

Publish Data to the IBM Cloud

Date	4 November 2022
Team Id	PNT2022TMID16954
Project Name	Project - Signs with smart connectivity for Better road safety

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("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)

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('iottopic', qos=1)
client.loop_forever()

```

The screenshot shows a Windows desktop environment. On the left, a text editor window titled 'subscribe.py - C:/Python/Python37/subscribe.py (3.7.4)' contains the following Python code:

```

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)
    (re,mid) = client.publish('iottopic',str(temp),qos=1)
    print(temp)
    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('iottopic',qos=1)
client.loop_forever()

```

On the right, a terminal window titled '*Python 3.7.4 Shell*' shows the output of the script. It displays the Python version and architecture, followed by a restart command and a series of 'Publish the data' messages with increasing line numbers (30, 22, 27, 27, 7, 3, 3).

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()

```

The screenshot shows a Windows desktop environment. On the left, a code editor window titled 'publish.py - C:/Python/Python37/publish.py (3.7.4)' contains the following Python code:

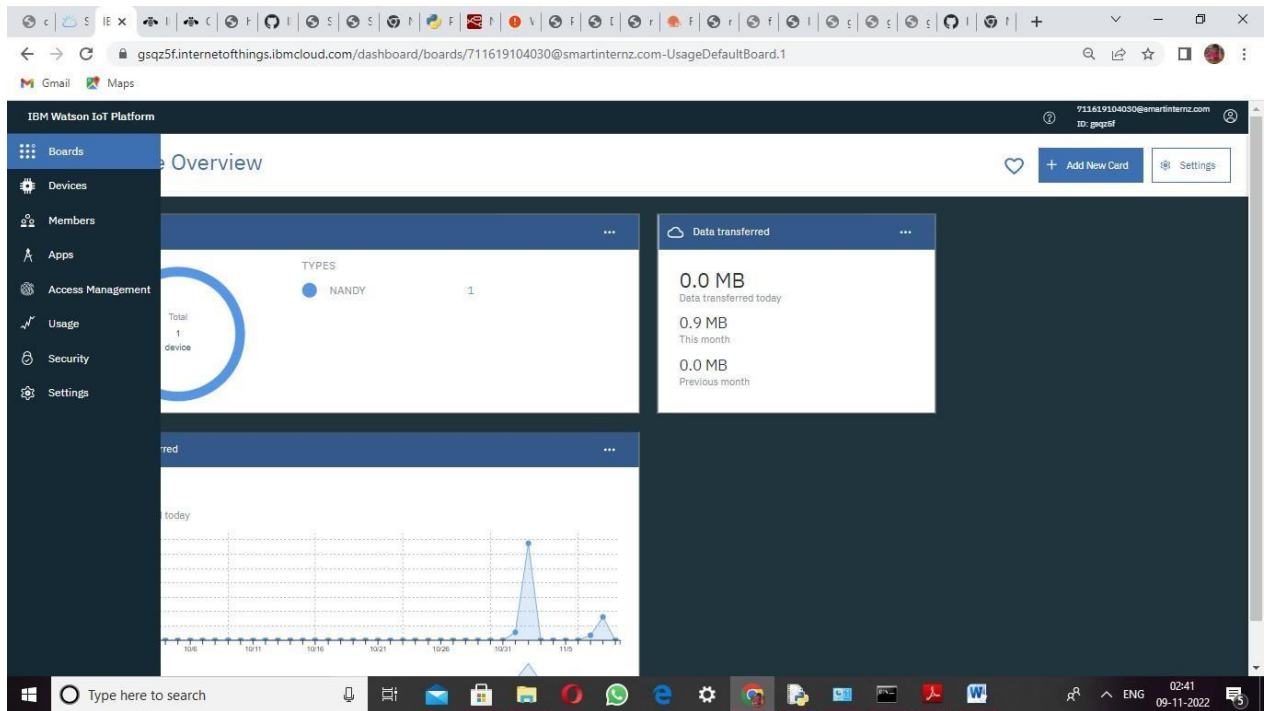
```

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()

```

On the right, a terminal window titled '*Python 3.7.4 Shell*' displays the output of the script. It shows a continuous stream of messages: 'Published data Successfully: %s ('temperature': X, 'humidity': Y)', where X and Y are random values. The messages are listed vertically, showing the script's execution over time. The Windows taskbar at the bottom indicates the time is 02:42 on 09-11-2022.

Publish the data to the ibm cloud:



The screenshot shows the IBM Watson IoT Platform dashboard, specifically the 'Devices' page. The page has tabs for 'Browse', 'Action', 'Device Types', and 'Interfaces'. A table lists the devices, with one device (ID 12345) shown as 'Connected'. Below the table, there is a detailed view of the selected device, including its identity, device information, recent events, state, and logs.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location	Added By	Device Class
12345	Connected	NANDY	Device	Nov 3, 2022 12:12 AM		711619104030@smartinternz.com	

Identity: Device ID 12345, Device Type NANDY, Date Added Nov 3, 2022 12:12 AM, Added By 711619104030@smartinternz.com, Connection Status Connected, Connection Time: Nov 9, 2022 2:40 AM, Client Address: 106.222.125.219 SecureToken

IBM Watson IoT Platform

711619104030@smartinternz.com

ID: gsqz5f

Browse

Action

Device Types

Interfaces

Add Device

12345

Connected

NANDY

Device

Nov 3, 2022 12:12 AM

711619104030@smartinternz.com

...

Identity

Device Information

Recent Events

State

Logs

X

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
status	{"temperature":27,"humidity":72}	json	a few seconds ago
status	{"temperature":3,"humidity":56}	json	a few seconds ago
status	{"temperature":21,"humidity":54}	json	a few seconds ago
status	{"temperature":3,"humidity":28}	json	a few seconds ago
status	{"temperature":0,"humidity":85}	json	a few seconds ago

Items per page 50

1-1 of 1 item

1 of 1 page

<

1

>

Type here to search

Taskbar icons: File Explorer, Mail, Calendar, Store, Edge, WhatsApp, etc.

System tray: ENG, 02:31, 09-11-2022

