

DEVELOP THE PYTHON SCRIPT

Publish data to the IBM Cloud

Team ID : PNT2022TMID22778

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 - /usr/python/python3.10/publish.py (3.10.0)
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
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
```

vimeo Solutions Features Resources Watch

```
Command Prompt - python subscriber.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,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('iottopic',qos=1)
client.loop_forever()
```

⋮

🔧

👤

📌

🌐

⚡

🔒

⚙️

Your boardsPublic boards

Create New Board

IOT MONITOR 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

The screenshot displays a web browser window with multiple tabs, including 'Welcome to Project! Deligi...', 'IBM', 'IBM-Project-39221-16604...', '(2) WhatsApp', and 'IBM Cloud'. The active tab is 'cloud.ibm.com'. The main content area shows the 'Dashboard' with a sidebar on the left containing icons for various services. The dashboard features a 'For you' section with six tiles: 'Build', 'Create a custom dashboard', 'Explore IBM Cloud Shell', 'Build cloud-native apps using IBM Cloud Object Storage', 'Monitor your resources', and 'Visit the IBM Cloud catalog'. Below this is a 'News' section with articles about IBM Cloud Pak for Network Automation Version 2.4, prerequisites for IBM Cloud Certifications, and the IBM Tech Now event on October 24, 2022. Other sections include 'Recent support cases', 'Planned maintenance', and 'IBM Cloud status'. The bottom of the browser window shows a Windows taskbar with the search bar, taskbar icons, and system tray information indicating 25°C Cloudy, 09:42, and 02-11-2022.

IoT monitor board

Line chart showing data over time (23:17 to 23:21). The chart displays three data series: **randomNumber** (blue line), **sampleObject.xcord** (green line), and **sampleObject.ycord** (red line). The y-axis ranges from 0 to 150. A red shaded area highlights the data between 23:18 and 23:21. A legend at the bottom indicates the series names. A status bar at the bottom right of the chart area shows **1 Simulation running**.

Dashboard

For you

- Build**: Explore IBM Cloud with this selection of easy starter tutorials and services. (Getting started: 3 min)
- Create a custom dashboard**: Create a shareable dashboard that you can customize with widgets, scope, and your own layout. (Getting started: 2 min)
- Explore IBM Cloud Shell**: Try a command-driven approach for creating, developing, and deploying a web project. (Getting started: 2 min)
- Build cloud-native apps using IBM Cloud Object Storage**: Build integrated apps using compute runtimes and microservices and use IBM Cloud® Object Storage services for data storage. (Getting started: 10 min)
- Monitor your resources**: Get visibility into the performance and health of your resources. (Getting started: 5 min)
- Visit the IBM Cloud catalog**: Explore our unique product catalog that contains 190+ services and software for your business solutions. (Getting started: 1 min)

News (View all)

- Announcing IBM Cloud Pak for Network Automation Version 2.4
- No Prerequisites for IBM Cloud Certifications
- IBM Tech Now: October 24, 2022
- Event Streams Adds Value to Kafka

Recent support cases (View all)

Planned maintenance (View all)

IBM Cloud status (View all)

25°C Cloudy 09:42 02-11-2022

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()

```

