PES UNIVERSITY
100 feet Ring Road, BSK 3rd Stage
Bengaluru 560085
Department of Computer Science and Engineering

Department of Computer Science and Engineering
B. Tech. CSE - 6th Semester
Jan – May 2025

UE22CS343BB3
DATABASE TECHNOLOGIES (DBT)

PROJECT REPORT
on

Real-Time Patient Health Monitoring and Alert System

Submitted by : Team #: 363_368

Mujaseem D	PES1UG22CS363	6F
N Swetha	PES1UG22CS368	6F

Class of Prof. Dr.NagaSundari

Table of Contents

1.Introduction

Our project focuses on monitoring patient health data in real-time. We created a simple Python-based API to simulate patient vitals such as heart rate, temperature, blood pressure and SpO2 level. These vitals were streamed using Kafka, processed using Apache Spark, and stored in a MySQL database and further analysed it in the batch modes

Technologies Used :

- Apache Kafka (Streaming)
- Apache Spark Streaming (Real-time processing)
- MySQL (Data storage + batch processing)
- Zookeeper (Kafka coordination)
- Language: Python
- Platform: Ubuntu(wsl)

Project Workflow Overview:

API Server
↓
Simulated Patient Sensor Data → Kafka Producer
↓
Kafka Topics
↓
Spark Streaming Application
(Window-based real-time processing + alerts)
↓
Store into MySQL DB
(patient_vitals table for stream processing)
↓
Batch Mode Analysis from DB
Store into MySQL DB
(to see the alerts)

2. Installation of Software

Installations:

Python and pip

Used for writing Flask API, Kafka producer and consumer scripts and batch analysis scripts

Zookeeper and Kafka

Kafka is used for streaming data. Zookeeper is required for Kafka to run .
Installation of Kafka 3.5.1 (with Zookeeper support + .bat files)

MySQL Server and MySQL Connector(JDBC Driver)

Used to store alert data after processing.

Required Python Libraries

Installed flask to build API.

Database Setup:

Database : healthcare

```
mysql> desc patient_vitals
    -> ;
+---------------------------+----------+------+-----+---------+-------+
| Field                     | Type     | Null | Key | Default | Extra |
+---------------------------+----------+------+-----+---------+-------+
| patient_id                | int      | NO   | PRI | NULL    |       |
| heart_rate                | int      | YES  |     | NULL    |       |
| body_temperature          | double   | YES  |     | NULL    |       |
| blood_pressure_systolic   | int      | YES  |     | NULL    |       |
| blood_pressure_diastolic  | int      | YES  |     | NULL    |       |
| spo2_level                | int      | YES  |     | NULL    |       |
| timestamp                 | datetime | NO   | PRI | NULL    |       |
+---------------------------+----------+------+-----+---------+-------+
7 rows in set (0.11 sec)
```

Table: patient_vitals

Setup Phase

- Kafka and Zookeeper running in CMD

Start Zookeeper:

```
mujaseemd@MujaseemD:~$ ls
CC_E4_PES1UG22CS363              health_env                  producer.py                 stream.py
PES1UG22CS363                    kafka_2.13-3.5.1            run_script.sh               stream_processor.py
apache-zookeeper-3.8.1-bin.tar.gz kafka_2.13-3.5.1.tgz       spark                       venv
api_server.py                    kproducer.py                spark-3.5.5-bin-hadoop3.tgz zookeeper
checkpoints                      mysql-connector-j-8.3.0.jar spark_kafka_consumer.py
docker-compose.yml               postgresql-42.7.5.jar       spark_processing.py
mujaseemd@MujaseemD:~$ cd  kafka_2.13-3.5.1
mujaseemd@MujaseemD:~/kafka_2.13-3.5.1$ bin/zookeeper-server-start.sh config/zookeeper.properties
[2025-04-18 10:38:02,045] INFO Reading configuration from: config/zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2025-04-18 10:38:02,048] WARN config/zookeeper.properties is relative. Prepend ./ to indicate that you're sure! (org.apache.zookeeper.server.quorum.QuorumP
eerConfig)
[2025-04-18 10:38:02,052] INFO clientPortAddress is 0.0.0.0:2181 (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2025-04-18 10:38:02,052] INFO secureClientPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2025-04-18 10:38:02,052] INFO observerMasterPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2025-04-18 10:38:02,052] INFO metricsProvider.className is org.apache.zookeeper.metrics.impl.DefaultMetricsProvider (org.apache.zookeeper.server.quorum.Quo
rumPeerConfig)
[2025-04-18 10:38:02,058] INFO autopurge.snapRetainCount set to 3 (org.apache.zookeeper.server.DatadirCleanupManager)
[2025-04-18 10:38:02,058] INFO autopurge.purgeInterval set to 0 (org.apache.zookeeper.server.DatadirCleanupManager)
[2025-04-18 10:38:02,058] INFO Purge task is not scheduled. (org.apache.zookeeper.server.DatadirCleanupManager)
[2025-04-18 10:38:02,059] WARN Either no config or no quorum defined in config, running in standalone mode (org.apache.zookeeper.server.quorum.QuorumPeerMai
n)
[2025-04-18 10:38:02,061] INFO Log4j 1.2 jmx support not found; jmx disabled. (org.apache.zookeeper.jmx.ManagedUtil)
[2025-04-18 10:38:02,062] INFO Reading configuration from: config/zookeeper.properties (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2025-04-18 10:38:02,062] WARN config/zookeeper.properties is relative. Prepend ./ to indicate that you're sure! (org.apache.zookeeper.server.quorum.QuorumP
eerConfig)
[2025-04-18 10:38:02,063] INFO clientPortAddress is 0.0.0.0:2181 (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2025-04-18 10:38:02,063] INFO secureClientPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2025-04-18 10:38:02,063] INFO observerMasterPort is not set (org.apache.zookeeper.server.quorum.QuorumPeerConfig)
[2025-04-18 10:38:02,063] INFO metricsProvider.className is org.apache.zookeeper.metrics.impl.DefaultMetricsProvider (org.apache.zookeeper.server.quorum.Quo
rumPeerConfig)
[2025-04-18 10:38:02,063] INFO Starting server (org.apache.zookeeper.server.ZooKeeperServerMain)
[2025-04-18 10:38:02,086] INFO ServerMetrics initialized with provider org.apache.zookeeper.metrics.impl.DefaultMetricsProvider@149e0f5d (org.apache.zookeep
er.server.ServerMetrics)
[2025-04-18 10:38:02,093] INFO zookeeper.snapshot.trust.empty : false (org.apache.zookeeper.server.persistence.FileTxnSnapLog)
[2025-04-18 10:38:02,112] INFO  (org.apache.zookeeper.server.ZooKeeperServer)
[2025-04-18 10:38:02,112] INFO    ______                     _                                          (org.apache.zookeeper.server.ZooKeeperServer)
[2025-04-18 10:38:02,112] INFO   |___  /                    | |                                         (org.apache.zookeeper.server.ZooKeeperServer)
[2025-04-18 10:38:02,112] INFO      / /    ___     ___      | |  __   ___    ___   _ __     ___   _ __   (org.apache.zookeeper.server.ZooKeeperServer)
[2025-04-18 10:38:02,112] INFO     / /    / _ \   / _ \     | | / /  / _ \  / _ \ | '_ \   / _ \ | '__| (org.apache.zookeeper.server.ZooKeeperServer)
[2025-04-18 10:38:02,113] INFO    / /__  | (_) | | (_) |    |  <   |  __/ |  __/ | |_) | |  __/ | |    (org.apache.zookeeper.server.ZooKeeperServer)
```

