DEVELOP THE PYTHON SCRIPT

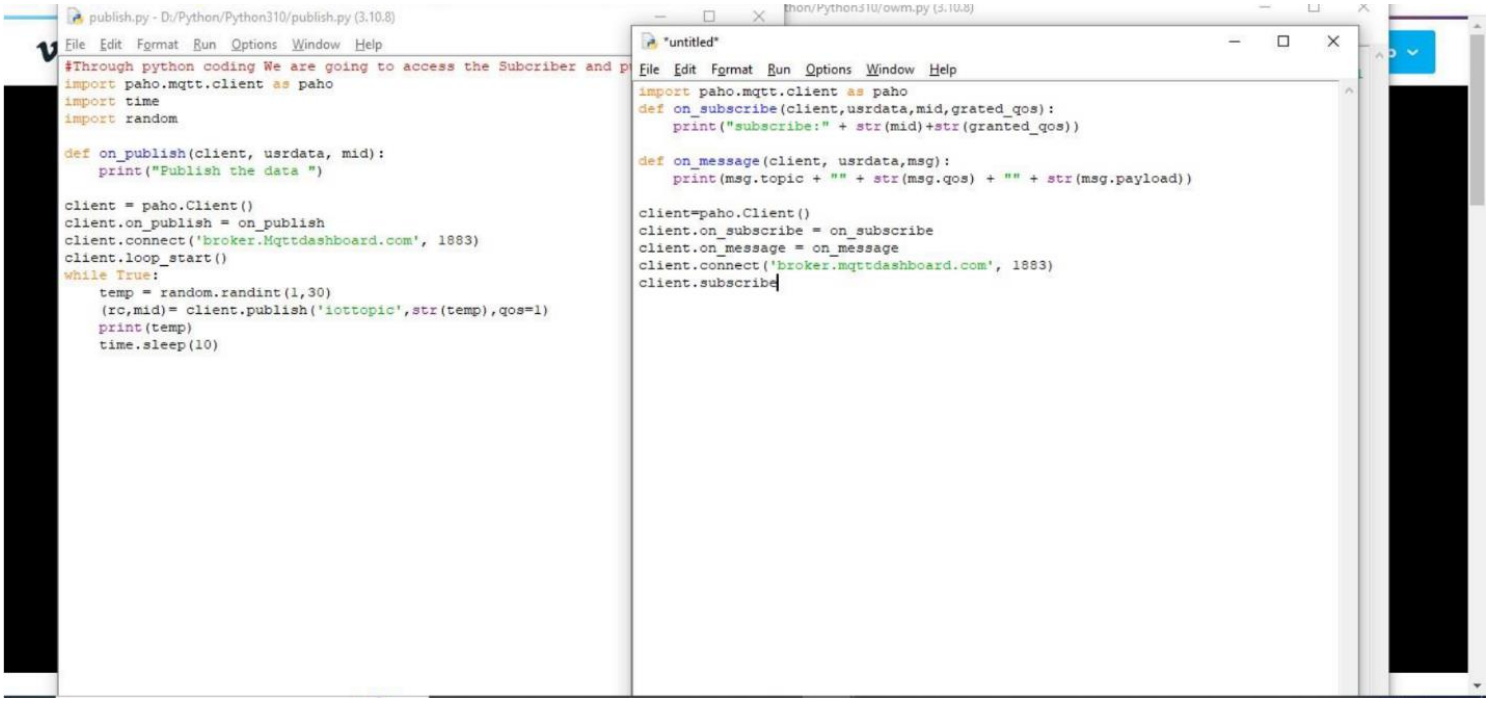
Publish data to the IBM Cloud

Team ID : PNT2022TMID11625

Project Name - SIGNS WITH SMART CONNECTIVITY FOR BETTER ROAD SAFETY

To make a Publisher and Subscriber in the process of Python and IBM cloud

1.



publish.py - D:/Python/Python310/publish.py (3.10.8)

File Edit Format Run Options Window Help

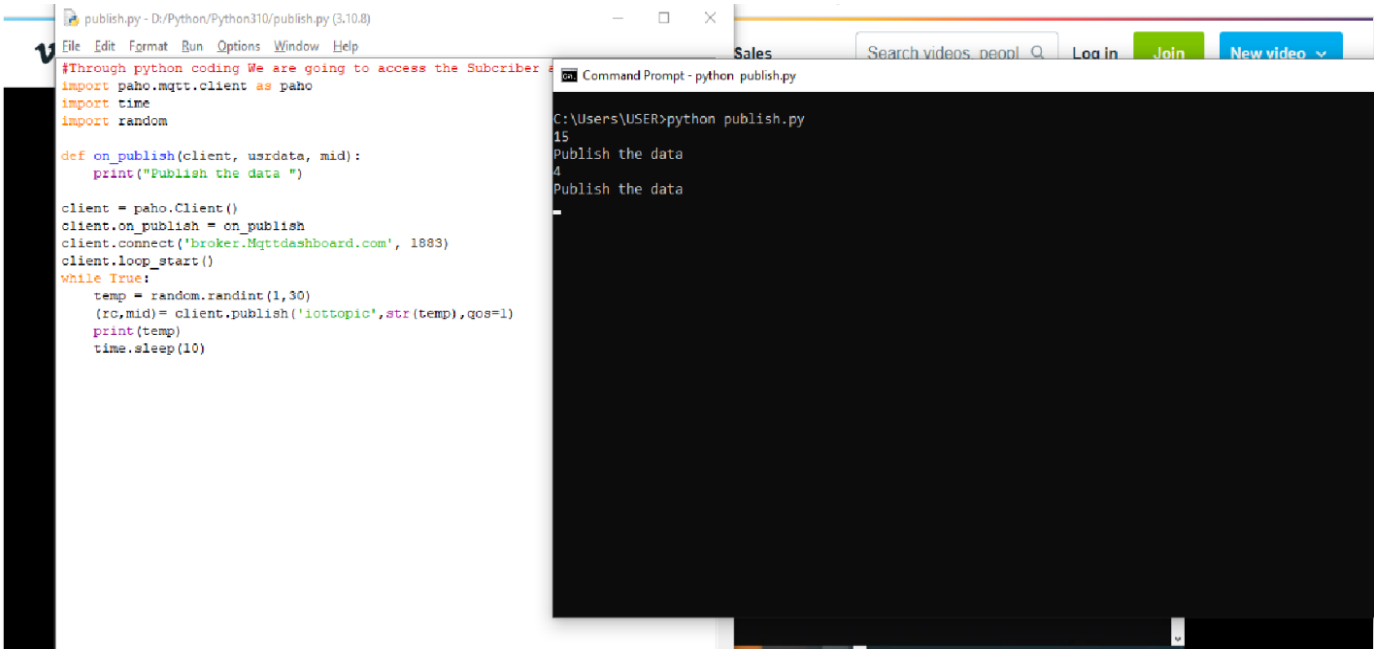
```
#Through python coding We are going to access the Subcriber and p
import paho.mqtt.client as paho
import time
import random

def on_publish(client, usrdata, mid):
    print("Publish the data ")

client = paho.Client()
client.on_publish = on_publish
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)
```

