

## Project Development Phase

### Sprint-2

Date	05 November 2022
Team Id	PNT2022TMID30274
Project Name	Gas Leakage Monitoring & Alerting System for Industries

In this sprint, we are getting the Data from python Code for random sensor data of Hazardous gas levels

Solution:

```
import wiotp.sdk.device
```

```
import time
```

```
import random
```

```
myConfig = {
```

```
    "identity": {
```

```
        "orgId": "b31tni",
```

```
        "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(0,100)  
        time.sleep(5)  
        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,  
'temperture':temp,'humidity':humidity}
```

```
client.publishEvent(eventId="Gas Sensor", msgFormat="json",
data=myData, qos=0, onPublish=None)

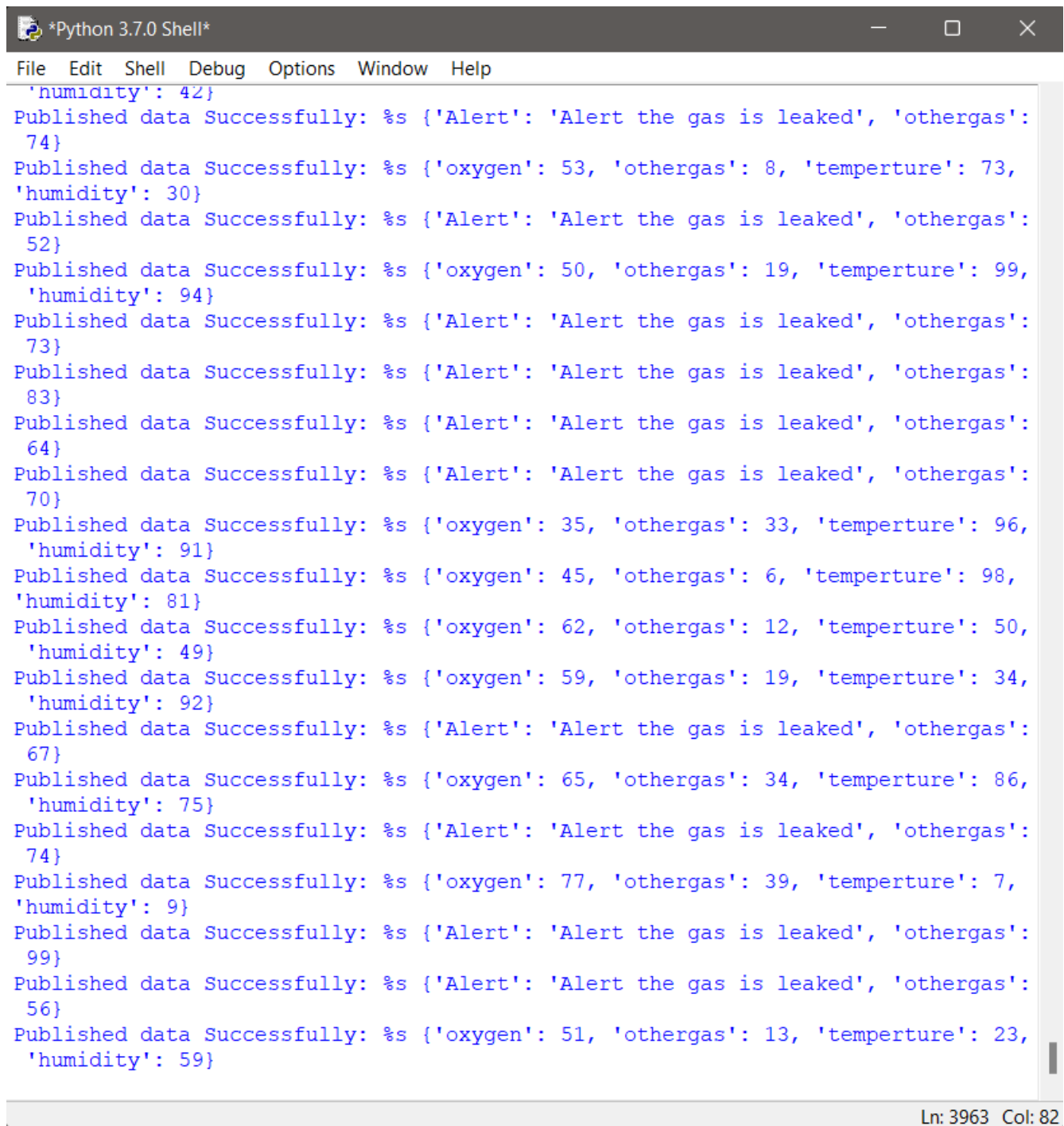
print("Published data Successfully: %s", myData)

client.commandCallback = myCommandCallback

time.sleep(5)

client.disconnect()
```