Start Kafka Server:

```
mujaseemd@MujaseemD:~/kafka_2.13-3.5.1$ bin/kafka-server-start.sh config/server.properties
[2025-04-18 12:43:43,341] INFO Registered kafka:type=kafka.Log4jController MBean (kafka.utils.Log4jControllerRegistration$)
[2025-04-18 12:43:43,563] INFO Setting -D jdk.tls.rejectClientInitiatedRenegotiation=true to disable client-initiated TLS renegotiation (org.apache.zookeeper.common.X509Util)
[2025-04-18 12:43:43,631] INFO Registered signal handlers for TERM, INT, HUP (org.apache.kafka.common.utils.LoggingSignalHandler)
[2025-04-18 12:43:43,634] INFO starting (kafka.server.KafkaServer)
[2025-04-18 12:43:43,634] INFO Connecting to zookeeper on localhost:2181 (kafka.server.KafkaServer)
[2025-04-18 12:43:43,652] INFO [ZooKeeperClient Kafka server] Initializing a new session to localhost:2181. (kafka.zookeeper.ZooKeeperClient)
[2025-04-18 12:43:43,655] INFO Client environment:zookeeper.version=3.6.4--d65253dcf68e9097c6e95a126463fd5fdeb4521c, built on 12/18/2022 18:10 GMT (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,655] INFO Client environment:host.name=MujaseemD. (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,655] INFO Client environment:java.version=21.0.6 (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,655] INFO Client environment:java.vendor=Ubuntu (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,655] INFO Client environment:java.home=/usr/lib/jvm/java-21-openjdk-amd64 (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,655] INFO Client environment:java.class.path=/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/activation-1.1.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/aopalliance-repackaged-2.6.1.jar:/home/mujaseemd/kafka_2.1
3-3.5.1/bin/../libs/argparse4j-0.7.0.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/audience-annotations-0.13.0.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/commons-cli-1.4.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/commons
-lang3-3.8.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/connect-api-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/connect-basic-auth-extension-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/connect-json-3.5.1.jar
:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/connect-mirror-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/connect-mirror-client-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/connect-runtime-3.5.1.jar:/home/mujaseemd/
kafka_2.13-3.5.1/bin/../libs/connect-transforms-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/hk2-api-2.6.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/hk2-locator-2.6.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/hk
2-utils-2.6.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jackson-annotations-2.13.5.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jackson-core-2.13.5.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jackson-databind-2.13.5.jar
:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jackson-dataformat-csv-2.13.5.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jackson-datatype-jdk8-2.13.5.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jackson-jaxrs-base-2.13.5.jar:/h
ome/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jackson-jaxrs-json-provider-2.13.5.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jackson-module-jaxb-annotations-2.13.5.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jackson-module-scala
_2.13-2.13.5.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jakarta.activation-api-1.2.2.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jakarta.annotation-api-1.3.5.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jakarta.inject-2.
6.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jakarta.validation-api-2.0.2.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jakarta.ws.rs-api-2.1.6.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jakarta.xml.bind-api-2.3.3.jar:
/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/javassist-3.29.2-GA.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/javax.activation-api-1.2.0.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/javax.annotation-api-1.3.2.jar:/home/mujaseem
d/kafka_2.13-3.5.1/bin/../libs/javax.servlet-api-3.1.0.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/javax.ws.rs-api-2.1.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jaxb-api-2.3.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../l
ibs/jersey-client-2.39.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jersey-common-2.39.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jersey-container-servlet-2.39.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jersey-con
tainer-servlet-core-2.39.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jersey-hk2-2.39.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jersey-server-2.39.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-client-9.4.51.v2
0230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-continuation-9.4.51.v20230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-http-9.4.51.v20230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-io-9.4.51.
v20230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-security-9.4.51.v20230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-server-9.4.51.v20230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-servlet-9.
4.51.v20230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-servlets-9.4.51.v20230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-util-9.4.51.v20230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jetty-util-aj
ax-9.4.51.v20230217.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jline-3.22.0.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jopt-simple-5.0.4.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/jose4j-0.9.3.jar:/home/mujaseemd/kafk
a_2.13-3.5.1/bin/../libs/kafka-clients-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-group-coordinator-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-log4j-appender-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.
1/bin/../libs/kafka-metadata-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-raft-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-server-common-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-shel
l-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-storage-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-storage-api-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-streams-3.5.1.jar:/home/mujase
emd/kafka_2.13-3.5.1/bin/../libs/kafka-streams-examples-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-streams-scala_2.13-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-streams-test-utils-3.5.1.jar:/home/muj
aseemd/kafka_2.13-3.5.1/bin/../libs/kafka-tools-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka-tools-api-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/kafka_2.13-3.5.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../
libs/lz4-java-1.8.0.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/maven-artifact-3.8.8.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/metrics-core-2.2.0.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/metrics-core-4.1.12.1.jar:/h
ome/mujaseemd/kafka_2.13-3.5.1/bin/../libs/netty-buffer-4.1.94.Final.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/netty-codec-4.1.94.Final.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/netty-common-4.1.94.Final.jar:/home/mujasee
md/kafka_2.13-3.5.1/bin/../libs/netty-handler-4.1.94.Final.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/netty-resolver-4.1.94.Final.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/netty-transport-4.1.94.Final.jar:/home/mujaseemd/k
afka_2.13-3.5.1/bin/../libs/netty-transport-classes-epoll-4.1.94.Final.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/netty-transport-native-epoll-4.1.94.Final.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/netty-transport-native-u
nix-common-4.1.94.Final.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/osgi-resource-locator-1.0.3.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/paranamer-2.8.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/plexus-utils-3.3.1.jar
:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/reflections-0.9.12.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/reload4j-1.2.25.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/rocksdbjni-7.1.2.jar:/home/mujaseemd/kafka_2.13-3.5.1/bi
n/../libs/scala-collection-compat_2.13-2.10.0.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/scala-java8-compat_2.13-1.0.2.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/scala-library-2.13.10.jar:/home/mujaseemd/kafka_2.13-3.5.1/bi
n/../libs/scala-logging_2.13-3.9.4.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/scala-reflect-2.13.10.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/slf4j-api-1.7.36.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/slf4j-reload4j
-1.7.36.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/snappy-java-1.1.10.1.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/swagger-annotations-2.2.8.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/trogdor-3.5.1.jar:/home/mujaseemd
/kafka_2.13-3.5.1/bin/../libs/zookeeper-3.6.4.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/zookeeper-jute-3.6.4.jar:/home/mujaseemd/kafka_2.13-3.5.1/bin/../libs/zstd-jni-1.5.5-1.jar (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:java.library.path=/usr/java/packages/lib:/usr/lib/x86_64-linux-gnu/jni:/lib/x86_64-linux-gnu:/usr/lib/x86_64-linux-gnu:/usr/lib/jni:/lib:/usr/lib (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:java.io.tmpdir=/tmp (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:java.compiler=<NA> (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:os.name=Linux (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:os.arch=amd64 (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:os.version=5.15.167.4-microsoft-standard-WSL2 (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:user.name=mujaseemd (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:user.home=/home/mujaseemd (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:user.dir=/home/mujaseemd/kafka_2.13-3.5.1 (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:os.memory.free=983MB (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:os.memory.max=1024MB (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,657] INFO Client environment:os.memory.total=1024MB (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,659] INFO Initiating client connection, connectString=localhost:2181 sessionTimeout=18000 watcher=kafka.zookeeper.ZooKeeperClient$ZooKeeperClientWatcher$@4e31276e (org.apache.zookeeper.ZooKeeper)
[2025-04-18 12:43:43,663] INFO jute.maxbuffer value is 4194304 Bytes (org.apache.zookeeper.ClientCnxnSocket)
[2025-04-18 12:43:43,669] INFO zookeeper.request.timeout value is 0. Feature enabled=false (org.apache.zookeeper.ClientCnxn)
[2025-04-18 12:43:43,671] INFO [ZooKeeperClient Kafka server] Waiting until connected. (kafka.zookeeper.ZooKeeperClient)
[2025-04-18 12:43:43,672] INFO Opening socket connection to server localhost/127.0.0.1:2181. (org.apache.zookeeper.ClientCnxn)
[2025-04-18 12:43:43,674] INFO Socket connection established, initiating session, client: /127.0.0.1:40598, server: localhost/127.0.0.1:2181 (org.apache.zookeeper.ClientCnxn)
[2025-04-18 12:43:43,696] INFO Session establishment complete on server localhost/127.0.0.1:2181, session id = 0x100085099850000, negotiated timeout = 18000 (org.apache.zookeeper.ClientCnxn)
```

