IoT Smart Bridge

Assignment Week 3

Shrenik Abrol   20BCE1436

Q. In Wokwi add LED and switch On and Off from Node-Red.

Link:

> https://wokwi.com/projects/367312848712545281

Code:

```cpp
#include <WiFi.h>
#include <PubSubClient.h>
#include "DHT.h"
#define DHTPIN 15
#define DHTTYPE DHT22
#define LED 2

DHT dht (DHTPIN, DHTTYPE);
```

```cpp
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);
```

```cpp
#define ORG "3j24p3"
#define DEVICE_TYPE "abcd"
#define DEVICE_ID "1234"
#define TOKEN "12345678"
String data3;
float h, t;
```

```cpp
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char subscribetopic[] = "iot-2/cmd/command/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
```

```cpp
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback ,wifiClient);
```

```cpp
void setup()
```

```cpp
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
  mqttconnect();
}

void loop()
{

  h = dht.readHumidity();
  t = dht.readTemperature();
  Serial.print("temp:");
  Serial.println(t);
  Serial.print("Humid:");
  Serial.println(h);

  PublishData(t, h);
  delay(4000);
  if (!client.loop()) {
    mqttconnect();
  }
}


void PublishData(float temp, float humid) {
  mqttconnect();
  String payload = "{\"temp\":";
  payload += temp;
  payload += "," "\"Humid\":";
  payload += humid;
  payload += "}";


  Serial.print("Sending payload: ");
  Serial.println(payload);


  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");
    }
    else {
    Serial.println("Publish failed");
  }

}
```

```cpp
void mqttconnect() {
  if (!client.connected()) {
    Serial.print("Reconnecting client to ");
    Serial.println(server);
    while (!!!client.connect(clientId, authMethod, token)) {
      Serial.print(".");
      delay(500);
    }

    initManagedDevice();
    Serial.println();
  }
}
void wificonnect()
{
  Serial.println();
  Serial.print("Connecting to ");
```

```cpp
  WiFi.begin("Wokwi-GUEST", "", 6);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
```

```cpp
void initManagedDevice() {
  if (client.subscribe(subscribetopic)) {
    Serial.println((subscribetopic));
    Serial.println("subscribe to cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED");
  }
}
```

```cpp
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength)
{

  Serial.print("callback invoked for topic: ");
  Serial.println(subscribetopic);
  for (int i = 0; i < payloadLength; i++) {
    data3 += (char)payload[i];
  }
  Serial.println("data: "+ data3);
  if(data3=="lighton")
```

```
{
  Serial.println(data3);
  digitalWrite(LED,HIGH);
}
else
{
  Serial.println(data3);
  digitalWrite(LED,LOW);
}
data3="";
}
```

Screenshots: