

---

# System Requirements Specification Index

For

## Kafka with python

Temperature Monitoring for an Industrial  
Facility

Version 1.0

Problem Statement : Real-Time Temperature Monitoring for an Industrial Facility

Description : Use relevant methods operations to perform specified activities which are given in the instructions.

An industrial facility requires a system to continuously monitor the temperature across various zones (e.g., warehouses, manufacturing units, server rooms) to ensure optimal operating conditions and prevent equipment failures. The facility is equipped with IoT sensors that record temperature data in real time. The data needs to be processed, stored, and analyzed to trigger alerts if temperatures exceed predefined thresholds.

System Components:

1. **IoT Sensors:**

- **Purpose:** Measure the temperature at different locations in the facility.
- **Data:** Each sensor sends temperature readings every second to a central system.

2. **Kafka Producer (`produce_temperature_data`):**

- **Purpose:** The Kafka Producer reads temperature data from the IoT sensors and sends it to a Kafka topic (`temperature_topic`).
- **Functionality:** It packages each temperature reading as a message and publishes it to Kafka, enabling real-time data streaming.

3. **Kafka Consumer (`consume_temperature_data`):**

- **Purpose:** The Kafka Consumer reads the temperature data from the Kafka topic.
- **Functionality:** It processes the data to check if any temperature readings exceed predefined thresholds. If so, it triggers an alert system to notify maintenance teams.

4. **Data Processing Module (`start_producer, start_consumer`):**

- **Purpose:** To continuously produce and consume temperature data.
- **Functionality:**
  - The `start_producer` function initiates the producer to start reading from IoT sensors and sending data to the Kafka topic.
  - The `start_consumer` function starts the consumer to process the incoming data, analyze it, and generate alerts if necessary.

Workflow:

1. **Data Collection:**

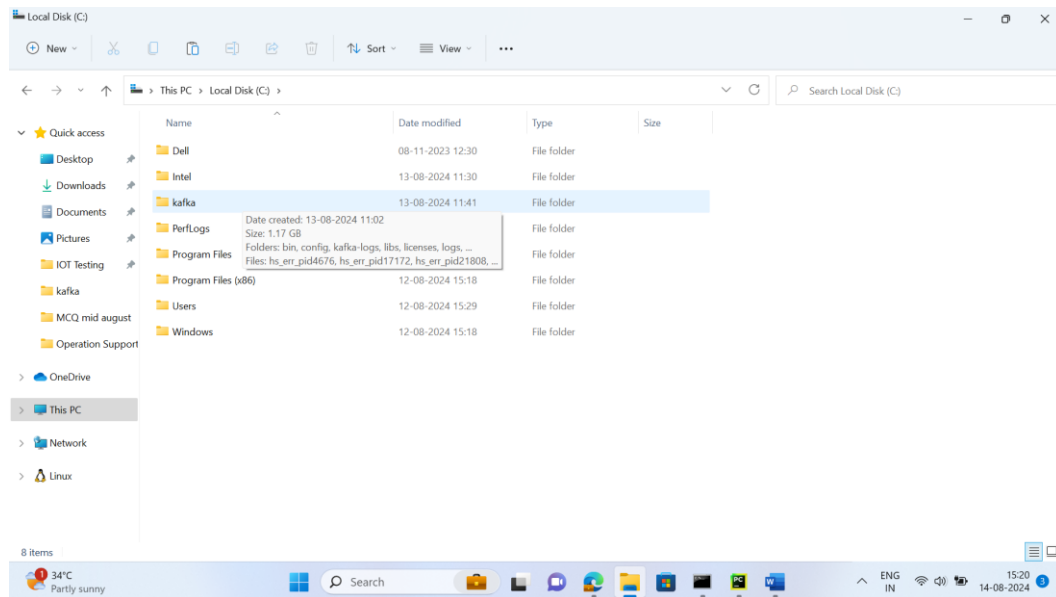
- IoT sensors deployed across the facility continuously measure the temperature and transmit this data to the Kafka Producer.

2. **Data Ingestion:**

- The Kafka Producer running in the system continuously reads these temperature readings and publishes them to the `temperature_topic` in the Kafka cluster.
- Each reading is sent as a message, ensuring that no data is lost and that all temperature data is available for processing.

## Kafka setup

You can see the kafka setup in the download folder



## 1. Extract Kafka:

- Extract the downloaded Kafka binaries into a directory of your choice (e.g., C:\kafka).

## 2. Set Up Zookeeper:

- Kafka uses Zookeeper to manage distributed brokers. Zookeeper is included in the Kafka distribution.

In the command prompt type the following command .

