

PYTHON SOURCE CODE

Team ID	PNT2022TMID29852
Project Name	SMART FARMER – IoT ENABLED SMART FARMING APPLICATION

PROGRAM:

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

#provide Your IBM Watson Device Credentials
organization = "9te1u1"
deviceType = "SFTTMS00"
deviceID = "SFTTMS11"
authMethod = "token"
authToken = "PNTIBMSb18"

#Initialize GPIO
def myCommandCallback(cmd):
    print ("command received: %s" %cmd.data['command'])
    status=cmd.data['command']
    if status=="lighton":print
        ("led is on")
    elif status == "lightoff":
        print ("led 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" info the
cloud as an event of type "greetings" 10 times
deviceCli.connect()

while True:
    #Get sensor Data from DHT11

    temp=random.randint(0,100)
    Humid=random.randint(0,100)
    soilmoisture=random.randint(0,100)

    data = { 'temp' : temp, 'Humid': Humid,
'soilmoisture':soilmoisture}
    #print data
    def myOnPublishCallback():
        print ("published Temperature = %s C" % temp, "Humidity =
is %s %%" % Humid, "soilmoisture= is %s %%" %soilmoisture,"to
IBM Watson")

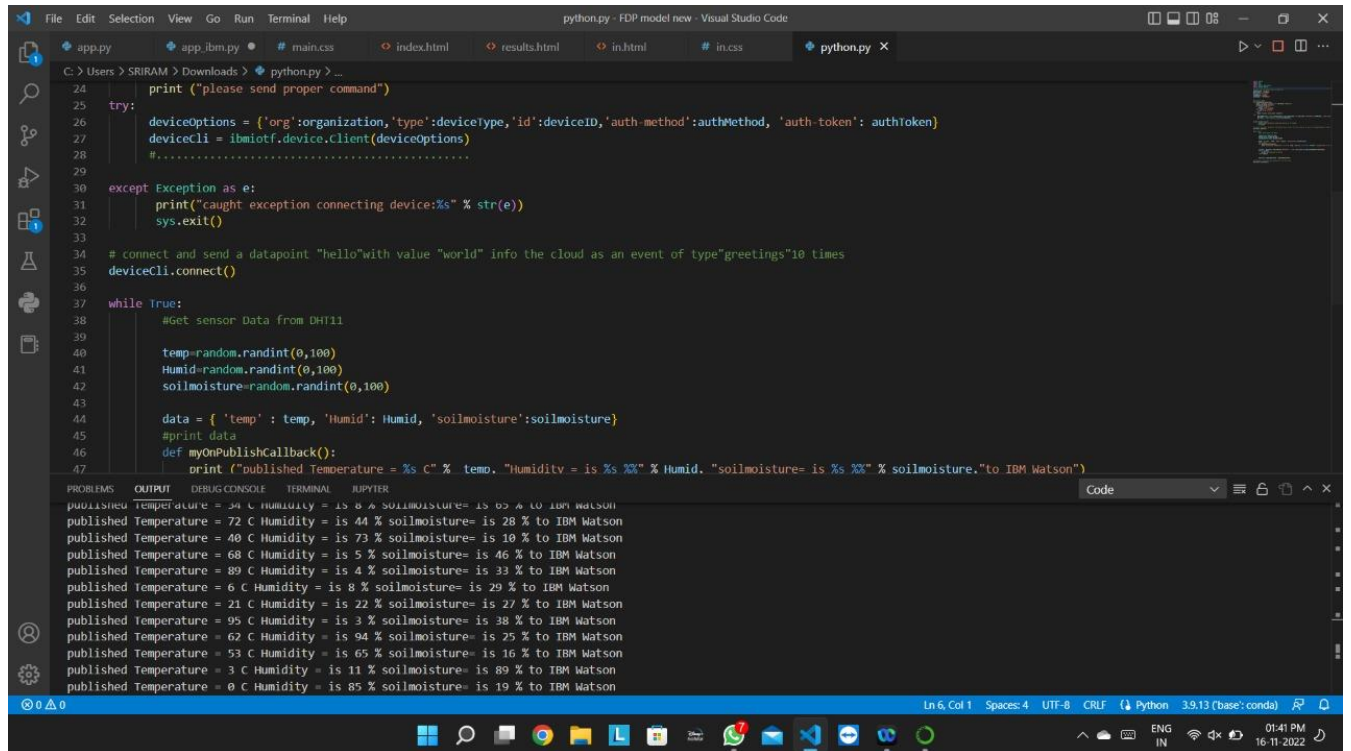
    success = deviceCli.publishEvent("IOTSensor",
"json",data,qos=0,on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IOTF")
        time.sleep(5)

