SPRINT – 2

TEAM ID:PNT2022TMID36751

SMART WASTE MANAGEMENT SYSTEM FOR METROPOLITAN CITIES

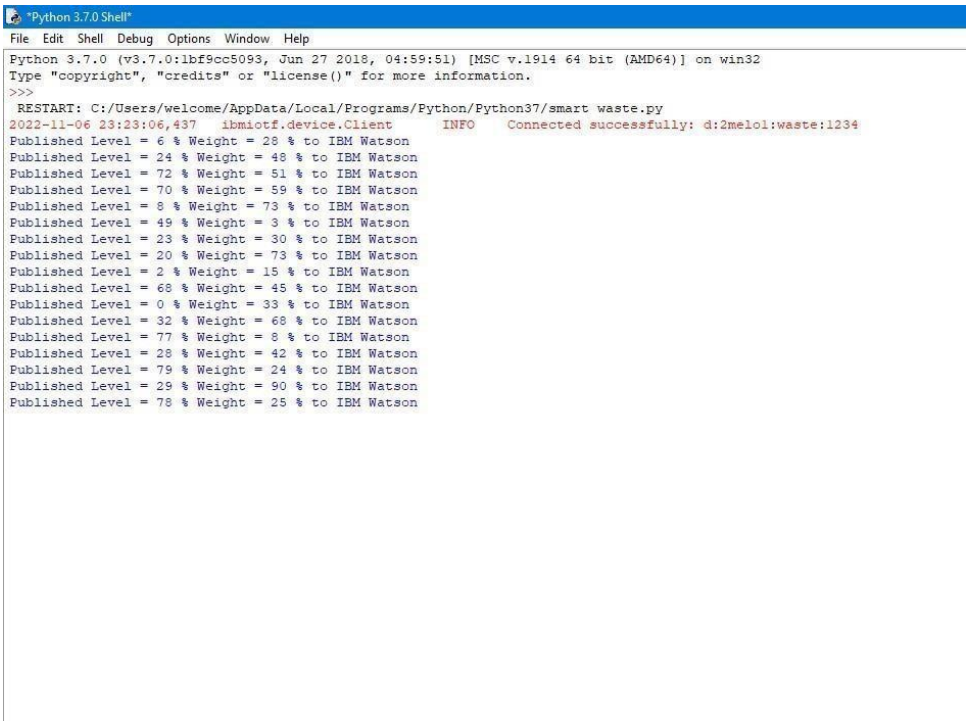
Python Code

```
import time import sys import

ibmiotf.application import

ibmiotf.device import random



#Provide your IBM Watson Device Credentials organization

= "dxjch6" deviceType =

"sprint2" deviceId = "abcd" authMethod =

"token" authToken = "fKX6j?2deWAL)Nsz9h"


# Initialize GPIO



def myCommandCallback(cmd):          print("Command received:

%s" % cmd.data['command']) status=cmd.data['command']  if

status=="waste level":

      print ("waste level monitored")

   else :

      print ("weight level monitored")
```

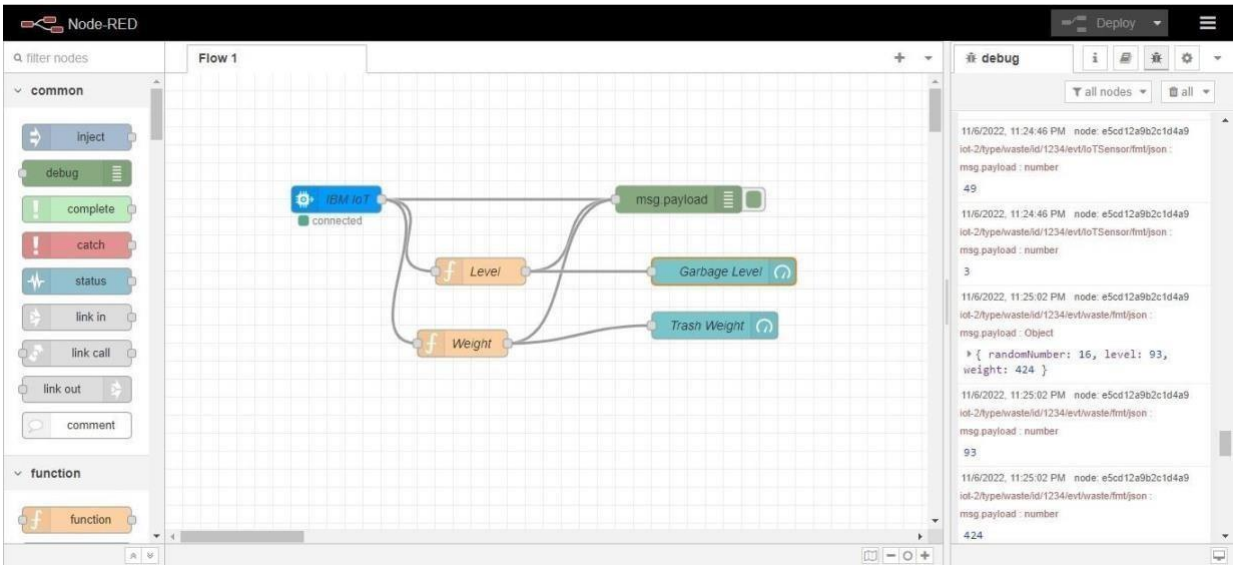
```
   #print(cmd)


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) weight=random.randint(0,100)



      data = { 'level' : level, 'weight': weight }

      #print data

      def myOnPublishCallback():

         print ("Published Level = %s %%" % level, "Weight = %s %%" % weight, "to IBM Watson")



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

```
        if not success:    print("Not  connected
        to                 IoTF")
time.sleep(20)


    deviceCli.commandCallback = myCommandCallback


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

OUTPUT:



NODE RED INPUT AND OUPUT:



Link :
https://node-red-kjery-2022-11-16.eu-gb.mybluemix.net/ui/#!/2?socketid=k8p8Gu58QfjcGOYpAAAD