

DEVELOP THE PYTHON SCRIPT(PUBLISH DATA TO IBM CLOUD)

Team ID	PNT2022TMID24427
Project Name	Industry-Specific Intelligent Fire Management System



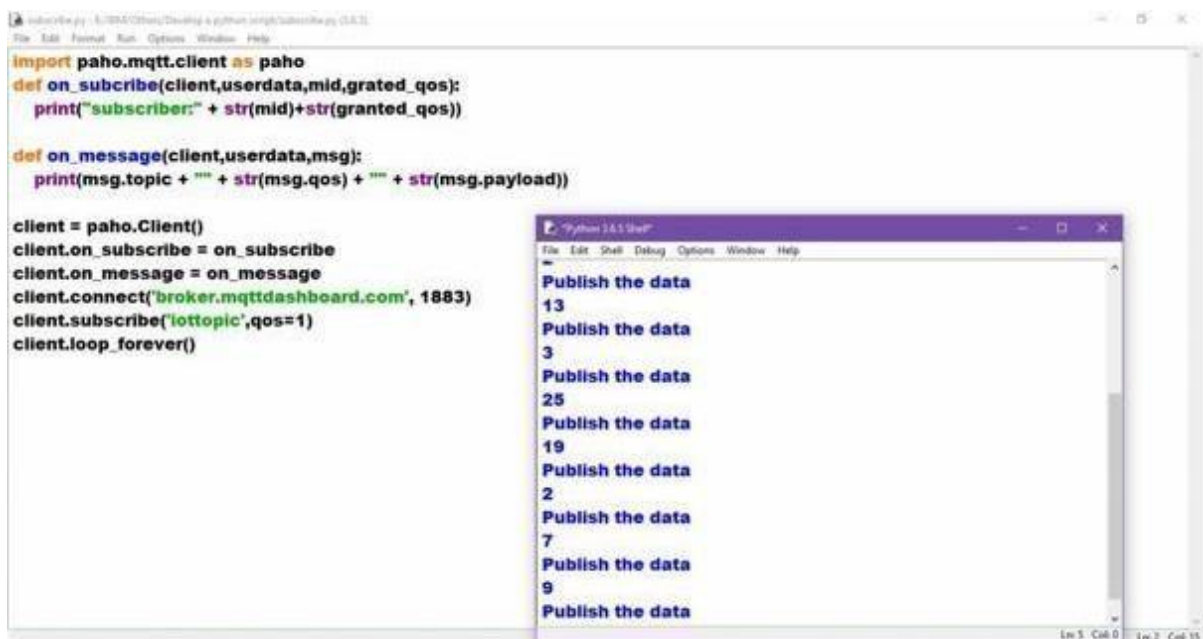
The screenshot shows a Python script in a text editor and its execution output in a terminal window. The script is titled "publish.py" and is located at "E:/IBM/Others/Develop a python script/publish.py (3.6.5)". The script imports the paho.mqtt.client module as paho, and also imports time and random modules. It defines a function on_publish(client, userdata, mid) that prints "Publish the data ". It then creates a paho.Client object, sets on_publish as the callback, connects to broker.mqttdashboard.com on port 1883, and starts the loop. A while True loop publishes a random integer between 1 and 30 to the topic "iottopic" with a QoS of 1, prints the value, and sleeps for 10 seconds.

```
#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 " followed by the random integer values: 7, 19, 10, and 10.



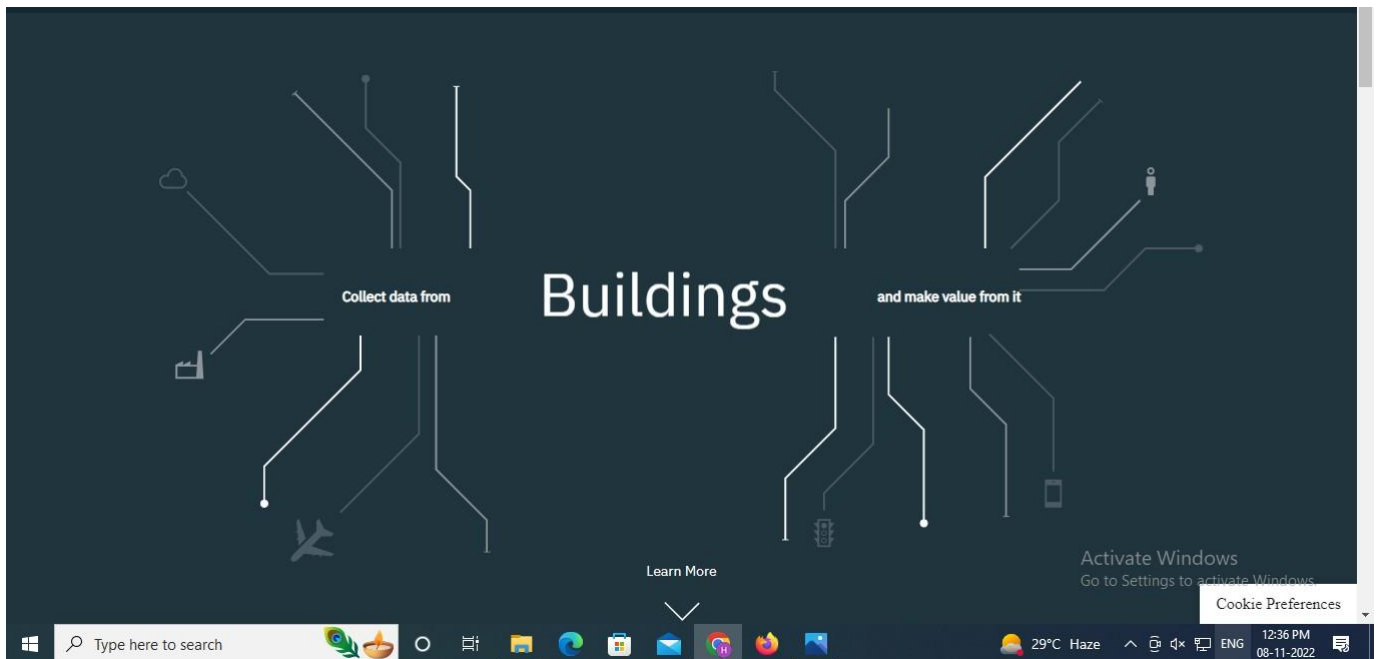
The screenshot shows a Python script in a text editor and its execution output in a terminal window. The script is titled "subscriber.py" and is located at "E:/IBM/Others/Develop a python script/subscriber.py (3.6.5)". The script imports the paho.mqtt.client module as paho. It defines a function on_subscribe(client, userdata, mid, granted_qos) that prints "subscriber: " followed by the mid and granted_qos values. It also defines a function on_message(client, userdata, msg) that prints the topic, qos, and payload of the message. It then creates a paho.Client object, sets on_subscribe and on_message as the callbacks, connects to broker.mqttdashboard.com on port 1883, subscribes to the topic "iottopic" with a QoS of 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 "subscriber: " followed by the mid and granted_qos values for each message received: 13, 3, 25, 19, 2, 7, 9, and 9.



Program :

#IBM Watson IOT
Platform

```
#pip install wiotp-sdk
import wiotp.sdk.device
import time
import random
```

```
myConfig = {"identity":
{
    "orgId": "b9nin6",
    "typeId": "NodeMCU",
    "deviceId": "12345"},
    "auth": {"token": "12345678"}
```

```
}
def myCommandCallback(cmd): print ("Message received from IBM
IoTPlatform: %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()
```