Sprint-1

Team ID	PNT2022TMID44430
Project Name	IOT BASED SMART CROP PROTECTION SYSTEM FOR AGRICULTURE .

Program:

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11
#define LED 2

DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr
of dht connected
```

```
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);
```

```
//-------credentials of IBM Accounts------
```

```
#define ORG "0eut3p"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32_Controller"//Device type mentioned in ibm watson
IOT Platform
#define DEVICE_ID "mohan123"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "37Dlx1Y7xEzNtbDm?W"     //Token
String data3;
float h, t;
```

```
//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server
Name
char publishTopic[] = "iot-2/evt/data/fmt/json";// topic name and type of
event perform and format in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT
command type AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
```

```
//-----------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the
predefined client id by passing parameter like server id,portand
wificredential
```

```
void setup()// configureing the ESP32
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
  Serial.println();
  wificonnect();
  mqttconnect();
}
```

```
void loop()// Recursive Function
{
```

```
  h = dht.readHumidity();
  t = dht.readTemperature();
  Serial.print("temp:");
  Serial.println(t);
  Serial.print("Humid:");
  Serial.println(h);
```

```
  PublishData(t, h);
  delay(1000);
  if (!client.loop()) {
    mqttconnect();
  }
}
```

```
/*.....................................retrieving to
Cloud..............................*/
```

```
void PublishData(float temp, float humid) {
  mqttconnect();//function call for connecting to ibm
  /*
     creating the String in in form JSon to update the data to ibm cloud
  */
  String payload = "{\"temp\":";
  payload += temp;
  payload += "," "\"Humid\":";
  payload += humid;
  payload += "}";
  Serial.print("Sending payload: ");
  Serial.println(payload);
```

```
  if (client.publish(publishTopic, (char*) payload.c_str())) {
    Serial.println("Publish ok");// if it sucessfully upload data on the cloud then it will print publish ok in Serial monitor or else it will print publish failed
  } else {
    Serial.println("Publish failed");
  }

}
```

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

     initManagedDevice();
     Serial.println();
  }
}
void wificonnect() //function defination for wificonnect
{
  Serial.println();
  Serial.print("Connecting to ");
```

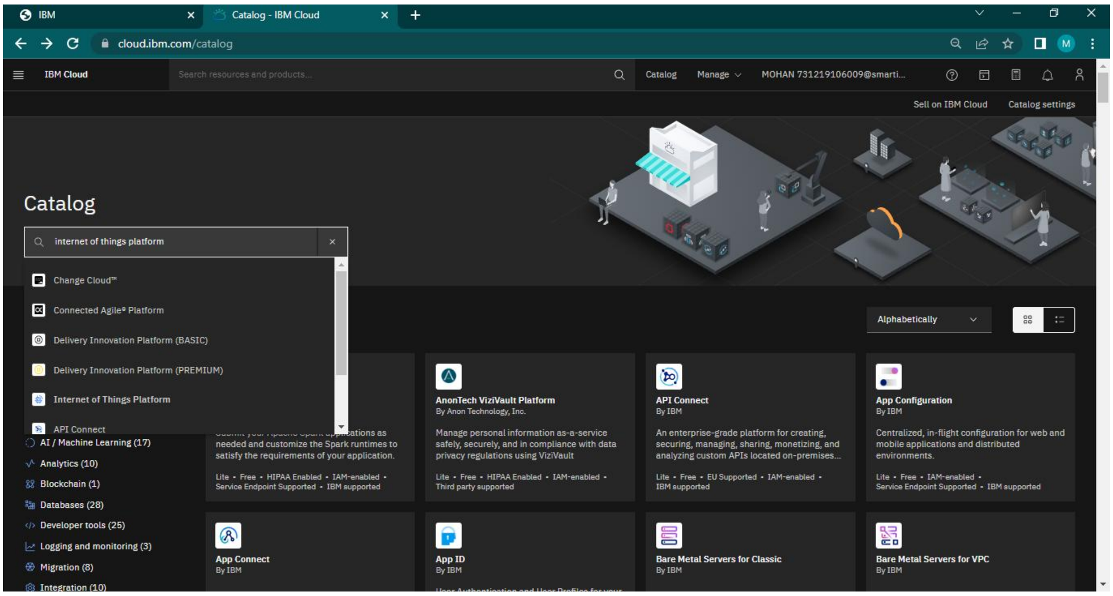
```
  WiFi.begin("Wokwi-GUEST", "", 6);//passing the wifi credentials to establish the connection
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
```

```
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++) {
    //Serial.print((char)payload[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="";
}
```

Reference:

https://wokwi.com/projects/348556069652398676



Catalog

internet of things platform

- Change Cloud™
- Connected Agile® Platform
- Delivery Innovation Platform (BASIC)
- Delivery Innovation Platform (PREMIUM)
- Internet of Things Platform
- API Connect

- AI / Machine Learning (17)
- Analytics (10)
- Blockchain (1)
- Databases (28)
- Developer tools (25)
- Logging and monitoring (3)
- Migration (8)
- Integration (10)

...ations as needed and customize the Spark runtimes to satisfy the requirements of your application.
Lite • Free • HIPAA Enabled • IAM-enabled • Service Endpoint Supported • IBM supported

AnonTech ViziVault Platform
By Anon Technology, Inc.
Manage personal information as-a-service safely, securely, and in compliance with data privacy regulations using ViziVault
Lite • Free • HIPAA Enabled • IAM-enabled • Third party supported

API Connect
By IBM
An enterprise-grade platform for creating, securing, managing, sharing, monetizing, and analyzing custom APIs located on-premises...
Lite • Free • EU Supported • IAM-enabled • IBM supported

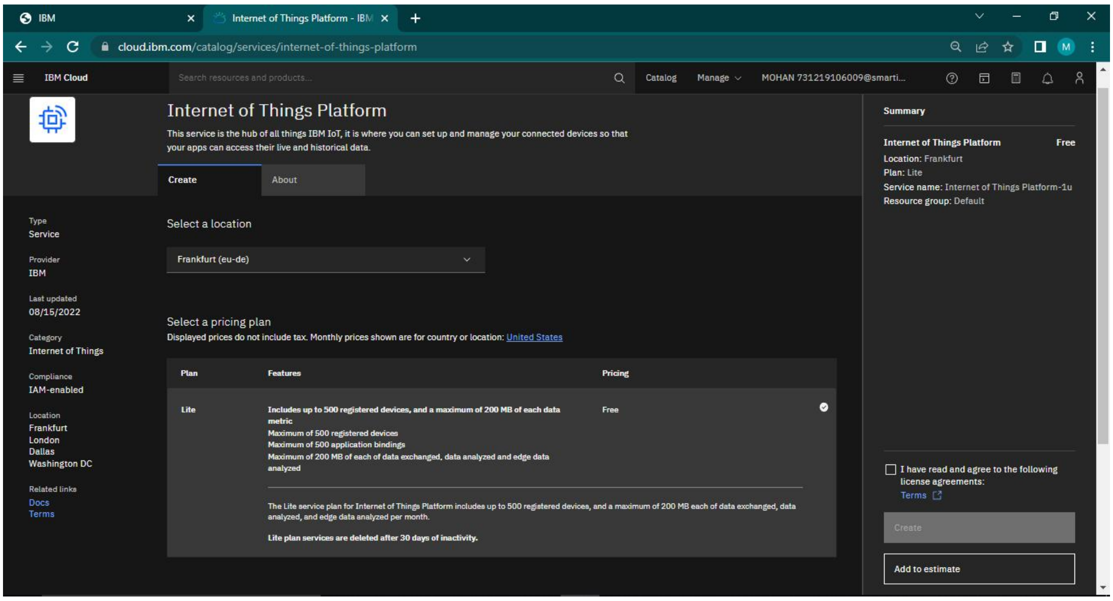
App Configuration
By IBM
Centralized, in-flight configuration for web and mobile applications and distributed environments.
Lite • Free • IAM-enabled • Service Endpoint Supported • IBM supported

