

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

Date	04 November 2022
Team ID	PNT2022TMID05343
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 titled "publish.py" and is located at "E:/IBM/Others/Develop a python script/publish.py (3.6.5)". The script's purpose is to publish data to the IBM Cloud IoT platform. It imports the paho.mqtt.client module as paho, and also imports the time and random modules. It defines a function on\_publish(client, userdata, mid) that prints "Publish the data ". The script then creates a paho.Client object, sets the on\_publish callback, connects to the broker.mqttdashboard.com on port 1883, and starts the loop. It enters a while True loop where it generates a random integer between 1 and 30, publishes it to the iottopic with a QoS of 1, prints the value, and sleeps for 10 seconds.

```
#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 Python 3.6.5 Shell output shows the following:

```
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 titled "subscribe.py" and is located at "E:/IBM/Others/Develop a python script/subscribe.py (3.6.5)". The script's purpose is to subscribe to data from the IBM Cloud IoT platform. It imports the paho.mqtt.client module as paho. It defines a function on\_subscribe(client, userdata, mid, granted\_qos) that prints "subscriber:" followed by the mid and granted\_qos. It also defines a function on\_message(client, userdata, msg) that prints the message topic, qos, and payload. The script then creates a paho.Client object, sets the on\_subscribe and on\_message callbacks, connects to the broker.mqttdashboard.com on port 1883, subscribes to the iottopic with a QoS of 1, and starts the loop\_forever.

```
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 Python 3.6.5 Shell output shows the following:

```
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
```

The screenshot shows the IBM Watson IoT Platform interface. At the top, there's a navigation bar with 'Browse', 'Action', 'Device Types', and 'Interfaces'. A table lists devices, with 'abcd' selected. Below the table, a modal window displays the 'Recent Events' for device 'abcd'. The events are listed in a table with columns: Event, Value, Format, and Last Received. The events are all 'event\_1' with different random number values in JSON format, received 'a few seconds ago'. A status message at the bottom of the modal says '1 Simulation running'.

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

The screenshot shows the IBM Watson IoT Platform landing page. The header includes the 'IBM Watson IoT Platform' logo and a user profile with email '210419104143@smartin...' and ID '(select org)'. The main content area features a dark background with a central graphic of a circuit board. The graphic is divided into two sections: 'Collect data from' on the left and 'and make value from it' on the right. The word 'Things' is prominently displayed in the center. Below the graphic, there's a 'Learn More' link. The bottom of the page shows a Windows taskbar with various application icons and system tray information.

## 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()

```

