SPRINT -1

GAS LEAKAGE MONITORING AND ALERTING SYSTEM

Team ID	PNT2022TMID30401
Project Name	Gas Leakage Monitoring and Alerting System for Industries

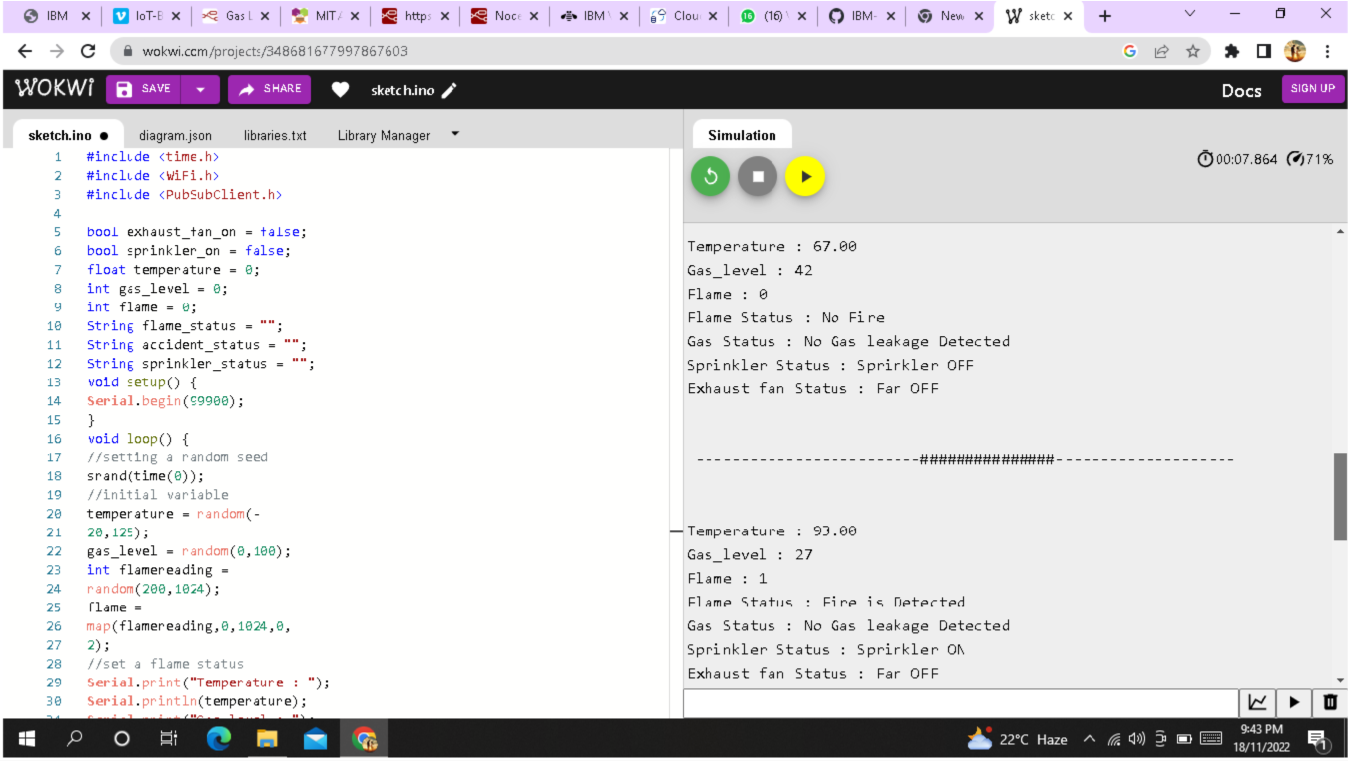
SIMULATION CREACTION USING WOKWI:

CODE:

```
#include  <time.h>
#include <WiFi.h>
#include <PubSubClient.h>
bool exhaust_fan_on = false;
bool sprinkler_on = false;
float temperature = 0;
int gas_level = 0;
int flame = 0;
String flame_status = "";
String accident_status = "";
String sprinkler_status = "";
void setup() {
Serial.begin(99900);
}
void loop() {
//setting a random seed
srand(time(0));
//initial variable
temperature = random(-
20,125);
gas_level = random(0,1000);
int flamereading =
random(200,1024);
flame =
map(flamereading,0,1024,0,
2);
//set a flame status
Serial.print("Temperature : ");
Serial.println(temperature);
Serial.print("Gas_level : ");
Serial.println(gas_level);
Serial.print("Flame : ");
Serial.println(flame);
switch (flame) {
case 0:
```

```
flame_status = "No Fire";
Serial.println("Flame Status : "+flame_status);
break;
case 1:
flame_status = "Fire is Detected";
Serial.println("Flame Status : "+flame_status);
break;
}
//Gas Detection
if(gas_level > 100){
Serial.println("Gas Status : Gas leakage Detected");
}
else{
exhaust_fan_on = false;
Serial.println("Gas Status : No Gas leakage Detected");
}
//send the sprinkler status
if(flame){
sprinkler_status =
"Sprinkler ON";
Serial.println("Sprinkler Status : "+sprinkler_status);
}
else{
sprinkler_status = "Sprinkler OFF";
Serial.println("Sprinkler Status : "+sprinkler_status);
}
//toggle the fan according to gas
if(gas_level > 100){
exhaust_fan_on = true;
Serial.println("Exhaust fan Status : Fan ON");
}
else{
exhaust_fan_on = false;
Serial.println("Exhaust fan Status : Fan OFF");
}
Serial.println("");
Serial.println("");
Serial.println(" ------------------------###############------------------- ");
Serial.println("");
Serial.println("");
delay(1000);
}
```