App Connect
By IBM

App ID
By IBM

Bare Metal Servers for Classic
By IBM

Bare Metal Servers for VPC
By IBM


IBM
Internet of Things Platform - IBM
cloud.ibm.com/catalog/services/internet-of-things-platform
IBM Cloud
Search resources and products...
Catalog
Manage
MOHAN 731219106009@smarti...
Create
About

Internet of Things Platform

This service is the hub of all things IBM IoT, it is where you can set up and manage your connected devices so that your apps can access their live and historical data.

Type
Service

Provider
IBM

Last updated
08/15/2022

Category
Internet of Things

Compliance
IAM-enabled

Location
Frankfurt
London
Dallas
Washington DC

Related links
Docs
Terms

Select a location

Frankfurt (eu-de)

Select a pricing plan
Displayed prices do not include tax. Monthly prices shown are for country or location: United States

Plan	Features	Pricing
Lite	**Includes up to 500 registered devices, and a maximum of 200 MB of each data metric** **Maximum of 500 registered devices** **Maximum of 500 application bindings** **Maximum of 200 MB of each of data exchanged, data analyzed and edge data analyzed** The Lite service plan for Internet of Things Platform includes up to 500 registered devices, and a maximum of 200 MB each of data exchanged, data analyzed, and edge data analyzed per month. **Lite plan services are deleted after 30 days of inactivity.**	Free

