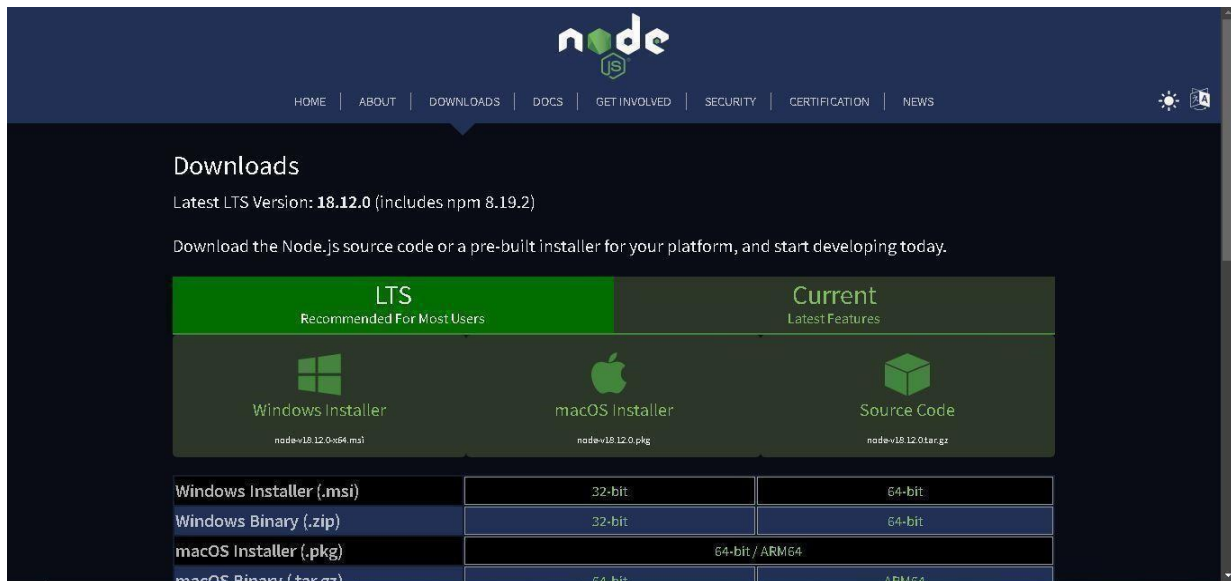


# SPRINT 2

<b>Date</b>	November 13, 2022
<b>Team ID</b>	PNT2022TMID13167
<b>Project Name</b>	Real-Time River Water Quality Monitoring and Control System
<b>Maximum Mark</b>	

**The following steps are involved:**

**STEP 1:** Download and Install node.js.



**STEP 2:** Setup node.js and configure command prompt for error check. Open node-red from the generated link.

```
npm
C:\Users\Ajay>npm install -g --unsafe-perm node-red
npm WARN deprecated @types/keyv@4.2.0: This is a stub types definition. keyv provides its own type definitions, so you do not need this installed.

added 292 packages, and audited 293 packages in 5m

39 packages are looking for funding
  run `npm fund` for details

5 vulnerabilities (4 low, 1 moderate)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.

C:\Users\Ajay>node-red
B Nov 14:35:28 - [info]

Welcome to Node-RED
=====
B Nov 14:35:29 - [info] Node-RED version: v3.0.2
B Nov 14:35:29 - [info] Node.js version: v18.12.0
B Nov 14:35:29 - [info] Windows_NT 10.0.19043 x64 LE
B Nov 14:35:44 - [info] Loading palette nodes
```

### STEP 3: Generating API key and Authentication token.


Browse

IBM Cloud Apps

## The API key has been added.

Authentication tokens are non-recoverable. If you misplace this token, you will need to re-register the API key to generate a new authentication token.

Generated Details		API Key Information	
API Key	a-2208jk-3wkprnmnx	Description	iot_sensor
Authentication Token	nWAY)i(H(hF0cDAxN0	Role	Standard Application
		Expires	Never



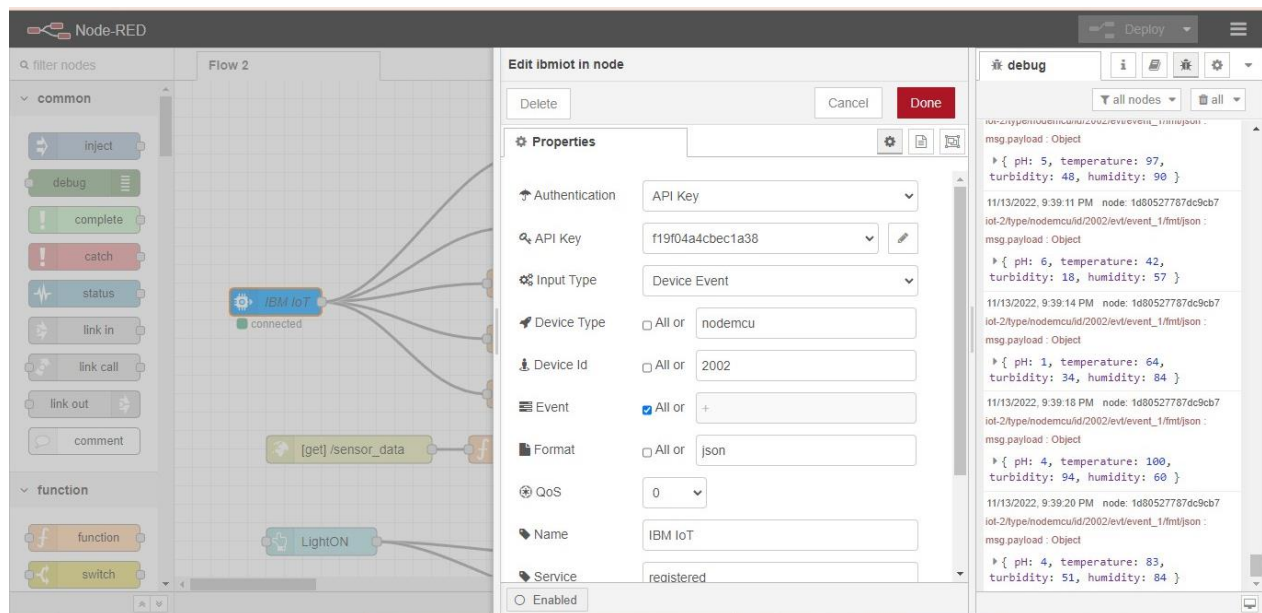
Make a note of the generated authentication token. Lost authentication tokens cannot be recovered. If you lose the token, you must reregister the API to generate a new token.

