Assignment - 4
Wowki & IBM Cloud

Assignment Date	27 October 2022
Student Name	Rahul raj.K
Student Roll Number	92172019104118
Maximum Marks	2 Marks

Question-1:

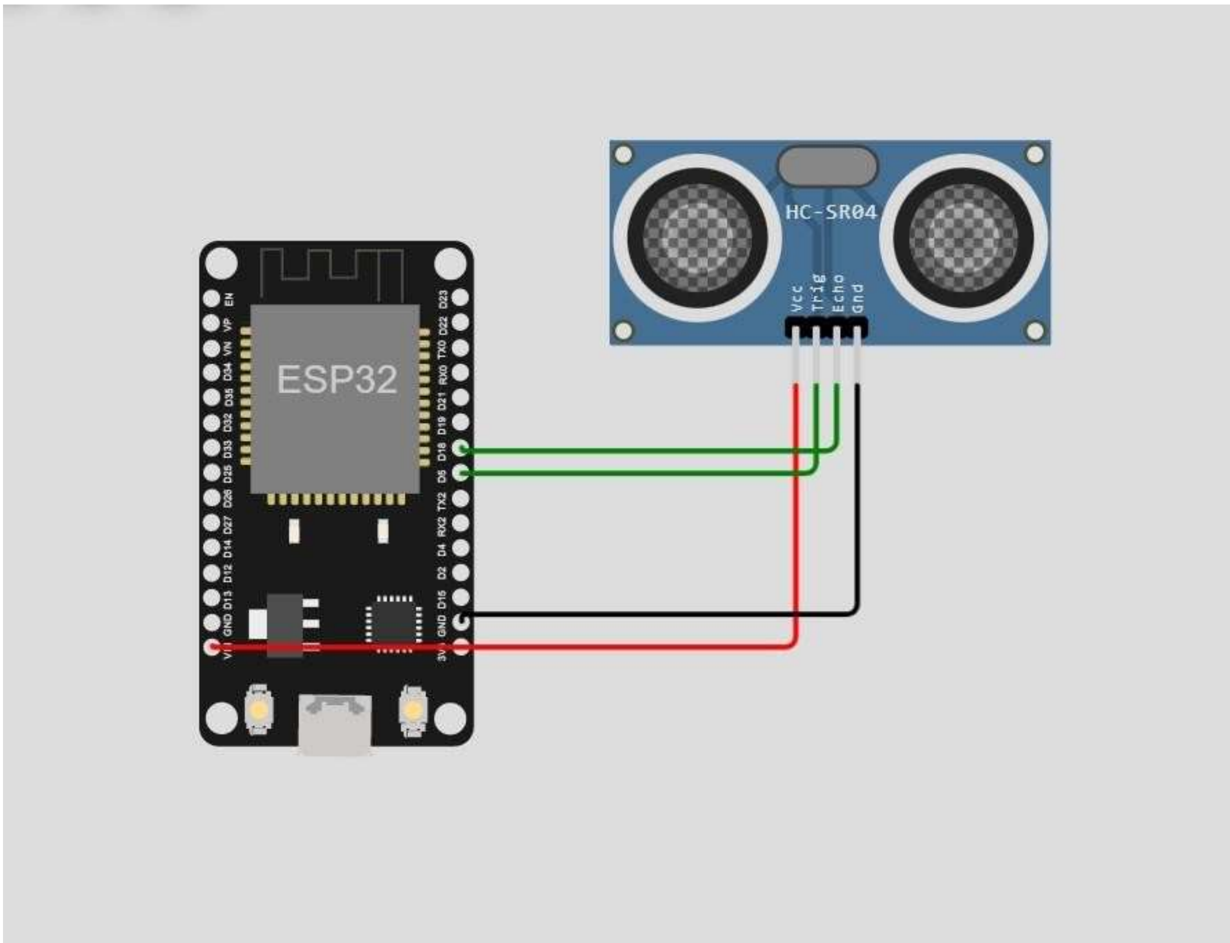
Write code and connections in wowki for the ultrasonic sensor. Whenever the distance is less than 100cms sent "alert" to IBM cloud and display in device recent events.

