

PYTHON CODE (GAS, TEMPERATURE, HUMIDITY, PRESSURE)

Date	3 NOVEMBER 2022
Team ID	PNT2022TMID17768
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-based interface for managing IoT devices. On the left, a sidebar contains navigation icons. The main panel is divided into tabs: 'Browse', 'Action', 'Device Types', and 'Interfaces'. Under the 'Device Types' tab, there are sub-tabs for 'Identity', 'Device Information', 'Recent Events', 'State', and 'Logs'. The 'Recent Events' tab is active, showing a table of events. The table has columns for 'Event', 'Value', 'Format', and 'Last Receive'. Below the table, it indicates 'Items per page 50' and '1-1 of 1 item'.

Overlaid on the right is a configuration window for a device named 'Nagarajan'. This window has a 'Send' button and a 'New event type' button. It shows an 'Event type name' of 'event_1'. The 'Schedule' section is set to '20' and 'Every Minute'. The 'Payload' section contains a JSON object with four fields: 'Hazardous Gas', 'Temperature', 'Humidity', and 'Pressure', each assigned a 'random(0, 100)' value. At the bottom of the window, there are 'Cancel' and 'Save' buttons, along with a watermark for 'Activate Windows'.

Event	Value	Format	Last Receive
event_1	{"Hazardous Gas":57,"Temperature":98,"Humidit...	json	a few secon
event_1	{"Hazardous Gas":3,"Temperature":35,"Humidit...	json	a few secon
event_1	{"Hazardous Gas":69,"Temperature":74,"Humidit...	json	a few secon
event_1	{"Hazardous Gas":85,"Temperature":51,"Humidit...	json	a few secon
event_1	{"Hazardous Gas":92,"Temperature":35,"Humidit...	json	a few secon

```
{  
  "Hazardous Gas": random(0, 100),  
  "Temperature": random(0, 100),  
  "Humidity": random(0, 100),  
  "Pressure": random(0, 100)  
}
```