Spring Boot and Spring Cloud

Duration: 40 hrs

Day 1: Spring Boot Micro Services

**Demystifying Microservices**

* The evolution of microservices
* What are microservices?
* Microservices – the honeycomb analogy
* Principles of microservices
* Characteristics of microservices
* Challenges for a successful microservice architecture
* Microservices examples
* Microservices benefits
* Microservices vs SOA
* Relationship with other architecture styles
* Microservice use cases

**Building Microservices with Spring Boot**

* Setting up a development environment
* Using Spring Boot to build RESTful microservices
* Getting started with Spring Boot
* Understanding Maven build tool
* Developing the Spring Boot Java microservice using STS
* The Spring Boot configuration
* REST APIs 101
* Implementing a Simple REST API with Spring boot
* Cloud Native Application
* Using “Twelve-Factor App” Style Configuration
* Read External Configuration
* Support multiple environments using Spring Profiles
* Implementing Spring Boot Controllers and Exception Handling
* Implementing Spring Boot Bean Validation
* Implementing Unit testing and Mock Testing for Spring Boot
* Implementing Spring Boot Developer Tools

Day 2 Spring Boot Micro Services Contd...

* Rest API and Richardson Maturity Model
* Implementing Spring Boot HAL Explorer
* Implementing Spring Boot JPA using H2
* Implementing Spring Data REST using H2
* Spring Boot actuator

**Spring Cloud**

* Overview of Spring cloud and its modules

**Spring Cloud Config**

* Configuring Server and client pointing to configuration file in github
* @EnableConfigServer

**LAB : Using Spring Cloud Config server and Config client**

**Spring Cloud Eureka**

* Why DiscoveryServer?
* Client side load balancing
* How to Create a Eureka Server using @EnableEurekaServer or @EnableDiscoveryServer
* How to register Eureka Client using @EnableEurekaClient or @EnableDiscoveryClient
* How to look up a service from Eureka Server

**LAB : Using Eureka Server and Client LAB : Configuring Eureka cluster**

**Spring Cloud Ribbon**

* Why ClientSide Load Balancing?
* What is ribbon and how to use spring ribbon?
* Using LoadBalanced RestTemplate
* Configuring Retries
* Using different Load balancing algorithms

**LAB : Implementing Client side load balancing using ribbon**

**Spring Cloud Feign**

* What is a Feign?
* Implementing REST Clients in declarative approach
* Configuring Fallbacks using feign
* Hystrix configurations when using feign

**LAB : Implementing REST Client using Feign**

**Spring Cloud Hystrix**

* What is a circuit breaker?
* Using Hystrix with Ribbon
* Various Hystrix configurations
* Hystrix Dashboard for monitoring

**LAB : Using Hystrix as circuit breaker**

**LAB : Using Hystrix Dashboard**

Day 3:

**API Gateway and ZUUL**

* Why API Gateway?
* What is ZUUL ?
* Disabling access to some services at zuul level
* Using Zuul Filters

**LAB : Using ZUUL to enable Proxying microservices**

**Day 4:** [**Connecting Microservices Through Messaging**](https://app.pluralsight.com/player?course=java-microservices-spring-cloud-coordinating-services&author=richard-seroter&name=3c4d1d96-609d-4980-8347-f1d5c259d79d&clip=0&mode=live)

* Connecting Microservices Through Messaging
* The Role of Messaging and the Problem with the Status Quo
* About Spring Cloud Stream
* Understanding Binders and Configuring Stream Applications
* Creating Message Senders and Receivers
* More Options for Producing and Consuming Messages
* The Role of Processors in Spring Cloud Stream
* Creating Custom Interfaces and Using Processors
* RabbitMQ and Apache Kafka Binding Properties
* Using Consumer Groups to Scale
* Pushing Messages and Configuring Consumer Groups
* Stateful Processing with Partitions
* Working with Content Types
* Spring Cloud Stream Health and Metrics
* **Spring Cloud – Zookeeper**
* **Spring Cloud - Security**
* **Spring Cloud –Sleuth**
* **Spring Cloud – Zipkin**
* **Final Considerations for Building Production-ready Systems**

Making system easy for beginners

Making system complete so that it can serve as the foundation for enterprise applications

**Day 5:** [**Building Data Processing Pipelines Out of Microservices**](https://app.pluralsight.com/player?course=java-microservices-spring-cloud-coordinating-services&author=richard-seroter&name=18412e5c-f352-4c68-b927-09d077a9ebba&clip=0&mode=live)

* Building Data Processing Pipelines Out of Microservices
* The Role of Orchestration and the Problem with the Status Quo
* About Spring Cloud Data Flow
* About the Data Flow Server
* Consider Streams vs. Tasks
* Installing Spring Cloud Data Flow
* Getting Spring Cloud Data Flow up and Running
* About Stream Starter Apps
* Creating Streams with Spring Cloud Data Flow
* Deploying Data Pipelines
* Creating Tasks with Spring Cloud Data Flow
* Using the Spring Cloud Data Flow Dashboard and Flo
* Building, Deploying, and Tapping Streams from the Dashboard
* Creating Custom Stream or Task Apps
* Creating, Registering, Using, and Partitioning Custom Apps
* Creating Composed Tasks
* Creating Custom Task Apps and Arranging in a Composed Task
* Monitoring Spring Cloud Data Flow Pipelines
* Updating Apps, Streams, and Tasks