Assignment -4

Assignment Date	21october 2022
Student Name	Mr. MAHESHKUMAR S
Student Roll Number	513419106021
Maximum Marks	2marks

Question-1:

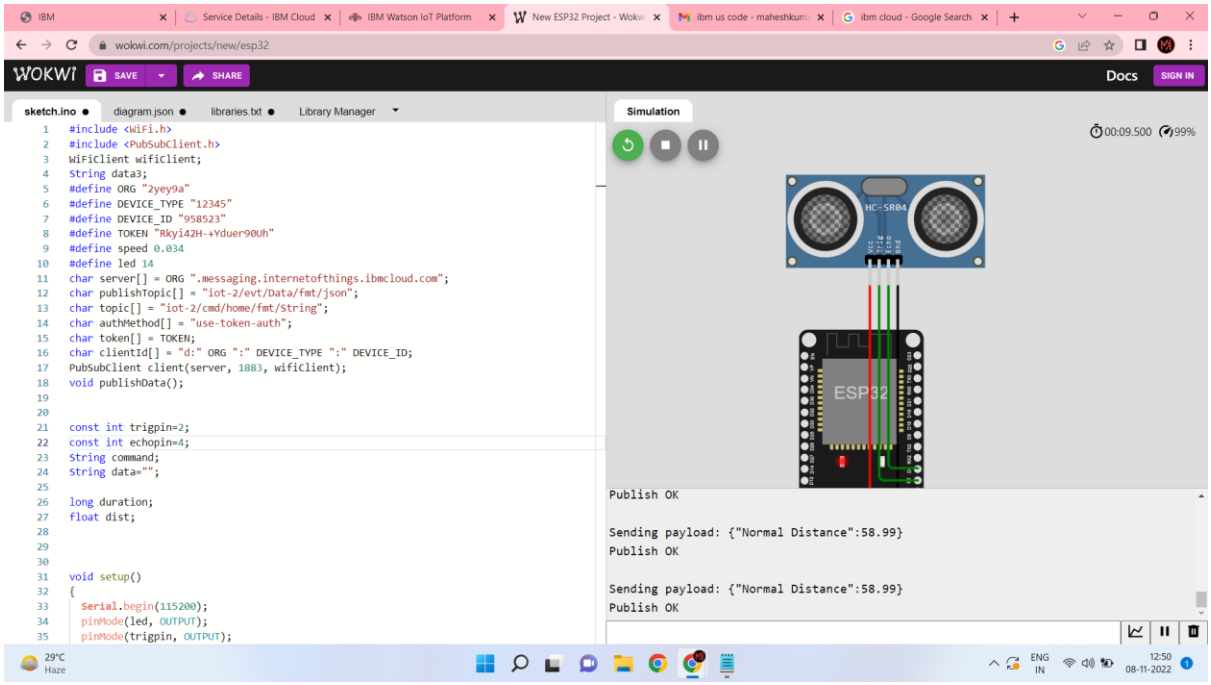
Write code and connection in wokwi for the ultrasonic sensor. Whenever the distance is less than 100cms send “alert” to the IBM cloud and display recent events.

