

# Assignment 4

## WOKWI PROGRAM

Student Name	Reju kannan
Student Roll Number	960219106701
Maximum Mark	2 marks

### CODE:

```
#include<WiFi.h>
#include<PubSubClient.h>
WiFiClient wificlient;
String data 3;
#define ORG "4yi0vc"
#define DEVICE_TYPE "nodeMcu"
#define DEVICE-ID "Assignment4"
#define TOKEN "123456789"
#define speed 0.034
#define led 14
char server[] = ORG".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/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 tripin =5;
const int echopin =18;
String command;
String data="";
long duration;
float dist;
void setup()
{
  Serial.begin(115200);
  pinMode(led,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_CONNECT) {
    delay(500);
    Serial.print(".");
  }
  Serial.print("WIFI connected ,IP address:");
}
void mqttConnect(){
  if (!client.connect() {
    Serial.print("Reconnecting MQTT client to"); Serial.println(server);
    while (!client.connect(clientId,authMethod,token)){
      Serial.print(".");
      delay(500);
    }
    intiManageddevice();
    Serial.println();
  }
}
void initManagedDevice() {
  if (client.subscribe(topic)) {
    // serial.priintln(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 = "{/\"Normal 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>101 && dist<111){
        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);
        }else {
            Serial.println("publish FAILED");
        }
    }
}

void callback(char* subscribeTopic, byte* payload, unsigned int
payloadLength){
    Serial.print("callback invoked for topic:");

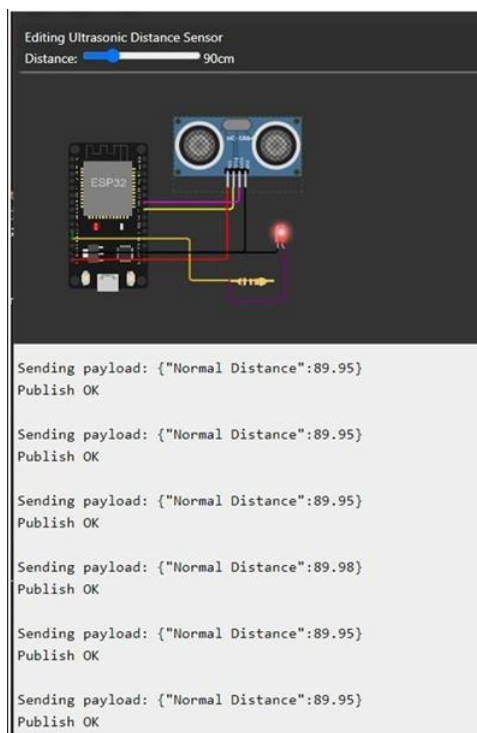
```

```

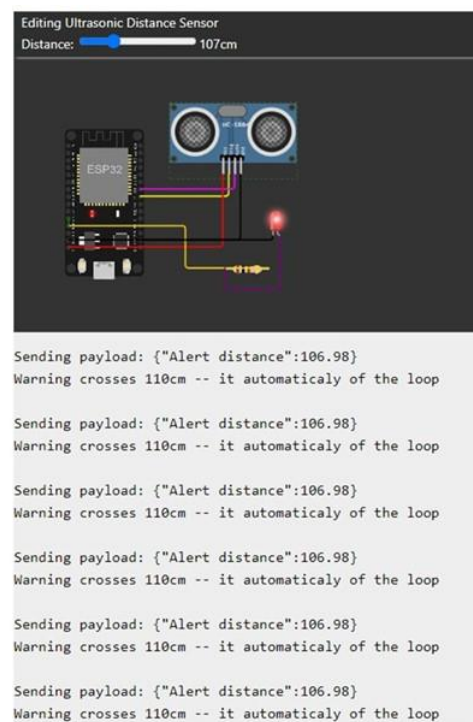
Serial.println(subscribeTopic);
for(int i=0; i<payloadLength; i++) {
    dist += (char)payload[i];
}
Serial.println("data:" + data3);
if(data3== "lighton"){
    Serial.println("data3");
    digitalWrite(led,HIGH),
}
Data3= "";
}

```

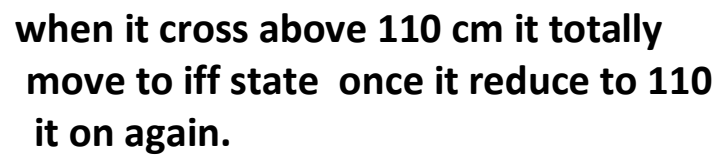
## OUTPUT:



**1)When distance under 100 cm  
it will show normal distance**



**2) When diatance cross 100  
cm it will show ALERT  
with warning message  
distance**



## 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
Data	["Normal Distance":89.95]	json	a few seconds ago
Data	["Normal Distance":89.95]	json	a few seconds ago
Data	["Normal Distance":89.95]	json	a few seconds ago
Data	["Normal Distance":89.95]	json	a few seconds ago
Data	["Normal Distance":89.95]	json	a few seconds ago

## 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
Data	["Alert distance":"106.98"]	json	a few seconds ago
Data	["Alert distance":"107.03"]	json	a few seconds ago
Data	["Alert distance":"106.98"]	json	a few seconds ago
Data	["Alert distance":"106.98"]	json	a few seconds ago
Data	["Alert distance":"106.98"]	json	a few seconds ago

