

# **PYTHON SCRIPT** **(HUMIDITY,TEMPERATURE,PRESSURE)**

DATE	17/11/2022
TEAM ID	PNT2022TMID22485
PROJECT NAME	INDUSTRY-SPECIFIC INTELLIGENT FIRE MANAGEMENTSYSTEM
MARKS	4 marks

## **PYTHONCODE**

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "b7ywuf"
deviceType = "py_code"
deviceId = "22"
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")
    elif status == "lightoff":
        print ("led 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 IoTTF")
            time.sleep(10)

        deviceCli.commandCallback = myCommandCallback

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

```

# OUTPUT

```
Pythoncode_NT.py - C:\Users\prasa\AppData\Local\Programs\Python\Python39\Pythoncode_NT.py (3.9.7)
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 = "b7ywuf"
deviceType = "py_code"
deviceId = "22"
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")
    elif status == "lightoff":
        print ("led is off")
    else :
        print ("please send proper command")

try:
    deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-meth": authMethod}
    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
deviceCli.connect()

while True:
    #Get Sensor Data from DHT11
    temper=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")

Python 3.9.7 (tags/v3.9.7:1016ef3, Aug 30 2021, 20:19:38) [MSC v.1929 64 bit (AMD64)] on win
32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
= RESTART: C:\Users\prasa\AppData\Local\Programs\Python\Python39\Pythoncode_NT.py
2022-11-17 13:58:38,187 ibmiotf.device.Client INFO Connected successfully: d:b7ywu
f:py_code:22
Published Temperature = 99 C Humidity = 74 % to IBM Watson
Published Temperature = 93 C Humidity = 85 % to IBM Watson
Published Temperature = 98 C Humidity = 64 % to IBM Watson
Published Temperature = 105 C Humidity = 87 % to IBM Watson
Published Temperature = 92 C Humidity = 64 % to IBM Watson
Published Temperature = 90 C Humidity = 84 % to IBM Watson
Published Temperature = 95 C Humidity = 71 % to IBM Watson
Published Temperature = 94 C Humidity = 75 % to IBM Watson
Published Temperature = 101 C Humidity = 81 % to IBM Watson
Published Temperature = 106 C Humidity = 96 % to IBM Watson
```