Assignment - 4

Assignment Date	17 November 2022
Student Name	Abbarna.N
Student Roll Number	732419104001
Maximum Marks	2 Marks
Team ID	PNT2022TMID44658

Question:

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.

Code:

```
#include <WiFi.h>
#include <WiFiClient.h>
#include <PubSubClient.h>
const int trigPin = 27;
const int echoPin = 26;
//define sound speed in cm/uS
#define SOUND_SPEED 0.034
#define CM_TO_INCH 0.393701
longduration;
float distanceCm;
float distanceInch;
void callback(char* subscribetopic, byte*payload,unsigned int payloadLength);
 //-------credentialsofIBMAccounts------


#define ORG "z22obn"//IBM ORGANITION ID
#defineDEVICE_TYPE"Assignment-ibm"//DevicetypementionedinibmwatsonIOT Platform
#defineDEVICE_ID"Sensor"//DeviceIDmentionedinibmwatsonIOTPlatform
#defineTOKEN"12345678"        //Token
String data3;
//--------Customisetheabovevalues--------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
```

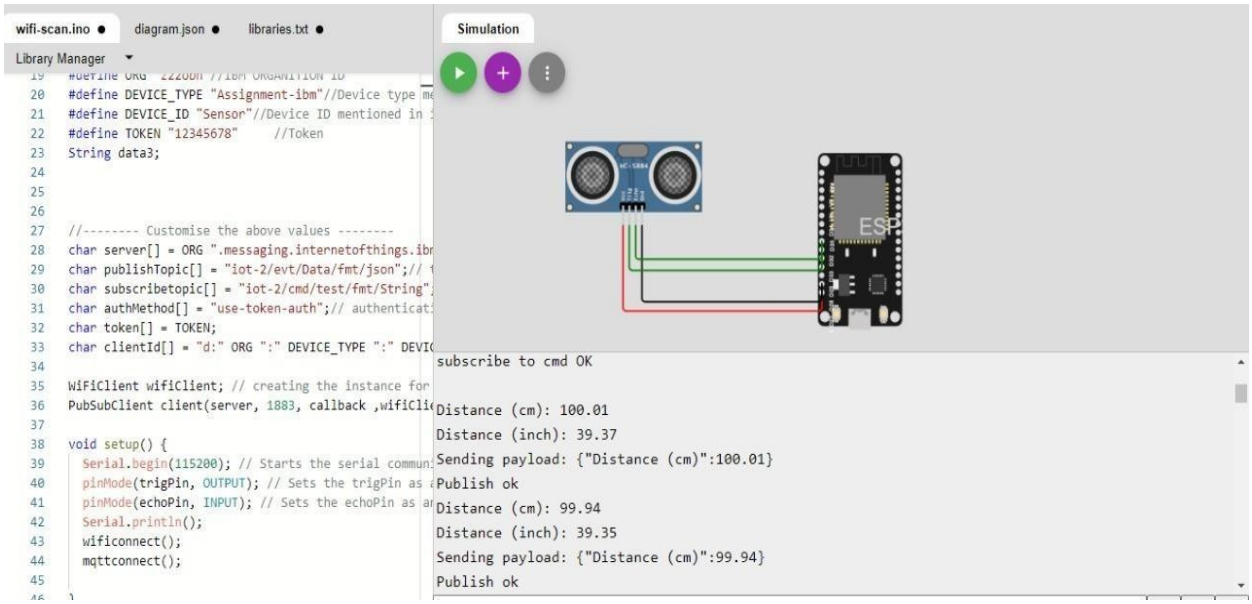
```
charpublishTopic[]="iot-2/evt/Data/fmt/json";//topicnameandtypeof event
perform and format in which data to be send
charsubscribetopic[]="iot-2/cmd/test/fmt/String";//cmd
                                                    REPRESENTcommandt
ypeANDCOMMANDISTESTOFFORMATSTRING
charauthMethod[]="use-token-auth";//authenticationmethod char
token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


WiFiClientwifiClient;//creatingtheinstanceforwificlient
PubSubClient client(server, 1883, callback ,wifiClient);

void setup() {
  Serial.begin(115200); // Starts the serial communication
  pinMode(trigPin, OUTPUT); // Sets the trigPin as an
  OutputpinMode(echoPin,INPUT);//SetstheechoPinasanInput
  Serial.println();
  wificonnect();
  mqttconnect();

}

void loop() {
  //ClearsthetrigPin
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  // Sets the trigPin on HIGH state for 10 micro
  secondsdigitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);

  // Reads the echoPin, returns the sound wave travel time in microseconds
  duration=pulseIn(echoPin,HIGH);
```

```
  // Calculate the distance
  distanceCm = duration * SOUND_SPEED/2;

  // Convert to inches
  distanceInch = distanceCm * CM_TO_INCH;

  //Printsthedistanceintheserialmonitor
  Serial.print("Distance (cm): ");
  Serial.println(distanceCm);
  Serial.print("Distance (inch): ");
  Serial.println(distanceInch);

  PublishData(distanceCm);
  delay(1000);
  if (!client.loop()) {
    mqttconnect();
  }
}
  void PublishData(float Cm) {
  mqttconnect();//function call for connecting to ibm
  /*
     creatingtheStringininformJSontoupdatethedatatoibmcloud
  */
  Stringpayload="{\"Distance(cm)\":";
  payload += Cm;
  payload += "}";
  Serial.print("Sending payload: ");
  Serial.println(payload);


  if (client.publish(publishTopic, (char*) payload.c_str())) {
```

```
    Serial.println("Publishok");//ifitsucessfullyuploaddataonthecloud  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("Reconnectingclientto");
    Serial.println(server);
    while(!!!client.connect(clientId,authMethod,token)){
      Serial.print(".");
      delay(500);
    }

     initManagedDevice();
     Serial.println();
  }
}
void wificonnect() //function defination forwificonnect
{
  Serial.println();
  Serial.print("Connecting to");

  WiFi.begin("Wokwi-GUEST","",6);//passingthewificredentialstoestablish 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 tocmd OK");
  } else
  {
    Serial.println("subscribe to cmd FAILED");
  }
}
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength)
{

  Serial.print("callbackinvokedfortopic:");
  Serial.println(subscribetopic);
  for (int i = 0; i < payloadLength; i++) {
    //Serial.print((char)payload[i]);
    data3 += (char)payload[i];
  }
}
```

Wokwi Output:



IBM Cloud Alert:

Event	Value	Format	Last Received
Data	{"Distance (cm)":99.98}	json	a few seconds ago
Data	{"Distance (cm)":99.96}	json	a few seconds ago
Data	{"Distance (cm)":99.98}	json	a few seconds ago
Data	{"Distance (cm)":99.98}	json	a few seconds ago
Data	{"Distance (cm)":99.98}	json	a few seconds ago

Wokwi Share Link:

https://wokwi.com/projects/305569599398609473