ASSIGNMENT 4

Name	**L. Archana**
Topic	**Smart Solutions for Railways**
Team id	**PNT2022TMID38248**

Write code and connections in wokwi for ultrasonic sensor

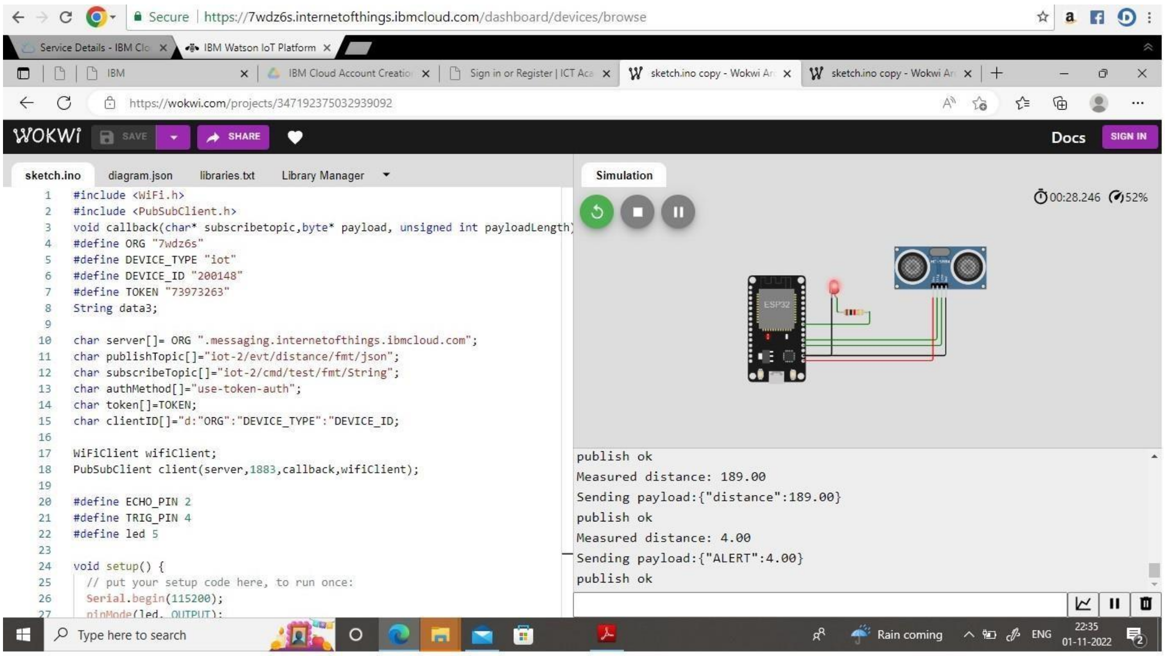
Whenever distance is less than 100cms send "alert" to ibm cloud and display in device recent events upload wokwi share images

```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic,byte* payload, unsigned int payloadLength);
#define ORG "7wdz6s"
#define DEVICE_TYPE "iot"
#define DEVICE_ID "200148"
#define TOKEN "73973263"
String data3;

char server[]= ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[]="iot-2/evt/distance/fmt/json";
char subscribeTopic[]="iot-2/cmd/test/fmt/String";
char authMethod[]="use-token-auth";
char token[]=TOKEN;
char clientID[]="d:"ORG":"DEVICE_TYPE":"DEVICE_ID;

WiFiClient wifiClient;
PubSubClient client(server,1883,callback,wifiClient);

#define ECHO_PIN 2
#define TRIG_PIN 4
#define led 5

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(led, OUTPUT);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
  wificonnect();
  mqttconnect();
}
float readDistanceCM() {
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  int duration=random(1,200);
  //Serial.println(duration);
  //duration = pulseIn(ECHO_PIN, HIGH);
  return duration ;
  //Serial.println(duration);

}

void loop() {
  float distance = readDistanceCM();
  //Serial.println(distance);

  bool isNearby = distance < 100;
  digitalWrite(led, isNearby);

  Serial.print("Measured distance: ");
  Serial.println(distance);
  if(distance<100){
    PublishData2(distance);

  }else{
    PublishData1(distance);

  }
  //PublishData(distance);
  delay(1000);
  if(!client.loop()){
    mqttconnect();
  }

  //delay(2000);
}
void PublishData1(float dist){
  mqttconnect();
  String payload= "{\"distance\":";
  payload += dist;
  payload+="}";
  Serial.print("Sending payload:");
  Serial.println(payload);

  if(client.publish(publishTopic,(char*)payload.c_str())){
    Serial.println("publish ok");
  } else{
    Serial.println("publish failed");
  }
}
void PublishData2(float dist){
  mqttconnect();
  String payload= "{\"ALERT\":";
  payload += dist;
  payload+="}";

  Serial.print("Sending payload:");
  Serial.println(payload);

  if(client.publish(publishTopic,(char*)payload.c_str())){
    Serial.println("publish ok");
  } else{
    Serial.println("publish failed");
  }
}
void mqttconnect(){
  if(!client.connected()){
    Serial.print("Reconnecting to ");
    Serial.println(server);
    while(!!!client.connect(clientID, authMethod, token)){
      Serial.print(".");
      delay(500);
    }
    initManagedDevice();
    Serial.println();
  }
}

void wificonnect(){
  Serial.println();
  Serial.print("Connecting to");

  WiFi.begin("Wokwi-GUEST","",6);
  while(WiFi.status()!=WL_CONNECTED){
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WIFI CONNECTED");
  Serial.println("IP address:");
  Serial.println(WiFi.localIP());
}

void initManagedDevice(){
  if(client.subscribe(subscribeTopic)){
    Serial.println((subscribeTopic));
    Serial.println("subscribe to cmd ok");
  }else{
    Serial.println("subscribe to cmd 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++){
    data3 += (char)payload[i];
  }
  Serial.println("data:"+ data3);
  if(data3=="lighton"){
    Serial.println(data3);
    digitalWrite(led,HIGH);
  }else{
    Serial.println(data3);
    digitalWrite(led,LOW);
  }
  data3="";
}
```

