Develop a Python Script To Publish And Subscribe To IBM IoT Platform

Team ID	PNT2022TMID38150
Project Name	Smart Farmer – IoT Enabled Smart Farming Application
Team Leader	DILLI BABU S
Team Members	ABIMANYU JE MANIMEGALAI D JAYASHREE D

CODE:

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


#Provide your IBM Watson Device Credentials
organization = "1jk4ps" deviceType
= "PNT2022TMID51719" deviceId =
"Smart_Farmer" authMethod =
"token"
authToken = "l1*53hCIhmEbf!&Es&"

# Initialize GPIO
```

```
def myCommandCallback(cmd): print("Command received:
   %s"                    %                    cmd.data['command'])
   status=cmd.data['command'] if status=="lighton":
      print ("led is on")
   else : print ("led is
      off")

   #print(cmd)




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)

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

```
        success = deviceCli.publishEvent("SDFRN", "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()
```

OUTPUT: