

SPRINT-1

Date	17 November 2022
Team ID	PNT2022TMID11549
Project Name	Smart Waste Management System For Metropolitan Cities

***creating python script and publish data to ibm watson**

The screenshot shows a Python script named `swms.py` in an IDE. The script imports `wiotp.sdk.device` and `time`, and uses `random` to generate data. It defines a `myConfig` dictionary with device details and a `myCommandCallback` function that prints received commands. The main loop connects to the IoT platform, publishes random level data, and sleeps for 2 seconds. The Run console on the right shows a stream of 'Published data Successfully' messages with varying level values.

```
1 import wiotp.sdk.device
2 import time
3 import random
4 myConfig = {
5     "identity": {
6         "orgId": "zal46w",
7         "typeId": "NodeMCU",
8         "deviceId": "12345"},
9     "auth": {
10         "token": "1234567890"
11     }
12 }
13 def myCommandCallback(cmd):
14     print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
15     m=cmd.data['command']
16     client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
17     client.connect()
18     while True:
19         level=random.randint(0,100)
20         level_d={'Level':level}
21         client.publishEvent(eventId="status", msgFormat="json", data=level_d, qos=0, onPublish=None)
22         print("Published data Successfully: %s", level_d)
23         client.commandCallback = myCommandCallback
24         time.sleep(2)
25     client.disconnect()
```

The screenshot shows the IBM Watson IoT Platform dashboard for device ID 12345. The device is a NodeMCU and is in a 'Connected' state. The 'Recent Events' tab is selected, showing a live stream of data. The table below lists the recent events.

Event	Value	Format	Last Received
status	{"Level":51}	json	a few seconds ago
status	{"Level":20}	json	a few seconds ago
status	{"Level":99}	json	a few seconds ago
status	{"Level":84}	json	a few seconds ago
status	{"Level":8}	json	a few seconds ago

```
pythonProject3 / swms.py
7  "typeId": "NodeMCU",
8  "deviceId": "12345",
9  "auth": {
10   "token": "1234567890"
11 }
12 }
13 def myCommandCallback(cmd):
14     print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
15     cmd = cmd.data['command']
16     client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
17     client.connect()
18     while True:
19         level=random.randint(0,100)
20         weight= random.randint(0, 1000)
21         level_d={'Level':level}
22         weight_d = {'Weight': weight}
23         client.publishEvent(eventId="status", msgFormat="json", data=level_d, qos=0, onPublish=None)
24         client.publishEvent(eventId="status", msgFormat="json", data=weight_d, qos=0, onPublish=None)
25         print("Published data Successfully: %s", level_d)
26         print("Published data Successfully: %s", weight_d)
27         client.commandCallback = myCommandCallback
28         time.sleep(2)
29     client.disconnect()
```

Run: swms x

```
Published data Successfully: %s {'Level': 49}
Published data Successfully: %s {'Weight': 577}
Published data Successfully: %s {'Level': 22}
Published data Successfully: %s {'Weight': 642}
Published data Successfully: %s {'Level': 26}
Published data Successfully: %s {'Weight': 203}
Published data Successfully: %s {'Level': 44}
Published data Successfully: %s {'Weight': 715}
Published data Successfully: %s {'Level': 56}
Published data Successfully: %s {'Weight': 790}
Published data Successfully: %s {'Level': 40}
Published data Successfully: %s {'Weight': 781}
Published data Successfully: %s {'Level': 49}
Published data Successfully: %s {'Weight': 292}
Published data Successfully: %s {'Level': 9}
Published data Successfully: %s {'Weight': 173}
Published data Successfully: %s {'Level': 86}
Published data Successfully: %s {'Weight': 455}
Published data Successfully: %s {'Level': 86}
Published data Successfully: %s {'Weight': 466}
Published data Successfully: %s {'Level': 19}
Published data Successfully: %s {'Weight': 481}
Published data Successfully: %s {'Level': 91}
Published data Successfully: %s {'Weight': 338}
Published data Successfully: %s {'Level': 1}
Published data Successfully: %s {'Weight': 453}
```

zal46w.internetofthings.ibmcloud.com/dashboard/devices/browse

IBM Watson IoT Platform

910619106301@smartinternz.com
ID: zal46w

Browse Action Device Types Interfaces

Add Device +

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
status	{"Weight":520}	json	a few seconds ago
status	{"Level":94}	json	a few seconds ago
status	{"Weight":308}	json	a few seconds ago
status	{"Level":68}	json	a few seconds ago
status	{"Weight":314}	json	a few seconds ago

