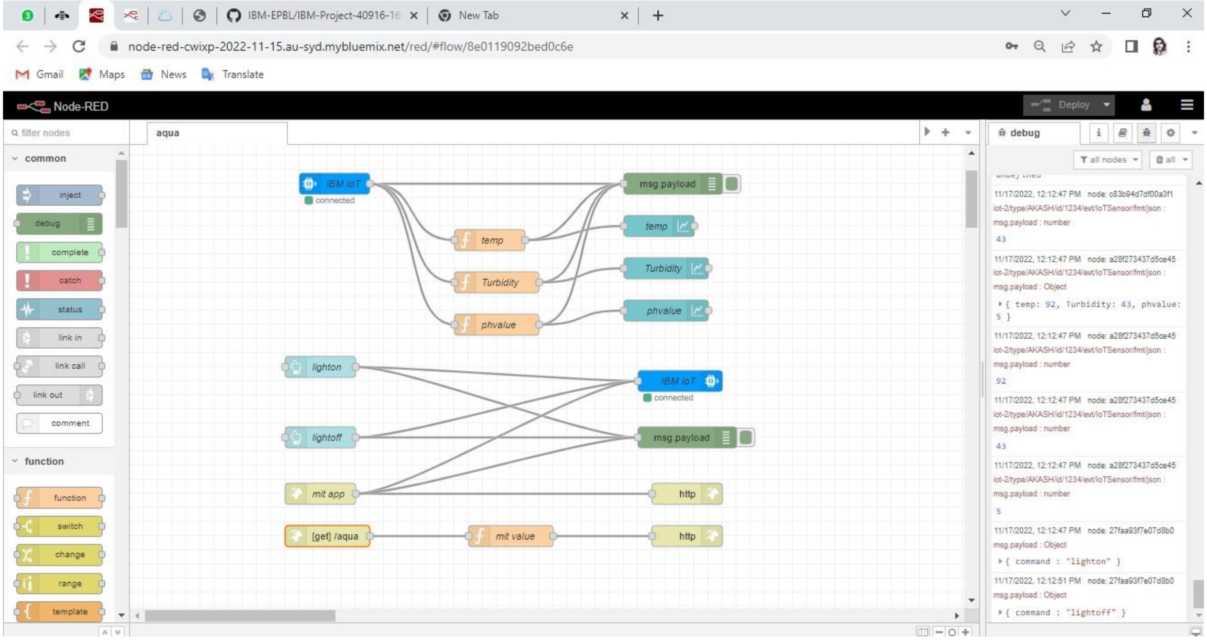
USE DASHBOARD NODES FOR CREATING UI WEB APP

DATE	17 November 2022
TEAM ID	PNT2022TMID12856
PROJECT NAME	Real-Time River Water Monitoring and Control System

Figure 1: when the light on and light off buttons are pressed



```
11/17/2022, 12:12:47 PM  node: c83b94d7df00a3f1
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
43
11/17/2022, 12:12:47 PM  node: a28f273437d5ce45
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : Object
▸{ temp: 92, Turbidity: 43, phvalue:
5 }
11/17/2022, 12:12:47 PM  node: a28f273437d5ce45
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
92
11/17/2022, 12:12:47 PM  node: a28f273437d5ce45
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
43
11/17/2022, 12:12:47 PM  node: a28f273437d5ce45
iot-2/type/AKASH/id/1234/evt/IoTSensor/fmt/json :
msg.payload : number
5
11/17/2022, 12:12:47 PM  node: 27faa93f7e07d8b0
msg.payload : Object
▸{ command : "lighton" }
11/17/2022, 12:12:51 PM  node: 27faa93f7e07d8b0
msg.payload : Object
▸{ command : "lightoff" }
```

Figure 2: WEB UI Created using NodeRed Dashboard

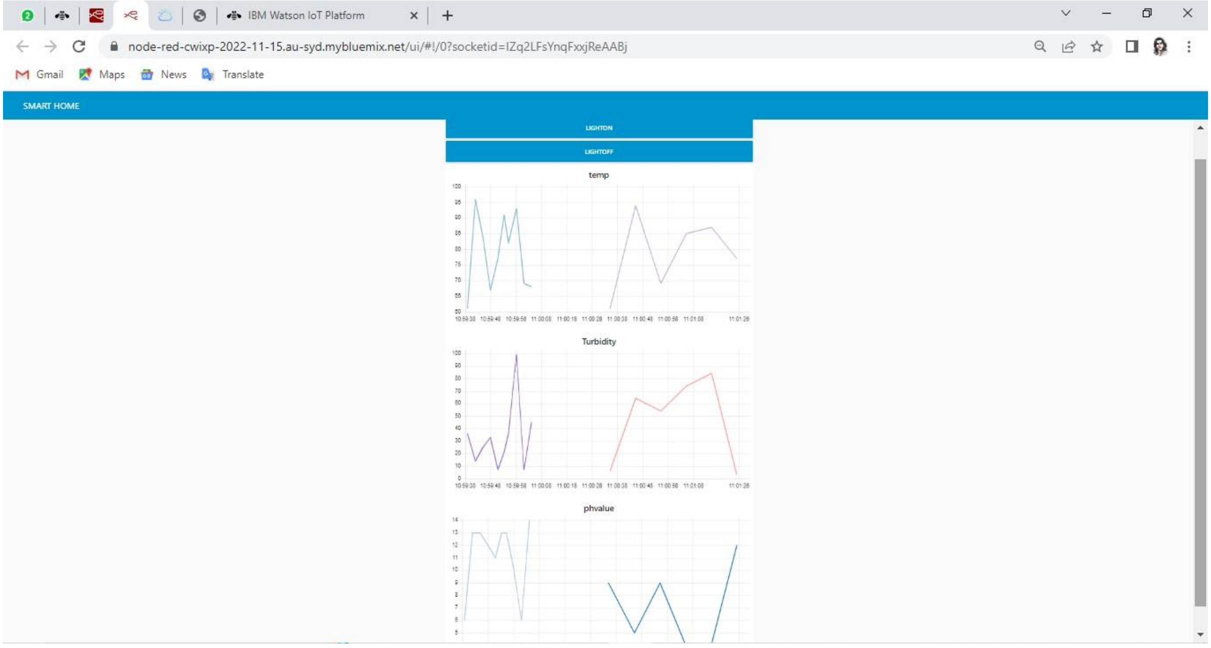
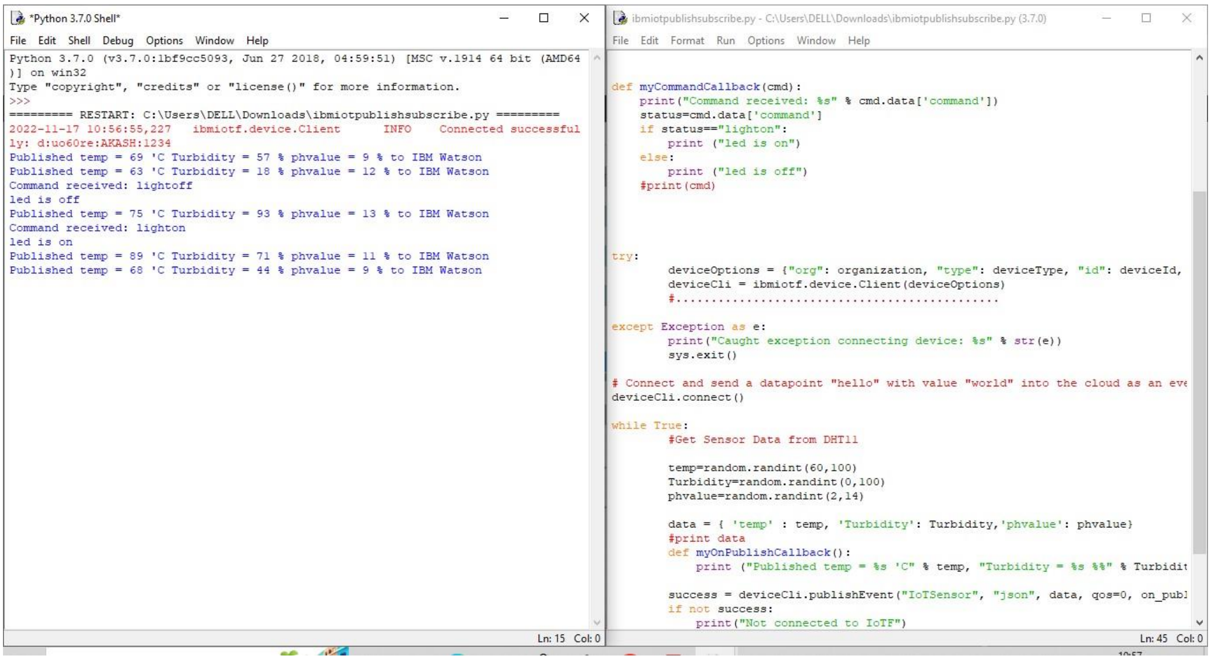


Figure 3: Python 3.7.0 software output when the LIGHT ON and LIGHT OFF buttons on the WEB UI is pressed



Python 3.7.0 Shell
File Edit Shell Debug Options Window Help
Ln: 15 Col: 0
ibmiotpublishsubscribe.py - C:\Users\DELL\Downloads\ibmiotpublishsubscribe.py (3.7.0)
File Edit Format Run Options Window Help
Ln: 45 Col: 0

```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:59:51) [MSC v.1914 64 bit (AMD64
)] on win32
Type "copyright", "credits" or "license()" for more information.
>>>
========= RESTART: C:\Users\DELL\Downloads\ibmiotpublishsubscribe.py =========
2022-11-17 10:56:55,227   ibmiotf.device.Client      INFO    Connected successful
ly: d:uo60re:AKASH:1234
Published temp = 69 'C Turbidity = 57 % phvalue = 9 % to IBM Watson
Published temp = 63 'C Turbidity = 18 % phvalue = 12 % to IBM Watson
Command received: lightoff
led is off
Published temp = 75 'C Turbidity = 93 % phvalue = 13 % to IBM Watson
Command received: lighton
led is on
Published temp = 89 'C Turbidity = 71 % phvalue = 11 % to IBM Watson
Published temp = 68 'C Turbidity = 44 % phvalue = 9 % to IBM Watson
```

```
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,
        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 eve
deviceCli.connect()

while True:
        #Get Sensor Data from DHT11

        temp=random.randint(60,100)
        Turbidity=random.randint(0,100)
        phvalue=random.randint(2,14)

        data = { 'temp' : temp, 'Turbidity': Turbidity,'phvalue': phvalue}
        #print data
        def myOnPublishCallback():
            print ("Published temp = %s 'C" % temp, "Turbidity = %s %%" % Turbidit

        success = deviceCli.publishEvent("IoTSensor", "json", data, qos=0, on_publ
        if not success:
            print("Not connected to IoTF")
```