SPRINT – 1

Simulation Creation (Connect Sensor with Python Code)

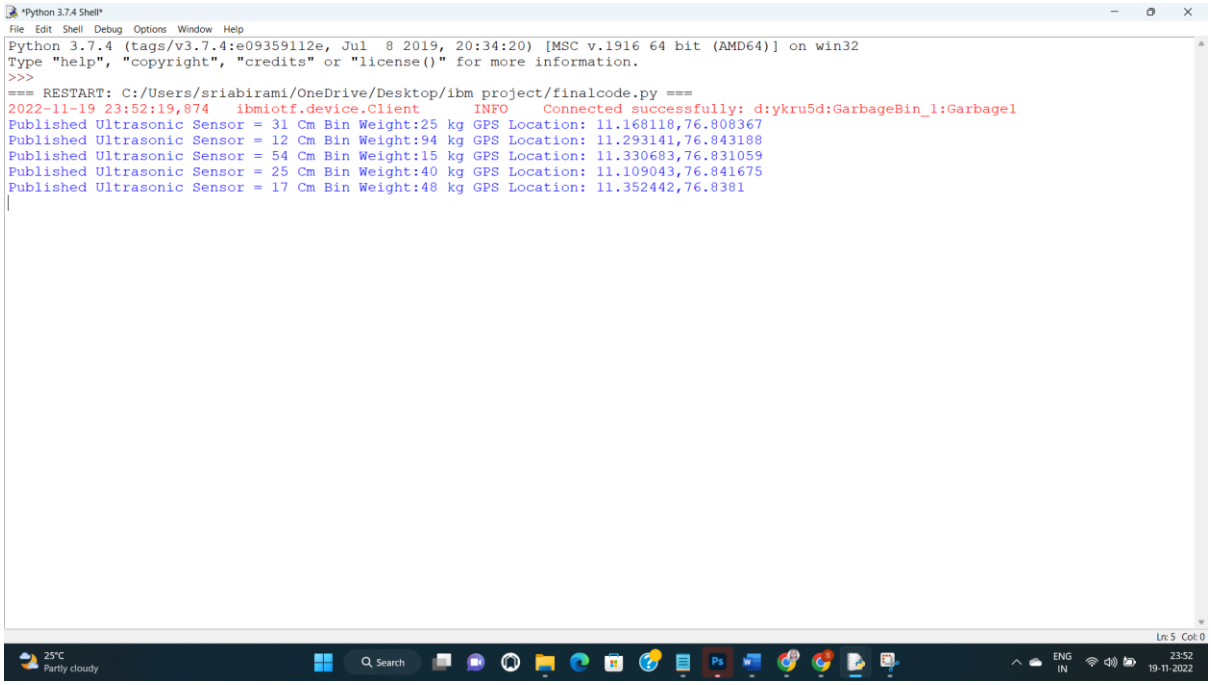
Date	15 November 2022
Project Name	Smart Waste Management System for Metropolitan Cities
Project ID	PNT2022TMID03917

PYTHON CODE

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "ykru5d"
deviceType = "GarbageBin_1"
deviceId = "Garbage1"
authMethod = "token"
authToken = "DKD_K)lt0Yn!yQIeUf"
# Initialize GPIO
def myCommandCallback(cmd):
   print("Command received: %s" % cmd.data['command'])
   status=cmd.data['command']
   if status =="lighton":
      print("led in on")
   else :
      print ("led is off")
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:
#Get Sensor Data from DHT11
    time.sleep(5)
    ult_son=random.randint(0,80)
    weight=random.randint(0,100)
    lat = round(random.uniform(11.03, 11.50), 6)
    long = round(random.uniform(76.80, 76.90), 6)
    gps = str(lat) + str(',') + str(long)
    data = {'Ultrasonic' : ult_son, 'Weight' : weight , 'GPS' : gps}
#print data
    def myOnPublishCallback():
        print ("Published Ultrasonic Sensor = %s Cm" %ult_son, "Bin Weight:%s kg" %weight, "GPS
Location: %s" %gps)
    success = deviceCli.publishEvent("IoTSensor", "json", data,
qos=0,on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTF")
time.sleep(1)
deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
deviceCli.disconnect()
```

Output:



NODERED FLOW FOR SENSOR , GPS AND WEIGHT