

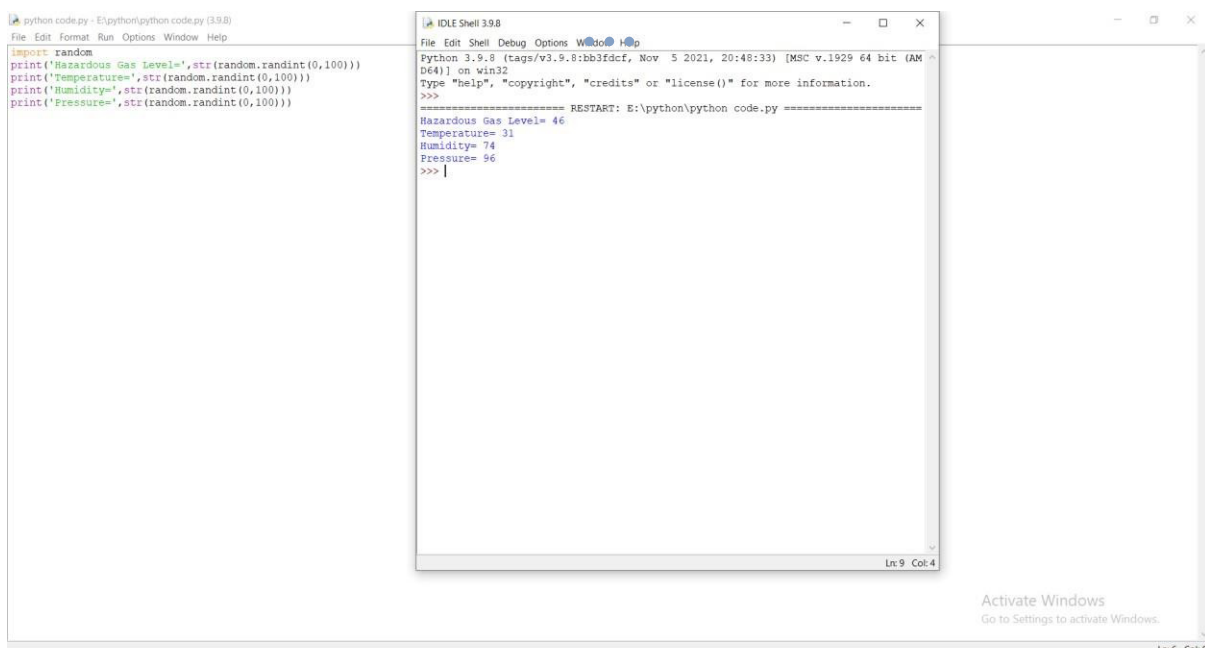
DEVELOPE A PYTHON CODE TO IBM CLOUD:

| | |
|--------------|---|
| Team ID | PNT2022TMID40877 |
| Project Name | GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES |

PYTHON CODE

```
import random
print('Hazardous Gas Level=',str(random.randint(0,100)))
print('Temperature=',str(random.randint(0,100)))
print('Humidity=',str(random.randint(0,100)))
print('Pressure=',str(random.randint(0,100)))
```

OUTPUT:

The image shows a screenshot of a computer screen with two windows. The background window is the IDLE Python IDE, showing a file named 'python code.py' with the following code:

```
import random
print('Hazardous Gas Level=',str(random.randint(0,100)))
print('Temperature=',str(random.randint(0,100)))
print('Humidity=',str(random.randint(0,100)))
print('Pressure=',str(random.randint(0,100)))
```

 The foreground window is the IDLE Shell 3.9.8, which displays the output of the code execution:

```
Python 3.9.8 (tags/v3.9.8:bb3fddf, Nov 5 2021, 20:48:33) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: E:\python\python code.py =====
Hazardous Gas Level= 46
Temperature= 31
Humidity= 74
Pressure= 96
>>> |
```

 At the bottom right of the screen, there is a watermark that says 'Activate Windows Go to Settings to activate Windows.'

```

#IBM Watson IOT
Platform #pip install
wiotp-sdk import
wiotp.sdk.device import
time
import
random
myConfig = {
"identity": {
    "orgId": "knk8wp ",
    "typeId": "lok ",
    "deviceId": "123"
},
"auth": {
    "token": "HNTDPRX@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()

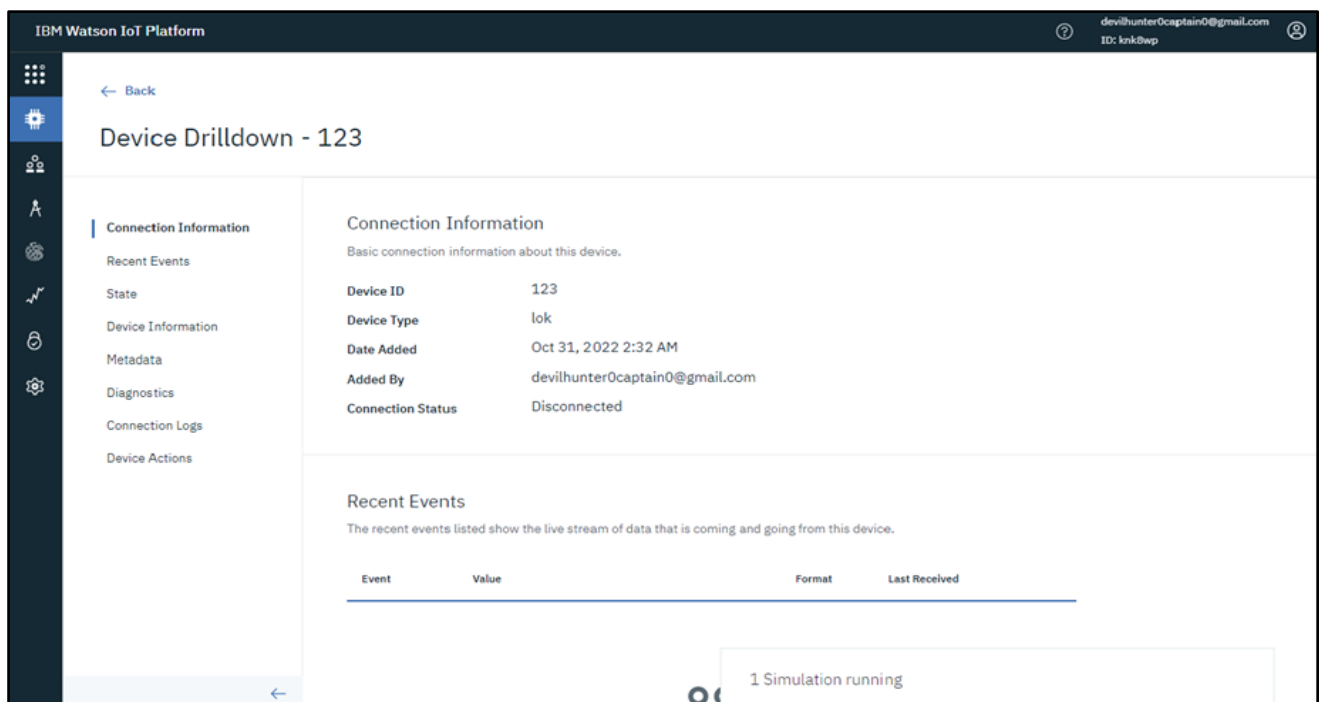
while True: gas=random.randint(

```

```

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 = myCommandCallback
time.sleep(2)
client.disconnect()

```



The screenshot displays the ThingsBoard web application. The main window shows the 'Recent Events' tab for a device named 'demo'. A table lists five recent events, each containing sensor data like Hazardous Gas, Temperature, Humidity, and Pressure. An overlay window titled 'Event Editor' is open, allowing configuration of a new event type named 'event_1'. It includes fields for schedule (set to 'Every Minute') and payload (JSON format with random values).

| 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 |

```

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