

# IBM ASSIGNMENT - 4

NAME: NIMISHA C

ROLL NO: 7376191EC214

CODE:

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

WiFiClient wifiClient;

#define ORG "uob8bn"
#define DEVICE_TYPE "Microprocessor"
#define DEVICE_ID "143"
#define TOKEN "Nimisha001"
#define speed 0.034

char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char topic[] = "iot-2/cmd/home/fmt/String";
char authMethod[] = "use-token-auth";
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
PubSubClient client(server, 1883, wifiClient);
void publishData();

const int trigpin=5;
const int echopin=18;
String command;
String data="";

long duration;
int dist;

void setup()
{
    Serial.begin(115200);
    pinMode(trigpin, OUTPUT);
    pinMode(echopin, INPUT);
    wifiConnect();
    mqttConnect();
}
```

```

void loop() {

    publishData();
    delay(500);

    if (!client.loop()) {
        mqttConnect();
    }
}

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

void mqttConnect() {
    if (!client.connected()) {
        Serial.print("Reconnecting MQTT client to "); Serial.println(server);
        while (!client.connect(clientId, authMethod, token)) {
            Serial.print(".");
            delay(1000);
        }
        initManagedDevice();
        Serial.println();
    }
}

void initManagedDevice() {
    if (client.subscribe(topic)) {
        Serial.println(client.subscribe(topic));
        Serial.println("subscribe to cmd OK");
    } else {
        Serial.println("subscribe to cmd FAILED");
    }
}

void publishData()
{
    digitalWrite(trigpin, LOW);
    digitalWrite(trigpin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigpin, LOW);
    duration=pulseIn(echopin, HIGH);
}

```

```

dist=duration*speed/2;

if(dist<100){
  DynamicJsonDocument doc(1024);
  String payload;
  doc["AlertDistance:"]=dist;
  serializeJson(doc, payload);
  delay(3000);
  Serial.print("\n");
  Serial.print("Sending payload: ");
  Serial.println(payload);
  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish OK");
  } else {
    Serial.println("Publish FAILED");
  }
}
}
}

```

## OUTPUT:

The screenshot displays the Wokwi IDE interface. On the left, the 'sketch.ino' file contains the following code:

```

1 #include <Wifi.h>
2 #include <PubSubClient.h>
3 #include <ArduinoJson.h>
4
5 WifiClient wifiClient;
6
7 #define ORG "uob8bn"
8 #define DEVICE_TYPE "Microprocessor"
9 #define DEVICE_ID "143"
10 #define TOKEN "Nimisha001"
11 #define speed 0.034
12
13 char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
14 char publishTopic[] = "iot-2/evt/Data/fmt/json";
15 char topic[] = "iot-2/cmd/home/fmt/String";
16 char authMethod[] = "use-token-auth";
17 char token[] = TOKEN;
18 char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
19 PubSubClient client(server, 1883, wifiClient);
20 void publishData();
21
22 const int trigpin=5;
23 const int echopin=18;
24 String command;
25 String data="";
26
27 long duration;
28 int dist;
29
30 https://wokwi.com/arduino/setup()

```

On the right, the 'Simulation' window shows a visual representation of the hardware: an ESP32 microcontroller connected to an HC-SR04 ultrasonic sensor. Below the simulation, the output console shows the following messages:

```

Publish OK
Sending payload: {"AlertDistance":98}
Publish OK
Sending payload: {"AlertDistance":98}
Publish OK

```

BrowseActionDevice TypesInterfaces

Add Device

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

Event	Value	Format	Last Received
Data	{"AlertDistance":"","98}	json	a few seconds ago
Data	{"AlertDistance":"","98}	json	a few seconds ago
Data	{"AlertDistance":"","98}	json	a few seconds ago
Data	{"AlertDistance":"","98}	json	a few seconds ago
Data	{"AlertDistance":"","98}	json	a few seconds ago

Items per page 50 | 1-1 of 1 item1 of 1 page<1>

LINK:

<https://wokwi.com/projects/348583257085837906>