DEVELOP A PYTHON SCRIPT

TEAM ID	**PNT2022TMID27900**
PROJECT NAME	**IOT BASED SMART CROP PROTECTIONSYSTEM FOR AGRICULTURE**

PROGRAM:

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

#Provide your IBM Watson Device Credentials
organization = "fl9pj"
deviceType = "NodeMCU"
deviceId = "12345"
authMethod = "token"
authToken = "oZDx3WdXYxtZ3Ixd-c"

# Initialize GPIO
def myCommandCallback(cmd):
    print("Command received: %s" % cmd.data['command'])
    status=cmd.data['command']
    if status=="sprinkleron":
        print ("sprinkler is on")
    elif status == "sprinkleroff":
        print ("sprinkler 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()
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11
         temp=random.randint(40,90)
        Humid=random.randint(60,100)
        Moisture=random.randint(80,100)
        data = { 'temp' : temp, 'Humid': Humid, 'Moisture': Moisture }
        #print data
        def myOnPublishCallback():
```

```
            print ("Published Temperature = %s C" % temp, "Humidity = %s %%" %
Humid,"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 the device and application from the cloud
deviceCli.disconnect()
```