RAM
* All participants’ system to be connected to internet

# Software Requirements

* Windows XP/7 Operating System
* Java 1.6 (preferred)
* Eclipse IDE (preferably 4.2 or higher) for Java EE developers
* Apache Tomcat (7.0 or above)