Apache Kafka with Spring Boot - 100 Interview Questions and Answers

Core Kafka Concepts (1-30)

1. **What is Apache Kafka?**
   - A distributed streaming platform used for building real-time data pipelines and applications.
2. **How does Kafka differ from traditional messaging systems?**
   - Kafka is more scalable, durable, and fault-tolerant.
3. **What are the main components of Kafka?**
   - Producer, Consumer, Broker, Topic, Partition, Zookeeper.
4. **What is a Kafka Broker?**
   - A Kafka server that stores data and serves clients.
5. **What is a Topic in Kafka?**
   - A category or feed to which records are published.
6. **What is a Kafka Partition?**
   - A log that allows Kafka to parallelize data.
7. **What is an Offset?**
   - A unique identifier for each message in a partition.
8. **How does Kafka ensure fault tolerance?**
   - Through replication of partitions across brokers.
9. **What is a Consumer Group?**
   - A group of consumers sharing the load of reading messages.
10. **What is ZooKeeper's role in Kafka?**
    - Coordinates brokers and manages metadata.
11. **How does Kafka achieve high throughput?**
    - Batching, compression, and efficient disk I/O.
12. **What is Kafka Retention Policy?**
    - Configurable period for which Kafka retains messages.
13. **What is a Kafka Leader and Follower?**
    - The leader handles all reads and writes, while followers replicate data.
14. **Explain Kafka replication.**
    - Partitions are replicated to multiple brokers for fault tolerance.
15. **What is ISR in Kafka?**
    - In-Sync Replica: set of replicas that are fully caught up with the leader.
16. **What is the role of Kafka Producer?**
    - Publishes records to one or more topics.
17. **What is the role of Kafka Consumer?**
    - Reads records from topics.
18. **How does Kafka ensure message ordering?**
    - Within a partition, messages are strictly ordered.
19. **How are messages distributed among partitions?**
    - Based on partition key or round-robin.
20. **What happens when a Kafka broker goes down?**
    - Another broker with replica becomes the leader.
21. **What is log compaction in Kafka?**

- Retains latest value for each key.

22. **What are Kafka serializers and deserializers?**
    - Convert objects to byte arrays and vice versa.
23. **How does Kafka handle schema evolution?**
    - Using schema registry and versioned schemas.
24. **What is Kafka Connect?**
    - Tool for scalable and reliable data import/export.
25. **Difference between Kafka and RabbitMQ?**
    - Kafka is distributed, log-based, better for large-scale data; RabbitMQ is message-based.
26. **Can Kafka lose messages?**
    - Rarely, unless not configured properly.
27. **How does Kafka handle message duplication?**
    - Idempotent producer and exactly-once semantics.
28. **What is Kafka throughput?**
    - Number of messages Kafka can process in a given time.
29. **What is Kafka latency?**
    - Time taken for a message to reach the consumer.
30. **How to monitor Kafka?**
    - Using tools like Kafka Manager, JMX, Prometheus.

Kafka with Spring Boot (31-70)

31. **How to produce messages using Spring Boot?**
    - Use KafkaTemplate to send messages.
32. **How to consume messages using Spring Boot?**
    - Use @KafkaListener annotation.
33. **How to configure Kafka in Spring Boot?**
    - Use application.yml or application.properties.
34. **What is KafkaTemplate?**
    - Helper class to send messages.
35. **What is @KafkaListener?**
    - Annotation to consume Kafka messages.
36. **How to send JSON data with Kafka?**
    - Configure JSON serializer/deserializer.
37. **How to create Kafka topic in Spring Boot?**
    - Use NewTopic bean configuration.
38. **How to configure consumer group in Spring Boot?**
    - Set spring.kafka.consumer.group-id property.
39. **What is spring.kafka.bootstrap-servers?**
    - Specifies Kafka server addresses.
40. **How to handle deserialization errors?**
    - Use ErrorHandlingDeserializer.
41. **How to implement retry mechanism in Spring Kafka?**
    - Use RetryTemplate or DeadLetterPublishingRecoverer.
42. **What is a dead-letter topic?**
    - Topic to which failed messages are routed.

43. **How to batch consume messages in Spring Kafka?**
    - Set containerFactory to support batch mode.
44. **What is ConcurrentKafkaListenerContainerFactory?**
    - Factory to configure Kafka listener containers.
45. **How to enable logging in Spring Kafka?**
    - Use logback or log4j configuration.
46. **Can you run Kafka locally in Spring Boot?**
    - Yes, using embedded Kafka or Docker.
47. **How to test Kafka consumers in Spring Boot?**
    - Use @EmbeddedKafka and Test Kafka setup.
48. **What is a Kafka Avro serializer?**
    - Serializes messages using Avro schemas.
49. **What is Schema Registry?**
    - Stores and retrieves Avro schemas.
50. **How to handle offsets manually?**
    - Disable auto-commit and commit manually.
51. **What is KafkaHeaderMapper?**
    - Maps headers between Kafka and message objects.
52. **How to use Kafka Streams in Spring Boot?**
    - Use spring-kafka-streams dependency and define stream processing beans.
53. **Difference between Kafka and Kafka Streams?**
    - Kafka is a messaging system; Streams is for processing data.
54. **What is the use of @SendTo in Kafka?**
    - Sends reply to another topic.
55. **How to consume from multiple topics?**
    - List multiple topics in @KafkaListener.
56. **How to handle exceptions globally?**
    - Use ErrorHandler bean.
57. **What is KafkaMessageListenerContainer?**
    - Container that handles Kafka listener lifecycle.
58. **What are record filters?**
    - Skip messages based on logic before processing.
59. **What is KafkaTransactionManager?**
    - Manages Kafka transactions.
60. **How to make producer transactional?**
    - Enable transactions and use KafkaTransactionManager.
61. **What is @KafkaHandler?**
    - Handles multiple types in a single listener.
62. **How to control concurrency in Kafka consumers?**
    - Set concurrency on container factory.
63. **How to configure Kafka SSL in Spring Boot?**
    - Set truststore and keystore properties.
64. **What is idempotent producer?**
    - Prevents message duplication.
65. **How to commit offset after processing?**

- Use acknowledgment.acknowledge().

66. **What is acknowledgment mode?**
    - Defines when offset is committed (manual, batch, etc).
67. **How to integrate Kafka with REST APIs?**
    - REST controller sends messages via KafkaTemplate.
68. **What is KafkaAdmin?**
    - Creates topics programmatically.
69. **How to pause and resume Kafka consumers?**
    - Use KafkaMessageListenerContainer methods.
70. **What are Kafka headers used for?**
    - Pass metadata with messages.

Advanced Kafka Topics (71-100)

71. **What is Kafka Streams API?**
    - A Java library for real-time processing.
72. **What is a KStream?**
    - Stream of records.
73. **What is a KTable?**
    - Table of changelogs.
74. **What is GlobalKTable?**
    - Replicated KTable across all instances.
75. **What is repartitioning in Kafka Streams?**
    - Re-distributes data to new partitions.
76. **What is windowing in Kafka Streams?**
    - Time-based data aggregation.
77. **What is a session window?**
    - Based on user activity/session.
78. **What is join in Kafka Streams?**
    - Combines data from different streams/tables.
79. **What is punctuator in Kafka Streams?**
    - Triggers actions periodically.
80. **How does state store work?**
    - Stores intermediate results.
81. **What are DSL and Processor API?**
    - High-level (DSL) and low-level (Processor) APIs.
82. **How does Kafka achieve exactly-once processing?**
    - Idempotent producer + transactional consumer.
83. **How to handle out-of-order messages?**
    - Use timestamps and windowing.
84. **What is watermarking?**
    - Helps with late-arriving data.
85. **What is changelog topic?**
    - Internal topic to back up state stores.
86. **What is standby replica in Kafka Streams?**
    - Hot backup of state store.
87. **What is Serde in Kafka Streams?**
    - Serializer/Deserializer wrapper.
88. **How to deploy Kafka Streams in Spring Boot?**
    - Use @Bean for KStream processing.
89. **What is the role of Kafka metrics?**
    - Monitor performance and health.
90. **How to handle Kafka rebalancing?**

- Use cooperative rebalancing or control with listeners.

91. **What is Kafka MirrorMaker?**
    - Tool for replicating data between clusters.
92. **What is Kafka lag?**
    - Difference between last produced and consumed offset.
93. **How to reduce consumer lag?**
    - Increase consumer threads or optimize processing.
94. **How to implement backpressure in Kafka?**
    - Control poll rate and processing speed.
95. **What is the difference between at-least-once and exactly-once delivery?**
    - At-least-once: no loss, possible duplicates; Exactly-once: no duplicates.
96. **What is compaction vs retention?**
    - Compaction: latest value per key; Retention: keeps messages for a time.
97. **How to encrypt Kafka messages?**
    - Use SSL or message-level encryption.
98. **What is SASL in Kafka?**
    - Authentication mechanism.
99. **How to audit Kafka messages?**
    - Use interceptors or log to audit topics.
100. **What tools can be used with Kafka?** - Confluent Control Center, Kafka Manager, Burrow, Prometheus.