CCDAK: Your Partner in Confluent CCDAK Exam Preparation with Free Demos and Updates

CCDAK Certification

CCDAK Exam Dumps - Confluent Certified Developer for Apache Kafka Certification Examination

Exam Code: CCDAK
Exam Name: Confluent Certified Developer for Apache Kafka Certification Examination
Certification Provider: Confluent Certification
Exam Name: Confluent Certified Developer

FOR MORE INFORMATION

https://dumpsarena.com/confluent-dumps/ccdak/

2026 Latest DumpsQuestion CCDAK PDF Dumps and CCDAK Exam Engine Free Share: https://drive.google.com/open?id=1tLAJudD2TdIyors_rkB4XhHsa3QcYkvS

To become more powerful and struggle for a new self, getting a professional CCDAK certification is the first step beyond all questions. We suggest you choose our CCDAK test prep ----an exam braindump leader in the field. Since we release the first set of the CCDAK quiz guide, we have won good response from our customers and until now---a decade later, our products have become more mature and win more recognition. And our CCDAK Exam Torrent will also be sold at a discount from time to time and many preferential activities are waiting for you.

The CCDAK certification exam consists of a series of multiple-choice questions that assess the candidate's knowledge in various aspects of Kafka development. CCDAK exam covers topics such as Kafka architecture, Kafka APIs, Kafka Connect, Kafka Streams, and Kafka Security. Candidates must pass the exam with a minimum score of 70% to earn the CCDAK certification. Confluent Certified Developer for Apache Kafka Certification Examination certification is valid for two years and can be renewed by passing the recertification exam. The CCDAK certification program is an excellent way for developers to demonstrate their Kafka development skills, increase their marketability, and advance their careers in the field of big data and streaming technologies.

Confluent Certified Developer for Apache Kafka (CCDAK) Certification Exam is a hands-on exam that tests developers' ability to work with Kafka and Confluent's platform. CCDAK Exam covers a wide range of topics related to Kafka and Confluent's tools, and passing the exam demonstrates a developer's ability to work with Kafka and Confluent's platform at a high level. The CCDAK Certification Exam is an important certification for developers who want to work with Kafka and Confluent's platform, and it can help advance their careers and increase their earning potential.

>> CCDAK Practice Guide <<

Valid Test CCDAK Testking, CCDAK Clear Exam

In the Confluent Certified Developer for Apache Kafka Certification Examination (CCDAK) Web-based Practice Test, you will get the CCDAK questions that are real and accurate. Furthermore, the CCDAK practice exam works smoothly on all operating systems including Mac, Linux, IOS, Android, and Windows. it is a browser-based Confluent Certified Developer for Apache Kafka Certification Examination (CCDAK) practice test software, there is no need for any specific software installation or additional plugins to function correctly.

The Confluent CCDAK exam is designed to test the candidate's knowledge in various areas such as Kafka architecture, deployment, Kafka configuration, message producers, message consumers, Kafka Streams, and Kafka Connect. CCDAK Exam consists of 60 multiple-choice questions and must be completed within 90 minutes.

Confluent Certified Developer for Apache Kafka Certification Examination Sample Questions (Q214-Q219):

NEW QUESTION # 214

Which message delivery semantic is guaranteed by Kafka Connect?

- A. Exactly once
- B. At least once
- C. At most once

Answer: B

NEW QUESTION # 215
A consumer application is using KafkaAvroDeserializer to deserialize Avro messages. What happens if message schema is not present in AvroDeserializer local cache?

- A. Throws SerializationException
- B. Fails silently
- C. Fetches schema from Schema Registry
- D. Throws DeserializationException

Answer: C

Explanation:
First local cache is checked for the message schema. In case of cache miss, schema is pulled from the schema registry. An exception will be thrown in the Schema Registry does not have the schema (which should never happen if you set it up properly)

NEW QUESTION # 216
You are writing a producer application and need to ensure proper delivery. You configure the producer with acks=all.
Which two actions should you take to ensure proper error handling?
(Select two.)

- A. Use a callback argument in producer.send() where you check delivery status.
- B. Check the value of ProducerRecord.status().
- C. Surround the call of producer.send() with a try/catch block to catch KafkaException.
- D. Check that producer.send() returned a RecordMetadata object and is not null.

Answer: A,C

Explanation:
For proper delivery handling with acks=all:
* Usecallbackto log or act on success/failure.
* Usetry/catchto handle synchronous exceptions like serialization errors or network failures.
FromKafka Producer Documentation:
"Errors can be caught either via the returned Future<RecordMetadata> or via the callback interface. For fatal errors, use a try/catch block around the send call." Option B is incorrect because send() returns a Future, not RecordMetadata directly.
Option D is invalid - ProducerRecord has no method called status().
Reference:Kafka Producer Error Handling and Callback APIs

NEW QUESTION # 217
There are five brokers in a cluster, a topic with 10 partitions and replication factor of 3, and a quota of producer_bytes_rate of 1 MB/sec has been specified for the client. What is the maximum throughput allowed for the client?

- A. 0.33 MB/s
- B. 10 MB/s
- C. 1 MB/s
- D. 5 MB/s

Answer: D

Explanation:
Each producer is allowed to produce @ 1MB/s to a broker. Max throughput 5 * 1MB, because we have 5 brokers.

NEW QUESTION # 218

What is a consequence of increasing the number of partitions in an existing Kafka topic?

- A. Consumers will need to process data from more partitions which will significantly increase consumer lag.
- B. Records with the same key could be located in different partitions.
- C. The acknowledgment process will increase latency for producers using acks=all.
- D. Existing data will be redistributed across the new number of partitions temporarily increasing cluster load.

Answer: A

Explanation:
Increasing partitions increasesparallelism, but also means:
* Consumers in a group may have to handlemore partitions, especially if the number of consumers is lower than the number of partitions.
* This can result inincreased lag, especially under high load.
FromKafka Topic Management Docs:
"Increasing the number of partitions increases consumer work, and if consumers can't keep up, lag can accumulate."
* A is false:existing data is not redistributed.
* B is false:records with the same key always map to the same partitionbased on hash.
* D is not directly impacted by the partition count.
Reference:Kafka Topic Management > Adding Partitions

NEW QUESTION # 219

......