



Free Questions for [CCDAK](#) by [certsdeals](#)

Shared by [Duran](#) on [24-05-2024](#)

For More Free Questions and Preparation Resources

[Check the Links on Last Page](#)

Question 1

Question Type: MultipleChoice

The rule "same key goes to the same partition" is true unless...

Options:

- A- the number of producer changes
- B- the number of kafka broker changes
- C- the number of partition changes
- D- the replication factor changes

Answer:

C

Explanation:

Increasing the number of partition causes new messages keys to get hashed differently, and breaks the guarantee 'same keys goes to the same partition'. Kafka logs are immutable and the previous messages are not re-shuffled.

Question 2

Question Type: MultipleChoice

A consumer starts and has `auto.offset.reset=none`, and the topic partition currently has data for offsets going from 45 to 2311. The consumer group has committed the offset 10 for the topic before. Where will the consumer read from?

Options:

- A- offset 45
- B- offset 10
- C- it will crash
- D- offset 2311

Answer:

C

Explanation:

auto.offset.reset=none means that the consumer will crash if the offsets it's recovering from have been deleted from Kafka, which is the case here, as $10 < 45$

Question 3

Question Type: MultipleChoice

A consumer wants to read messages from partitions 0 and 1 of a topic topic1. Code snippet is shown below.

```
consumer.subscribe(Arrays.asList("topic1"));
```

```
List pc = new ArrayList();
```

```
pc.add(new PartitionTopic("topic1", 0));
```

```
pc.add(new PartitionTopic("topic1", 1));
```

```
consumer.assign(pc);
```

Options:

A- This works fine. subscribe() will subscribe to the topic and assign() will assign partitions to the consumer.

B- Throws IllegalStateException

Answer:

B

Explanation:

subscribe() and assign() cannot be called by the same consumer, subscribe() is used to leverage the consumer group mechanism, while assign() is used to manually control partition assignment and reads assignment

Question 4

Question Type: MultipleChoice

Once sent to a topic, a message can be modified

Options:

A- No

B- Yes

Answer:

A

Explanation:

Kafka logs are append-only and the data is immutable

Question 5

Question Type: MultipleChoice

To prevent network-induced duplicates when producing to Kafka, I should use

Options:

A- max.in.flight.requests.per.connection=1

B- enable.idempotence=true

C- retries=200000

D- batch.size=1

Answer:

B

Explanation:

Producer idempotence helps prevent the network introduced duplicates. More details here <https://cwiki.apache.org/confluence/display/KAFKA/Idempotent+Producer>

Question 6

Question Type: MultipleChoice

How can you gracefully make a Kafka consumer to stop immediately polling data from Kafka and gracefully shut down a consumer application?

Options:

- A- Call consumer.wakeup() and catch a WakeUpException
- B- Call consumer.poll() in another thread
- C- Kill the consumer thread

Answer:

A

Explanation:

See <https://stackoverflow.com/a/37748336/3019499>

To Get Premium Files for CCDAK Visit

<https://www.p2pexams.com/products/ccdak>

For More Free Questions Visit

<https://www.p2pexams.com/confluent/pdf/ccdak>