Items per page 50 | 1-1 of 1 item

1 of 1 page

0 Simulations running

```
pythonProject3 - main.py

main.py | iot.py | swms.py
1 import wiotp.sdk.device
2 import time
3 from geopy.geocoders import Nominatim
4 import random
5 geoloc=Nominatim(user_agent="geoapiExercises")
6 lat="9.914470"
7 long="78.143418"
8 location=geoloc.reverse(lat+","+long)
9 addr=location.raw['address']
10 latitude = {'Lat': lat}
11 longitude = {'Long': long}
12 print(addr.get('suburb',''))
13 myConfig = {
14     "identity": {
15         "orgId": "zal46w",
16         "typeId": "NodeMCU",
17         "deviceId": "12345",
18         "auth": {
19             "token": "1234567890"
20         }
21     }
22 }
23 def myCommandCallback(cmd):
24     print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
25     m=cmd.data['command']
26     client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
27     client.connect()
28
29 while True:
30     if level>50 and weight>500
```

Run: iot | main

```
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
2022-11-17 21:17:56,619 wiotp.sdk.device.client
.DeviceClient INFO Connected successfully:
d:zal46w:NodeMCU:12345
Published data Successfully: %s {'Level': 6}
Published data Successfully: %s {'Weight': 328}
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
Published data Successfully: %s {'Level': 71}
Published data Successfully: %s {'Weight': 528}
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
Published data Successfully: %s {'Level': 76}
Published data Successfully: %s {'Weight': 573}
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
Published data Successfully: %s {'Level': 97}
Published data Successfully: %s {'Weight': 79}
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
```

Version Control | TODO | Problems | Terminal | Python Packages | Python Console

30:29 CRLF UTF-8 1 space Python 3.9

9:18 PM 11/17/2022

```
pythonProject3 - main.py

main.py | iot.py | swms.py
24 client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
25 client.connect()
26
27 while True:
28     level=random.randint(0,100)
29     weight= random.randint(0, 1000)
30     if level>50 and weight>500:
31         client.publishEvent(eventId="status", msgFormat="json", data={'V_alert': "pick dumpster i n the
32         time.sleep(2)
33         client.publishEvent(eventId="status", msgFormat="json", data={'V_alert': ""}, qos=0,
34         onPublish=None)
35         level_d={'Level':level}
36         weight_d = {'Weight': weight}
37         client.publishEvent(eventId="status", msgFormat="json", data=level_d, qos=0, onPublish=None)
38         client.publishEvent(eventId="status", msgFormat="json", data=weight_d, qos=0, onPublish=None)
39         client.publishEvent(eventId="status", msgFormat="json", data=latitude, qos=0, onPublish=None)
40         client.publishEvent(eventId="status", msgFormat="json", data=longitude, qos=0, onPublish=None)
41         print("Published data Successfully: %s", level_d)
42         print("Published data Successfully: %s", weight_d)
43         print("Published data Successfully: %s", latitude)
44         print("Published data Successfully: %s", longitude)
45         client.commandCallback = myCommandCallback
46         time.sleep(2)
47         client.disconnect()
48
49 while True:
50     if level>50 and weight>500
```

Run: iot | main

```
'78.143418'}
Published data Successfully: %s {'Level': 50}
Published data Successfully: %s {'Weight': 246}
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
Published data Successfully: %s {'Level': 69}
Published data Successfully: %s {'Weight': 873}
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
Published data Successfully: %s {'Level': 84}
Published data Successfully: %s {'Weight': 637}
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
Published data Successfully: %s {'Level': 75}
Published data Successfully: %s {'Weight': 145}
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
Published data Successfully: %s {'Level': 57}
Published data Successfully: %s {'Weight': 5}
Published data Successfully: %s {'Lat': '9.914470'}
Published data Successfully: %s {'Long': '78.143418'}
```

Version Control | TODO | Problems | Terminal | Python Packages | Python Console

30:29 CRLF UTF-8 1 space Python 3.9

9:19 PM 11/17/2022

The screenshot displays the IBM Watson IoT Platform interface. At the top, there's a header bar with the logo, navigation links (Browse, Action, Device Types, Interfaces), user information (910619106301@smartinternz.com, ID: zal46w), and an 'Add Device' button.

The main content area shows a list of devices. The selected device has ID 12345, status 'Connected', type 'NodeMCU', class 'Device', and was added on Nov 15, 2022 at 10:41 PM. Below this, tabs are visible: Identity, Device Information, Recent Events (selected), State, and Logs.

The 'Recent Events' tab displays a message: "The recent events listed show the live stream of data that is coming and going from this device." Below this message is a table of events:

Event	Value	Format	Last Received
status	{"V_alert": "pick dumpster i n theAnna Nagarstre..."}	json	a few seconds ago
status	{"Long": "78.143418"}	json	a few seconds ago
status	{"Lat": "-9.914470"}	json	a few seconds ago
status	{"Weight": "671"}	json	a few seconds ago
status	{"Level": "6"}	json	a few seconds ago

At the bottom right, a status box indicates "0 Simulations running".

Event	Value	Format	Last Received
status	{"Long": "78.143418"}	json	a few seconds ago
status	{"Lat": "9.914470"}	json	a few seconds ago
status	{"Weight": 533}	json	a few seconds ago
status	{"Level": 3}	json	a few seconds ago
status	{"V_alert": ""}	json	a few seconds ago