Assignment -4

ESP32 Programming

Assignment Date	31 October 2022
Student Name	SHAIMA M
Student Roll Number	962819106039
Maximum Marks	2 Marks

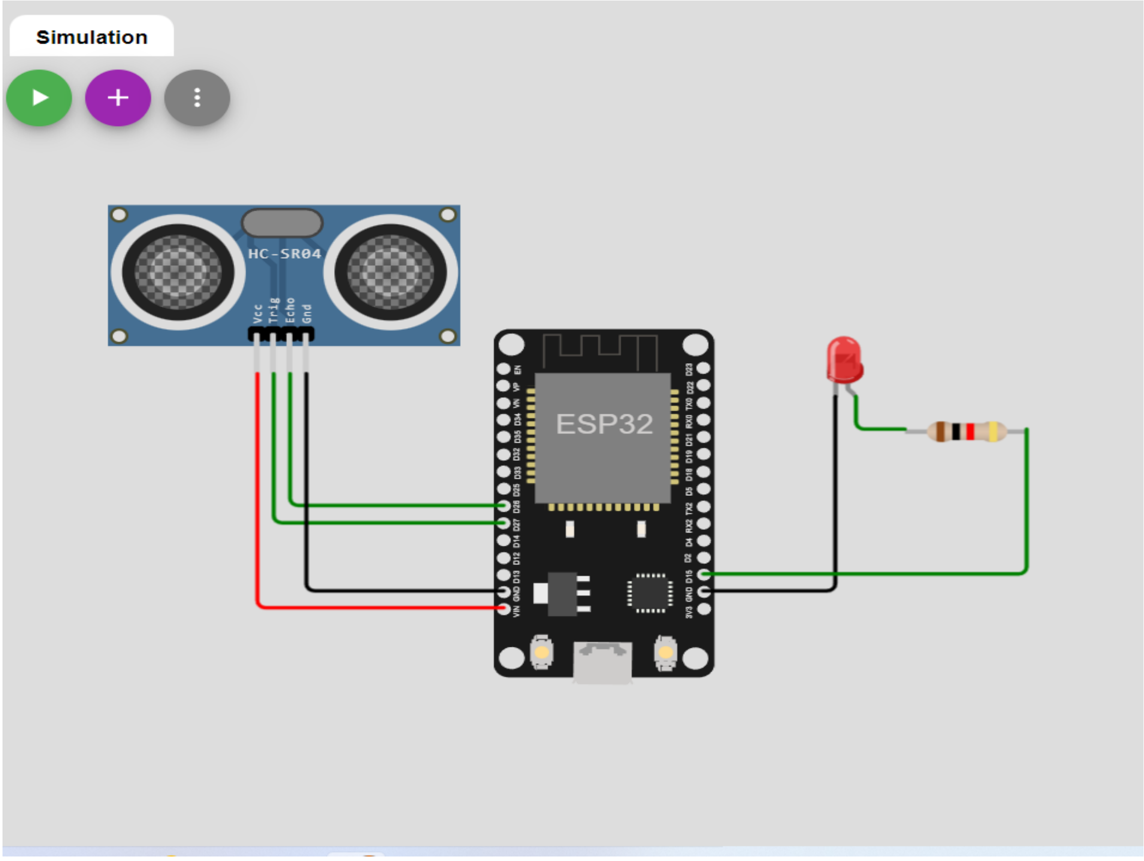
Question-1:

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 the document with share link and images of IBM cloud.

Wokwi share link :

https://wokwi.com/projects/347051468171248212

Circuit diagram :



