

Developing Event-Driven Applications with Apache Kafka and Red Hat AMQ Streams

Kód kurzu: AD482

Develop, scale, and troubleshoot event-driven microservice applications. Learn to use Kafka and AMQ Streams to design, develop, and test event-driven applications. Event-driven microservices scale globally, store and stream process data, and provide low-latency feedback to customers. This course is for application developers and is based on Red Hat AMQ Streams 1.8 and Red Hat OpenShift Container Platform 4.6.

| Pobočka | Dnů | Cena kurzu | ITB |
|------------|-----|------------|-----|
| Praha | 3 | 1 905 € | 0 |
| Bratislava | 3 | 1 905 € | 0 |

Uvedené ceny jsou bez DPH.

Termíny kurzu

| Datum | Dnů | Cena kurzu | Typ výuky | Jazyk výuky | Lokalita |
|-------|-----|------------|-----------|-------------|----------|
|-------|-----|------------|-----------|-------------|----------|

Uvedené ceny jsou bez DPH.

Pro koho je kurz určen

Application developers with microservice development experience.

Co Vás naučíme

- Describe the basics of Kafka and its architecture.
- Develop applications with the Kafka Streams API.
- Integrate applications with Kafka Connect.
- Capture data change with Debezium.
- Troubleshoot common application streaming issues.

Impact on the organization

Organizations are recognizing that traditional synchronous applications are not able to scale consistently and adjust to the massive amounts of data from customers while still meeting customers' expectations of immediate results. With event-driven applications using Kafka and AMQ Streams, organizations can expect to be able to globally scale their applications, store and stream process data, and provide feedback to customers with extremely low latency.

Impact of this training

As a result of attending this course, students will understand the architecture of Kafka and AMQ Streams and will be able to identify proper use cases for event-driven applications. In addition to learning the fundamental principles and features of Kafka and AMQ Streams, Students will learn how to design, develop, and test event-driven applications. Students should be able to demonstrate the following skills:

- Design, build, and use event-driven applications for relevant scenarios with standard patterns.
- Detect and react to data changes with Debezium to improve application performance.
- Troubleshoot common problems with event-driven applications.

Požadované vstupní znalosti

- Experience with microservice application development and design, such as DO378 or equivalent experience.
- OpenShift experience is recommended, but not required.

Studijní materiály

Studijní materiál Red Hat.

GOPAS Praha

Kodaňská 1441/46
101 00 Praha 10
Tel.: +420 234 064 900-3
info@gopas.cz

GOPAS Brno

Nové sady 996/25
602 00 Brno
Tel.: +420 542 422 111
info@gopas.cz

GOPAS Bratislava

Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 248 282 701-2
info@gopas.sk



Copyright © 2020 GOPAS, a.s.,
All rights reserved

Developing Event-Driven Applications with Apache Kafka and Red Hat AMQ Streams

Osnova kurzu

Designing Event-Driven Applications

Describe the principles of event-driven applications.

Introducing Kafka and AMQ Streams Concepts

Build applications with basic read-and-write messaging capabilities.

Building Applications with the Streams API

Leverage the Streams API to create data streaming applications.

Creating Asynchronous Services with Event Collaboration

Create and migrate to asynchronous services using the event collaboration pattern.

Integrating Data Systems with Kafka Connect

Connect data systems and react to data changes using Kafka Connect and Debezium.

Troubleshooting AMQ Streams Applications

Handle common problems in Kafka and AMQ Streams applications.

GOPAS Praha

Kodaňská 1441/46
101 00 Praha 10
Tel.: +420 234 064 900-3
info@gopas.cz

GOPAS Brno

Nové sady 996/25
602 00 Brno
Tel.: +420 542 422 111
info@gopas.cz

GOPAS Bratislava

Dr. Vladimíra Clementisa 10
Bratislava, 821 02
Tel.: +421 248 282 701-2
info@gopas.sk



Copyright © 2020 GOPAS, a.s.,
All rights reserved