# ASSIGMNENT-4

**ASSIGNMENTQUESTION:**

Write code and connections in Wokwi for ultrasonic sensor. Whenever distance is less than 100 cm send"alert" to IBM cloud and display in device recent events Upload document with Wokwi share link andimages ofIBMcloud.

# CODING:

#include<WiFi.h>#include <PubSubClient.h>WiFiClientwifiClient;Stringdata3;

#defineORG"lclqxu"

#defineDEVICE\_TYPE"1234"

#define DEVICE\_ID "Nhidhees-Device"#define TOKEN "QFgbWeRmmxLBAIq2De"#definespeed 0.034

#defineled14

charserver[] = ORG ".[messaging.internetofthings.ibmcloud.com](http://messaging.internetofthings.ibmcloud.com/)";charpublishTopic[] ="iot-2/evt/shreedharen/fmt/json";

chartopic[] = "iot-2/cmd/led/fmt/String";char authMethod[] = "use-token-auth";chartoken[]=TOKEN;

charclientId[]="d:"ORG":"DEVICE\_TYPE":"DEVICE\_ID;

PubSubClientclient(server, 1883, wifiClient);constinttrigpin=5;

constintechopin=18;

Stringcommand;

Stringdata="";

longduration;floatdist;

voidsetup()

{

**Serial**.begin(115200);pinMode(led, OUTPUT);pinMode(trigpin, OUTPUT);pinMode(echopin, INPUT);

wifiConnect();mqttConnect();

}

voidloop(){

boolisNearby = dist<100;digitalWrite(led,isNearby);

publishData();delay(500);

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

}

}

voidwifiConnect()

{

**Serial**.print("Connecting to "); **Serial**.print("Wifi");WiFi.begin("Wokwi-GUEST","", 6);

while(WiFi.status() != WL\_CONNECTED) {delay(500);

**Serial**.print(".");

}

**Serial**.print("WiFiconnected,IPaddress:");**Serial**.println(WiFi.localIP());

}

voidmqttConnect(){

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();

}

}

voidinitManagedDevice() {if(client.subscribe(topic)){

//Serial.println(client.subscribe(topic));

**Serial**.println("IBMsubscribetocmdOK");

}else{

**Serial**.println("subscribetocmdFAILED");

}

}

voidpublishData()

{

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("PublishOK");

}

}

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("PublishOK");

}else{

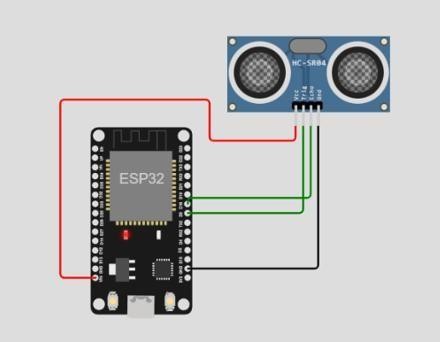
**Serial**.println("PublishFAILED");

}

}

}

**CONNECTION:**



**OUTPUT:**

