

Smart Bridge Externship Program

Internet of Things

Name: Ruthala Meher Bhavana

Reg.No: 19BEE0242

Mail Id: ruthalameher.bhavana2019@vitstudent.ac.in

Assignment – 3

Q. Develop a code to upload the water tank level and light intensity values to the IBM IoT platform and visualize them in the web application.

Python Code:

```
import wiotp.sdk.device
import time
import random

myConfig = {
    "identity": {
        "orgId": "cvtwoa",
        "typeId": "Device1",
        "deviceId": "mydev123"
    },
    "auth": {
```

```
    "token": "123456789"
  }
}
```

```
def myCommandCallback(cmd):
```

```
    print("Message received from IBM IoT Platform: %s" %
cmd.data['command'])
```

```
    m=cmd.data['command']
```

```
client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
```

```
client.connect()
```

```
while True:
```

```
    waterlevel=random.randint(0,100)
```

```
    lightintensity=random.randint(0,100)
```

```
    myData={'WaterLevel':waterlevel, 'LightIntensity':lightintensity}
```

```
    client.publishEvent(eventId="status", msgFormat="json", data=myData,
qos=0, onPublish=None)
```

```
    print("Published data Successfully: %s", myData)
```

```
    client.commandCallback = myCommandCallback
```

```
    time.sleep(2)
```

```
client.disconnect()
```

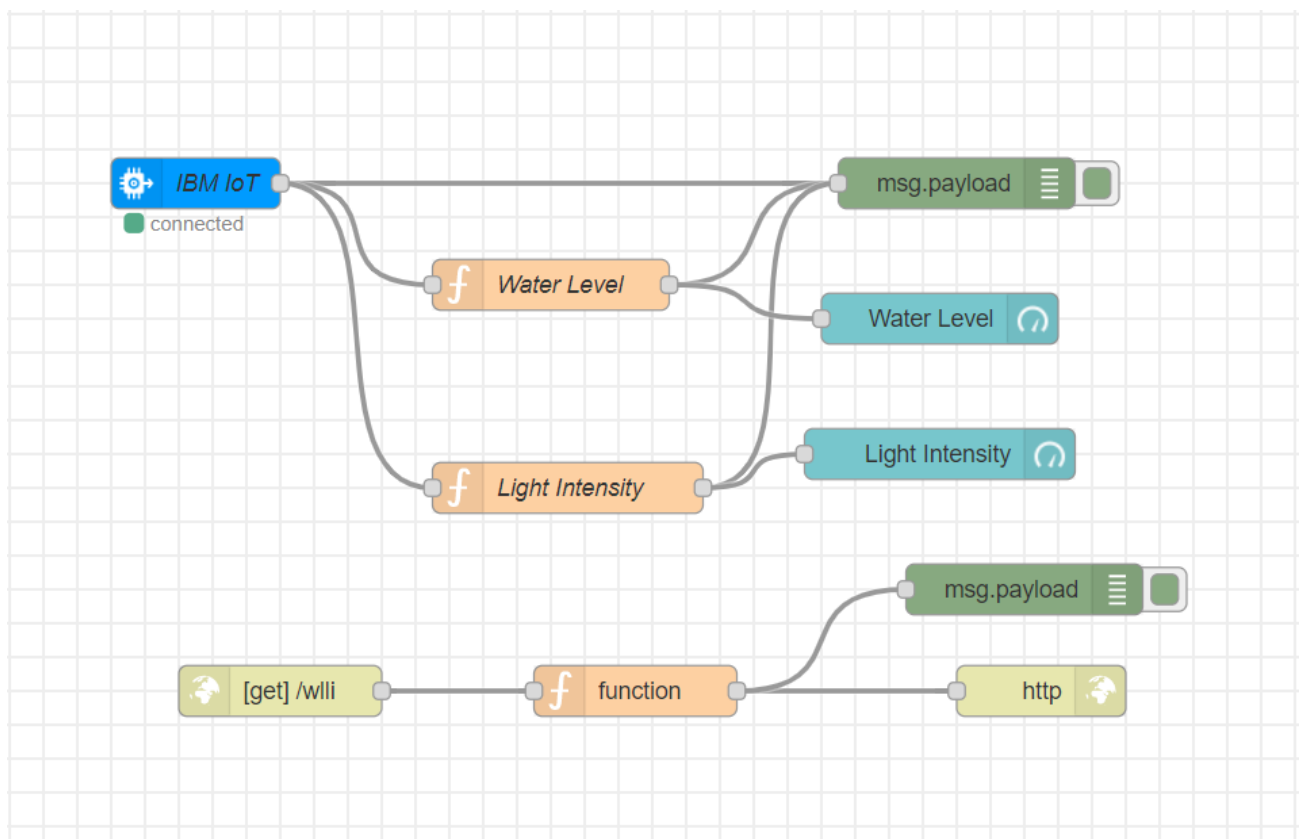
```
Assignment3-code.py - D:/SMART INTERNZ/Assignments/Assignment 3/Assignment3-code.py (3.9.6)
File Edit Format Run Options Window Help
import wiotp.sdk.device
import time
import random
myConfig = {
    "identity": {
        "orgId": "cvtwoa",
        "typeId": "Device1",
        "deviceId": "mydev123"
    },
    "auth": {
        "token": "123456789"
    }
}

def myCommandCallback(cmd):
    print("Message received from IBM IoT Platform: %s" % cmd.data['command'])
    m=cmd.data['command']

client = wiotp.sdk.device.DeviceClient(config=myConfig, logHandlers=None)
client.connect()

while True:
    waterlevel=random.randint(0,100)
    lightintensity=random.randint(0,100)
    myData={'WaterLevel':waterlevel, 'LightIntensity':lightintensity}
    client.publishEvent(eventId="status", msgFormat="json", data=myData, qos=0, onPublish=None)
    print("Published data Successfully: %s", myData)
    client.commandCallback = myCommandCallback
    time.sleep(2)
client.disconnect()
|
```

