Sprint – 2

Date	**05 November 2022**
Team ID	**PNT2022TMID37209**
Project Name	**Smart waste management system for metropolitan cities.**

Python Code:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random



#Provide your IBM Watson Device Credentials
organization = "nw3318"
deviceType = "123"
deviceId = "1234567"
authMethod = "token"
authToken = "12345687"

# Initialize GPIO



def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
```

```
    status=cmd.data['command']
    if status=="alarmon":
        print ("Message is sent")
    elif status == "alarmoff":
        print("Message not sent")
    else :
        print ("please send proper command")

    #print(cmd)
```

```
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()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11
```

```
        level=random.randint(0,100)



        data = { 'level' : level }
        #print data
        def myOnPublishCallback():
            print ("Published Garbage level = %s C" % level, "to IBM Watson")


        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(10)


        deviceCli.commandCallback = myCommandCallback


# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

Node-Red Input and Output: