

## PYTHON CODE TO PUBLISH DATA TO IBM CLOUD

Assignment Date	07 NOVEMBER2022
Team ID	PNT2022TMID42277
Project Name	Gas Leakage Monitoring and Alerting System

### **Python Code:**

```
#IBM Watson IOT Platform
```

```
#pip install wiotp-sdk
```

```
import wiotp.sdk.device
```

```
import time
```

```
import random
```

```
myConfig = {
```

```
    "identity": {
```

```
        "orgId": "0tus0f",
```

```
        "typeId": "ESP32",
```

```
        "deviceId": "01"
```

```
    },
```

```
    "auth": {
```

```
        "token": "Gowth@m@nk18"
```

```
    }
```

```
}
```

```
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:

```
    temp=random.randint(-20,125)
```

```
    hum=random.randint(0,100)
```

```
    myData={'temperature':temp, 'humidity':hum}
```

```
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0,
onPublish=None)
```

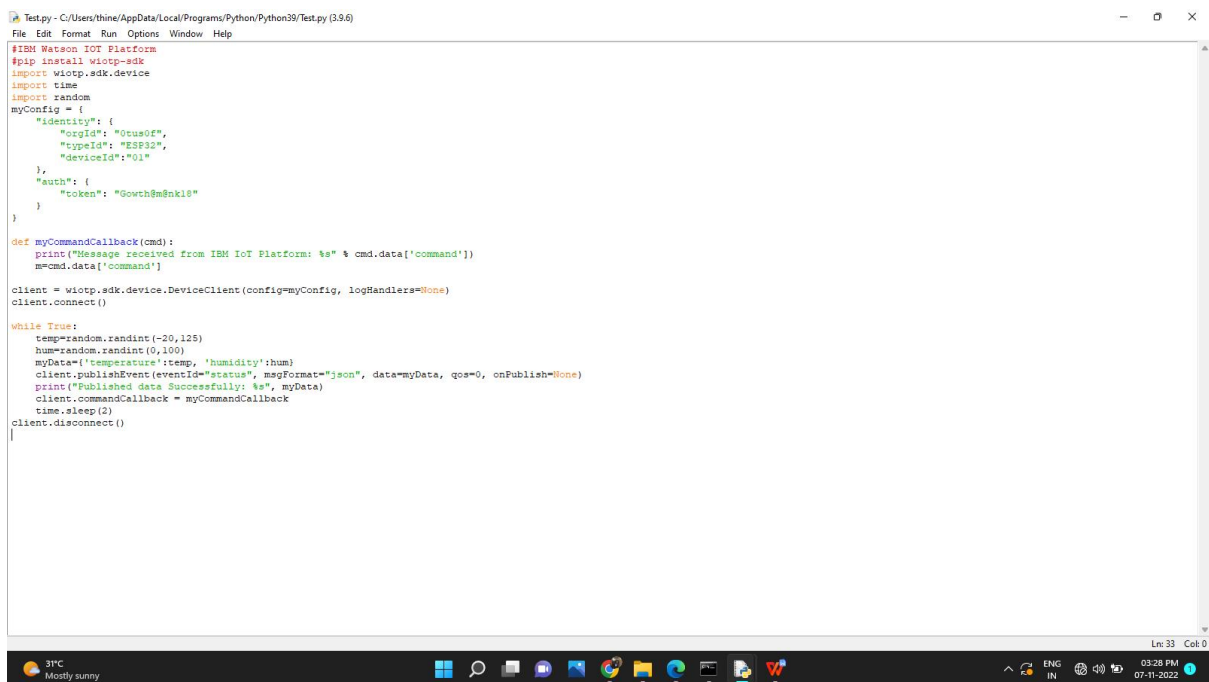
```
    print("Published data Successfully: %s", myData)
```

```
    client.commandCallback = myCommandCallback
```

```
    time.sleep(2)
```

```
client.disconnect()
```

## PYTHON CODE:



```
Test.py - C:/Users/thine/AppData/Local/Programs/Python/Python39/Test.py (3.9.6)
File Edit Format Run Options Window Help

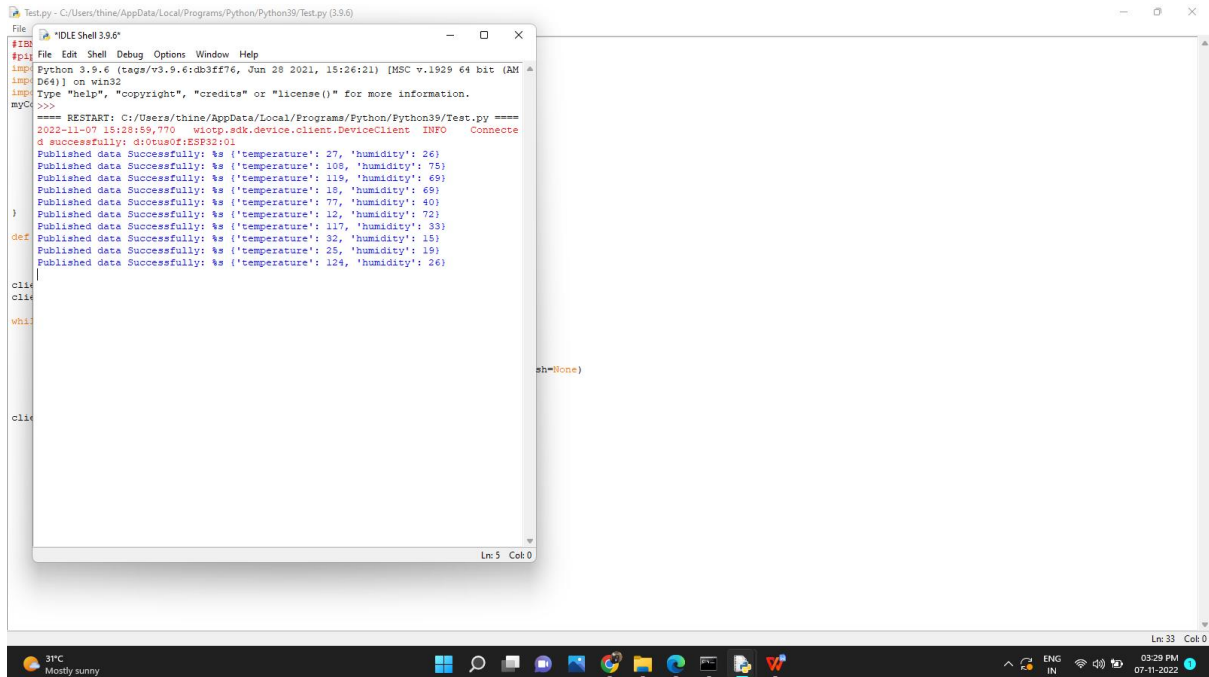
#IBM Watson IoT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "0tus06f",
        "typeId": "ESP32",
        "deviceId": "01"
    },
    "auth": {
        "token": "Gowth$@nkl8"
    }
}