NodeRed flow:



After running the code, values returned to the debug console:

The screenshot shows the Node-RED web interface. The main workspace displays a flow with the following components:

- Flow 1:** An **IBM IoT** node (connected) feeds into two function nodes: **Water Level** and **Light Intensity**.
- Flow 2:** A **[get] /wli** node feeds into a **function** node, which then feeds into an **http** node.

The **debug** console on the right shows the following log entries:

```
8/2/2021, 2:44:50 AM node: 82955261.9e90c
iot-2/type/Device1/Id/mydev123/evt/status/fmt/json :
msg.payload : Object
  { WaterLevel: 1, LightIntensity: 43 }

8/2/2021, 2:44:50 AM node: 82955261.9e90c
iot-2/type/Device1/Id/mydev123/evt/status/fmt/json :
msg.payload : number
1

8/2/2021, 2:44:50 AM node: 82955261.9e90c
iot-2/type/Device1/Id/mydev123/evt/status/fmt/json :
msg.payload : number
43

8/2/2021, 2:44:50 AM node: 82955261.9e90c
iot-2/type/Device1/Id/mydev123/evt/status/fmt/json :
msg.payload : Object
  { WaterLevel: 68, LightIntensity: 34 }

8/2/2021, 2:44:50 AM node: 82955261.9e90c
iot-2/type/Device1/Id/mydev123/evt/status/fmt/json :
msg.payload : number
68

8/2/2021, 2:44:50 AM node: 82955261.9e90c
iot-2/type/Device1/Id/mydev123/evt/status/fmt/json :
msg.payload : number
34
```

Values returned to IBM Watson IoT Platform:

