Develop the python script

Publish data to the IBM Cloud

Date	09 November 2022
Team id	PNT2022TMID51225
Project name	Signs with smart connectivity for better road safety

To make a publisher and subscriber in the process of python and IBM cloud:

publish.py - C:\Users\Sri\AppData\Local\Programs\Python\Python39\publish.py (3.9.8)

File Edit Format Run Options Window Help

```
import paho.mqrtt.client as paho
import time
import random
def on_publish(client,usrdata,mid):
    print("Publish the data")
client=paho.Client()
client.connect('broker.Mqttdashboard.com',1883)
client.loop_start()
while True:
    temp=random.randint(1,30)
    (rc,mid)=client.publish('iottopic',str(temp),qos=1)
    print(temp)
    time.sleep(10)
```

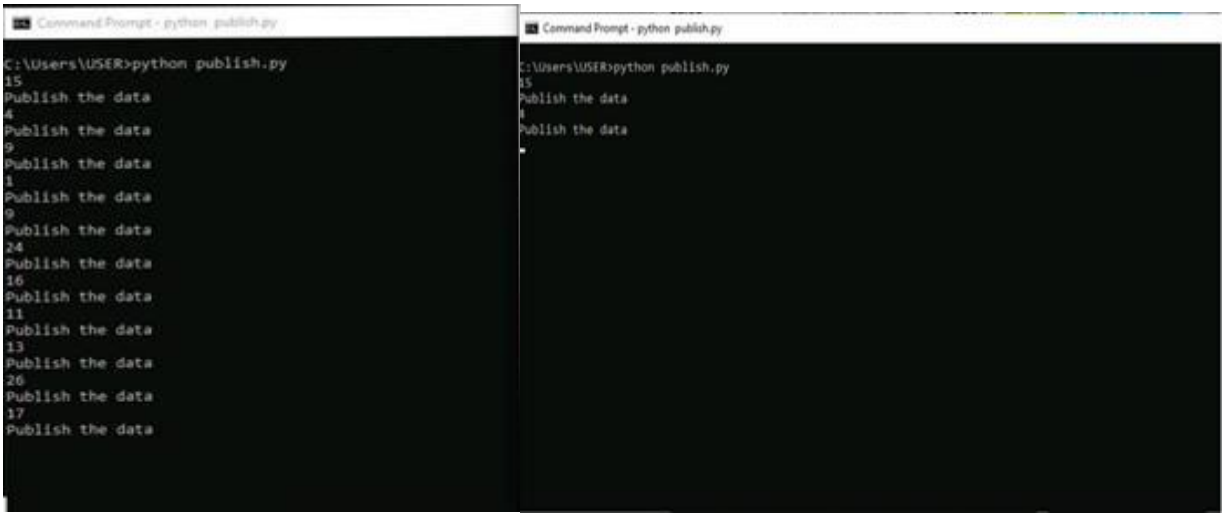
Ln: 1 Col: 0

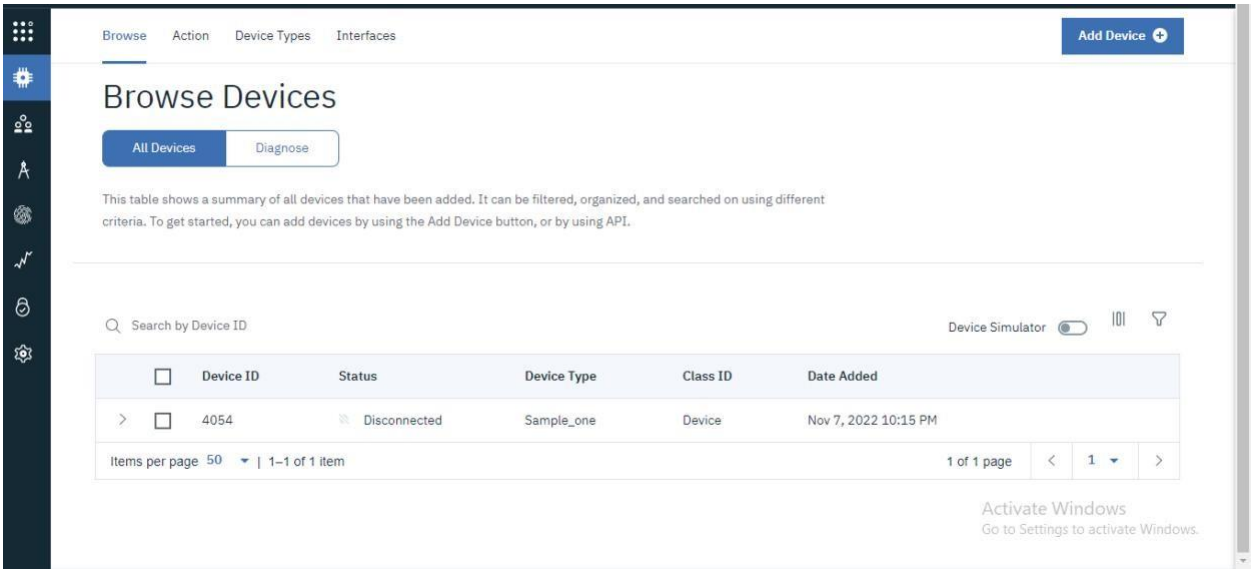
subscribe.py - C:/Users/Sri/AppData/Local/Programs/Python/Python39/subscribe.py (3.9.8)

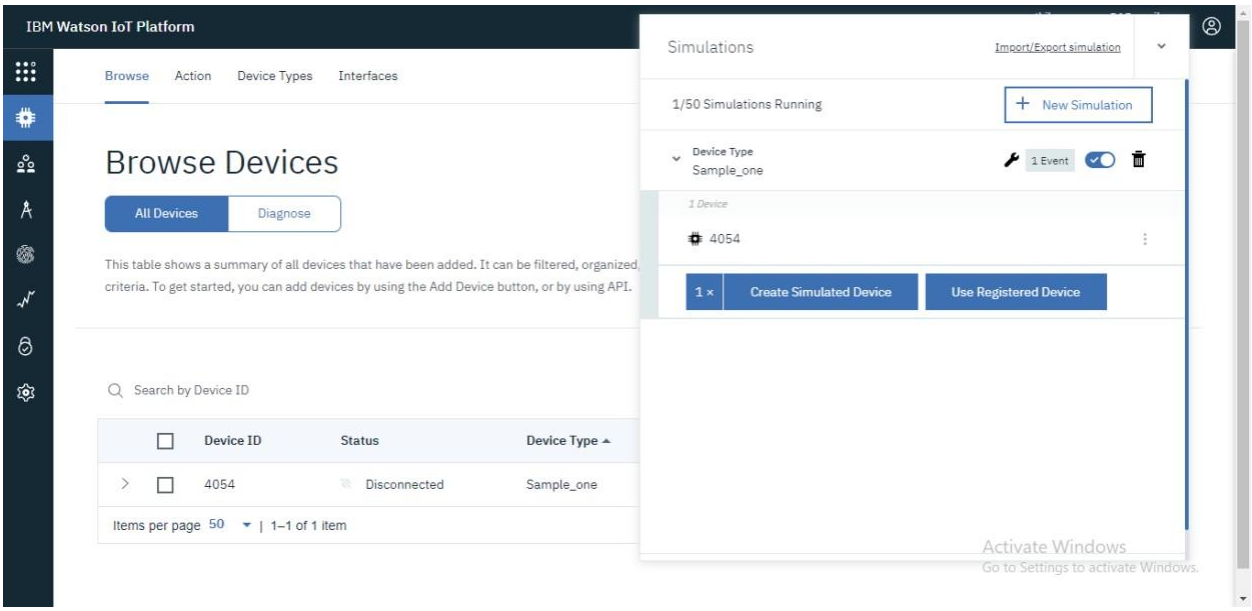
File Edit Format Run Options Window Help

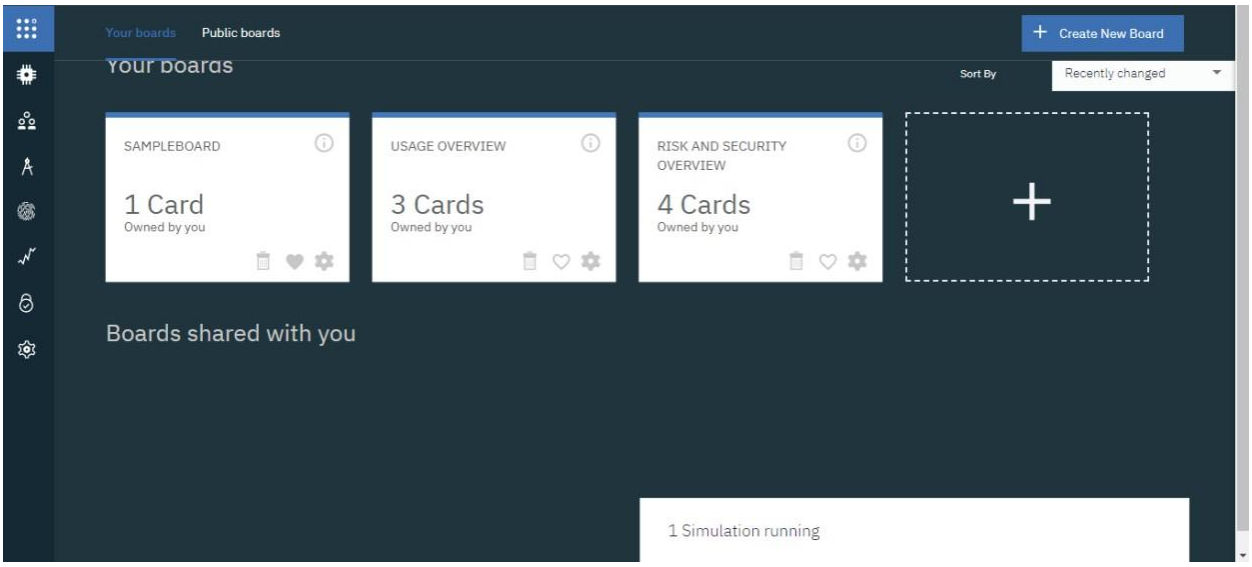
```
import paho.mqrtt.client as paho
def on_subscribe(client,usrdata,mid,grated_qos):
    print("subscribe:",+str(mid)+str(granted_qos))
def on_message(client,usrdata,msg):
    print(msg.topic+" "+str(msg.qos)+" "+str(msg.payload))

client= paho.Client()
client.on_subsribe = on_subscribe
client.on_message = on_message
client.connect('broker.mqrttdashboard.com',1883)
client.subscribe('iot',qos=1)
client.loo_forever()
```

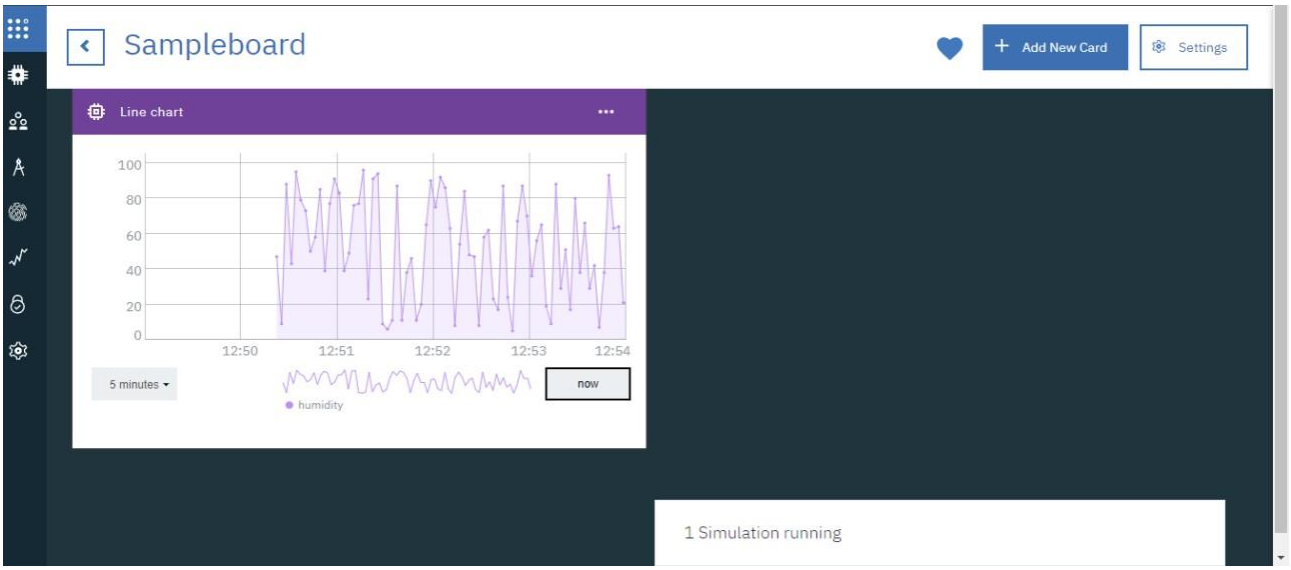
Ln: 1 Col: 0


Command Prompt - python publish.py
C:\Users\USER>python publish.py
15
Publish the data
4
Publish the data
9
Publish the data
1
Publish the data
9
Publish the data
24
Publish the data
16
Publish the data
11
Publish the data
13
Publish the data
26
Publish the data
17
Publish the data
Command Prompt - python publish.py
C:\Users\USER>python publish.py
15
Publish the data
4
Publish the data


Browse
Action
Device Types
Interfaces
Add Device
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
4054
Disconnected
Sample_one
Device
Nov 7, 2022 10:15 PM
Items per page 50
1–1 of 1 item
1 of 1 page
1
Activate Windows
Go to Settings to activate Windows.


IBM Watson IoT Platform
Browse
Action
Device Types
Interfaces
Browse Devices
All Devices
Diagnose
This table shows a summary of all devices that have been added. It can be filtered, organized,
criteria. To get started, you can add devices by using the Add Device button, or by using API.
Search by Device ID
Device ID
Status
Device Type
4054
Disconnected
Sample_one
Items per page 50
1–1 of 1 item
Simulations
Import/Export simulation
1/50 Simulations Running
New Simulation
Device Type
Sample_one
1 Event
1 Device
4054
1 ×
Create Simulated Device
Use Registered Device
Activate Windows
Go to Settings to activate Windows.


Your boards
Public boards
Create New Board
Your boards
Sort By
Recently changed
SAMPLEBOARD
1 Card
Owned by you
USAGE OVERVIEW
3 Cards
Owned by you
RISK AND SECURITY OVERVIEW
4 Cards
Owned by you
Boards shared with you
1 Simulation running



CODING:

```
#IBM WatsonIoT Platform

import wiotp.sdk.device

import time

import random

myConfig = {

"identity": {

"orgId": "hj5fmy",

"typeId": "NodeMCU",

"deviceId":"12345"

},

"auth": {

"token": "12345678"

}

}


def myCommandCallback(cmd):
```

```
print("Message received from IBM IoT Platform: %s" % cmd.data['command'])

m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)

client.connect()

while True:

temp=random.randint(-20,125)

hum=random.randint(0,100)

myData={'temperature':temp, 'humidity':hum}

client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)

print("Published data Successfully: %s", myData)

client.commandCallback = myCommandCallback

time.sleep(2)

client.disconnect()
```