TEAM ID	PNT2022TMID47365
PROJECT	Smart Waste Management System For Metropolitan Cities

SPRINT DELIVERY 1:

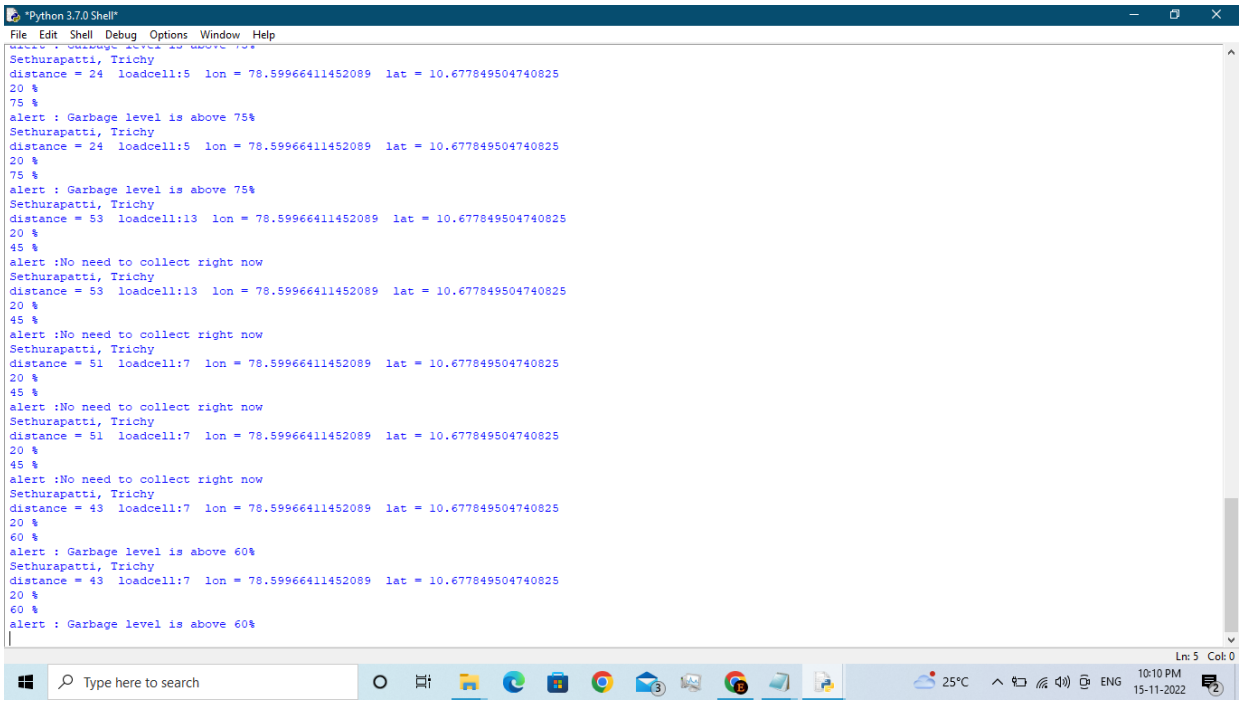
Python script:

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

#IBM Watson Device Credentials.
organization = "sl1jtd"
deviceType = "abcde"
deviceId = "08"
authMethod = "token"
authToken = "830119106008"

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

 #weight of the bin
 if loadcell <= 13 and loadcell >= 15:
     load = "90 %"

 elif loadcell <= 10 and loadcell >= 12:
     load = "75 %"

 elif loadcell <= 8 and loadcell >= 10:
     load = "75 %"

 elif loadcell <= 4 and loadcell >= 7:
     load = "45 %"
```

```
else:
    load = "20 %"

#empty distance in the bin
if distance <= 15:
    dist = '90 %'

elif distance <= 30 and distance >= 16:
    dist = '75 %'

elif distance <= 45 and distance >= 31 :
    dist = '60 %'

elif distance <= 60 and distance >= 46:
    dist = '45 %'

else:
    dist = '25 %'

#alert and warning for garbage level and weight

if load == "90 %" or dist == "90 %":
    warn = 'alert : Garbage level in the trash can is going to be full, Time to collect '

elif load == "75 %" or dist == "75 %":
    warn = 'alert : Garbage level is above 75%'

elif load == "60 %" or dist == "60 %":
    warn = 'alert : Garbage level is above 60%'

else :
    warn = 'alert :' 'No need to collect right now '

def myOnPublishCallback(lat=10.677849504740825,long=78.59966411452089):
    print("Sethurapatti, Trichy")
    print("distance = %s " %distance,"loadcell:%s " %loadcell,"lon = %s " %long,"lat = %s"
%lat)
    print(load)
    print(dist)
    print(warn)

time.sleep(5)


success=deviceCli.publishEvent ("IoTSensor","json",warn,qos=0,on_publish=
myOnPublishCallback)

success=deviceCli.publishEvent ("IoTSensor","json",data,qos=0,on_publish=
myOnPublishCallback)


if not success:
    print("not connected to ibmiot")
```

```
time.sleep(20)


deviceCli.commandCallback=myCommandCallback
#disconnect the device
deviceCli.disconnect
```

OUTPUT:



IBM WATSON: