

## ASSIGNMENT IV

### Python Programming

Assignment Date	25 October 2022
Student Name	Vijay A
Student Roll Number	822119106026
Maximum Marks	2 Marks

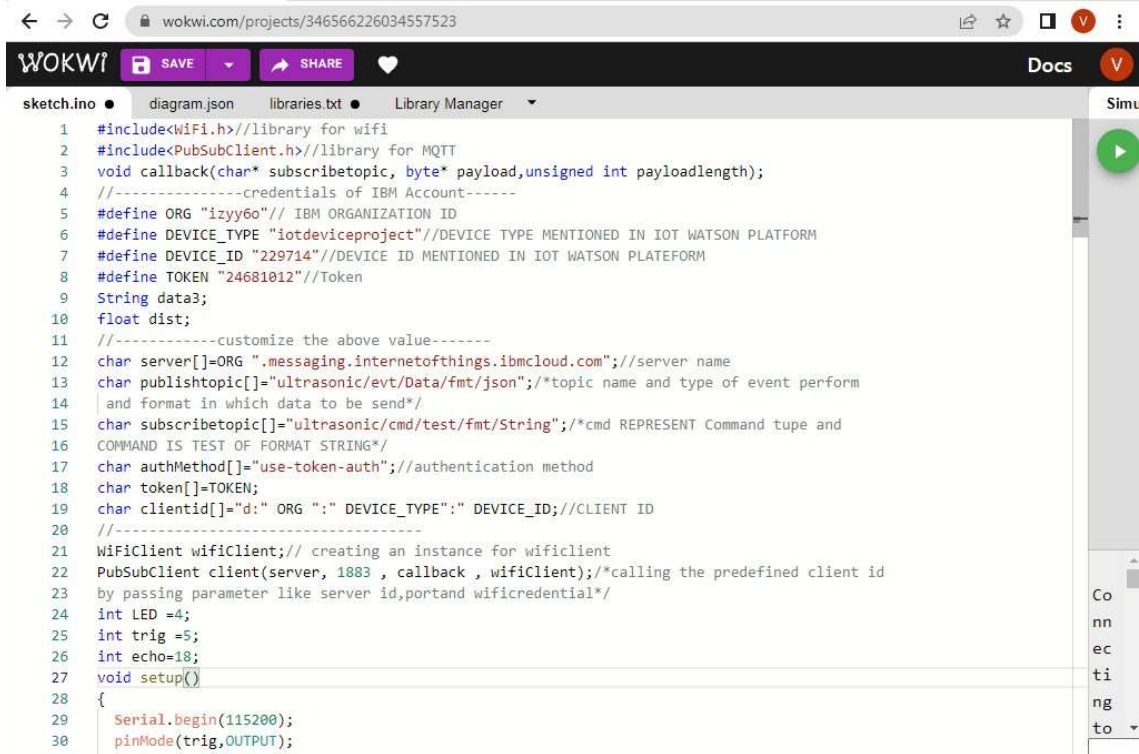
#### Question-1:

Write code and connections in wokwi for ultrasonic sensor.

Whenever distance is less than 100 cms send "alert" to ibm cloud and display in devicerecent events.

Upload document with wokwi share link and images of ibm cloud

#### Solution:



```
1 #include<WiFi.h> //library for wifi
2 #include<PubSubClient.h> //library for MQTT
3 void callback(char* subscribetopic, byte* payload,unsigned int payloadlength);
4 //-----credentials of IBM Account-----
5 #define ORG "izyy6o" // IBM ORGANIZATION ID
6 #define DEVICE_TYPE "iotdeviceproject" //DEVICE TYPE MENTIONED IN IOT WATSON PLATFORM
7 #define DEVICE_ID "229714" //DEVICE ID MENTIONED IN IOT WATSON PLATFORM
8 #define TOKEN "24681012" //Token
9 String data3;
10 float dist;
11 //-----customize the above value-----
12 char server[] = ORG ".messaging.internetofthings.ibmcloud.com"; //server name
13 char publishtopic[] = "ultrasonic/evt/Data/fmt/json"; /*topic name and type of event perform
14 and format in which data to be send*/
15 char subscribetopic[] = "ultrasonic/cmd/test/fmt/String"; /*cmd REPRESENT Command tupe and
16 COMMAND IS TEST OF FORMAT STRING*/
17 char authMethod[] = "use-token-auth"; //authentication method
18 char token[] = TOKEN;
19 char clientid[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID; //CLIENT ID
20 //-----
21 WiFiClient wificlient; // creating an instance for wificlient
22 PubSubClient client(server, 1883, callback, wificlient); /*calling the predefined client id
23 by passing parameter like server id,portand wificredential*/
24 int LED = 4;
25 int trig = 5;
26 int echo = 18;
27 void setup()
28 {
29   Serial.begin(115200);
30   pinMode(trig,OUTPUT);
```

← → ↻

wokwi.com/projects/346566226034557523

🔖 ☆ 🗄 ⌵

WOKWI

SAVE

SHARE

🔖

Docs

⌵

sketch.ino

diagram.json

libraries.txt

Library Manager

Simu

```
31 pinMode(echo,INPUT);
32 pinMode(LED,OUTPUT);
33 delay(10);
34 wificonnect();
35 mqttconnect();
36 }
37 void loop()//recursive function
38 {
39   digitalWrite(trig,LOW);
40   digitalWrite(trig,HIGH);
41   delayMicroseconds(10);
42   digitalWrite(trig,LOW);
43   float dur=pulseIn(echo,HIGH);
44   float dist=(dur * 0.0343)/2;
45   Serial.print("distance in cm");
46   Serial.println(dist);
47   PublishData(dist);
48   delay(1000);
49   if (!client.loop()){
50     mqttconnect();
51   }
52 }
53 /*.....retriving to cloud.....*/
54 void PublishData(float dist){
55   mqttconnect();//function call for connecting to ibm
56   /*creating the string in form of JSON to update the data to ibm cloud*/
57   String object;
58   if(dist<100)
59   {
60     digitalWrite(LED,HIGH);
```

Co  
nn  
ec  
ti  
ng  
to

← → ↻

wokwi.com/projects/346566226034557523

🔖 ☆ 🗄 ⌵

WOKWI

SAVE

SHARE

🔖

Docs

⌵

sketch.ino

diagram.json

libraries.txt

Library Manager

Simu

```
61 Serial.println("no object is near");
62 object="Near";
63 }
64 else
65 {
66   digitalWrite(LED,LOW);
67   Serial.println("no object found");
68   object="No";
69 }
70 String payload="{\"distance\":";
71 payload +=dist;
72 payload +=",\"object\":\":";
73 payload += object;
74 payload += "\":";
75
76 Serial.print("Sending payload: ");
77 Serial.println(payload);
78 if(client.publish(publishtopic, (char*) payload.c_str())){
79   Serial.println("Publish ok");/* if its sucessfully upload data on the cloud then it will print
80   publish ok in serial monitor or else it will print publish failed*/
81 } else{
82   Serial.println("Publish failed");
83 }
84 }
85 void mqttconnect(){
86   if(!client.connected()){
87     Serial.print("Reconnecting client to ");
88     Serial.println(server);
89     while(!client.connect(clientid,authMethod, token)){
90       Serial.print(".");
91       delay(500);
```

