ASSIGNMENT 4

Date	24/10/22
Name	Ashika.G
Team ID	B2-2M4E
Project Name	Smart Waste Management System For Metropolitan Cities.

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

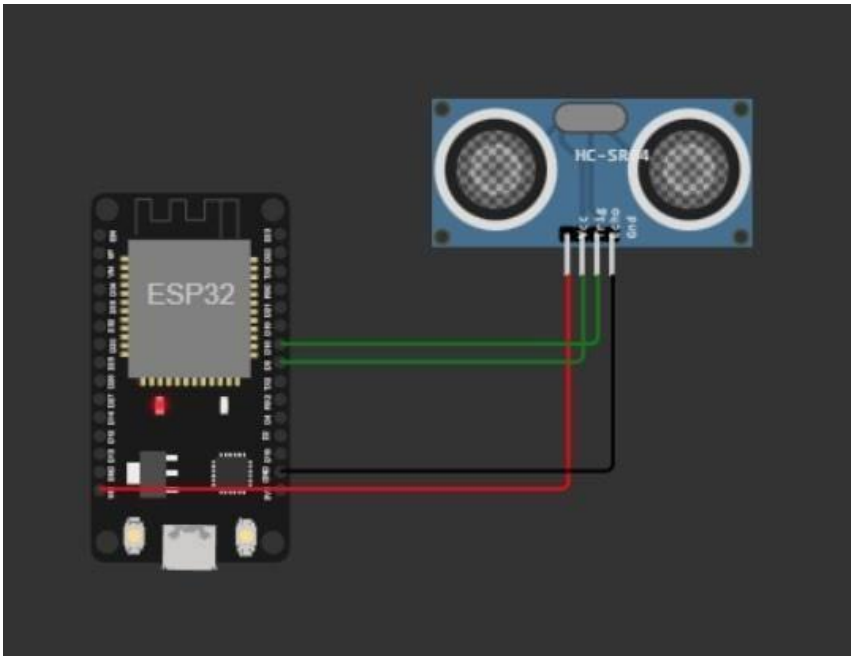
CODE:

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

#defineORG"nhpwjc"
#defineDEVICE_TYPE"NodeMCU"#defineD
EVICE_ID"USEYOURID"#defineTOKEN"USE
YOUR TOKEN"
#definespeed0.034

charserver[]
=ORG".messaging.internetofthings.ibmcloud.com";charpublishTo
pic[] ="iot-2/evt/Data/fmt/json";chartopic[] ="iot-
2/cmd/home/fmt/String";charauthMethod[]="use-token-
auth";chartoken[] =TOKEN;
charclientId[]="d:"ORG":"DEVICE_TYPE":"DEVICE_ID;

PubSubClient   client(server,   1883,wifiClient);void
publishData();
 constinttrigpin=5;
constintechopin=18;S
tringcommand;Stringd
ata="";
 longduration
;floatdist;

 voidse
tup()
{
  Serial.begin(115200);pinM
  ode(trigpin,OUTPUT);
```

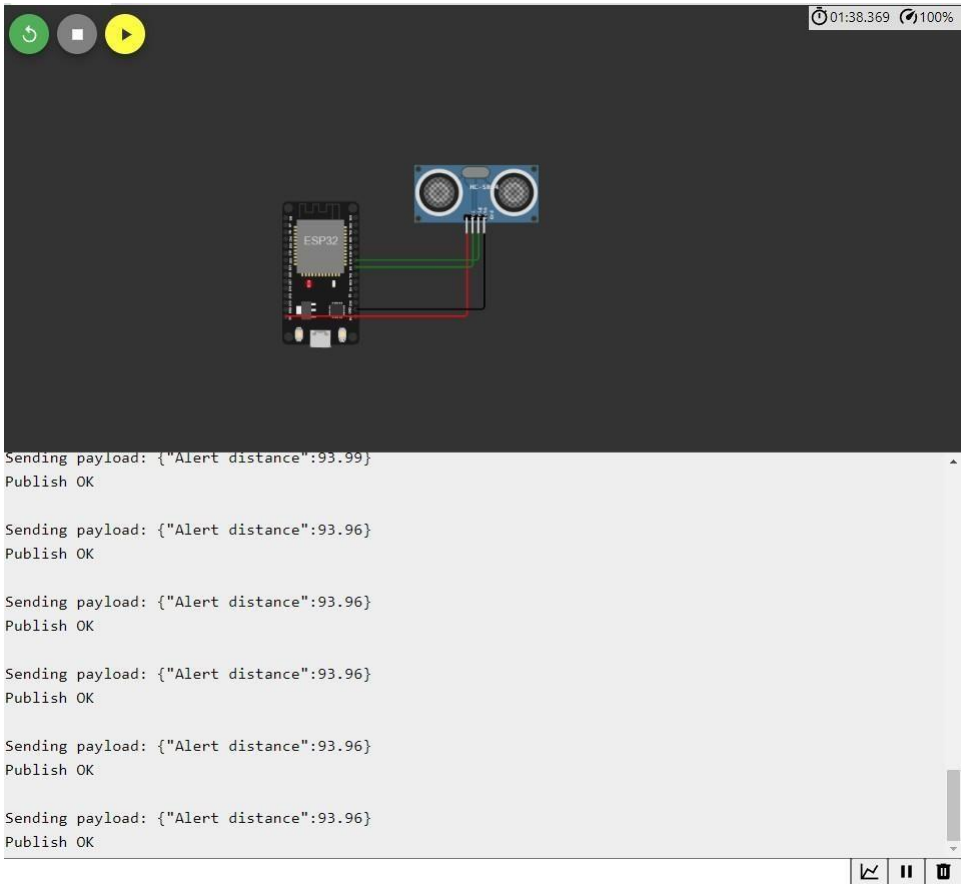
```
  pinMode(echopin, INPUT);
  wifiConnect(); mqttConnect();
} void loop() {