View API Key

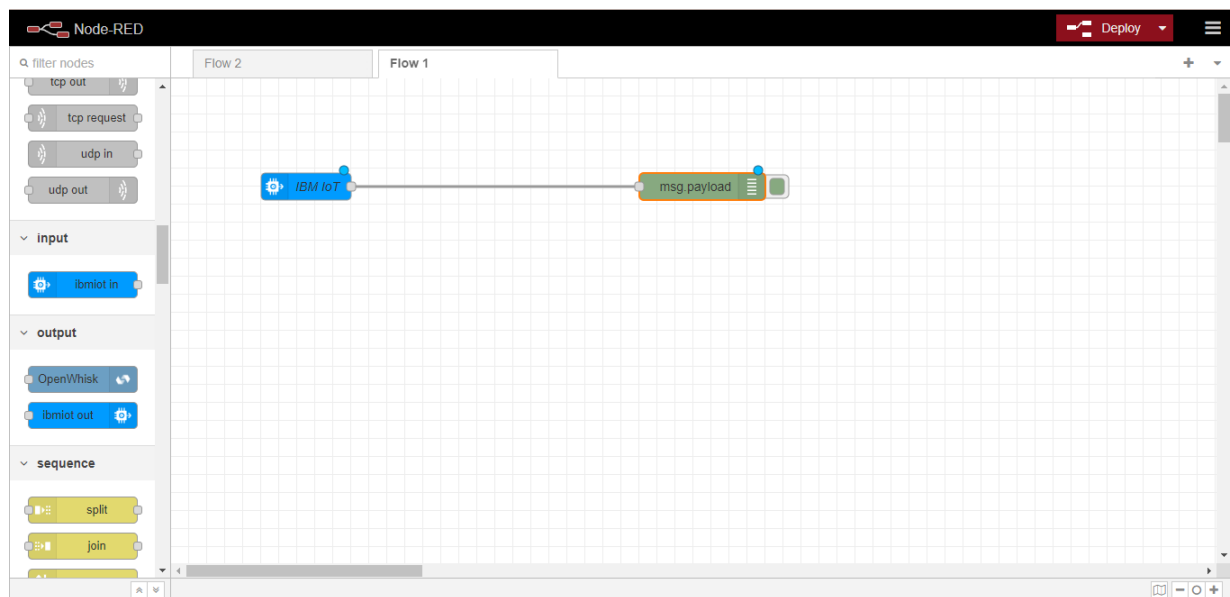
Add Another

Close

### STEP 4: Edit IbmIoT in node.



**STEP 5:** Connect Ibmiot in and debug 1 and deploy.



## STEP 6: Edit gauge node (here the gauge nodes are named as Temperature, pH and Turbidity).

The screenshot shows the Node-RED web interface. In the center workspace, a flow named 'Flow 2' is visible, featuring an 'IBM IoT' node connected to a '[get] /sensor\_data' node, which is then connected to a 'LightON' node. On the left sidebar, the 'output' section contains 'ibmiot out' and 'OpenWhisk'. The right sidebar shows the 'debug' console with a log of JSON messages from the IoT node, including pH, temperature, and turbidity data.

**Edit gauge node**

Properties

- Group: [Home] iot\_sensor\_Analytics
- Size: auto
- Type: Gauge
- Label: pH
- Value format: {{value}}
- Units: units
- Range: min 0 max 7
- Colour gradient: [Green, Yellow, Red]
- Sectors: 0 ... optional ... optional ... 7
- Class: Optional CSS class name(s) for widget
- Enabled: ☐

**debug**

```
{ "pH": 6, "temperature": 76, "turbidity": 73, "humidity": 20 }
```

```
{ "pH": 7, "temperature": 55, "turbidity": 12, "humidity": 84 }
```

```
{ "pH": 5, "temperature": 1, "turbidity": 3, "humidity": 5 }
```

```
{ "pH": 2, "temperature": 95, "turbidity": 17, "humidity": 57 }
```

```
{ "pH": 0, "temperature": 38, "turbidity": 100, "humidity": 80 }
```

This screenshot shows the same Node-RED interface as the previous one, but the 'Edit gauge node' configuration is now for a temperature gauge. The 'Label' is set to 'temperature' and the 'Units' are set to 'Degree celcius'. The 'Range' is updated to 'min 0 max 100'. The 'debug' console shows a new set of JSON messages with updated temperature and humidity values.

**Edit gauge node**

Properties

- Group: [Home] iot\_sensor\_Analytics
- Size: auto
- Type: Gauge
- Label: temperature
- Value format: {{value}}
- Units: Degree celcius
- Range: min 0 max 100
- Colour gradient: [Green, Yellow, Red]
- Sectors: 0 ... optional ... optional ... 100
- Class: Optional CSS class name(s) for widget
- Enabled: ☐

**debug**

```
{ "pH": 1, "temperature": 41, "turbidity": 38, "humidity": 8 }
```

```
{ "pH": 4, "temperature": 75, "turbidity": 46, "humidity": 4 }
```

```
{ "pH": 5, "temperature": 28, "turbidity": 50, "humidity": 45 }
```

```
{ "pH": 3, "temperature": 68, "turbidity": 72, "humidity": 10 }
```

```
{ "pH": 4, "temperature": 0, "turbidity": 23, "humidity": 2 }
```

Node-RED interface showing a flow configuration for an IBM IoT sensor. The flow includes an IBM IoT node connected to a [get] /sensor\_data node, which triggers a LightON node. The Edit gauge node is configured with the following properties:

- Group: [Home] iot\_sensor\_Analytics
- Size: auto
- Type: Gauge
- Label: Humidity
- Value format: {{value}}
- Units: %
- Range: min 0, max 100
- Colour gradient: Green, Yellow, Red
- Sectors: 0, optional, optional, 100
- Class: Optional CSS class name(s) for widget
- Enabled: ☐

The debug console shows the following log entries:

```
11/13/2022, 9:43:00 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 0,  
  temperature: 84,  
  turbidity: 19,  
  humidity: 77  
}  
11/13/2022, 9:43:03 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 5,  
  temperature: 77,  
  turbidity: 67,  
  humidity: 47  
}  
11/13/2022, 9:43:06 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 5,  
  temperature: 34,  
  turbidity: 62,  
  humidity: 41  
}  
11/13/2022, 9:43:09 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 2,  
  temperature: 6,  
  turbidity: 80,  
  humidity: 19  
}  
11/13/2022, 9:43:09 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 6,  
  temperature: 43,  
  turbidity: 97,  
  humidity: 80  
}
```

Node-RED interface showing the same flow configuration, but the Edit gauge node is now configured for Turbidity. The flow includes an IBM IoT node connected to a [get] /sensor\_data node, which triggers a LightON node. The Edit gauge node is configured with the following properties:

- Group: [Home] iot\_sensor\_Analytics
- Size: auto
- Type: Gauge
- Label: Turbidity
- Value format: {{value}}
- Units: NTU
- Range: min 0, max 100
- Colour gradient: Green, Yellow, Red
- Sectors: 0, optional, optional, 100
- Class: Optional CSS class name(s) for widget
- Enabled: ☐

The debug console shows the following log entries:

```
11/13/2022, 9:43:18 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 5,  
  temperature: 54,  
  turbidity: 68,  
  humidity: 55  
}  
11/13/2022, 9:43:21 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 5,  
  temperature: 99,  
  turbidity: 24,  
  humidity: 53  
}  
11/13/2022, 9:43:24 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 3,  
  temperature: 86,  
  turbidity: 18,  
  humidity: 45  
}  
11/13/2022, 9:43:27 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 4,  
  temperature: 8,  
  turbidity: 23,  
  humidity: 49  
}  
11/13/2022, 9:43:27 PM node:1d80527787dc9cb7  
iot-2/type/nodemcu/2002/evt/event_1/fmt/json : msg.payload :  
Object  
{  
  pH: 3,  
  temperature: 54,  
  turbidity: 0,  
  humidity: 33  
}
```

## STEP 7: Simulated program to get the random values.

The screenshot displays the IBM Watson IoT Platform interface. On the left, a sidebar contains navigation icons. The main panel shows a table of devices with columns for Device ID, Status, and Device Type. Two devices are listed: ID 123456 (Disconnected, iotsura) and ID 2002 (Disconnected, nodemcu). The 'Recent Events' tab is selected for device 2002. On the right, a configuration window for 'Device Type: nodemcu' is open. It shows an event type named 'event\_1' with a schedule of '20' every minute. The payload is a JSON object with random values for pH, temperature, turbidity, and humidity.

Device ID	Status	Device Type
123456	Disconnected	iotsura
2002	Disconnected	nodemcu

Event type name: event\_1

Schedule: 20 Every Minute

Payload:

```
{
  "pH": random(0, 7),
  "temperature": random(0, 100),
  "turbidity": random(0, 100),
  "humidity": random(0, 100)
}
```

## STEP 8: Generate debug message from IBM Watson IoT Platform and connect the nodes.

The screenshot shows the Node-RED interface. A flow diagram is visible with a 'msg payload' node connected to four function nodes labeled 'pH', 'temperature', 'humidity', and 'turbidity'. These function nodes are connected to corresponding output nodes. Below this, there are nodes for 'LightON', 'LightOFF', and 'LightON', each connected to an 'IBM IoT' node. The 'debug' console on the right displays a series of messages, including the payload object and the resulting JSON data for each event.

```
msg payload: Object
{
  pH: 6, temperature: 18, turbidity: 62, humidity: 71
}

11/13/2022, 9:46:58 PM node: 1d80527787dc9cb7
iot-2/type/nodemcu/id/2002/evt/event_1/fmt/json:
msg payload: Object
{
  pH: 4, temperature: 45, turbidity: 33, humidity: 50
}

11/13/2022, 9:47:03 PM node: 1d80527787dc9cb7
iot-2/type/nodemcu/id/2002/evt/event_1/fmt/json:
msg payload: Object
{
  pH: 1, temperature: 14, turbidity: 2, humidity: 3
}

11/13/2022, 9:47:07 PM node: 1d80527787dc9cb7
iot-2/type/nodemcu/id/2002/evt/event_1/fmt/json:
msg payload: Object
{
  pH: 2, temperature: 65, turbidity: 12, humidity: 51
}

11/13/2022, 9:47:09 PM node: 1d80527787dc9cb7
iot-2/type/nodemcu/id/2002/evt/event_1/fmt/json:
msg payload: Object
{
  pH: 3, temperature: 78, turbidity: 22, humidity: 53
}
```



