Assignment 4

Name	Durairasu S
Team ID	PNT2022TMID47401
Project Name	IoT Based Safety Gadget for Child Safety Monitoring & Notification

1.Write Code and connections in wokwi for ultrasonic sensor. whatever distance is less than 100 cms send "Alert" to ibm cloud aand display in device recent events.

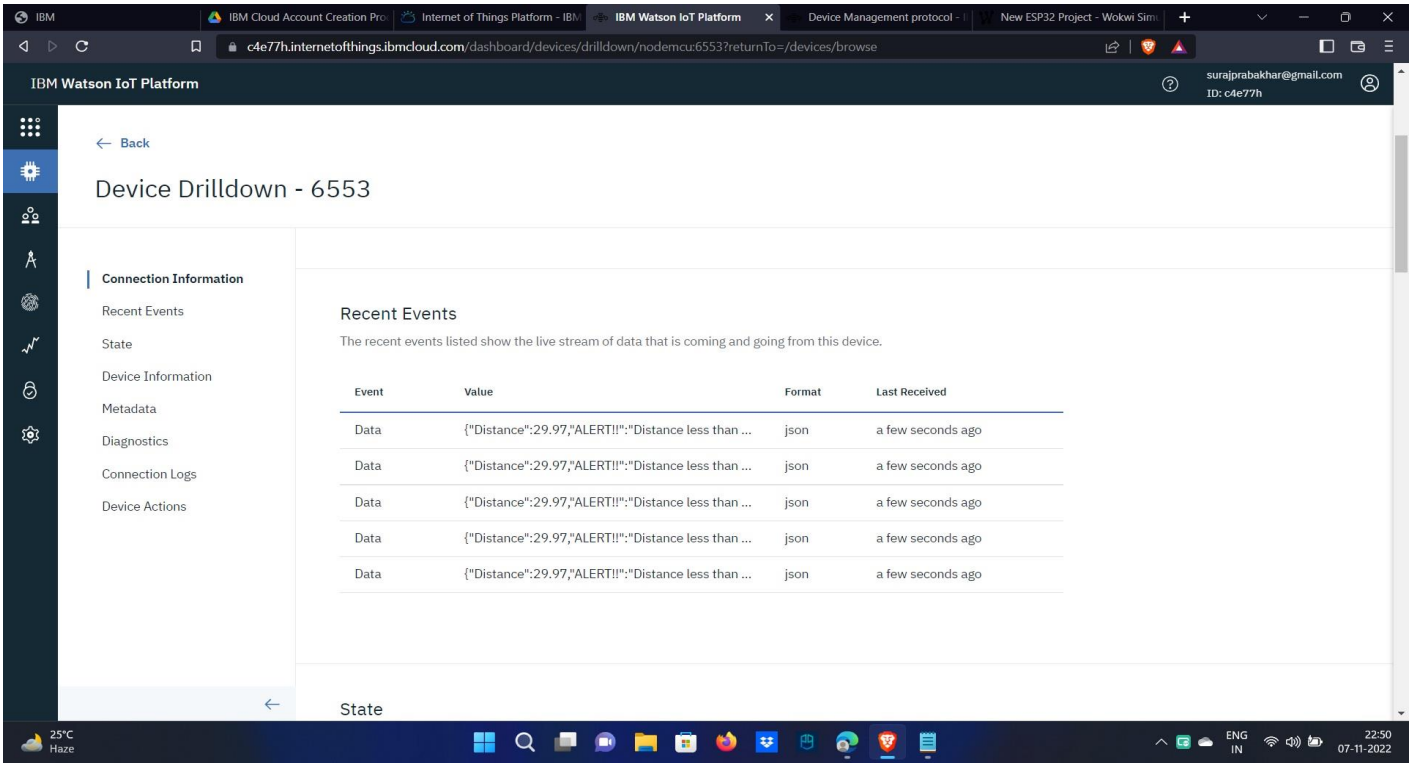
```
#include <WiFi.h>
#include <PubSubClient.h>


void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);


//-------credentials of IBM Accounts------


#define ORG "c4e77h"//IBM ORGANITION ID
#define DEVICE_TYPE "nodemcu"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "6553"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "S3FUWX1R&OpcJZ-ijf"     //Token
String data3;


char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/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);
const int trigPin = 5;
const int echoPin = 18;
#define SOUND_SPEED 0.034
long duration;
float distance;


void setup() {
  Serial.begin(115200);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
  wificonnect();
  mqttconnect();
}


void loop()
{
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance = duration * SOUND_SPEED/2;
  Serial.print("Distance (cm): ");
```

