IBM Confluent Developer Skills for Building Apache Kafka

Duration: 3 Days Course Code: DEVKAFKAG

Overview:

Build, integrate, and architect real-time streaming applications using Apache Kafka and Confluent.

Through interactive lessons and real-world exercises, you'll gain end-to-end expertise across the entire streaming pipeline covering client development, stream processing, and enterprise data governance.
Several of the hands-on lab exercises in this course follow the story of building and upgrading a driver location app. Throughout the course concepts are applied directly to a working application. Exercises are available in Java, C# and Python. Exercises include: Working with Kafka command line tools Producing driver location data to Kafka and consuming that data in real-timeRefactoring the application to use Avro and Schema RegistryPulling the driver data into a Kafka Streams app to enrich itExtracting a table from an external database into Kafka using Kafka ConnectExperimenting with semantic partitioning

Updated 16/09/2026

Virtual Learning

This interactive training can be taken from any location, your office or home and is delivered by a trainer. This training does not have any delegates in the class with the instructor, since all delegates are virtually connected. Virtual delegates do not travel to this course, Global Knowledge will send you all the information needed before the start of the course and you can test the logins.

Target Audience:

Application Developers and Architects who want to write applications that interact with Apache Kafka.
The course treats Java as a first-class citizen, but students will derive value even if Java is not their primary programming language. C# and Python clients will also be used in some options for labs.

Objectives:

- The lessons and activities in this course enable participants to build the skills to:
- Write Producers and Consumers to send data to and read data from Kafka
- Create schemas, describe schema evolution, and integrate with Confluent Schema Registry
- Integrate Kafka with external systems using Kafka Connect
- Write streaming applications with Kafka Streams
- Make design decisions about acks, keys, partitions, batching, replication, and retention polices

Prerequisites:

Attendees should be familiar with developing professional apps in Java (preferred), C#, or Python.

Additionally, a working knowledge of the Apache Kafka architecture is required for this course.

Participants are required to provide a laptop computer with unobstructed internet access to fully participate in the class.

Testing and Certification

- *None*

Follow-on-Courses:

- *None recommended*

Content:

Introductory Concepts

- Configure your Kafka applications to connect to a Kafka cluster
- Distinguish between leaders and followers and work with replicas
- Explain what a segment is and explore retention
- Use the CLI to work with topics, producers, and consumers

Working with Producers

- Describe the work a producer performs, and the core components needed to produce messages
- Create producers and specify configuration properties
- Explain how to configure producers to know that Kafka receives messages
- Delve into how batching works and explore batching configurations
- Explore reacting to failed delivery and tuning producers with timeouts
- Use the APIs for Java, C#/.NET, or Python to create a Producer

Consumers, Groups, and Partitions

- Create and manage consumers and their property files
- Illustrate how consumer groups and partitions provide scalability and fault tolerance
- Explore managing consumer offsets
- Tune fetch requests
- Explain how consumer groups are managed and their benefits
- Compare and contrast group management strategies and when you might use each
- Use the API for Java, C#/.NET, or Python to create a Consumer

Schemas and the Confluent Schema Registry

- Describe Kafka schemas and how they work
- Write an Avro compatible schema and explore using Protobuf and JSON schemas
- Write schemas that can evolve
- Write and read messages using schema-enabled Kafka client applications
- Build an Avro Kafka producer or consumer in Python, Java or C# using Confluent Schema Registry

Streaming with Kafka Streams and Flink

- Develop an appreciation for what streaming applications can do for you back on the job
- Describe Kafka Streams and explore streams properties and topologies
- Compare and contrast streams and tables, and relate events in streams to records/messages in topics
- Write a Kafka Streams application using the Streams DSL (Domain-Specific Language)
- Detail Apache Flink's key capabilities and APIs, identifying their target use cases

Kafka Connect

- List the components of Kafka Connect and describe how they relate
- Set configurations for components of Kafka Connect
- Describe connect integration and how data flows between applications and Kafka
- Explore some use-cases where Kafka Connect makes development efficient
- Use Kafka Connect in conjunction with other tools to process data in motion in the most efficient way
- Create a Connector and import data from a database to a Kafka cluster

Design Decisions and Considerations

- Delve into how compaction affects consumer offsets
- Explore how consumers work with offsets in scenarios outside of normal processing behavior
- Understand how to manipulate offsets to deal with anomalies
- Evaluate decisions about consumer and partition counts and how they relate
- Address decisions that arise from default key-based partitioning and consider alternative partitioning strategies
- Configure producers to deliver messages without duplicates and with ordering guarantees
- List ways to manage large message sizes
- Describe how to work with messages in transactions and how Kafka enables transactions

Additional Information:

Official course book provided to participants

Further Information:

For More information, or to book your course, please call us on Head Office Tel.: +974 40316639

training@globalknowledge.qa

www.globalknowledge.com/en-qa/

Global Knowledge, Qatar Financial Center, Burj Doha,Level 21, P.O.Box 27110, West Bay, Doha, Qatar