

ASSIGNMENT 4

Assignment Date	27 OCT 2022
Student Name	SM. Swethaa
Team ID	TNT2022TMID46761
Project Title	Real-Time River Water Quality Monitoring and Control System
Maximum Marks	2 Marks

QUESTION:

Write code and connections in wokwi for the ultrasonic sensor. Whenever the distance is less than 100 cms send an "alert" to the IBM cloud and display in the device recent events. Upload document with wokwi share link and images of IBM cloud.

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "8egjb2"
#define DEVICE_TYPE "Swethaa"
#define DEVICE_ID "Swethaa123"
#define TOKEN "123456789"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Swethaa/fmt/json";
char topic[] = "iot-2/cmd/home/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);
void publishData();

const int trigpin=5;
const int echopin=18;
String command;
String data="";

long duration;
float dist;

void setup()
```

```

{
    Serial.begin(115200);
    pinMode(led, OUTPUT);
    pinMode(trigpin, OUTPUT);
    pinMode(echopin, INPUT);
    wifiConnect();
    mqttConnect();
}

void loop() {
    bool isNearby = dist < 100;
    digitalWrite(led, isNearby);

    publishData();
    delay(500);

    if (!client.loop()) {
        mqttConnect();
    }
}

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

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

void initManagedDevice() {
    if (client.subscribe(topic)) {
        // Serial.println(client.subscribe(topic));
        Serial.println("IBM subscribe to cmd OK");
    } else {

```

```

        Serial.println("subscribe to cmd FAILED");
    }
}
void publishData()
{
    digitalWrite(trigpin,
LOW);
    digitalWrite(trigpin,
HIGH);
    delayMicroseconds(10)
;
    digitalWrite(trigpin,
LOW);
    duration=pulseIn(echo
pin,HIGH);
    dist=duration*speed/2
; if(dist<100){
        String payload = "{\"Alert
Distance\":\"";payload += dist;
        payload += "}";

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

        if(client.publish(publishTopic, (char*) payload.c_str())) {
            Serial.println("Warning crosses 110cm -- it automatically of
the loop");digitalWrite(led,HIGH);
        }
    }

    if(dist>101 && dist<111){
        String payload = "{\"Normal
Distance\":\"";payload += dist;
        payload += "}";

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

    }

}

Void callback(char*subscribeTopic,byte* payload,unsigned intpayloadLength)
{
    Serial.print("callback invoked for topic:");
    Serial.println(subscribeTo
pic); for(int i=0;
i<payloadLength; i++){

