Report

Kafka Container running

A screenshot of a computer program

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer

AI-generated content may be incorrect.

Creating first topic

A computer screen with white text

AI-generated content may be incorrect.

You can run a Kafka Connect worker directly as a JVM process on a virtual machine or bare metal, but you might prefer the convenience of running it in a container, using a technology like Kubernetes or Docker. Note that containerized Connect via Docker will be used for many of the examples in this series.

Confluent maintains its own image for Kafka Connect, cp-kafka-connect-base, which provides a basic Connect worker to which you can add your desired JAR files for sink and source connectors, single message transforms, and converters.

Setting up Kafka connect  
A screen shot of a computer

AI-generated content may be incorrect.

Connector for posgresql

A screenshot of a computer

AI-generated content may be incorrect.

Youtube videos: <https://www.youtube.com/watch?v=aTl2iSCynVc&ab_channel=NanthakumaranSenthilnathan>

<https://github.com/nanthakumaran-s/Learn-Kafka/blob/main/commands.md>

Streming package: <https://hub.docker.com/r/bitnami/kafka>

Verification of the connection:  
A screenshot of a computer

AI-generated content may be incorrect.

Packages to download:  
scoop, jq.exe

 **Kafka Container:** Running and accessible.

 **Kafka Connect Container:** Running and accessible on port 8083.

 **JDBC Connectors Installed:** Both the JDBC Source Connector (io.confluent.connect.jdbc.JdbcSourceConnector) and the JDBC Sink Connector (io.confluent.connect.jdbc.JdbcSinkConnector) are installed in your Kafka Connect instance.

Repository for different connectors: <https://github.com/lensesio/stream-reactor>

Download SBT

Install JDK 8+

Run commando “sbt clean compile” to build a connection between kafka and mqtt. Takes time

Download Git

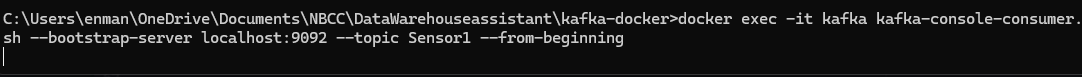
Pykafka

Paho-mqtt

Checking kafka topics with sensor 1

A black screen with white text

AI-generated content may be incorrect.

Create a consumer to see messages:  


Receiving and showing data

A screenshot of a computer

AI-generated content may be incorrect.

Checking:  
A screen shot of a computer

AI-generated content may be incorrect.

## Code:

Directory:   
C:\Users\enman\OneDrive\Documents\NBCC\DataWarehouseassistant\kafka-docker

Get inside kafka:  
docker exec -it kafka /bin/bash

Check topics in container:  
docker exec -it kafka kafka-topics.sh --list --bootstrap-server localhost:9092

Create topic in container:  
docker exec kafka kafka-topics.sh --create --topic Sensor1 --bootstrap-server localhost:9092 --partitions 1 --replication-factor 1

List running containers:  
docker ps

Run docker:  
docker-compose up -d

Restart:  
docker restart <container\_name\_or\_id>

Restart all running container:  
docker restart $(docker ps -q)

Stop and remove containers:  
docker-compose down

Run consumer:  
docker exec -it kafka kafka-console-consumer.sh --bootstrap-server localhost:9092 --topic Sensor1 --from-beginning

Start Kafka in the background:  
docker-compose up -d