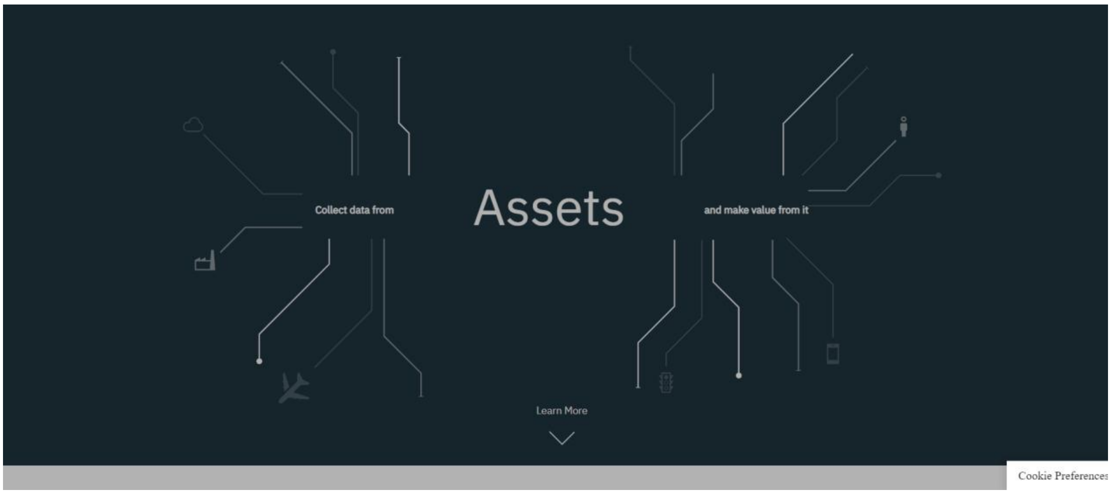
Summary

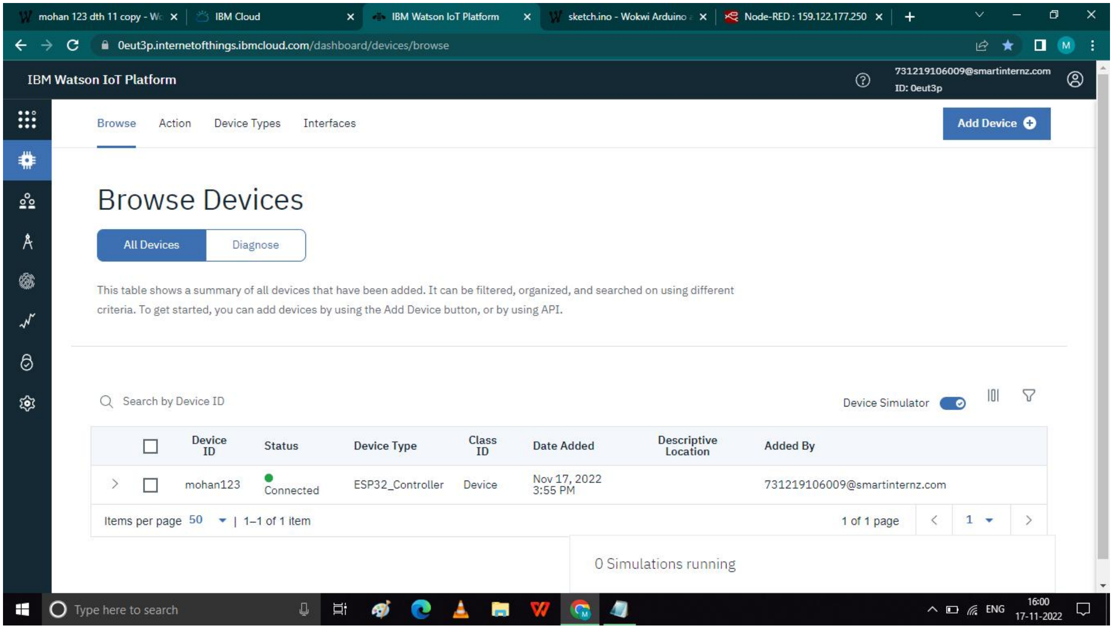
Internet of Things Platform **Free**
Location: Frankfurt
Plan: Lite
Service name: Internet of Things Platform-1u
Resource group: Default

□ I have read and agree to the following license agreements:
Terms

Create

Add to estimate




mohan 123 dth 11 copy - Wo
IBM Cloud
IBM Watson IoT Platform
sketch.ino - Wokwi Arduino
Node-RED : 159.122.177.250
0eut3p.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
731219106009@smartinternz.com
ID: 0eut3p
Browse
Action
Device Types
Interfaces
Add Device
Type here to search
ENG
16:00
17-11-2022