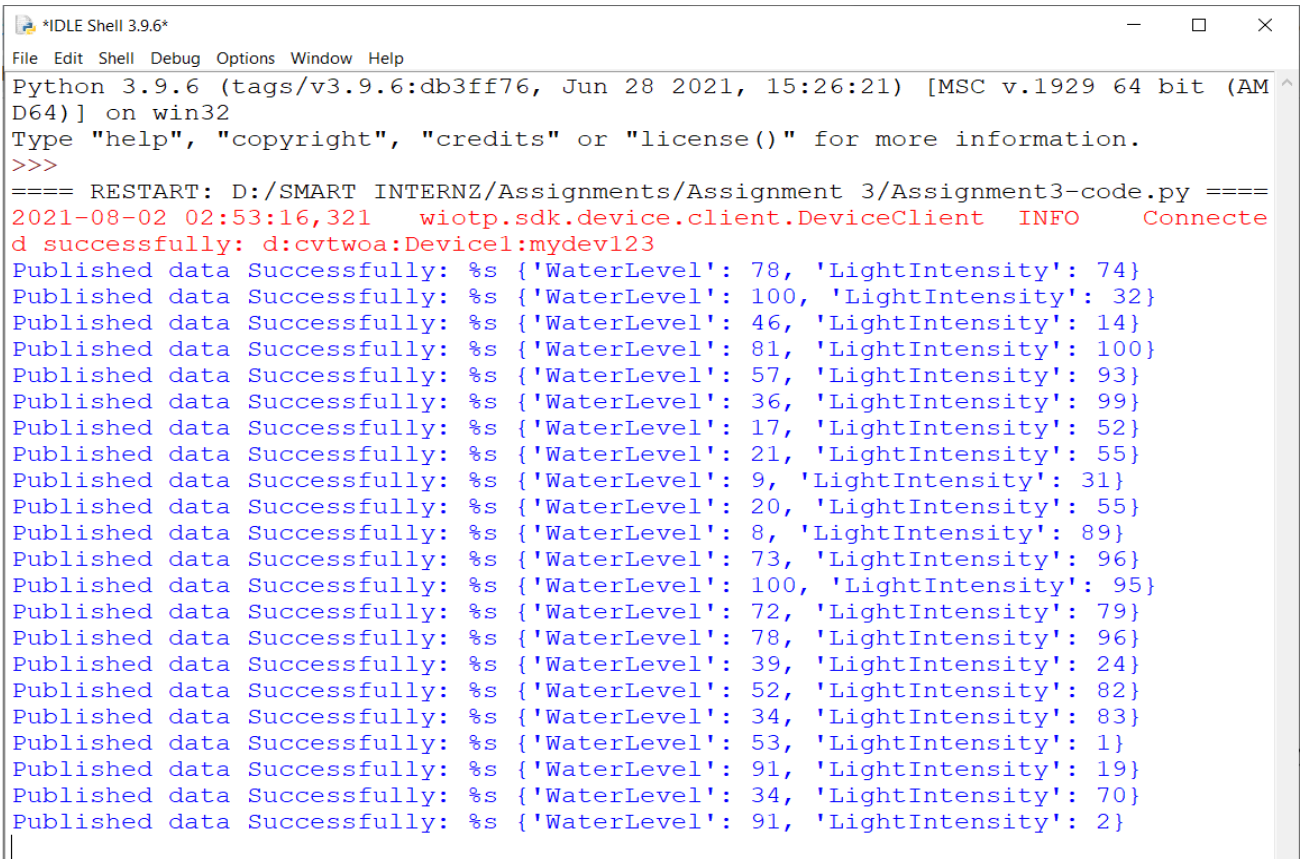
The screenshot shows the IBM Watson IoT Platform dashboard. The main section displays a table of devices, with the following data:

Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
mydev123	Connected	Device1	Device	Jul 20, 2021 4:15 PM	

Below the table, the **Recent Events** tab is selected, showing a list of events:

Event	Value	Format	Last Received
status	{"WaterLevel":71,"LightIntensity":24}	json	a few seconds ago
status	{"WaterLevel":70,"LightIntensity":46}	json	a few seconds ago
status	{"WaterLevel":54,"LightIntensity":24}	json	a few seconds ago
status	{"WaterLevel":2,"LightIntensity":29}	json	a few seconds ago
status	{"WaterLevel":16,"LightIntensity":40}	json	a few seconds ago

Values returned to Python Shell:



The screenshot shows a Python IDLE Shell window with the following content:

```
*IDLE Shell 3.9.6*
File Edit Shell Debug Options Window Help
Python 3.9.6 (tags/v3.9.6:db3ff76, Jun 28 2021, 15:26:21) [MSC v.1929 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: D:/SMART INTERNZ/Assignments/Assignment 3/Assignment3-code.py =====
2021-08-02 02:53:16,321 wiotp.sdk.device.client.DeviceClient INFO Connected successfully: d:cvtwoa:Device1:mydev123
Published data Successfully: %s {'WaterLevel': 78, 'LightIntensity': 74}
Published data Successfully: %s {'WaterLevel': 100, 'LightIntensity': 32}
Published data Successfully: %s {'WaterLevel': 46, 'LightIntensity': 14}
Published data Successfully: %s {'WaterLevel': 81, 'LightIntensity': 100}
Published data Successfully: %s {'WaterLevel': 57, 'LightIntensity': 93}
Published data Successfully: %s {'WaterLevel': 36, 'LightIntensity': 99}
Published data Successfully: %s {'WaterLevel': 17, 'LightIntensity': 52}
Published data Successfully: %s {'WaterLevel': 21, 'LightIntensity': 55}
Published data Successfully: %s {'WaterLevel': 9, 'LightIntensity': 31}
Published data Successfully: %s {'WaterLevel': 20, 'LightIntensity': 55}
Published data Successfully: %s {'WaterLevel': 8, 'LightIntensity': 89}
Published data Successfully: %s {'WaterLevel': 73, 'LightIntensity': 96}
Published data Successfully: %s {'WaterLevel': 100, 'LightIntensity': 95}
Published data Successfully: %s {'WaterLevel': 72, 'LightIntensity': 79}
Published data Successfully: %s {'WaterLevel': 78, 'LightIntensity': 96}
Published data Successfully: %s {'WaterLevel': 39, 'LightIntensity': 24}
Published data Successfully: %s {'WaterLevel': 52, 'LightIntensity': 82}
Published data Successfully: %s {'WaterLevel': 34, 'LightIntensity': 83}
Published data Successfully: %s {'WaterLevel': 53, 'LightIntensity': 1}
Published data Successfully: %s {'WaterLevel': 91, 'LightIntensity': 19}
Published data Successfully: %s {'WaterLevel': 34, 'LightIntensity': 70}
Published data Successfully: %s {'WaterLevel': 91, 'LightIntensity': 2}
```

Same Values in NodeRed Debug Console and NodeRed Web UI:



The screenshot shows the NodeRed Debug Console interface. At the top, there is a toolbar with icons for 'debug', 'info', 'logs', 'nodes', 'settings', and 'storage'. Below the toolbar, there is a filter button labeled 'all nodes' and a trash icon. The main area displays a list of messages. Each message entry includes a timestamp, a node ID, a topic, and a payload. The payloads are JSON objects containing 'WaterLevel' and 'LightIntensity' values. The messages are grouped by node ID, with the first group (node: 82955261.9e90c) showing three messages and the second group (node: 1478ea14.1d3956) showing one message.

Timestamp	Node ID	Topic	Payload
8/2/2021, 2:44:50 AM	82955261.9e90c	iot-2/type/Device1/id/mydev123/evt/status/fmt/json	{ WaterLevel: 1, LightIntensity: 43 }
8/2/2021, 2:44:50 AM	82955261.9e90c	iot-2/type/Device1/id/mydev123/evt/status/fmt/json	1
8/2/2021, 2:44:50 AM	82955261.9e90c	iot-2/type/Device1/id/mydev123/evt/status/fmt/json	43
8/2/2021, 2:44:50 AM	82955261.9e90c	iot-2/type/Device1/id/mydev123/evt/status/fmt/json	{ WaterLevel: 68, LightIntensity: 34 }
8/2/2021, 2:44:50 AM	82955261.9e90c	iot-2/type/Device1/id/mydev123/evt/status/fmt/json	68
8/2/2021, 2:44:50 AM	82955261.9e90c	iot-2/type/Device1/id/mydev123/evt/status/fmt/json	34
8/2/2021, 2:45:33 AM	1478ea14.1d3956	iot-2/type/Device1/id/mydev123/evt/status/fmt/json	{ Water Level: 68, Light Intensity: 34 }

NodeRed Web UI:

