

Publish the data to IBM Cloud

Date	13 November 2022
Team ID	PNT2022TMID54421
Project Name	Signs with Smart Connectivity for Better Road Safety

Python code to access subscriber:

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

def on_publish(client,userdata,mid):
    print("published")

client=paho.Client()
client.on_publish=on_publish
client.connect("broker.mqttdashboard.com",1883)
client.loop_start()

while True:
    tem=random.randint(1,100)
    (rc,mid)=client.publish('ibm1',str(tem),qos=1)
    print(tem)
    time.sleep(10)

import paho.mqtt.client as paho

def on_subscribe(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('ibm1',qos=1)

client.loop_forever()

```

The screenshot shows a Python script editor on the left and an IDLE Shell on the right. The script in the editor is a MQTT client that connects to a broker, subscribes to a topic, and publishes random numbers. The shell shows the output of the script, which is a series of 'published' messages followed by random numbers.

```

pythonscript.py - D:/IBM work in progress/pythonscript.py (3.11.0)
File Edit Format Run Options Window Help
import paho.mqtt.client as paho
import time
import random

def on_publish(client,userdata,mid):
    print("published")

client=paho.Client()
client.on_publish=on_publish
client.connect("broker.mqttdashboard.com",1883)
client.loop_start()

while True:
    tem=random.randint(1,100)
    (rc,mid)=client.publish('ibm1',str(tem),qos=1)
    print(tem)
    time.sleep(10)

import paho.mqtt.client as paho
def on_subscribe(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('ibm1',qos=1)
client.loop_forever()

IDLE Shell 3.11.0*
File Edit Shell Debug Options Window Help
Python 3.11.0 (main, Oct 24 2022, 18:26:48) [MSC v.1933 64 bit (AMD64)]
Type "help", "copyright", "credits" or "license()" for more
>>>
===== RESTART: D:/IBM work in progress/pythonscript.py =
6
published
39
published
97
published
44
published
63
published
63
published
64
published

```

PROGRAM:

#IBM Watson IOT Platform

#pip install wiotp-sdk

import wiotp.sdk.device

import time

import random

myConfig = {

"identity": {

```
"orgId": "gsqz5f",  
"typeId": "NANDY",  
"deviceId": "12345" },  
"auth": { "token": "9876543210" }  
}
```

```
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()
```

```

#IBM Watson IoT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "gsqz5f",
        "typeId": "NANDY",
        "deviceId": "12345" },
    "auth": { "token": "9876543210" }
}
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, onP
        rint("Published data Successfully: %s", myData)
        client.commandCallback = myCommandCallback
        time.sleep(2)
    client.disconnect()

```

```

Python 3.7.4 Shell
File Edit Shell Debug Options Window Help
Published data Successfully: %s ('temperature': 50, 'humidity': 45)
Published data Successfully: %s ('temperature': 8, 'humidity': 28)
Published data Successfully: %s ('temperature': 46, 'humidity': 1)
Published data Successfully: %s ('temperature': 78, 'humidity': 49)
Published data Successfully: %s ('temperature': 81, 'humidity': 41)
Published data Successfully: %s ('temperature': 73, 'humidity': 50)
Published data Successfully: %s ('temperature': 76, 'humidity': 34)
Published data Successfully: %s ('temperature': 2, 'humidity': 81)
Published data Successfully: %s ('temperature': 33, 'humidity': 32)
Published data Successfully: %s ('temperature': 18, 'humidity': 76)
Published data Successfully: %s ('temperature': 68, 'humidity': 79)
Published data Successfully: %s ('temperature': 85, 'humidity': 39)
Published data Successfully: %s ('temperature': -10, 'humidity': 96)
Published data Successfully: %s ('temperature': 112, 'humidity': 23)
Published data Successfully: %s ('temperature': 63, 'humidity': 52)
Published data Successfully: %s ('temperature': -18, 'humidity': 36)
Published data Successfully: %s ('temperature': 103, 'humidity': 70)
Published data Successfully: %s ('temperature': 38, 'humidity': 12)
Published data Successfully: %s ('temperature': 6, 'humidity': 31)
Published data Successfully: %s ('temperature': 0, 'humidity': 71)
Published data Successfully: %s ('temperature': 76, 'humidity': 24)
Published data Successfully: %s ('temperature': -16, 'humidity': 73)
Published data Successfully: %s ('temperature': -8, 'humidity': 35)
Published data Successfully: %s ('temperature': 60, 'humidity': 49)
Published data Successfully: %s ('temperature': 79, 'humidity': 97)
Published data Successfully: %s ('temperature': 93, 'humidity': 72)
Published data Successfully: %s ('temperature': -13, 'humidity': 72)
Published data Successfully: %s ('temperature': 67, 'humidity': 50)
Published data Successfully: %s ('temperature': 108, 'humidity': 83)
Published data Successfully: %s ('temperature': 71, 'humidity': 32)
Published data Successfully: %s ('temperature': 47, 'humidity': 75)
Published data Successfully: %s ('temperature': 65, 'humidity': 16)
Published data Successfully: %s ('temperature': 10, 'humidity': 83)
Published data Successfully: %s ('temperature': 24, 'humidity': 76)
Published data Successfully: %s ('temperature': 109, 'humidity': 31)
Published data Successfully: %s ('temperature': 15, 'humidity': 24)
Published data Successfully: %s ('temperature': 0, 'humidity': 34)
Published data Successfully: %s ('temperature': 44, 'humidity': 87)
Published data Successfully: %s ('temperature': 59, 'humidity': 94)
Ln 5

```