Browse Devices

All Devices | Diagnose

This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

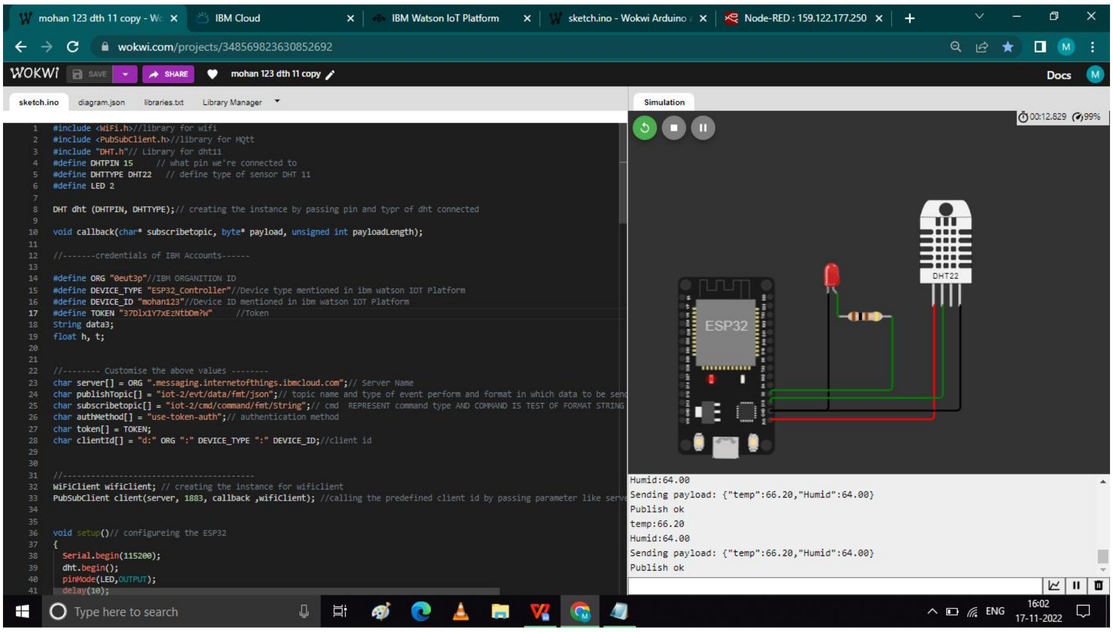
Search by Device ID

Device Simulator

□	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By
□	mohan123	● Connected	ESP32_Controller	Device	Nov 17, 2022 3:55 PM		731219106009@smartinternz.com

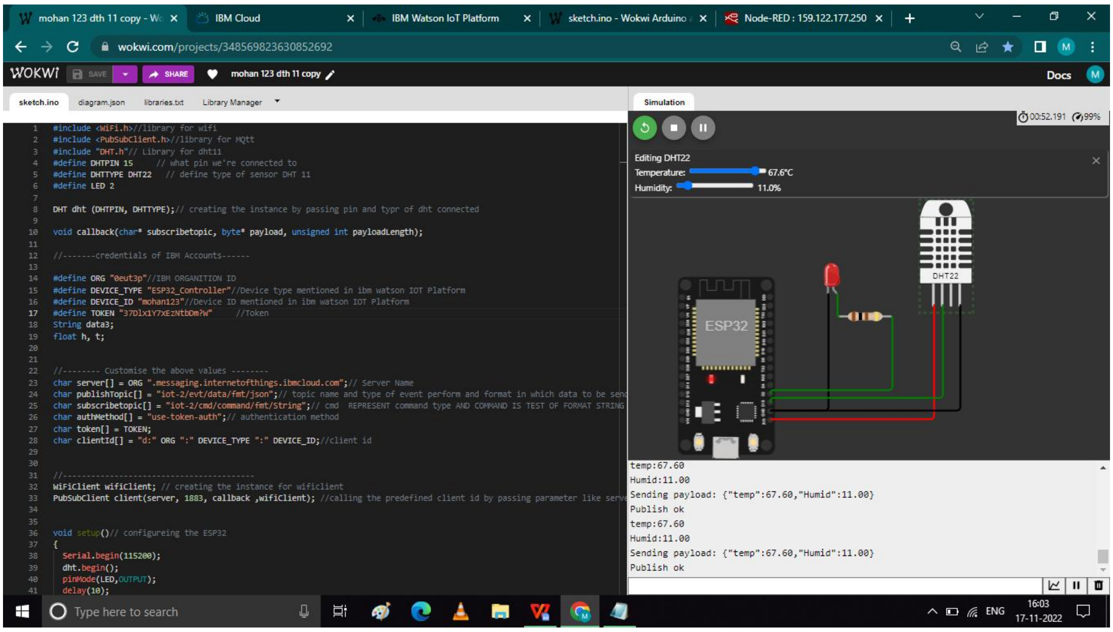
Items per page 50 | 1–1 of 1 item   1 of 1 page   1

0 Simulations running



```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11
#define LED 2

DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//-------credentials of IBM Accounts------

#define ORG "0eut3p"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32_Controller"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "mohan123"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "37Dlx1Y7xEzNtbDm?W"     //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/data/fmt/json";// topic name and type of event perform and format in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


//-----------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passing parameter like serve


void setup()// configureing the ESP32
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
```

