# DEVELOP THE PYTHON CODE

| | |
|---|---|
| Date | 22October2022 |
| Team ID | PNT2022TMID4963 |
| Project name | Gas leakage monitoring & Alerting System for Industries |

This is the python code we have developed and used for programming our webpage and mobile application.

```
#TEAM ID=PNT2022TMID4963
#pip install wiotp-sdk
import ibmiotf.application
import ibmiotf.device
import time
import sys
import random


#Provide your IBM Watson Device Credentials
organization = "pox0qm"
deviceType = "123456"
deviceId = "4699"
authMethod = "token"
authToken = "12345678"

# 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 device: %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

        temperature=random.randint(0,100)
        Humidity=random.randint(0,100)
        gas=random.randint(0,100)

        data = { 'temperature' : temperature, 'Humid': Humidity, 'Gas':gas }

        #print data
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temperature, "Humidity = %s
%%" % Humidity, "Gas Concentration = %s" % gas," to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(10)

        deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()
```