Final Delivery

Date	19 November 2022
Team ID	PNT2022TMID49521
Project Name	IOT – Smart Waste Management Systems for Metropolitan Cities

1. Final Code for Project

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

#IBM Watson Device Credentials
organization = "sfouh6"
deviceType = "Python"
deviceId = "9238"
authMethod = "token"
authToken = "vishnuprabhu923819106057"



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
deviceCli.connect()

while True:
   print("\nDon't use any capital letters to given an input \nSenosr sensing gas is")     #Unavailable of sensors     in the wokwi and tinkercad, we give inputs manually
```

```
    detect = input()
    Sensing = ()    #Detecting Harmful Wastage
    Location_info = ()  #Sending Location & Contact information

    if detect == "ammonia":     #Harmful material sensing by MQ-137 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"

    elif detect == "hydrogen sulfide": #Harmful material sensing by MQ-136 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    elif detect == "methane":     #Harmful material sensing by TGS-2611 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"

    else:
        Sensing = "Harmful Waste is not detected"
        Location_info = "Everything is normal in the particular house"

    data = { 'Sensing' : Sensing, 'Location_info' : Location_info}
        #print data

    def myOnPublishCallback():
        print ("Published Sensing data - %s " % Sensing, "\nLocation_info - %s" %Location_info, "to IBM Watson")

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

    if not success:
            print("Not connected to IoTF")
    time.sleep(1)


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

Disconnect the device and

application from the cloud

deviceCli.disconnect()

sprint 2.py - C:\Users\ELCOT\Desktop\Smart Waste Management System for Metropolitan cities\Z - Programmaing\Sensing Program\sprint 2.py (3.7.0)

File Edit Format Run Options Window Help

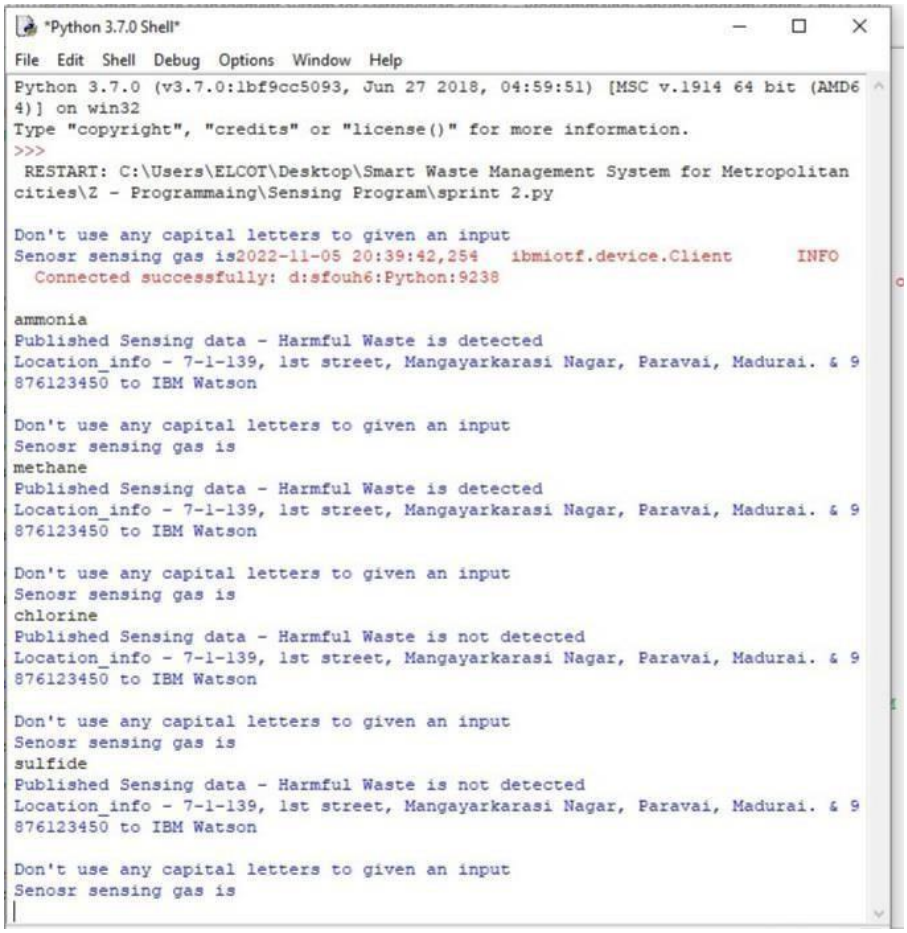
```
import time
import sys
import ibmiotf.application
import ibmiotf.device

#Provide your IBM Watson Device Credentials
organization = "sfouh6"
deviceType = "Python"
deviceId = "9238"
authMethod = "token"
authToken = "vishnuprabhu923819106057"


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
deviceCli.connect()

while True:
    print("\nDon't use any capital letters to given an input \nSenosr sensing gas is")    #Unavailable of sensors in the wokwi and tinkercad, we give inputs manually
    detect = input()
    Sensing = ()
    Location_info = ()
    if detect == "ammonia":     #Harmful material sensing by MQ-137 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    elif detect == "hydrogen sulfide":   #Harmful material sensing by MQ-136 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    elif detect == "methane":     #Harmful material sensing by TGS-2611 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    else:
        Sensing = "Harmful Waste is not detected"
```

sprint 2.py - C:\Users\ELCOT\Desktop\Smart Waste Management System for Metropolitan cities\Z - Programmaing\Sensing Program\sprint 2.py (3.7.0)

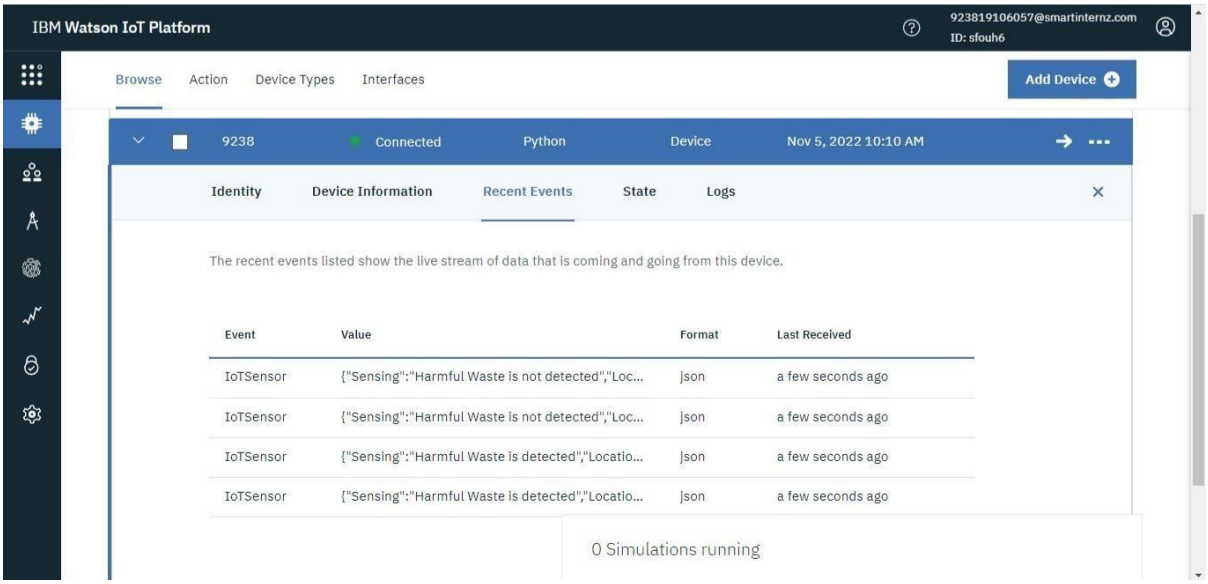
File Edit Format Run Options Window Help

```
        #.....................................

except Exception as e:
        print("Caught exception connecting device: %s" % str(e))
        sys.exit()

# Connect and send a datapoint
deviceCli.connect()

while True:
    print("\nDon't use any capital letters to given an input \nSenosr sensing gas is")    #Unavailable of sensors in the wokwi and tinkercad, we give inputs manually
    detect = input()
    Sensing = ()
    Location_info = ()
    if detect == "ammonia":     #Harmful material sensing by MQ-137 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    elif detect == "hydrogen sulfide":   #Harmful material sensing by MQ-136 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    elif detect == "methane":     #Harmful material sensing by TGS-2611 gas sensor
        Sensing = "Harmful Waste is detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"
    else:
        Sensing = "Harmful Waste is not detected"
        Location_info = "7-1-139, 1st street, Mangayarkarasi Nagar, Paravai, Madurai. & 9876123450"

    data = { 'Sensing' : Sensing, 'Location_info' : Location_info}
        #print data
    def myOnPublishCallback():
        print ("Published Sensing data - %s " % Sensing, "\nLocation_info - %s" %Location_info, "to IBM Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=myOnPublishCallback)
    if not success:
            print("Not connected to IoTF")
    time.sleep(1)


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

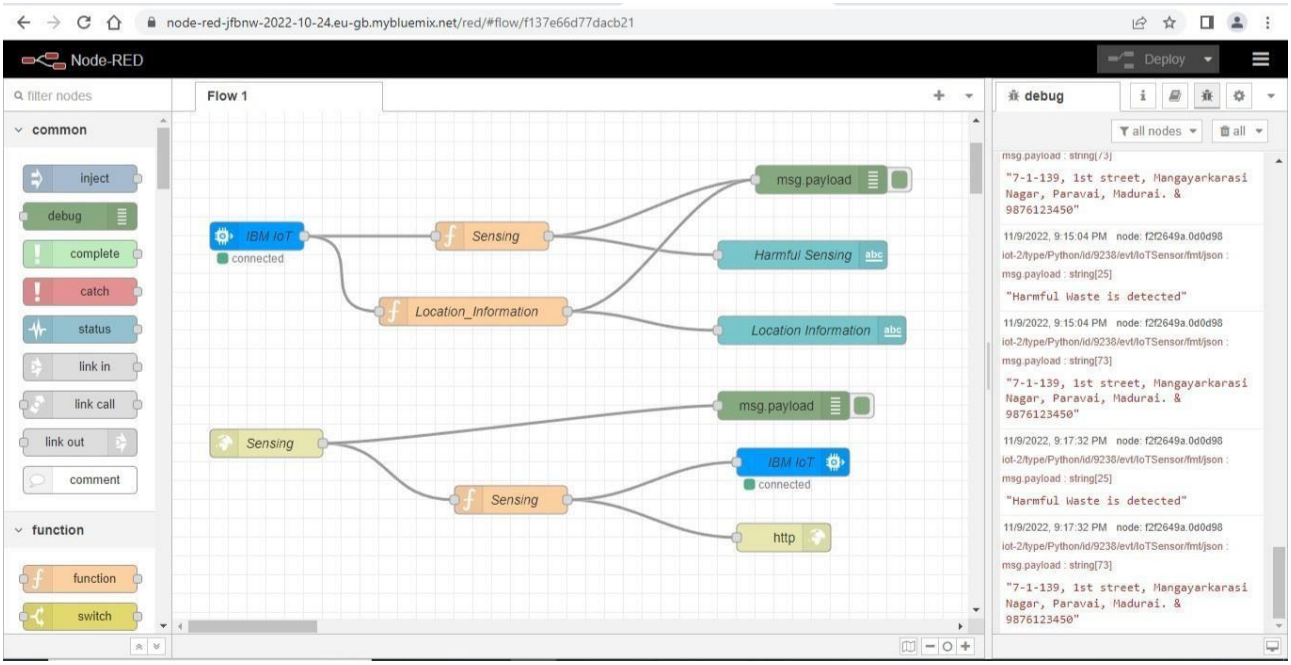
Output in Python Shell Module:



Output in IBM IOT Cloud Recent Events:



Output in Node-Red Debug:



Output in Web UI: