

Smart Waste Management System for Metropolitan Cities

ASSIGNMENT 4:

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 "nhpwjc"
#define DEVICE_TYPE "NodeMCU"
#define DEVICE_ID "USE YOUR ID"
#define TOKEN "USE YOUR TOKEN"
#define speed 0.034

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 trigpin=5;
const int echopin=18;
String command;
String data="";

long duration;
float dist;

void setup()
{
  Serial.begin(115200);
  pinMode(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);
}

```

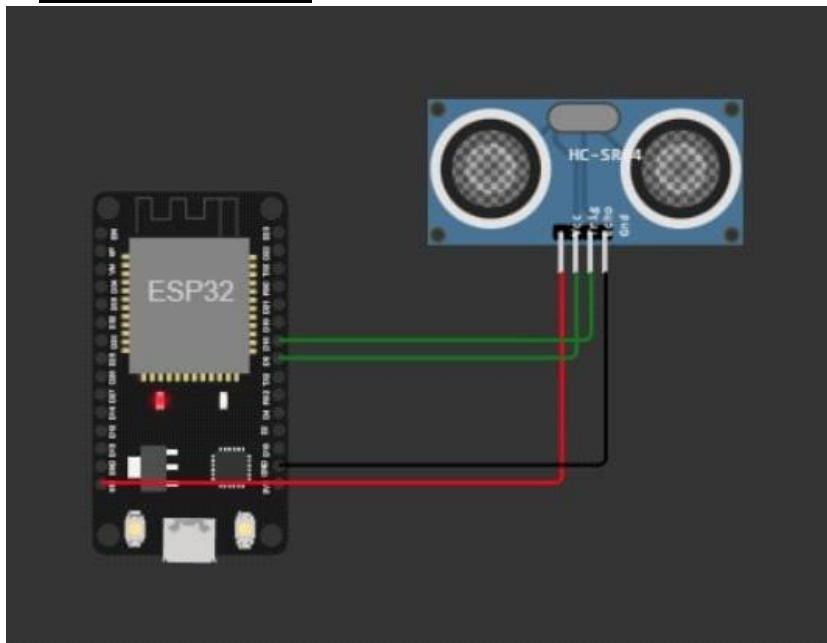
```

delayMicroseconds(10);
digitalWrite(trigpin,LOW);
duration=pulseIn(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");
    } else {
        Serial.println("Publish FAILED");
    }
}
}
}

```

CONNECTIONS:



WOKWI LINK:

<https://wokwi.com/projects/346665867630608980>

OUTPUT:

The screenshot displays the IBM Watson IoT Platform interface. At the top, there's a dark header with navigation icons and a status bar showing '01:38.369' and '100%'. Below the header, a large black area shows a simulated device (ESP32) connected to a blue sensor module. The main content area is a light gray grid with a text overlay: 'Sending payload: {"Alert distance":93.99}' followed by 'Publish OK'. This sequence is repeated five times with varying 'Alert distance' values (93.99, 93.96, 93.96, 93.96, 93.96). At the bottom, a dark blue sidebar contains navigation icons. The main panel shows a table of devices with columns: Device ID, Status, Device Type, Class ID, Date Added, Descriptive Location, Added By, Device Class, and Firmware Version. A device with ID 12345 is highlighted, showing a status of 'Connected' and a device type of 'NodeMCU'. Below the table, a 'Recent Events' section shows a list of events with columns: Event, Value, Format, and Last Received. The events are all 'Data' events with the value '{"Alert distance":93.96}' and a format of 'json', received 'a few seconds ago'. The bottom of the interface shows pagination controls: 'Items per page 100' and '1 of 1 page'.

IBM Watson IoT Platform

111719106009@smarterintenz.com
ID: nhpage

111719106009@smarterintenz.com

12345

Connected

NodeMCU

Device

Oct 17, 2022 2:36 PM

111719106009@smarterintenz.com

Device Class

Firmware Version

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
Data	{"Alert distance":93.96}	json	a few seconds ago
Data	{"Alert distance":93.96}	json	a few seconds ago
Data	{"Alert distance":93.96}	json	a few seconds ago
Data	{"Alert distance":93.96}	json	a few seconds ago

Items per page 100 | 1-1 of 1 item

1 of 1 page