TRAINING COURSE

Confluent Developer Skills for Apache Kafka®

Course 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
- Integrate Kafka with external systems using Kafka Connect
- Write streaming applications with Kafka Streams & ksqlDB
- Integrate a Kafka client application with Confluent Cloud

Hands-on Training

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-time
- Refactoring the application to use Avro and Schema Registry
- Pulling the driver data into a Kafka Streams app to enrich it
- Extracting a table from an external database into Kafka using Kafka Connect
- Exploring data using ksqlDB
- Experimenting with semantic partitioning

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, either through:

- Prior experience, or
- By completing training beforehand to ensure familiarity with the relevant concepts. Visit www.confluent.io/training to learn the fundamentals of data streaming and Apache Kafka.

Participants are required to provide a laptop computer with unobstructed internet access to fully participate in the class.
To evaluate your Kafka knowledge for this course, please complete the self-assessment: https://cnfl.io/fundamentals-quiz

To sign-up for one of our courses, visit us here.

Who Should Attend?

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.

Content	This course will enable your skills to:
Introductory Concepts	• Write code 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 • Using Avro, the API for Java, C#/.NET, or Python, write a schema-enabled producer or consumer that leverages the Confluent Schema Registry
Streaming and Kafka Streams	• Develop an appreciation for what streaming applications can do for you back on the job • Describe Kafka Streams and explore steams properties and topologies • Compare and contrast steams and tables, and relate events in streams to records/messages in topics • Write an application using the Streams DSL (Domain-Specific Language)
Introduction to Confluent ksqlDB	• Describe how Kafka Streams and ksqlDB relate • Explore the ksqlDB CLI • Use ksqlDB to filter and transform data • Compare and contrast types of ksqlDB queries • Leverage ksqlDB to perform time-based stream operations • Write a ksqlDB query that relates data between two streams or a stream and a table
Kafka Connect	• List some of 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

Content	This course will enable your skills to:
Design Decisions and Considerations	• Delve into how compaction affects consumer offsets • Explore how consumers work with offsets in scenarios outside of normal processing behavior and 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
Robust Development	• Compare and contrast error handling options with Kafka Connect, including the dead letter queue • Distinguish between various categories of testing • List considerations for stress and load test a Kafka system

Visit confluent.io/training for more information.