

PROJECT NAME	GAS LEAKAGE MONITORING & ALERTING SYSTEM FOR INDUSTRIES
TEAM ID	PNT2022TMID16564
TEAM MEMBERS	SANTHOSH S,SARAN S,PRATHAP A,SUNIL KUMAR K

## Code:

```
import time
```

```
import sys
```

```
import ibmiotf.application
```

```
import ibmiotf.device
```

```
import random
```

```
#Provide your IBM Watson Device Credentials
```

```
organization = "ioz5i8"
```

```
deviceType = "raspberrypi"
```

```
deviceId = "123456"
```

```
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**

**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 IoTF")
```

```
time.sleep(10)
```

```
deviceCli.commandCallback = myCommandCallback
```

# Disconnect the device and application from the cloud

```
deviceCli.disconnect()
```

The image shows a screenshot of a Windows desktop with two windows open. The left window is a text editor titled 'sample.py - C:\Users\ELCOT\Desktop\sample.py (3.7.0)'. It contains a Python script that imports modules like time, sys, ibmiotf, and random. The script sets up IBM Watson IoT credentials, initializes GPIO, and defines a command callback function. It then connects to the cloud and enters a loop where it publishes temperature and humidity data every 10 seconds. The right window is a 'Python 3.7.0 Shell' terminal showing the output of the script. It displays a series of log messages indicating the command received, the status of the LED, and the published data points (Temperature and Humidity) to IBM Watson. The terminal output shows that the device successfully connects and publishes data.

```
sample.py - C:\Users\ELCOT\Desktop\sample.py (3.7.0)
File Edit Format Run Options Window Help

import time
import sys
import ibmiotf.application
import ibmiotf.device
import random

#Provide your IBM Watson Device Credentials
organization = "10z518"
deviceType = "raspberrypi"
deviceId = "123456"
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
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 e
deviceCli.connect()

while True:
    #Get Sensor Data from DHT11
    temp=random.randint(90,110)

Python 3.7.0 Shell
File Edit Shell Debug Options Window Help

Published Temperature = 91 C Humidity = 65 % to IBM Watson
Published Temperature = 103 C Humidity = 78 % to IBM Watson
Published Temperature = 104 C Humidity = 90 % to IBM Watson
Published Temperature = 96 C Humidity = 61 % to IBM Watson
Published Temperature = 100 C Humidity = 90 % to IBM Watson
Published Temperature = 105 C Humidity = 63 % to IBM Watson
Published Temperature = 105 C Humidity = 79 % to IBM Watson
Published Temperature = 109 C Humidity = 71 % to IBM Watson
Published Temperature = 94 C Humidity = 98 % to IBM Watson
Published Temperature = 97 C Humidity = 97 % to IBM Watson
Published Temperature = 109 C Humidity = 93 % to IBM Watson
Published Temperature = 104 C Humidity = 99 % to IBM Watson
Published Temperature = 90 C Humidity = 72 % to IBM Watson
Published Temperature = 96 C Humidity = 94 % to IBM Watson
Published Temperature = 95 C Humidity = 77 % to IBM Watson
Published Temperature = 105 C Humidity = 70 % to IBM Watson
Published Temperature = 98 C Humidity = 73 % to IBM Watson
Published Temperature = 103 C Humidity = 76 % to IBM Watson
Published Temperature = 95 C Humidity = 60 % to IBM Watson
Published Temperature = 102 C Humidity = 95 % to IBM Watson
Published Temperature = 105 C Humidity = 73 % to IBM Watson
Published Temperature = 103 C Humidity = 63 % to IBM Watson
Published Temperature = 105 C Humidity = 80 % to IBM Watson
Published Temperature = 94 C Humidity = 68 % to IBM Watson
Published Temperature = 106 C Humidity = 87 % to IBM Watson
Published Temperature = 109 C Humidity = 67 % to IBM Watson
Published Temperature = 99 C Humidity = 92 % to IBM Watson
Published Temperature = 105 C Humidity = 62 % to IBM Watson
Published Temperature = 102 C Humidity = 98 % to IBM Watson
Published Temperature = 110 C Humidity = 86 % to IBM Watson
Published Temperature = 92 C Humidity = 85 % to IBM Watson
Published Temperature = 96 C Humidity = 66 % to IBM Watson
Published Temperature = 101 C Humidity = 84 % to IBM Watson
Published Temperature = 94 C Humidity = 60 % to IBM Watson
Published Temperature = 99 C Humidity = 66 % to IBM Watson
Published Temperature = 109 C Humidity = 72 % to IBM Watson
Published Temperature = 91 C Humidity = 80 % to IBM Watson
Published Temperature = 103 C Humidity = 67 % to IBM Watson
Published Temperature = 95 C Humidity = 70 % to IBM Watson
```

IBM-EPBLService DeIBM WatsNode-REDNode-RED169.51.19MIT AppDownload

ioz5i8.internetofthings.ibmcloud.com/dashboard/devices/browse

IBM Watson IoT Platformsanthosh.santhosh8667@gmail.comID: ioz5i8

BrowseActionDevice TypesInterfacesAdd Device

123456DisconnectedraspberrypiDeviceNov 10, 2022 11:54 AM

IdentityDevice InformationRecent EventsStateLogs

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

Event	Value	Format	Last Received
IoTSensor	{"temp":103,"Humid":67}	json	a few seconds ago
IoTSensor	{"temp":91,"Humid":80}	json	a minute ago
IoTSensor	{"temp":109,"Humid":72}	json	a minute ago
IoTSensor	{"temp":99,"Humid":66}	json	a minute ago
IoTSensor	{"temp":94,"Humid":60}	json	a minute ago

1 Simulation running

Type here to search30°C HazeENG IN1:17 PM11/20/2022