1. Graphical user interface, application

   Description automatically generated
2. In the previous section, we talked about Kafka Storage Architecture.
3. Next logical step is to look at the Scalability side of Apache Kafka & how Kafka Cluster is formed.
4. 
5. Schematic

   Description automatically generated with medium confidence  
   Kafka Brokers are configured to form a cluster.
6. **Cluster**: A set of Brokers working together to share workload.
7. This is how Apache Kafka became distributed and scalable system.
8. You can start your cluster with a single broker in DEV environment and 3 brokers in production environment.  
   Later, as the workload grows, you can increase the cluster by adding more Brokers.
9. **However, the notion of clustering brings out two critical questions**.  
   Text

   Description automatically generated with medium confidence
   1. **Cluster Membership**: Who manages Cluster membership?
   2. **Administrative Tasks**: Who performs the routine administrative tasks in the cluster.
10. Diagram, schematic

    Description automatically generated  
    Let’s try to understand the question then will go deep into the answer.
    1. **Cluster Membership**:
       1. In a typical distributed system, there is a **Master Node** that maintains the list of active cluster members (nodes).
       2. The Master always knows the state of other member.
          1. How does Master Node know when
             1. a Broker is crashed.
             2. A new Broker has recently joined the Cluster.
    2. **Administrative Tasks**: It is also managed by the Master Node in the typical cluster environment.
       1. Let’s try to understand with an example.
       2. Suppose, a Broker is active and it is taking care of some responsibilities in the cluster.
       3. Suddenly, the Broker leaves the cluster and dies for some reason.
       4. At this stage, who will perform those responsibilities.
       5. We need someone to assign those responsibilities to some other Broker so that Cluster can perform its functions.