

DEVELOP THE PYTHON SCRIPT

Publish data to the IBM Cloud

Team ID : PNT2022TMID17040

Project Title : 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:/Python/Python310/publish.py (3.10.8)
File Edit Format Run Options Window Help
#Through python coding We are going to access the Subscriber and p
import paho.mqtt.client as paho
import time
import random

def on_publish(client, userdata, 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,userdata,mid,grated_qos):
    print("subscribe:" + str(mid)+str(granted_qos))

def on_message(client, userdata,msg):
    print(msg.topic + "" + str(msg.qos) + "" + str(msg.payload))

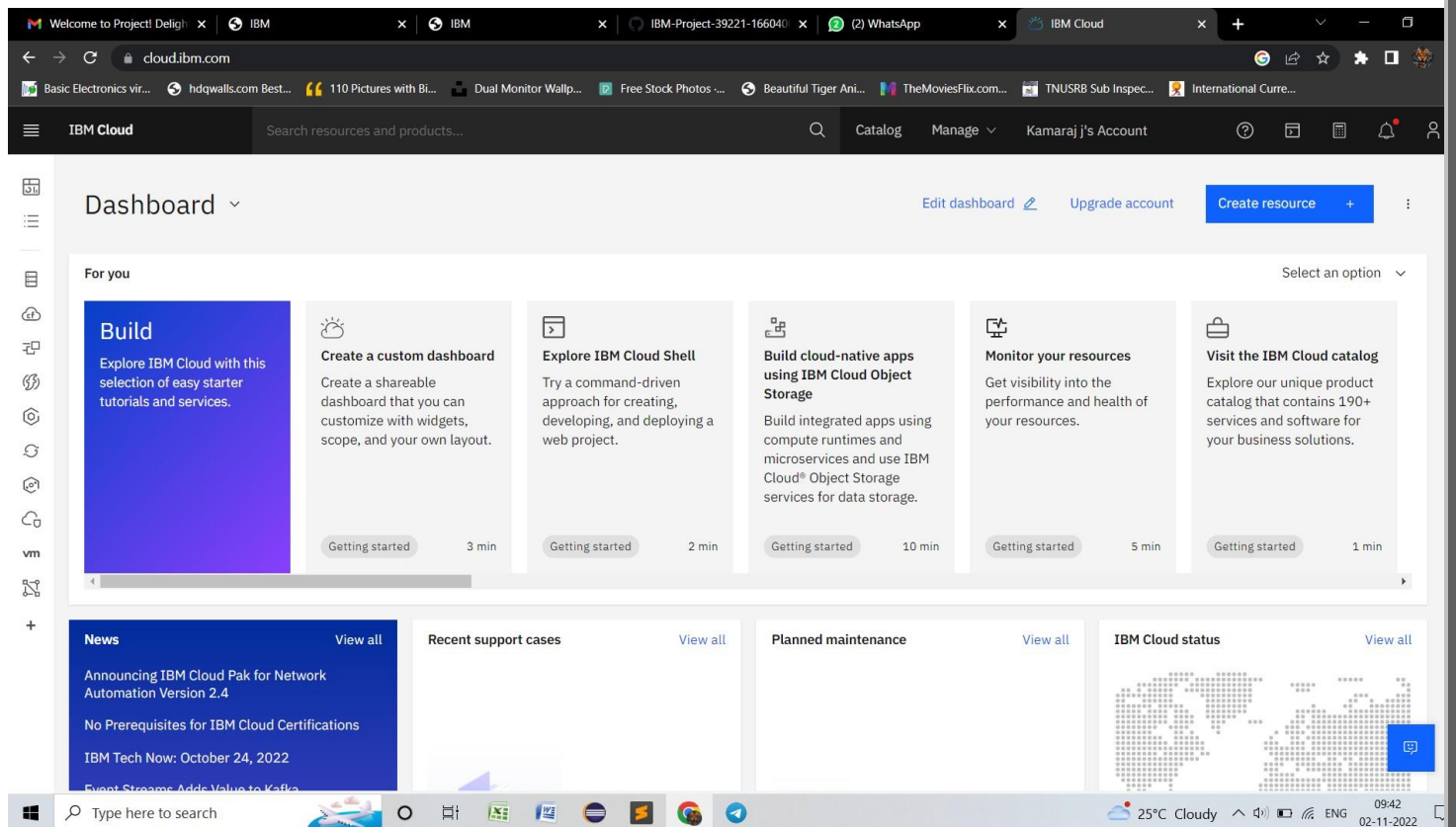
client=paho.Client()
client.on_subscribe = on_subscribe
client.on_message = on_message
client.connect('broker.mqttdashboard.com', 1883)
client.subscribe
```

```
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 Subscriber d
import paho.mqtt.client as paho
import time
import random

def on_publish(client, userdata, 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
-
```

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()
```