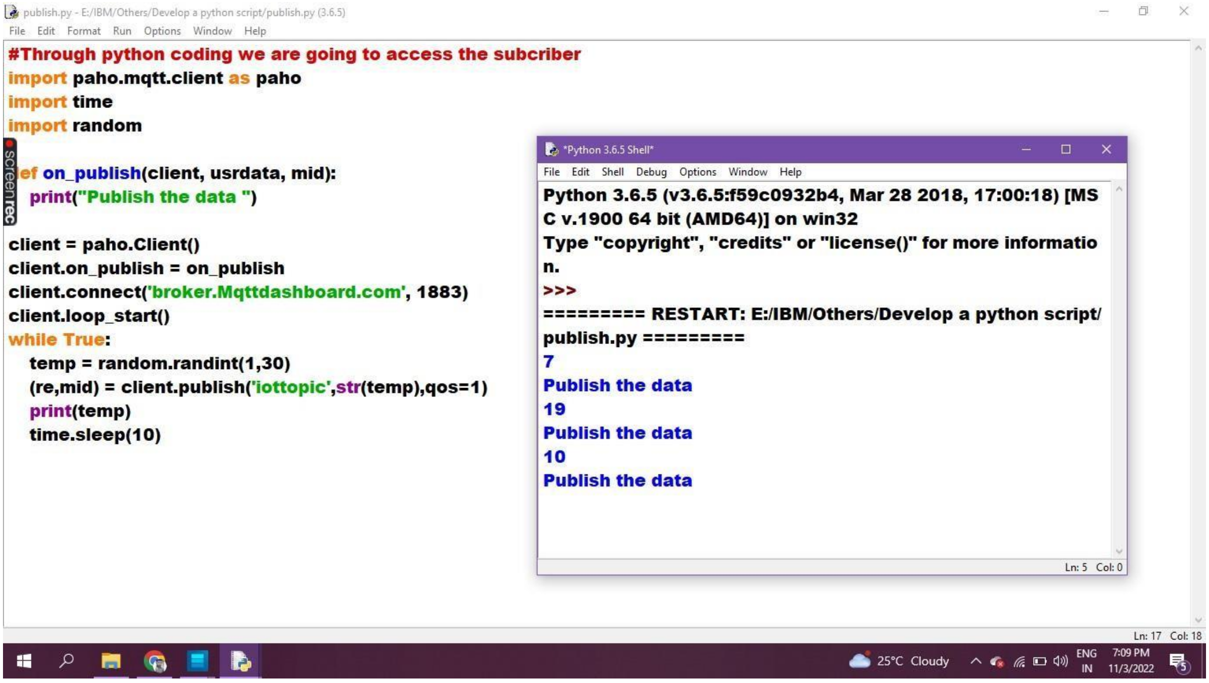
Develop a python script
Publish Data to the IBM Cloud

Date	13 November 2022
Team ID	PNT2022TMID07843
Project Name	Project - Signs with smart connectivity for Better road safety
Maximum Marks	4 Marks

Signs with smart connectivity for Better road safety

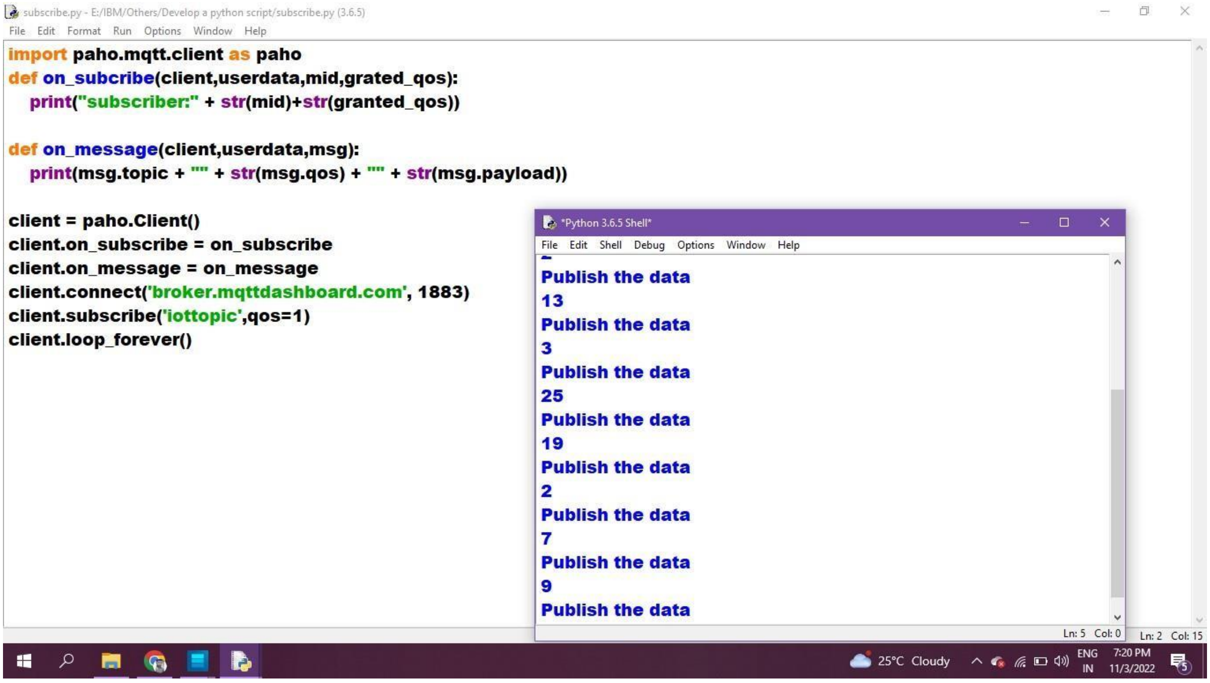

publish.py - E:/IBM/Others/Develop a python script/publish.py (3.6.5)
File Edit Format Run Options Window Help
screenrec
Python 3.6.5 Shell
File Edit Shell Debug Options Window Help
Ln: 5 Col: 0
Ln: 17 Col: 18
25°C Cloudy
ENG IN
7:09 PM 11/3/2022

```
#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)
   (re,mid) = client.publish('iottopic',str(temp),qos=1)
   print(temp)
   time.sleep(10)
```

```
Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 17:00:18) [MS
C v.1900 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more informatio
n.
>>>
========= RESTART: E:/IBM/Others/Develop a python script/
publish.py =========
7
Publish the data
19
Publish the data
10
Publish the data
```

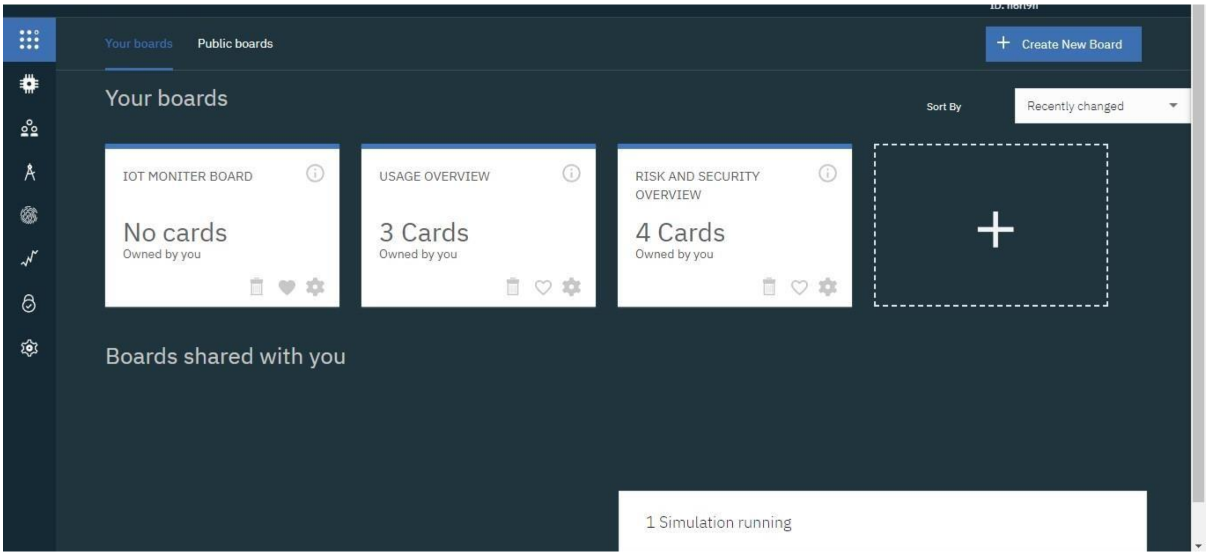

subscribe.py - E:/IBM/Others/Develop a python script/subscribe.py (3.6.5)
File Edit Format Run Options Window Help
Python 3.6.5 Shell
File Edit Shell Debug Options Window Help
Ln: 5 Col: 0
Ln: 2 Col: 15
25°C Cloudy
ENG IN
7:20 PM 11/3/2022

```
import paho.mqtt.client as paho
def on_subcribe(client,userdata,mid,grated_qos):
   print("subscriber:" + str(mid)+str(granted_qos))

def on_message(client,userdata,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()
```

```
Publish the data
13
Publish the data
3
Publish the data
25
Publish the data
19
Publish the data
2
Publish the data
7
Publish the data
9
Publish the data
```



Your boards

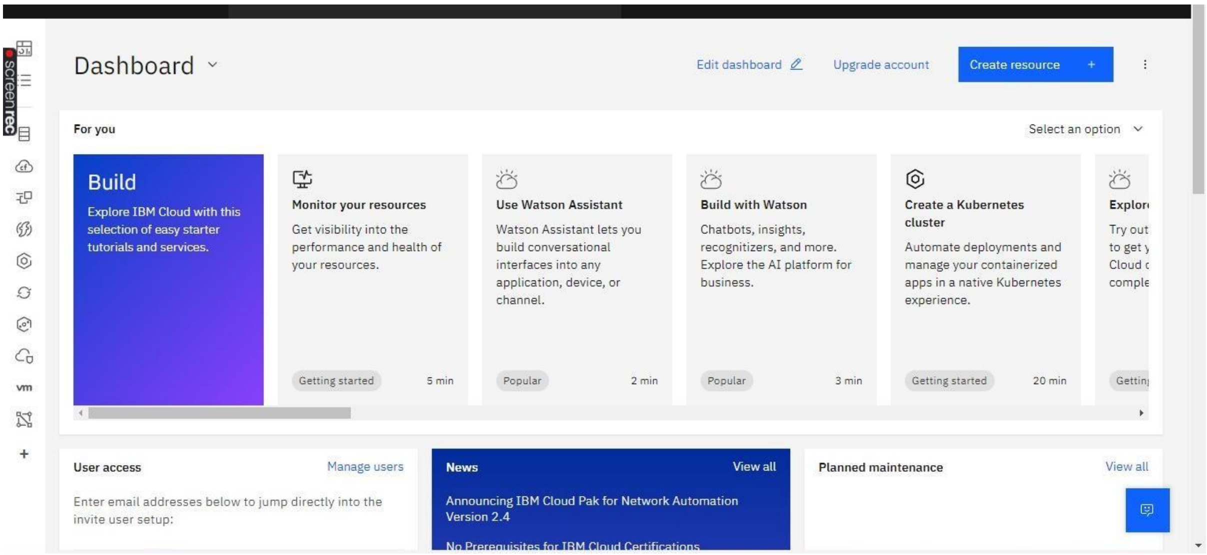
IOT MONITER BOARD
No cards
Owned by you

USAGE OVERVIEW
3 Cards
Owned by you

RISK AND SECURITY OVERVIEW
4 Cards
Owned by you

Boards shared with you

1 Simulation running



Build

Explore IBM Cloud with this selection of easy starter tutorials and services.

Monitor your resources

Get visibility into the performance and health of your resources.

Use Watson Assistant

Watson Assistant lets you build conversational interfaces into any application, device, or channel.

Build with Watson

Chatbots, insights, recognitizers, and more. Explore the AI platform for business.

Create a Kubernetes cluster

Automate deployments and manage your containerized apps in a native Kubernetes experience.