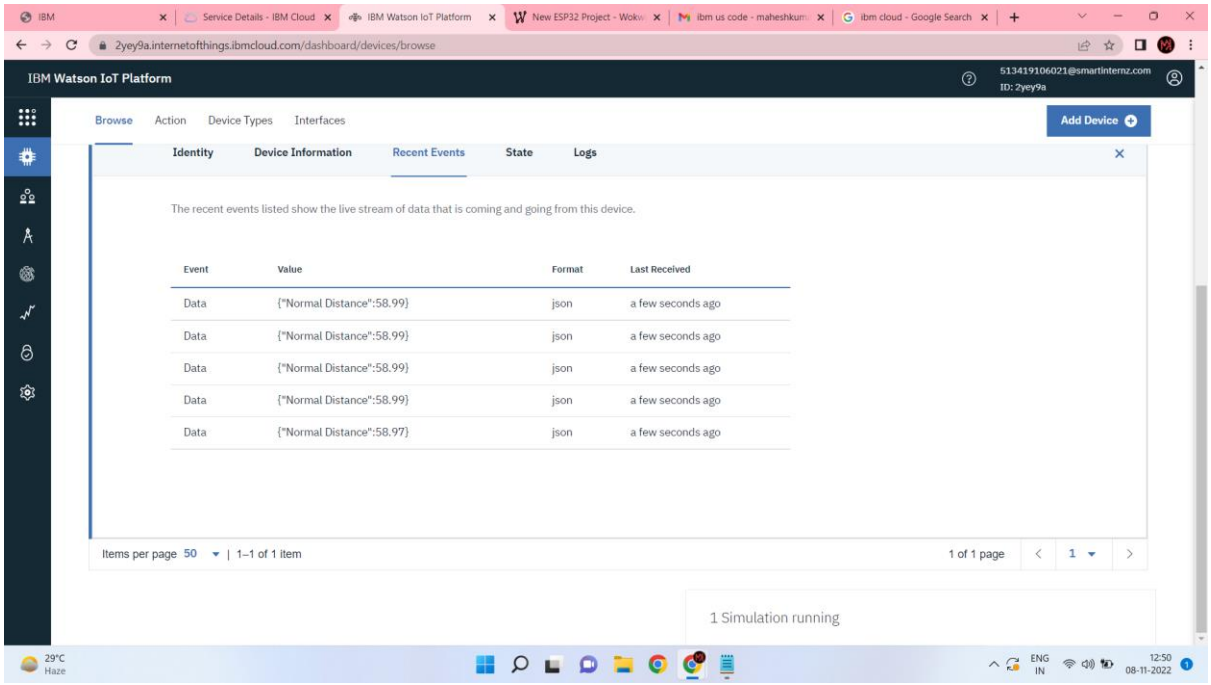
Solution:

```
#include <WiFi.h>

#include <PubSubClient.h>

WiFiClient wifiClient;
String data3;
#define ORG "2yey9a"
#define DEVICE_TYPE "12345"
#define DEVICE_ID "958523"
#define TOKEN "Rkyi42H-+Yduer90Uh"
#define speed 0.034
#define led 14
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=2;
const int echopin=4;
String command;
String data="";

long duration;
float dist;



void setup()
{
  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();
  }
}

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("IBM 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 = "{\"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("Publish FAILED");
    }

  }


  }
  void callback(char* subscribeTopic, byte* payload, unsigned int payloadLength){
  Serial.print("callback invoked for topic:");
  Serial.println(subscribeTopic);
  for(int i=0; i<payloadLength; i++){
    dist += (char)payload[i];
  }
  Serial.println("data:"+ data3);
  if(data3=="lighton"){
    Serial.println(data3);
    digitalWrite(led,HIGH);
  }
  data3="";
}
```


wokwi.com/projects/new/esp32
WOKWI
SAVE
SHARE
Docs
SIGN IN
sketch.ino
diagram.json
libraries.txt
Library Manager
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
String data3;
#define ORG "2yey9a"
#define DEVICE_TYPE "12345"
#define DEVICE_ID "958523"
#define TOKEN "Rkyi42H-+Yduer90Uh"
#define speed 0.034
#define led 14
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=2;
const int echopin=4;
String command;
String data="";
long duration;
float dist;
void setup()
{
Serial.begin(115200);
pinMode(led, OUTPUT);
pinMode(trigpin, OUTPUT);
Simulation
00:09.500
99%
HC-SR04
ESP32
Publish OK
Sending payload: {"Normal Distance":58.99}
Publish OK
Sending payload: {"Normal Distance":58.99}
Publish OK
29°C
Haze
ENG
IN
12:50
08-11-2022


2yey9a.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
513419106021@smartinternz.com
ID: 2yey9a
Browse
Action
Device Types
Interfaces
Add Device
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 {"Normal Distance":58.99} json a few seconds ago
Data {"Normal Distance":58.99} json a few seconds ago
Data {"Normal Distance":58.99} json a few seconds ago
Data {"Normal Distance":58.99} json a few seconds ago
Data {"Normal Distance":58.97} json a few seconds ago
Items per page 50
1–1 of 1 item
1 of 1 page
1
1 Simulation running
29°C
Haze
ENG
IN
12:50
08-11-2022