

Apache Kafka for Developers

Duration: 3 Days

Description

The "Kafka for Developers" training aims to teach software developers the basic concepts, structure and usage of Apache Kafka. This course teaches participants to understand Kafka, how to use it in their applications, and how to work with Kafka on real-world projects.

Audience

- Developers who want to build a streaming data application
- Teams getting started or working on Kafka based projects

Outline

Day 1: Introduction to Apache Kafka and Producer/Consumer API

Introduction to Apache Kafka

- Fundamentals of Kafka
- Kafka's Working Model (Pub/Sub)
- Kafka Components (Broker, Topic, Partition, Zookeeper)

Kafka Terminology

- Topic, Partition, Offset, Producer, Consumer
- Concepts of Consumer Group and Consumer Lag

First Application with Kafka (Java)

- Sending Messages with Kafka Producer API
- Synchronous and Asynchronous Message Delivery
- Error Handling with Callback
- Reading Messages with Kafka Consumer API
- Parallel Data Processing with Consumer Group

First Application with Kafka (Python)

- KafkaProducer and KafkaConsumer Libraries
- Simple Producer and Consumer Application
- Monitoring Consumer Lag and Managing Offsets

- Error Handling with Python

Day 2: Advanced API Usage and Data Serialization

Advanced Kafka Applications with Java

- Writing a Custom Partitioner
- Using Custom Serializer/Deserializer
- Message Serialization with Avro and Protobuf
- Producer and Consumer Configurations
- Performance Tuning for High Throughput

Advanced Kafka Applications with Python

- Writing Custom Serializer/Deserializer
- Message Serialization (JSON, Avro, Protobuf)
- Offset Management and Manual Commit
- Parallel Processing for Python Consumers

Kafka Streams API (Java)

- Real-Time Data Processing with Kafka Streams API
- Examples of Stream-Table Joins
- Stateless and Stateful Transformations
- Windowing Concepts

Kafka-Python Streams Usage

- Simple Stream Processing Applications
- Examples of Stream Aggregation and Filtering

Day 3: Application Development and Deployment

Real-Time Data Processing Project (Java)

- Reading and Processing Data from Multiple Topics
- Writing Processed Data to a Different Topic
- Using Transactional Producer

Real-Time Data Processing Project (Python)

- Reading Data from Different Sources
- Data Transformation and Writing to a New Topic
- Application Logging and Error Handling

Deploying Kafka Applications with Docker

- Docker Compose for Kafka, Zookeeper, and Applications

- Dockerizing Java and Python Applications
- Testing Scenarios on a Local Cluster

CI/CD for Kafka Applications

- Configuring Pipelines
- Testing Kafka-Based Applications
- Deployment Strategies

Security and Authorization

- Encryption with SSL/TLS
- Authentication with SASL/PLAIN
- Authorization with ACL

Prerequisites

Basic Java programming knowledge is required.