# Assignment-4

**NAME;PRAKASH.K,KEERTHI ABINOV.G,SABARISHWARAN.K,PRAVIN.T.V**

**ROLLNUMBER :714019106075,714019106046,714019106049,714019106079**

Write code and connections in wokwi for ultrasonic sensor. Whenever distance is lessthan100 cmssend "alert" toibm cloudanddisplay indevicerecent events.

Uploaddocument withwokwi sharelink andimages ofibmcloud

CODE

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

#defineORG"4yi0vc"

#define DEVICE\_TYPE "nodeMcu"#define DEVICE\_ID "Assignment4"#defineTOKEN"123456789"

#definespeed0.034

#defineled 14

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";charpublishTopic[]="iot-2/evt/Data/fmt/json";

char topic[] = "iot-2/cmd/home/fmt/String";charauthMethod[]="use-token-auth";

chartoken[]=TOKEN;

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

PubSubClient client(server, 1883, wifiClient);voidpublishData();

constinttrigpin=5;constintechopin=18;String command;Stringdata="";

long duration;floatdist;

voidsetup()

{

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();

}

}

voidwifiConnect(){

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());

}

voidmqttConnect(){

if (!client.connected()) {Serial.print("ReconnectingMQTTclientto");

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("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 = "{\"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 automaticaly of the loop");digitalWrite(led,HIGH);

}else{

Serial.println("PublishFAILED");

}

}

}

void callback(char\* subscribeTopic, byte\* payload, unsigned intpayloadLength){Serial.print("callback invoked fortopic:");

Serial.println(subscribeTopic);for(inti=0; i<payloadLength; i++){dist +=(char)payload[i];

}

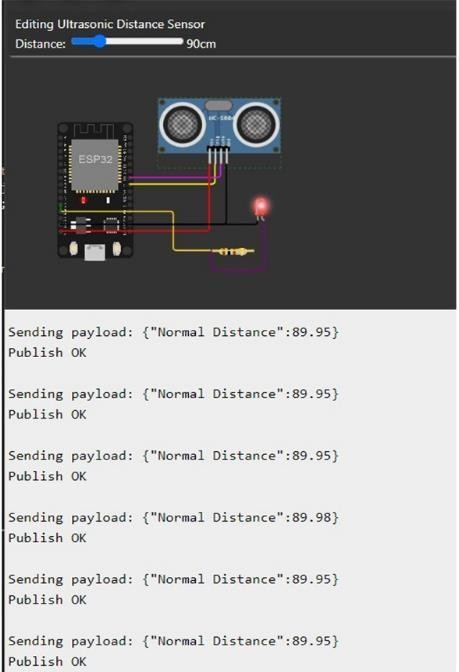
Serial.println("data:"+ data3);if(data3=="lighton"){Serial.println(data3);digitalWrite(led,HIGH);

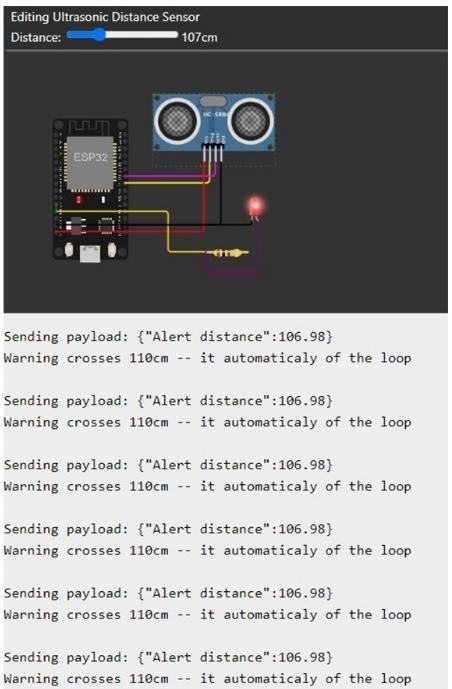
}

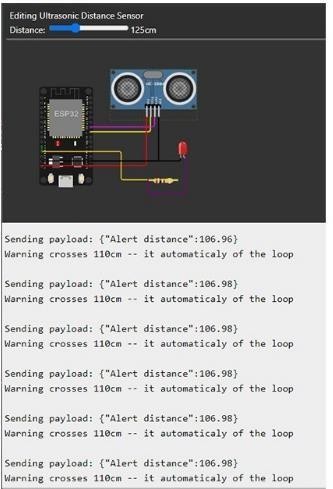
data3="";

}

# NODE-RED







**OUTPUT**