Co  
nn  
ec  
ti  
ng  
to

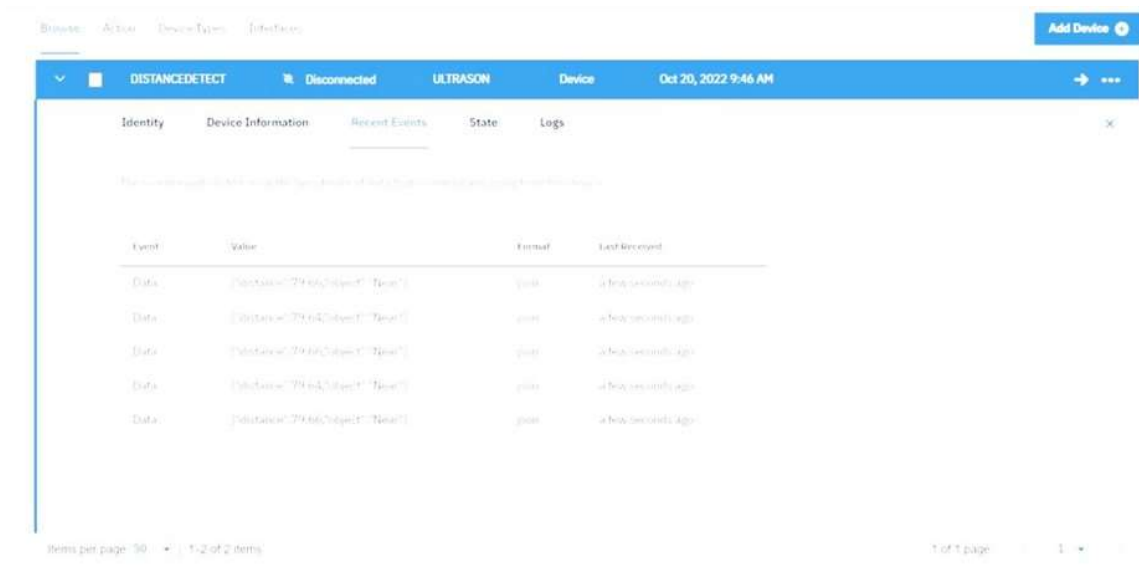
```
WOKWI SAVE SHARE Docs V
sketch.ino diagram.json libraries.txt Library Manager Simu
92 }
93   initManagedDevice();
94   Serial.println();
95 }
96
97 void wificonnect()//function defenition for wificonnect
98 {
99   Serial.println();
100   Serial.print("Connecting to ");
101   WiFi.begin("Wokwi.GUEST", "",6);//PASSING THE WIFI CREDENTIALS TO ESTABLISH CONNECTION
102   while (WiFi.status() !=WL_CONNECTED){
103     delay(500);
104     Serial.print(".");
105   }
106   Serial.println("");
107   Serial.println("WiFi connected");
108   Serial.println("IP address");
109   Serial.println(WiFi.localIP());
110 }
111 void initManagedDevice(){
112   if(client.subscribe(subscribetopic)){
113     Serial.println((subscribetopic));
114     Serial.println("subscribe to cmd OK");
115   }else{
116     Serial.println("subscribe to cmd failed");
117   }
118 }
119 void callback(char* subscribetopic,byte*payload,unsigned int payloadLength)
120 {
121   Serial.print("callback invoked for topic: ");
122   Serial.println(subscribetopic);
```

```
WOKWI SAVE SHARE Docs V
sketch.ino diagram.json libraries.txt Library Manager Simu
123   for(int i=0; i< payloadLength; i++){
124     //Serial.print((char)payload[i]);
125     data3 +=(char)payload[i];
126   }
127   //Serial.println("dta: "+ data3);
128   //if(data3=="Near")
129   //{
130   //Serial.println(data3);
131   //digitalWrite(LED,HIGH);
132   //}
133   //else
134   //{
135   //Serial.println(data3);
136   //digitalWrite(LED,LOW);
137   //}
138   data3="";
139 }
```

OUTPUT:

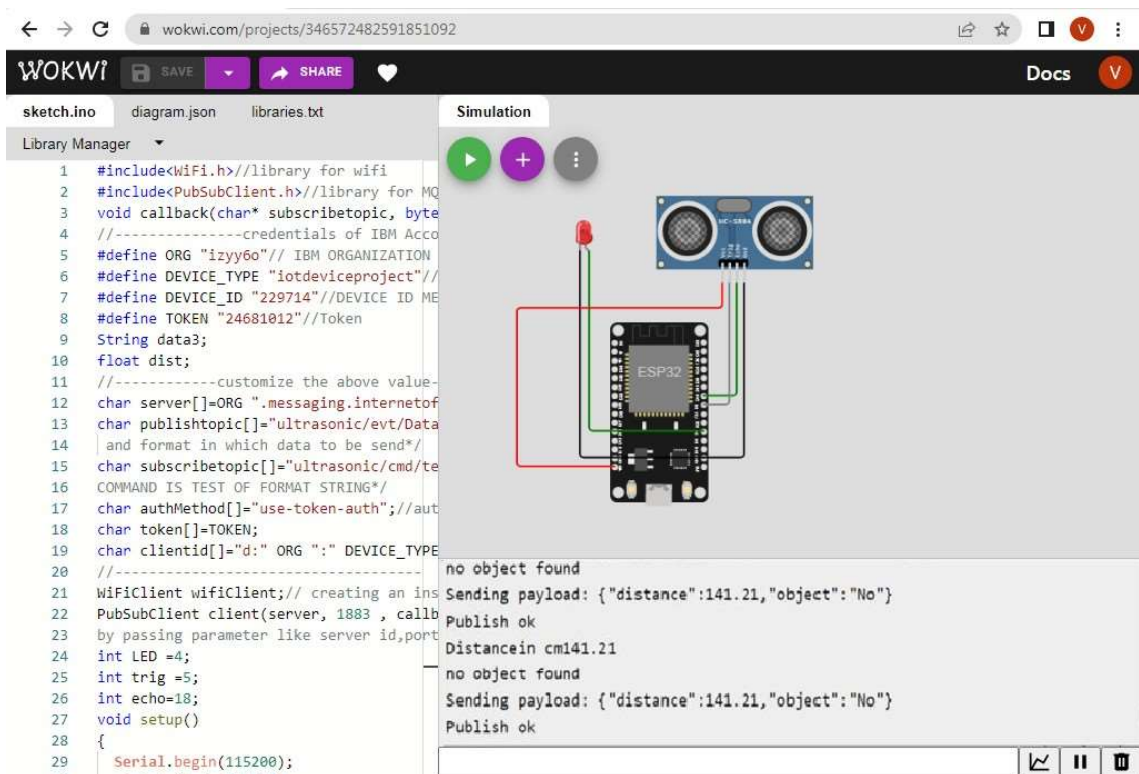
<https://wokwi.com/projects/346572482591851092>