untitled

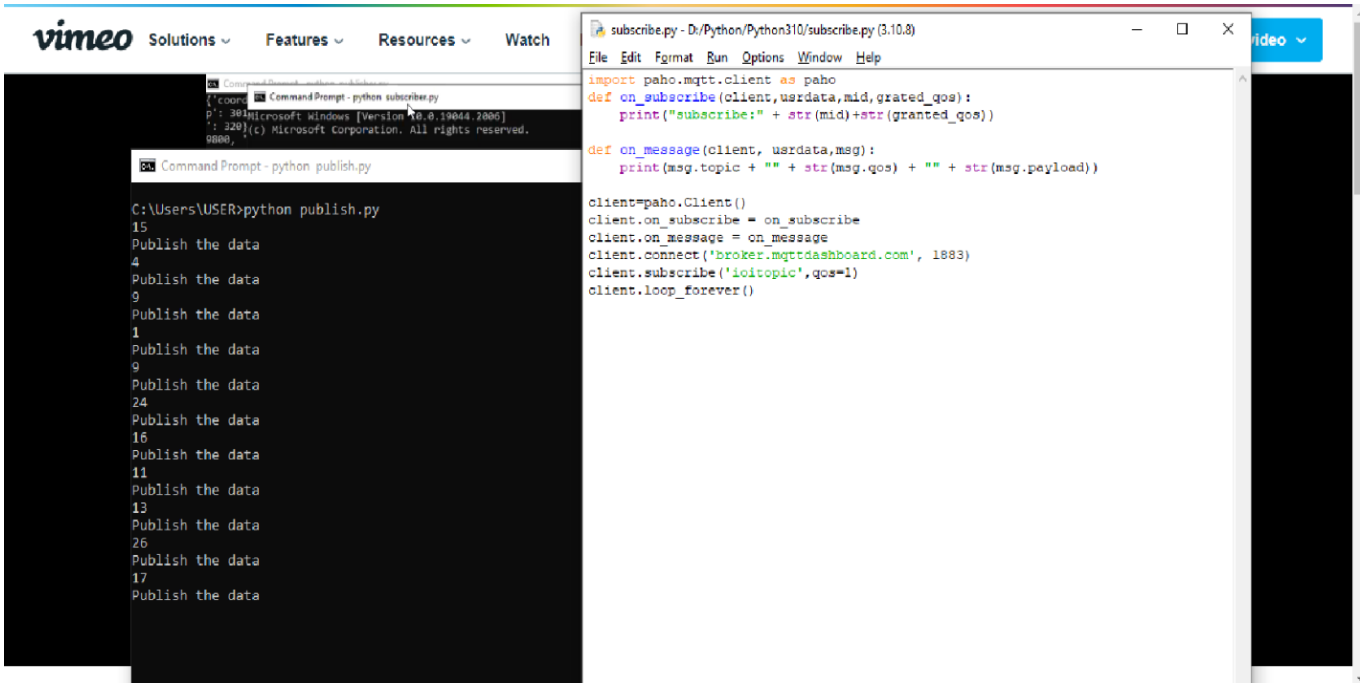
File Edit Format Run Options Window Help

```
import paho.mqtt.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_subscribe = on_subscribe
client.on_message = on_message
client.connect('broker.mqttdashboard.com', 1883)
client.subscribe
```

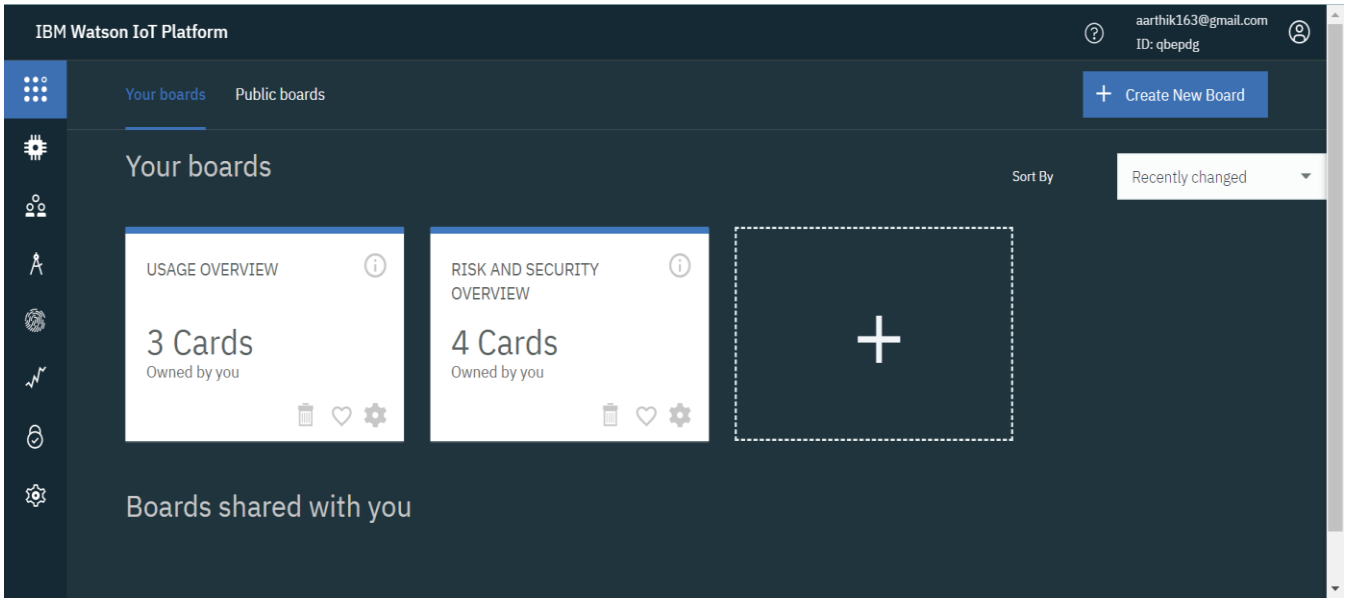
2.


publish.py - D:/Python/Python310/publish.py (3.10.8)
File Edit Format Run Options Window Help
#Through python coding We are going to access the Subcriber
import paho.mqtt.client as paho
import time
import random
def on_publish(client, usrdata, mid):
print("Publish the data ")
client = paho.Client()
client.on_publish = on_publish
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)
Command Prompt - python publish.py
C:\Users\USER>python publish.py
15
Publish the data
4
Publish the data

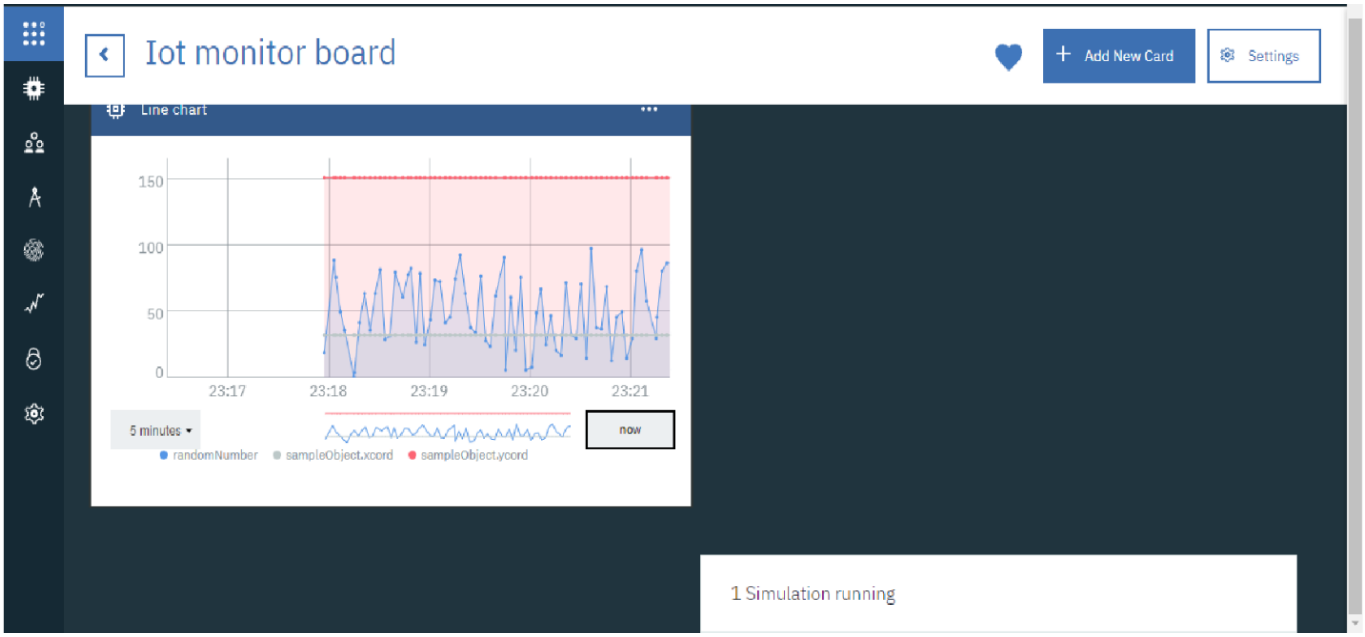
3.


