Nama : Bernard Santosa

Group : siao langs

import network

import time

from machine import Pin

from umqtt.simple import MQTTClient

# MQTT Server Parameters

MQTT\_CLIENT\_ID = "demoBernard123keafawd"

MQTT\_BROKER    = "broker.emqx.io"

MQTT\_USER      = ""

MQTT\_PASSWORD  = ""

LED\_CONTROL\_TOPIC = "/siao\_langs/bernard\_santosa/aktuasi\_led"  # Topic untuk kontrol LED

STATUS\_TOPIC = "/siao\_langs/bernard\_santosa/data\_sensor"       # Topic untuk status LED

# Inisialisasi LED pada pin D33

led = Pin(33, Pin.OUT)

# WIFI Connection

print("Connecting to WiFi", end="")

sta\_if = network.WLAN(network.STA\_IF)

sta\_if.active(True)

sta\_if.connect('Wokwi-GUEST', '')

while not sta\_if.isconnected():

    print(".", end="")

    time.sleep(0.1)

print(" Connected!")

# MQTT Server connection

try:

    print("Connecting to MQTT server... ", end="")

    client = MQTTClient(MQTT\_CLIENT\_ID, MQTT\_BROKER, user=MQTT\_USER, password=MQTT\_PASSWORD, keepalive=60)

    client.connect()

    print("Connected!")

    # Tambahkan log bahwa perangkat akan mengirim data ke STATUS\_TOPIC

    print(f"Will publish data to {STATUS\_TOPIC}")

except OSError as e:

    print(f"Failed to connect to MQTT server: {e}")

    time.sleep(5)

    machine.reset()  # Restart perangkat jika koneksi gagal

# Fungsi untuk mengirim status LED ke MQTT

def send\_led\_status():

    status = "ON" if led.value() == 1 else "OFF"

    message = f"Status LED = {status}"

    client.publish(STATUS\_TOPIC, message)  # Kirim status ke MQTT

    print(f"Sent: {message}")

# Fungsi untuk menangani pesan MQTT

def on\_message(topic, msg):

    print("Received message:", msg)

    if msg == b"ON":

        led.value(1)  # Nyalakan LED

        print("LED ON")

        send\_led\_status()  # Kirim status LED setelah diubah

    elif msg == b"OFF":

        led.value(0)  # Matikan LED

        print("LED OFF")

        send\_led\_status()  # Kirim status LED setelah diubah

# Subscribe ke topic untuk mengendalikan LED

client.set\_callback(on\_message)

client.subscribe(LED\_CONTROL\_TOPIC)

print(f"Subscribed to {LED\_CONTROL\_TOPIC}")

# Loop untuk mendengarkan pesan MQTT

while True:

    client.check\_msg()  # Cek pesan masuk dari broker MQTT

    time.sleep(0.1)  # Beri waktu untuk memproses pesan