## DATA SENT TO IBM CLOUD ON NO OBJECT DETECTED



Event	Value	Format	Last Received
Data	{\"distance\":141.21,\"object\":\"No\"}	json	4 hrs, 55 seconds ago
Data	{\"distance\":141.21,\"object\":\"No\"}	json	4 hrs, 55 seconds ago
Data	{\"distance\":141.21,\"object\":\"No\"}	json	4 hrs, 55 seconds ago
Data	{\"distance\":141.21,\"object\":\"No\"}	json	4 hrs, 55 seconds ago
Data	{\"distance\":141.21,\"object\":\"No\"}	json	4 hrs, 55 seconds ago

## WHEN NO OBJECT DETECTED BY ULTRASONIC DETECTOR



```
1 #include<WiFi.h>//library for wifi
2 #include<PubSubClient.h>//library for MQ
3 void callback(char* subscribetopic, byte
4 //-----credentials of IBM Acco
5 #define ORG "izyy6o"// IBM ORGANIZATION
6 #define DEVICE_TYPE "iotdeviceproject"//
7 #define DEVICE_ID "229714"//DEVICE ID ME
8 #define TOKEN "24681012"//Token
9 String data3;
10 float dist;
11 //-----customize the above value-
12 char server[]=ORG ".messaging.internetof
13 char publishtopic[]="ultrasonic/evt/Data
14 and format in which data to be send*/
15 char subscribetopic[]="ultrasonic/cmd/te
16 COMMAND IS TEST OF FORMAT STRING*/
17 char authMethod[]="use-token-auth";//aut
18 char token[]=TOKEN;
19 char clientId[]="d:" ORG ":" DEVICE_TYPE
20 //-----
21 WiFiClient wifiClient;// creating an ins
22 PubSubClient client(server, 1883, callb
23 by passing parameter like server id,port
24 int LED =4;
25 int trig =5;
26 int echo=18;
27 void setup()
28 {
29   Serial.begin(115200);
```

Simulation

no object found  
Sending payload: {\"distance\":141.21,\"object\":\"No\"}  
Publish ok  
Distancein cm141.21  
no object found  
Sending payload: {\"distance\":141.21,\"object\":\"No\"}  
Publish ok

## DATA SENT TO IBM CLOUD ON OBJECT BEING DETECTED

Wokwi IoT Platform interface showing a device named DISTANCEDETECT (ULTRASON) with a status of Disconnected. The interface displays a table of recent events.

Event	Value	Format	Last Received
Data	{"distance": 97.82, "object": "Near"}	json	14 hrs, 2 seconds ago
Data	{"distance": 97.82, "object": "Near"}	json	14 hrs, 2 seconds ago
Data	{"distance": 97.82, "object": "Near"}	json	14 hrs, 2 seconds ago
Data	{"distance": 97.82, "object": "Near"}	json	14 hrs, 2 seconds ago
Data	{"distance": 97.82, "object": "Near"}	json	14 hrs, 2 seconds ago

Items per page: 50 | 1-2 of 2 items | 1 of 1 page

## WHEN OBJECT DETECTED BY ULTRASONIC DETECTOR SENSOR

Wokwi IoT Platform interface showing a simulation of an ESP32 microcontroller connected to an HC-SR04 ultrasonic sensor. The simulation is running, and the console output shows the sensor detecting an object near the sensor.

```

1 object is near
1 Sending payload: {"distance":97.82,"object":"Near"}
1 Publish ok
1 Distance in cm 97.82
2 object is near
2 Sending payload: {"distance":97.82,"object":"Near"}
2 Publish ok
2
2

```