

PYTHON CODE (GAS, TEMPERATURE, HUMIDITY, PRESSURE)

Date	3 NOVEMBER 2022
Team ID	PNT2022TMID10501
Project Name	GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES

PYTHON CODE

```
#IBM Watson IOT Platform
import wiotp.sdk.device
import time
import random

myConfig = {
"identity": {
    "orgId": "yf0dyy ",
    "typeId": "Faraaz ",
    "deviceId": "12345"
},
"auth": {
    "token": "VJTDPRX@f&4Vuox8ms "
}
}

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:

```
    gas=random.randint(0,100)
```

```
    temp=random.randint(0,100)
```

```
    hum=random.randint(0,100)
```

```
    pre=random.randint(0,100)
```

```
    myData={'Hazardous Gas':gas, 'Temperature':temp, 'Humidity':hum,
'Pressure':pre }
```

```
    client.publishEvent(eventId="status", msgFormat="json",
```

```
    data=myData,qos=0, onPublish=None)
```

```
    print("Published data Successfully: %s", myData)
```

```
    client.commandCallback                                =
```

```
    myCommandCallbacktime.sleep(2)
```

client.disconnect() OUTPUT:

The screenshot displays a web application interface for managing devices. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains icons for various functions. The main content area shows a table of devices with columns: Device ID, Status, Device Type, Class ID, Date Added, and Descriptive Location. The first device listed is '12345', which is 'Disconnected', of type 'Nagarajan', class 'Device', and was added on 'Oct 31, 2022 11:38 AM'. Below the table, there is a section for 'Recent Events' with a sub-header 'Identity'. A message states: 'The recent events listed show the live stream of data that is coming and going from this device.' Below this is a table of events with columns: Event, Value, Format, and Last Received. The events are listed as 'event_1' with values like '{"Hazardous Gas":61,"Temperature":88,"Humidit...}' and format 'json', all received 'a few seconds ago'. At the bottom right, there is a watermark 'Activate Windows' and a status bar indicating '1 Simulation running'.

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
12345	Disconnected	Nagarajan	Device	Oct 31, 2022 11:38 AM	

Event	Value	Format	Last Received
event_1	{"Hazardous Gas":61,"Temperature":88,"Humidit...	json	a few seconds ago
event_1	{"Hazardous Gas":20,"Temperature":36,"Humidit...	json	a few seconds ago
event_1	{"Hazardous Gas":79,"Temperature":56,"Humidit...	json	a few seconds ago
event_1	{"Hazardous Gas":52,"Temperature":82,"Humidit...	json	a few seconds ago
event_1	{"Hazardous Gas":26,"Temperature":33,"Humidit...	json	a few seconds ago

1 Simulation running



Browse Action Device Types Interfaces

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
event_1	{"Hazardous Gas":57,"Temperature":98,"Humidity":...	json	a few seconds ago
event_1	{"Hazardous Gas":3,"Temperature":35,"Humidity":...	json	a few seconds ago
event_1	{"Hazardous Gas":69,"Temperature":74,"Humidity":...	json	a few seconds ago
event_1	{"Hazardous Gas":85,"Temperature":51,"Humidity":...	json	a few seconds ago
event_1	{"Hazardous Gas":92,"Temperature":35,"Humidity":...	json	a few seconds ago

Items per page: 50 | 1-1 of 1 item

Device Type: Nagarajan

Events 1 New event type

Event type name event_1 Send

Schedule: 20 Every Minute

Payload

Specify the event payload in the editor window or by uploading a CSV file.

```
0 {
1   "Hazardous Gas": random(0, 100),
2   "Temperature": random(0, 100),
3   "Humidity": random(0, 100),
4   "Pressure": random(0, 100)
5 }
6
```

Cancel Save

Activate Windows
Go to Settings to activate Windows.