

Langkah 1: Buat Publisher dan Subscriber MQTT (HiveMQ)

Tools yang digunakan:

- Python (pakai paho-mqtt)
- HiveMQ public broker: broker.hivemq.com port 1883
- Topic: misal iot/edge/suhu

1A. Publisher Python Script

```
import paho.mqtt.client as mqtt
import time
import random

broker = "broker.hivemq.com"
port = 1883
topic = "iot/edge/suhu"

client = mqtt.Client()
client.connect(broker, port)

try:
    while True:
        suhu = round(random.uniform(20.0, 40.0), 2)
        client.publish(topic, str(suhu))
        print(f"Data suhu dikirim: {suhu} °C")
        time.sleep(5)
except KeyboardInterrupt:
    print("Publisher dihentikan.")
```

1B. Subscriber Python Script

```
import paho.mqtt.client as mqtt

broker = "broker.hivemq.com"
port = 1883
topic = "iot/edge/suhu"

def on_connect(client, userdata, flags, rc):
    print("Terhubung ke broker")
    client.subscribe(topic)

def on_message(client, userdata, msg):
    print(f"Suhu diterima: {msg.payload.decode()} °C")

client = mqtt.Client()
client.on_connect = on_connect
client.on_message = on_message

client.connect(broker, port)
client.loop_forever()
```

Langkah 2: Integrasi ke Monitoring

Tools:

- Telegraf: agen pengumpul data
- InfluxDB: penyimpanan time-series
- Grafana: visualisasi data

2A. docker-compose.yml

```
version: '3'

services:
  influxdb:
    image: influxdb:1.8
    ports:
      - "8086:8086"
    volumes:
      - influxdb:/var/lib/influxdb
    environment:
      - INFLUXDB_DB=iotdb
      - INFLUXDB_ADMIN_ENABLED=true

  telegraf:
    image: telegraf:latest
    volumes:
      - ./telegraf.conf:/etc/telegraf/telegraf.conf
    depends_on:
      - influxdb

  grafana:
    image: grafana/grafana
    ports:
      - "3000:3000"
    depends_on:
      - influxdb

volumes:
  influxdb:
```

2B. Contoh telegraf.conf (input MQTT + output InfluxDB)

```
[agent]
interval = "10s"

[[inputs.mqtt_consumer]]
servers = ["tcp://broker.hivemq.com:1883"]
topics = ["iot/edge/suhu"]
data_format = "value"
data_type = "float"
name_override = "suhu_sensor"

[[outputs.influxdb]]
urls = ["http://influxdb:8086"]
database = "iotdb"
```

Langkah 3: Menjalankan yang telah dibuat

1. Buka VSCode
2. Buat folder proyek baru, misal: `iot_mqtt_project`
3. Buat dua file:
 - `publisher.py`
 - `subscriber.py`

4. Install `paho-mqtt`:
Buka terminal dan ketik:

```
pip install paho-mqtt
```

5. Buka dua terminal di VSCode:

Terminal 1: jalankan publisher

```
python publisher.py
```

Terminal 2: jalankan subscriber

```
python subscriber.py
```

Langkah 4: Jalankan Docker Desktop

Buka Start Menu → cari Docker Desktop → klik untuk menjalankannya

Atau jalankan perintah berikut di vscode:

```
docker compose up -d
```

Tunggu sampai ikon Docker di system tray berubah jadi warna hijau / ready

Langkah 5: Buka Grafana

1. Buka: <http://localhost:3000>
2. Login: admin / admin
3. Tambahkan Data Source:
 - Type: InfluxDB
 - URL: <http://influxdb:8086>
 - Database: iotdb
 - Test & Save

Tambahkan Data Source (InfluxDB)

1. Masuk ke Gear → Data Sources → Add Data Source
2. Pilih: InfluxDB
3. Isi konfigurasi:
 - URL: <http://influxdb:8086>
 - Database: iotdb
 - HTTP Method: GET
 - Leave Username & Password kosong (karena tidak pakai auth)
4. Klik Save & Test

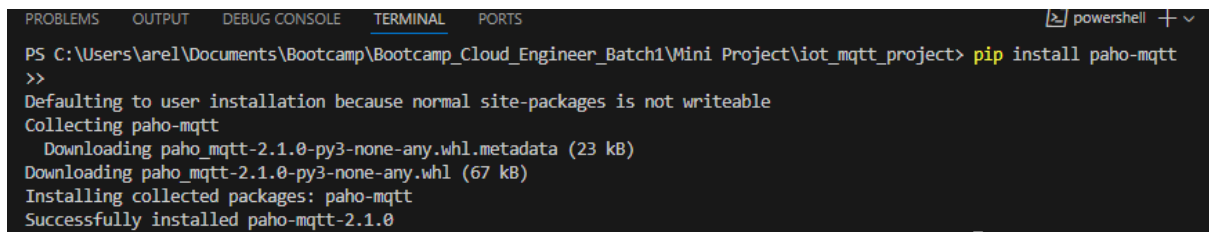
Buat Dashboard untuk Sensor Suhu

1. Masuk ke + → Dashboard → New dashboard → Add visualization
2. Di panel query (bagian bawah):

- Pilih Data source: pastikan InfluxDB yang sudah dibuat sebelumnya
3. Di query area:
- Measurement: suhu_sensor (pakai name_override di telegraf.conf)
 - Field: biasanya value (jika data_format = "value" di telegraf.conf)
 - Pilih Aggregation: misalnya mean() atau last()
4. Ubah visualisasi jadi Line Chart (default-nya biasanya Line)
5. Klik tombol Apply

Berikut screenshot dari hasil yang telah dikerjakan :

(Install paho-mqtt)



```
PS C:\Users\arel\Documents\Bootcamp\Bootcamp_Cloud_Engineer_Batch1\Mini Project\iot_mqtt_project> pip install paho-mqtt
>>
Defaulting to user installation because normal site-packages is not writeable
Collecting paho-mqtt
  Downloading paho_mqtt-2.1.0-py3-none-any.whl.metadata (23 kB)
  Downloading paho_mqtt-2.1.0-py3-none-any.whl (67 kB)
Installing collected packages: paho-mqtt
Successfully installed paho-mqtt-2.1.0
```

(python publisher.py)



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Data suhu dikirim: 20.79 °C
Data suhu dikirim: 22.13 °C
Data suhu dikirim: 21.85 °C
Data suhu dikirim: 35.78 °C
Data suhu dikirim: 37.93 °C
Data suhu dikirim: 29.21 °C
Data suhu dikirim: 38.31 °C
Data suhu dikirim: 23.72 °C
Data suhu dikirim: 34.22 °C
Data suhu dikirim: 20.81 °C
Data suhu dikirim: 38.42 °C
Data suhu dikirim: 33.84 °C
Data suhu dikirim: 24.21 °C
Data suhu dikirim: 38.55 °C
Data suhu dikirim: 27.78 °C
Data suhu dikirim: 24.18 °C
Data suhu dikirim: 30.07 °C
Data suhu dikirim: 20.46 °C
Data suhu dikirim: 21.76 °C
```

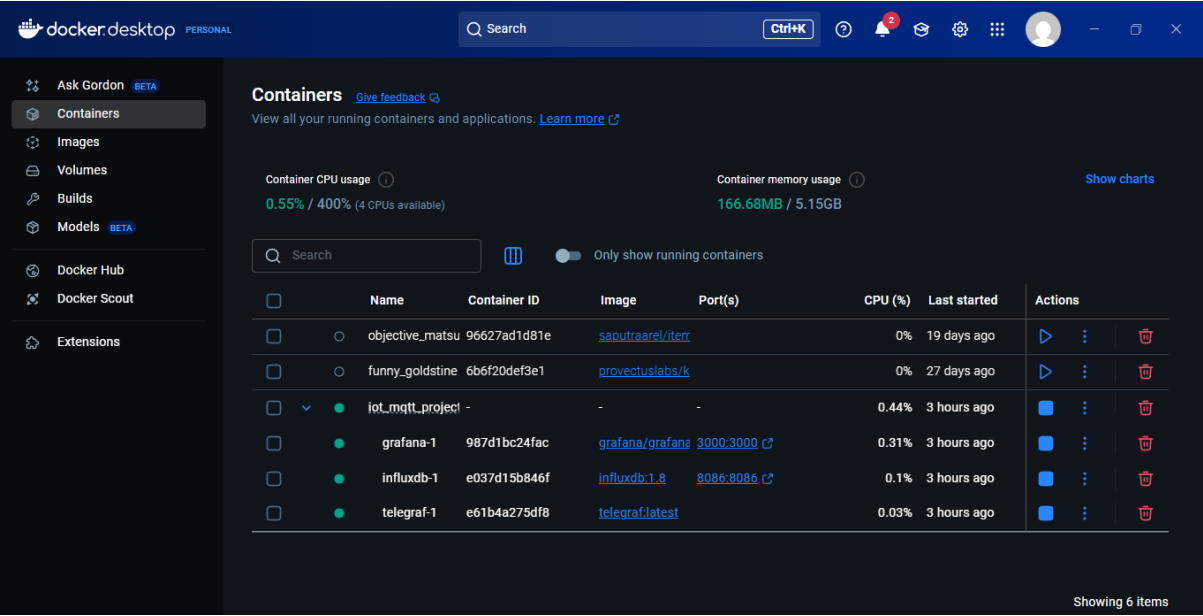
(python subscriber.py)

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
Suhu diterima: 35.06 °C
Suhu diterima: 28.66 °C
Suhu diterima: 21.82 °C
Suhu diterima: 22.8 °C
Suhu diterima: 35.32 °C
Suhu diterima: 31.92 °C
Suhu diterima: 22.32 °C
Suhu diterima: 28.8 °C
Suhu diterima: 23.37 °C
Suhu diterima: 38.18 °C
Suhu diterima: 20.43 °C
Suhu diterima: 38.11 °C
Suhu diterima: 35.74 °C
Suhu diterima: 22.25 °C
Suhu diterima: 35.95 °C
Suhu diterima: 24.85 °C
Suhu diterima: 26.95 °C
Suhu diterima: 22.08 °C
Suhu diterima: 36.57 °C
```

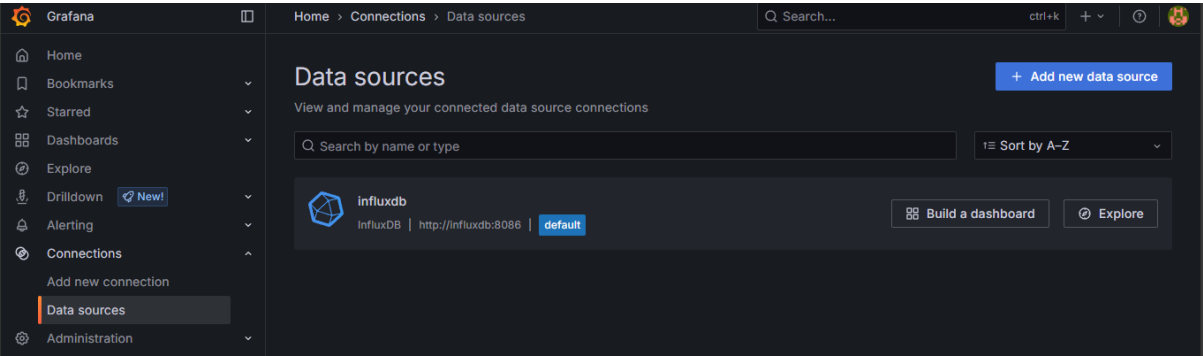
(docker compose up -d)

```
PS C:\Users\arel\Documents\Bootcamp\Bootcamp_Cloud_Engineer_Batch1\Mini Project\iot_mqtt_project> docker compose up -d
>>
time="2025-07-23T11:49:57+07:00" level=warning msg="C:\\Users\\arel\\Documents\\Bootcamp\\Bootcamp_Cloud_Engineer_Batch1\\Mini Proje
ct\\iot_mqtt_project\\docker-compose.yml: the attribute `version` is obsolete, it will be ignored, please remove it to avoid potenti
al confusion"
[+] Running 26/26
 ✓ telegraf Pulled                               518.9s
 ✓ grafana Pulled                                536.3s
 ✓ influxdb Pulled                               393.4s
[+] Running 4/4
 ✓ Network iot_mqtt_project_default              Created              0.3s
 ✓ Container iot_mqtt_project-grafana-1          Started                   36.7s
 ✓ Container iot_mqtt_project-influxdb-1         Started                   36.7s
 ✓ Container iot_mqtt_project-telegraf-1         Started                   37.1s
```

(Docker Desktop)



(Data Source (InfluxDB))



(Grafana Dashboard untuk Sensor Suhu)

