

Team ID	PNT2022TMID08724
Project Name	Signs with Smart Connectivity for Better Road Safety

```

import wiotp.sdk.device
import time
import random
import ibmiotf.application
import ibmiotf.device
import requests, json
myConfig = {
#Configuration
"identity": {
"orgId": "aoi7bz",
"typeId": "ESP32",
"deviceId": "12345"
},
#API Key
"auth": {
"token": "12345678"
}
}
#Receiving callbacks from IBM IOT platform
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()
#OpenWeatherMap Credentials
BASE_URL = "https://api.openweathermap.org/data/2.5/weather?"
CITY ="Coimbatore, IN"
URL = BASE_URL + "q=" + CITY + "&units=metric"&appid=" +
"f58e4720c739a54c439aba9b05176839"
while True:
    response = requests.get(URL)
    if response.status_code == 200:
        data =response.json()
        main = data['main']
        temperature= main['temp']
        humidity =main['humidity']
        pressure =main['pressure']
        report =data['visibility']
#messge part
    msg=random.randint(0,5)

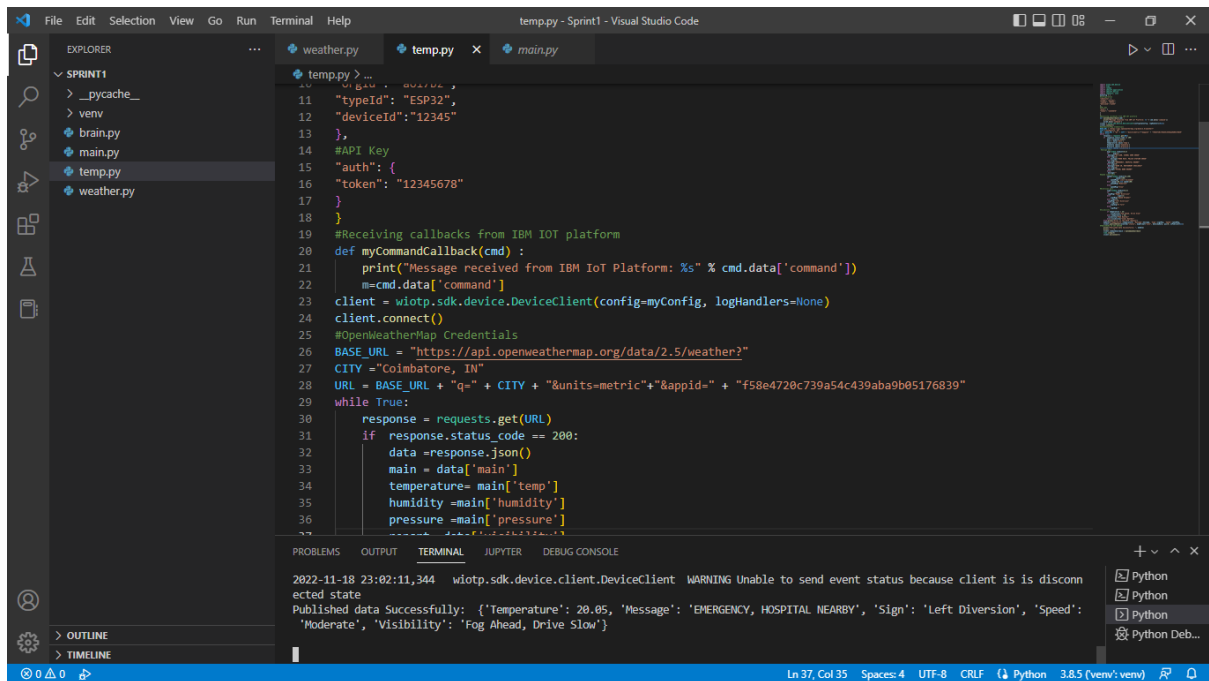
```

```

        if msg==1:
            message="GO SLOW, SCHOOL ZONE AHEAD"
        elif msg==2:
            message="NEED HELP, POLICE STATION AHEAD"
        elif msg==3:
            message="EMERGENCY, HOSPITAL NEARBY"
        elif msg==4:
            message="DINE IN, RESTAURANT AVAILABLE"
        elif msg==5:
            message="PETROL BUNK NEARBY"
        else:
            message=""
#Speed Limit part
        speed=random.randint(0,150)
        if speed>=100:
            speedMsg=" Limit Exceeded"
        elif speed>=60 and speed<100:
            speedMsg="Moderate"
        else:
            speedMsg="Slow"
#Diversion part
        sign=random.randint(0,5)
        if sign==1:
            signMsg="Right Diversion"
        elif sign==2:
            signMsg="Speed Breaker"
        elif sign==3:
            signMsg="Left Diversion"
        elif sign==4:
            signMsg="U Turn"
        else:
            signMsg=""
#Visibility
        if temperature < 24:
            visibility="Fog Ahead, Drive Slow"
        elif temperature < 20:
            visibility="Bad Weather"
        else:visibility="Clear Weather"
        else:print("Error in the HTTP request")
        myData={'Temperature':temperature, 'Message':message, 'Sign':signMsg,
'Speed':speedMsg,
'Visibility':visibility}
        client.publishEvent(eventId="status", msgFormat="json", data=myData,
qos=0, onPublish=None)
#PUBLISHING TO IOT WATSON
        print("Published data Successfully: ", myData)
        print("....")
        client.commandCallback = myCommandCallback

```

```
time.sleep(5)
client.disconnect()
```



The screenshot shows the Visual Studio Code interface with a Python file named `temp.py` open. The code defines a `DeviceClient` for IBM IoT, sets up API credentials, and uses a `while` loop to fetch weather data from OpenWeatherMap. The terminal output shows a warning about the client disconnecting and a successful JSON response containing temperature, message, sign, speed, and visibility data.

```
temp.py > ...
11 "typeId": "ESP32",
12 "deviceId": "12345"
13 },
14 #API Key
15 "auth": {
16 "token": "12345678"
17 }
18 }
19 #Receiving callbacks from IBM IoT platform
20 def myCommandCallback(cmd) :
21     print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
22     m=cmd.data['command']
23 client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
24 client.connect()
25 #OpenWeatherMap Credentials
26 BASE_URL = "https://api.openweathermap.org/data/2.5/weather?"
27 CITY = "Coimbatore, IN"
28 URL = BASE_URL + "q=" + CITY + "&units=metric"&"&appid=" + "f58e4720c739a54c439aba9b05176839"
29 while True:
30     response = requests.get(URL)
31     if response.status_code == 200:
32         data = response.json()
33         main = data['main']
34         temperature= main['temp']
35         humidity =main['humidity']
36         pressure =main['pressure']
37         data['temp']=temperature
38         data['humidity']=humidity
39         data['pressure']=pressure
40         data['message']=message
41         data['sign']=sign
42         data['speed']=speed
43         data['visibility']=visibility
44         print(data)
45         time.sleep(5)
46         client.disconnect()
```

2022-11-18 23:02:11,344 wiotp.sdk.device.client.DeviceClient WARNING Unable to send event status because client is is disconnected state
Published data Successfully: {'Temperature': 20.05, 'Message': 'EMERGENCY, HOSPITAL NEARBY', 'Sign': 'Left Diversion', 'Speed': 'Moderate', 'Visibility': 'Fog Ahead, Drive Slow'}

OUTPUT:

```
{
"Temperature": 20.05,
"Message": "",
"Sign": "Left Diversion",
"Speed": "Slow",
"Visibility": "Fog Ahead, Drive Slow"
}
```