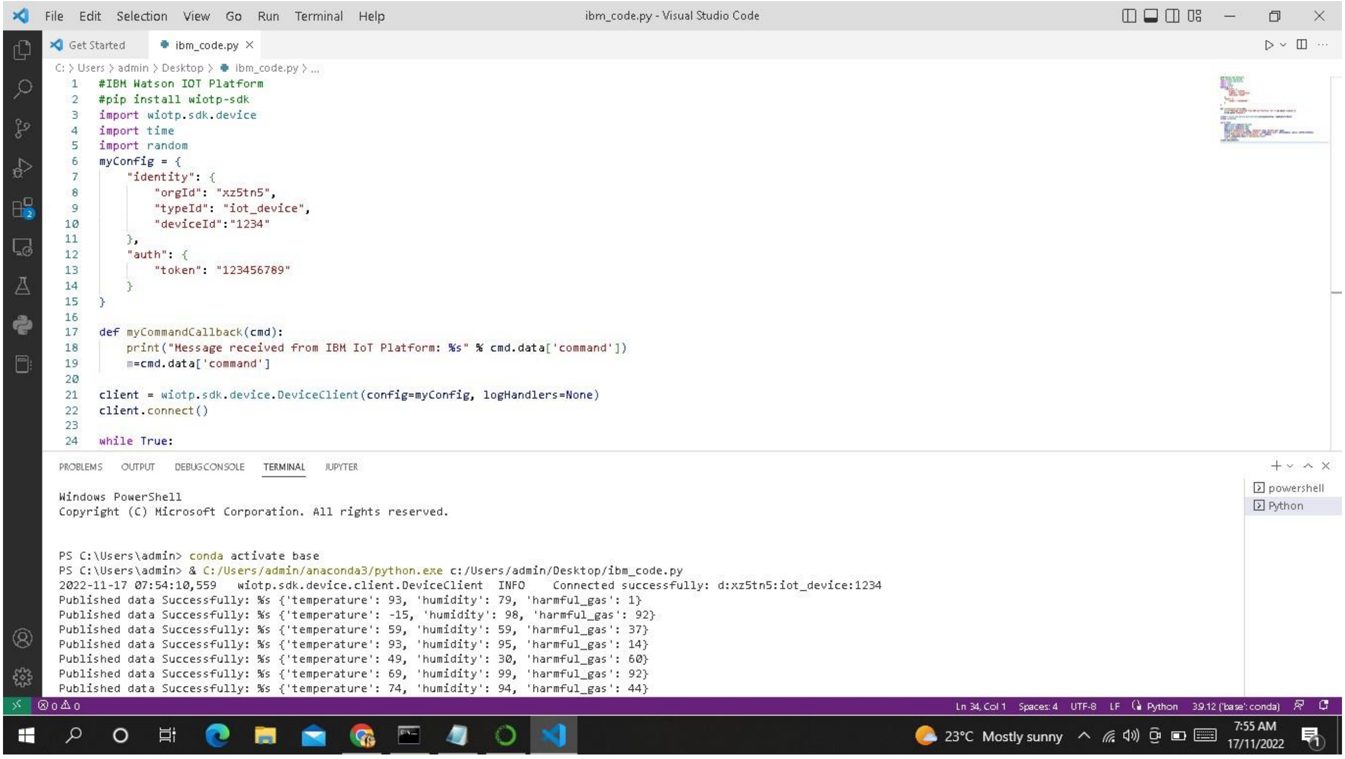
SIMULATION OUTPUT:



```
#include <time.h>
#include <WiFi.h>
#include <PubSubClient.h>

bool exhaust_fan_on = false;
bool sprinkler_on = false;
float temperature = 0;
int gas_level = 0;
int flame = 0;
String flame_status = "";
String accident_status = "";
String sprinkler_status = "";
void setup() {
Serial.begin(99900);
}
void loop() {
//setting a random seed
srand(time(0));
//initial variable
temperature = random(-
20,125);
gas_level = random(0,100);
int flamereading =
random(200,1024);
flame =
map(flamereading,0,1024,0,
2);
//set a flame status
Serial.print("Temperature : ");
Serial.println(temperature);
```

```
Temperature : 67.00
Gas_level : 42
Flame : 0
Flame Status : No Fire
Gas Status : No Gas leakage Detected
Sprinkler Status : Sprinkler OFF
Exhaust fan Status : Fan OFF


 ------------------------################--------------------


Temperature : 93.00
Gas_level : 27
Flame : 1
Flame Status : Fire is Detected
Gas Status : No Gas leakage Detected
Sprinkler Status : Sprinkler ON
Exhaust fan Status : Fan OFF
```

