

```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "ki82qi"
#define DEVICE_TYPE "athithiya"
#define DEVICE_ID "athithiya123"
#define TOKEN "123456789"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/athithiya/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(echopin, HIGH);
    dist=duration*speed/2;
    if(dist<100){
        String payload = "{\"Alert Distance\"":";
        payload += dist;
    }
}
```

```
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 automaticaly 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);
}
```

```

32-dht22.ino  diagram.json  libraries.txt  Library Manager
1  #include <WiFi.h>
2  #include <PubSubClient.h>
3  WiFiClient wifiClient;
4  String data3;
5  #define ORG "ki82q1"
6  #define DEVICE_TYPE "athithiya"
7  #define DEVICE_ID "athithiya123"
8  #define TOKEN "123456789"
9  #define speed 0.034
10 #define led 14
11 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
12 char publishTopic[] = "iot-2/evt/athithiya/fmt/json";
13 char topic[] = "iot-2/cmd/home/fmt/String";
14 char authMethod[] = "use-token-auth";
15 char token[] = TOKEN;
16 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
17 PubSubClient client(server, 1883, wifiClient);
18 void publishData();
19
20 const int trigpin=5;
21 const int echopin=18;
22 String command;
23 String data="";
24
25 long duration;
26 float dist;

```

Simulation

00:12.762 46%

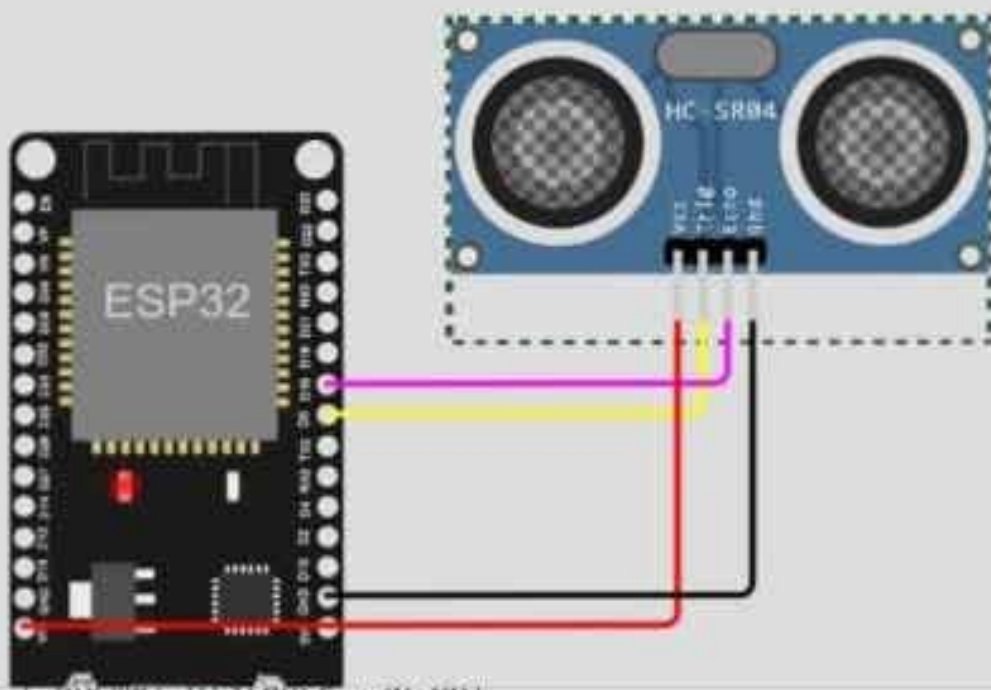
Warning crosses 110cm -- it automatically of the loop

Sending payload: {"Alert Distance":51.97}

Warning crosses 110cm -- it automatically of the loop

Sending payload: {"Alert Distance":51.97}

Warning crosses 110cm -- it automatically of the loop



Sending payload: {"Normal Distance":89.95}

Publish OK

Sending payload: {"Normal Distance":89.95}

Publish OK

Sending payload: {"Normal Distance":89.95}

Publish OK

[← Back](#)

Device Drilldown - athithiya123

Device Credentials

- Connection Information
- Recent Events
- State
- Device Information
- Metadata
- Diagnostics
- Connection Logs
- Device Actions

Device Credentials

You registered your device to the organization. Add these credentials to the device to connect it to the platform. After the device is connected, you can navigate to view connection and event details.

Organization ID	ki82qi
Device Type	athithiya
Device ID	athithiya123
Authentication Method	use-token-auth
Authentication Token	123456789

 Authentication failed. You will need to re-register the device to generate a new authentication token.

meet.google.com is sharing your screen. [Stop sharing](#) [Hide](#)

[← Back](#)

Device Drilldown - athithiya123

- Device Credentials
- Connection Information
- Recent Events**
- State
- Device Information
- Metadata
- Diagnostics
- Connection Logs
- Device Actions

Recent Events

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
athithiya	{"Alert Distance":51.97}	json	a few seconds ago
athithiya	{"Alert Distance":51.97}	json	a few seconds ago
athithiya	{"Alert Distance":51.99}	json	a few seconds ago
athithiya	{"Alert Distance":51.97}	json	a few seconds ago
athithiya	{"Alert Distance":51.97}	json	a few seconds ago