ASSIGNMENT–4

UltrasonicsensorsimulationinWokwi

|  |  |
| --- | --- |
| Name | Soumya S |
| ProjectName | RealTimeWaterQualityMonitoringandControlSystem |

**QUESTION:**

Writecodeandconnectionsinwokwifortheultrasonicsensor.Wheneverthedistanceislessthan100cms sendan"alert"to theIBMcloud anddisplay inthedevicerecentevents.

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

void callback(char\* subscribetopic, byte\* payload, unsigned intpayloadLength);

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

#defineORG"ksus4d"//IBMORGANITIONID

#define DEVICE\_TYPE "Sensordata"//Device type mentioned in ibm watson IOTPlatform

#define DEVICE\_ID "Selva18"//Device ID mentioned in ibm watson IOT Platform#defineTOKEN"Realtimewater"//Token

Stringdata3;

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";charpublishTopic[]= "iot-2/evt/Data/fmt/json";

char subscribetopic[] = "iot-2/cmd/test/fmt/String";charauthMethod[]= "use-token-auth";

chartoken[]=TOKEN;

charclientId[]="d:"ORG":"DEVICE\_TYPE":"DEVICE\_ID;

WiFiClientwifiClient;

PubSubClient client(server, 1883, callback, wifiClient);constinttrigPin= 5;

constintechoPin=18;

#define SOUND\_SPEED 0.034longduration;

float distance;voidsetup(){

**Serial**.begin(115200);pinMode(trigPin,OUTPUT);

pinMode(echoPin, INPUT);wificonnect();mqttconnect();

}

voidloop()

{

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("Sendingpayload:");

**Serial**.println(payload);

if(client.publish(publishTopic,(char\*)payload.c\_str())){

**Serial**.println("Publishok");

} 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()

{

**Serial**.println();**Serial**.print("Connecting to ");WiFi.begin("Wokwi-GUEST","",6);

while (WiFi.status() != WL\_CONNECTED) {delay(500);

**Serial**.print(".");

}

**Serial**.println("");**Serial**.println("WiFiconnected");**Serial**.println("IP address: ");**Serial**.println(WiFi.localIP());

}

voidinitManagedDevice(){

if (client.subscribe(subscribetopic)) {**Serial**.println((subscribetopic));**Serial**.println("subscribetocmdOK");