`.\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties`

```
C:\Windows\System32\cmd.exe - .\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties
Microsoft Windows [Version 10.0.22000.2538]
(c) Microsoft Corporation. All rights reserved.

C:\kafka>.\bin\windows\zookeeper-server-start.bat .\config\zookeeper.properties
[2024-08-13 11:37:25,658] INFO Reading configuration from: .\config\zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,191] INFO clientPortAddress is 0.0.0.0:2181 (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,191] INFO secureClientPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,191] INFO observerMasterPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,191] INFO metricsProvider.className is org.apache.zookeeper.metrics.impl.DefaultMetricsProvider (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,207] INFO autopurge.snapRetainCount set to 3 (org.apache.zookeeper.server.DataDirCleanupManager)
[2024-08-13 11:37:26,207] INFO autopurge.purgeInterval set to 0 (org.apache.zookeeper.server.DataDirCleanupManager)
[2024-08-13 11:37:26,207] INFO Purge task is not scheduled. (org.apache.zookeeper.server.DataDirCleanupManager)
[2024-08-13 11:37:26,207] WARN Either no config or no quorum defined in config, running in standalone mode (org.apache.zookeeper.server.quorum.QuorumPeerMain)
[2024-08-13 11:37:26,223] INFO Log4j 1.2 jmx support not found; jmx disabled. (org.apache.zookeeper.jmx.ManagedUtil)
[2024-08-13 11:37:26,223] INFO Reading configuration from: .\config\zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,238] INFO clientPortAddress is 0.0.0.0:2181 (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,238] INFO secureClientPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,238] INFO observerMasterPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,238] INFO metricsProvider.className is org.apache.zookeeper.metrics.impl.DefaultMetricsProvider (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2024-08-13 11:37:26,238] INFO Starting server (org.apache.zookeeper.server.ZooKeeperServerMain)
[2024-08-13 11:37:26,307] INFO ServerMetrics initialized with provider org.apache.zookeeper.metrics.impl.DefaultMetricsProvider@22e357dc (org.apache.zookeeper.server.ServerMetrics)
[2024-08-13 11:37:26,333] INFO ACL digest algorithm is: SHA1 (org.apache.zookeeper.server.auth.DigestAuthenticationProvider)
[2024-08-13 11:37:26,333] INFO zookeeper.DigestAuthenticationProvider.enabled = true (org.apache.zookeeper.server.auth.DigestAuthenticationProvider)
Running kafka PRODUCER and CONSUMER...
```

In an another command prompt

Start Kafka Broker:

`.\bin\windows\kafka-log-server-start.bat .\config\server.properties`

```
C:\Windows\System32\cmd.exe - .\bin\windows\kafka-server-start.bat .\config\server.properties
Microsoft Windows [Version 10.0.22000.2538]
(c) Microsoft Corporation. All rights reserved.

C:\kafka>.\bin\windows\kafka-server-start.bat .\config\server.properties
[2024-08-13 11:41:07,496] INFO Registered kafka:type=kafka.Log4jController MBean (kafka.utils.Log4jControllerRegistration$)
[2024-08-13 11:41:09,315] INFO Setting -D jdk.tls.rejectClientInitiatedRenegotiation=true to disable client-initiated TLS renegotiation (org.apache.zookeeper.common.X509Util)
[2024-08-13 11:41:09,407] INFO RemoteLogManagerConfig values:
    log.local.retention.bytes = -2
    log.local.retention.ms = -2
    remote.fetch.max.wait.ms = 500
    remote.log.index.file.cache.total.size.bytes = 1073741824
    remote.log.manager.copier.thread.pool.size = 10
    remote.log.manager.copy.max.bytes.per.second = 9223372036854775807
    remote.log.manager.copy.quota.window.num = 11
    remote.log.manager.copy.quota.window.size.seconds = 1
    remote.log.manager.expiration.thread.pool.size = 10
    remote.log.manager.fetch.max.bytes.per.second = 9223372036854775807
    remote.log.manager.fetch.quota.window.num = 11
    remote.log.manager.fetch.quota.window.size.seconds = 1
    remote.log.manager.task.interval.ms = 30000
    remote.log.manager.task.retry.backoff.max.ms = 30000
```

Question using the template provide your are requested to create python code

1. Write a function to produce temperature data
2. Write a function to consume temperature data
3. Write a function to fetch the data with the produced to consumer

### Execution Steps to Follow:

1. Use the same dataset as follows temperatures = [22.5, 23.0, 23.5, 24.0, 24.5]
2. Capture the data from producer to consumer
3. Perform all the methods
4. Upload the code to the Github
5. Submit the code the get the test auto graded

-----X-----