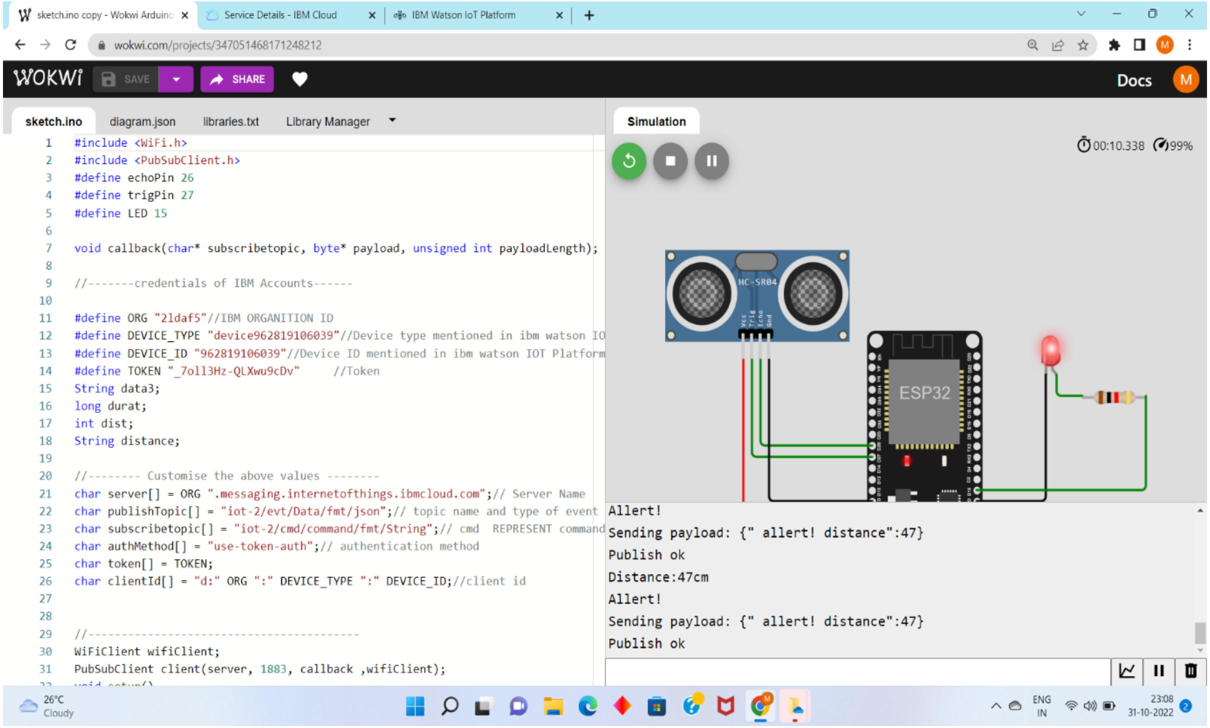
Code :

```
#include <WiFi.h>
#include <PubSubClient.h>
#define echoPin 26
#define trigPin 27
#define LED 15

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

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

#define ORG "2ldaf5"//IBM ORGANITION ID
#define DEVICE_TYPE "device962819106039"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "962819106039"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "_7oll3Hz-QLXwu9cDv"     //Token
String data3;
long durat;
int dist;
String distance;

//-------- 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/command/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;
PubSubClient client(server, 1883, callback ,wifiClient);
void setup()
{ pinMode(trigPin, OUTPUT);
  pinMode(echoPin,INPUT);
  Serial.begin(115200);
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
  mqttconnect();
}
```

```
void loop()
{

  digitalWrite(trigPin,LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin,LOW);
  durat=pulseIn(echoPin,HIGH);
  dist=durat*0.034/2;
  Serial.println("Distance:"+String(dist)+"cm");
  if(dist<100){
    digitalWrite(LED,HIGH);
    Serial.print("Allert! \n");

  }
  else{
    digitalWrite(LED,LOW);


  }
  if(dist<100)
  {

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



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

void PublishData(String distance) {
  mqttconnect();
  String payload = "{\" allert\! 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 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]);
    data3 += (char)payload[i];
  }

  Serial.println("data: "+ data3);



}
```

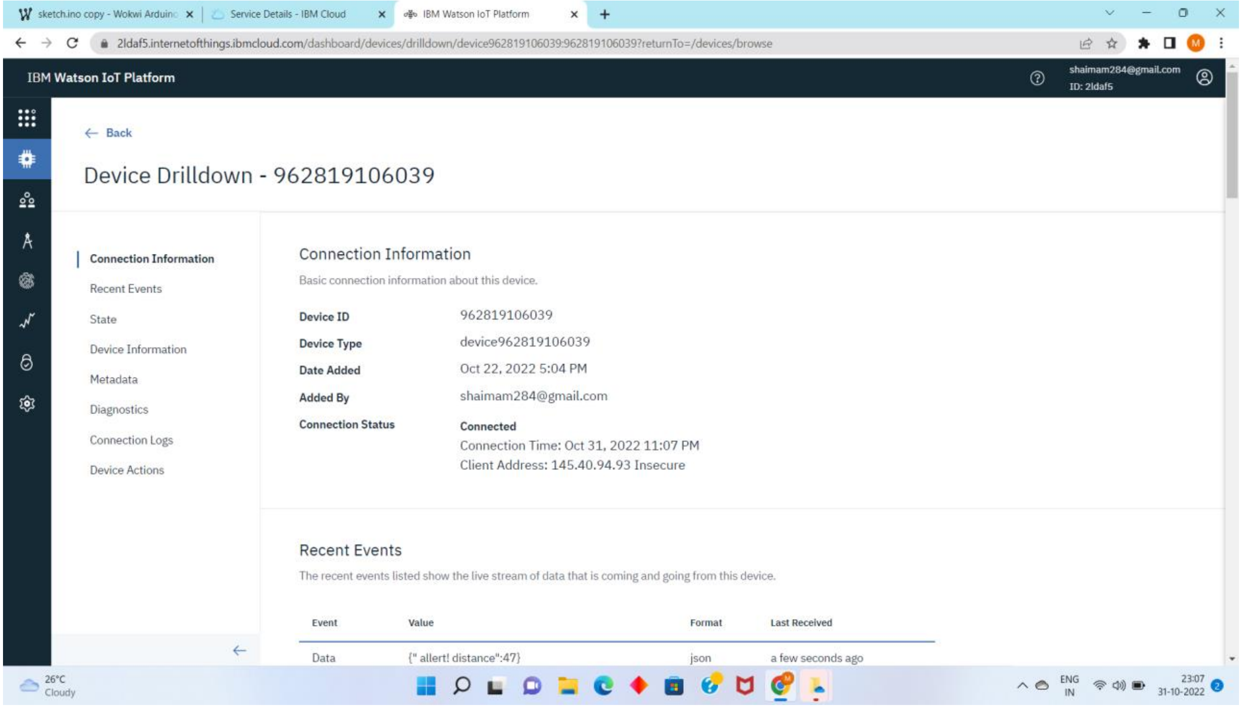
Output:



```
#include <WiFi.h>
#include <PubSubClient.h>
#define echoPin 26
#define trigPin 27
#define LED 15

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

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

#define ORG "2ldaf5"//IBM ORGANITION ID
#define DEVICE_TYPE "device962819106039"//Device type mentioned in ibm watson IO
#define DEVICE_ID "962819106039"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "_7oll3Hz-QLXwu9cDv"     //Token
String data3;
long durat;
int dist;
String distance;

//-------- 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
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


//---------------------------------------
WiFiClient wifiClient;
PubSubClient client(server, 1883, callback ,wifiClient);
```

```
Allert!
Sending payload: {" allert! distance":47}
Publish ok
Distance:47cm
Allert!
Sending payload: {" allert! distance":47}
Publish ok
```

