DEVELOP A PYTHON SCRIPT TO PUBLISH AND SUBSCRIBE TO IBM IOT PLATFORM

Team ID	PNT2022TMID30034
Project Name	Smart Farmer - IoT Enabled Smart Farming Application

Python Script:

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


#Provide your IBM Watson Device Credentials
organization = "o7kvsp"
deviceType = "Aurdino"
deviceId = "123"
authMethod = "token"
authToken = "87654321"

# Initialize GPIO


def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
```

```
    if status=="Lighton":
        print ("Light is on")
    elif status=="Motoron":
        print ("Motor is on")
    elif status=="Lightoff":
        print ("Light is off")
    else:
        print("Motor 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

        temp=random.randint(0,100)
        hum=random.randint(0,100)
        moisture=random.randint(0,100)
        distance=random.randint(0,500)
```

```
        data = { 'temp' : temp, 'hum': hum, 'moisture' : moisture, 'distance':
distance}
        #print data
        def myOnPublishCallback():
            print ("Published Temperature = %s C" % temp, "Humidity = %s
%%" % hum, "Moisture = %s %%" % moisture,"Distance = %s %%" %
distance, "to IBM Watson")

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
        if not success:
            print("Not connected to IoTF")
        time.sleep(50)

        deviceCli.commandCallback = myCommandCallback

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