vimeo
Solutions
Features
Resources
Watch
Command Prompt - python subscriber.py
Microsoft Windows [Version 10.0.19044.2006]
(c) Microsoft Corporation. All rights reserved.
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
subscribe.py - D:/Python/Python310/subscribe.py (3.10.8)
File Edit Format Run Options Window Help
import paho.mqtt.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_subscribe = on_subscribe
client.on_message = on_message
client.connect('broker.mqttdashboard.com', 1883)
client.subscribe('iottopic',qos=1)
client.loop_forever()

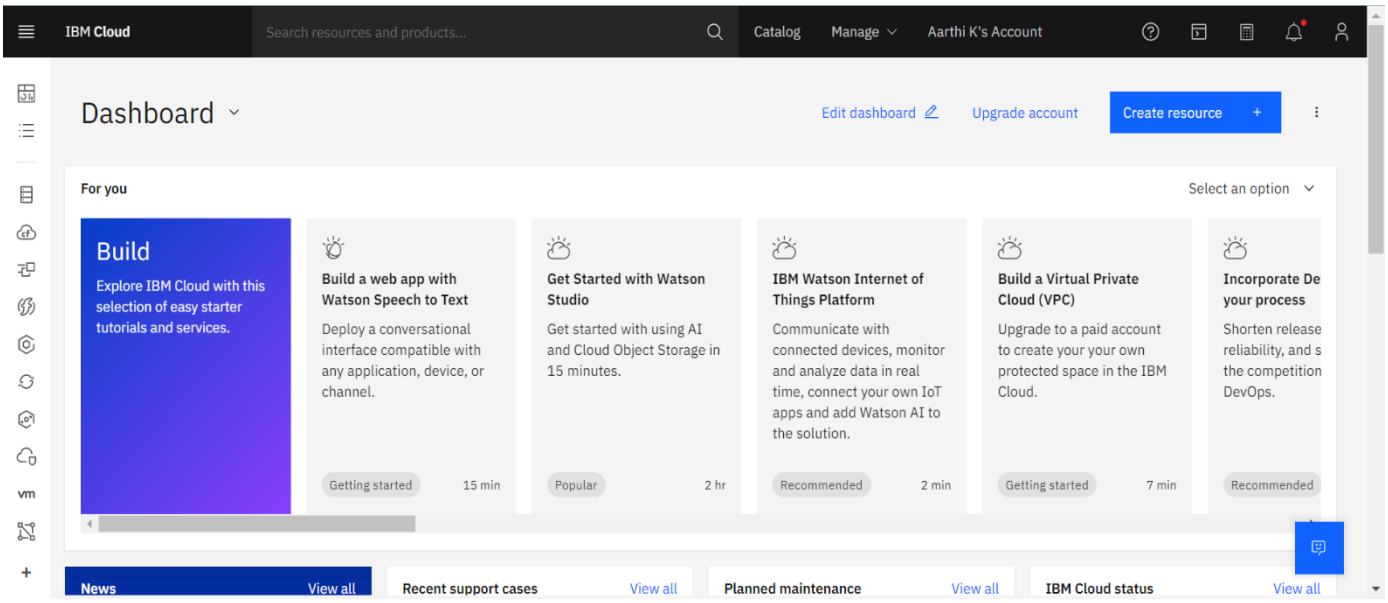
4.


IBM Watson IoT Platform
aarthik163@gmail.com
ID: qbepdg
Your boards
Public boards
Create New Board
Your boards
Sort By
Recently changed
USAGE OVERVIEW
3 Cards
Owned by you
RISK AND SECURITY OVERVIEW
4 Cards
Owned by you
Boards shared with you

5.


Iot monitor board
Add New Card
Settings
Line chart
150
100
50
0
23:17
23:18
23:19
23:20
23:21
5 minutes
now
randomNumber
sampleObject.xcord
sampleObject.ycord
1 Simulation running

6.



PROGRAM

#IBM

Watson IOT

Platform

#pip install wiotp-sdk

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()
```