Assignment 4

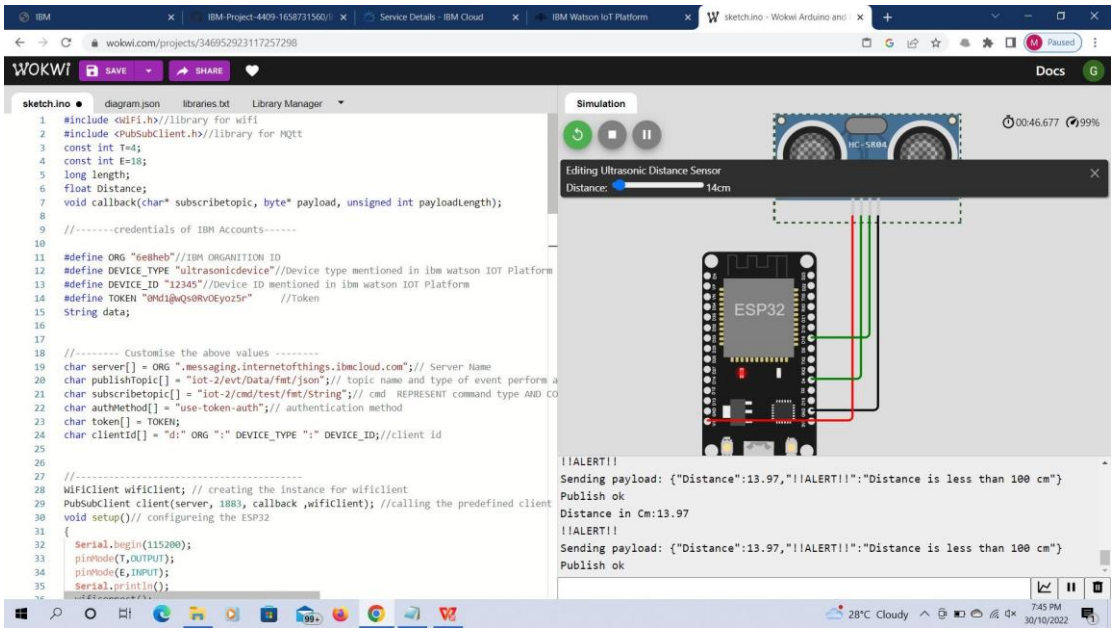
Write code and connections in wokwi for the ultrasonic sensor.
Whenever the distance is less than 100 cms send an "alert" to the IBM cloud and display in the device recent events.
Upload document with wokwi share link and images of IBM cloud

```
#include <WiFi.h>//library for wifi #include
<PubSubClient.h>//library for MQtt const int
T=4; const int E=18; long length; float
Distance;
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

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

#define ORG "6e8heb"//IBM ORGANITION ID
#define DEVICE_TYPE "ultrasonicdevice"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "12345"//Device ID mentioned in ibm watson IOT Platform
 #define TOKEN "0Md1@wQs0RvOEyoz5r"     //Token
String data;


//-------- Customise the above values --------
 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event perform and format in which data
to be send
char subscribetopic[] = "iot-2/cmd/test/fmt/String";// cmd REPRESENT command type AND COMMAND IS
TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


//-----------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passing parameter like
server id,portand wificredential void setup()// configureing the ESP32
{
  Serial.begin(115200);
  pinMode(T,OUTPUT);
  pinMode(E,INPUT);
  Serial.println();
  wificonnect();
  mqttconnect();
}

void loop()// Recursive Function
{ digitalWrite(T,LOW);
  delay(1000);
  digitalWrite(T,HIGH);
  delay(1000);
  digitalWrite(T,LOW);
  length=pulseIn(E,HIG
  H);
  Distance=length*(0.03
  4/2);
```

```
  Serial.print("Distance in Cm:");
  Serial.println(Distance);
  if(Distance<100)
  {
    Serial.println("!!ALERT!!");
  delay(1000);
  PublishData(Distance);
  delay(1000); if (!client.loop())
  {
    mqttconnect();
  } }
  delay(1000)
  ;
}

/*.....retrieving to Cloud.....*/

void PublishData(float dist) {
  mqttconnect();//function call for connecting to ibm
  /* creating the String in in form JSon to update the data to ibm
     cloud
  */
  String payload = "{\"Distance\":";
  payload += dist;
  payload += ",\"!!ALERT!!\":""\"Distance is less than 100 cm\""";
  payload += "}";
  Serial.print("Sending payload: ");
  Serial.println(payload);


  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud then it will print publish ok in Serial
monitor or else it will print publish failed
  } 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() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");

  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
  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]);
    data += (char)payload[i];
  }

  Serial.println("data: "+ data);
data=""; }
```