Code:

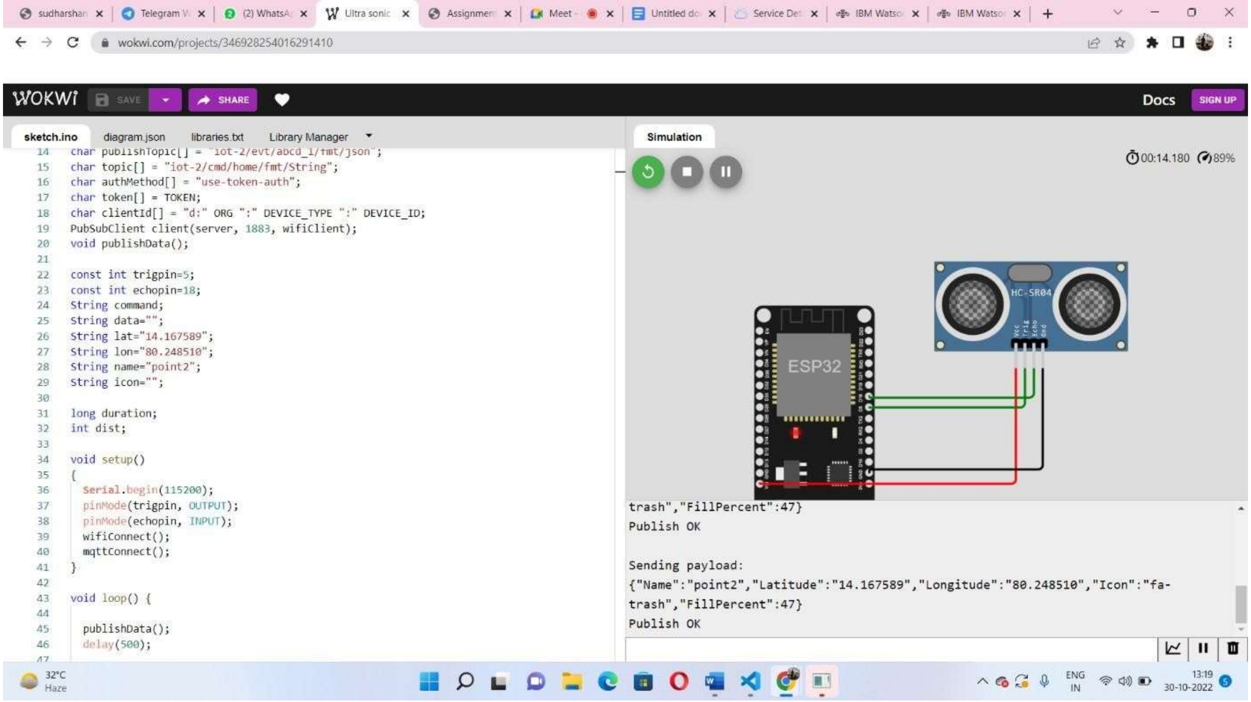
```
#include <WiFi.h>
#include <PubSubClient.h>
#include <ArduinoJson.h>

WiFiClient wifiClient;

#define ORG "42cic2"
#define DEVICE_TYPE "sudharshanlaptop"
#define DEVICE_ID "12345"
#define TOKEN "*zSbeu!vc(!+Xn@520 "
#define speed 0.034
 char server[] = ORG
".messaging.internetofthings.ibmcloud.com"; char publishTopic[]
= "iot-2/evt/abcd_1/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="";
String lat="14.167589";
String lon="80.248510";
String name="point2";
String icon="";
```

```
 long
duration; int
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(1000);
    }
    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){
dist=100-dist;
icon="fa-trash";
}else{     dist=0;
icon="fa-trash-o";
  }
  DynamicJsonDocument doc(1024);
String payload;   doc["Name"]=name;
doc["Latitude"]=lat;
doc["Longitude"]=lon;
doc["Icon"]=icon;
doc["FillPercent"]=dist;
serializeJson(doc, payload);
delay(3000);
  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:


ESP32
EN VP VN D34 D35 D32 D33 D25 D26 D27 D14 D12 D13 GND VIN
D23 D22 TX0 RX0 D21 D19 D18 D5 TX2 RX2 D4 D2 D15 GND 3V3
HC-SR04
Vcc
Trig
Echo
Gnd

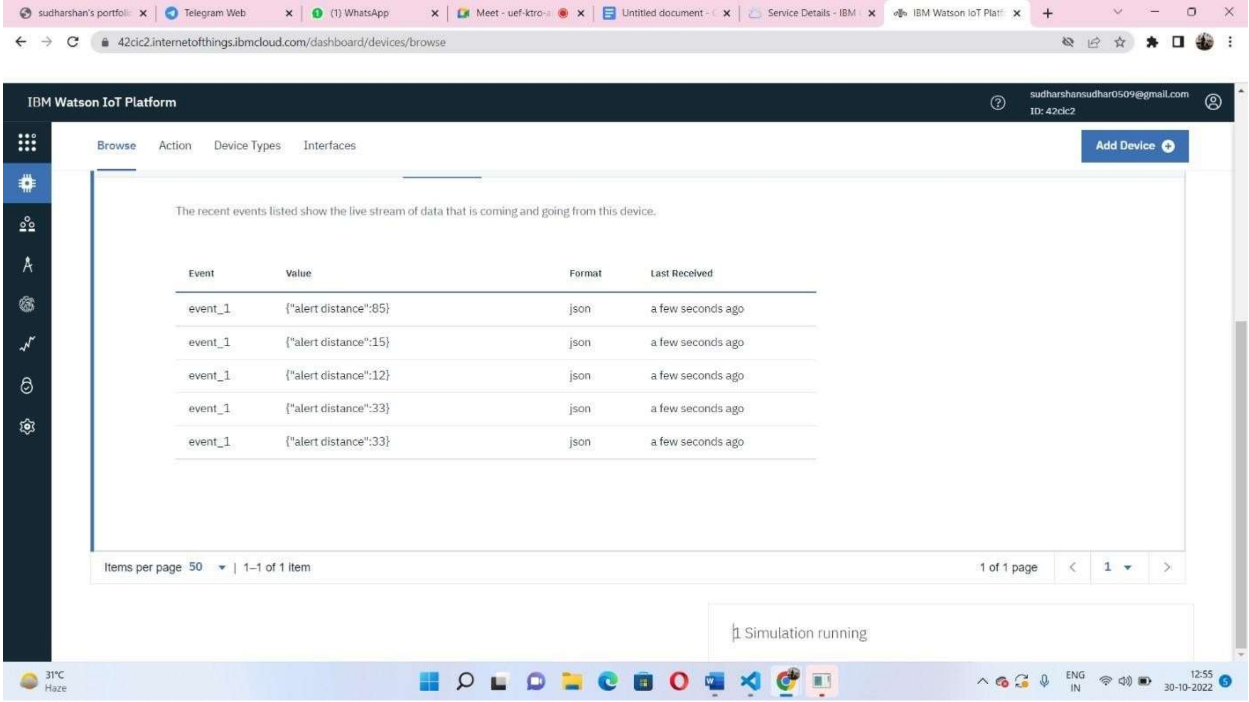
Output:



```
char publishTopic[] = "iot-2/evt/abcd_1/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="";
String lat="14.167589";
String lon="80.248510";
String name="point2";
String icon="";

long duration;
int dist;

void setup()
{
  Serial.begin(115200);
  pinMode(trigpin, OUTPUT);
  pinMode(echopin, INPUT);
  wifiConnect();
  mqttConnect();
}

void loop() {

  publishData();
  delay(500);
```

```
trash","FillPercent":47}
Publish OK

Sending payload:
{"Name":"point2","Latitude":"14.167589","Longitude":"80.248510","Icon":"fa-
trash","FillPercent":47}
Publish OK
```

Output:(IBM Cloud)


42cic2.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
sudharshansudhar0509@gmail.com
ID: 42cic2
Browse
Action
Device Types
Interfaces
Add Device

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
event_1	{"alert distance":85}	json	a few seconds ago
event_1	{"alert distance":15}	json	a few seconds ago
event_1	{"alert distance":12}	json	a few seconds ago
event_1	{"alert distance":33}	json	a few seconds ago
event_1	{"alert distance":33}	json	a few seconds ago

Items per page 50 | 1–1 of 1 item   1 of 1 page 1

Simulation running

Link :https://wokwi.com/projects/346857404558738004