

**Developing Microservices**

**Course Number:** MSV-112WA
**Duration:** 5 days

**Overview**

This Developing Microservices training course teaches attendees how to leverage Docker, Jenkins, Couchbase, Spring Boot, Kafka, and more to ensure the microservices development process is efficient, scalable, and robust.

**Prerequisites**

All attendees must have basic knowledge of a modern programming language and software design principles.

**Materials**

All Microservices training students receive comprehensive courseware.

**Software Needed on Each Student PC**

Attendees will not need to install any software on their computer for this class. The class will be conducted in a remote environment that Accelebrate will provide; students will only need a local computer with a web browser and a stable Internet connection. Any recent version of Microsoft Edge, Mozilla Firefox, or Google Chrome will be fine.

**Objectives**

* Confidently use Docker, Jenkins, Couchbase, Spring Boot, and Kafka to construct and deploy data-driven microservices
* Migrate applications to Microservice-architected solutions on Docker, Kubernetes, and Jenkins with OpenShift
* Understand the various components in a containerized Microservice environment for CI/CD (continuous integration/continuous deployment)

**Outline**

* Introduction to Microservices Development
	+ What are Microservices?
	+ Microservices vs. Classic SOA
	+ Principles of Microservices Architecture Design
	+ Business Domain-Centric Design
	+ Designing for failure
	+ Microservices Architecture
	+ Docker and Microservices
	+ Microservice Deployment with Docker
	+ Writing Dockerfile
	+ Kubernetes
	+ What is OpenShift
	+ OpenShift Architecture
	+ Microservices and Various Applications
	+ Web Applications
	+ Rich Client Applications
	+ Rich Internet Applications
	+ Mobile Applications
	+ Service Applications
	+ Single Page Applications
	+ Traditional Enterprise Application Architecture
	+ Sample Microservices Architecture
	+ Serverless & Event-driven Microservice – AWS Lambda
* REST Services
	+ Many Flavors of Services
	+ Understanding REST
	+ Principles of RESTful Services
	+ SOAP Equivalent Examples
	+ REST Example – JSON
	+ REST vs. SOAP Communication
	+ RESTful Services Usage
	+ Additional Resources
* Advanced Objects and Functionality in JavaScript
	+ JavaScript Evolution
	+ Basic Objects
	+ Constructor Function
	+ More on the Constructor Function
	+ Object Properties
	+ Deleting a Property
	+ The Instance of Operator
	+ Object Properties
	+ Constructor and Instance Objects
	+ Constructor Level Properties
	+ Namespace
	+ Functions Are First-Class Objects
	+ Closures
	+ Private Variables with Closures
	+ Immediately Invoked Function Expression (IIFE)
	+ The Module Pattern
	+ Module Pattern Example
	+ Prototype
	+ Inheritance in JavaScript
	+ The Prototype Chain
	+ Traversing Prototype Property Hierarchy
	+ Prototype Chain
	+ Inheritance Using Prototype
	+ Extending Inherited Behavior
	+ Enhancing Constructors
	+ Improving Constructor Performance
	+ Inheritance with Object.create
	+ The hasOwnProperty Method
* React Overview
	+ What is React?
	+ What's in a Name?
	+ React Component Model
	+ What React Is Not
	+ What You Will Not Find in React
	+ Motivation for Creating React
	+ A React JavaScript Example
	+ One-Way Data Flow
	+ JSX
	+ A JSX Example
	+ The Virtual (Mock) DOM
	+ Only Sub-components that Actually Change are Re-Rendered
	+ React Libraries
* Programming with React API
	+ React Programming Options
	+ Components vs. Elements
	+ Three Ways to Create a React UI Component
	+ React API On-Line Documentation
	+ Setting Up the Libraries
	+ The ReactDOM Object
	+ The React Object
	+ The React.createElement Method
	+ The ReactElement Object
	+ The ReactElement Structure
	+ The React.DOM Object
	+ The React.PropTypes Object
	+ The React.Children Object
	+ The propTypes Object
	+ Lifecycle Methods (Applied only to ES6 Classes)
* Basic Components and JSX
	+ What is JSX?
	+ JSX Transpilation to React Code Example
	+ Running the Transpiled Code
	+ Babel
	+ Babel JavaScript Library
	+ Script Import Skeleton Code
	+ Playing Around in CodePen
	+ React Components and Properties (Props)
	+ Ways to Create UI Components
	+ Creating a Functional Component Example
	+ Component Names Must Be Capitalized
	+ Creating a UI Component with React.createClass()
	+ The render Method Object
	+ Creating a UI Component Using ES6 Class Notation
	+ Using ES6 Classes with React
	+ Which UI Component Creation Syntax Should I Use?
	+ Components vs Elements
	+ Elements Are Immutable
	+ Properties
	+ Property Naming Convention
	+ Properties Default to 'True'
	+ Spread Attributes (an ES6 Feature)
	+ Expressions
* Introduction to Node.js
	+ What Is Node.js?
	+ Application of Node.js
	+ Installing Node.js and NPM
	+ "Hello, Node World!"
	+ How It Works
	+ Built on JavaScript: Benefits
	+ Traditional Server-Side I/O Model
	+ Disadvantages of the Traditional Approach
	+ Event-Driven, Non-Blocking I/O
	+ Concurrency
	+ Using Node Package Manager (NPM)
	+ Express
	+ Microservices with Node.js
	+ The Express Package
	+ Installing and Using Express
	+ Defining Routing Rules in Express
	+ Route Path
	+ The Response Object
	+ A Simple Web Service with Express Example
	+ Composite Services
	+ Call an API Using a Promise
	+ Using the callApi() Function
* Extending React
	+ The Need to Extend React
	+ Redux Design Ideas
	+ React Router Code Examples
	+ Issues With Manual Module Management
	+ Webpack
	+ Testing React Apps: ReactTestUtils and Jest
	+ Testing with Jest and Enzyme
* React Component Concepts
	+ Nesting JSX Elements
	+ Example of JSX Nesting
	+ Comments in JSX Code
	+ JSX Escapes Values
	+ Event Handling
	+ Event Handler Example
	+ Working with Lists of Items
	+ Keys in Lists
	+ Example List With Key
	+ Container vs. Presentational Components
	+ State
	+ Types of State Data
	+ State Hierarchy
	+ Lift State Up
	+ Props vs. State
	+ Pass Down a Function
	+ Immutability
	+ Virtual DOM and State
	+ Setting state
	+ Updating Input fields
	+ Passing Props to Components
	+ Passing Functions to Components
	+ Event Binding - Dos and Dont's
	+ Passing Parameters to Event Handlers
	+ App Development Workflow
* Introduction to Spring Boot
	+ What is Spring Boot?
	+ Spring Framework
	+ How is Spring Boot Related to Spring Framework?
	+ Spring Boot Main Features
	+ Spring Boot on the PaaS
	+ Understanding Java Annotations
	+ Spring MVC Annotations
	+ Example of Spring MVC-based RESTful Web Service
	+ Spring Booting Your RESTful Web Service
	+ Spring Boot Skeletal Application Example
	+ Converting a Spring Boot Application to a WAR File
	+ Externalized Configuration
	+ Starters
	+ Maven - The 'pom.xml' File
	+ Spring Boot Maven Plugin
	+ Gradle - The 'build.gradle' File
	+ Spring Boot Maven Plugin
	+ Creating a Spring Boot Application
* Spring MVC
	+ Spring MVC
	+ Spring Web Modules
	+ Spring MVC Components
	+ DispatcherServlet
	+ Template Engines
	+ Spring Boot MVC Example
	+ Spring MVC Mapping of Requests
	+ Advanced @RequestMapping
	+ Composed Request Mappings
	+ Spring MVC Annotation Controllers
	+ Controller Handler Method Parameters
	+ Controller Handler Method Return Types
	+ View Resolution
	+ Spring Boot Considerations
* Overview of Spring Database Integration
	+ DAO Support in Spring
	+ Spring Data Access Modules
	+ Spring JDBC Module
	+ Spring ORM Module
	+ DataAccessException
	+ @Repository Annotation
	+ Using DataSources
	+ DAO Templates
	+ DAO Templates and Callbacks
	+ ORM Tool Support in Spring
* Using Spring with JPA or Hibernate
	+ Spring JPA
	+ Benefits of Using Spring with ORM
	+ Spring @Repository
	+ Using JPA with Spring
	+ Configure Spring Boot JPA EntityManagerFactory
	+ Application JPA Code
	+ "Classic" Spring ORM Usage
	+ Spring JpaTemplate
	+ Spring JpaCallback
	+ JpaTemplate Convenience Features
	+ Spring Boot Considerations
	+ Spring Data JPA Repositories
	+ Database Schema Migration
	+ Database Schema Migration for CI/CD using Liquibase
	+ How Liquibase Works?
	+ Changelogs in Liquibase
	+ Preconditions in Changelogs
	+ Sample Empty Changelog
	+ Sample Precondition in Changelog
	+ Sample Changeset in Changelog
	+ Running Liquibase
	+ Liquibase Commands
* Spring REST Services
	+ REST Services With Spring MVC
	+ Spring MVC Components
	+ Spring MVC @RequestMapping with REST
	+ Working With the Request Body and Response Body
	+ @RestController Annotation
	+ Implementing JAX-RS Services and Spring
	+ JAX-RS Annotations
	+ Spring Security
	+ Spring Security Options
	+ Spring Security Features
	+ Java Clients Using RestTemplate
	+ RestTemplate Methods
* Spring Security
	+ Securing Web Applications with Spring Security 3.0
	+ Spring Security 3.0
	+ Authentication and Authorization
	+ Programmatic v Declarative Security
	+ Getting Spring Security from Maven
	+ Spring Security Configuration
	+ Spring Security Configuration Example
	+ Authentication Manager
	+ Using Database User Authentication
	+ LDAP Authentication
* Spring JMS
	+ Spring JMS
	+ JmsTemplate
	+ Connection and Destination
	+ JmsTemplate Configuration
	+ Transaction Management
	+ Transaction Configuration
	+ Producer
	+ Consumer
	+ Converting Messages
	+ Message Listener Containers
	+ Message-Driven POJO's Async Receiver Configuration
	+ Spring Boot Considerations
* Introduction to Couchbase
	+ Key Components of Couchbase
	+ Benefits of Couchbase
	+ Basics of Data Modeling
	+ Modeling One-to-many Relationship
	+ Doing a Query
	+ About Query Index
	+ Example MapReduce View
* Introduction to Couchbase Programming Using Java
	+ Getting Started
	+ Opening a Connection
	+ Creating Index
	+ Doing a Query Using MapReduce View
	+ Doing an N1QL Query
	+ Retrieve a Document by ID
	+ Adding a Document
	+ Updating a Document
	+ Deleting a Document
* Introduction to KAFKA
	+ Messaging Architectures
	+ Need for Kafka
	+ Kafka Partitions
	+ Kafka Architecture
	+ Core concepts in Kafka
	+ Why Kafka Cluster?
	+ Sample Multi-Broker Cluster
	+ Overview of ZooKeeper
	+ Kafka Cluster & ZooKeeper
	+ Who Uses Kafka?
* Using Apache Kafka
	+ Installing Apache Kafka
	+ Configuration Files
	+ Starting Kafka
	+ Using Kafka Command Line Client Tools
	+ Setting up a Multi-Broker Cluster
	+ Using Multi-Broker Cluster
	+ Kafka Connect – Configuration Files
	+ Using Kafka Connect to Import/Export Data
	+ Creating a Spring Boot Producer
