DEVELOP A PYTHON SCRIPT TO PUBLISH

Team ID	PNT2022TMID29852
Project Name	SMART FARMER – IoT ENABLED SMART FARMING APPLICATION

PROGRAM:

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

#provide Your IBM Watson Device Credentials
organization = "9te1u1"
deviceType = "SFTTMS00"
deviceID = "SFTTMS11"
authMethod = "token"
authToken = "PNTIBMSb18"


#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" info the
cloud as an event of type"greetings"10 times
deviceCli.connect()

while True:
      #Get sensor Data from DHT11

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

      data = { 'temp' : temp, 'Humid': Humid,
'soilmoisture':soilmoisture}
      #print data
      def myOnPublishCallback():
         print ("published Temperature = %s C" %  temp,
"Humidity = is %s %%" % Humid, "soilmoisture= is %s %%" %
soilmoisture,"to IBM Watson")



      success = deviceCli.publishEvent("IOTSensor",
"json",data,qos=0,on_publish=myOnPublishCallback)
      if not success:
         print("Not connected to IOTF")
      time.sleep(5)



      deviceCli.commandCallback = myCommandCallback

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