

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

Date	19 November 2022
Team ID	PNT2022TMID26725
Project Name	SMART WASTE MANAGEMENT SYSTEM FOR METROPOLITIAN CITIES
Maximum Marks	4 Marks

**SMART WASTE MANAGEMENT SYSTEM FOR
METROPOLITIAN CITIES**

```
#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)
```

Python 3.6.5 Shell

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 information.

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

```
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()
```

Python 3.6.5 Shell

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 are tabs for 'Browse', 'Action', 'Device Types', and 'Interfaces'. A search bar is on the left. A table lists devices with columns: Device ID, Status, Device type, Class ID, and Date Added. The device 'abcd' is highlighted, showing a status of 'Disconnected' and a device type of '123'. Below this, a modal window titled 'Recent Events' is open, showing a table of events. The table has columns: Event, Value, Format, and Last Received. The events are all 'event_1' with values like '{"randomNumber":75}', '{"randomNumber":5}', '{"randomNumber":33}', '{"randomNumber":56}', and '{"randomNumber":67}', all in 'json' format and received 'a few seconds ago'. A status bar at the bottom indicates '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 homepage. The main graphic features the word 'Things' in the center, with lines connecting to various icons representing data collection and processing. The text 'Collect data from' is on the left and 'and make value from it' is on the right. A 'Learn More' button is at the bottom. The browser's address bar shows 'internetofthings.ibmcloud.com'. The taskbar at the bottom includes a search bar and several application icons.

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()

```

