

## DEVELOP THE PYTHON SCRIPT

<b>Date:</b>	10 November 2022
<b>Team Id:</b>	PNT2022TMID30274
<b>Project Name:</b>	Gas leakage Monitoring & Alerting System for Industries

We are getting oxygen, Toxic Gas level, Temperature and Humidity of environment of workers as input through the beacon scanner (python code)

### Publishing the Python Script:



```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "b3ltn1",
        "typeId": "print1",
        "deviceId": "printid"
    },
    "auth": {
        "token": "z?7tcRfcekco08R6f2"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']

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

while True:
    o2=random.randint(25,100)
    def my_function():
        othergas=random.randint(2,100)
        time.sleep(20)
        return othergas
    othergas=my_function()
    temp=random.randint(0,100)
    humidity=random.randint(0,100)
    limit=50
    if(othergas >= limit):
        myData = { 'Alert': "Alert the gas is leaked",'othergas':othergas}
    else:
        myData={'oxygen':o2, 'othergas':othergas, 'temperature':temp,'humidity':humidity}
    client.publishEvent(eventId="Gas Sensor", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s" % myData)
    client.commandCallback = myCommandCallback
    time.sleep(15)
client.disconnect()
```

Sprint-2.py - C:\Users\user\Downloads\Sprint-2.py (3.7.0)

File Edit Format Run Options Window Help

```
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "b31tni",
        "typeId": "print1",
        "deviceId": "printid"
    },
    "auth": {
        "token": "z97tcRfcekco08R6f2"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']

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

while True:
    o2=random.randint(25,100)
    def my_function():
        othergas=random.randint(2,100)
        time.sleep(20)
        return othergas
    othergas=my_function()
    temp=random.randint(0,100)
    humidity=random.randint(0,100)
    limit=50
    if(othergas >= limit):
        myData = { 'Alert': "Alert the gas is leaked", 'othergas':othergas }
    else:
        myData={'oxygen':o2, 'othergas':othergas, 'temperature':temp,'humidity':humidity}
    client.publishEvent(eventId="Gas Sensor", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(15)
client.disconnect()
```

Python 3.7.0 Shell

File Edit Shell Debug Options Window Help

```
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 82}
Published data Successfully: %s {'oxygen': 89, 'othergas': 47, 'temperature': 94, 'humidity': 66}
Published data Successfully: %s {'oxygen': 69, 'othergas': 16, 'temperature': 17, 'humidity': 30}
Published data Successfully: %s {'oxygen': 35, 'othergas': 35, 'temperature': 98, 'humidity': 15}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 54}
Published data Successfully: %s {'oxygen': 65, 'othergas': 37, 'temperature': 45, 'humidity': 57}
Published data Successfully: %s {'oxygen': 56, 'othergas': 6, 'temperature': 16, 'humidity': 14}
Published data Successfully: %s {'oxygen': 39, 'othergas': 30, 'temperature': 82, 'humidity': 68}
Published data Successfully: %s {'oxygen': 98, 'othergas': 36, 'temperature': 28, 'humidity': 46}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 71}
Published data Successfully: %s {'oxygen': 100, 'othergas': 41, 'temperature': 99, 'humidity': 57}
Published data Successfully: %s {'oxygen': 78, 'othergas': 4, 'temperature': 10, 'humidity': 67}
Published data Successfully: %s {'oxygen': 61, 'othergas': 46, 'temperature': 8, 'humidity': 75}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 82}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 66}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 71}
Published data Successfully: %s {'oxygen': 62, 'othergas': 17, 'temperature': 38, 'humidity': 15}
Published data Successfully: %s {'oxygen': 26, 'othergas': 15, 'temperature': 93, 'humidity': 52}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 89}
Published data Successfully: %s {'oxygen': 52, 'othergas': 4, 'temperature': 22, 'humidity': 14}
Published data Successfully: %s {'oxygen': 33, 'othergas': 34, 'temperature': 40, 'humidity': 57}
Published data Successfully: %s {'oxygen': 95, 'othergas': 9, 'temperature': 58, 'humidity': 12}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 54}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 68}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 73}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 88}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 94}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 59}
Published data Successfully: %s {'oxygen': 91, 'othergas': 45, 'temperature': 72, 'humidity': 6}
Published data Successfully: %s {'oxygen': 28, 'othergas': 29, 'temperature': 50, 'humidity': 100}
Published data Successfully: %s {'oxygen': 88, 'othergas': 28, 'temperature': 80, 'humidity': 7}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas': 61}
```

Ln 11 Col 0

IBM Cloud x IBM Wat x Node-RE x Node-RE x KNOWL x IBM x IBM-EP x IBM-39 x IBM-39 x IBM-Pro x Settings x

← → ↻

b31tni.internetofthings.ibmcloud.com/dashboard/devices/browse

Knowledge Institut... Google Online Courses - L... MAKING INDIA THE... KNOWLEDGE INSTL... Oracle Live SQL Node-RED YouTube

IBM Watson IoT Platform

2k19ccc039@iot.ac.in ID: b31tni

Browse Action Device Types Interfaces

Add Device

Browse Devices

All Devices Diagnose

This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.

Q Search by Device ID

Device Simulator

	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
>	assignment	Disconnected	Assignment-4	Device	Nov 1, 2022 3:31 PM	
>	printid	Connected	print1	Device	Nov 4, 2022 10:52 AM	
>	sprint-1	Disconnected	sprint	Device	Nov 3, 2022 1:56 PM	

Items per page 50 | 1-3 of 3 items 1 of 1 page < 1 >

IBM Cloud x IBM Wat... x Node-RE x Node-RE x KNOWL... x IBM x IBM-EP... x IBM-395... x IBM-397... x IBM-Pro... x Settings x +

b31tni.internetofthings.ibmcloud.com/dashboard/devices/browse

Knowledge Institut... Google Online Courses - Le... MAKING INDIA THE... KNOWLEDGE INSTL... Oracle Live SQL Node-RED YouTube

IBM Watson IoT Platform

2k19ece039@klot.ac.in ID: b31tni

Browse

Action

Device Types

Interfaces

Add Device

>

assignment

Disconnected

Assignment-4

Device

Nov 1, 2022 3:31 PM

printid

Connected

print1

Device

Nov 4, 2022 10:52 AM

→

...

Identity

Device Information

Recent Events

State

Logs

×

The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
Gas Sensor	{"Alert":"Alert the gas is leaked","othergas":52}	json	a few seconds ago
Gas Sensor	{"oxygen":87,"othergas":31,"temperture":7,"hum...	json	a few seconds ago

>

sprint-1

Disconnected

sprint

Device

Nov 3, 2022 1:56 PM

Items per page 50

| 1-3 of 3 items

1 of 1 page

<

1

>