I

Sprint Delivery–3

TeamID	PNT2022TMID41351
ProjectName	Smart Farmer-IOT Enabled Smart Farming Application

PYTHON CODE

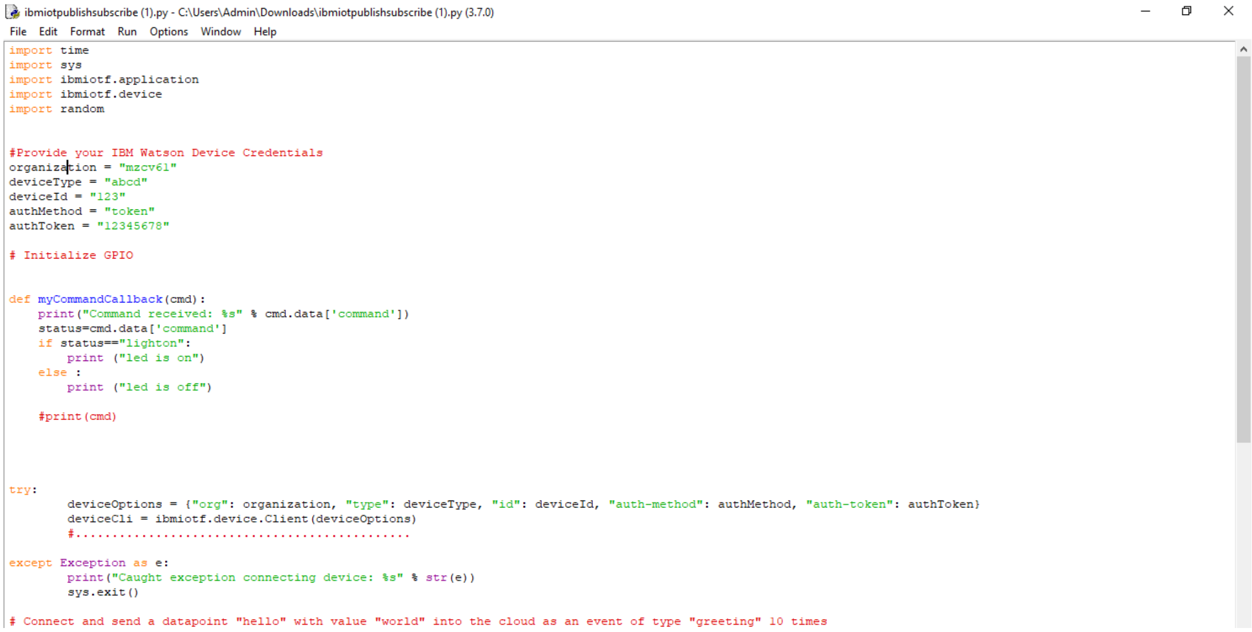
```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random


#Provide your IBM Watson Device Credentials
organization = "mzcv61"
deviceType = "abcd"
deviceId = "123"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO
```

I

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

    #print(cmd)




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

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

# Connect and send a datapoint "hello" with value "world" into the cloud as
an event of type "greeting" 10 times
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11
```

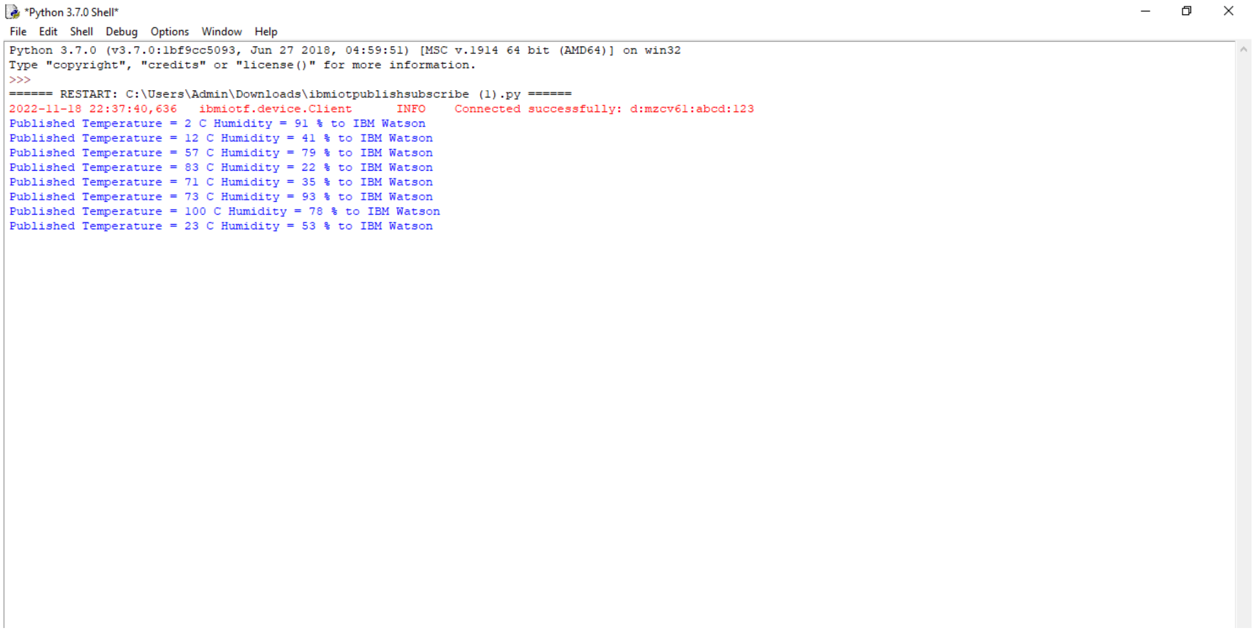
I

```
        temp=random.randint(0,100)
        Humid=random.randint(0,100)

        data = { 'temp' : temp, 'Humid': Humid }
        #print data
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temp, "Humidity = %s %%" %
Humid, "to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(1)

        deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud
deviceCli.disconnect()
```


ibmiotpublishsubscribe (1).py - C:\Users\Admin\Downloads\ibmiotpublishsubscribe (1).py (3.7.0)
File Edit Format Run Options Window Help

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random


#Provide your IBM Watson Device Credentials
organization = "mzcv61"
deviceType = "abcd"
deviceId = "123"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO


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

    #print(cmd)



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

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

# Connect and send a datapoint "hello" with value "world" into the cloud as an event of type "greeting" 10 times
```

OUTPUT



```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
====== RESTART: C:\Users\Admin\Downloads\ibmiotpublishsubscribe (1).py ======
2022-11-18 22:37:40,636   ibmiotf.device.Client      INFO    Connected successfully: d:mzcv61:abcd:123
Published Temperature = 2 C Humidity = 91 % to IBM Watson
Published Temperature = 12 C Humidity = 41 % to IBM Watson
Published Temperature = 57 C Humidity = 79 % to IBM Watson
Published Temperature = 83 C Humidity = 22 % to IBM Watson
Published Temperature = 71 C Humidity = 35 % to IBM Watson
Published Temperature = 73 C Humidity = 93 % to IBM Watson
Published Temperature = 100 C Humidity = 78 % to IBM Watson
Published Temperature = 23 C Humidity = 53 % to IBM Watson
```