- Kafka topic creation
  We created 2 kafka topics -
  health_data_source1
  health_data_source2
  health_data_source3

Python dependencies;

```
mujaseemd@MujaseemD:~$ pip install flask
Collecting flask
  Downloading flask-3.0.3-py3-none-any.whl.metadata (3.2 kB)
Collecting Werkzeug>=3.0.0 (from flask)
  Downloading werkzeug-3.0.6-py3-none-any.whl.metadata (3.7 kB)
Collecting Jinja2>=3.1.2 (from flask)
  Using cached jinja2-3.1.6-py3-none-any.whl.metadata (2.9 kB)
Collecting itsdangerous>=2.1.2 (from flask)
  Using cached itsdangerous-2.2.0-py3-none-any.whl.metadata (1.9 kB)
Collecting click>=8.1.3 (from flask)
  Using cached click-8.1.8-py3-none-any.whl.metadata (2.3 kB)
Collecting blinker>=1.6.2 (from flask)
  Downloading blinker-1.8.2-py3-none-any.whl.metadata (1.6 kB)
Collecting importlib-metadata>=3.6.0 (from flask)
  Downloading importlib_metadata-8.5.0-py3-none-any.whl.metadata (4.8 kB)
Collecting zipp>=3.20 (from importlib-metadata>=3.6.0->flask)
  Downloading zipp-3.20.2-py3-none-any.whl.metadata (3.7 kB)
Collecting MarkupSafe>=2.0 (from Jinja2>=3.1.2->flask)
  Downloading MarkupSafe-2.1.5-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (3.0 kB)
Downloading flask-3.0.3-py3-none-any.whl (101 kB)
Downloading blinker-1.8.2-py3-none-any.whl (9.5 kB)
Using cached click-8.1.8-py3-none-any.whl (98 kB)
Downloading importlib_metadata-8.5.0-py3-none-any.whl (26 kB)
Using cached itsdangerous-2.2.0-py3-none-any.whl (16 kB)
Using cached jinja2-3.1.6-py3-none-any.whl (134 kB)
Downloading werkzeug-3.0.6-py3-none-any.whl (227 kB)
Downloading MarkupSafe-2.1.5-cp38-cp38-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (26 kB)
Downloading zipp-3.20.2-py3-none-any.whl (9.2 kB)
Installing collected packages: zipp, MarkupSafe, itsdangerous, click, blinker, Werkzeug, Jinja2, importlib-metadata, fla
sk
```

```
Successfully built pyspark
Installing collected packages: py4j, kafka-python, findspark, pyspark, pymysql
Successfully installed findspark-2.0.1 kafka-python-2.1.5 py4j-0.10.9.7 pymysql-1.1.1 pyspark-3.5.5
```

1. Input Data

a.Source

Created a simple API using Flask application

b.Description

This API generates random patient vital signs that include:

- patient_id
- heart_rate
- body_temperature
- blood_pressure_systolic
- blood_pressure_diastolic
- spo2_level
- timestamp

2. Streaming Mode Experiment

a.Description

We first set up a real-time patient health monitoring using Apache Kafka and Spark Streamin.We created z simple flask-based API to simulate patient data that generated values like heart rate,temperature,blood pressure and SpO2 level every 2 seconds.
The Kafka producer continuously sends this data to a Kafka topic – health_data
The spark streaming producer received the patient data from Kafka ,checked for any abnormal values and stored them into the table named patient_vitals in the MySQL database . Alerts were triggered when heart rate is above 90 or when SpO2 level dropped below 90. These alerts were automatically stored in the database as soon as they were detected

b.Windows

We simulated a streaming window of 2 seconds, i.e., new data from the API was generated and sent to Kafka every 2 seconds.

c.Results

Running API server file:

```
mujaseemd@MujaseemD:~$ python api_server.py
 * Serving Flask app 'api_server'
 * Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI server instead.
 * Running on http://127.0.0.1:5000
Press CTRL+C to quit
127.0.0.1 - - [18/Apr/2025 10:49:59] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:01] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:03] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:05] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:07] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:09] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:11] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:13] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:16] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:18] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:20] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:22] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:24] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:26] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:28] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:30] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:32] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:34] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:36] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:38] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:40] "GET /patient_vitals HTTP/1.1" 200 -
127.0.0.1 - - [18/Apr/2025 10:50:42] "GET /patient_vitals HTTP/1.1" 200 -
```

Running multi_p.py file:

It streams live patient data every 2 seconds

Code:

```
  GNU nano 7.2
import requests
import json
import time
from kafka import KafkaProducer

producer = KafkaProducer(
    bootstrap_servers='localhost:9092',
    value_serializer=lambda v: json.dumps(v).encode('utf-8')
)

# Map of API URLs to their respective Kafka topics
API_TOPIC_MAP = {
    "http://localhost:5000/source1": "health_data_source1",
    "http://localhost:5000/source2": "health_data_source2",
    "http://localhost:5000/source3": "health_data_source3"
}

while True:
    for url, topic in API_TOPIC_MAP.items():
        try:
            response = requests.get(url)
            if response.status_code == 200:
                data = response.json()
                print(f"Sending from {data['source']} to topic {topic}: {data}")
                producer.send(topic, data)
            else:
                print(f"API error from {url}: {response.status_code}")
        except Exception as e:
            print(f"Error fetching from {url}: {e}")
        time.sleep(1)
```

