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
```