**Kafka**

* Kafka is an open-source distributed stream-processing software platform which is used to handle the real-time data storage. It works as a broker between two parties, i.e., a sender and a receiver. It can handle about trillions of data events in a day.
* In Kafka we have Producer and consumer . **producer** who publishes messages, and a receiver is known as a **consumer** who consumes that message by subscribing it.

* In Kafka we have Four Api : Producer ,consumer ,connector and stream Api.

1. **Producer API:** This API allows/permits an application to publish streams of records to one or more topics(like queues or table in db).
2. **Consumer API:** This API allows an application to subscribe one or more topics and process the stream of records produced to them.
3. **Streams API:** This API allows an application to effectively transform the input streams to the output streams. It permits an application to act as a stream processor which consumes an input stream from one or more topics and produce an output stream to one or more output topics.
4. **Connector API:** This API executes the reusable producer and consumer APIs with the existing data systems or applications.

**Topics**: First we need to create topic in Kafka using :

Topic refers to a category or a common name used to store and publish a particular stream of data.

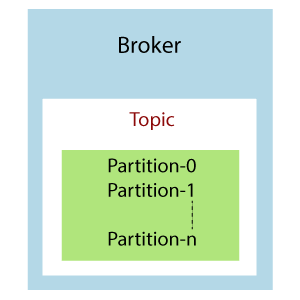
kafka-topics.bat --create --zookeeper localhost:2181 --replication-factor 1 --partitions 1 --topic test

Here test is topic name.

**Partitions**: A topic is split into several parts which are known as the partitions of the topic. While creating topic we need to specify partition and can be changes later as well. Each message can be stored in partition of topic in incremental id this is known as **offset value.**

**The data once written to a partition can never be changed. It is immutable. The offset value always remains in an incremental state, it never goes back to an empty space. Also, the data is kept in a partition for a limited time only.**

**Broker:** A  Kafka cluster is comprised of one or more servers which are known as brokers or Kafka brokers. A broker is a container that holds several topics with their multiple partitions



**Kafka Topic Replication:** Just to avoid loss of data from broker we used to create replication factor while creating topic*. A replication factor is the number of copies of data over multiple brokers.* It Should be always greater than 1.

Now we have multiple replicas of topic with partition so there will be confusion which will serve the client req .So, there is something called as leader and ISR(In -Sync -Replica). Leader only will serve client req ,if leader fails than in sync replica will become leader.

**Producer:** Producer write to Topic. Producer can write data to topic with or without keys .

* If Producers writes data with keys than data will be send to that partition of the topic.eg. So, data of Prod\_id\_1(say) will always be sent to partition 0 under Broker 1, and data of Prod\_id\_2 will always be in partition 1 under Broker 2.
* If Producers send data without keys than it will be distributed in partition using round-robin manner.

Once data is sent will get back with acknowledgement :

* acks=0 means no wait for acknowledge
* Acks=1 only leader acknowledge
* Acks=all leaders as well as replicas acknowledge.

**Consumers:** Consumer reads data from Kafka cluster via topic. Consumers read data in orderly manner . It means that the consumer is not supposed to read data from offset 1 before reading from offset 0. Also, a consumer can easily read data from multiple brokers at the same time.