## Output:



```
*Python 3.7.0 Shell*
File Edit Shell Debug Options Window Help
'humidity': 42}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
74}
Published data Successfully: %s {'oxygen': 53, 'othergas': 8, 'temperture': 73,
'humidity': 30}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
52}
Published data Successfully: %s {'oxygen': 50, 'othergas': 19, 'temperture': 99,
'humidity': 94}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
73}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
83}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
64}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
70}
Published data Successfully: %s {'oxygen': 35, 'othergas': 33, 'temperture': 96,
'humidity': 91}
Published data Successfully: %s {'oxygen': 45, 'othergas': 6, 'temperture': 98,
'humidity': 81}
Published data Successfully: %s {'oxygen': 62, 'othergas': 12, 'temperture': 50,
'humidity': 49}
Published data Successfully: %s {'oxygen': 59, 'othergas': 19, 'temperture': 34,
'humidity': 92}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
67}
Published data Successfully: %s {'oxygen': 65, 'othergas': 34, 'temperture': 86,
'humidity': 75}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
74}
Published data Successfully: %s {'oxygen': 77, 'othergas': 39, 'temperture': 7,
'humidity': 9}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
99}
Published data Successfully: %s {'Alert': 'Alert the gas is leaked', 'othergas':
56}
Published data Successfully: %s {'oxygen': 51, 'othergas': 13, 'temperture': 23,
'humidity': 59}
Ln: 3963 Col: 82
```

# IBM Cloud:

The screenshot shows the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains various icons for navigation. The main content area displays details for a device named 'print1', which is currently 'Disconnected'. The device was last seen on 'Nov 4, 2022 10:52 AM'. Below this, there are tabs for 'Identity', 'Device Information', 'Recent Events', 'State', and 'Logs'. The 'Recent Events' tab is selected, showing a list of events from a 'Gas Sensor'. A message states: '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	{ "oxygen":86,"othergas":8,"temperture":37,"hum...	json	a few seconds ago
Gas Sensor	{ "oxygen":36,"othergas":46,"temperture":67,"hu...	json	a few seconds ago
Gas Sensor	{ "Alert":"Alert the gas is leaked","othergas":84}	json	a few seconds ago
Gas Sensor	{ "Alert":"Alert the gas is leaked","othergas":71}	json	a few seconds ago
Gas Sensor	{ "Alert":"Alert the gas is leaked","othergas":77}	json	a few seconds ago

Below the table, another device entry 'sprint-1' is partially visible, also 'Disconnected' and last seen on 'Nov 3, 2022 1:56 PM'. The bottom of the interface shows 'Items per page 50' and '1-3 of 3 items'.

# Node-Red Receiver:

The screenshot shows the Node-RED web interface in a browser. The address bar indicates the URL: 'node-red-haefn-2022-11-07.eu-gb.mybluemix.net/red/#flow/7ecd4d8bf00659c'. The interface includes a top bar with various tabs and a 'Deploy' button. The main workspace shows a flow with two nodes: an 'IBM IoT' node (blue) and a 'msg.payload' node (green), connected by an orange line. The 'debug' console on the right displays a log of messages received from the IoT device, including JSON payloads with sensor data and alert messages like 'Alert: "Alert the gas is leaked", othergas: 55'.