Assignment-4

Assignment date	31th October,2022
Student Name	Beninal. S
Roll Number	412919106002
Maximum marks	2 Marks

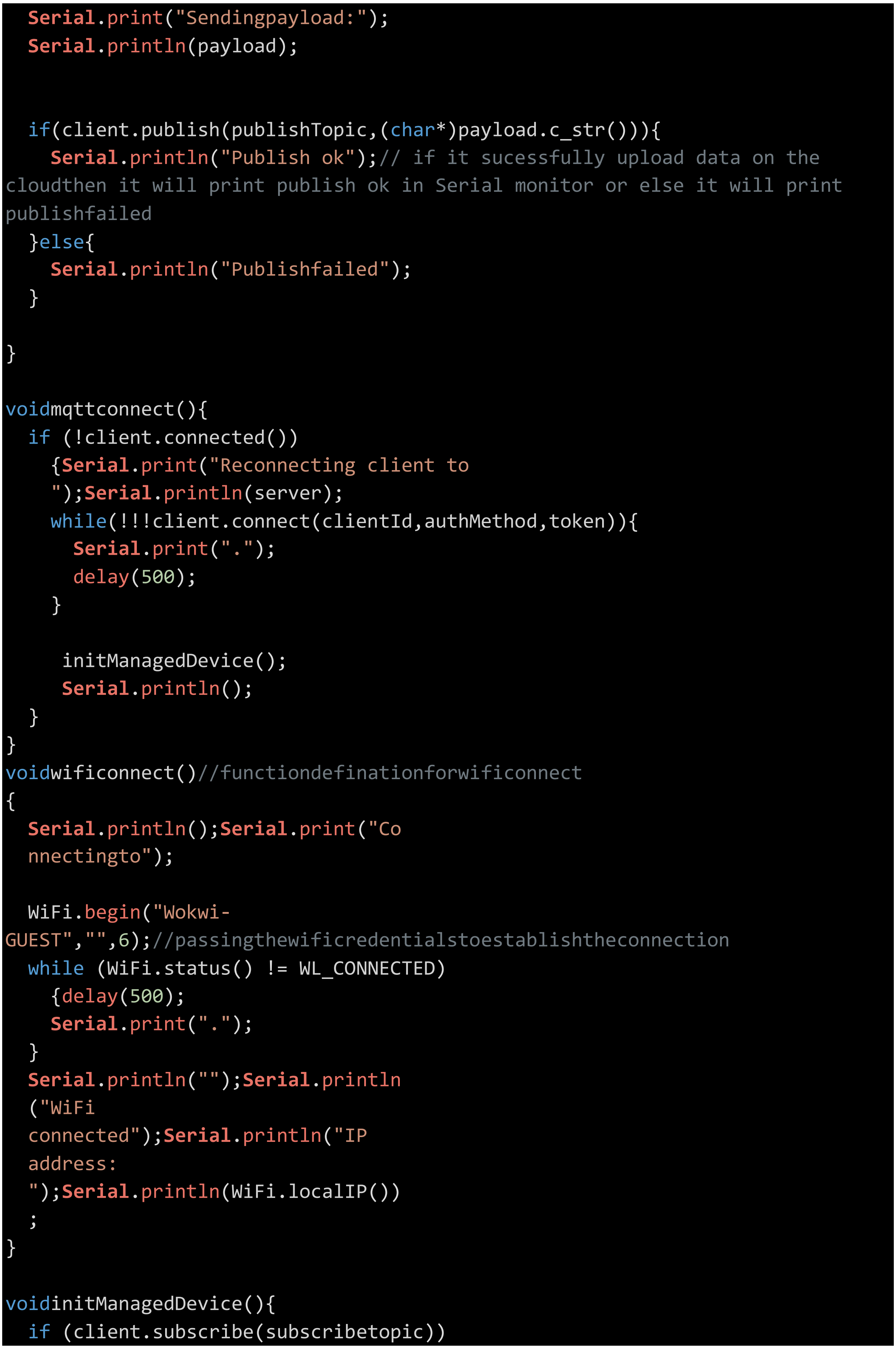
Question:

Write code and connections in wokwi for ultrasonic sensor. Whenever distance is less than 100cmssend “alert” to ibm cloud and display in device recent events. Upload document with wokwi sharelinkandimagesofibmcloud.

Code:

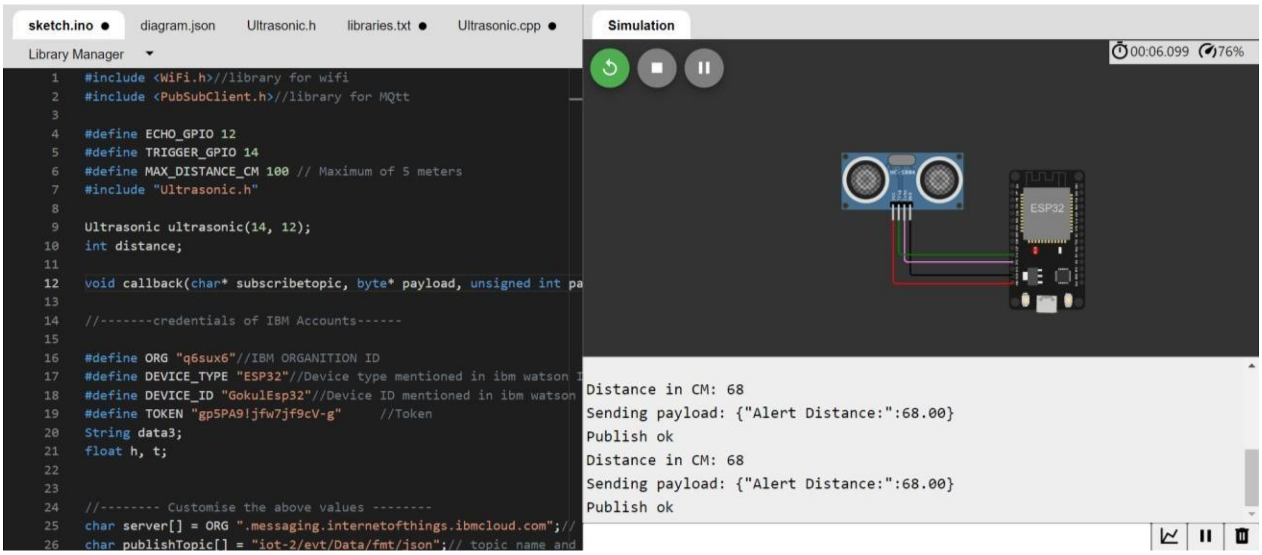
```
#include <WiFi.h>//library for
wifi#include<PubSubClient.h>//libraryforMQ
tt

#defineECHO_GPIO12
#defineTRIGGER_GPIO14
#defineMAX_DISTANCE_CM100//Maximumof5meters#inclu
de"Ultrasonic.h"
Ultrasonicultrasonic(14,12);i
ntdistance;

voidcallback(char*subscribetopic,byte*payload,unsignedintpayloadLength);

//-------credentialsofIBMAccounts------

#defineORG"q6sux6"//IBMORGANITIONID
#define DEVICE_TYPE "ESP32"//Device type mentioned in ibm watson IOT
Platform#defineDEVICE_ID"GokulEsp32"//DeviceIDmentionedinibmwatsonIOTPlatform
#defineTOKEN"gp5PA9!jfw7jf9cV-g"       //Token
Stringdata3;float
h,t;


//--------Customisetheabovevalues--------
charserver[]=ORG".messaging.internetofthings.ibmcloud.com";//ServerNamechar
publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type
ofeventperformandformatinwhichdatatobesend
charsubscribetopic[]="iot-
2/cmd/command/fmt/String";//cmdREPRESENTcommandtypeANDCOMMANDISTESTOFFORMATST
RING
charauthMethod[]="use-token-
auth";//authenticationmethodchartoken[]=TOKEN;
charclientId[]="d:"ORG":"DEVICE_TYPE":"DEVICE_ID;//clientid
```

```
//.................................................................................-
WiFiClient wifiClient; // creating the instance for
wificlientPubSubClient client(server, 1883, callback ,wifiClient);
//calling thepredefined client id by passing parameter like server
id,portandwificredential

voidsetup()//configureingtheESP32
{
  Serial.begin(115200);
  delay(10);Serial.prin
  tln();wificonnect();m
  qttconnect();
}

voidloop()//RecursiveFunction
{

  distance =
  ultrasonic.read(CM);if(distance <
  100){Serial.print("Distance in
  CM:
  ");Serial.println(distance);Publi
  shData(distance);delay(1000);
  if (!client.loop())
    {mqttconnect();
  }

  }

  delay(1000);


}

/*....................................retrievingto
Cloud. ............................ */

void PublishData(float temp)
  {mqttconnect();//functioncallforconnectingtoibm
  /*
     creatingtheStringininformJSontoupdatethedatatoibmcloud
  */
  String payload = "{\"Alert
  Distance:\":";payload+=temp;
  payload+="}";
```