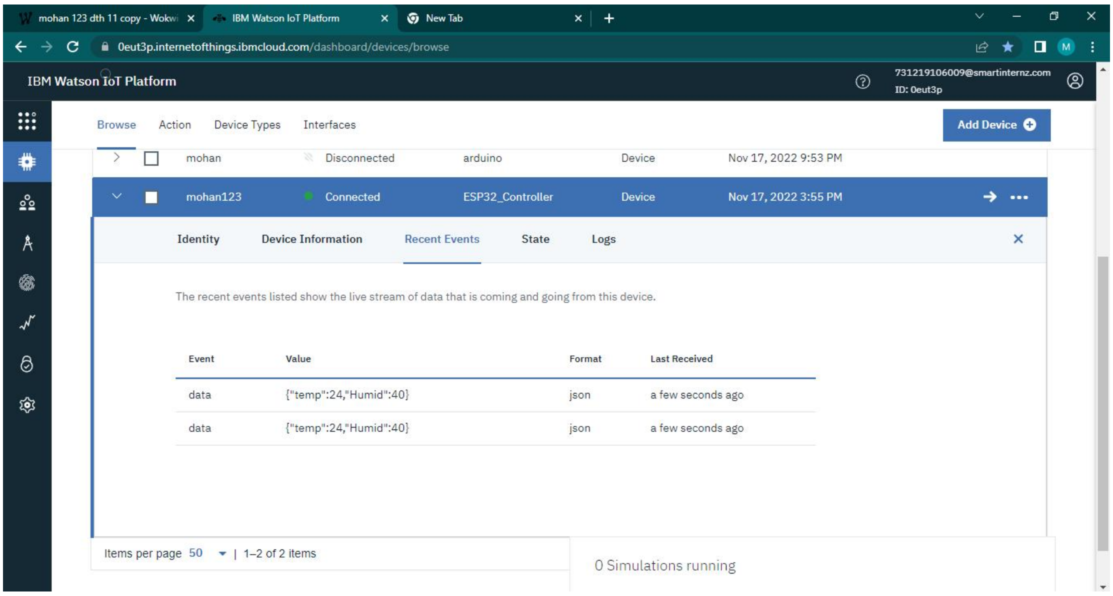
```
Humid:64.00
Sending payload: {"temp":66.20,"Humid":64.00}
Publish ok
temp:66.20
Humid:64.00
Sending payload: {"temp":66.20,"Humid":64.00}
Publish ok
```



Editing DHT22
Temperature: 67.6°C
Humidity: 11.0%

```
#include <WiFi.h>//library for wifi
#include <PubSubClient.h>//library for MQtt
#include "DHT.h"// Library for dht11
#define DHTPIN 15     // what pin we're connected to
#define DHTTYPE DHT22   // define type of sensor DHT 11
#define LED 2

DHT dht (DHTPIN, DHTTYPE);// creating the instance by passing pin and typr of dht connected

void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//-------credentials of IBM Accounts------

#define ORG "0eut3p"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32_Controller"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "mohan123"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "37Dlx1Y7xEzNtbDm?W"     //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/data/fmt/json";// topic name and type of event perform and format in which data to be send
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command type AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id


//-----------------------------------------
WiFiClient wifiClient; // creating the instance for wificlient
PubSubClient client(server, 1883, callback ,wifiClient); //calling the predefined client id by passing parameter like serve


void setup()// configureing the ESP32
{
  Serial.begin(115200);
  dht.begin();
  pinMode(LED,OUTPUT);
  delay(10);
```

```
temp:67.60
Humid:11.00
Sending payload: {"temp":67.60,"Humid":11.00}
Publish ok
temp:67.60
Humid:11.00
Sending payload: {"temp":67.60,"Humid":11.00}
Publish ok
```


mohan 123 dth 11 copy - Wokwi
IBM Watson IoT Platform
New Tab
0eut3p.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
731219106009@smartinternz.com
ID: 0eut3p
Browse
Action
Device Types
Interfaces
Add Device

mohan	Disconnected	arduino	Device	Nov 17, 2022 9:53 PM
mohan123	Connected	ESP32_Controller	Device	Nov 17, 2022 3:55 PM

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	{"temp":24,"Humid":40}	json	a few seconds ago
data	{"temp":24,"Humid":40}	json	a few seconds ago

Items per page 50 | 1–2 of 2 items

0 Simulations running