

SOURCE CODE

TEAM ID : PNT2022TMID38806

PROJECT TITLE : IoT BASED SMARTCROP PROTECTION SYSTEM
FOR AGRICULTURE

PYTHON SOURCE CODE:

```
import time
import sys

import ibmiotf.application
import ibmiotf.device

import random

#Provide your IBM Watson Device Credentials
organization = "c5ah4g"
deviceType = "App-1"
deviceId = "13"
authMethod = "use-token-auth"
authToken = "12345678"

# 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" into the cloud as
an event of type "greeting" 10 times deviceCli.connect()
```

```
while True:
```

```
    #Get Sensor Data from DHT11
```

```
    temp=random.randint(90,110)
```

```
    Humid=random.randint(60,100)
```

```
    Moist=random.randint(20,100)
```

```
    Animal_dect=random.randint(1,20)
```

```
    data = { 'temp' : temp, 'Humid': Humid, 'Moist' : Moist, 'Animal_dect' :
Animal_dect }
```

```
    #print data      def
```

```
myOnPublishCallback():
```

```
    print ("Published Temperature = %s C" % temp, "Humidity = %s
```

```
%%" % Humid, "to IBM Watson", "Published Moisture= %s" % Moist,  
"Published Animal detection = ", Animal_dect)
```

```
    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,  
on_publish=myOnPublishCallback)    if not success:  
print("Not connected to IoT")    time.sleep(10)
```

```
deviceCli.commandCallback = myCommandCallback
```

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

NODE-RED SOURCE CODE:

TEMPERATURE:

```
msg.payload=msg.payload."temp"
```

```
return msg; HUMIDITY:
```

```
msg.payload=msg.payload."Humid"
```

```
return msg; MOISTURE:
```

```
msg.payload=msg.payload."Moist"
```

```
return msg;
```

ANIMAL DETECTION:

```
msg.payload=msg.payload."Animal_dect" return
```

```
msg;
```