SPRINT 2

Date	06 November 2022
Team ID	PNT2022TMID30020
Project Name	Project - Signs with Smart Connectivity for Better Road Safety

SCRIPT :

```
#include <ESP8266WiFi.h>
#include "DHT.h"
#include <ArduinoJson.h>
#include <PubSubClient.h>

// Watson IoT connection details
#define MQTT_HOST "3xr4l4.messaging.internetofthings.ibmcloud.com" //Organization ID.messaging.internetofthings.ibmcloud.com
//change  3xr4l4
#define MQTT_PORT 1883
#define MQTT_DEVICEID "d:3xr4l4:ESP8266:Device1"  //d:Organization ID:Device Type:Device ID
//change  3xr4l4
#define MQTT_USER "use-token-auth"
#define MQTT_TOKEN "Dty58gpPMi9Ll@vU11" // change your auth_id :
#define MQTT_TOPIC "iot-2/evt/status/fmt/json"
#define MQTT_TOPIC_DISPLAY "iot-2/cmd/display/fmt/json"

// Add GPIO pins used to connect devices

#define DHT_PIN 2 // GPIO pin the data line of the DHT sensor is connected to

// Specify DHT11 (Blue) or DHT22 (White) sensor
#define DHTTYPE DHT11


// Add WiFi connection information
char ssid[] = "raspberr";     //  your network SSID (name)
char pass[] = "dayo2022";  // your network password


DHT dht(DHT_PIN, DHTTYPE);


// MQTT objects
```

```
void callback(char* topic, byte* payload, unsigned int length);
WiFiClient wifiClient;
PubSubClient mqtt(MQTT_HOST, MQTT_PORT, callback, wifiClient);

// variables to hold data
StaticJsonDocument<100> jsonDoc;
JsonObject payload = jsonDoc.to<JsonObject>();
JsonObject status = payload.createNestedObject("d");
static char msg[50];

float h = 0.0;
float t = 0.0;



void callback(char* topic, byte* payload, unsigned int length) {
  // handle message arrived
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] : ");

  payload[length] = 0; // ensure valid content is zero terminated so can treat as c-string
  Serial.println((char *)payload);
}

void setup() {
 // Start serial console
  Serial.begin(115200);
  Serial.setTimeout(2000);
  while (!Serial) { }
  Serial.println();
  Serial.println("ESP8266 IBM Cloud Application");

  // Start WiFi connection
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, pass);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi Connected");

  // Start connected devices
```

```
  dht.begin();

  // Connect to MQTT - IBM Watson IoT Platform
  if (mqtt.connect(MQTT_DEVICEID, MQTT_USER, MQTT_TOKEN)) {
    Serial.println("MQTT Connected");
    mqtt.subscribe(MQTT_TOPIC_DISPLAY);

  } else {
    Serial.println("MQTT Failed to connect!");
    ESP.reset();
  }
}

void loop() {
  mqtt.loop();
  while (!mqtt.connected()) {
    Serial.print("Attempting MQTT connection...");
    // Attempt to connect
    if (mqtt.connect(MQTT_DEVICEID, MQTT_USER, MQTT_TOKEN)) {
      Serial.println("MQTT Connected");
      mqtt.subscribe(MQTT_TOPIC_DISPLAY);
      mqtt.loop();
    } else {
      Serial.println("MQTT Failed to connect!");
      delay(5000);
    }
{
      Serial.println("MQTT Publish failed");
    }
  }

  // Pause - but keep polling MQTT for incoming messages
  for (int i = 0; i < 10; i++) {
    mqtt.loop();
    delay(1000);
  }
}
```