

SPRINT 1

Date	29 October 2022
Team ID	PNT2022TMID46860
Project Name	Project – Smart Farmer- IoT based Smart Farming Application

PYTHON CODE

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

#Provide your IBM Watson Device Credentials
organization = "j3bgcj"
deviceType = "nodeMCU"
deviceId = "1234"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO

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

```

        print ('Motor is on')
    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(0,30)
    Humid=random.randint(0,100)
    soil=random.randint(30,50)

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

```

```

#print data

def myOnPublishCallback():

    print ('Published Temperature = %s C' % temp, "Humidity = %s %%" %
Humid,"soil_moisture =%sC"% soil ,"to IBM Watson")

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

    if not success:

        print("Not connected to IoT")

        time.sleep(1)

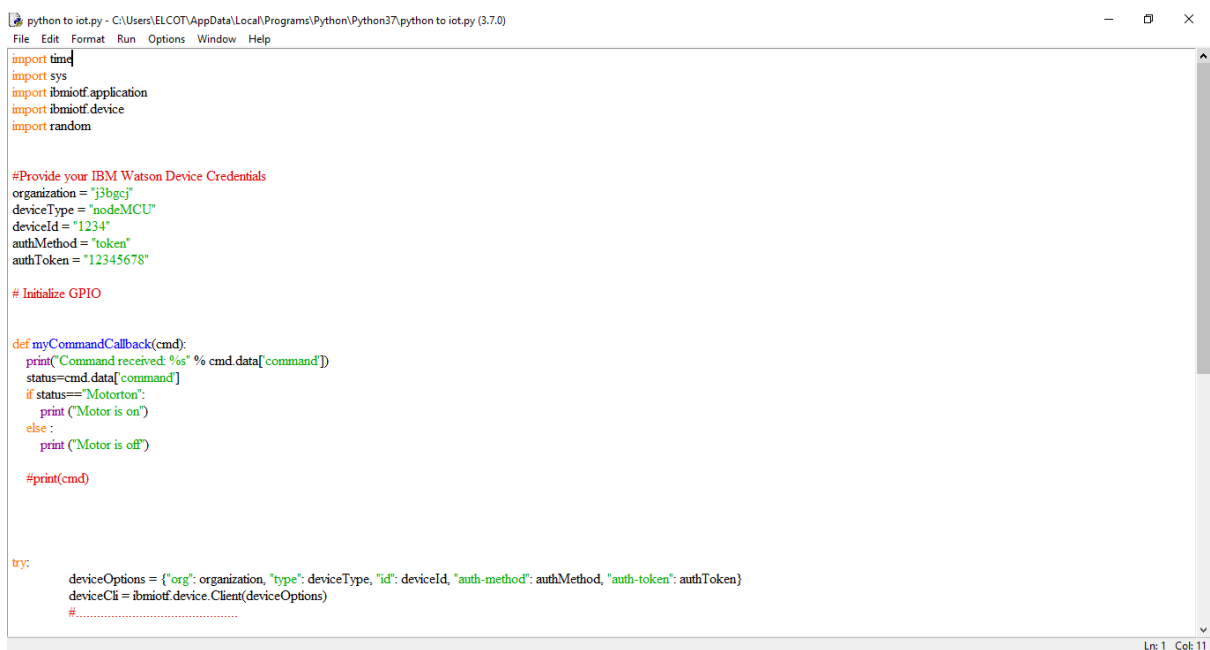
    deviceCli.commandCallback = myCommandCallback

# Disconnect the device and application from the cloud

deviceCli.disconnect()

```

PYTHON CODE ON PYTHON IDLE 3.7.0



```

python to iot.py - C:\Users\ELCOT\AppData\Local\Programs\Python\Python37\python to iot.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 = "j3bgej"
deviceType = "nodeMCU"
deviceId = "1234"
authMethod = "token"
authToken = "12345678"

# Initialize GPIO

def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="Motorton":
        print ("Motor is on")
    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)
    # .....

```

```
python to iot.py - C:\Users\ELCOT\AppData\Local\Programs\Python\Python37\python to iot.py (3.7.0)
File Edit Format Run Options Window Help

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(0,30)
    Humid=random.randint(0,100)
    soil=random.randint(30,50)

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

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

    deviceCli.commandCallback = myCommandCallback

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

OUTPUT

```
*Python 3.7.0 Shell*
File Edit Shell Debug Options Window Help

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\ELCOT\AppData\Local\Programs\Python\Python37\python to iot.py
2022-11-18 16:00:39.774 ibmiotf.device.Client INFO Connected successfully: dj3bgcjnodeMCU:1234
Published Temperature = 3 C Humidity = 79 % soil_moisture =48C to IBM Watson
Published Temperature = 2 C Humidity = 87 % soil_moisture =31C to IBM Watson
Published Temperature = 8 C Humidity = 95 % soil_moisture =45C to IBM Watson
Published Temperature = 24 C Humidity = 43 % soil_moisture =38C to IBM Watson
Published Temperature = 16 C Humidity = 91 % soil_moisture =33C to IBM Watson
Published Temperature = 24 C Humidity = 82 % soil_moisture =50C to IBM Watson
Published Temperature = 24 C Humidity = 11 % soil_moisture =45C to IBM Watson
Published Temperature = 28 C Humidity = 1 % soil_moisture =42C to IBM Watson
Published Temperature = 21 C Humidity = 39 % soil_moisture =38C to IBM Watson
Published Temperature = 19 C Humidity = 69 % soil_moisture =42C to IBM Watson
Published Temperature = 14 C Humidity = 58 % soil_moisture =42C to IBM Watson
Published Temperature = 29 C Humidity = 23 % soil_moisture =43C to IBM Watson
Published Temperature = 8 C Humidity = 17 % soil_moisture =31C to IBM Watson
Published Temperature = 29 C Humidity = 44 % soil_moisture =31C to IBM Watson
|
```