DEVELOP THE PYTHON CODE

Team ID	PNT2022TMID46404
Project Name	Smart Farmer - IOT Enabled Smart Farming Application

PYTHON CODE:

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "4tot8b"
deviceType = "smart_farming"
deviceId = "farm_today"
authMethod = "token"
authToken = "oiJYpRYqYNUC)E2eAt"
# Initialize 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 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)
     Mois=random.randint(0,100)
     data = {"d":{'temp' : temp, 'Humid': Humid, 'Mois' :Mois}}
     #print data
     def myOnPublishCallback():
       print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid, "Moisture =%s deg c" %Mois, "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()
```

ibm iot code.py - C:\Users\PC\Desktop\ibm iot code.py (3.7.0)

File Edit Format Run Options Window Help

```
import time
import sys
import ibmiotf.application
import ibmiotf.device
import random
#Provide your IBM Watson Device Credentials
organization = "4tot8b"
deviceType = "smart_farming"
deviceId = "farm_today"
authMethod = "token"
authToken = "oiJYpRYqYNUC)E2eAt"
# Initialize 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 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)
     Mois=random.randint(0,100)
     data = {"d":{'temp' : temp, 'Humid': Humid, 'Mois' :Mois}}
     #print data
     def myOnPublishCallback():
        print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid, "Moisture =%s deg c" %Mois, "to IBM Watson")
```

ibm iot code.py - C:\Users\PC\Desktop\ibm iot code.py (3.7.0)

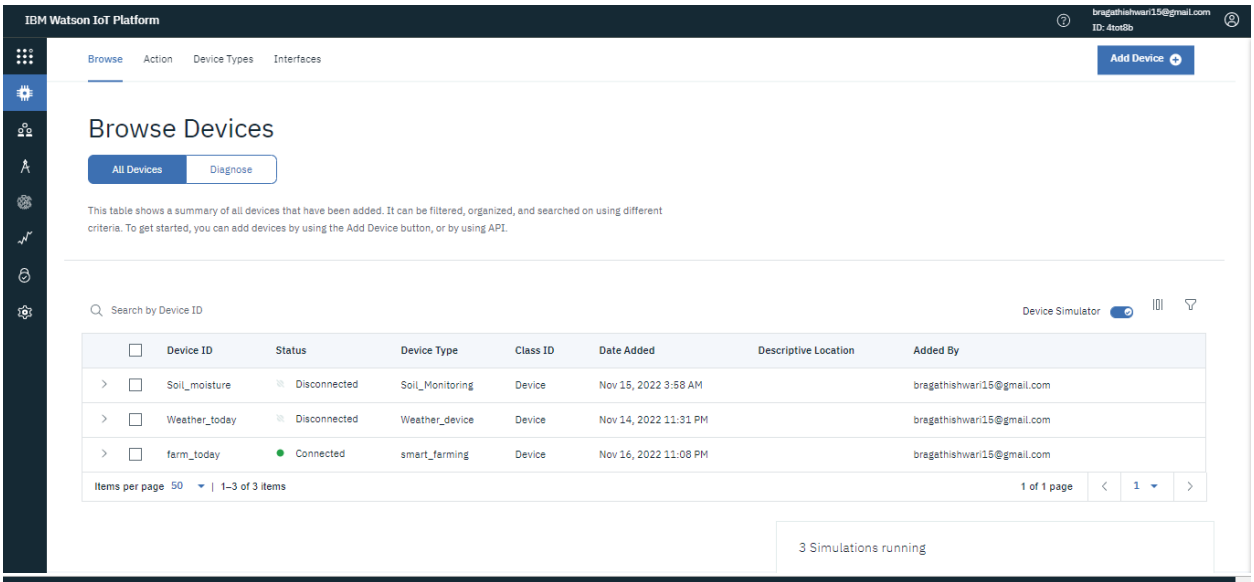
File Edit Format Run Options Window Help

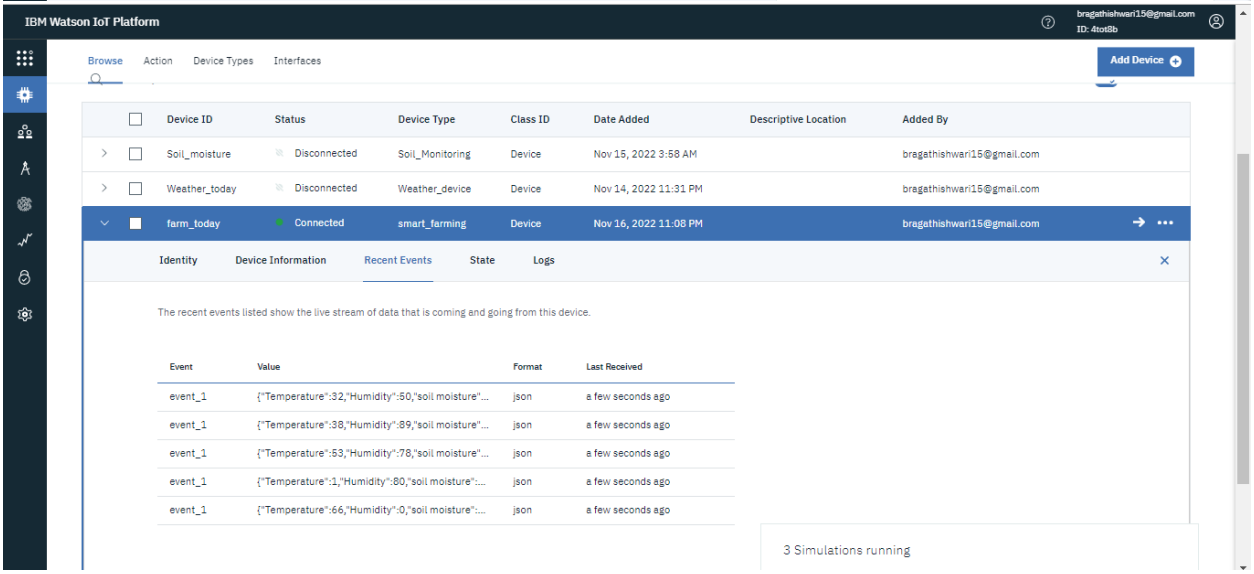
```
deviceId = "farm_today"
authMethod = "token"
authToken = "oiJYpRYqYNUC)E2eAt"
# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="motoron":
        print ("motor is on")
    elif status == "motoroff":
        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 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)
     Mois=random.randint(0,100)
     data = {"d":{'temp' : temp, 'Humid': Humid, 'Mois' :Mois}}
     #print data
     def myOnPublishCallback():
        print ("Published Temperature = %s C" % temp, "Humidity = %s %%" % Humid, "Moisture =%s deg c" %Mois, "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()
```

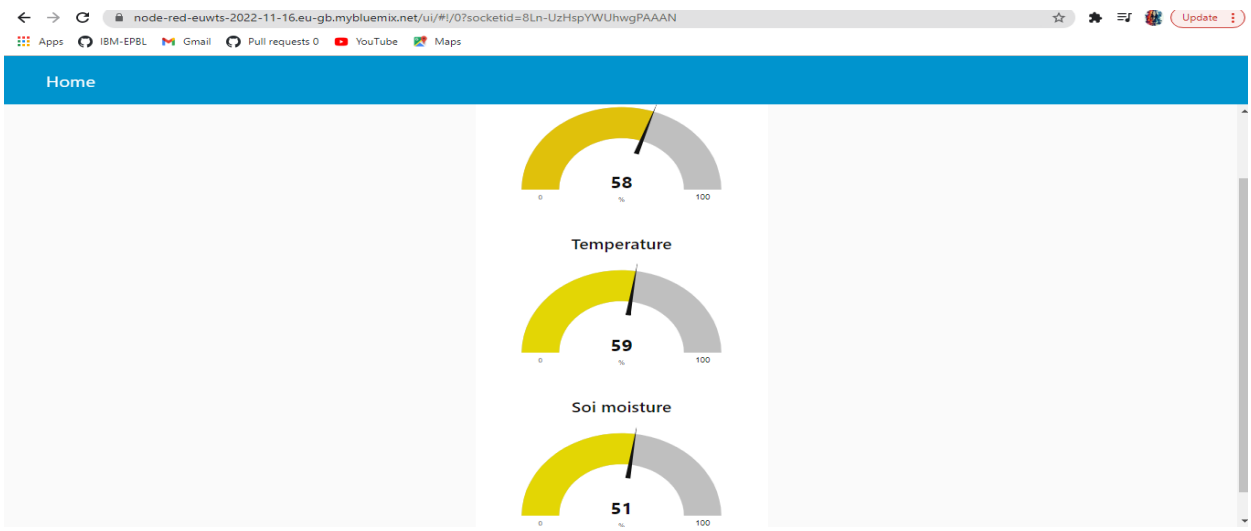
Python 3.7.0 Shell

File Edit Shell Debug Options Window Help

