Students Prerequisites

- Basic understanding of distributed systems, messaging systems, and Linux/Unix commands.
- Knowledge of a programming language like Java or Python; ability to write scripts.
- Understanding of TCP/IP, HTTP, ports, and hostnames.
- Familiarity with cloud platforms (Azure, AWS, etc.) and deploying applications.
- Basic knowledge of SQL/NoSQL databases, data consistency, and partitioning.
- Experience with CLI tools, text editors, Git, and basic development tools.
- Awareness of event-driven architectures and Kafka alternatives like RabbitMQ.

Scope

- Practical exercises provided for CLI tools, topic creation, streaming, and AWS integrations.

Day 1 (4 Hours): Kafka Fundamentals & Basic Concepts

- **Introduction to Apache Kafka**
    - Overview of Kafka and its role in modern data architectures.
    - Real-World Use Case: Kafka in Event-Driven Architectures (e.g., fraud detection in financial systems).
    - **Optional**: Discuss Kafka's advantages over traditional messaging systems (e.g., RabbitMQ).
- **Kafka Core Concepts**
    - Topics, Partitions, and Offsets: Key concepts for organizing and accessing data.
    - Producers and Consumers: What they are and how they interact.
    - Data flow in Kafka: understanding producer-consumer interaction and offsets.
- **Kafka Architecture Overview**
    - Brokers, Replicas, Consumer Groups: Basic architecture and scaling.
    - Real-Time Use Case: Kafka in high-volume event logging (e.g., user behavior tracking in e-commerce systems).
- **Hands-On Exercise**:
    - Setting up a basic Kafka environment.
    - Simple producer-consumer setup using MSK.
- **Q&A & Discussion**:
    - Address any initial queries and discuss common Kafka-related challenges.

Day 2 (4 Hours): Advanced Producer and Consumer Concepts

- **Kafka Producers and Consumers Deep Dive**
    - Producer Configuration: High throughput and reliability settings.
    - Consumer Groups: How consumer groups enable horizontal scaling and load balancing.

- **Real-World Use Case**:
    - Log Aggregation: Building a data pipeline to send microservice logs to Kafka and process them in real-time.
    - **Optional**: Building data pipelines for real-time analytics.
- **Troubleshooting Kafka Producers and Consumers**
    - Common Issues: Message duplication, lost messages, and consumer lag.
    - Kafka lag
    - **Optional**: Strategies for dealing with consumer group rebalancing issues.
- **Hands-On Exercise**:
    - Debugging producer and consumer issues in a sandbox environment.
- **Q&A & Best Practices**:
    - Discuss best practices for configuring and scaling Kafka producers and consumers.

Day 3 (4 Hours): Kafka Installation on AWS (MSK) and Setup Troubleshooting

- **Kafka Installation and Setup on AWS MSK**
    - Benefits of using AWS MSK (Managed Streaming for Kafka).
    - **Optional**: Configuring MSK for different Kafka use cases (e.g., IoT data streaming, large-scale event processing).
- **Real-World Use Case**:
    - Implementing Kafka for real-time analytics with IoT sensor data (e.g., smart building temperature monitoring).
- **Troubleshooting Installation Issues**
    - Common setup problems (e.g., brokers not connecting, Kafka service failures).
    - **Optional**: Network configuration and security setup issues in MSK.
- **Hands-On Exercise**:
    - Setting up a basic Kafka cluster using MSK.
    - Troubleshooting and resolving installation issues.
- **Q&A & Discussion**:
    - Exploring real-world challenges during Kafka setup.

Day 4 (4 Hours): Kafka Integration with AWS Services (S3, Lambda, etc.)

- **Kafka Integration with AWS Services**
  - Using Kafka Connect to stream data to AWS S3.
  - Triggering AWS Lambda from Kafka to process real-time data.
  - **Optional**: Integrating Kafka with other AWS services like DynamoDB or Redshift.
- **Real-World Use Case**:
  - Log Processing: Building a pipeline where Kafka streams logs to S3, triggering Lambda functions for real-time processing.
  - **Optional**: Building a real-time recommendation engine using Kafka + Lambda.
- **Troubleshooting Kafka Integration**
  - Common issues with Kafka Connectors and AWS integration.
  - **Optional**: Addressing security misconfigurations and data delivery failures.
- **Hands-On Exercise**:
  - Implementing the log processing pipeline using Kafka and troubleshooting any integration issues.
- **Q&A & Best Practices**:
  - Discuss common integration pitfalls and strategies to avoid them.

Day 5 (4 Hours): Kafka Monitoring, Troubleshooting & Performance Optimization

- **Monitoring Kafka with AWS CloudWatch**
  - Setting up CloudWatch to monitor Kafka clusters and Kafka-related metrics.
  - **Optional**: Using CloudWatch Logs and Alarms for proactive monitoring.
- **Real-World Use Case**:
  - Monitoring Kafka in financial transactions processing systems.
  - Ensuring Kafka handles high throughput and does not fall behind or lose data.
- **Kafka Security Basics**
  - Encryption, authentication, and authorization for Kafka clusters on AWS.
- **Performance Optimization**:
  - Tuning producer and consumer configurations for higher throughput.
  - **Optional**: Kafka performance benchmarking and stress testing.
- **Troubleshooting Kafka Performance Issues**

- Common performance bottlenecks (e.g., network issues, disk I/O).

- **Hands-On Exercise**:
  - Monitoring Kafka clusters with CloudWatch and troubleshooting performance issues.

Day 6 (4 Hours): Kafka Scaling Strategies & Debugging Complex Issues

- **Scaling Kafka Clusters on AWS MSK**
  - Horizontal scaling of Kafka clusters to handle more producers and consumers.
  - **Optional**: Kafka replication and partition strategies for large-scale setups.
- **Kafka Backup and Recovery**
  - Setting up backup strategies for Kafka topics.
  - Ensuring data recovery and fault tolerance.
- **Debugging Complex Kafka Issues**
  - Handling data loss, message reordering, and network bottlenecks.
  - **Optional**: Kafka's internal mechanisms for fault tolerance and recovery.
- **Hands-On Exercise**:
  - Scaling Kafka clusters and setting up backup and recovery.
  - Troubleshooting Kafka issues in a pre-configured setup.

Day 7 (4 Hours): Advanced Kafka Streams and KSQL

- **Introduction to Kafka Streams**
  - Stream processing concepts: Stateless vs. Stateful operations.
  - **Optional**: Kafka Streams architecture and fault tolerance.
- **Real-Time Use Case**:
  - Implementing a real-time aggregation system (e.g., calculating moving averages of financial transactions).
- **Kafka Streams in Action**
  - Hands-on exercise building stream processing applications with Kafka Streams.
- **Introduction to KSQL**
  - Using KSQL for stream processing with SQL-like queries.
  - **Optional**: Real-time analytics with KSQL.

- **Hands-On Exercise**:
    - Implementing stream processing with Kafka Streams and KSQL.

Day 8 (4 Hours): Advanced Kafka Security & Compliance

- **Kafka Security Deep Dive**
    - Kerberos authentication and role-based access control (RBAC) in Kafka.
    - **Optional**: Best practices for securing Kafka in production.
- **Compliance & Auditing**
    - Implementing data encryption and logging for auditing purposes.
    - **Optional**: Ensuring GDPR and other compliance frameworks for Kafka.
- **Kafka Security Use Case**
    - Securing sensitive data while streaming through Kafka.
- **Hands-On Exercise**:
    - Setting up security and compliance features in Kafka.

Day 9 (4 Hours): Kafka Use Cases & Real-World Scenarios

- **Event Sourcing with Kafka**
    - Building an event-driven architecture using Kafka.
    - **Optional**: Event Sourcing and Command Query Responsibility Segregation (CQRS) with Kafka.
- **Real-Time Data Pipelines**
    - Building robust, real-time ETL pipelines using Kafka.
- **Kafka Use Cases**
    - Exploring different use cases such as fraud detection, IoT data processing, and financial transactions.
- **Group Case Study**:
    - Design a Kafka-based solution for a complex, real-time use case.

Day 10 (4 Hours): Capstone Project & Review

- **Capstone Project**:

    - Each participant will build a Kafka-based data pipeline or stream processing solution based on real-world requirements.
    - Participants will integrate concepts from the entire course into their project.
- **Q&A & Final Troubleshooting Session**
    - Address any lingering issues or challenges.
- **Review & Best Practices**:
    - Recap the key takeaways and Kafka best practices.
- **Certification & Closing Remarks**