

Date	5 November 2022
Team ID	PNT2022TMID42565
Project Name	Smart waste management system for metropolitan cities
Story Points	15

Sprint 2

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 requests
import json
import ibmiotf.application
import ibmiotf.device
import time
import random
import sys

organization = "70icwf"
deviceType="1234"
deviceId="12345678"
authMethod="token"
authToken="S_OVsw4ICr5-Vk9A9x"
```

```

def myCommandCallback(cmd):
    global a
    print("Command received: %s" %cmd.data['command'])
    control=cmd.data['command']
    print(control)
try:
    deviceOptions={"org":organization, "type": deviceType, "id" : deviceId, "auth-method":
authMethod, "auth-token": authToken}
    deviceCli = ibmiotf.device.Client(deviceOptions)
except Exception as e:
    print("Caught exception connecting device: %s" %str(e))
    sys.exit()
deviceCli.connect()
while True:

    distance= random.randint(10,70)
    loadcell= random.randint(5,15)
    data= {'dist':distance,'load':loadcell}

    if loadcell < 13 and loadcell > 15:
        load= "90 %"
    elif loadcell < 8 and loadcell > 12:
        load= "60 %"
    elif loadcell < 4 and loadcell > 7:
        load= "40 %"
    else:
        load = "0 %"

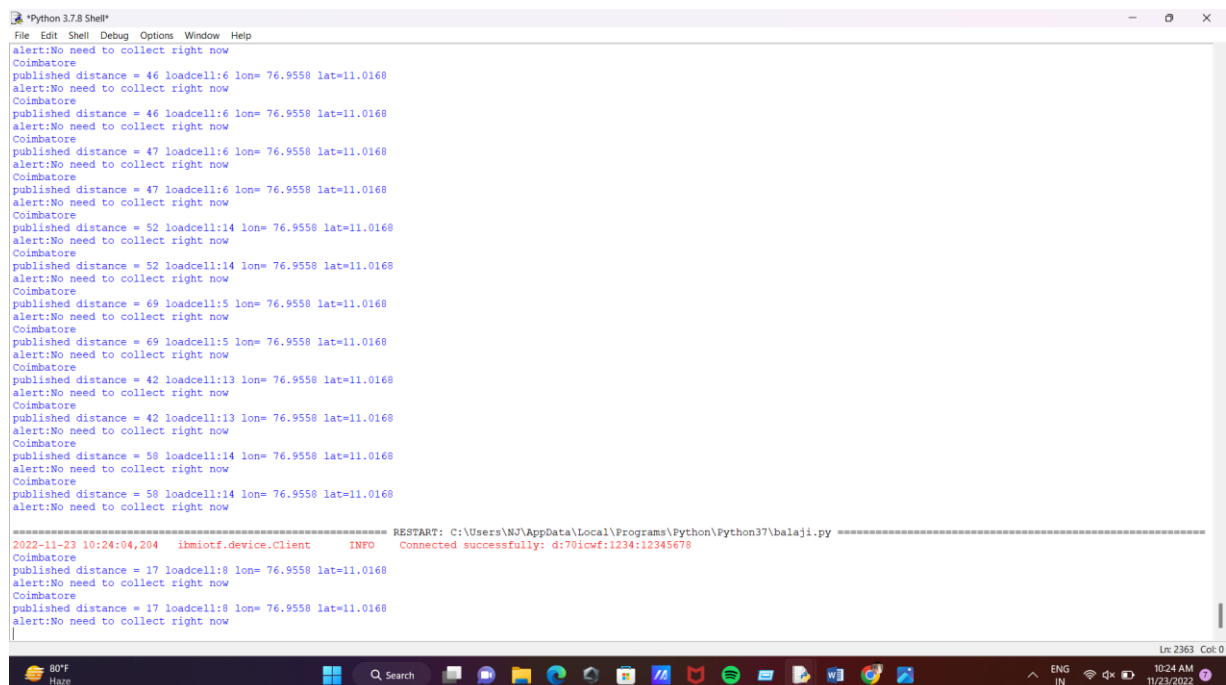
    if distance < 15:
        dist = 'Warning:' 'Trash is getting high, Time to collect 90 %'

    elif distance < 40 and distance >16:
        dist = 'Warning:' 'Trash is above 70 %'

    elif distance < 60 and distance > 41:
        dist = 'Warning:' '40 %'
    else:
        dist = 'Warning:' '17 %'
    if load == "90 %" or distance == "90 %":

```

Output in python IDLE :



IBM Watson IoT platform :

The screenshot shows the IBM Watson IoT Platform interface. The top navigation bar includes links for Application Details, IBM Cloud, Node-RED, Cloudant Dashboard, Node-RED Dashboard, Service Details, and IBM Watson IoT Platform. The main content area displays a table of devices. The first device, with ID 12345678, is in a 'Connected' state. Below the table, a detailed view of this device is shown, including its identity, device information, recent events, state, and logs. The device information section lists the Device ID (12345678), Device Type (1234), Date Added (Nov 22, 2022 5:15 PM), Added By (esakkiammalagu@gmail.com), and Connection Status (Connected). The connection status also shows the Connection Time (Nov 23, 2022 9:59 AM) and Client Address (157.51.194.68 SecureToken). The bottom of the screen shows a Windows taskbar with various application icons and system information.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
12345678	Connected	1234	Device	Nov 22, 2022 5:15 PM	

Device Information

- Device ID: 12345678
- Device Type: 1234
- Date Added: Nov 22, 2022 5:15 PM
- Added By: esakkiammalagu@gmail.com
- Connection Status: Connected
- Connection Time: Nov 23, 2022 9:59 AM
- Client Address: 157.51.194.68 SecureToken

Node Red Platform :

The screenshot displays the Node-RED interface. The left sidebar shows a list of nodes categorized into 'common' and 'function'. The main workspace contains a flow diagram with two parallel processing paths. Each path starts with an 'IBM IoT' node, followed by a 'dist' node, a 'load' node, and a 'msg.payload' node. The first path also includes a 'distance-bin 1' node, and the second path includes a 'loadcell-bin-1' node. The right sidebar shows a 'debug' console with a log of messages, including timestamps, node IDs, and payload data. The bottom of the screen shows a Windows taskbar with various application icons and system information.

Flow Diagram:

- Path 1: IBM IoT → dist → load → msg.payload → distance-bin 1
- Path 2: IBM IoT → dist → load → msg.payload → loadcell-bin-1

Debug Console Log:

```
11/23/2022, 10:25:38 AM node: 0222472b5851107  
iot-2/hyper/1234/id/12345678/ev/iotSensor/rm/json :  
msg.payload : Object  
{ dist: 13, load: 10 }  
11/23/2022, 10:25:38 AM node: 022649a0d098  
iot-2/hyper/1234/id/12345678/ev/iotSensor/rm/json :  
msg.payload : Object  
{ dist: 13, load: 10 }  
11/23/2022, 10:25:38 AM node: 022649a0d098  
iot-2/hyper/1234/id/12345678/ev/iotSensor/rm/json :  
msg.payload : Object  
{ dist: 13, load: 10 }  
11/23/2022, 10:25:38 AM node: 0222472b5851107  
iot-2/hyper/1234/id/12345678/ev/iotSensor/rm/json :  
msg.payload : string[34]  
"alert:No need to collect right now"  
11/23/2022, 10:25:39 AM node: 022649a0d098  
iot-2/hyper/1234/id/12345678/ev/iotSensor/rm/json :  
msg.payload : string[34]  
"alert:No need to collect right now"  
11/23/2022, 10:25:39 AM node: 022649a0d098  
iot-2/hyper/1234/id/12345678/ev/iotSensor/rm/json :  
msg.payload : string[34]  
"alert:No need to collect right now"
```