  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("subscribe to
    cmd OK");
  } else {
    Serial.println("subscribe to cmd FAILED"); } }
void publishData()
{ digitalWrite(trigpin,LOW);
  digitalWrite(trigpin,HIGH);
```

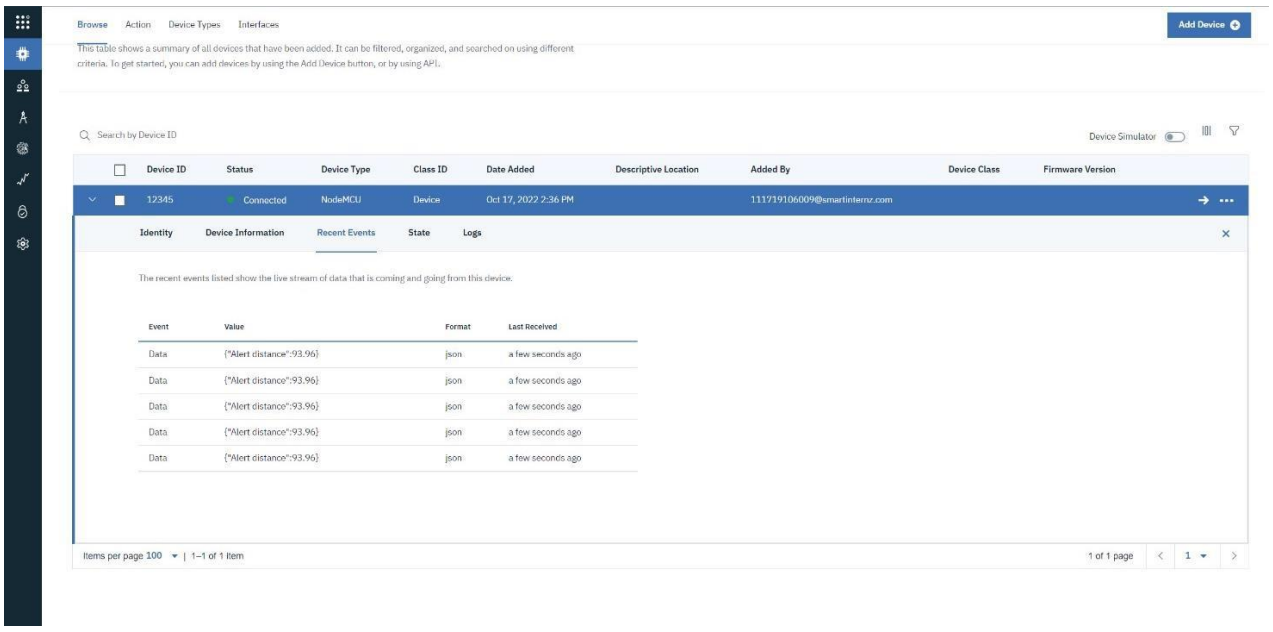
```
  delayMicroseconds(10);digitalWrite
  (trigpin,LOW);duration=pulseIn(ech
  opin,HIGH);dist=duration*speed/2;i
  f(dist<100){
    Stringpayload="{\"Alertdistance\":";payload+=d
    ist;
    payload+="}";Serial.print("\n");Se
    rial.print("Sendingpayload:");
    Serial.println(payload);
    if(client.publish(publishTopic,(char*)payload.c_str())){
      Serial.println("PublishOK");
    }else{
      Serial.println("PublishFAILED");
    }
  }
}
```

CONNECTIONS:



OUTPUT:


01:38.369
100%
ESP32
Sending payload: {"Alert distance":93.99}
Publish OK
Sending payload: {"Alert distance":93.96}
Publish OK
Sending payload: {"Alert distance":93.96}
Publish OK
Sending payload: {"Alert distance":93.96}
Publish OK
Sending payload: {"Alert distance":93.96}
Publish OK
Sending payload: {"Alert distance":93.96}
Publish OK


Browse
Action
Device Types
Interfaces
Add Device
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
Added By
Device Class
Firmware Version
12345
Connected
NodeMCU
Device
Oct 17, 2022 2:36 PM
111719106009@smartinternz.com
Identity
Device Information
Recent Events
State
Logs
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":93.96}
json
a few seconds ago
Items per page 100
1–1 of 1 item
1 of 1 page