

**Kafka for Application Developers**

**Course Number:** DVOP-166WA  
**Duration:** 2 days

**Overview**

Apache Kafka is a robust and scalable messaging platform over which microservices interact using an event-driven architecture.

This Kafka training course teaches attendees how to create and consume messages through Kafka’s APIs. Attendees learn how to use the two dominant components of the Kafka ecosystem to interact with Kafka: Kafka Connect for integration with third-party systems and Kafka Streams for creating serverless streaming applications.

**Prerequisites**

Prior day-to-day development experience with Java is assumed.

**Materials**

All Kafka training students receive comprehensive courseware.

**Software Needed on Each Student PC**

Attendees will not need to install any software on their computers 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 work well.

**Objectives**

* Understand the use of Kafka for high-performance messaging
* Identify the usages for Kafka in Microservices
* Explain the benefits of Kafka patterns
* Differentiate between messaging and message brokers
* Describe Kafka messaging environments
* Develop producers and consumers for Kafka
* Recognize how Kafka enables Cloud-native applications
* Summarize the characteristics and architecture of Kafka
* Demonstrate how to process messages with Kafka
* Design distributed high throughput systems based on Kafka
* Describe the built-in partitioning, replication, and inherent fault-tolerance of Kafka

**Outline**

* Introduction to Kafka
  + Messaging Architectures – What is Messaging?
  + Messaging Architectures – Steps to Messaging
  + Messaging Architectures – Messaging Models
  + What is Kafka?
  + Kafka Overview
  + Need for Kafka
  + When to Use Kafka?
  + Kafka Architecture
  + Core concepts in Kafka
  + Kafka Topic
  + Kafka Partitions
  + Kafka Producer
  + Kafka Consumer
  + Kafka Broker
  + Kafka Cluster
  + 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 Cluster Planning – Producer/Consumer Throughput
  + Kafka Cluster Planning – Number of Brokers (and ZooKeepers)
  + Kafka Cluster Planning – Sizing for Topics and Partitions
  + Kafka Cluster Planning – Sizing for Storage
  + 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
* Building Data Pipelines
  + What to Consider When Building Data Pipelines
  + Timeliness
  + Reliability
  + High and Varying Throughput
  + Data Formats
  + Transformations
  + Transformations - ELT
  + Security
  + Failure Handling
  + Agility and Coupling
  + Ad-hoc Pipelines
  + Metadata Loss
  + Extreme Processing
  + Kafka Connect vs. Producer and Consumer
* Integrating Kafka with Other Systems
  + Introduction to Kafka Integration
  + Kafka Connect
  + Running Kafka Connect Operating Modes
  + Key Configurations for Connect workers:
  + Kafka Connect API
  + Kafka Connect Example – File Source
  + Kafka Connect Example – File Sink
  + Kafka Connector Example – MySQL to Elasticsearch
  + Kafka Connector Example – MySQL to Elasticsearch (Contd.)
  + Write the data to Elasticsearch
  + Building Custom Connectors
  + Kafka Connect – Connectors
  + Kafka Connect - Tasks
  + Kafka Connect - Workers
  + Kafka Connect - Offset management
  + Alternatives to Kafka Connect
  + Introduction to Storm
  + Integrating Storm with Kafka
  + Integrating Storm with Kafka – Sample Code
  + Integrating Storm with Kafka
  + Integrating Hadoop with Kafka
  + Hadoop Consumers – Produce Topic
  + Hadoop Consumers – Fetch Generated Topic
  + Kafka at Uber
  + Kafka at LinkedIn – Core Kafka Services
  + Kafka at LinkedIn – Core Kafka Services (Contd.)
  + Kafka at LinkedIn – Libraries
  + Kafka at LinkedIn – Monitoring and Stream Processing
* Kafka and Schema Management
  + Evolving Schema
  + Protobuf (Protocol Buffers) Overview
  + Avro Overview
  + Managing Data Evolution Using Schemas
  + Confluent Platform
  + Confluent Schema Registry
  + Schema Change and Backward Compatibility
  + Collaborating over Schema Change
  + Handling Unreadable Messages
  + Deleting Data
  + Segregating Public and Private Topics
* Kafka Streams and KSQL
  + What Kafka can be used for?
  + What Exactly is Kafka?
  + The APIs for Stream Processing
  + Kafka: A Streaming Platform
  + What is KSQL?
  + Starting KSQL
  + Using the KSQL CLI
  + KSQL Data Types
  + Review the Structure of an Existing STREAM
  + Query the STREAM
  + KSQL Functions
  + Writing to a Topic
  + KSQL Table vs. Stream
  + KSQL JOIN
  + Windows in KSQL Queries
  + Miscellaneous KSQL Commands
* KSQL UDF and Deployment
  + KSQL Custom Functions
  + KSQL UDF/UDAF
  + Implement a Custom Function
  + Creating UDF and UDAF
  + UDFs and Null Handling
  + Sample UDF Class
  + Build Engine
  + UDAF Sample Class
  + Supported Types
  + Deploying Custom Functions
  + Using Custom Functions
* Conclusion