

PYTHON CODE (GAS, TEMPERATURE, HUMIDITY, PRESSURE)

Date	3 NOVEMBER 2022
Team ID	PNT2022TMID39619
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": "Tajamul ",
    "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)
```

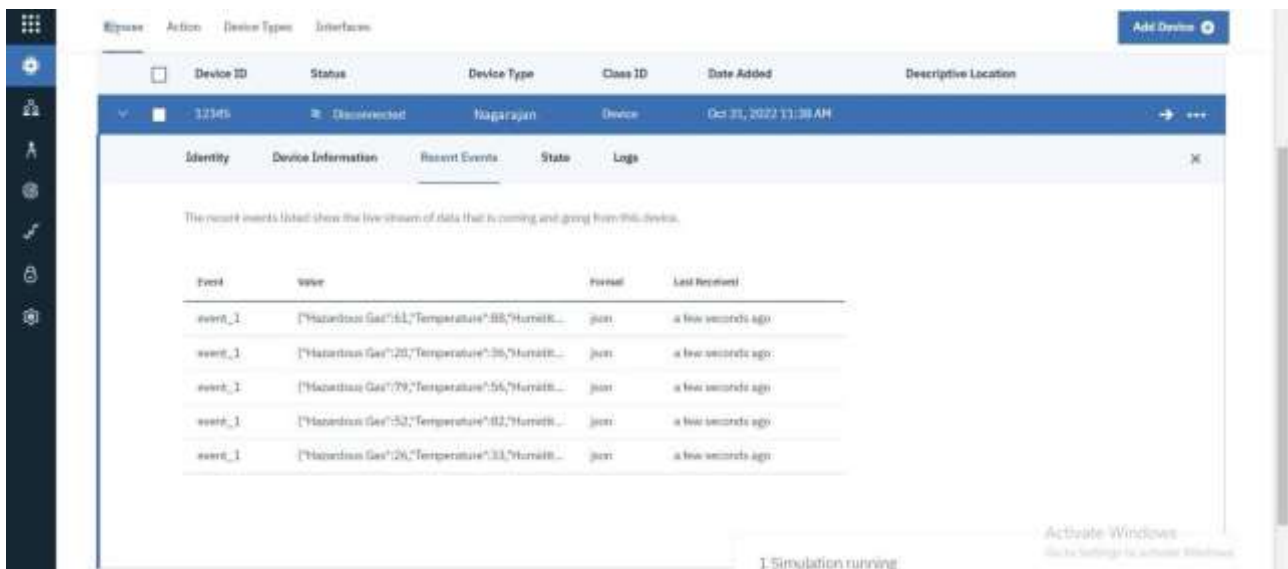
```
    print("Published data Successfully: %s",
```

```
    myData)    client.commandCallback    =
```

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

```
client.disconnect()
```

OUTPUT:



The screenshot displays a simulation interface with a sidebar on the left and a main content area. The main area shows a table of recent events. The table has columns for 'Event', 'Value', 'Format', and 'Last Received'. The data is as follows:

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

At the bottom of the interface, there is a status bar that reads "1. Simulation running".

The screenshot displays the NagaraJen IoT dashboard. The main interface has a top navigation bar with 'Browse', 'Action', 'Device Types', and 'Interfaces'. Below this is a sub-navigation bar with 'Identify', 'Device Information', 'Recent Events', 'State', and 'Logs'. The 'Recent Events' section shows a list of events with columns for Event, Value, Format, and Last Receive. A modal window titled 'Device Type: Nagarajan' is open, showing a form to create a new event type. The form includes fields for 'Event type name' (set to 'event_1'), 'Schedule' (set to 'Every 15 min'), and 'Payload' (a JSON object with sensor data). The modal also has 'Cancel' and 'Save' buttons.

Event	Value	Format	Last Receive
event_1	{"Hazardous Gas":37,"Temperature":50,"Humidity":...	json	a few sec(s)
event_1	{"Hazardous Gas":3,"Temperature":35,"Humidity":...	json	a few sec(s)
event_1	{"Hazardous Gas":69,"Temperature":70,"Humidity":...	json	a few sec(s)
event_1	{"Hazardous Gas":35,"Temperature":51,"Humidity":...	json	a few sec(s)
event_1	{"Hazardous Gas":92,"Temperature":35,"Humidity":...	json	a few sec(s)

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

Device Type: Nagarajan

Events: 1

New event type

Event type name: event_1

Schedule: 20 Every 15 min

Payload: Specify the event payload in the editor window or by uploading a .JSON file.

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

Cancel Save