DEVELOPTHEPYTHONCODE

Date	19 November 2022
TeamID	PNT2022TMID30401
ProjectName	GasLeakageMonitoringandAlertin gSystem
MaximumMark	4marks

SanthoshP,AkashRajM, BadhrMuhammedS,LakshmiNarayananB

PYTHONCODE:

```
Import
timeImpor
tsys
import
ibmiotf.applicationimpo
rtibmiotf.deviceimportr
andom

#Provide your IBM Watson
DeviceCredentialsorganization =
"5py6q9"deviceType =
"Weather_now"deviceId =
"Weather1234"authMethod="tok
en"
authToken="XeJFia7_@@t9@@eq_?"

#InitializeGPIO
defmyCommandCallback(cmd):
print("Commandreceived:%s"%cmd.data['command'])status
=cmd.data['command']
ifstatus=="lighton":
print("ledison")
elifstatus=="lightoff":
print(“ledisoff”)
else:
print(“pleasesendpropercommand”)

try:
        deviceOptions={“org”:organization,“type”:deviceType,”id”:deviceId,“auth-
method”:authMethod,           “auth-token”:
authToken}deviceCli=ibmiotf.device.Client(
deviceOptions)#............................
```

```
exceptExceptionase:
   print("Caught exception
   connectingdevice:%s"%
   str(e))sys.exit()

# Connect and send a datapoint "hello" with value
world"into the cloudas an event of type "greeting" 10
timesdeviceCli.connect()


whileTrue:

#Get Sensor Data from

DHT11temp=random.randint

(90,110)Humid=random.rand

int(60,100)

      data={'temp':temp,'H
      umid': Humid
       }#printdata
      defmyOnPublishCallback():
         print("PublishedTemperature=%sC"%temp,"Humidity=%s
%%"%Humid,"toIBMWatson")
      success=deviceCli.publishEvent("IoTSens
or","json",data,qos=0,on_publish=myOnPublish
Callback)
       ifnotsucces
      s:print("Notc
       onnectedto
      IoTE")
      time.sleep(10)


      deviceCli.commandCallback=myCommandCallback


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