```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
================ RESTART: C:\Users\PC\Desktop\ibm iot code.py ================
2022-11-17 12:42:39,934   ibmiotf.device.Client      INFO    Connected successfully: d:4tot8b:smart_farming:farm_today
```


IBM Watson IoT Platform
bragathishwari15@gmail.com
ID: 4tot8b
Browse
Action
Device Types
Interfaces
Add Device
Browse Devices
All Devices
Diagnose
This table shows a summary of all devices that have been added. It can be filtered, organized, and searched on using different criteria. To get started, you can add devices by using the Add Device button, or by using API.
Search by Device ID
Device Simulator
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
Added By
Soil_moisture
Disconnected
Soil_Monitoring
Device
Nov 15, 2022 3:58 AM
bragathishwari15@gmail.com
Weather_today
Disconnected
Weather_device
Device
Nov 14, 2022 11:31 PM
bragathishwari15@gmail.com
farm_today
Connected
smart_farming
Device
Nov 16, 2022 11:08 PM
bragathishwari15@gmail.com
Items per page 50
1–3 of 3 items
1 of 1 page
1
3 Simulations running


IBM Watson IoT Platform
bragathishwari15@gmail.com
ID: 4tot8b
Browse
Action
Device Types
Interfaces
Add Device
Device ID
Status
Device Type
Class ID
Date Added
Descriptive Location
Added By
Soil_moisture
Disconnected
Soil_Monitoring
Device
Nov 15, 2022 3:58 AM
bragathishwari15@gmail.com
Weather_today
Disconnected
Weather_device
Device
Nov 14, 2022 11:31 PM
bragathishwari15@gmail.com
farm_today
Connected
smart_farming
Device
Nov 16, 2022 11:08 PM
bragathishwari15@gmail.com
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
event_1
{"Temperature":32,"Humidity":50,"soil moisture"...
json
a few seconds ago
event_1
{"Temperature":38,"Humidity":89,"soil moisture"...
json
a few seconds ago
event_1
{"Temperature":53,"Humidity":78,"soil moisture"...
json
a few seconds ago
event_1
{"Temperature":1,"Humidity":80,"soil moisture":...
json
a few seconds ago
event_1
{"Temperature":66,"Humidity":0,"soil moisture":...
json
a few seconds ago
3 Simulations running


node-red-euwts-2022-11-16.eu-gb.mybluemix.net/ui/#!/0?socketid=8Ln-UzHspYWUhwgPAAAN
Update
Apps
IBM-EPBL
Gmail
Pull requests 0
YouTube
Maps
Home
58
Temperature
59
Soi moisture
51