

ASSIGNMENT 4

Date	4 Nov 22
Name	Nathiya A
Team ID	PNT2022TMID38288
Project Name	GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES

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 "n9547y"

#define DEVICE_TYPE "NodeMCU"
#define DEVICE_ID "12345"
#define TOKEN "12345678"

#define speed 0.034 #define led
14
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/NodeMCU/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");

```

```

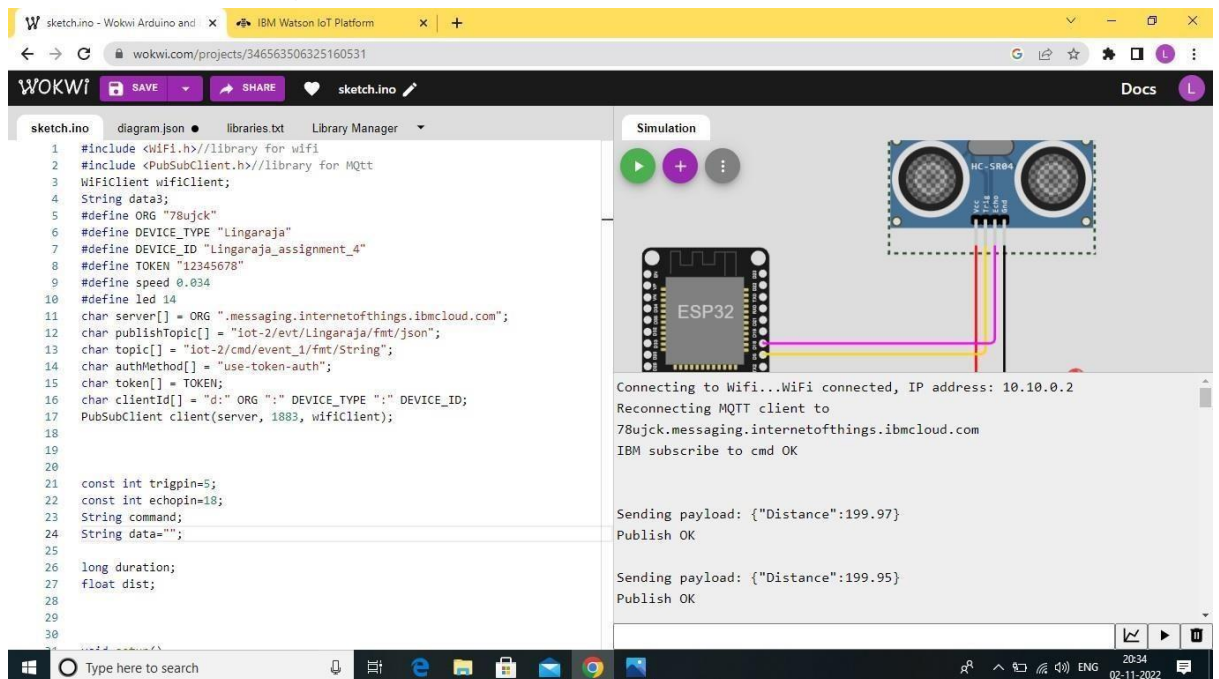
    Serial.print("Sending payload: ");
    Serial.println(payload);    if (client.publish(publishTopic,
(char*) payload.c_str())){
        Serial.println("Publish OK");
    } } if(dist>100){
    String payload = "{\"Distance\":\"";    payload
+= dist;    payload += "\"}";

