

DEVELOP A PYTHON SCRIPT

Team ID	PNT2022TMID04058
Project Name	Smart waste management system for metropolitan cities

Step 1: Open python

idleStep2: Type the program

Step 3: Then click on file and save the

documentStep 4: Then click on Run then Run Module

Step 5: output will be appeared in the idle window

PYTHON SCRIPT CODE

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

```
#Provide your IBM Watson Device Credentials
organization = "ztyu1i"
deviceType = "python"
deviceId = "12345"
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 IoT")
time.sleep(10)

deviceCli.commandCallback = myCommandCallback

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

Screenshots Python script :

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

#Provide your IBM Watson Device Credentials
organization = "ztyuli"
deviceType = "python"
deviceId = "12345"
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, "aut
    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 c
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,

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

# Connect and send a datapoint "hello" with value "world" into the cloud as an event c
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,

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publish=
    if not success:
        print("Not connected to IoT")
        time.sleep(10)

    deviceCli.commandCallback = myCommandCallback

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

```
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/KISHORE KUMAR/OneDrive/Desktop/script.py =====
```

```
2022-11-18 23:53:57,226 ibmiotf.device.Client INFO Connected successfully:
```

```
d:ztyuli:python:12345
```

```
Published Temperature = 99 C Humidity = 65 % to IBM Watson  
Published Temperature = 90 C Humidity = 73 % to IBM Watson  
Published Temperature = 98 C Humidity = 64 % to IBM Watson  
Published Temperature = 97 C Humidity = 61 % to IBM Watson  
Published Temperature = 110 C Humidity = 66 % to IBM Watson  
Published Temperature = 98 C Humidity = 89 % to IBM Watson  
Published Temperature = 106 C Humidity = 74 % to IBM Watson  
Published Temperature = 108 C Humidity = 74 % to IBM Watson  
Published Temperature = 101 C Humidity = 60 % to IBM Watson  
Published Temperature = 98 C Humidity = 85 % to IBM Watson  
Published Temperature = 96 C Humidity = 76 % to IBM Watson  
Published Temperature = 106 C Humidity = 98 % to IBM Watson  
Published Temperature = 95 C Humidity = 65 % to IBM Watson  
Published Temperature = 109 C Humidity = 95 % to IBM Watson  
Published Temperature = 95 C Humidity = 83 % to IBM Watson  
Published Temperature = 91 C Humidity = 66 % to IBM Watson  
Published Temperature = 102 C Humidity = 61 % to IBM Watson
```