Output:

```
mujaseemd@MujaseemD:~$ python multi_p.py
Sending from Device_A to topic health_data_source1: {'blood_pressure_diastolic': 81, 'blood_pressure_systolic': 126, 'body_temperature': 37.5, '
heart_rate': 85, 'patient_id': 1002, 'source': 'Device_A', 'spo2_level': 91, 'timestamp': '2025-04-22T10:50:53.520838'}
Sending from Device_B to topic health_data_source2: {'blood_pressure_diastolic': 81, 'blood_pressure_systolic': 137, 'body_temperature': 36.4, '
heart_rate': 97, 'patient_id': 2002, 'source': 'Device_B', 'spo2_level': 100, 'timestamp': '2025-04-22T10:50:54.624450'}
Sending from Device_C to topic health_data_source3: {'blood_pressure_diastolic': 90, 'blood_pressure_systolic': 121, 'body_temperature': 38.2, '
heart_rate': 82, 'patient_id': 3001, 'source': 'Device_C', 'spo2_level': 98, 'timestamp': '2025-04-22T10:50:55.643990'}
Sending from Device_A to topic health_data_source1: {'blood_pressure_diastolic': 69, 'blood_pressure_systolic': 124, 'body_temperature': 37.2, '
heart_rate': 99, 'patient_id': 1003, 'source': 'Device_A', 'spo2_level': 100, 'timestamp': '2025-04-22T10:50:56.652561'}
Sending from Device_B to topic health_data_source2: {'blood_pressure_diastolic': 68, 'blood_pressure_systolic': 140, 'body_temperature': 37.7, '
heart_rate': 69, 'patient_id': 2002, 'source': 'Device_B', 'spo2_level': 91, 'timestamp': '2025-04-22T10:50:57.658032'}
Sending from Device_C to topic health_data_source3: {'blood_pressure_diastolic': 85, 'blood_pressure_systolic': 129, 'body_temperature': 37.9, '
heart_rate': 67, 'patient_id': 3002, 'source': 'Device_C', 'spo2_level': 99, 'timestamp': '2025-04-22T10:50:58.663117'}
Sending from Device_A to topic health_data_source1: {'blood_pressure_diastolic': 63, 'blood_pressure_systolic': 138, 'body_temperature': 36.9, '
heart_rate': 94, 'patient_id': 1001, 'source': 'Device_A', 'spo2_level': 92, 'timestamp': '2025-04-22T10:50:59.668393'}
Sending from Device_B to topic health_data_source2: {'blood_pressure_diastolic': 66, 'blood_pressure_systolic': 133, 'body_temperature': 36.9, '
heart_rate': 99, 'patient_id': 2001, 'source': 'Device_B', 'spo2_level': 95, 'timestamp': '2025-04-22T10:51:00.674683'}
Sending from Device_C to topic health_data_source3: {'blood_pressure_diastolic': 79, 'blood_pressure_systolic': 125, 'body_temperature': 36.2, '
heart_rate': 60, 'patient_id': 3003, 'source': 'Device_C', 'spo2_level': 99, 'timestamp': '2025-04-22T10:51:01.681361'}
Sending from Device_A to topic health_data_source1: {'blood_pressure_diastolic': 81, 'blood_pressure_systolic': 113, 'body_temperature': 38.3, '
heart_rate': 72, 'patient_id': 1002, 'source': 'Device_A', 'spo2_level': 94, 'timestamp': '2025-04-22T10:51:02.687126'}
Sending from Device_B to topic health_data_source2: {'blood_pressure_diastolic': 60, 'blood_pressure_systolic': 128, 'body_temperature': 38.1, '
heart_rate': 70, 'patient_id': 2001, 'source': 'Device_B', 'spo2_level': 94, 'timestamp': '2025-04-22T10:51:03.692367'}
Sending from Device_C to topic health_data_source3: {'blood_pressure_diastolic': 66, 'blood_pressure_systolic': 100, 'body_temperature': 36.5, '
heart_rate': 92, 'patient_id': 3001, 'source': 'Device_C', 'spo2_level': 98, 'timestamp': '2025-04-22T10:51:04.698387'}
```

Running dbt.py file:

It consumed the data and if any alerts were found , they were inserted into the patient_vitals table

in MySQL

Code:

```
  GNU nano 7.2                                                                                                dbt.py
from pyspark.sql import SparkSession
from pyspark.sql.functions import col, from_json
from pyspark.sql.types import StructType, StructField, IntegerType, DoubleType, StringType

# Initialize Spark session
spark = SparkSession.builder \
    .appName("PatientMonitoringConsumer") \
    .config("spark.jars.packages", "org.apache.spark:spark-sql-kafka-0-10_2.12:3.5.0," + "mysql:mysql-connector-java:8.0.26") \
    .getOrCreate()

# Define the schema for incoming data
schema = StructType([
    StructField("patient_id", IntegerType(), True),
    StructField("heart_rate", IntegerType(), True),
    StructField("body_temperature", DoubleType(), True),
    StructField("blood_pressure_systolic", IntegerType(), True),
    StructField("blood_pressure_diastolic", IntegerType(), True),
    StructField("spo2_level", IntegerType(), True),
    StructField("timestamp", StringType(), True)
])

# Read streaming data from Kafka topics 'health_data_source1', 'health_data_source2', 'health_data_source3'
df = spark.readStream \
    .format("kafka") \
    .option("kafka.bootstrap.servers", "localhost:9092") \
    .option("subscribe", "health_data_source1,health_data_source2,health_data_source3") \
    .load()

# Convert the Kafka message value from binary to string
json_df = df.selectExpr("CAST(value AS STRING) AS json_value") \
    .select(from_json("json_value", schema).alias("data")) \
    .select("data.*")

# Process alerts: filter patients with abnormal heart rate or SPO2 level
alerts = json_df.filter((col("heart_rate") > 80) | (col("spo2_level") < 90))

# Define function to write the processed data to MySQL
def write_to_mysql(batch_df, batch_id):
    try:
        print(f"Writing batch {batch_id} to MySQL...")
        batch_df.show(truncate=False)
        batch_df.write \
            .format("jdbc") \
            .option("url", "jdbc:mysql://localhost:3306/healthcare") \
            .option("dbtable", "patient_vitals") \
            .option("user", "sparkuser") \
            .option("password", "Mujju@2004") \
            .option("driver", "com.mysql.cj.jdbc.Driver") \
            .mode("append") \
            .save()
        print(f"Batch {batch_id} written successfully.")
    except Exception as e:
        print(f"[ERROR] Failed to write batch {batch_id} to MySQL: {e}")

# Start the streaming query to process and store the alerts into MySQL
query = alerts.writeStream \
    .foreachBatch(write_to_mysql) \
    .outputMode("append") \
    .start()

# Await termination of the streaming query
query.awaitTermination()
```

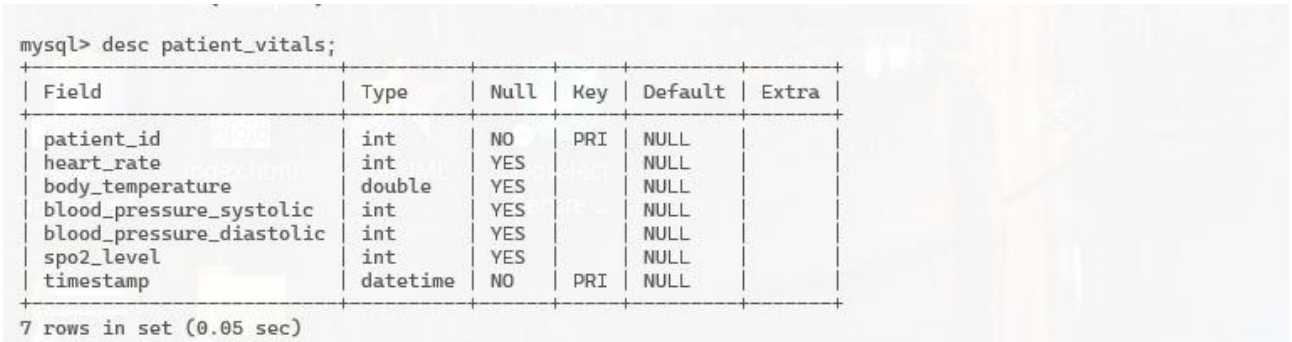
Output:

```
25/04/22 10:52:52 INFO Executor: Finished task 1.0 in stage 3.0 (TID 4). 2048 bytes result sent to driver
25/04/22 10:52:52 INFO TaskSetManager: Finished task 1.0 in stage 3.0 (TID 4) in 556 ms on 10.255.255.254 (executor driver) (2/2)
25/04/22 10:52:52 INFO TaskSchedulerImpl: Removed TaskSet 3.0, whose tasks have all completed, from pool
25/04/22 10:52:52 INFO DAGScheduler: ResultStage 3 (start at NativeMethodAccessorImpl.java:0) finished in 0.566 s
25/04/22 10:52:52 INFO DAGScheduler: Job 4 is finished. Cancelling potential speculative or zombie tasks for this job
25/04/22 10:52:52 INFO TaskSchedulerImpl: Killing all running tasks in stage 3: Stage finished
25/04/22 10:52:52 INFO DAGScheduler: Job 4 finished: start at NativeMethodAccessorImpl.java:0, took 0.569072 s
25/04/22 10:52:52 INFO CodeGenerator: Code generated in 9.17882 ms
+----------+----------+----------------+-----------------------+------------------------+-----------+--------------------------+
|patient_id|heart_rate|body_temperature|blood_pressure_systolic|blood_pressure_diastolic|spo2_level|timestamp                 |
+----------+----------+----------------+-----------------------+------------------------+-----------+--------------------------+
|2001      |93        |36.3            |104                    |75                      |92         |2025-04-22T10:52:49.432014|
|1003      |99        |37.3            |102                    |71                      |95         |2025-04-22T10:52:51.446592|
+----------+----------+----------------+-----------------------+------------------------+-----------+--------------------------+

25/04/22 10:52:53 INFO SparkContext: Starting job: start at NativeMethodAccessorImpl.java:0
25/04/22 10:52:53 INFO DAGScheduler: Got job 5 (start at NativeMethodAccessorImpl.java:0) with 3 output partitions
25/04/22 10:52:53 INFO DAGScheduler: Final stage: ResultStage 4 (start at NativeMethodAccessorImpl.java:0)
25/04/22 10:52:53 INFO DAGScheduler: Parents of final stage: List()
25/04/22 10:52:53 INFO DAGScheduler: Missing parents: List()
25/04/22 10:52:53 INFO DAGScheduler: Submitting ResultStage 4 (MapPartitionsRDD[35] at start at NativeMethodAccessorImpl.java:0), which has no missing parents
25/04/22 10:52:53 INFO MemoryStore: Block broadcast_4 stored as values in memory (estimated size 45.4 KiB, free 434.3 MiB)
25/04/22 10:52:53 INFO MemoryStore: Block broadcast_4_piece0 stored as bytes in memory (estimated size 18.9 KiB, free 434.2 MiB)
25/04/22 10:52:53 INFO BlockManagerInfo: Added broadcast_4_piece0 in memory on 10.255.255.254:45527 (size: 18.9 KiB, free: 434.4 MiB)
25/04/22 10:52:53 INFO SparkContext: Created broadcast 4 from broadcast at DAGScheduler.scala:1585
25/04/22 10:52:53 INFO DAGScheduler: Submitting 3 missing tasks from ResultStage 4 (MapPartitionsRDD[35] at start at NativeMethodAccessorImpl.java:0) (first 15 tasks are for partitions Vector(0, 1, 2))
25/04/22 10:52:53 INFO TaskSchedulerImpl: Adding task set 4.0 with 3 tasks resource profile 0
25/04/22 10:52:53 INFO TaskSetManager: Starting task 0.0 in stage 4.0 (TID 5) (10.255.255.254, executor driver, partition 0, PROCESS_LOCAL, 12209 bytes)
25/04/22 10:52:53 INFO TaskSetManager: Starting task 1.0 in stage 4.0 (TID 6) (10.255.255.254, executor driver, partition 1, PROCESS_LOCAL, 12209 bytes)
25/04/22 10:52:53 INFO TaskSetManager: Starting task 2.0 in stage 4.0 (TID 7) (10.255.255.254, executor driver, partition 2, PROCESS_LOCAL, 12209 bytes)
25/04/22 10:52:53 INFO Executor: Running task 0.0 in stage 4.0 (TID 5)
25/04/22 10:52:53 INFO Executor: Running task 1.0 in stage 4.0 (TID 6)
25/04/22 10:52:53 INFO Executor: Running task 2.0 in stage 4.0 (TID 7)
25/04/22 10:52:53 INFO KafkaBatchReaderFactory: Creating Kafka reader topicPartition=health_data_source1-0 fromOffset=1732 untilOffset=1734, for query queryId=174474e6-9dbd-4c43-9e71-9d9ddf19fd84 batchId=2 task
```

Table:

Alerts i.e., data when the heart rate was above 90 or when the SpO2 level dropped below 90 are being stored in the database



```
mysql> desc patient_vitals;
+--------------------------+-----------+------+-----+---------+-------+
| Field                    | Type      | Null | Key | Default | Extra |
+--------------------------+-----------+------+-----+---------+-------+
| patient_id               | int       | NO   | PRI | NULL    |       |
| heart_rate               | int       | YES  |     | NULL    |       |
| body_temperature         | double    | YES  |     | NULL    |       |
| blood_pressure_systolic  | int       | YES  |     | NULL    |       |
| blood_pressure_diastolic | int       | YES  |     | NULL    |       |
| spo2_level               | int       | YES  |     | NULL    |       |
| timestamp                | datetime  | NO   | PRI | NULL    |       |
+--------------------------+-----------+------+-----+---------+-------+
7 rows in set (0.05 sec)
```

```
mysql> select * from patient_vitals limit 50;
```

patient_id	heart_rate	body_temperature	blood_pressure_systolic	blood_pressure_diastolic	spo2_level	timestamp
1001	87	37.5	123	89	91	2025-04-22 10:07:13
1001	95	36.5	116	83	100	2025-04-22 10:18:16
1001	86	38.1	124	75	90	2025-04-22 10:18:25
1001	99	36.2	108	69	94	2025-04-22 10:18:43
1001	81	38.3	135	88	94	2025-04-22 10:21:59
1001	95	36.7	112	69	91	2025-04-22 10:22:05
1001	84	36.7	123	68	90	2025-04-22 10:22:48
1001	92	38	114	65	92	2025-04-22 10:23:39
1001	100	36	100	86	97	2025-04-22 10:24:00
1001	90	36.1	106	72	95	2025-04-22 10:24:06
1001	83	37.8	100	87	95	2025-04-22 10:24:18
1001	85	37.1	105	70	91	2025-04-22 10:24:24
1001	98	38.3	108	74	90	2025-04-22 10:24:52
1001	92	37.2	126	70	90	2025-04-22 10:25:13
1001	99	37.7	128	84	93	2025-04-22 10:25:19
1001	81	36.7	128	73	91	2025-04-22 10:26:13
1001	92	38.1	121	86	99	2025-04-22 10:26:22
1001	100	36.4	135	81	92	2025-04-22 10:27:05
1001	81	36.3	126	72	93	2025-04-22 10:27:08
1001	100	37.8	136	70	96	2025-04-22 10:27:20
1001	89	38.4	136	68	99	2025-04-22 10:27:23
1001	83	37.7	134	80	92	2025-04-22 10:27:29
1001	92	37.2	120	80	96	2025-04-22 10:27:32
1001	88	36.1	120	66	96	2025-04-22 10:28:20
1001	89	36.3	120	82	93	2025-04-22 10:28:45
1001	84	37.9	118	84	90	2025-04-22 10:28:48
1001	93	37.3	129	67	99	2025-04-22 10:29:30
1001	91	36.2	133	86	94	2025-04-22 10:29:45
1001	88	37.9	134	89	91	2025-04-22 10:29:51
1001	92	37.1	136	83	99	2025-04-22 10:30:09
1001	89	38.4	115	88	99	2025-04-22 10:31:13
1001	92	36.1	105	71	99	2025-04-22 10:31:25
1001	87	36.2	136	75	98	2025-04-22 10:31:34
1001	88	37.4	122	88	98	2025-04-22 10:31:46
1001	86	37.4	115	89	99	2025-04-22 10:32:01
1001	91	36.2	110	85	91	2025-04-22 10:32:16
1001	84	36.8	132	80	96	2025-04-22 10:32:25
1001	99	38	131	84	92	2025-04-22 10:32:37
1001	89	36.5	104	75	95	2025-04-22 10:33:14
1001	84	37.2	111	71	94	2025-04-22 10:33:23
1001	91	37.9	129	75	98	2025-04-22 10:33:50
1001	85	38	129	64	90	2025-04-22 10:36:51
1001	98	37.2	104	75	99	2025-04-22 10:37:00
1001	96	37.5	137	66	90	2025-04-22 10:39:13
1001	83	37.5	103	68	91	2025-04-22 10:39:19
1001	86	37.4	140	65	95	2025-04-22 10:39:40
1001	82	37.8	122	71	96	2025-04-22 10:40:04
1001	85	38.4	121	88	91	2025-04-22 10:40:10
1001	98	38.2	127	85	95	2025-04-22 10:40:25
1001	84	36.9	124	64	94	2025-04-22 10:40:31

50 rows in set (0.00 sec)

3. **Batch Mode Experiment**

a.Description

In batch mode, we used the patient data that was already stored in the patient_vitals table during streaming.

