**Report on IoT Data Pipeline Using MQTT**

## **Objective**

The objective of this assignment is to simulate an Internet of Things (IoT) data flow where sensor data is generated, transmitted over MQTT, received back through subscription, and stored locally. This setup represents a simplified version of how IoT edge devices communicate with brokers and store or forward data.

## **System Setup**

### **1. MQTT Broker**

* An MQTT broker is required as the central communication hub.
* Two options were considered:  
  1. **Local Broker** – Using *Eclipse Mosquitto*, installed and run on the local machine.  
     + This allows testing without internet dependency.
     + Broker address: localhost:1883.
  2. **Public Broker** – Using test.mosquitto.org.  
     + Useful when a local broker is unavailable.
     + Requires internet connectivity.

### **2. Python Script**

The script was designed to:

1. **Simulate Sensor Data**
   * Two parameters were chosen:  
     + **Temperature (temp)** – Random float between 20.0°C and 30.0°C.
     + **Relative Humidity (rh)** – Random float between 40% and 60%.
2. **Publish Data to Broker**
   * Generated data is published to a topic (e.g., scitech/assignment) at fixed intervals.
3. **Subscribe to Same Topic**
   * The script subscribes to the topic so that it can receive the same data it publishes.
   * This loopback ensures the end-to-end flow works correctly.
4. **Store Data Locally**
   * Received messages are parsed and appended into a sensor\_data.json file.
   * This creates a local log of all transmitted data.

## **Workflow**

The overall data flow is as follows:

1. Simulated sensor data is generated in Python.
2. The data is **published** to the MQTT broker.
3. Since the script is also subscribed, it immediately **receives** the same message.
4. The message is parsed and stored in a local JSON file.

## **Flow Diagram**

[Simulated Sensors] → [Python Script] → (Publish)

↓

[MQTT Broker]

↓

(Subscribe/Receive)

↓

[Local JSON File Storage]

## **Results**

* A continuous stream of simulated data was successfully generated.
* The data was transmitted through the broker and logged locally in JSON format.
* Both local (localhost) and public (test.mosquitto.org) brokers worked as expected, with the difference being local offline operation vs. internet dependency.

## **Conclusion**

This assignment demonstrates the fundamental IoT workflow using MQTT:

* Data generation (sensors)
* Data transmission (publish)
* Data reception (subscribe)
* Data storage (local logging)

It provides a foundation for extending the system toward real hardware sensors and cloud-based dashboards.