Explor

Try out to get y Cloud c comple

User access

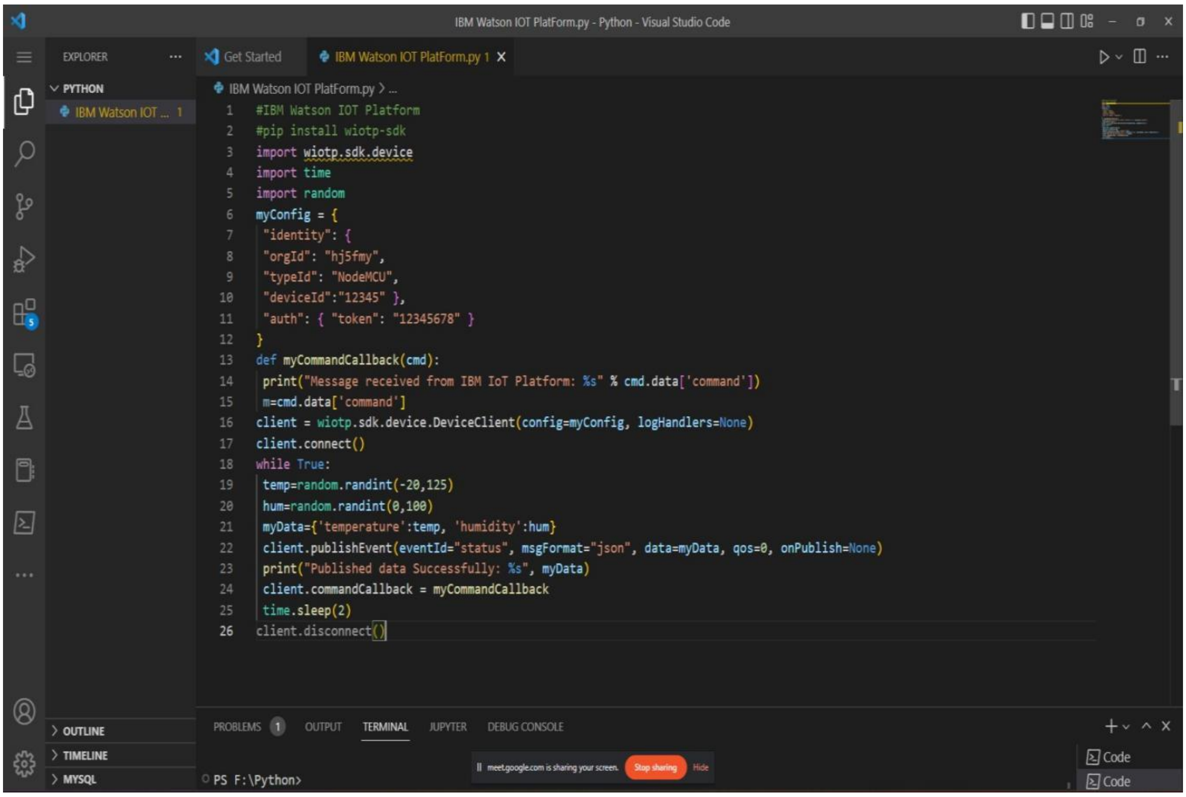
Enter email addresses below to jump directly into the invite user setup:

News

Announcing IBM Cloud Pak for Network Automation Version 2.4

No Prerequisites for IBM Cloud Certifications

Planned maintenance


IBM Watson IOT PlatForm.py - Python - Visual Studio Code
EXPLORER
PYTHON
IBM Watson IOT ... 1
Get Started
IBM Watson IOT PlatForm.py 1
IBM Watson IOT PlatForm.py > ...
OUTLINE
TIMELINE
MYSQL
PROBLEMS 1
OUTPUT
TERMINAL
JUPYTER
DEBUG CONSOLE
meet.google.com is sharing your screen.
Stop sharing
Hide
Code
Code

```
#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()
```

```
PS F:\Python>
```

CODE:

```
#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=nyData, qos=0, onPublish=None)
print("Published data Successfully: %s", myData)

client.commandCallback = myCommandCallback

time.sleep(2)

client.disconnect
```