	+ Adding Kafka dependency to pom.xml
	+ Defining a Spring Boot Service to Send Message(s)
	+ Defining a Spring Boot Controller
	+ Testing the Spring Boot Producer
	+ Creating a Nodejs Consumer
* Introduction to Kubernetes
	+ What is a Container?
	+ Pros and Cons of Containers
	+ Composition of a Container
	+ Control Groups
	+ Namespaces
	+ Union Filesystems
	+ Popular Containerization Software
	+ Microservices and Containers / Clusters
	+ Microservices and Orchestration
	+ Microservices and Infrastructure-as-Code
	+ Kubernetes Container Networking
	+ Kubernetes Networking Options
* Kubernetes – From the Firehose
	+ Container Orchestration
	+ Kubernetes Basic Architecture
	+ Kubernetes Detailed Architecture
	+ Kubernetes Concepts
	+ Cluster and Namespace
	+ Node
	+ Master
	+ Pod
	+ Label
	+ Annotation
	+ Label Selector
	+ Replication Controller and Replica Set
	+ Service
	+ Storage Volume
	+ Secret
	+ Resource Quota
	+ Authentication and Authorization
	+ Routing
	+ Registry
	+ Using Docker Registry
* Docker Introduction
	+ Where Can I Run Docker?
	+ Docker and Containerization on Linux
	+ Linux Kernel Features: cgroups and namespaces
	+ The Docker-Linux Kernel Interfaces
	+ Docker Containers vs. Traditional Virtualization
	+ Docker as Platform-as-a-Service
	+ Docker Integration
	+ Docker Services
	+ Docker Application Container Public Repository
	+ Competing Systems
	+ Docker Command-line
	+ Starting, Inspecting, and Stopping Docker Containers
* CI/CD with OpenShift, Jenkins, and Blue Ocean
	+ What is OpenShift?
	+ OpenShift CLI
	+ Jenkins Continuous Integration
	+ Jenkins Features
	+ Running Jenkins
	+ Downloading and Installing Jenkins
	+ Running Jenkins as a Stand-Alone Application
	+ Running Jenkins on an Application Server
	+ Installing Jenkins as a Windows Service
	+ Different types of Jenkins job
	+ Configuring Source Code Management (SCM)
	+ Working with Subversion
	+ Working with Git
	+ Build Triggers
	+ Schedule Build Jobs
	+ Polling the SCM
	+ Maven Build Steps
	+ Jenkins / OpenShift Pipeline
	+ Jenkins / OpenShift Pipeline Output
	+ Installing Jenkins Plugins
	+ The Blue Ocean Plugin
	+ Blue Ocean Plugin Features
	+ New modern user experience
	+ Advanced Pipeline visualizations with built-in failure diagnosis
	+ Branch and Pull Request awareness
	+ Personalized View
	+ OpenShift Pipeline Output
	+ Creating OpenShift Blue Ocean Pipeline
* Operational Readiness
	+ What is Operational Readiness?
	+ Telemetry
	+ End-to-end Requirements Traceability
	+ Log Strategy
	+ Monitoring Strategy
	+ Runbooks
* Application Modernization
	+ Typical App Modernization Projects
	+ Why Modernization?
	+ Goals for Application Modernization
	+ Modernization Process
	+ Modernization in a Nutshell
	+ Twelve-factor Applications
	+ Twelve Factors, Microservices, and App Modernization
	+ 12-Factor Microservice
	+ Monolithic revisited
	+ Monolithic vs. Microservices
	+ Maintaining State in App Modernization
	+ Cloud Service Fabric
* Introduction to Feign
	+ Feign - Annotations
	+ Creating a REST client with Feign
	+ Benefits of using Feign
	+ Feign – Default Beans
	+ FeignFeign – Simple Example
	+ Multiple Interfaces
	+ Ribbon
	+ Ribbon Load Balancing Architecture
	+ Using Ribbon
	+ Ribbon and Feign
	+ Hystrix
	+ Hystrix Dependency
	+ Using Hystrix
* Activiti Workflow
	+ Business Process Management
	+ Business Process Model and Notation
	+ BPMN – Elements
	+ BPMN 2.0
	+ What is Activiti?
	+ Using Spring Boot with Activiti
* Summary