

## ASSIGNMENT 4

DATE	3 NOV 2022
NAME	A.KAVIYA
TEAM ID	PNT2022TMID49661
PROJECT NAME	Project-IOT Based Safety Gadget For Child Safety Monitoring & Notification

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

### CODE

```
#include <WiFi.h>
#include <PubSubClient.h> wifiClient WiFiClient;
#define ORG "q6wu16"
#define DEVICE_TYPE "NodeMCU"
#define DEVICE_ID "Test"
#define TOKEN "07Wygbmfhub0vU1bE5"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/shreedharen/fmt/json";
char topic [] = "iot-2/cmd/led/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
char PubSubClient client (server, 1883, WiFiClient);
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");
}
}

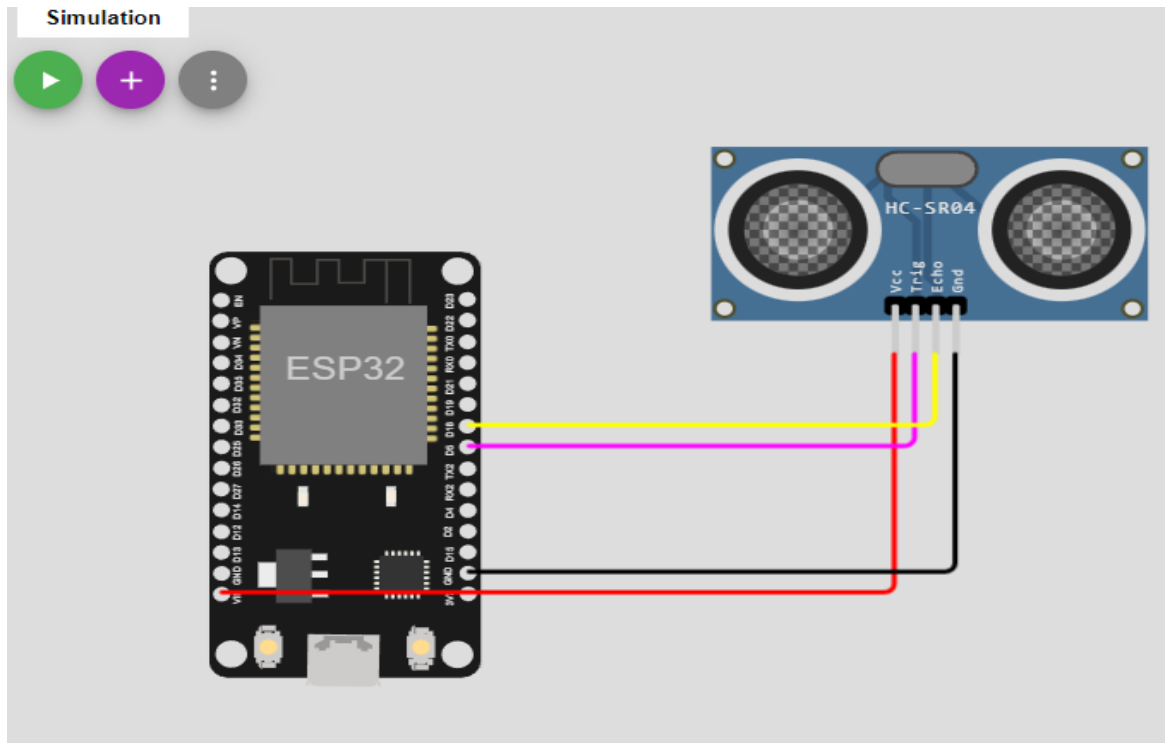
```

```

}
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;
    payload += "}"; Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
    if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish OK");
    }
    }
    if(Dist>100){
    String payload = "{\"Distance\":\"";payload +=Dist;
    payload += "}"; Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
    if(client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish OK");
    }else {
    Serial.println("Publish FAILED");
    }
    }
}

```

## CONNECTION



WOKWI SAVE SHARE esp32-dht22.ino Docs SIGN IN  
by urish

esp32-dht22.ino • diagram.json • libraries.txt • Library Manager

```
1 #include <WiFi.h>
2 #include <PubSubClient.h> WiFiClient;
3 #define ORG "q6wu16"
4 #define DEVICE_TYPE "NodeMCU"
5 #define DEVICE_ID "Test"
6 #define TOKEN "O7Wygbmfnub0vU1bE5"
7 #define speed 0.034
8 #define led 14
9 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
10 char publishTopic[] = "iot-2/evt/shreedharen/fmt/json";
11 char topic [] = "iot-2/cmd/led/fmt/String";
12 char authMethod[] = "use-token-auth";
13 char token[] = TOKEN;
14 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
15 char PubSubClient client (server, 1883, WiFiClient);
16 const int trigpin=5;
17 const int echopin=18;
18 String command;
19 String data="";
20 long duration;
21 float Dist;
22 void setup()
23 {
24   Serial.begin(115200);
25   pinMode(led, OUTPUT);
```

Simulation