

## ASSIGNMENT 4

Date	2 Nov 22
Name	Sakthikumar R
Team ID	PNT2022TMID38287
Project Name	IOT Based Smart Solution for railways

### 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 "m72m9a"
#define DEVICE_TYPE "sakthi"
#define DEVICE_ID "assignment_4"
#define TOKEN "98765432"
#define speed 0.034
#define led 14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/sakthi/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");
    }
}

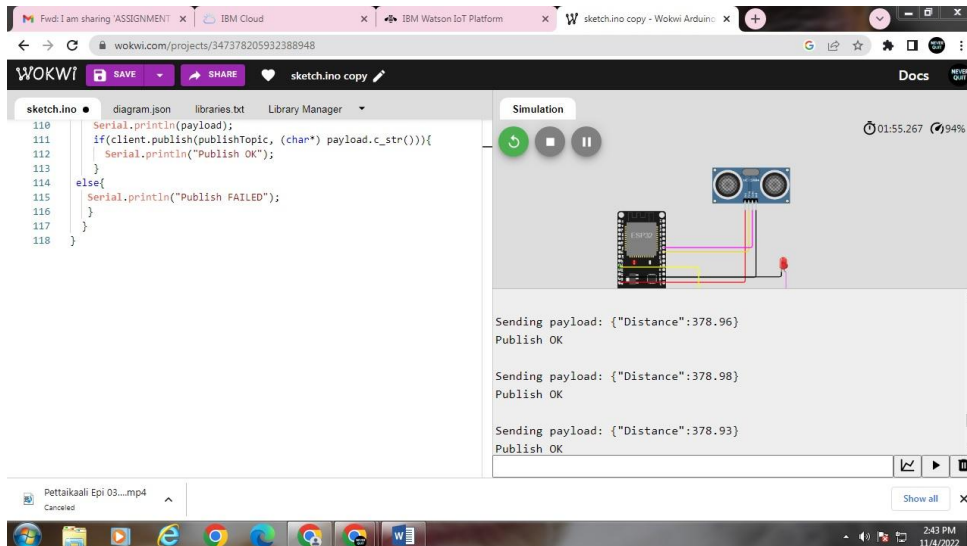
void publishData(){
    digitalWrite(trigpin,LOW);
    digitalWrite(trigpin,HIGH);
    delayMicroseconds(10);
    digitalWrite(trigpin,LOW);
    duration=pulseIn(echopin,HIGH);
    dist=duration*speed/2;
    if(dist<100){
        String payload = "{\"Alert Distance\":";
        payload += dist;
        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");
    }
}
}