## STEP 9: Edit button mode [light ON and light OFF].

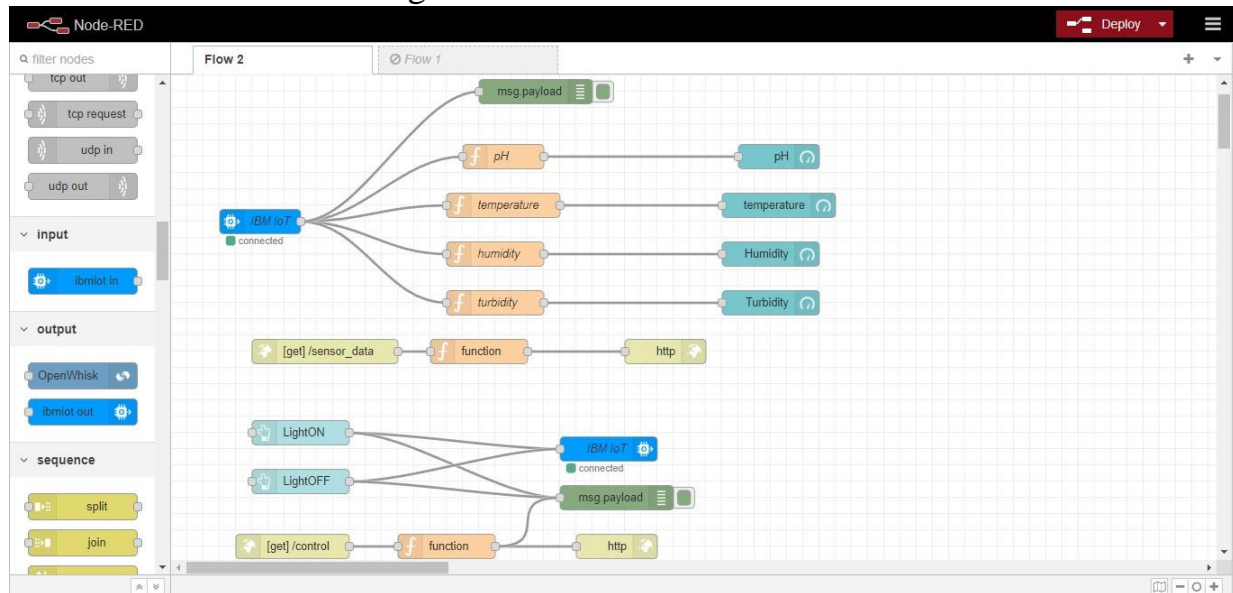
The screenshot shows the Node-RED interface with the 'Edit button node' dialog open for the 'LightON' button. The dialog is titled 'Edit button node' and has a 'Done' button. The 'Properties' section includes:

- Group: [Home] Control\_Dashboard
- Size: auto
- Icon: optional icon
- Label: LightON
- Tooltip: optional tooltip
- Color: optional text/icon color
- Background: optional background color
- When clicked, send: Payload: lighton, Topic: msg.topic