    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");
    }
}
}
}

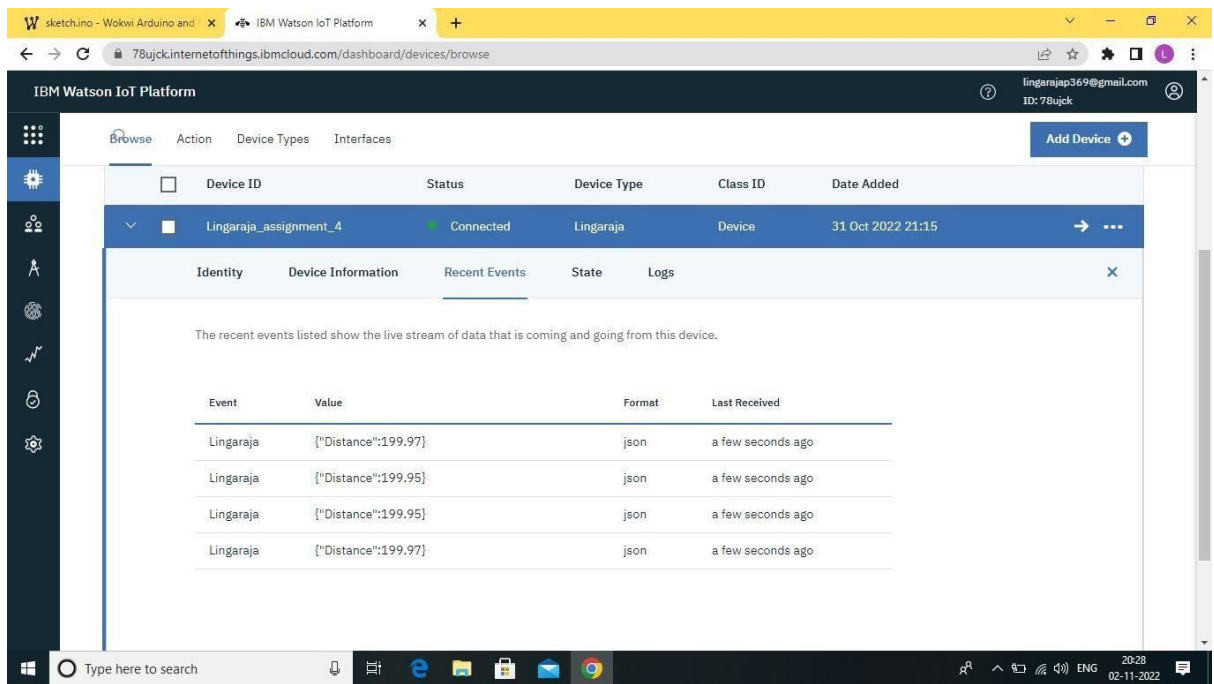
```

OUTPUT :

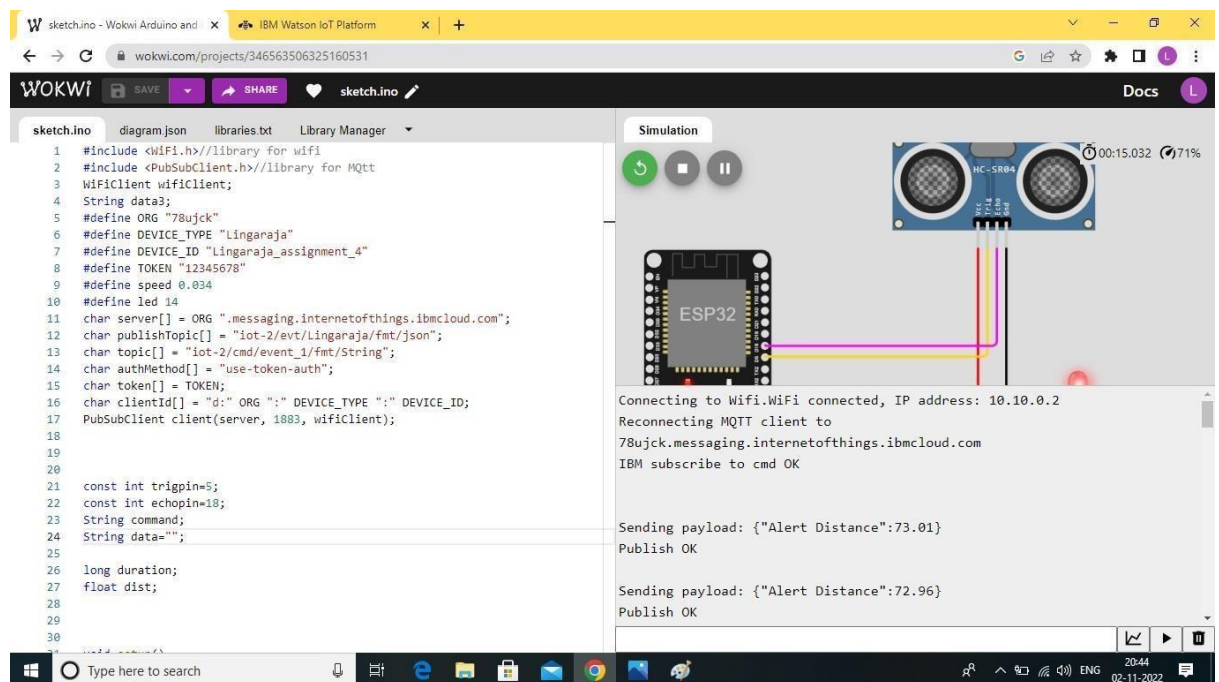
1) When Distance greater than 100 cm



IBM RECENT EVENTS



2) When distance less than 100



IBM RECENT EVENTS

The screenshot shows the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains icons for various functions. The main content area displays a device named 'Lingaraja_assignment_4' with a status of 'Connected'. Below this, the 'Recent Events' tab is active, showing a table of events. The table has columns for 'Event', 'Value', 'Format', and 'Last Received'. The events listed are all from the device 'Lingaraja' and contain the JSON payload '{"Alert Distance":72.96}' in 'json' format, received 'a few seconds ago'.

Event	Value	Format	Last Received
Lingaraja	{"Alert Distance":72.96}	json	a few seconds ago
Lingaraja	{"Alert Distance":72.96}	json	a few seconds ago
Lingaraja	{"Alert Distance":72.96}	json	a few seconds ago
Lingaraja	{"Alert Distance":72.96}	json	a few seconds ago

WOKWI

LINK <https://wokwi.com/projects/346563506325160531>