The batch file picked 100 records at a time and checked each one for abnormal values.

If the heart rate was below 60 or above 80, or the SpO2 value below 90, the record was marked as an “alert”.

If not, it was marked as “normal”.

The processed records were saved in a new table called processed_health_data.

The script also showed how many alerts were found and how much time the batch took to

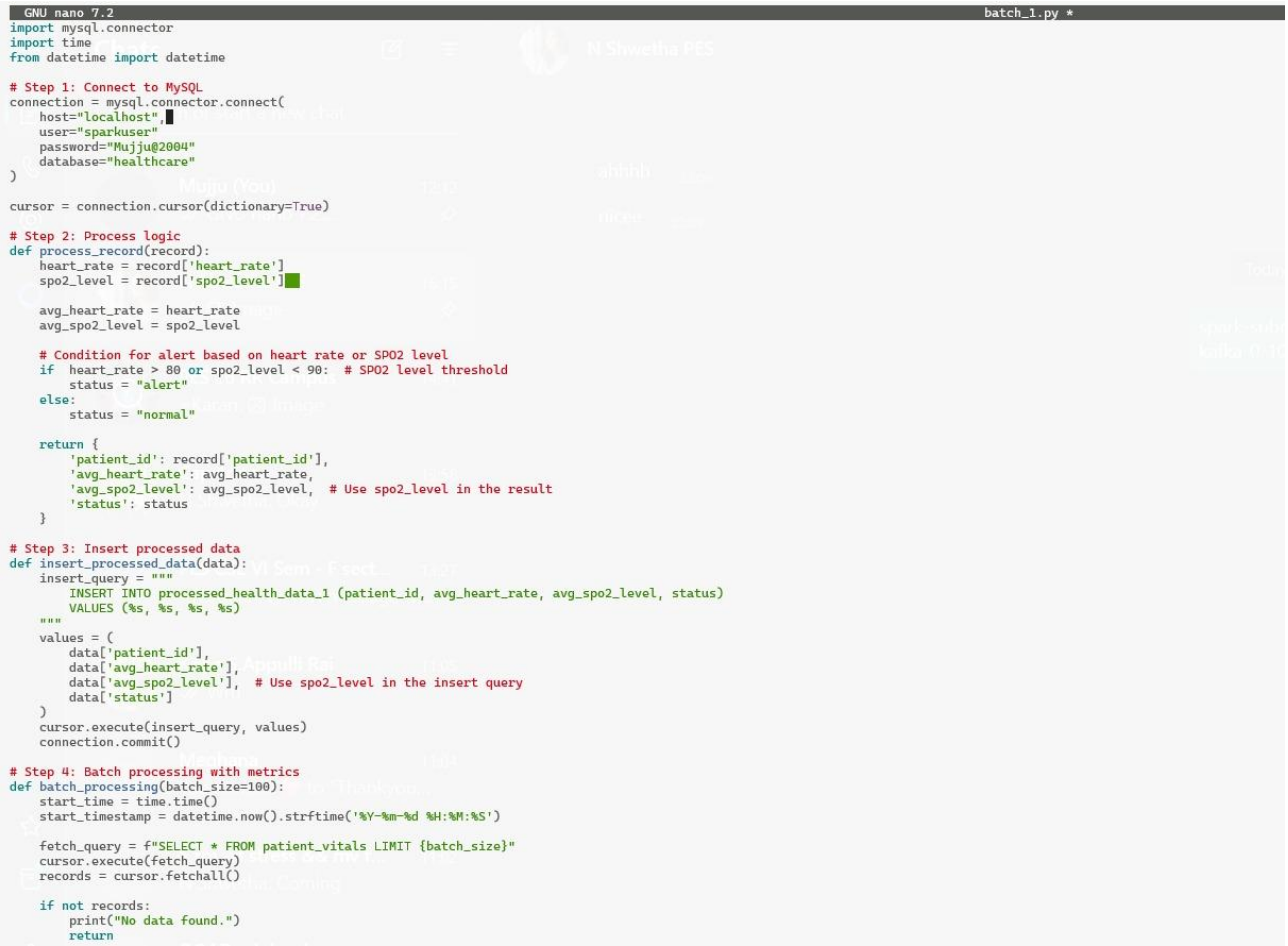
process

b. Data Size

We processed 100 records per batch from the patient_vitals table.
Each record had details like heart rate, body temperature, and timestamp.
This batch size helped simulate how a hospital might analyze a group of patient readings at once.

c.Results

Code:



```
GNU nano 7.2                                                    batch_1.py *
import mysql.connector
import time
from datetime import datetime

# Step 1: Connect to MySQL
connection = mysql.connector.connect(
    host="localhost",
    user="sparkuser"
    password="Mujju@2004"
    database="healthcare"
)

cursor = connection.cursor(dictionary=True)

# Step 2: Process logic
def process_record(record):
    heart_rate = record['heart_rate']
    spo2_level = record['spo2_level']

    avg_heart_rate = heart_rate
    avg_spo2_level = spo2_level

    # Condition for alert based on heart rate or SPO2 level
    if  heart_rate > 80 or spo2_level < 90:  # SPO2 level threshold
        status = "alert"
    else:
        status = "normal"

    return {
        'patient_id': record['patient_id'],
        'avg_heart_rate': avg_heart_rate,
        'avg_spo2_level': avg_spo2_level,  # Use spo2_level in the result
        'status': status
    }

# Step 3: Insert processed data
def insert_processed_data(data):
    insert_query = """
        INSERT INTO processed_health_data_1 (patient_id, avg_heart_rate, avg_spo2_level, status)
        VALUES (%s, %s, %s, %s)
    """
    values = (
        data['patient_id'],
        data['avg_heart_rate'],
        data['avg_spo2_level'],  # Use spo2_level in the insert query
        data['status']
    )
    cursor.execute(insert_query, values)
    connection.commit()

# Step 4: Batch processing with metrics
def batch_processing(batch_size=100):
    start_time = time.time()
    start_timestamp = datetime.now().strftime('%Y-%m-%d %H:%M:%S')

    fetch_query = f"SELECT * FROM patient_vitals LIMIT {batch_size}"
    cursor.execute(fetch_query)
    records = cursor.fetchall()

    if not records:
        print("No data found.")
        return
```

```
    alert_count = 0

    for record in records:
        processed = process_record(record)
        if processed['status'] == "alert":
            alert_count += 1
        insert_processed_data(processed)

    end_time = time.time()
    duration = round(end_time - start_time, 2)
    end_timestamp = datetime.now().strftime('%Y-%m-%d %H:%M:%S')

    print(f"[{end_timestamp}] Batch processing complete.")
    print(f"Records processed: {len(records)}")
    print(f"Alerts triggered: {alert_count}")
    print(f"Time taken: {duration} seconds")

# Step 5: Run
batch_processing(batch_size=100)

# Step 6: Close
cursor.close()
connection.close()
```

