

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

Date	18 Nov 2022
Team ID	PNT2022TMID04699
Project Name	Smart Farmer - IoT Enabled Smart FarmingApplication

PROGRAM

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

#Provide your IBM Watson Device Credentials
organization = "95a96q"
deviceType = "NodeMCu"
deviceId = "123456"
authMethod = "use-token-auth"
authToken = "P123@456"

# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="motoron":
        print ("motor is on")
    elif status == "motoroff":
        print ("motor is off")
    else :
```

```

        print ("please send proper command")
    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(90,110)
    Humid=random.randint(60,100)
    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(10)
        deviceCli.commandCallback = myCommandCallback
# Disconnect the device and application from the cloud
deviceCli.disconnect()

```