

## PROJECT DEVELOPMENT PHASE

### SPRINT-1

Team ID	PNT2022TMID01775
Project Name	Gas Leakage Monitoring And Alerting System

#### PYTHON CODE:

```
import time

import sys

import ibmiotf.application

import ibmiotf.device

import random

#Provide your IBM Watson Device Credentials

organization = "d19wub"

deviceType = "Arduino"

deviceId = "1234"

authMethod = "token"

authToken = "cfspzFCmWpFlaA*aWR"

# Initialize GPIO

def myCommandCallback(cmd):

    print("Command received: %s" % cmd.data['command'])

    status=cmd.data['command']

    if status=="lighton":

        print ("led is on")

    elif status == "lightoff":

        print ("led is off")

    else :

        print ("please send proper command")

    try:

        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth-token": authToken}

        deviceCli = ibmiotf.device.Client(deviceOptions)

        #.....

    except Exception as e:

        print("Caught exception connecting : %s" % str(e))

    sys.exit()

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times

deviceCli.connect()
```

```

while True:

#Get Sensor Data from DHT11

temp=random.randint(90,110)

Humid=random.randint(60,100)

data = { 'temp' : temp, 'Humid': Humid }

#print data

def myOnPublishCallback():

print ("Published Temperature = %s C" % temp, "Humidity = %s %" % Humid, "to IBM Watson") success =
deviceCli.publishEvent("IoTSensor", "json", data, qos=0,

on_publish=myOnPublishCallback)

if not success:

print("Not connected to IoT")

time.sleep(10)

deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud

deviceCli.disconnect()

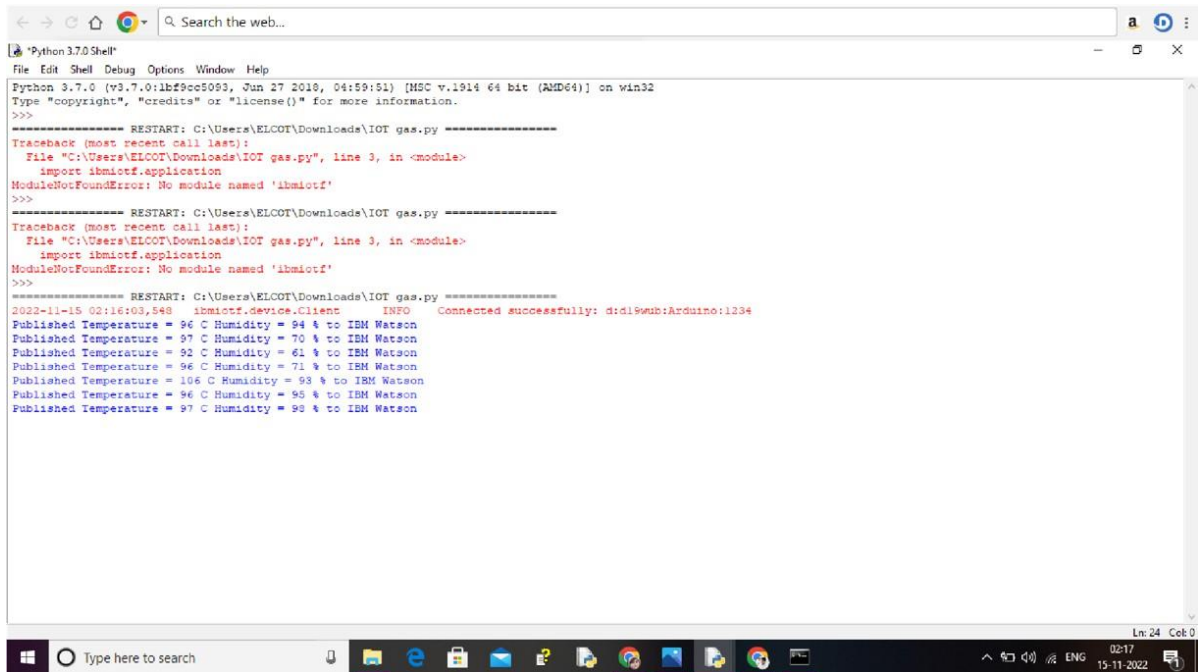
```

## OUTPUT:

The screenshot displays the IBM Watson IoT Platform interface. At the top, there's a navigation bar with 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains various icons for navigation. The main content area shows a device card for '1234' (Arduino) with a 'Connected' status and a timestamp of '15 Nov 2022 01:35'. Below this, a 'Showing Raw Data' section is visible, indicating 'No Interfaces Available'. A table displays the following data:

Property	Value	Type	Event	Last Received
temp	97	Number	IoT Sensor	a few seconds ago
Humid	100	Number	IoT Sensor	a few seconds ago

At the bottom of the dashboard, it indicates '0 Simulations running'. The Windows taskbar is visible at the very bottom of the image.



```
Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Python 3.7.0 (tags/v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\ELCOT\Downloads\IOT gas.py =====
Traceback (most recent call last):
  File "C:\Users\ELCOT\Downloads\IOT gas.py", line 3, in <module>
    import ibmiotf.application
ModuleNotFoundError: No module named 'ibmiotf'
>>>
===== RESTART: C:\Users\ELCOT\Downloads\IOT gas.py =====
Traceback (most recent call last):
  File "C:\Users\ELCOT\Downloads\IOT gas.py", line 3, in <module>
    import ibmiotf.application
ModuleNotFoundError: No module named 'ibmiotf'
>>>
===== RESTART: C:\Users\ELCOT\Downloads\IOT gas.py =====
2022-11-15 02:16:03.548 ibmiotf.device.Client INFO Connected successfully: did9web:Arduino:1234
Published Temperature = 96 C Humidity = 94 % to IBM Watson
Published Temperature = 97 C Humidity = 70 % to IBM Watson
Published Temperature = 92 C Humidity = 61 % to IBM Watson
Published Temperature = 96 C Humidity = 71 % to IBM Watson
Published Temperature = 106 C Humidity = 93 % to IBM Watson
Published Temperature = 96 C Humidity = 95 % to IBM Watson
Published Temperature = 97 C Humidity = 95 % to IBM Watson
```

## RESULT:

The sensor Arduino with python code is connected successfully.