#Project: Smart Waste Management System for Metropolitan cities

#Team ID: PNT2022TMID05358

#Installing necessary libraries import wiotp.sdk.device import time import random import requests

import math

#Configuration details for connecting python script to IBM Watson IOT

Platform myConfig = {

"identity": {

"orgId": "mldk59",

"typeId": "pythoncode",

"deviceId":"252525"

},

"auth": {

"token": "QZqODYo6U\*Q6b+IpuC"

} }

def myCommandCallback(cmd):

print("Message received from IBM IoT Platform: %s" % cmd.data['co mmand'])

m=cmd.data['command']

#Connecting the client to ibm watson iot platform

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

#Generate Random values for latitude, longitude in a circular distributio n from the current location and

#alert the garbage collector to go to the particular location where the bin level and bin weight exceeds the threshold

while True:

res = requests.get('https://ipinfo.io/') data = res.json() loc = data['loc'].split(',') theta = random.uniform(0,2\*math.pi) area = (0.05\*\*2)\*math.pi

radius = math.sqrt(random.uniform(0,area/math.pi))

latitude,longitude = [float(loc[0])+radius\*math.cos(theta), float(loc[1] )+radius\*math.sin(theta)]

binlevel=random.randint(10,100)

binweight = random.randint(50,1500)

if binweight>=1000 and binlevel>80:

myData={'latitude':latitude, 'longitude':longitude,'binlevel':binlevel,

'binweight':binweight}

client.publishEvent(eventId="status", msgFormat="json", data=my Data, qos=0, onPublish=None)

##print("Published data Successfully: %s", myData) print("BIN IS FULL..TIME TO EMPTY IT!!!!\n",myData) client.commandCallback = myCommandCallback time.sleep(2)

#break

else :

print("BIN IS IN NORMAL LEVEL...")

time.sleep(2)

#Disconnect the client connection

client.disconnect()