

Sprint 2

Date	19 November 2022
Team ID	PNT2022TMID48693
Project Name	Smart waste management system for metropolitan cities
Story Points	15

Develop the python code to find the GPS location using Latitude and Longitude (random values) and send it to Node red using IBM Watson platform and view location of bins on map

PYTHON CODE :

```
import wiotp.sdk.device

import time import random

myConfig = {
    "identity": {
        "orgId": "fb3a19",
        "typeId": "sample",
        "deviceId": "123"
    },
    "auth": {
        "token": "Abcdefgh"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']
```

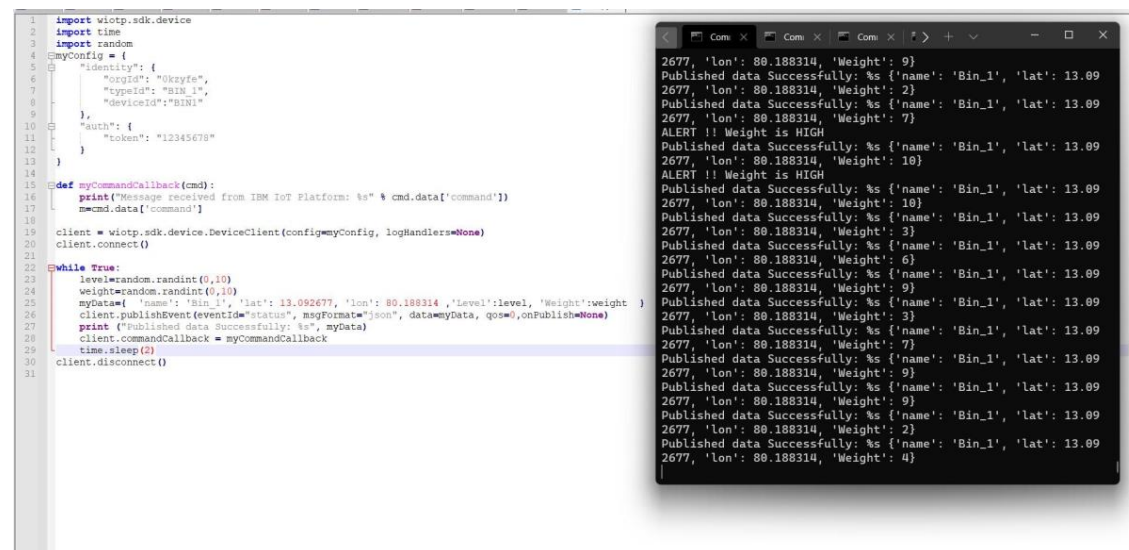
```
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
```

```
client.connect()
```

```
while True:
```

```
    level=random.randint(0,10) weight=random.randint(0,10) myData={ 'name': 'Bin_1', 'lat':  
13.092677, 'lon': 80.188314, 'Level':level, 'Weight':weight } client.publishEvent(eventId="status",  
msgFormat="json", data=myData, qos=0,onPublish=None) print ("Published data Successfully:  
%", myData) client.commandCallback = myCommandCallback time.sleep(2) client.disconnect()
```

Output in python IDLE :

The image shows a screenshot of a Python IDE (IDLE) with two windows. The left window displays the Python code for connecting to the IBM Watson IoT platform and publishing data. The right window shows the output of the program, which includes timestamps, location data, and status updates. The code in the left window is as follows:

```
1 import wiotp.sdk.device
2 import time
3 import random
4 myConfig = {
5     "identity": {
6         "keyId": "0kzyfe",
7         "keyType": "BIN_1",
8         "deviceId": "BIN1"
9     },
10    "auth": {
11        "token": "12345678"
12    }
13 }
14
15 def myCommandCallback(cmd):
16     print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
17     #cmd.data['command']
18
19 client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
20 client.connect()
21
22 while True:
23     level=random.randint(0,10)
24     weight=random.randint(0,10)
25     myData={ 'name': 'Bin_1', 'lat': 13.092677, 'lon': 80.188314, 'Level':level, 'Weight':weight }
26     client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0,onPublish=None)
27     print ("Published data Successfully: %s", myData)
28     client.commandCallback = myCommandCallback
29     time.sleep(2)
30 client.disconnect()
```

The output window on the right shows the following text:

```
2677, 'lon': 80.188314, 'Weight': 9}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 2}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 7}
ALERT !! Weight is HIGH
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 10}
ALERT !! Weight is HIGH
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 10}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 3}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 6}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 9}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 3}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 7}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 9}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 9}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 2}
Published data Successfully: %s {'name': 'Bin_1', 'lat': 13.09
2677, 'lon': 80.188314, 'Weight': 4}
```

IBM Watson IOT platform :

IBM Watson IoT Platform

rsangeetrsangeet07@gmail.com
ID: 0kzyle

Browse

Action

Device Types

Interfaces

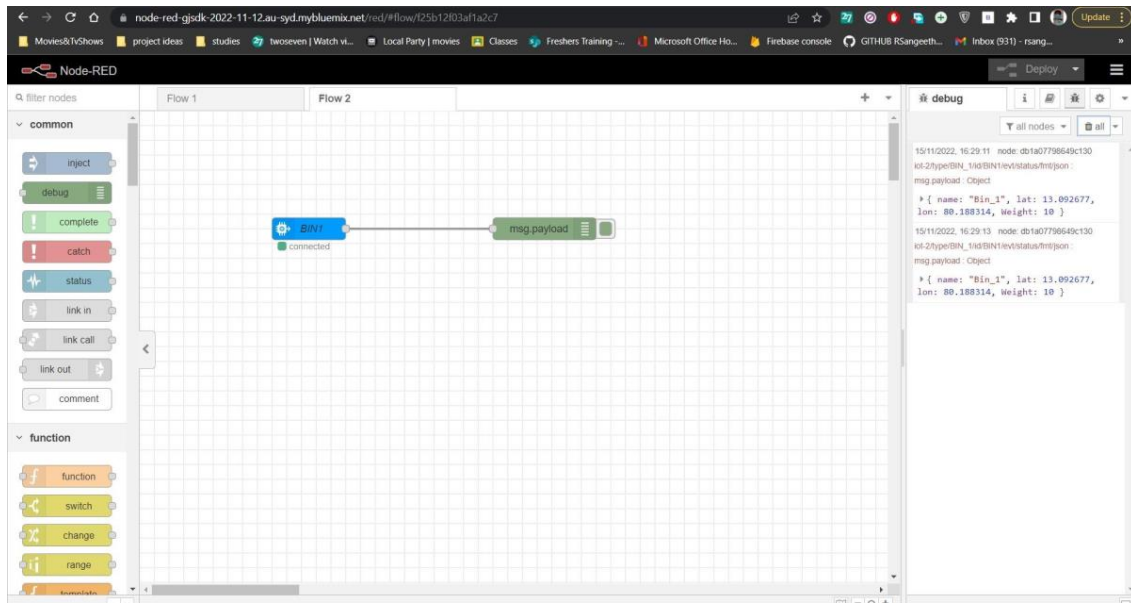
Add Device

>	<input type="checkbox"/>	987654321	Connected	Ultrasonic_sensor	Device	Nov 12, 2022 1:48 PM	
>	<input type="checkbox"/>	BIN1	Connected	BIN_1	Device	Nov 15, 2022 12:18 PM	→ ...
<div><div>Identity</div><div>Device Information</div><div>Recent Events</div><div>State</div><div>Logs</div><div>×</div></div> <div><div>Device ID</div><div>BIN1</div><div>Device Type</div><div>BIN_1</div><div>Date Added</div><div>Nov 15, 2022 12:18 PM</div><div>Added By</div><div>rsangeetrsangeet07@gmail.com</div><div>Connection Status</div><div>Connected</div><div>Connection Time: Nov 15, 2022 2:26 PM</div><div>Client Address: 106.198.11.75 SecureToken</div></div>							
>	<input type="checkbox"/>	BIN2	Connected	BIN_2	Device	Nov 15, 2022 12:22 PM	
>	<input type="checkbox"/>	BIN3	Connected	BIN_3	Device	Nov 15, 2022 12:24 PM	

Items per page 50 | 1-5 of 5 items

1 of 1 page < 1 >

Node Red Platform :



Location of Trash Bins Shown in World Map :

