

```

import time
import sys
import ibmiotf.device
import random

# Provide your IBM Watson Device Credentials
organization = "rnxod0"
deviceType = "iotsensor" # Credentials of Watson IoT sensor simulator
deviceId = "123456"
authMethod = "token"
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")
    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

    lat = random.uniform(-90, 90)
    lon = random.uniform(-180, 180)

```

```
data = {'lat': lat, 'log': lon}

# print data
def myOnPublishCallback():
    print("Published Latitude = %s " % lat, "Longitude = %s %" % lon,
"to IBM Watson")

    success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0,
on_publish=myOnPublishCallback)
    if not success:
        print("Not connected to IoTTF")
        time.sleep(1)

    deviceCli.commandCallback = myCommandCallback

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