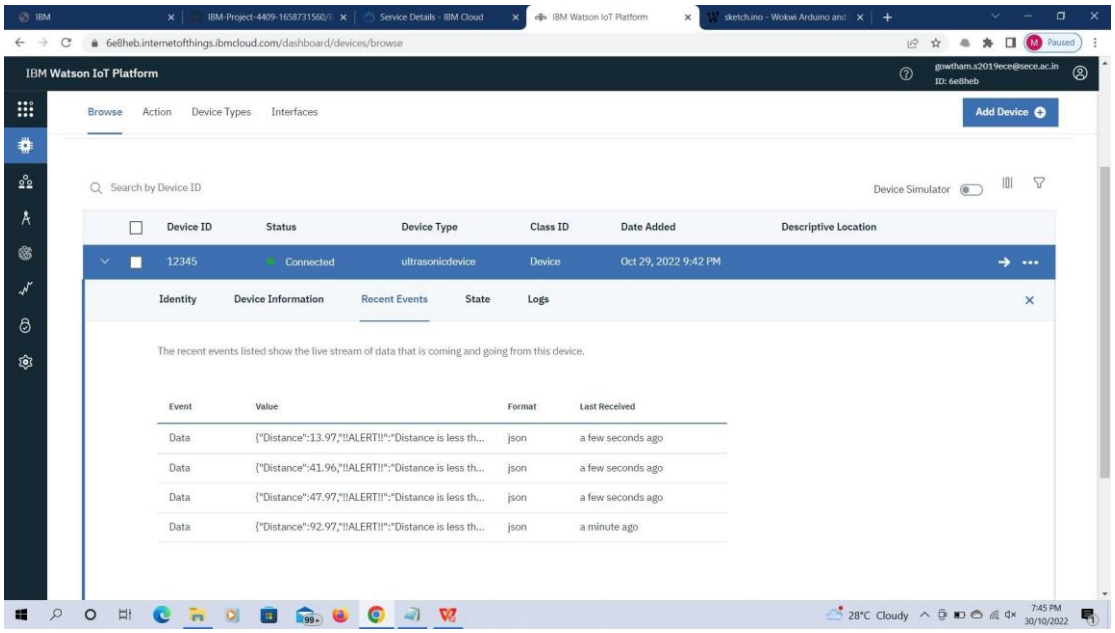
OUTPUT



Wokwi simulation link:

https://wokwi.com/projects/346952923117257298

Recent Events in IBM cloud:


IBM Watson IoT Platform
Browse
Action
Device Types
Interfaces
Add Device
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
12345
Connected
ultrasonicdevice
Device
Oct 29, 2022 9:42 PM
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
Data
{"Distance":13.97,"!!ALERT!!":"Distance is less th...
json
a few seconds ago
Data
{"Distance":41.96,"!!ALERT!!":"Distance is less th...
json
a few seconds ago
Data
{"Distance":47.97,"!!ALERT!!":"Distance is less th...
json
a few seconds ago
Data
{"Distance":92.97,"!!ALERT!!":"Distance is less th...
json
a minute ago
28°C Cloudy
7:45 PM
30/10/2022