

Develop the Python Script

(Publish data to IBM cloud)

Date	08 November 2022
Team ID	PNT2022TMID09975
Project Name	Industry-specific intelligent fire management system

Industry-specific intelligent fire management system

The screenshot shows a Python script in a text editor and its execution output in a terminal window. The script is titled "publish.py" and is located at "E:\IBM\Others\Develop a python script\publish.py (3.6.5)". The script imports the paho.mqtt.client module, time, and random. It defines a function on_publish that prints "Publish the data". The main code creates a paho.Client, sets on_publish as the callback, connects to "broker.mqttdashboard.com" on port 1883, and starts the loop. A while True loop generates random data (temp) and publishes it to the "iottopic" with qos=1, sleeping 10 seconds between publishes.

```
#Through python coding we are going to access the subscriber
import paho.mqtt.client as paho
import time
import random

def on_publish(client, userdata, mid):
    print("Publish the data ")

client = paho.Client()
client.on_publish = on_publish
client.connect('broker.mqttdashboard.com', 1883)
client.loop_start()
while True:
    temp = random.randint(1,30)
    (re,mid) = client.publish('iottopic',str(temp),qos=1)
    print(temp)
    time.sleep(10)
```

The terminal window shows the output of the script, which is "Publish the data" repeated multiple times. The terminal also shows the restart command: "RESTART: E:\IBM\Others\Develop a python script\publish.py".

The screenshot shows a Python script in a text editor and its execution output in a terminal window. The script is titled "subscribe.py" and is located at "E:\IBM\Others\Develop a python script\subscribe.py (3.6.5)". The script imports the paho.mqtt.client module. It defines a function on_subscribe that prints "subscriber:" followed by the mid and granted_qos. It also defines a function on_message that prints the topic, qos, and payload. The main code creates a paho.Client, sets on_subscribe and on_message as the callbacks, connects to "broker.mqttdashboard.com" on port 1883, subscribes to the "iottopic" with qos=1, and starts the loop forever.

```
import paho.mqtt.client as paho
def on_subscribe(client,userdata,mid,grated_qos):
    print("subscriber:" + str(mid)+str(granted_qos))

def on_message(client,userdata,msg):
    print(msg.topic + " " + str(msg.qos) + " " + str(msg.payload))

client = paho.Client()
client.on_subscribe = on_subscribe
client.on_message = on_message
client.connect('broker.mqttdashboard.com', 1883)
client.subscribe('iottopic',qos=1)
client.loop_forever()
```

The terminal window shows the output of the script, which is "Publish the data" repeated multiple times. The terminal also shows the restart command: "RESTART: E:\IBM\Others\Develop a python script\subscribe.py".

MSN IndiaIBMIBM-EPBLIBM-EPBLReset youPonni NatTraining CIBM-ProjeMy IBMService DIBM V.XMy IBM

pq685h.internetofthings.ibmcloud.com/dashboard/devices/browse

IBM Watson IoT Platformhh0109477@gmail.comID: (select org)

BrowseActionDevice TypesInterfaces

Add Device

Device ID	Status	Device Type	Class ID	Date Added	
abcd	Disconnected	123	Device	Nov 3, 2022 12:13 PM	→ ...

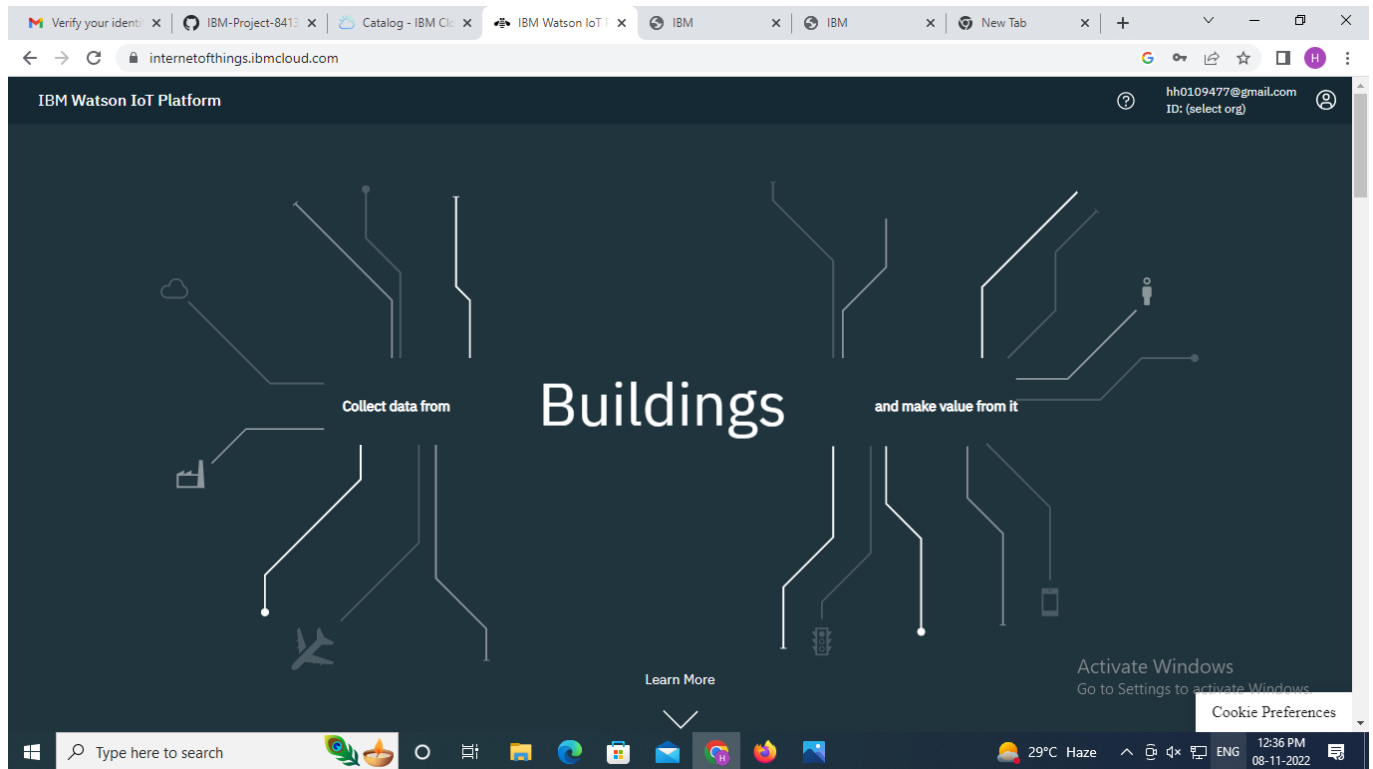
IdentityDevice InformationRecent EventsStateLogs

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

Event	Value	Format	Last Received
event_1	{"randomNumber":74}	json	a few seconds ago
event_1	{"randomNumber":47}	json	a few seconds ago
event_1	{"randomNumber":45}	json	a minute ago
event_1	{"randomNumber":19}	json	a minute ago
event_1	{"randomNumber":79}	json	a minute ago

1 Simulation running

Type here to searchIBM ...Scree...IBM ...Node...What...node...IBM C...ENG00:1004-11-2022



Program :

```
#IBM Watson IOT Platform
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random

myConfig = {"identity":
{
    "orgId": "hj5fmy",
    "typeId": "NodeMCU",
    "deviceId": "12345" },
    "auth": { "token": "12345678" }
}

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()
```