DEVELOP THE PYTHON SCRIPT

Develop a Python Script

Team ID	PNT2022TMID54322
Project Name	Signs with Smart Connectivity for Better Road Safety

PYTHON SCRIPT

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "efr0if"
deviceType = "rasberrypi"
deviceId = "123"
authMethod = "token"
authToken = "12345678"
 #Intialize GPIO
def myCommandCallback(command):
    print("Command received: %s" % command.data['command'])
    status=command.data['command']
    if status=="lighton":
        print("led is on")
    elif status=="lightoff":
        print("led is off")
    else:
        print("please send proper command")
        #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()
```