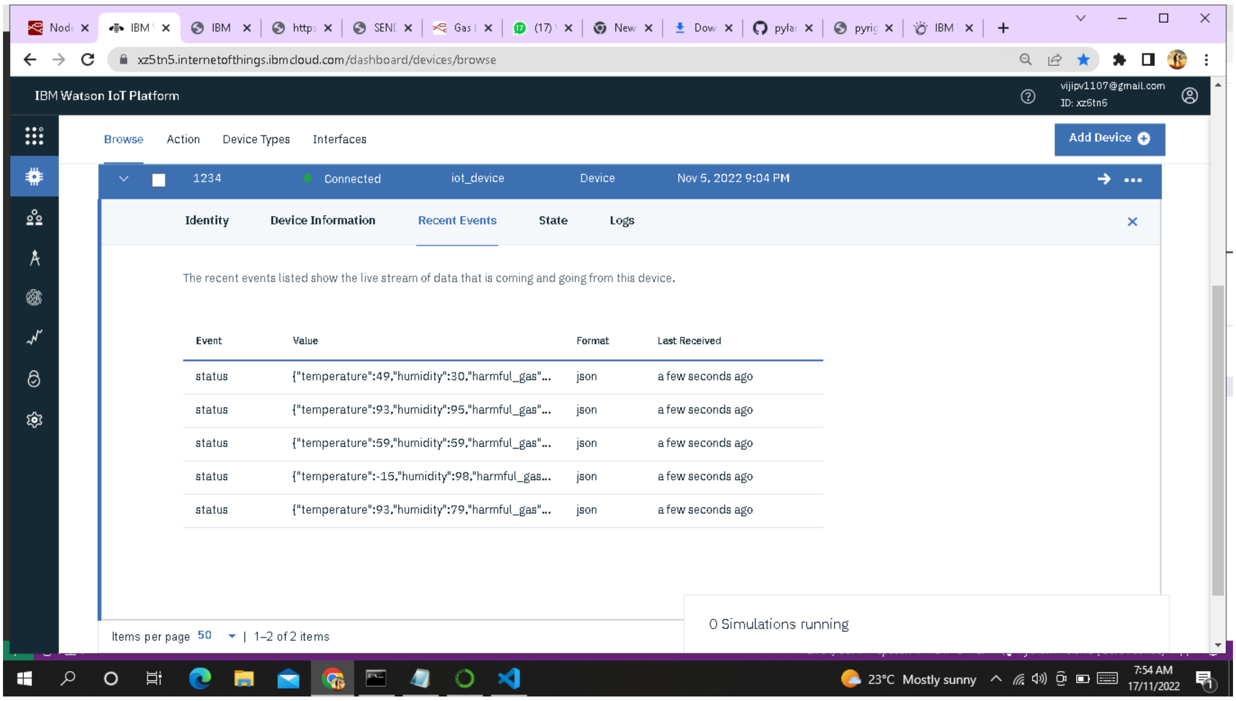
CONNECTING IBM CLOUD USING PYTHON CODE:
CODE:



```python
#IBM Watson IOT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "xz5tn5",
        "typeId": "iot_device",
        "deviceId":"1234"
    },
    "auth": {
        "token": "123456789"
    }
}

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:
```

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.


PS C:\Users\admin> conda activate base
PS C:\Users\admin> & C:/Users/admin/anaconda3/python.exe c:/Users/admin/Desktop/ibm_code.py
2022-11-17 07:54:10,559   wiotp.sdk.device.client.DeviceClient  INFO    Connected successfully: d:xz5tn5:iot_device:1234
Published data Successfully: %s {'temperature': 93, 'humidity': 79, 'harmful_gas': 1}
Published data Successfully: %s {'temperature': -15, 'humidity': 98, 'harmful_gas': 92}
Published data Successfully: %s {'temperature': 59, 'humidity': 59, 'harmful_gas': 37}
Published data Successfully: %s {'temperature': 93, 'humidity': 95, 'harmful_gas': 14}
Published data Successfully: %s {'temperature': 49, 'humidity': 30, 'harmful_gas': 60}
Published data Successfully: %s {'temperature': 69, 'humidity': 99, 'harmful_gas': 92}
Published data Successfully: %s {'temperature': 74, 'humidity': 94, 'harmful_gas': 44}
```

<u>OUTPUT IN IBM CLOUD:</u>



The recent events listed show the live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
status	{"temperature":49,"humidity":30,"harmful_gas"...	json	a few seconds ago
status	{"temperature":93,"humidity":95,"harmful_gas"...	json	a few seconds ago
status	{"temperature":59,"humidity":59,"harmful_gas"...	json	a few seconds ago
status	{"temperature":-15,"humidity":98,"harmful_gas...	json	a few seconds ago
status	{"temperature":93,"humidity":79,"harmful_gas"...	json	a few seconds ago