|  |
| --- |
| 1. What is Kafka?  It is distributed event store and stream-processing platform. |
| What are the main Kafka architecture elements?  Topics  clusters  Producers  Consumers  Consumer groups  Clusters  Brokers  Partitions  Replicas  Leaders  followers |
| What is a Kafka broker?  Network of machines called brokers  Each brokers hosts some set of partitions and handles incoming request to write events to those partitions or read events from them.  Brokers also handle replication of partitions between each other. |
| How Kafka topics are organized? |
| What is the need in using zookeeper to run Kafka? |
| What is an in-sync replica? |
| What network protocol is used for client-to-server communication in Kafka? |
| Why Kafka doesn't support async calls for producer API? (provocative question) |
| Can we publish the messege without key? Could you describe how Kafka choose the partition (with and without key)? |
| What delivery models do you know? Describe them. |
| When can we use 'at most once' deleivery model?(use cases) |
| How would you implement exactly once processing? |
| What is the primarily unit of paralellism in Kafka? Explain how to scale out? |
| Which kafka component is responcibele for message storing? |
| How many brokers coould be run in 1 physical server? |
| It is known that for 1 partition only 1 consumer could be present. How then 2 applications can read 1 topic? |
| Could you explain what is log compaction and why we can need it? |
| Imagine one of consumer in consumer group died with OOM error. How newly created consumer knows the offset to start processing with? |
| How can we achieve global ordering in Kafka? |
| How can we find if all replicas are in sync? |
| What is Dead Letter Queue? Why can we need it? |