Develop a python script

Team ID	PNT2022TMID33002
Project Name	Project-Smart Waste Management System for Metropolitan Cities

Step 1: Open python idle

Step 2: Type the program

Step 3: Then click on file and save the document

Step 4: Then click on Run then Run Module

Step 5: output will be appeared in the idle window

Python script :

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "w36lp6"
deviceType = "Monalisa"
deviceId = "2802"
authMethod = "token"
authToken = "28022002"
# Initialize GPIO
def myCommandCallback(cmd):
   print("Command received: %s" % cmd.data['command'])
   status=cmd.data['command']
   if status=="lighton":
      print ("led is on")
   else :
      print ("led 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(0,100)
      Humid=random.randint(0,100)
      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()
```

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 = "w36lp6"
deviceType = "Monalisa"
deviceId = "2802"
authMethod = "token"
authToken = "28022002"

# Initialize GPIO


def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":
        print ("led is on")
    else :
        print ("led is off")

    #print(cmd)



try:
        deviceOptions = {"org": organization, "type": deviceType, "id": deviceId, "auth-method": au
        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 "greet
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11

        temp=random.randint(0,100)
        Humid=random.randint(0,100)
```

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 (AMD6
4)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
========== RESTART: C:\MONALISA\DOCUMENTS\ibmiotpublishsubscribe.py ==========
2022-11-15 12:51:05,831    ibmiotf.device.Client      INFO    Connected successfu
lly: d:w36lp6:Monalisa:2802
Published Temperature = 40 C Humidity = 73 % to IBM Watson
Published Temperature = 12 C Humidity = 48 % to IBM Watson
Published Temperature = 21 C Humidity = 65 % to IBM Watson
Published Temperature = 63 C Humidity = 65 % to IBM Watson

========== RESTART: C:\MONALISA\DOCUMENTS\ibmiotpublishsubscribe.py ==========
2022-11-15 12:51:46,793    ibmiotf.device.Client      INFO    Connected successfu
lly: d:w36lp6:Monalisa:2802
Published Temperature = 14 C Humidity = 2 % to IBM Watson
Published Temperature = 28 C Humidity = 8 % to IBM Watson
```

Ln: 12 Col: 0