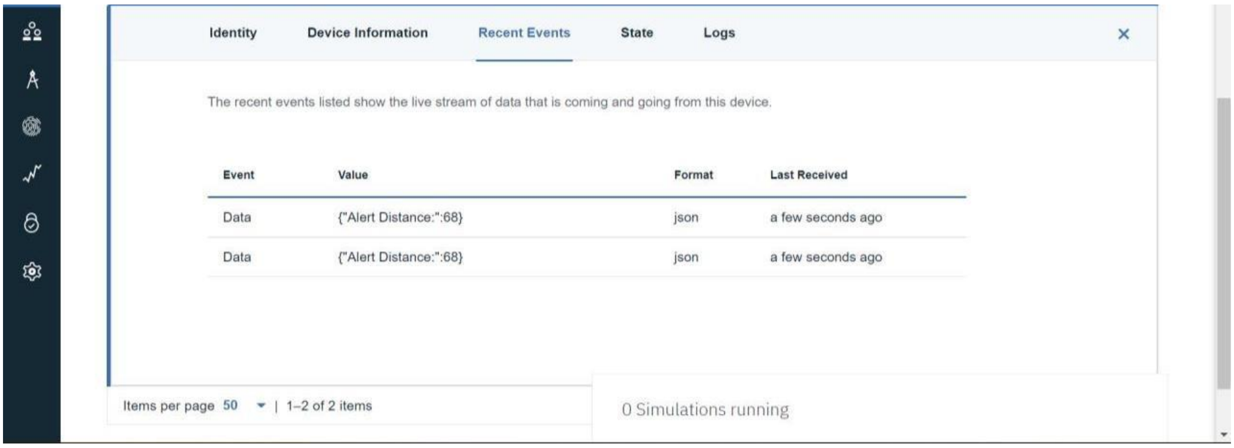
```
  Serial.print("Sendingpayload:");
  Serial.println(payload);


  if(client.publish(publishTopic,(char*)payload.c_str())){
    Serial.println("Publish ok");// if it sucessfully upload data on the
cloudthen it will print publish ok in Serial monitor or else it will print
publishfailed
  }else{
    Serial.println("Publishfailed");
  }

}

voidmqttconnect(){
  if (!client.connected())
    {Serial.print("Reconnecting client to
    ");Serial.println(server);
    while(!!!client.connect(clientId,authMethod,token)){
      Serial.print(".");
      delay(500);
    }

     initManagedDevice();
     Serial.println();
  }
}
voidwificonnect()//functiondefinationforwificonnect
{
  Serial.println();Serial.print("Co
  nnectingto");

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

voidinitManagedDevice(){
  if (client.subscribe(subscribetopic))
    {Serial.println((subscribetopic));Serial.println(
    "subscribetocmdOK");
  }else{
    Serial.println("subscribetocmdFAILED");
  }
}

voidcallback(char*subscribetopic,byte*payload,unsignedintpayloadLength)
{

  Serial.print("callbackinvokedfortopic:");
  Serial.println(subscribetopic);
  for(inti= 0;i<payloadLength;
    i++){data3+=(char)payload[i];
  }
  Serial.println("data:"+data3);i
  f(data3=="lighton")
  {
Serial.println(data3);
  }
  else
  {
Serial.println(data3);
  }
data3="";
}
```

Output:



```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt

#define ECHO_GPIO 12
#define TRIGGER_GPIO 14
#define MAX_DISTANCE_CM 100 // Maximum of 5 meters
#include "Ultrasonic.h"

Ultrasonic ultrasonic(14, 12);
int distance;

void callback(char* subscribetopic, byte* payload, unsigned int pa

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

#define ORG "q6sux6"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32"//Device type mentioned in ibm watson I
#define DEVICE_ID "GokulEsp32"//Device ID mentioned in ibm watson
#define TOKEN "gp5PA9!jfw7jf9cV-g"      //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";//
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and
```

```
Distance in CM: 68
Sending payload: {"Alert Distance:":68.00}
Publish ok
Distance in CM: 68
Sending payload: {"Alert Distance:":68.00}
Publish ok
```

