Assignment -4

Student Name	Aakash.P
Student Roll Number	720319106001

Question-1:

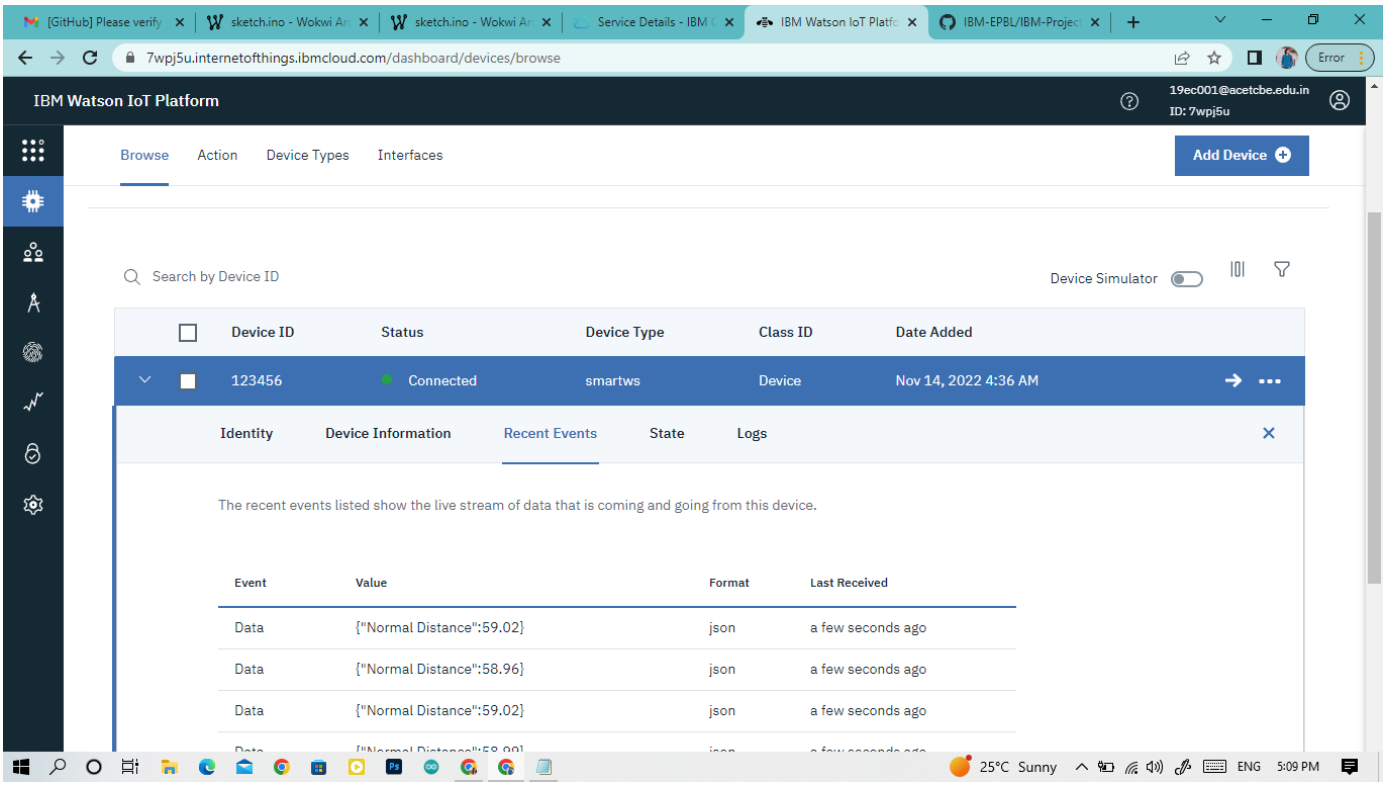
Wokwi Link: https://wokwi.com/projects/348008716247761491

code:

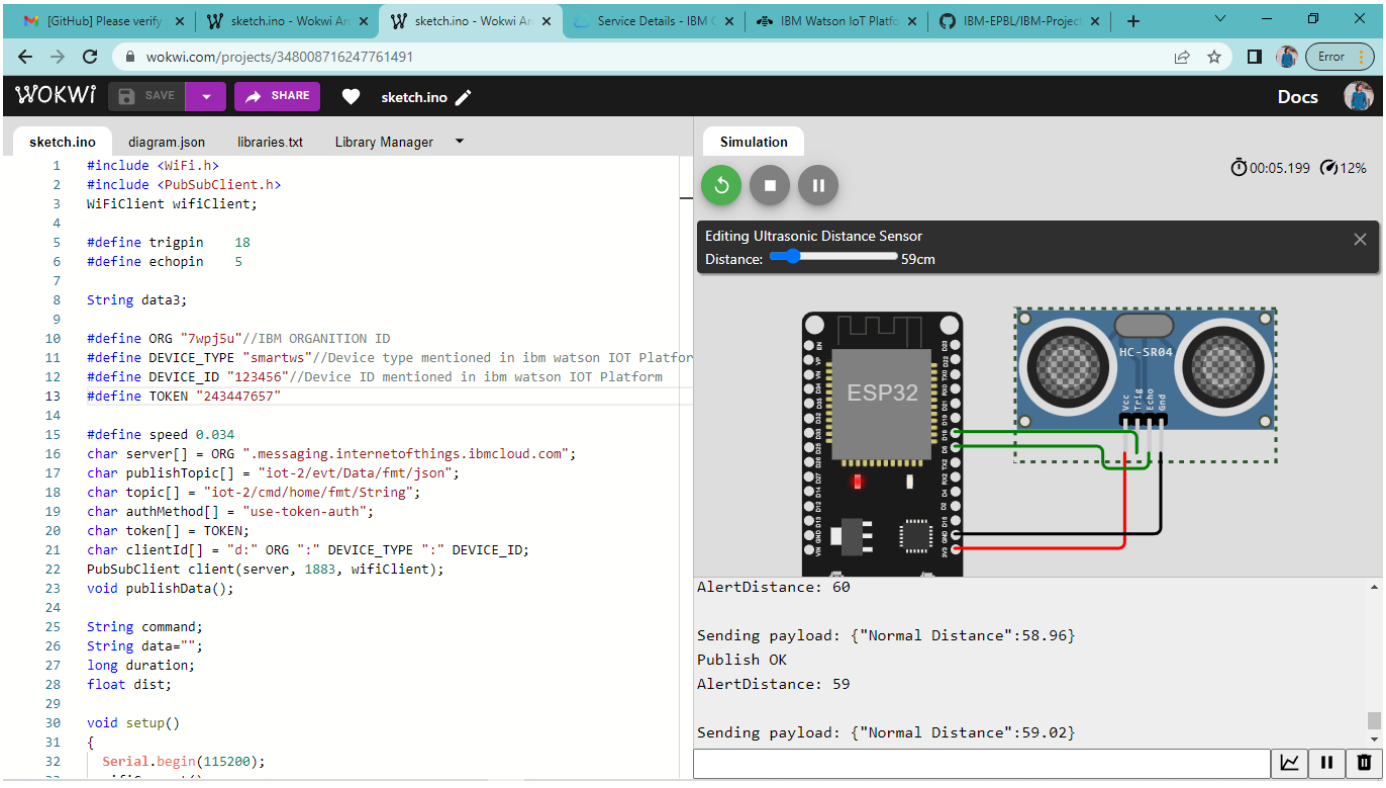
```
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;

#define trigpin    18
#define echopin    5

String data3;

#define ORG "7wpj5u"//IBM ORGANITION ID
#define DEVICE_TYPE "smartws"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "123456"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "243447657"

#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();

String command;
String data="";
long duration;
float dist;

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

 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 loop()
{

    int pulseWidth = 0;
    digitalWrite(trigpin, HIGH);
    delayMicroseconds(100);
    digitalWrite(trigpin, LOW);
    pulseWidth = pulseIn(echopin, HIGH);
    Serial.print("AlertDistance: ");
    Serial.println(pulseWidth/58);

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

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

void initManagedDevice() {
  if (client.subscribe(topic)) {
    // Serial.println(client.subscribe(topic));
    Serial.println("IBM 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){
    String payload = "{\"Normal Distance\":";
    payload += dist;
    payload += "}";

    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
    if (client.publish(publishTopic, (char*) payload.c_str())) {
      Serial.println("Publish OK");
    }

  }
    if(dist>101 && dist<111){
    String payload = "{\"Alert distance\":";
    payload += dist;
    payload += "}";

    Serial.print("\n");
    Serial.print("Sending payload: ");
    Serial.println(payload);
     if(client.publish(publishTopic, (char*) payload.c_str())) {
      Serial.println("Warning crosses 110cm -- it automaticaly of the loop");

    }else {
      Serial.println("Publish FAILED");
    }

  }


  }
  void callback(char* subscribeTopic, byte* payload, unsigned int payloadLength){
  Serial.print("callback invoked for topic:");
  Serial.println(subscribeTopic);
  for(int i=0; i<payloadLength; i++){
    dist += (char)payload[i];
  }
  Serial.println("data:"+ data3);
  if(data3=="lighton"){
    Serial.println(data3);

  }
  data3="";
}
```


IBM Watson IoT Platform
Browse
Action
Device Types
Interfaces
Add Device
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
123456
Connected
smartws
Device
Nov 14, 2022 4:36 AM
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
Data
{"Normal Distance":59.02}
json
a few seconds ago
Data
{"Normal Distance":58.96}
json
a few seconds ago
Data
{"Normal Distance":59.02}
json
a few seconds ago

Output:


WOKWI
SAVE
SHARE
sketch.ino
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
#include <WiFi.h>
#include <PubSubClient.h>
WiFiClient wifiClient;
#define trigpin 18
#define echopin 5
String data3;
#define ORG "7wpj5u"//IBM ORGANITION ID
#define DEVICE_TYPE "smartws"//Device type mentioned in ibm watson IOT Platfor
#define DEVICE_ID "123456"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "243447657"
#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();
String command;
String data="";
long duration;
float dist;
void setup()
{
Serial.begin(115200);
Simulation
00:05.199
12%
Editing Ultrasonic Distance Sensor
Distance:
59cm
ESP32
HC-SR04
AlertDistance: 60
Sending payload: {"Normal Distance":58.96}
Publish OK
AlertDistance: 59
Sending payload: {"Normal Distance":59.02}