**1. Explain Kafka--message Queueing service**

Apache Kafka is a distributed messaging system designed to handle real-time data streams at scale. While traditionally categorized as a messaging queue, Kafka operates more as a distributed commit log. The Kafka functions as a message queueing service as follows

* Topics and Messages: Kafka organizes messages into topics, which act as logical channels or categories. Producers publish messages to these topics, and consumers subscribe to one or more topics to receive and process messages.
* Scalability and Fault Tolerance: Kafka is built for horizontal scalability by distributing data across multiple brokers (servers). Topics are partitioned across brokers, allowing Kafka to handle large volumes of data and provide fault tolerance. Each partition can be replicated across multiple brokers, ensuring that data is not lost even if some brokers fail.
* Publish-Subscribe Model: Producers publish messages to Kafka topics without necessarily knowing who the consumers are. Consumers can subscribe to specific topics or groups of topics to receive messages in real time.
* Retention and Durability: Kafka retains messages on disk for a configurable period, enabling consumers to fetch historical data or replay messages. This feature supports use cases such as auditing, analytics, and data integration.
* Consumer Groups: Consumers can be organized into consumer groups, where each group receives a copy of the messages in a topic. Kafka ensures that messages within a partition are delivered to only one consumer within a group, allowing for parallel processing and load balancing.
* High Throughput: Kafka is optimized for high throughput and low-latency message delivery. It can handle millions of messages per second across numerous topics and consumers.
* Integration Capabilities: Kafka integrates well with other systems in the data ecosystem, such as stream processing frameworks (e.g., Apache Flink, Apache Spark Streaming) and data storage systems (e.g., Hadoop, Elasticsearch), making it a versatile component in modern data architectures.

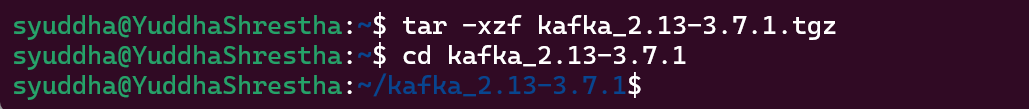
Hence, Kafka serves as a message queuing service by providing scalable, durable, and fault-tolerant message delivery for real-time data streams.

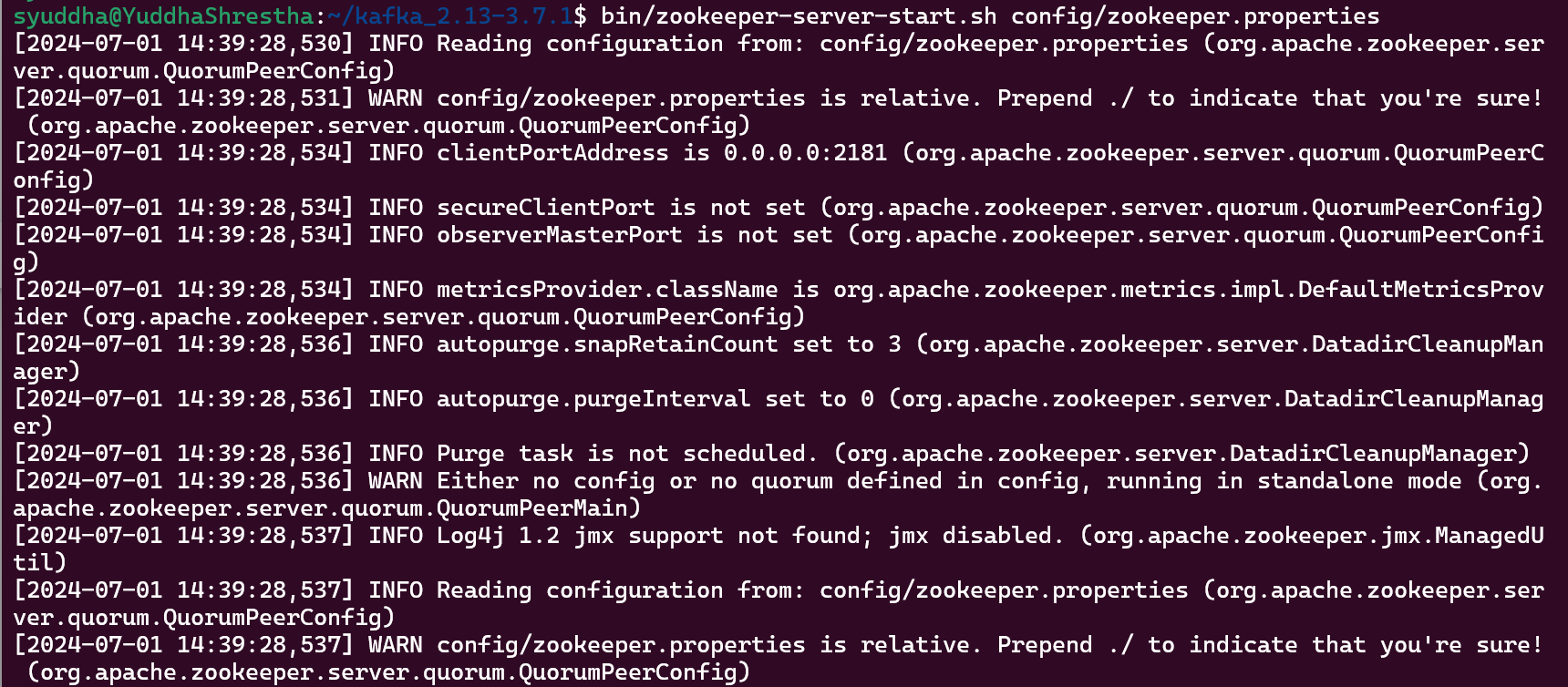
**2. Implement the Kafka installation and run Kafka topic commands**

Screenshots shows the implementation of Kafka installation and running Kafka topic commands as follows:

**A computer screen shot of a computer

Description automatically generated**

****

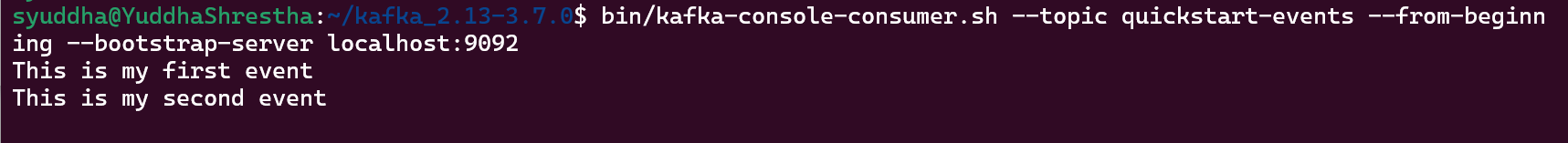
****

**A screenshot of a computer screen

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

****

**A screenshot of a computer screen

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

**A screenshot of a computer program

Description automatically generated**

**A computer screen with white text

Description automatically generated**

**A computer screen with white text

Description automatically generated**

**3. Run the pyspark kafka commands & others. Pyspark streaming command execution with kafka.**

Screenshots:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer code

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screen shot of a computer code

Description automatically generatedA screenshot of a computer program

Description automatically generatedA screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

https://databricks-prod-cloudfront.cloud.databricks.com/public/4027ec902e239c93eaaa8714f173bcfc/2205660161044287/2684060304601441/6764485214689999/latest.html