Images of IBM Cloud :



Device Drilldown - 962819106039

Connection Information

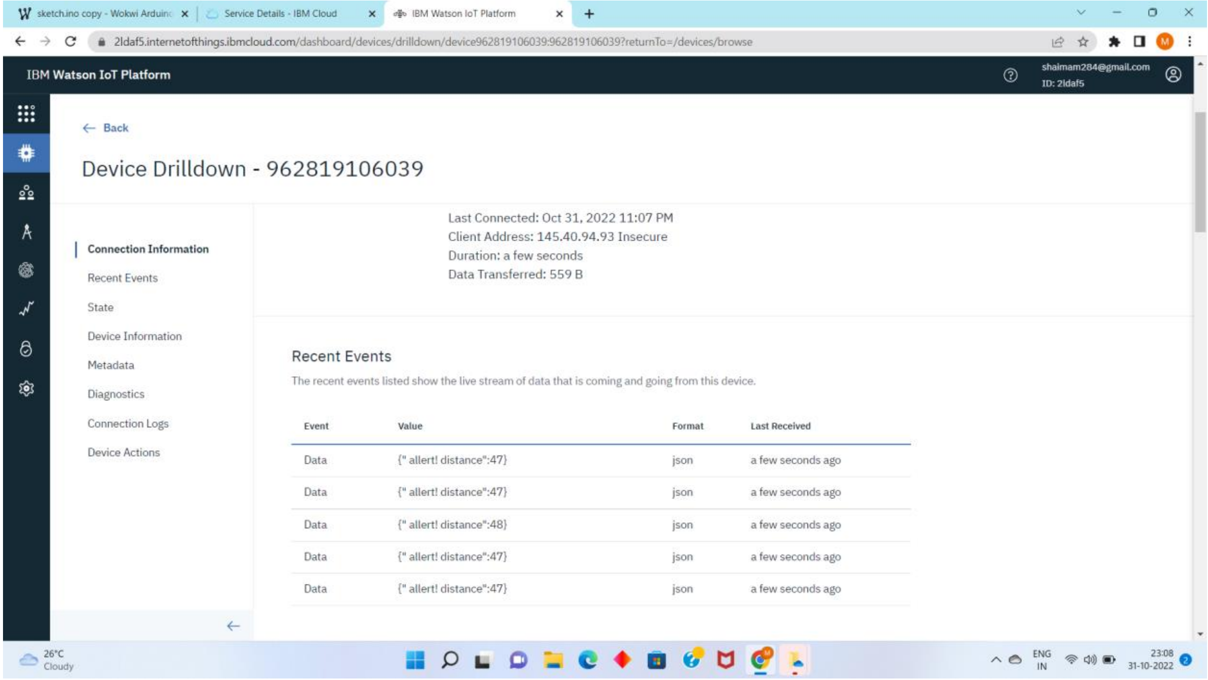
Basic connection information about this device.

Device ID	962819106039
Device Type	device962819106039
Date Added	Oct 22, 2022 5:04 PM
Added By	shaimam284@gmail.com
Connection Status	Connected Connection Time: Oct 31, 2022 11:07 PM Client Address: 145.40.94.93 Insecure

Recent Events

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

Event	Value	Format	Last Received
Data	{" allert! distance":47}	json	a few seconds ago



Device Drilldown - 962819106039

Last Connected: Oct 31, 2022 11:07 PM
Client Address: 145.40.94.93 Insecure
Duration: a few seconds
Data Transferred: 559 B

Recent Events

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

Event	Value	Format	Last Received
Data	{" allert! distance":47}	json	a few seconds ago
Data	{" allert! distance":47}	json	a few seconds ago
Data	{" allert! distance":48}	json	a few seconds ago
Data	{" allert! distance":47}	json	a few seconds ago
Data	{" allert! distance":47}	json	a few seconds ago