} else{

**Serial**.println("subscribeto cmd FAILED");

}

}

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

{

**Serial**.print("callbackinvokedfortopic:");

**Serial**.println(subscribetopic);

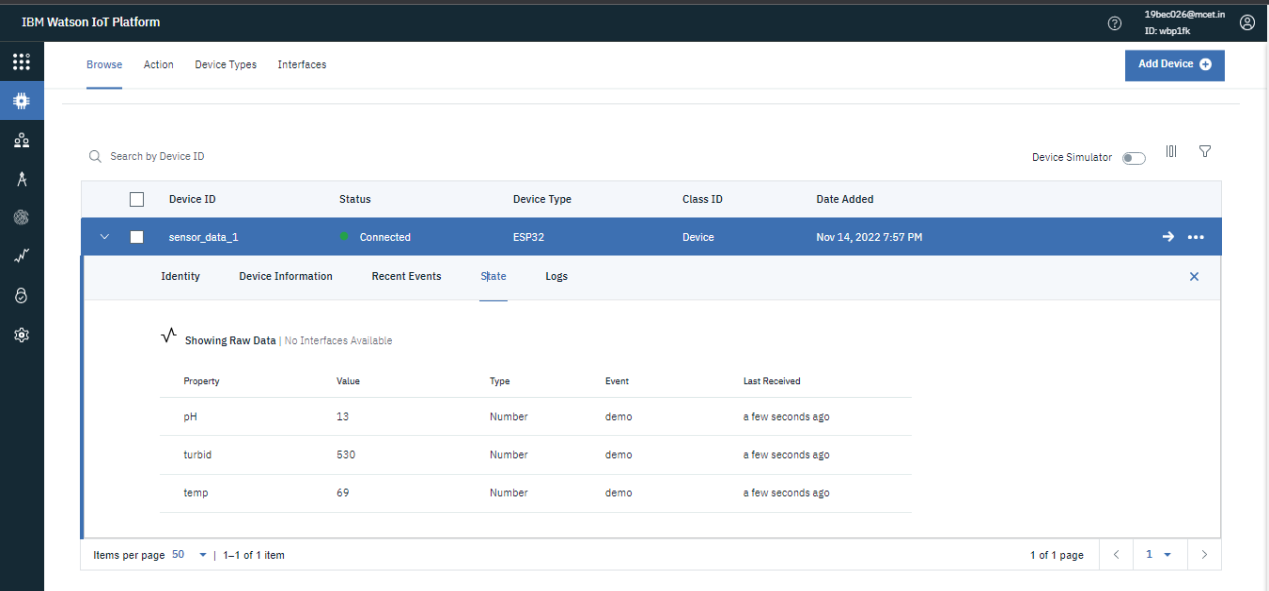
for (inti=0;i<payloadLength;i++){

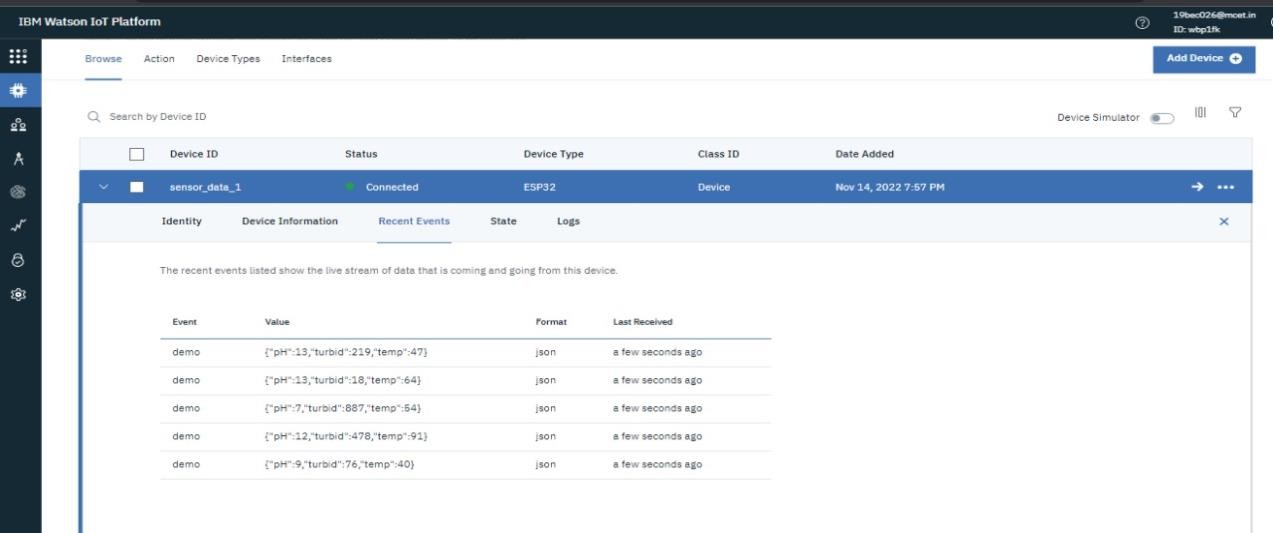
//Serial.print((char)payload[i]);data3+=(char)payload[i];

}

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

}





{

"version":1,

"author": "SRIHARANV",

"editor": "wokwi","parts":[

{ "type": "wokwi-esp32-devkit-v1", "id": "esp", "top": -56, "left": -120,"attrs":{}},

{ "type": "wokwi-hc-sr04", "id": "ultrasonic1", "top": -36.04, "left": 27.5,"attrs":{}}

],

"connections":[

[ "esp:TX0","$serialMonitor:RX","",[]],

["esp:RX0","$serialMonitor:TX","",[]],

["ultrasonic1:VCC","esp:VIN","red",["v87.91","h-246.45","v-36" ]],

[ "ultrasonic1:TRIG","esp:D4","green",["v0"] ],

[ "ultrasonic1:ECHO", "esp:D21", "magenta", [ "v53.24", "h-116.89", "v-88.67" ] ],["ultrasonic1:GND","esp:GND.1", "black",["v0"]

