Develop the Python code

Team ID	PNT2022TMID28270
Project Name	Smart Farmer - IoT Enabled Smart Farming Application

DEVELOP A PYTHON SCRIPT TO PUBLISH AND SUBSCRIBE TO IBM IOT PLATFORM:

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

#NARASIMHAN IBM

organization = "59lw3i"
deviceType = "device_1"
deviceId = "12345"
authMethod = "token"
authToken = "123456789"

#GPIO

def mycommandCallback(cmd):
    print("Command Received: %s" %cmd.data['command'])
    status = cmd.data['command']
    if status=="motoron":
        print("MOTOR is ON")
    elif status=="motoroff":
        print("MOTOR 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

deviceCli.connect()

while True:
    temperature=random.randint(0,100)
    humidity=random.randint(0,100)
    moisture=random.randint(0,100)

    data={'temperature':temperature,'humidity':humidity,'moisture':moisture}

    def myOnPublishCallback():
        print("Published Temperature = %s C"%temperature,"Humidity = %s %%" %humidity,"Moisture = %s %%" %moisture, "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

deviceCli.disconnect()
```