

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

Date	30 October 2022
Team ID	PNT2022TMID40817
Project Name	SMARTFARMER – IoT ENABLED SMART FARMING APPLICATION

PROGRAM :

```
import time

import sys

import
ibmiotf.application

import ibmiotf.device

import random

#Provide your IBM
Watson Device
Credentials

organization = "x0fxss"
#replace the ORG ID

deviceType =
"Testing"#replace the
Device type wi

deviceId =
"Testdevice1"#replace
Device ID

authMethod = "token"

authToken =
"123456789" #Replace
the authtoken

# Initialize GPIO

#Receives Command
from Node-red

def
```

```

myCommandCallback(
cmd):

print ("Command
received: %s" %
cmd.data['command'])

status=cmd.data['comm
and']

if status=="motoron":

print ("motor is on")

elif status ==
"motoroff" :

print ("motor is off")

elif status == "motor30"
:

print ("motor is on for
30 minutes")

try:

deviceOptions = {"org":
organization, "type":
deviceType, "id":
deviceId, "auth-
method": authMethod,
"auth-token":
authToken}

deviceCli =
ibmiotf.device.Client(d
eviceOptions)

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)

Humid=random.randint
(0,100)

soilmoisture=random.ra
ndint(0,100)

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

#print data

def
myOnPublishCallback()
:

print ("Published
Temperature = %s C"
% temp, "Humidity =
%s %% " % Humid,
"soilmoisture = %s
%% "

%soilmoisture, "to IBM
Watson")

success =
deviceCli.publishEvent(
"IoTSensor", "json",
data, qos=0,
on_publish=myOnPubli
shCallback)

if not success:

print("Not connected to

IoTF")

time.sleep(5)

deviceCli.commandCall

back =

myCommandCallback

deviceCli.disconnect()