Running batch.py file:

```
(venv) mujaseemd@MujaseemD:~$ spark-submit  --jars mysql-connector-j-8.3.0.jar  --packages org.apache.spark:spark-sql-kafka-0-10_2.12:3.5.1  batch_1.py
25/04/22 10:42:00 WARN Utils: Your hostname, MujaseemD resolves to a loopback address: 127.0.1.1; using 10.255.255.254 instead (on interface lo)
25/04/22 10:42:00 WARN Utils: Set SPARK_LOCAL_IP if you need to bind to another address
:: loading settings :: url = jar:file:/home/mujaseemd/spark/jars/ivy-2.5.1.jar!/org/apache/ivy/core/settings/ivysettings.xml
Ivy Default Cache set to: /home/mujaseemd/.ivy2/cache
The jars for the packages stored in: /home/mujaseemd/.ivy2/jars
org.apache.spark#spark-sql-kafka-0-10_2.12 added as a dependency
:: resolving dependencies :: org.apache.spark#spark-submit-parent-2c3ef2f4-15d9-4fab-b7e5-3e1e5d9b559a;1.0
        confs: [default]
        found org.apache.spark#spark-sql-kafka-0-10_2.12;3.5.1 in central
        found org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.5.1 in central
        found org.apache.kafka#kafka-clients;3.4.1 in central
        found org.lz4#lz4-java;1.8.0 in central
        found org.xerial.snappy#snappy-java;1.1.10.3 in central
        found org.slf4j#slf4j-api;2.0.7 in central
        found org.apache.hadoop#hadoop-client-runtime;3.3.4 in central
        found org.apache.hadoop#hadoop-client-api;3.3.4 in central
        found commons-logging#commons-logging;1.1.3 in central
        found com.google.code.findbugs#jsr305;3.0.0 in central
        found org.apache.commons#commons-pool2;2.11.1 in central
:: resolution report :: resolve 408ms :: artifacts dl 15ms
        :: modules in use:
        com.google.code.findbugs#jsr305;3.0.0 from central in [default]
        commons-logging#commons-logging;1.1.3 from central in [default]
        org.apache.commons#commons-pool2;2.11.1 from central in [default]
        org.apache.hadoop#hadoop-client-api;3.3.4 from central in [default]
        org.apache.hadoop#hadoop-client-runtime;3.3.4 from central in [default]
        org.apache.kafka#kafka-clients;3.4.1 from central in [default]
        org.apache.spark#spark-sql-kafka-0-10_2.12;3.5.1 from central in [default]
        org.apache.spark#spark-token-provider-kafka-0-10_2.12;3.5.1 from central in [default]
        org.lz4#lz4-java;1.8.0 from central in [default]
        org.slf4j#slf4j-api;2.0.7 from central in [default]
        org.xerial.snappy#snappy-java;1.1.10.3 from central in [default]
        ---------------------------------------------------------------------
        |                  |            modules            ||   artifacts   |
        |       conf       | number| search|dwnlded|evicted|| number|dwnlded|
        ---------------------------------------------------------------------
        |      default     |   11  |   0   |   0   |   0   ||   11  |   0   |
        ---------------------------------------------------------------------
:: retrieving :: org.apache.spark#spark-submit-parent-2c3ef2f4-15d9-4fab-b7e5-3e1e5d9b559a
        confs: [default]
        0 artifacts copied, 11 already retrieved (0kB/9ms)
25/04/22 10:42:01 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-java classes where applicable
[2025-04-22 10:42:01] Fetched 100 records for batch processing...
[2025-04-22 10:42:02] Batch processing complete.
Records processed: 100
Alerts triggered: 100
Time taken: 0.41 seconds
25/04/22 10:42:02 INFO ShutdownHookManager: Shutdown hook called
25/04/22 10:42:02 INFO ShutdownHookManager: Deleting directory /tmp/spark-cb00f6cc-b40a-4830-82bb-cb67ee978bc5
```

Table:

```
mysql> desc processed_health_data_1;
+---------------+-----------------------+------+-----+-------------------+-------------------+
| Field         | Type                  | Null | Key | Default           | Extra             |
+---------------+-----------------------+------+-----+-------------------+-------------------+
| id            | int                   | NO   | PRI | NULL              | auto_increment    |
| patient_id    | int                   | NO   |     | NULL              |                   |
| avg_heart_rate | double               | NO   |     | NULL              |                   |
| avg_spo2_level | double               | NO   |     | NULL              |                   |
| status        | enum('normal','alert') | NO   |     | NULL              |                   |
| processed_at  | timestamp             | YES  |     | CURRENT_TIMESTAMP | DEFAULT_GENERATED |
+---------------+-----------------------+------+-----+-------------------+-------------------+
6 rows in set (0.01 sec)
```

```
mysql> select * from processed_health_data_1;
+-----+------------+----------------+----------------+--------+---------------------+
| id  | patient_id | avg_heart_rate | avg_spo2_level | status | processed_at        |
+-----+------------+----------------+----------------+--------+---------------------+
| 401 |       1001 |             87 |             91 | alert  | 2025-04-22 10:59:20 |
| 402 |       1001 |             95 |            100 | alert  | 2025-04-22 10:59:20 |
| 403 |       1001 |             86 |             90 | alert  | 2025-04-22 10:59:20 |
| 404 |       1001 |             99 |             94 | alert  | 2025-04-22 10:59:20 |
| 405 |       1001 |             81 |             94 | alert  | 2025-04-22 10:59:20 |
| 406 |       1001 |             95 |             91 | alert  | 2025-04-22 10:59:20 |
| 407 |       1001 |             84 |             90 | alert  | 2025-04-22 10:59:20 |
| 408 |       1001 |             92 |             92 | alert  | 2025-04-22 10:59:20 |
| 409 |       1001 |            100 |             97 | alert  | 2025-04-22 10:59:20 |
| 410 |       1001 |             90 |             95 | alert  | 2025-04-22 10:59:20 |
| 411 |       1001 |             83 |             95 | alert  | 2025-04-22 10:59:20 |
| 412 |       1001 |             85 |             91 | alert  | 2025-04-22 10:59:20 |
| 413 |       1001 |             98 |             90 | alert  | 2025-04-22 10:59:20 |
| 414 |       1001 |             92 |             90 | alert  | 2025-04-22 10:59:20 |
| 415 |       1001 |             99 |             93 | alert  | 2025-04-22 10:59:20 |
| 416 |       1001 |             81 |             91 | alert  | 2025-04-22 10:59:20 |
| 417 |       1001 |             92 |             99 | alert  | 2025-04-22 10:59:20 |
| 418 |       1001 |            100 |             92 | alert  | 2025-04-22 10:59:20 |
| 419 |       1001 |             81 |             93 | alert  | 2025-04-22 10:59:20 |
| 420 |       1001 |            100 |             96 | alert  | 2025-04-22 10:59:20 |
| 421 |       1001 |             89 |             99 | alert  | 2025-04-22 10:59:20 |
| 422 |       1001 |             83 |             92 | alert  | 2025-04-22 10:59:20 |
| 423 |       1001 |             92 |             96 | alert  | 2025-04-22 10:59:20 |
| 424 |       1001 |             88 |             96 | alert  | 2025-04-22 10:59:20 |
| 425 |       1001 |             89 |             93 | alert  | 2025-04-22 10:59:20 |
| 426 |       1001 |             84 |             90 | alert  | 2025-04-22 10:59:20 |
| 427 |       1001 |             93 |             99 | alert  | 2025-04-22 10:59:20 |
| 428 |       1001 |             91 |             94 | alert  | 2025-04-22 10:59:20 |
| 429 |       1001 |             88 |             91 | alert  | 2025-04-22 10:59:20 |
| 430 |       1001 |             92 |             99 | alert  | 2025-04-22 10:59:20 |
| 431 |       1001 |             89 |             99 | alert  | 2025-04-22 10:59:20 |
| 432 |       1001 |             92 |             99 | alert  | 2025-04-22 10:59:20 |
| 433 |       1001 |             87 |             98 | alert  | 2025-04-22 10:59:20 |
| 434 |       1001 |             88 |             98 | alert  | 2025-04-22 10:59:20 |
| 435 |       1001 |             86 |             99 | alert  | 2025-04-22 10:59:20 |
| 436 |       1001 |             91 |             91 | alert  | 2025-04-22 10:59:20 |
| 437 |       1001 |             84 |             96 | alert  | 2025-04-22 10:59:20 |
| 438 |       1001 |             99 |             92 | alert  | 2025-04-22 10:59:20 |
| 439 |       1001 |             89 |             95 | alert  | 2025-04-22 10:59:20 |
| 440 |       1001 |             84 |             94 | alert  | 2025-04-22 10:59:20 |
| 441 |       1001 |             91 |             98 | alert  | 2025-04-22 10:59:20 |
| 442 |       1001 |             85 |             90 | alert  | 2025-04-22 10:59:20 |
| 443 |       1001 |             98 |             99 | alert  | 2025-04-22 10:59:20 |
| 444 |       1001 |             96 |             90 | alert  | 2025-04-22 10:59:20 |
| 445 |       1001 |             83 |             91 | alert  | 2025-04-22 10:59:20 |
| 446 |       1001 |             86 |             95 | alert  | 2025-04-22 10:59:20 |
| 447 |       1001 |             82 |             96 | alert  | 2025-04-22 10:59:20 |
| 448 |       1001 |             85 |             91 | alert  | 2025-04-22 10:59:20 |
| 449 |       1001 |             98 |             95 | alert  | 2025-04-22 10:59:20 |
| 450 |       1001 |             84 |             94 | alert  | 2025-04-22 10:59:20 |
| 451 |       1001 |             90 |             90 | alert  | 2025-04-22 10:59:20 |
| 452 |       1001 |             82 |             91 | alert  | 2025-04-22 10:59:20 |
| 453 |       1001 |             92 |             93 | alert  | 2025-04-22 10:59:20 |
| 454 |       1001 |             98 |             95 | alert  | 2025-04-22 10:59:20 |
| 455 |       1001 |             93 |             91 | alert  | 2025-04-22 10:59:20 |
| 456 |       1001 |             91 |             96 | alert  | 2025-04-22 10:59:20 |
| 457 |       1001 |             95 |             93 | alert  | 2025-04-22 10:59:20 |
| 458 |       1001 |             82 |             97 | alert  | 2025-04-22 10:59:20 |
| 459 |       1001 |             89 |             92 | alert  | 2025-04-22 10:59:20 |
| 460 |       1001 |             93 |             96 | alert  | 2025-04-22 10:59:20 |
| 461 |       1001 |             94 |             90 | alert  | 2025-04-22 10:59:20 |
| 462 |       1001 |             83 |             92 | alert  | 2025-04-22 10:59:20 |
```

437	1001	84	96	alert	2025-04-22 10:59:20
438	1001	99	92	alert	2025-04-22 10:59:20
439	1001	89	95	alert	2025-04-22 10:59:20
440	1001	84	94	alert	2025-04-22 10:59:20
441	1001	91	98	alert	2025-04-22 10:59:20
442	1001	85	90	alert	2025-04-22 10:59:20
443	1001	98	99	alert	2025-04-22 10:59:20
444	1001	96	90	alert	2025-04-22 10:59:20
445	1001	83	91	alert	2025-04-22 10:59:20
446	1001	86	95	alert	2025-04-22 10:59:20
447	1001	82	96	alert	2025-04-22 10:59:20
448	1001	85	91	alert	2025-04-22 10:59:20
449	1001	98	95	alert	2025-04-22 10:59:20
450	1001	84	94	alert	2025-04-22 10:59:20
451	1001	90	90	alert	2025-04-22 10:59:20
452	1001	82	91	alert	2025-04-22 10:59:20
453	1001	92	93	alert	2025-04-22 10:59:20
454	1001	98	95	alert	2025-04-22 10:59:20
455	1001	93	91	alert	2025-04-22 10:59:20
456	1001	91	96	alert	2025-04-22 10:59:20
457	1001	95	93	alert	2025-04-22 10:59:20
458	1001	82	97	alert	2025-04-22 10:59:20
459	1001	89	92	alert	2025-04-22 10:59:20
460	1001	93	96	alert	2025-04-22 10:59:20
461	1001	94	90	alert	2025-04-22 10:59:20
462	1001	83	92	alert	2025-04-22 10:59:20
463	1001	97	95	alert	2025-04-22 10:59:20
464	1001	99	90	alert	2025-04-22 10:59:20
465	1001	89	96	alert	2025-04-22 10:59:20
466	1001	93	99	alert	2025-04-22 10:59:20
467	1001	96	100	alert	2025-04-22 10:59:20
468	1001	85	99	alert	2025-04-22 10:59:20
469	1001	88	91	alert	2025-04-22 10:59:20
470	1001	94	93	alert	2025-04-22 10:59:20
471	1001	97	99	alert	2025-04-22 10:59:20
472	1001	84	93	alert	2025-04-22 10:59:20
473	1001	88	99	alert	2025-04-22 10:59:20
474	1001	98	94	alert	2025-04-22 10:59:20
475	1001	100	98	alert	2025-04-22 10:59:20
476	1001	88	90	alert	2025-04-22 10:59:20
477	1001	82	95	alert	2025-04-22 10:59:20
478	1001	98	90	alert	2025-04-22 10:59:20
479	1001	81	90	alert	2025-04-22 10:59:20
480	1001	85	92	alert	2025-04-22 10:59:20
481	1001	98	90	alert	2025-04-22 10:59:20
482	1001	94	92	alert	2025-04-22 10:59:20
483	1001	84	92	alert	2025-04-22 10:59:20
484	1001	86	90	alert	2025-04-22 10:59:20
485	1001	95	96	alert	2025-04-22 10:59:20
486	1001	88	96	alert	2025-04-22 10:59:20
487	1001	98	100	alert	2025-04-22 10:59:20
488	1001	97	97	alert	2025-04-22 10:59:20
489	1001	98	91	alert	2025-04-22 10:59:20
490	1001	82	94	alert	2025-04-22 10:59:20
491	1001	90	100	alert	2025-04-22 10:59:20
492	1001	82	96	alert	2025-04-22 10:59:20
493	1001	86	92	alert	2025-04-22 10:59:20
494	1001	81	90	alert	2025-04-22 10:59:20
495	1001	92	92	alert	2025-04-22 10:59:20
496	1001	100	95	alert	2025-04-22 10:59:20
497	1001	93	98	alert	2025-04-22 10:59:20
498	1001	93	90	alert	2025-04-22 10:59:20
499	1001	92	95	alert	2025-04-22 10:59:20
500	1001	92	95	alert	2025-04-22 10:59:20

100 rows in set (0.00 sec)

6.Comparison of Streaming & Batch Modes

a.Results and Discussions

We used both streaming and batch processing for patient health data:

Streaming Mode :

- Captured live data every 2 seconds.
- Detected abnormal values (e.g., high heart rate, low SpO_2) and inserted alerts into MySQL in real time.

Batch Mode :

- Analyzed stored data in batches of 100 records.
- Labeled data as “alert” or “normal” and stored results in the processed_health_data table.

Discussion

- Streaming is great for real-time alerts.
- Batch is better for trend analysis and large data processing.
- Combining both offers real-time detection and historical insights.

Aspect	**Batch Processing Script**	**Streaming (Spark Kafka) Script**
Data Source	Stored records in MySQL (patient_vitals)	Live data from 3 different kafka topics
Trigger Type	Manual or scheduled run (LIMIT 100)	Automatically triggered as new data arrives in Kafka
Alert Condition	heart_rate >80 spo2_level < 90	heart_rate > 80 OR spo2_level < 90
Insert Destination	processed_health_data table in MySQL	patient_vitals table in MySQL
Programming Language	Python + MySQL connector	PySpark + Kafka + MySQL JDBC
Execution Model	One-time batch run	Continuous real-time stream (runs until stopped)

7.Conclusion

Our project used both streaming and batch processing to manage patient health data. Streaming helped with real-time alerts, while batch processing handled data analysis over time. Combining both gave us a system that can quickly respond and analyze data effectively.

8.References

- Apache Kafka Documentation. (2023). *Apache Kafka: A Distributed Streaming Platform*. Retrieved from https://kafka.apache.org/documentation/
- https://youtu.be/KerNf0NANMo?si=yVwUiyrniBnyolhJ

9.Source Code link (GitHub repo):

https://github.com/mujaseemd/Real_Time_Patient_Health_Monitoring_System