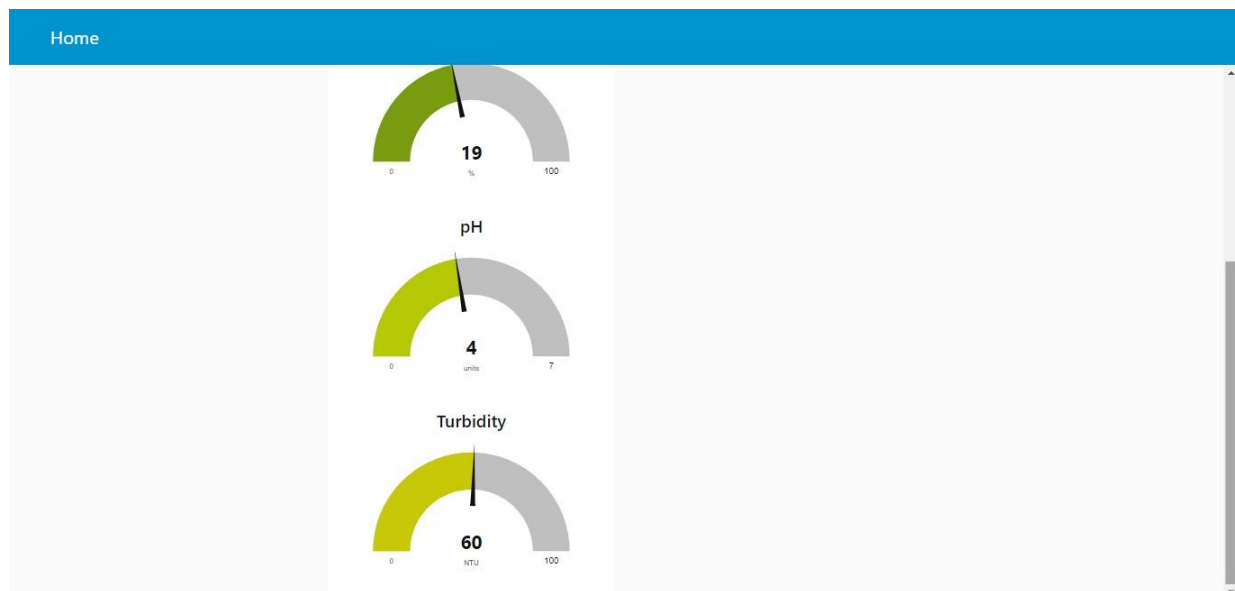
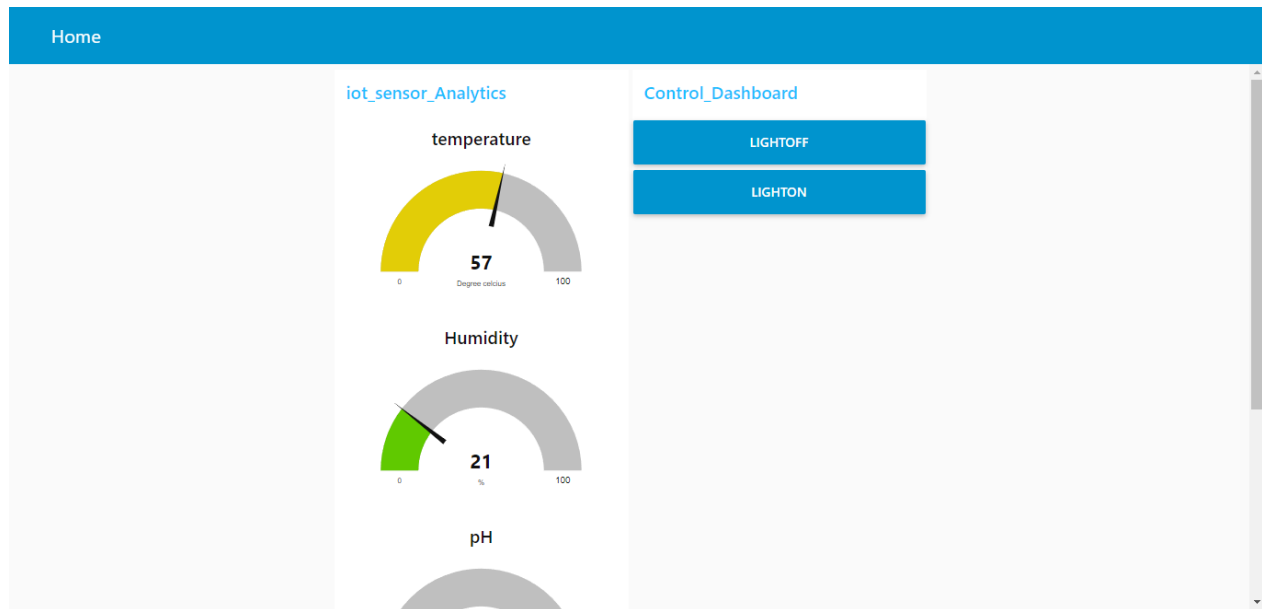
The 'debug' console on the right shows the following log entries:

```
11/13/2022, 9:48:24 PM node: 1d80527787dc9cb7  
iot-2-hypemodemcu/id/2002/evl/event_1/fmt/json :  
msg.payload : Object  
  { pH: 4, temperature: 18, turbidity: 26, humidity: 9 }  
  
11/13/2022, 9:48:27 PM node: 1d80527787dc9cb7  
iot-2-hypemodemcu/id/2002/evl/event_1/fmt/json :  
msg.payload : Object  
  { pH: 1, temperature: 2, turbidity: 79, humidity: 19 }  
  
11/13/2022, 9:48:27 PM node: 1d80527787dc9cb7  
iot-2-hypemodemcu/id/2002/evl/event_1/fmt/json :  
msg.payload : Object  
  { pH: 3, temperature: 55, turbidity: 13, humidity: 55 }  
  
11/13/2022, 9:48:31 PM node: 1d80527787dc9cb7  
iot-2-hypemodemcu/id/2002/evl/event_1/fmt/json :  
msg.payload : Object  
  { pH: 0, temperature: 67, turbidity: 63, humidity: 84 }  
  
11/13/2022, 9:48:33 PM node: 1d80527787dc9cb7  
iot-2-hypemodemcu/id/2002/evl/event_1/fmt/json :  
msg.payload : Object  
  { pH: 0, temperature: 99, turbidity: 16, humidity: 18 }
```

## STEP 10: Entire flow diagram in Node-RED.



## STEP 11: Generate the output from recent events.





## STEP 12: Implementing url in the function node to generate output