Link:

https://wokwi.com/projects/347292100667114066

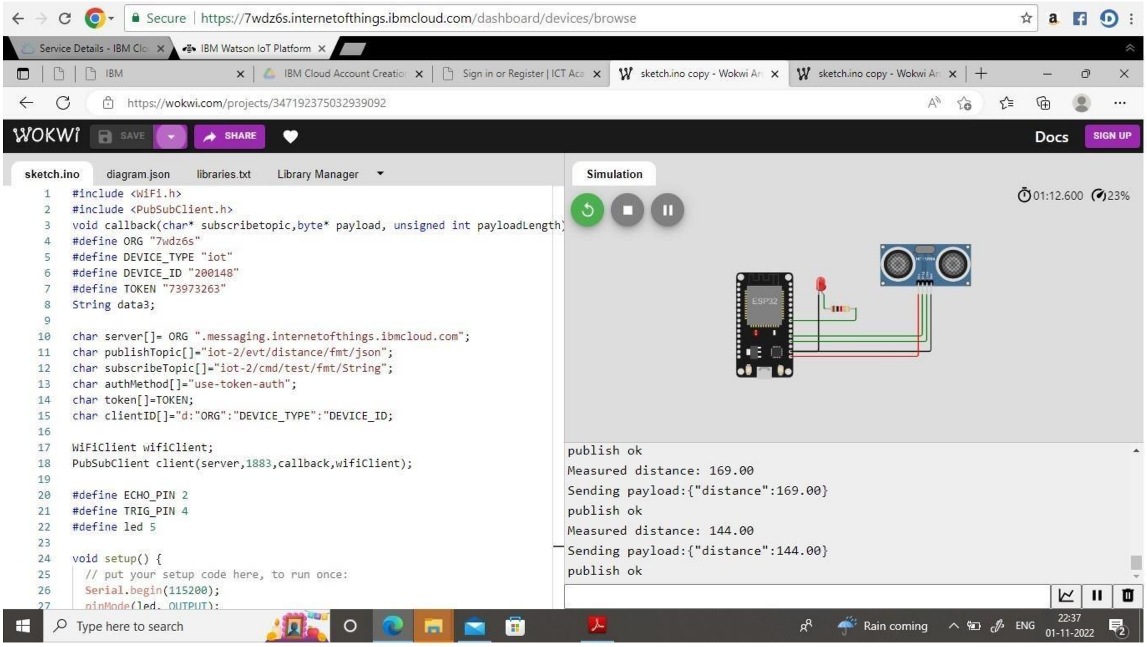
NORMAL CASE:


Secure
https://7wdz6s.internetofthings.ibmcloud.com/dashboard/devices/browse
Service Details - IBM Clo
IBM Watson IoT Platform
IBM
IBM Cloud Account Creatio
Sign in or Register | ICT Aca
sketch.ino copy - Wokwi Ar
sketch.ino copy - Wokwi Ar
https://wokwi.com/projects/347192375032939092
WOKWi
SAVE
SHARE
Docs
SIGN IN
sketch.ino
diagram.json
libraries.txt
Library Manager
Simulation
00:28.246
52%
ESP32
Type here to search
Rain coming
ENG
22:35
01-11-2022

```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic,byte* payload, unsigned int payloadLength)
#define ORG "7wdz6s"
#define DEVICE_TYPE "iot"
#define DEVICE_ID "200148"
#define TOKEN "73973263"
String data3;

char server[]= ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[]="iot-2/evt/distance/fmt/json";
char subscribeTopic[]="iot-2/cmd/test/fmt/String";
char authMethod[]="use-token-auth";
char token[]=TOKEN;
char clientID[]="d:"ORG":"DEVICE_TYPE":"DEVICE_ID;

WiFiClient wifiClient;
PubSubClient client(server,1883,callback,wifiClient);

#define ECHO_PIN 2
#define TRIG_PIN 4
#define led 5

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(led, OUTPUT);
```

```
publish ok
Measured distance: 189.00
Sending payload:{"distance":189.00}
publish ok
Measured distance: 4.00
Sending payload:{"ALERT":4.00}
publish ok
```

