Develop A Python Script To Publish And Subscribe To IBM IoT Platform

Develop The Python Code

Date	1 November 2022
Team ID	PNT2022TMID26645
Project Name	Project – Gas leakage monitoring and alerting system for industries

Code:

```
import time import sys

import ibmiotf.application

import ibmiotf.device

import random



#Provide your IBM Watson Device Credentials

organization = "oefkwc"

deviceType = "PNT2022TMID26645"

deviceId = "PNT2022TMID426645DEVICEID"

authMethod = "use-token-auth"

authToken = "0RZfFDkDNSK8o@gcpd"


# Initialize GPIO



def myCommandCallback(cmd):     print("Command

received: %s" % cmd.data['command'])

status=cmd.data['command']     if status=="alarmon":

print ("Alarm is on")     elif (status == "alarmoff") :

       print ("Alarm is off")

   elif status == "sprinkleron":

print("Sprinkler is OFF")     elif
```

```
status == "sprinkleron":
print("Sprinkler is ON")
   #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)       gas=random.randint(0,100)


       data = { 'temp' : temp, 'Humid': Humid, 'gas' : gas }
       #print data
       def myOnPublishCallback():
          print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid, "Gas_Level =
%s %%" %gas, "to IBM Watson")
```

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

    if not success:           print("Not

connected to IoTF")

time.sleep(1)


    deviceCli.commandCallback = myCommandCallback


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