    deviceCli.commandCallback = myCommandCallback

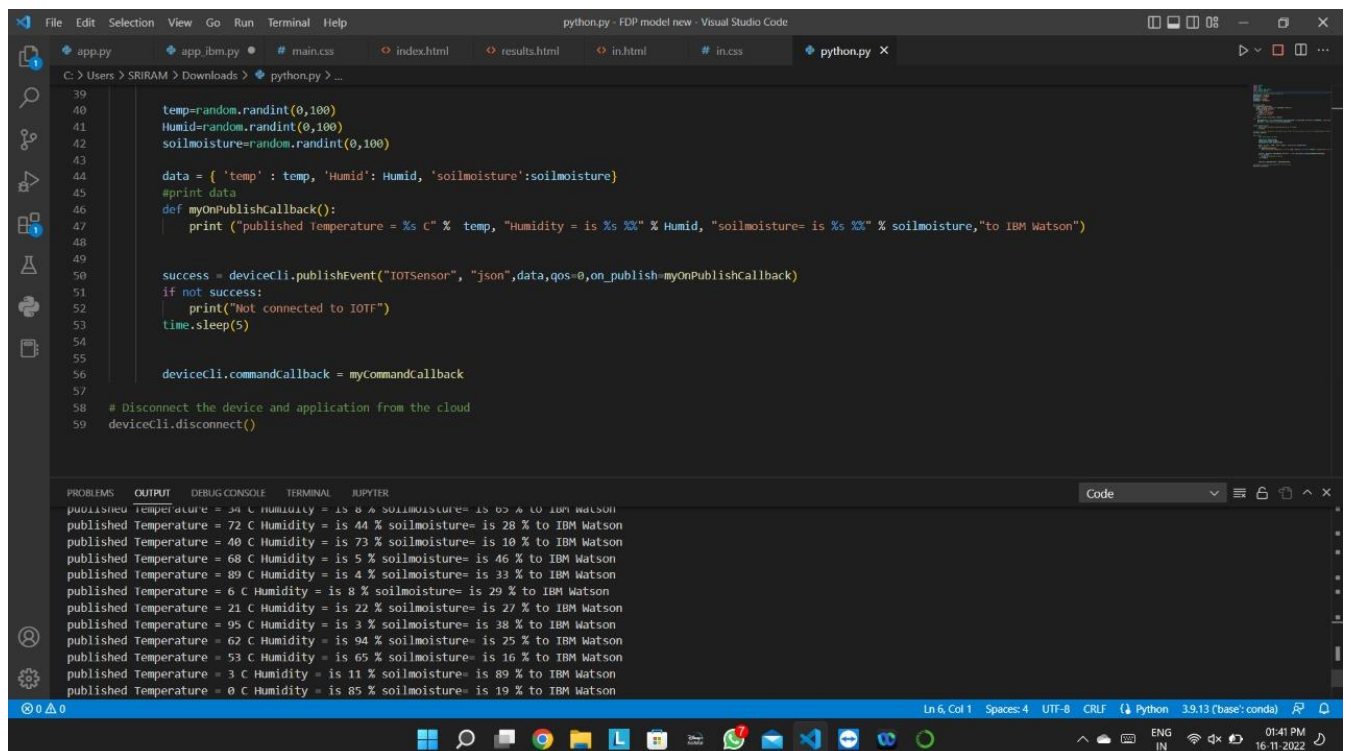
# Disconnect the device and application from the cloud
deviceCli.disconnect()

```

PYTHON CODE:



```
python.py - FDP model new - Visual Studio Code
C:\Users> SRIRAM > Downloads > python.py > ...
24     print("please send proper command")
25
26     deviceOptions = {'org':organization,'type':deviceType,'id':deviceId,'auth-method':authMethod, 'auth-token': authToken}
27     deviceCli = ibmiotf.device.Client(deviceOptions)
28     #.....
29
30 except Exception as e:
31     print("caught exception connecting device:%s" % str(e))
32     sys.exit()
33
34 # connect and send a datapoint "hello"with value "world" info the cloud as an event of type" greetings"10 times
35 deviceCli.connect()
36
37 while True:
38     #Get sensor Data from DH11
39
40     temp=random.randint(0,100)
41     Humid=random.randint(0,100)
42     soilmoisture=random.randint(0,100)
43
44     data = { 'temp' : temp, 'Humid': Humid, 'soilmoisture':soilmoisture}
45     #print data
46     def myOnPublishCallback():
47         print ("published Temperature = %s C" % temp, "Humidity = is %s %" % Humid, "soilmoisture= is %s %" % soilmoisture,"to IBM Watson")
48
49 PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
published Temperature = 34 C Humidity = is 8 % soilmoisture= is 65 % to IBM Watson
published Temperature = 72 C Humidity = is 44 % soilmoisture= is 28 % to IBM Watson
published Temperature = 40 C Humidity = is 73 % soilmoisture= is 10 % to IBM Watson
published Temperature = 68 C Humidity = is 5 % soilmoisture= is 46 % to IBM Watson
published Temperature = 89 C Humidity = is 4 % soilmoisture= is 33 % to IBM Watson
published Temperature = 6 C Humidity = is 8 % soilmoisture= is 29 % to IBM Watson
published Temperature = 21 C Humidity = is 22 % soilmoisture= is 27 % to IBM Watson
published Temperature = 95 C Humidity = is 3 % soilmoisture= is 38 % to IBM Watson
published Temperature = 62 C Humidity = is 94 % soilmoisture= is 25 % to IBM Watson
published Temperature = 53 C Humidity = is 65 % soilmoisture= is 16 % to IBM Watson
published Temperature = 3 C Humidity = is 11 % soilmoisture= is 89 % to IBM Watson
published Temperature = 0 C Humidity = is 85 % soilmoisture= is 19 % to IBM Watson
Ln 6, Col 1 Spaces: 4 UTF-8 CRLF Python 3.9.13 (base: conda)
```



