

**Develop a python script  
Publish Data to the IBM Cloud**

Date	04 November 2022
Team ID	PNT2022TMID13627
Project Name	SMART WASTE MANAGEMENT SYSTEM FOR METROPOLITIAN CITIES
Maximum Marks	4 Marks

**SMART WASTE MANAGEMENT SYSTEM FOR  
METROPOLITIAN CITIES**

The screenshot shows a Python script in a text editor and its execution in a Python 3.6.5 Shell. The script is designed to publish data to the IBM Cloud IoT Platform using the Paho MQTT client. It includes a comment in red: **#Through python coding we are going to access the subscriber**. The script imports the Paho MQTT client, time, and random modules. It defines a function `on_publish` that prints the data. The main logic creates a client, connects to `broker.mqttdashboard.com` on port 1883, and enters a loop where it publishes random integers (1-30) to the topic `iottopic` with a QoS of 1, sleeping for 10 seconds between publishes.

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

The execution output in the Python 3.6.5 Shell shows the restart of the script and the following output:

```
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 informati
n.
>>>
===== RESTART: E:/IBM/Others/Develop a python script/
publish.py =====
7
Publish the data
19
Publish the data
10
Publish the data
```

The screenshot shows a Python script in a text editor and its execution in a Python 3.6.5 Shell. The script is designed to subscribe to data from the IBM Cloud IoT Platform using the Paho MQTT client. It imports the Paho MQTT client and defines two functions: `on_subscribe` to print the subscriber ID and granted QoS, and `on_message` to print the message topic, QoS, and payload. The main logic creates a client, connects to `broker.mqttdashboard.com` on port 1883, subscribes to the topic `iottopic` with a QoS of 1, and enters a loop to receive messages.

```
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 execution output in the Python 3.6.5 Shell shows the following output:

```
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
```

Service Details - IBM Cloud x IBM Watson IoT Platform x +

hqb56m.internetofthings.ibmcloud.com/dashboard/devices/browse

IBM Watson IoT Platform

Search

Browse Action Device Types Interfaces

Add Device +

Device ID	Status	Device Type	Class ID	Date Added
abcd	Disconnected	123	Device	Nov 4, 2022 11:51 AM

Identity Device Information **Recent Events** State Logs

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

Event	Value	Format	Last Received
event_1	{\"randomNumber\":75}	json	a few seconds ago
event_1	{\"randomNumber\":5}	json	a few seconds ago
event_1	{\"randomNumber\":33}	json	a few seconds ago
event_1	{\"randomNumber\":56}	json	a few seconds ago
event_1	{\"randomNumber\":67}	json	a few seconds ago

1 Simulation running

30°C Cloudy 11:53 AM 11/4/2022

Service Details - IBM Cloud x IBM Watson IoT Platform x +

internetofthings.ibmcloud.com

IBM Watson IoT Platform

Search

Collect data from

Cars

and make value from it

Learn More

Cookie Preferences

30°C Cloudy 11:54 AM 11/4/2022

## 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()

```

