PUBLISH THE DATA TO THE IBM CLOUD

Date	17 November 2022
Team Id	PNT2022TMID25311
Project title	Smart waste management system for metropolitan cities

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']
```

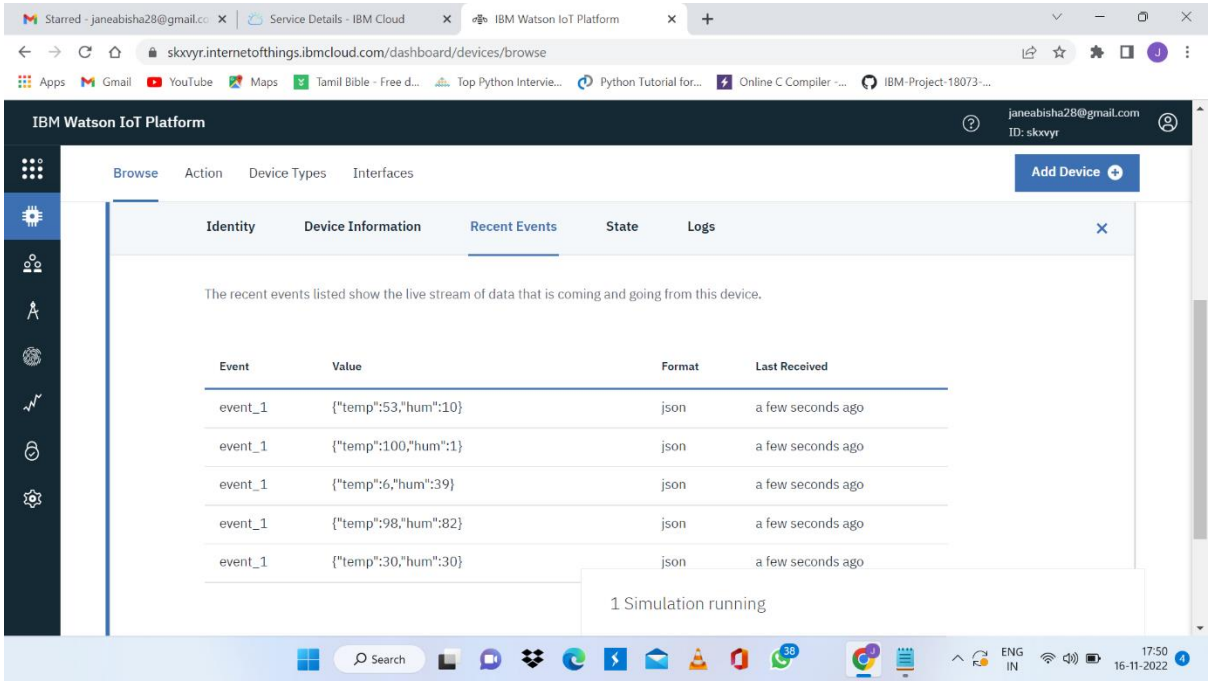
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

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()

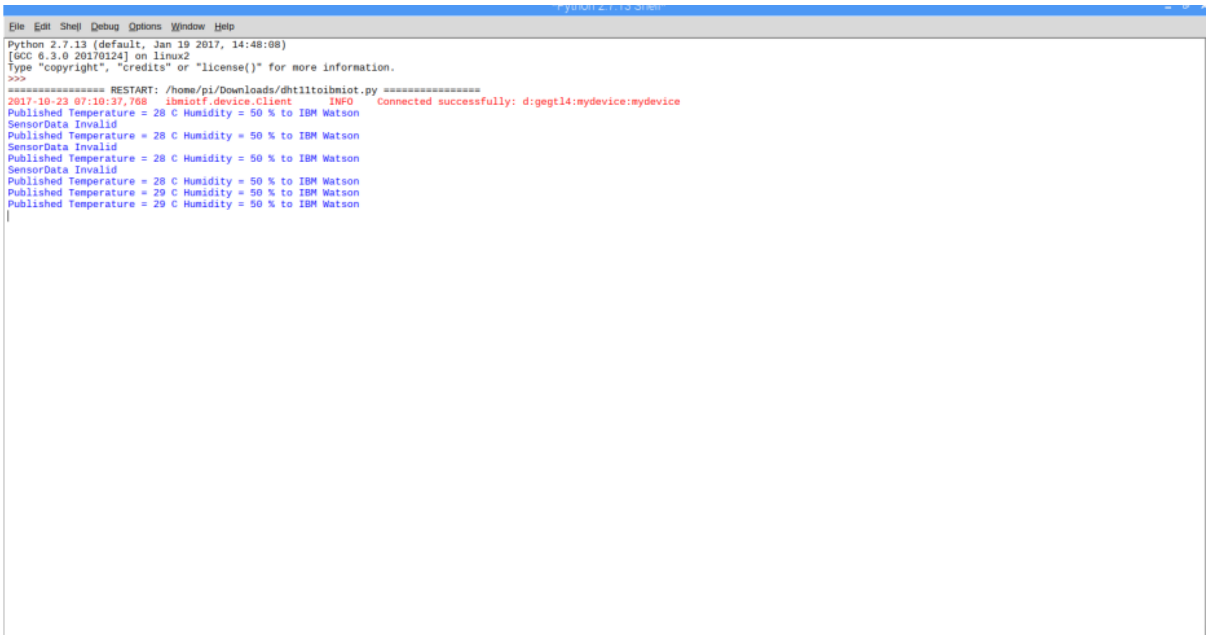


waston 1.py - C:\Users\abish\AppData\Local\Programs\Python\Python39\waston 1.py (3.9.6)

File Edit Format Run Options Window Help

```
import time
import sys
import ibmiotf.application
import randam

organization = "abisha"
deviceType = "swms"
devicedId = "6032"
authMethod = "token"
authToken = "3115191060031"

def mycommandcallback(cmd):
  print("commond received: %s" % cmd.data['command'])
  status=cmd.data['command']
  if status=="lighton":
    print("led is on")
  else :
    print ("led is off")

try:
   deviceOptions = {  "org": organization, "type": deviceType, "id": deviceId, "auth-method": authMethod, "auth_token": authToken
   //deviceCli = ibmiotf.device.Client(deviceOptions)

expect Exception as e:
   print("caught exception connecting device: %s" % str(e))
   sys.exit()

deviceCli.connect()
while True:

    temp=randam.randiant(0,100)
    Humid=randam.randiant(0,100)

    data={ 'temp' :temp, 'Humid':Humid }

    def myonpublishcallback():
       print("published Temperature = %s C" % temp,Humidity = %s %%" % Humid, "to IBM waston"
     success = deviceCli.publishEvent("IOTSensor","jason",data,qos=0,on_publish=myonpublishcallback)
     if not success:
        print(Not connect to IOTF")
      time.sleep(1)
```

Ln: 1 Col: 0


Python 2.7.13 Shell
File Edit Shell Debug Options Window Help
Python 2.7.13 (default, Jan 19 2017, 14:48:08)
[GCC 6.3.0 20170124] on linux2
Type "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: /home/pi/Downloads/dht11toibmiot.py ==========
2017-10-23 07:10:37,768 ibmiotf.device.Client INFO Connected successfully: d:gegtl4:mydevice:mydevice
Published Temperature = 28 C Humidity = 50 % to IBM Watson
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
SensorData Invalid
Published Temperature = 28 C Humidity = 50 % to IBM Watson
Published Temperature = 29 C Humidity = 50 % to IBM Watson
Published Temperature = 29 C Humidity = 50 % to IBM Watson