```
python.py - FDP model new - Visual Studio Code
C:\Users> SRIRAM > Downloads > python.py > ...
39
40 temp=random.randint(0,100)
41 Humid=random.randint(0,100)
42 soilmoisture=random.randint(0,100)
43
44 data = { 'temp' : temp, 'Humid': Humid, 'soilmoisture':soilmoisture}
45 #print data
46 def myOnPublishCallback():
47     print ("published Temperature = %s C" % temp, "Humidity = is %s %" % Humid, "soilmoisture= is %s %" % soilmoisture,"to IBM Watson")
48
49
50 success = deviceCli.publishEvent("IOTSensor", "json",data,qos=0,on_publish=myOnPublishCallback)
51 if not success:
52     print("Not connected to IOT")
53     time.sleep(5)
54
55
56 deviceCli.commandcallback = myCommandcallback
57
58 # Disconnect the device and application from the cloud
59 deviceCli.disconnect()
60
61 PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
published Temperature = 34 C Humidity = is 8 % soilmoisture= is 65 % to IBM Watson
published Temperature = 72 C Humidity = is 44 % soilmoisture= is 28 % to IBM Watson
published Temperature = 40 C Humidity = is 73 % soilmoisture= is 10 % to IBM Watson
published Temperature = 68 C Humidity = is 5 % soilmoisture= is 46 % to IBM Watson
published Temperature = 89 C Humidity = is 4 % soilmoisture= is 33 % to IBM Watson
published Temperature = 6 C Humidity = is 8 % soilmoisture= is 29 % to IBM Watson
published Temperature = 21 C Humidity = is 22 % soilmoisture= is 27 % to IBM Watson
published Temperature = 95 C Humidity = is 3 % soilmoisture= is 38 % to IBM Watson
published Temperature = 62 C Humidity = is 94 % soilmoisture= is 25 % to IBM Watson
published Temperature = 53 C Humidity = is 65 % soilmoisture= is 16 % to IBM Watson
published Temperature = 3 C Humidity = is 11 % soilmoisture= is 89 % to IBM Watson
published Temperature = 0 C Humidity = is 85 % soilmoisture= is 19 % to IBM Watson
Ln 6, Col 1 Spaces: 4 UTF-8 CRLF Python 3.9.13 (base: conda)
```

python.py - FDP model new - Visual Studio Code

```
1 import random
2 import sys
3 import time
4 import ibmiotf.application
5 import ibmiotf.device
6
7 #provide Your IBM Watson Device Credentials
8 organization = "9teu1"
9 deviceType = "SFITMS00"
10 deviceId = "SFITMS11"
11 authMethod = "token"
12 authToken = "PMTIBMSb18"
13
14
15 #Initialize GPIO
16 def myCommandCallback(cmd):
17     print("command received: %s" %cmd.data['command'])
18     status=cmd.data['command']
19     if status=="lighton":
20         print("led is on")
21     elif status == "lightoff":
22         print("led is off")
23     else:
24         print("please send proper command")
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER

published Temperature = 34 C Humidity = 15 % soilmoisture= 15 % to IBM Watson
published Temperature = 72 C Humidity = 44 % soilmoisture= 28 % to IBM Watson
published Temperature = 40 C Humidity = 73 % soilmoisture= 10 % to IBM Watson
published Temperature = 68 C Humidity = 5 % soilmoisture= 46 % to IBM Watson
published Temperature = 89 C Humidity = 4 % soilmoisture= 33 % to IBM Watson
published Temperature = 6 C Humidity = 8 % soilmoisture= 29 % to IBM Watson
published Temperature = 21 C Humidity = 22 % soilmoisture= 27 % to IBM Watson
published Temperature = 95 C Humidity = 3 % soilmoisture= 38 % to IBM Watson
published Temperature = 62 C Humidity = 94 % soilmoisture= 25 % to IBM Watson
published Temperature = 53 C Humidity = 65 % soilmoisture= 16 % to IBM Watson
published Temperature = 3 C Humidity = 11 % soilmoisture= 89 % to IBM Watson
published Temperature = 0 C Humidity = 85 % soilmoisture= 19 % to IBM Watson

Ln 6, Col 1 Spaces: 4 UTF-8 CRLF Python 3.9.13 (base: conda) 01:41 PM 16-11-2022