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:
    temp=random.randint(-20,125)
    hum=random.randint(0,100)
    myData={'temperature':temp, 'humidity':hum}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(2)
client.disconnect()
```

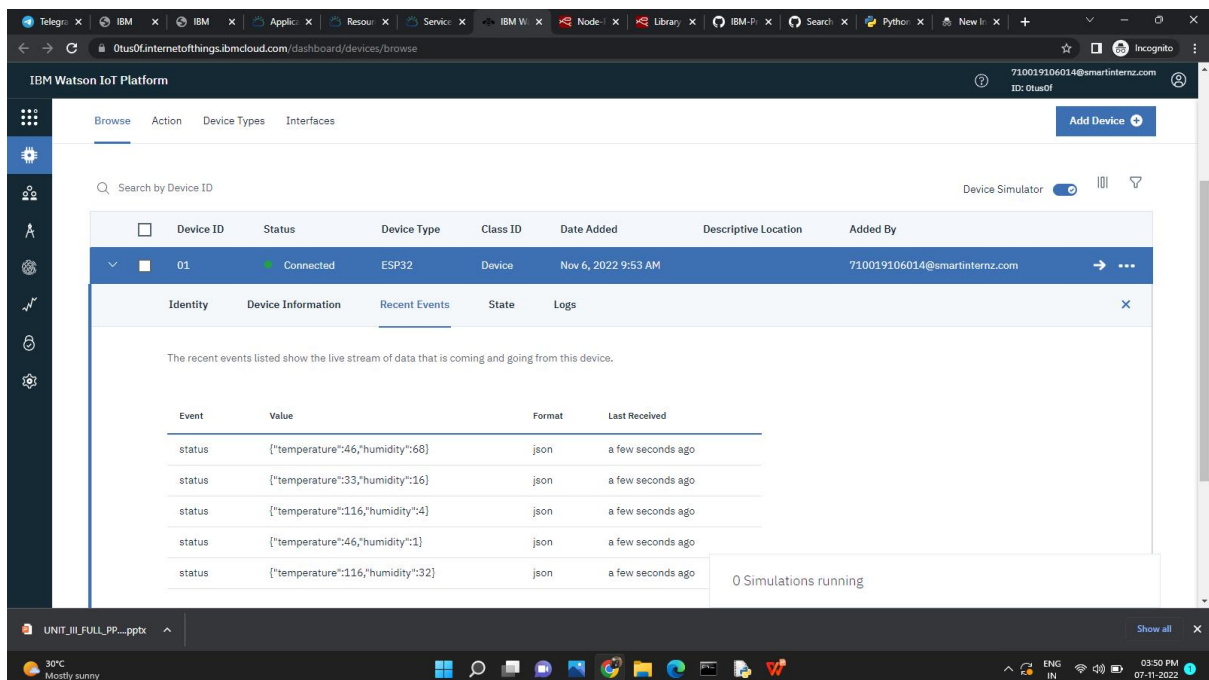
## PYTHON OUTPUT:



The screenshot shows a Python IDE window titled "Test.py - C:/Users/thine/AppData/Local/Programs/Python/Python39/Test.py (3.9.6)". The code is a simple test script that prints the output of a device client. The output shows a successful connection and several data points being published successfully.

```
File Edit Shell Debug Options Window Help
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:/Users/thine/AppData/Local/Programs/Python/Python39/Test.py =====
2022-11-07 15:28:59,770 wiotp.sdk.device.client.DeviceClient INFO Connecte
d successfully: d:OtusOf:ESP32:01
Published data Successfully: %s ('temperature': 27, 'humidity': 26)
Published data Successfully: %s ('temperature': 108, 'humidity': 75)
Published data Successfully: %s ('temperature': 119, 'humidity': 69)
Published data Successfully: %s ('temperature': 18, 'humidity': 69)
Published data Successfully: %s ('temperature': 77, 'humidity': 40)
Published data Successfully: %s ('temperature': 12, 'humidity': 72)
Published data Successfully: %s ('temperature': 117, 'humidity': 33)
Published data Successfully: %s ('temperature': 32, 'humidity': 15)
Published data Successfully: %s ('temperature': 25, 'humidity': 19)
Published data Successfully: %s ('temperature': 124, 'humidity': 26)
```

## Watson Cloud IBM:



The screenshot shows the IBM Watson IoT Platform dashboard. The main view is the "Recent Events" tab for a device with ID 01. The device is connected and is an ESP32. The recent events listed show a live stream of data that is coming and going from this device.

Event	Value	Format	Last Received
status	["temperature":46,"humidity":68]	json	a few seconds ago
status	["temperature":33,"humidity":16]	json	a few seconds ago
status	["temperature":116,"humidity":4]	json	a few seconds ago
status	["temperature":46,"humidity":1]	json	a few seconds ago
status	["temperature":116,"humidity":32]	json	a few seconds ago

0 Simulations running