#include <WiFi.h>

#include "PubSubClient.h"

// Abra o link https://wokwi.com/projects/316434364352168512

// Apague o programa todo e cole este programa.

// MQTT a publicar: cedup/led.

const char\* ssid = "Wokwi-GUEST";

const char\* password = "";

const char\* mqttServer = "broker.hivemq.com";

int port = 1883; // Aqui no ESP32 é esta porta, no navegador continue usando a porta 8000

String stMac;

char mac[50];

char\* clientId = "";

WiFiClient espClient;

PubSubClient client(espClient);

const int ledPin = 2; // Pino que o LED está ligado

void setup() {

**Serial**.begin(115200);

randomSeed(analogRead(0));

delay(10);

**Serial**.println();

**Serial**.print("Connecting to ");

**Serial**.println(ssid);

wifiConnect();

**Serial**.println("");

**Serial**.println("WiFi connected");

**Serial**.println("IP address: ");

**Serial**.println(WiFi.localIP());

client.setServer(mqttServer, port);

client.setCallback(callback);

pinMode(ledPin, OUTPUT);

}

void wifiConnect() {

WiFi.mode(WIFI\_STA);

WiFi.begin(ssid, password);

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

**Serial**.print(".");

}

}

void mqttReconnect() {

while (!client.connected()) {

**Serial**.print("Aguardando se conectar com o servidor MQTT...");

long r = random(1000);

sprintf(clientId, "clientId-%ld", r);

if (client.connect(clientId)) {

**Serial**.println(" connectado");

client.subscribe("cedup/texto"); // Tópico que está lendo no servidor MQTT

} else {

**Serial**.print("failed, rc=");

**Serial**.print(client.state());

**Serial**.println(" tentando se conectar em 5s");

delay(5000);

}

}

}

void callback(char\* topic, byte\* message, unsigned int length) {

**Serial**.print("Mensagem chegou no tópico: ");

**Serial**.print(topic);

**Serial**.print(". Mensagem: ");

String stMessage;

for (int i = 0; i < length; i++) {

**Serial**.print((char)message[i]);

stMessage += (char)message[i];

}

**Serial**.println();

if (String(topic) == "cedup/texto") {

**Serial**.print("Alterando o LED para ");

if(stMessage == "on"){

**Serial**.println("on");

digitalWrite(ledPin, HIGH); // Liga o led

}

else if(stMessage == "off"){

**Serial**.println("off");

digitalWrite(ledPin, LOW); // desliga o led

}

}

}

void loop() {

delay(10);

if (!client.connected()) {

mqttReconnect();

}

client.loop();

}