

Program szkolenia:

Apache Kafka - niezbędny w ujęciu dla programistów i programistek Java

Informacje:

Nazwa:	Apache Kafka - niezbędny w ujęciu dla programistów i programistek Java
Kod:	kafka-java
Kategoria:	Architektura Java
Odbiorcy:	
Czas trwania:	3 dni
Forma:	50% wykłady / 50% warsztaty

W trakcie tego praktycznego warsztatu zapoznamy się z narzędziami i bibliotekami, które pozwolą programistom Java w efektywny sposób implementować oraz testować systemy wykorzystujące klaster Kafka.

Przedstawione zostaną innowacyjne sposoby implementacji, w których Kafka to nie tylko brokera wymiany wiadomości, to też narzędzie do trwałego zapisywania i odczytywania danych, ich agregacji oraz strumieniowania. Omówimy sposoby integracji danych z klastra z popularnymi bazami danych oraz sposoby łączenia ze sobą strumieni danych.

W części warsztatowej położymy nacisk na testowanie automatyczne wytworzonego kodu, zgodność z wzorcami projektowymi oraz dobrymi praktykami wytwarzania oprogramowania.

Szkolenie to niezbędny programistyczny dla każdego dewelopera Java rozpoczynającego swoją przygodę z implementacją aplikacji posługujących się Kafka. Jest to zbiór najbardziej efektywnych praktyk zebranych na bazie doświadczenia ponad sześciu lat pracy z systemami wykorzystującymi Apache Kafka.

Szczegółowy program:

1. Architektura Apache Kafka

1.1. Wprowadzenie do komunikacji asynchronicznej

1.1.1. Komunikacja w monolicie vs mikrouslugi

1.1.2. Przykłady zastosowań komunikacji asynchronicznej

1.1.3. Systemy oparte o zdarzenia

1.2. Architektura Apache Kafka

1.2.1. Wprowadzenie do podstawowych pojęć

1.2.1.1. Topics, Partitions, Offsets

1.2.1.2. Brokers

1.2.1.3. Topics Replication

1.2.1.4. Producers

1.2.1.5. Consumers, Consumers Groups

1.2.1.6. Offsets

1.2.1.7. Acks

1.2.2. Co zapewnia Apache Kafka

1.2.3. Skąd bierze się wydajność?

1.3. Architektura Confluent Platform

1.3.1. Kafka Streams

1.3.2. Kafka Connect

1.3.3. KSQL

1.3.4. Rest Proxy

1.3.5. Schema Registry

1.4. Kafka CLI

1.4.1. Zarządzanie Topic'ami

1.4.2. Wysyłanie wiadomości

1.4.3. Odbieranie wiadomości

1.4.4. Monitorowanie i zarządzanie Consumer Groups

1.4.5. Zarządzanie Offset'ami

1.5. Natywny klient Java

1.5.1. Kafka Producer

1.5.2. Kafka Consumer

1.5.3. Zarządzanie wątkami

1.5.4. Kafka Admin

1.5.5. Transakcje

1.6. Testowanie operacji asynchronicznych

2. Wsparcie dla Apache Kafka w Spring Boot

2.1. Podstawowe operacje

2.1.1. Wysyłka wiadomości

2.1.2. Odbiór wiadomości

2.1.3. Zarządzanie potwierdzeniami

2.1.4. Transakcje

2.2. Prawidłowa konfiguracja aplikacji

2.2.1. Zarządzanie błędami

2.2.2. Serializacja i deserializacji

2.2.3. Nagłówki

2.3. Testowanie integracyjne z wykorzystaniem Spring Boot i TestContainers

2.4. Implementacja aplikacji biznesowej z wykorzystaniem Spring Boot i Apache Kafka

3. Kafka Streams, Kafka Connect

3.1. Kafka Streams

3.1.1. Architektura aplikacji opartych o Kafka Streams

3.1.1.1. Wątki

3.1.1.2. Lokalna baza danych

3.1.1.3. Compacted Topics

3.1.1.4. Gorące repliki

3.1.2. Analiza topologii strumieni

3.1.3. Serializacja i deserializacja wiadomości

3.1.4. Obsługa błędów

3.1.5. Operacje bezstanowe

3.1.6. Operacje stanowe

3.1.6.1. KTable

3.1.6.2. GlobalKTable

3.1.6.3. Filtrowanie

3.1.6.4. Agregowanie

3.1.6.5. Joins

3.1.6.6. Windowing

3.1.7. Interactive queries

3.1.7.1. KeyValueStores

3.1.8. Transformers

3.1.8.1. Przetwarzanie zgodnie z podejściem stateful record-by-record

3.1.9. Exactly once processing

3.1.10. Testowanie aplikacji opartych o Kafka Streams

3.2. Kafka Connect

3.2.1. Co zapewnia Kafka Connect

3.2.2. Konfiguracja i uruchomienie

3.2.3. Implementacja własnych Connectors

3.3. Implementacja aplikacji biznesowej z wykorzystaniem Kafka Streams, Kafka Connect i PostgreSQL