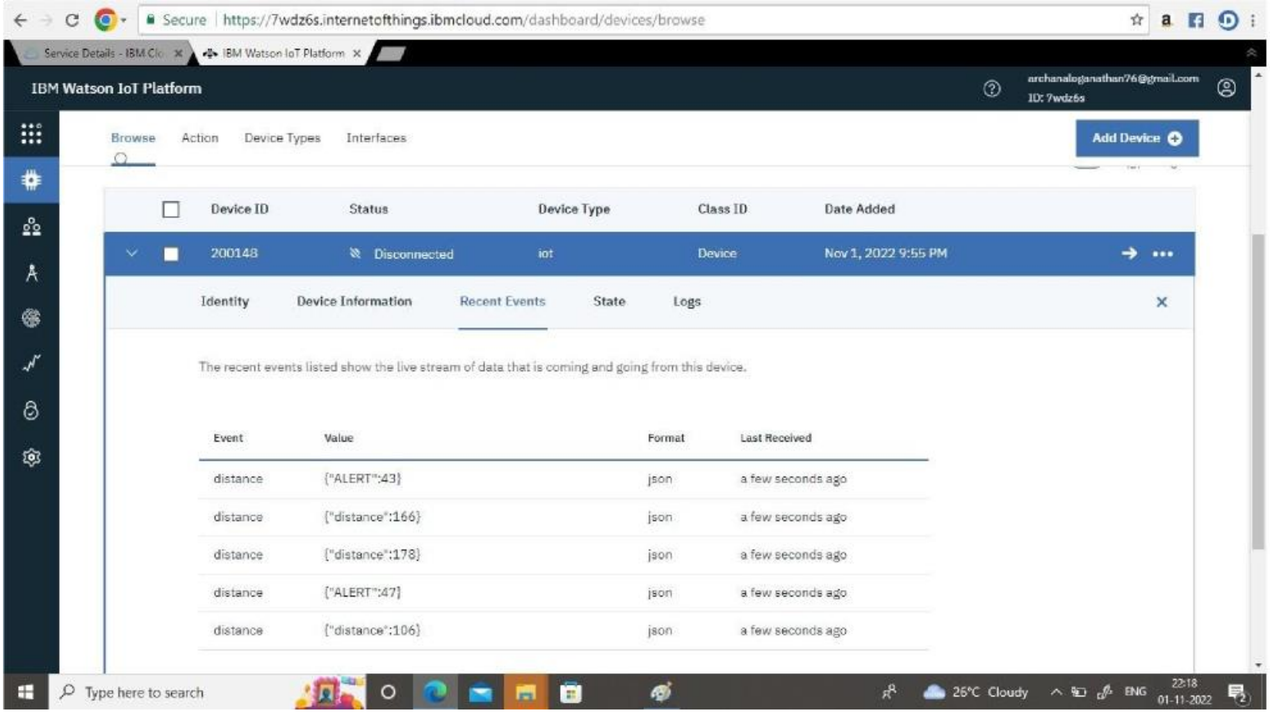
ALTER CASE:


Secure
https://7wdz6s.internetofthings.ibmcloud.com/dashboard/devices/browse
Service Details - IBM Clo
IBM Watson IoT Platform
IBM
IBM Cloud Account Creatio
Sign in or Register | ICT Aca
sketch.ino copy - Wokwi Ar
sketch.ino copy - Wokwi Ar
https://wokwi.com/projects/347192375032939092
WOKWi
SAVE
SHARE
Docs
SIGN UP
sketch.ino
diagram.json
libraries.txt
Library Manager
Simulation
01:12.600
23%
ESP32
Type here to search
Rain coming
ENG
22:37
01-11-2022

```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic,byte* payload, unsigned int payloadLength)
#define ORG "7wdz6s"
#define DEVICE_TYPE "iot"
#define DEVICE_ID "200148"
#define TOKEN "73973263"
String data3;

char server[]= ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[]="iot-2/evt/distance/fmt/json";
char subscribeTopic[]="iot-2/cmd/test/fmt/String";
char authMethod[]="use-token-auth";
char token[]=TOKEN;
char clientID[]="d:"ORG":"DEVICE_TYPE":"DEVICE_ID;

WiFiClient wifiClient;
PubSubClient client(server,1883,callback,wifiClient);

#define ECHO_PIN 2
#define TRIG_PIN 4
#define led 5

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(led, OUTPUT);
```

```
publish ok
Measured distance: 169.00
Sending payload:{"distance":169.00}
publish ok
Measured distance: 144.00
Sending payload:{"distance":144.00}
publish ok
```

IBM CLOUD



Device ID	Status	Device Type	Class ID	Date Added
200148	Disconnected	iot	Device	Nov 1, 2022 9:55 PM

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

Event	Value	Format	Last Received
distance	{"ALERT":43}	json	a few seconds ago
distance	{"distance":166}	json	a few seconds ago
distance	{"distance":178}	json	a few seconds ago
distance	{"ALERT":47}	json	a few seconds ago
distance	{"distance":106}	json	a few seconds ago