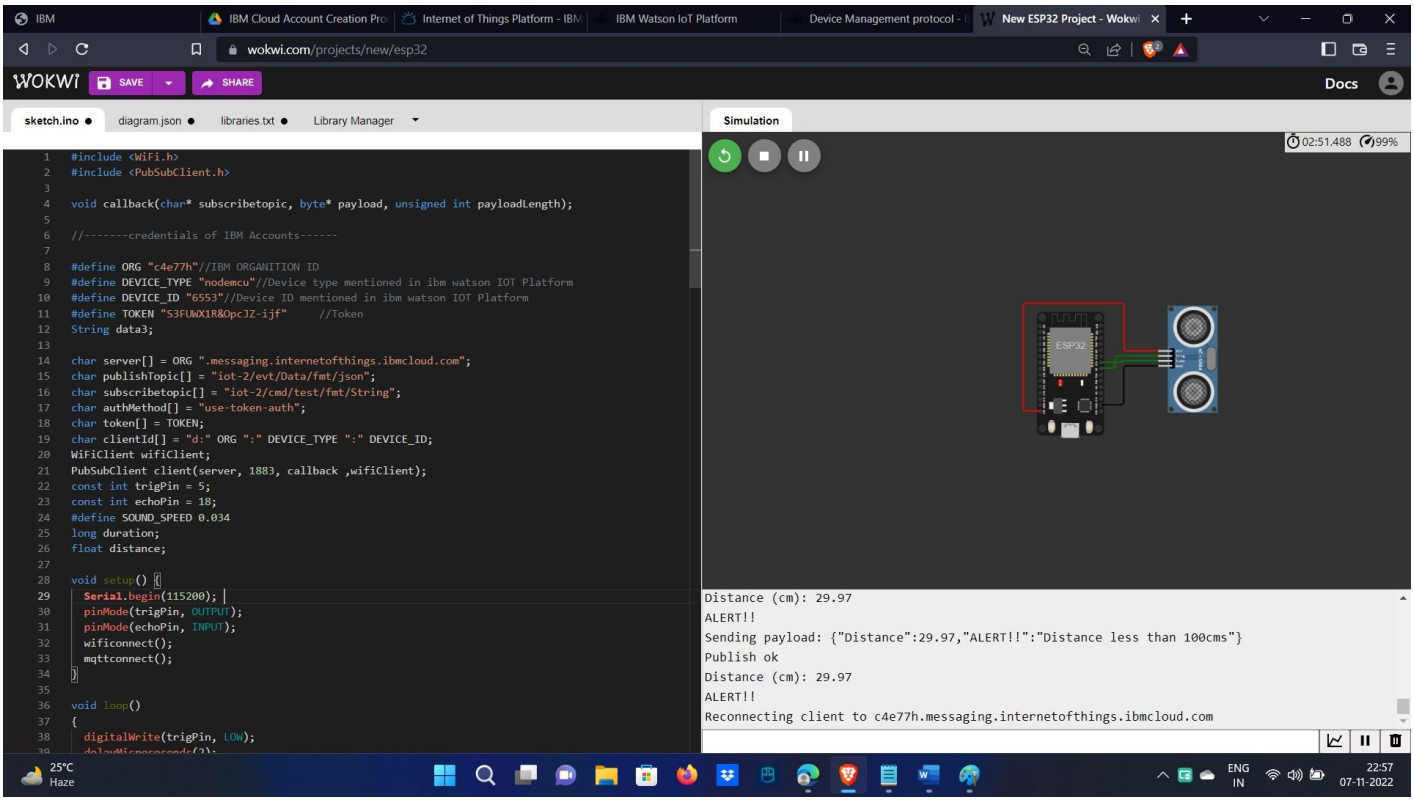
```
  Serial.println(distance);
  if(distance<100)
  {
    Serial.println("ALERT!!");
    delay(1000);

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

}


void PublishData(float dist) {
  mqttconnect();

  String payload = "{\"Distance\":";
  payload += dist;
  payload += ",\"ALERT!!\":""\"Distance less than 100cms\"";
  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 client 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++) {
    //Serial.print((char)payload[i]);
    data3 += (char)payload[i];
  }
Serial.println("data: "+ data3);
data3="";
}
```



Wokwi link: https://wokwi.com/projects/new/esp32


IBM
IBM Cloud Account Creation Pro
Internet of Things Platform - IBM
IBM Watson IoT Platform
Device Management protocol -
New ESP32 Project - Wokwi
wokwi.com/projects/new/esp32
WOKWI
SAVE
SHARE
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);
//-------credentials of IBM Accounts------
#define ORG "c4e77h"//IBM ORGANITION ID
#define DEVICE_TYPE "nodemcu"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "6553"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "S3FUWX1R&OpcJZ-ijf" //Token
String data3;
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/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);
const int trigPin = 5;
const int echoPin = 18;
#define SOUND_SPEED 0.034
long duration;
float distance;
void setup() {
Serial.begin(115200);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
wificonnect();
mqttconnect();
}
void loop()
{
digitalWrite(trigPin, LOW);
Simulation
02:51.488
99%
ESP32
Distance (cm): 29.97
ALERT!!
Sending payload: {"Distance":29.97,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 29.97
ALERT!!
Reconnecting client to c4e77h.messaging.internetofthings.ibmcloud.com
25°C
Haze
ENG
IN
22:57
07-11-2022