ASSIGNMENT4

WOKWI PROGRAM

ASSIGNMNET DATE	23 OCTOBER 2022
STUDENT NAME	LAVANYA V
STUDENT ROLL NUMBER	110519106701
MAXIMUM NUMBER	2 MARKS
TEAM ID	PNT2022TMID36194

CODE :

```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic,byte* payload, unsigned int payloadLength);
#define ORG "y6u2ck"
#define DEVICE_TYPE "lavanya3104"
#define DEVICE_ID "3104"
#define token_"dyz3s1)BvKB1DmAMH"
String data3;

char server[]= ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[]="iot-2/evt/lavanya3104/fmt/json";
char subscribeTopic[]="iot-2/cmd/test/fmt/String";
char authMethod[]="use-token-auth";
char token[]=TOKEN;
char clientID[]="d:"ORG":"DEVICE_TYPE":"DEVICE_ID;
```

```
WiFiClientwifiClient;
PubSubClient client(server,1883,callback,wifiClient);

#define ECHO_PIN 12
#define TRIG_PIN 13
#define led 14

void setup() {
  // put your setup code here, to run once:
Serial.begin(115200);
pinMode(led, OUTPUT);
pinMode(TRIG_PIN, OUTPUT);
pinMode(ECHO_PIN, INPUT);
wificonnect();
mqttconnect();
}
float readDistanceCM() {
digitalWrite(TRIG_PIN, LOW);
delayMicroseconds(2);
digitalWrite(TRIG_PIN, HIGH);
delayMicroseconds(10);
digitalWrite(TRIG_PIN, LOW);
  int duration=random(1,200);
```

```
  //Serial.println(duration);
  //duration = pulseIn(ECHO_PIN, HIGH);
  return duration ;
  //Serial.println(duration);

}

void loop() {
  float distance = readDistanceCM();
  //Serial.println(distance);

  bool isNearby = distance < 100;
digitalWrite(led, isNearby);

Serial.print("Measured distance: ");
Serial.println(distance);
  if(distance<100){
    PublishData2(distance);

}else{
    PublishData1(distance);

  }
  //PublishData(distance);
```

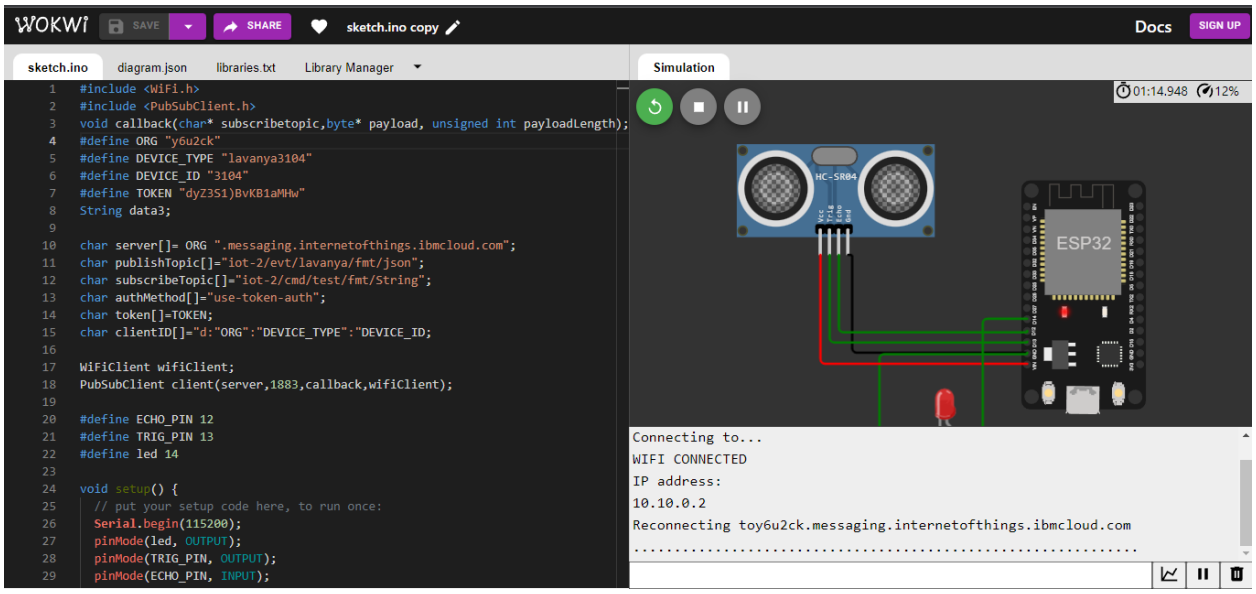
```
delay(1000);
  if(!client.loop()){
mqttconnect();
  }

  //delay(2000);
}
void PublishData1(float dist){
mqttconnect();
  String payload= "{\"distance\":";
  payload += dist;
  payload+="}";

Serial.print("Sending payload:");
Serial.println(payload);

  if(client.publish(publishTopic,(char*)payload.c_str())){
Serial.println("publish ok");
  } else{
Serial.println("publish failed");
  }
}
void PublishData2(float dist){
mqttconnect();
```

```
  String payload= "{\"ALERT\":";
  payload += dist;
  payload+="}";

Serial.print("Sending payload:");
Serial.println(payload);

  if(client.publish(publishTopic,(char*)payload.c_str())){
Serial.println("publish ok");
  } else{
Serial.println("publish failed");
  }
}
void mqttconnect(){
  if(!client.connected()){
Serial.print("Reconnecting to");
Serial.println(server);
while(!!!client.connect(clientID, authMethod, token)){
Serial.print(".");
delay(500);
    }
initManagedDevice();
Serial.println();
  }
```

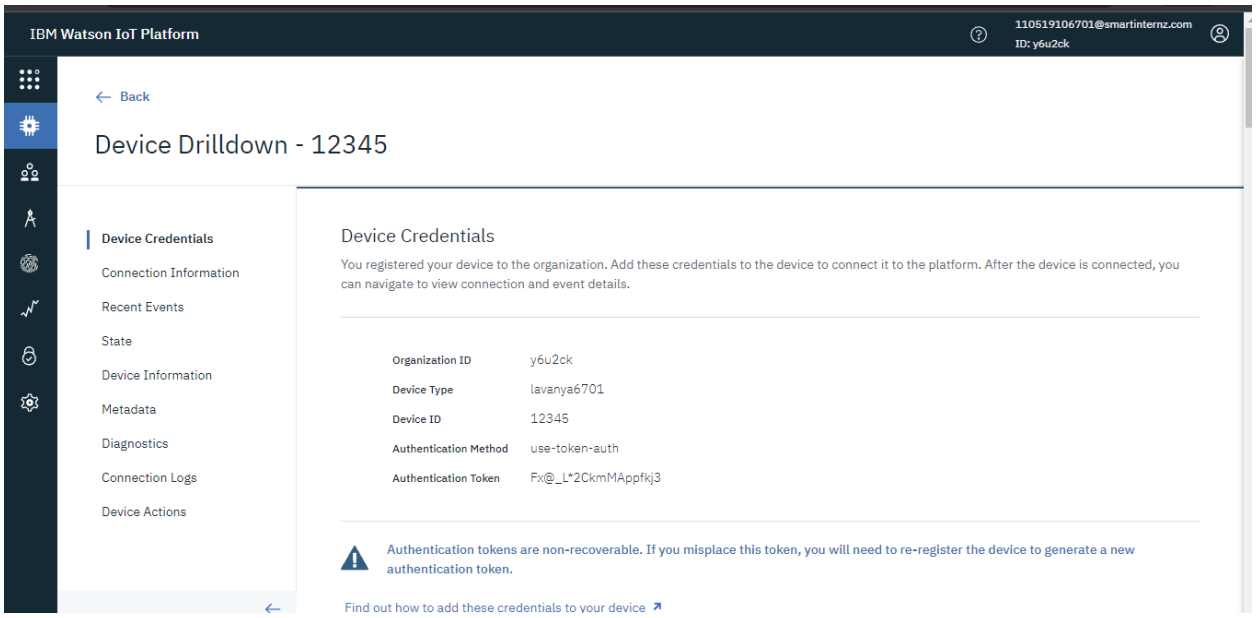
```
}

void wificonnect(){
Serial.println();
Serial.print("Connecting to");

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

void initManagedDevice(){
  if(client.subscribe(subscribeTopic)){
Serial.println((subscribeTopic));
Serial.println("subscribe to cmd ok");
}else{
Serial.println("subscribe to cmd 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++){
    data3 += (char)payload[i];
  }
Serial.println("data:"+ data3);
  if(data3=="lighton"){
Serial.println(data3);
digitalWrite(led,HIGH);
}else{
Serial.println(data3);
digitalWrite(led,LOW);
  }
  data3="";
}
```

IBM CLOUD OUTPUT :




```
#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic,byte* payload, unsigned int payloadLength);
#define ORG "y6u2ck"
#define DEVICE_TYPE "lavanya3104"
#define DEVICE_ID "3104"
#define TOKEN "dyZ3S1)BvKB1aMHw"
String data3;

char server[]= ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[]="iot-2/evt/lavanya/fmt/json";
char subscribeTopic[]="iot-2/cmd/test/fmt/String";
char authMethod[]="use-token-auth";
char token[]=TOKEN;
char clientID[]="d:"ORG":"DEVICE_TYPE":"DEVICE_ID;

WiFiClient wifiClient;
PubSubClient client(server,1883,callback,wifiClient);

#define ECHO_PIN 12
#define TRIG_PIN 13
#define led 14

void setup() {
  // put your setup code here, to run once:
  Serial.begin(115200);
  pinMode(led, OUTPUT);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
```


IBM Watson IoT Platform
110519106701@smartinternz.com
ID: y6u2ck
Back
Device Drilldown - 12345
Device Credentials
Connection Information
Recent Events
State
Device Information
Metadata
Diagnostics
Connection Logs
Device Actions
Device Credentials
You registered your device to the organization. Add these credentials to the device to connect it to the platform. After the device is connected, you can navigate to view connection and event details.
Organization ID y6u2ck
Device Type lavanya6701
Device ID 12345
Authentication Method use-token-auth
Authentication Token Fx@_L*2CkmMAppfkj3
Authentication tokens are non-recoverable. If you misplace this token, you will need to re-register the device to generate a new authentication token.
Find out how to add these credentials to your device