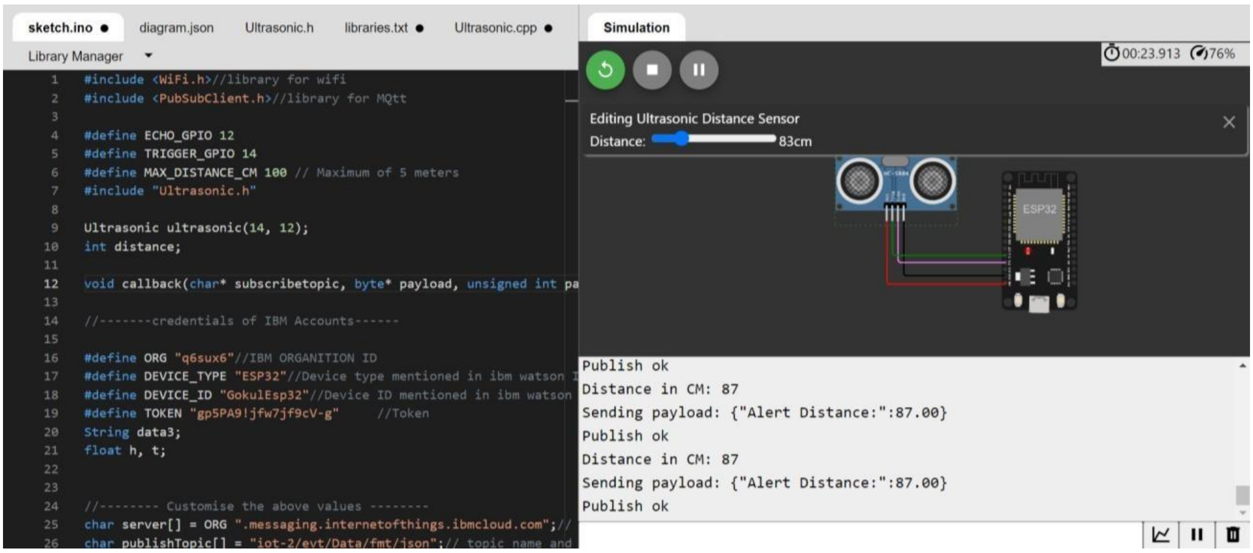


The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
Data	{"Alert Distance:":68}	json	a few seconds ago
Data	{"Alert Distance:":68}	json	a few seconds ago

Items per page 50 | 1–2 of 2 items

0 Simulations running

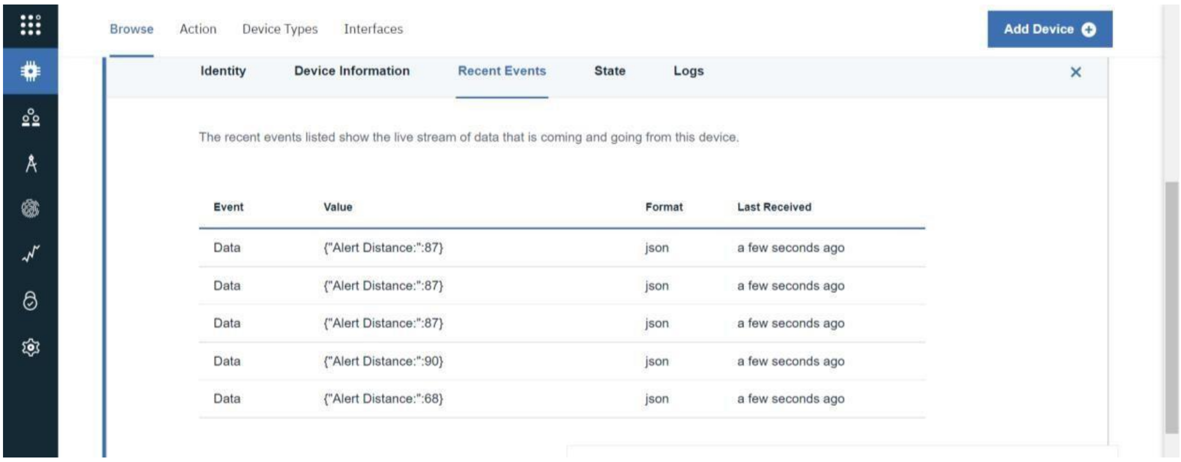


```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt

#define ECHO_GPIO 12
#define TRIGGER_GPIO 14
#define MAX_DISTANCE_CM 100 // Maximum of 5 meters
#include "Ultrasonic.h"

Ultrasonic ultrasonic(14, 12);
int distance;

void callback(char* subscribetopic, byte* payload, unsigned int pa

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

#define ORG "q6sux6"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32"//Device type mentioned in ibm watson I
#define DEVICE_ID "GokulEsp32"//Device ID mentioned in ibm watson
#define TOKEN "gp5PA9!jfw7jf9cV-g"     //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";//
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and
```

Editing Ultrasonic Distance Sensor

Distance: 83cm

```
Publish ok
Distance in CM: 87
Sending payload: {"Alert Distance:":87.00}
Publish ok
Distance in CM: 87
Sending payload: {"Alert Distance:":87.00}
Publish ok
```



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

Event	Value	Format	Last Received
Data	{"Alert Distance:":87}	json	a few seconds ago
Data	{"Alert Distance:":87}	json	a few seconds ago
Data	{"Alert Distance:":87}	json	a few seconds ago
Data	{"Alert Distance:":90}	json	a few seconds ago
Data	{"Alert Distance:":68}	json	a few seconds ago

ProjectLink:
https://wokwi.com/projects/346773638705316434