

## ASSIGNMENT 4

Date	2 Nov 22
Name	Balasankari R
Team ID	PNT2022TMID38273
Project Name	IOT Based Smart Crop Protection System for Agriculture

### QUESTION :

Write code and connection in wovki for ultrasonic sensor. Whenever distance is less than 100 cms send "alert" to IBM cloud and display in device recent events

### CODE :

```
#include <WiFi.h> //library for wifi
#include <PubSubClient.h> //library for MQTT
WiFiClient wifiClient;
String data3;
#define ORG "i7dctw"
#define DEVICE_TYPE "Balasankari"
#define DEVICE_ID "assignment_4"
#define TOKEN "1G2IpvYjCkYC6zZjC3"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/ Balasankari /fmt/json";
char topic[] = "iot-2/cmd/event_1/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);
```

```
const int trigpin=5;
const int echopin=18;
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");
    }
}

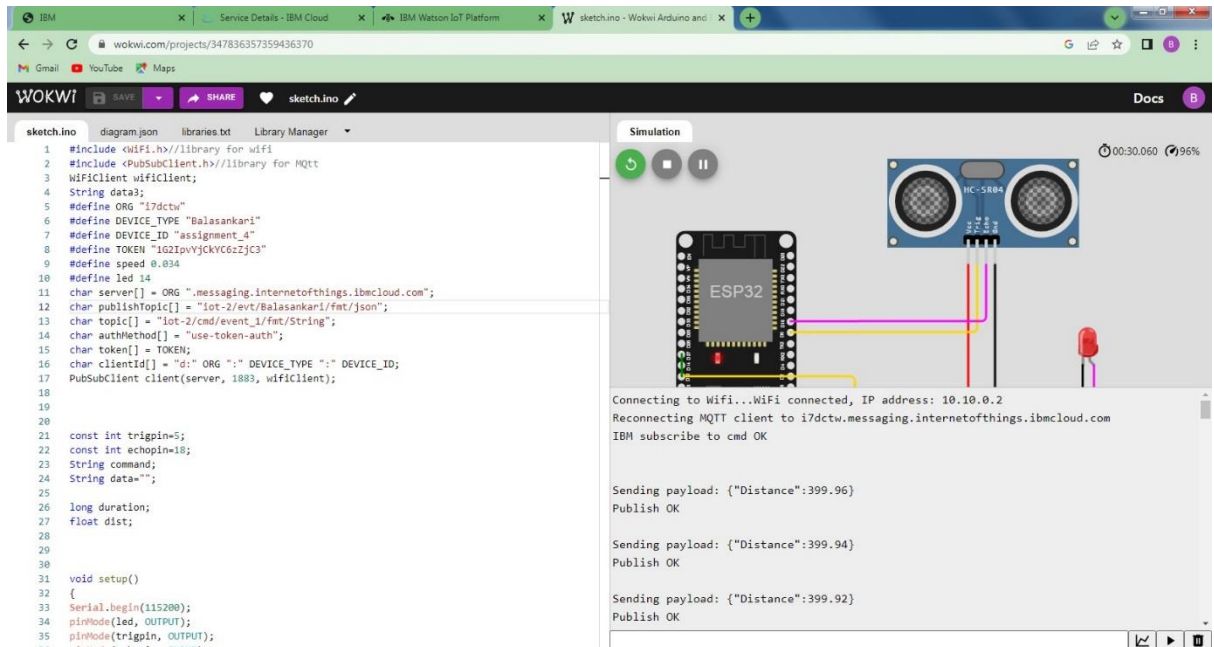
void publishData(){
    digitalWrite(trigpin,LOW);
    digitalWrite(trigpin,HIGH);
    delayMicroseconds(10);
    digitalWrite(trigpin,LOW);
    duration=pulseIn(echopin,HIGH);
    dist=duration*speed/2;
    if(dist<100){
        String payload = "{\"Alert Distance\":";
        payload += dist;
        payload += "}";
        Serial.print("\n");
        Serial.print("Sending payload: ");
        Serial.println(payload);
        if (client.publish(publishTopic, (char*) payload.c_str())){
            Serial.println("Publish OK");
        }
    }
    if(dist>100){
        String payload = "{\"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");
    }
}
}

```

## OUTPUT :

### 1) When Distance greater than 100 cm



The screenshot displays the Wokwi IDE interface. On the left, the sketch code is visible, which includes the necessary libraries, device definitions, and the main logic for connecting to the IBM Watson IoT Platform and publishing distance data. The right side shows a simulation of the hardware (ESP32, HC-SR04, and LED) and a console window with the following output:

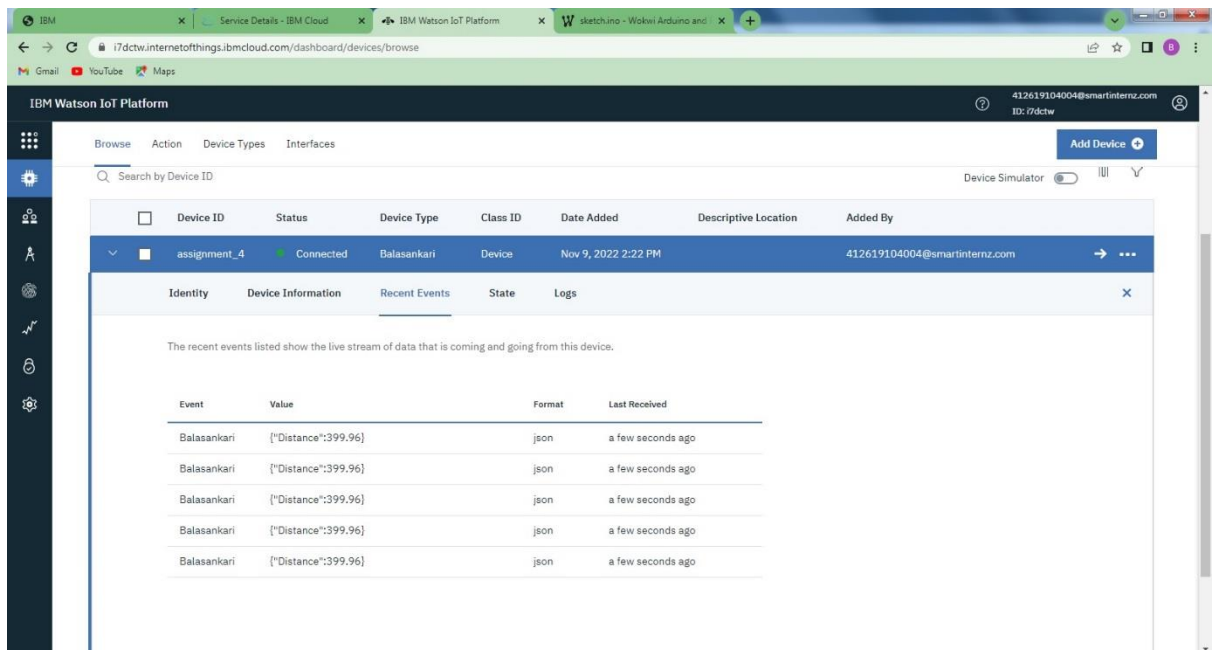
```
Connecting to Wifi...WiFi connected, IP address: 10.10.0.2
Reconnecting MQTT client to i7dctw.messaging.internetofthings.ibmcloud.com
IBM subscribe to cmd OK

Sending payload: {"Distance":399.96}
Publish OK

Sending payload: {"Distance":399.94}
Publish OK

Sending payload: {"Distance":399.92}
Publish OK
```

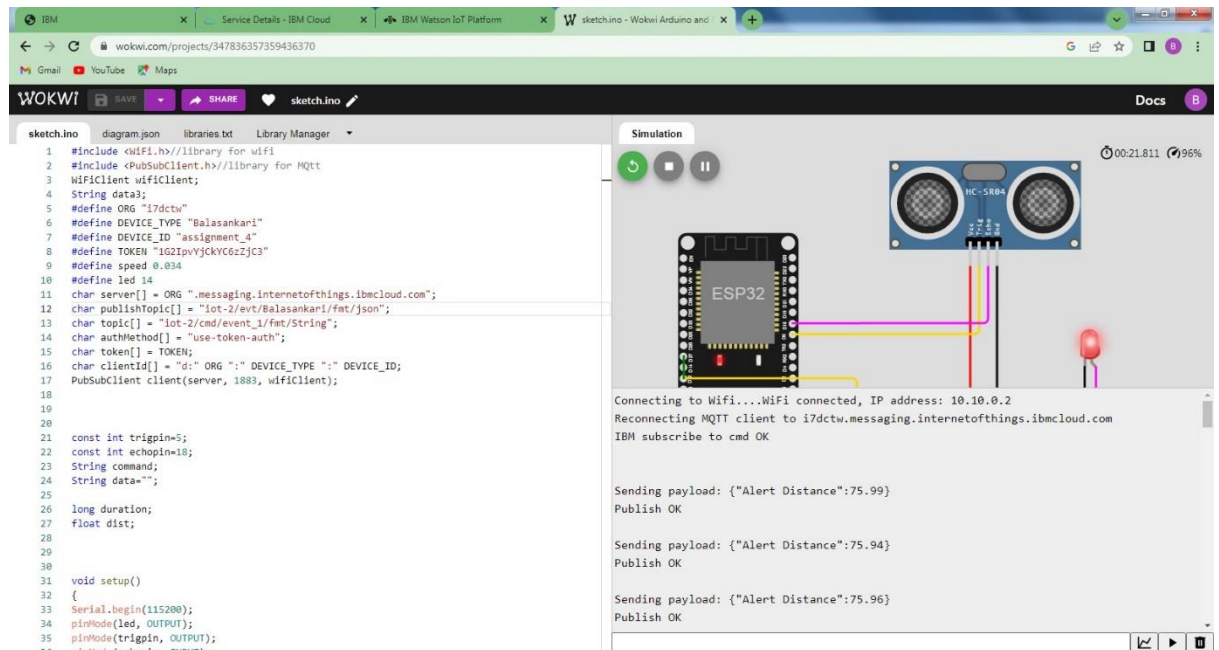
## IBM RECENT EVENTS



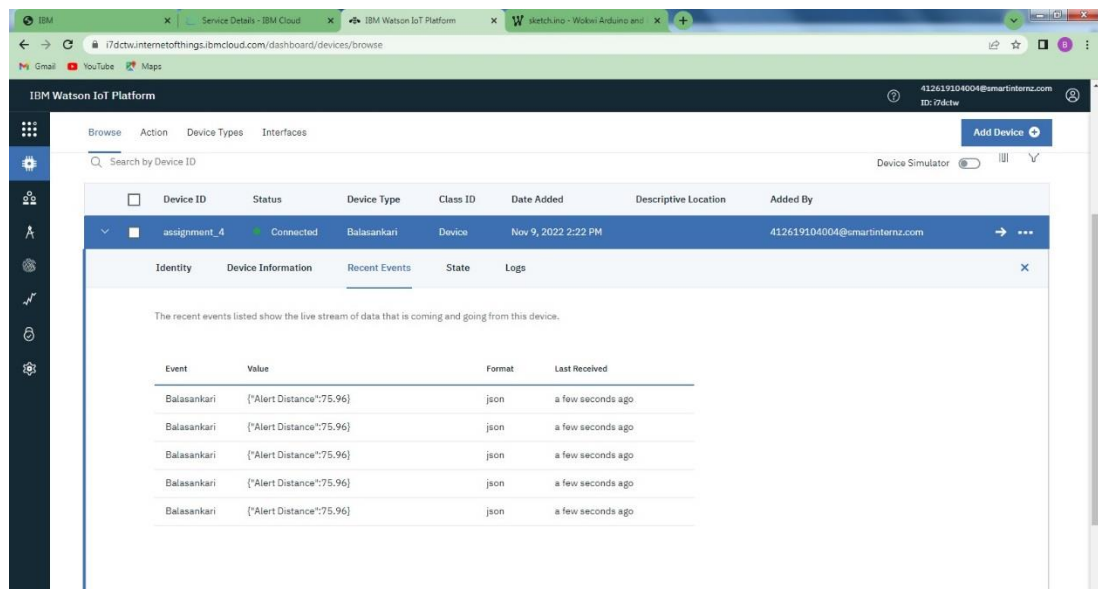
The screenshot shows the IBM Watson IoT Platform dashboard. The 'Recent Events' tab is selected, displaying a table of events for the device 'assignment\_4'. The table columns are Event, Value, Format, and Last Received.

Event	Value	Format	Last Received
Balasankari	{"Distance":399.96}	json	a few seconds ago
Balasankari	{"Distance":399.96}	json	a few seconds ago
Balasankari	{"Distance":399.96}	json	a few seconds ago
Balasankari	{"Distance":399.96}	json	a few seconds ago
Balasankari	{"Distance":399.96}	json	a few seconds ago

## 2) When distance less than 100



## IBM RECENT EVENTS



## WOVKI LINK-

<https://wokwi.com/projects/347836357359436370>