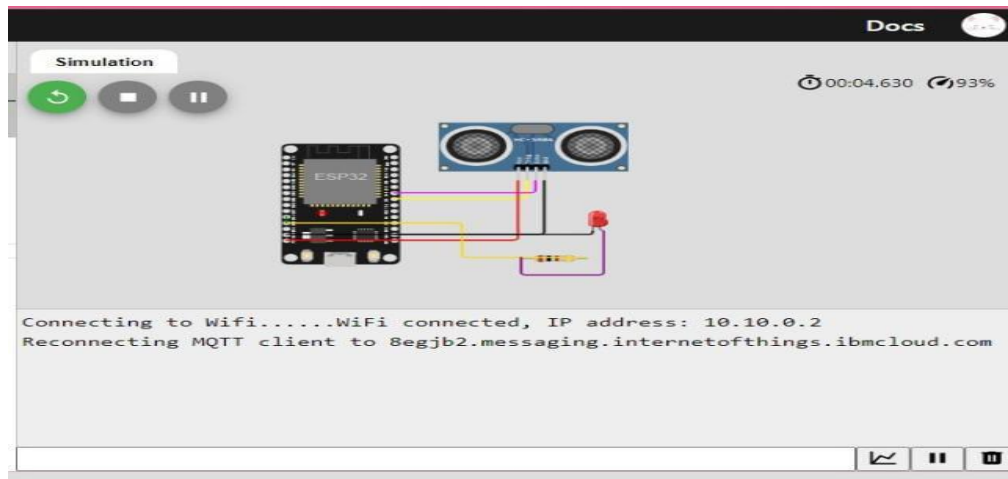
```

```

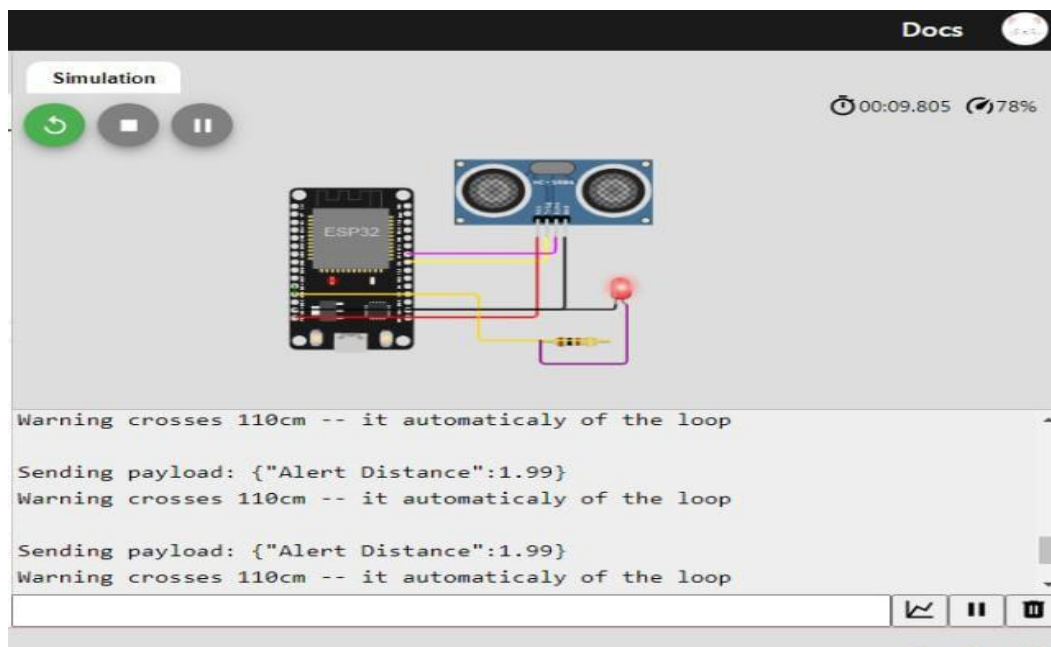
    dist += (char)payload[i];
}
Serial.println("data:" + data3);
if(data3=="lighton"){
    Serial.println(data3);
    digitalWrite(led,HIGH);
}
data3="";
}

```

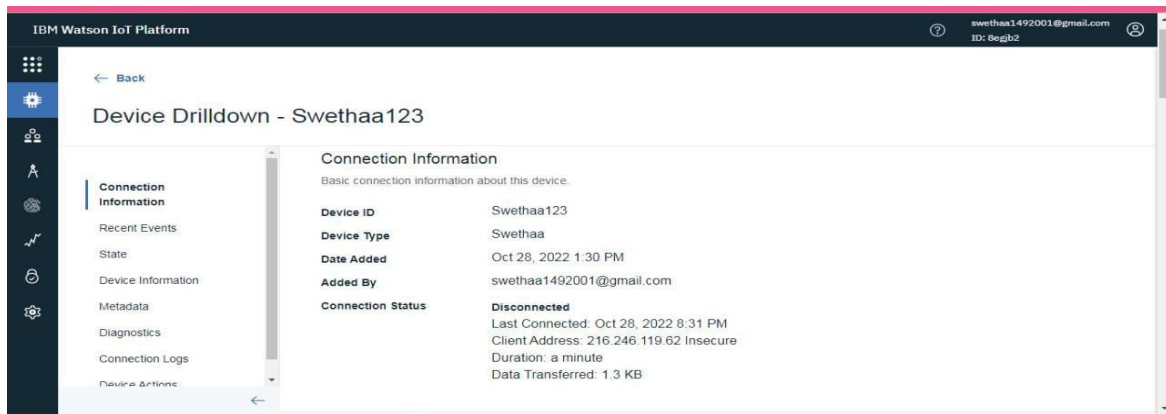
OUTPUT:



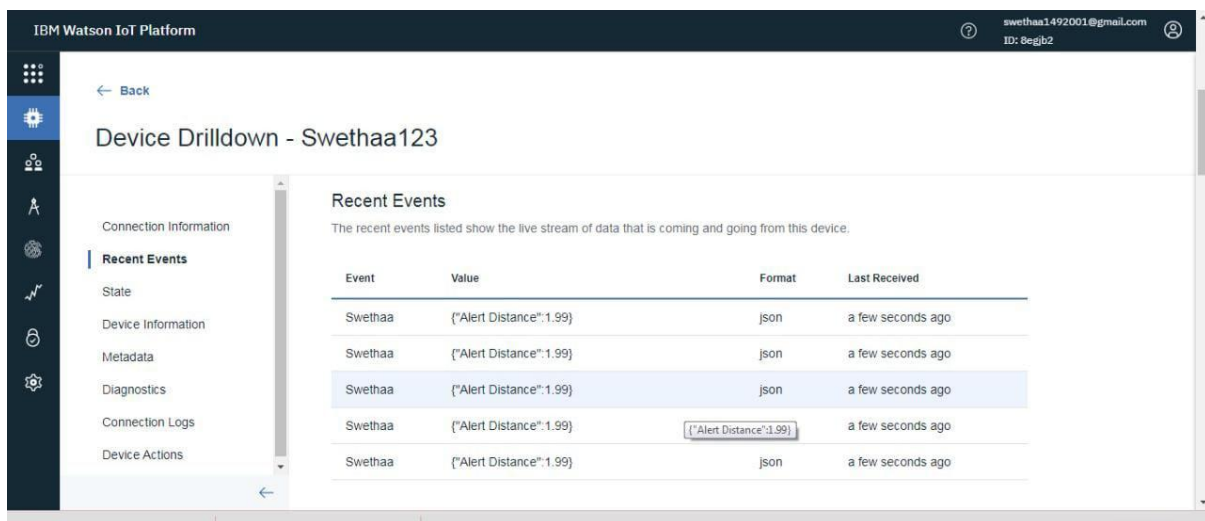
While Distance is greater than 100cm there is no alert message in the IBM cloud.



When the distance is less than 100cm alert message will appear in the IBM cloud.



IBM CLOUD OUTPUT:



WOKWI URL:

<https://wokwi.com/projects/346756519381959252>