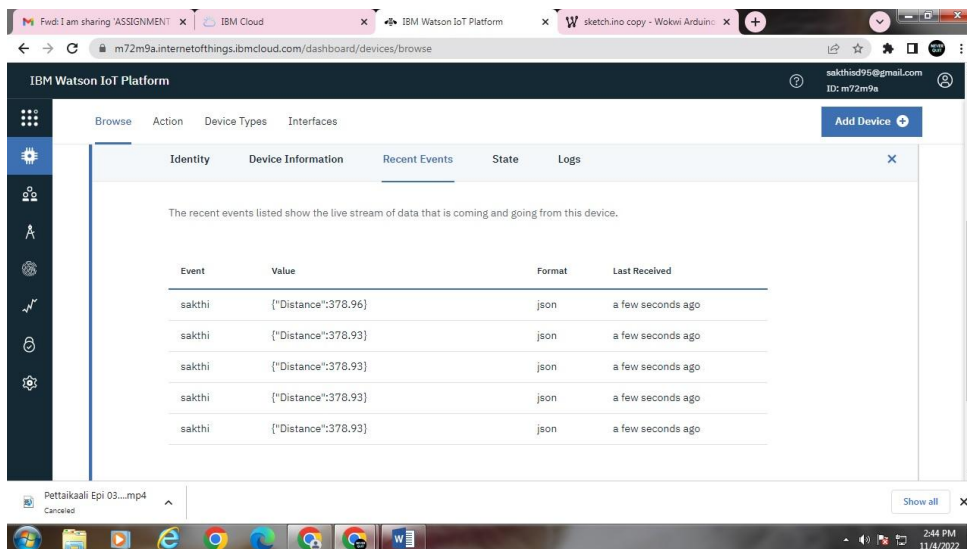
```

## OUTPUT :

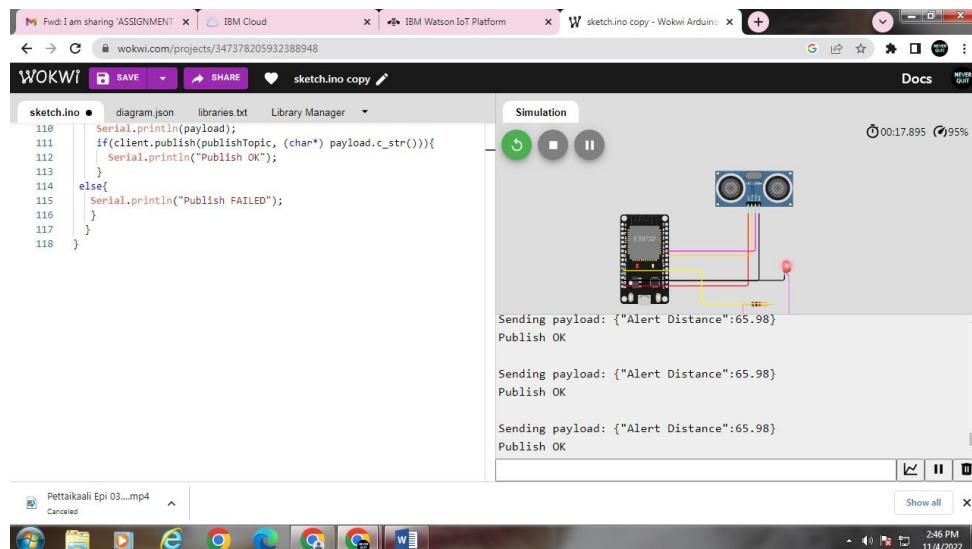
### 1) When Distance greater than 100 cm



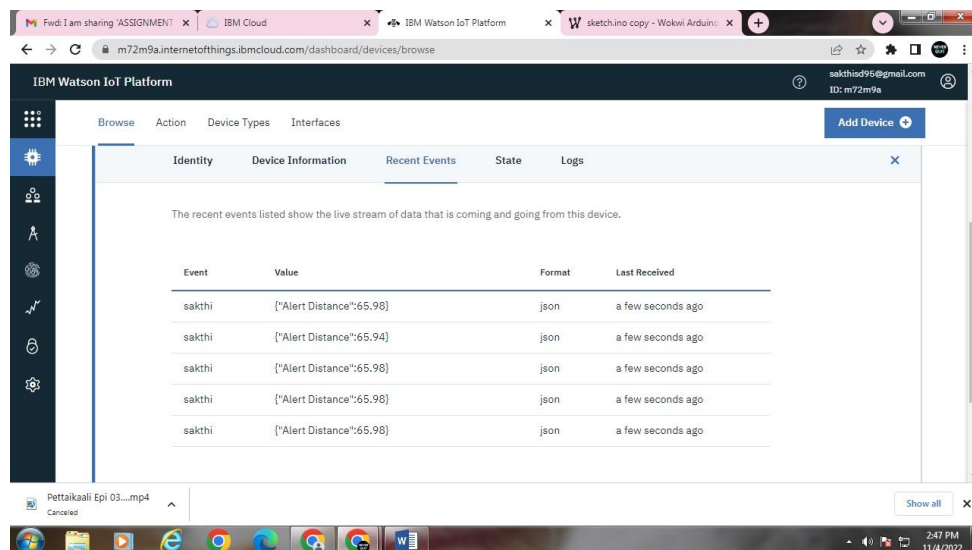
## IBM RECENT EVENTS



2) When distance less than 100



## IBM RECENT EVENTS



## WOVKI LINK-

<https://wokwi.com/projects/347378205932388948>