Real time river Quality monitoring and control system

SPRINT – 2

TITLE	Real time river Quality monitoring and control system
DOMAIN NAME	INTERNET OF THINGS
TEAM ID	PNT2022TMID20521

Building Project

Connecting IoT Simulator to IBM Watson IoT Platform

Give the credentials of your device in IBM Watson IoT PlatformClick on connect

My credentials given to simulator are:

Organization ID: fwe3x0

api: 0cf661005104a3a0

Device type: IOT_device

Token: 1911010abcdefgh

Device ID : 1911010

- ➢ You will receive the simulator data in cloud
- ➢ You can see the received data in Recent Events under your device

➢ Data received in this format(json)

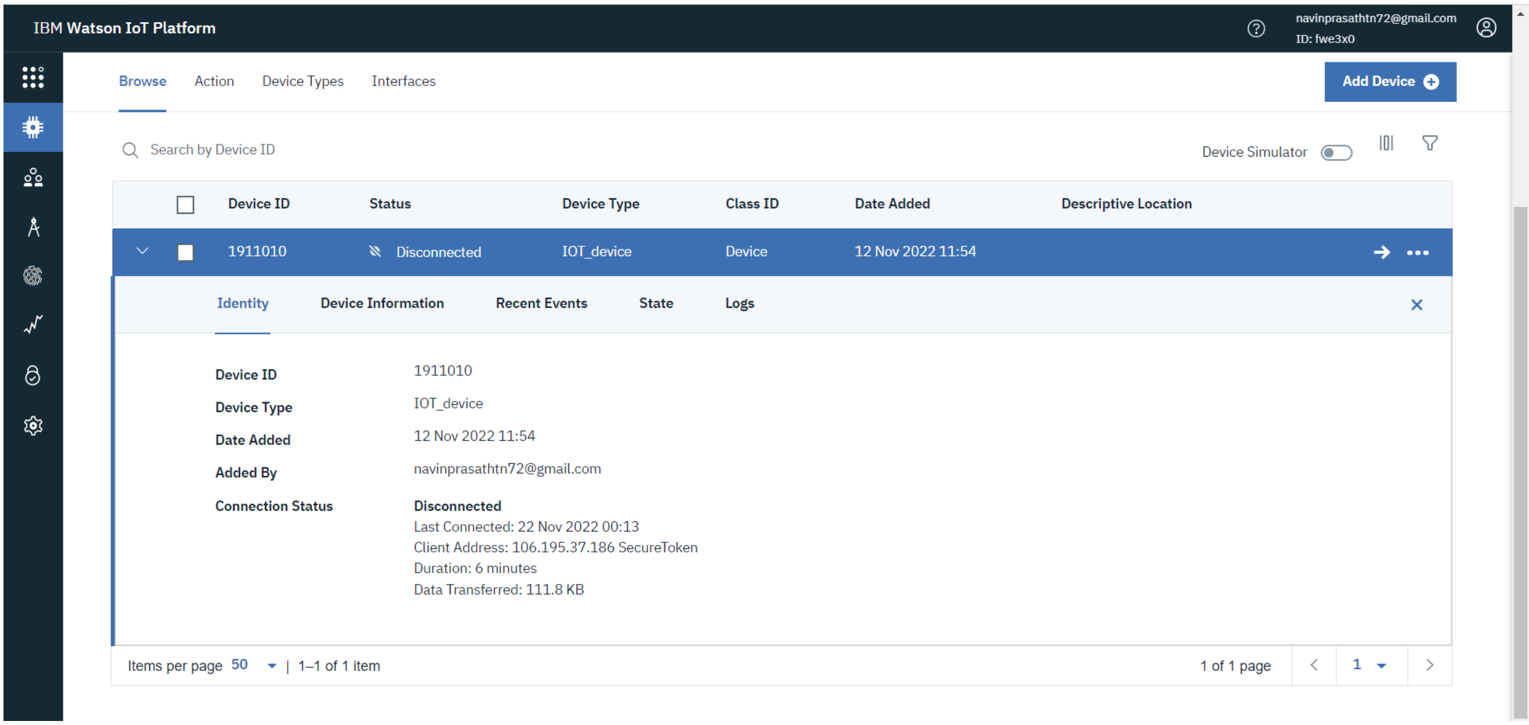
{

"d": {

- "name": "1911010",
- "pH": 8,
- "turbidity": 80

}

}



□	Device ID	Status	Device Type	Class ID	Date Added	Descriptive Location
□	1911010	Disconnected	IOT_device	Device	12 Nov 2022 11:54	

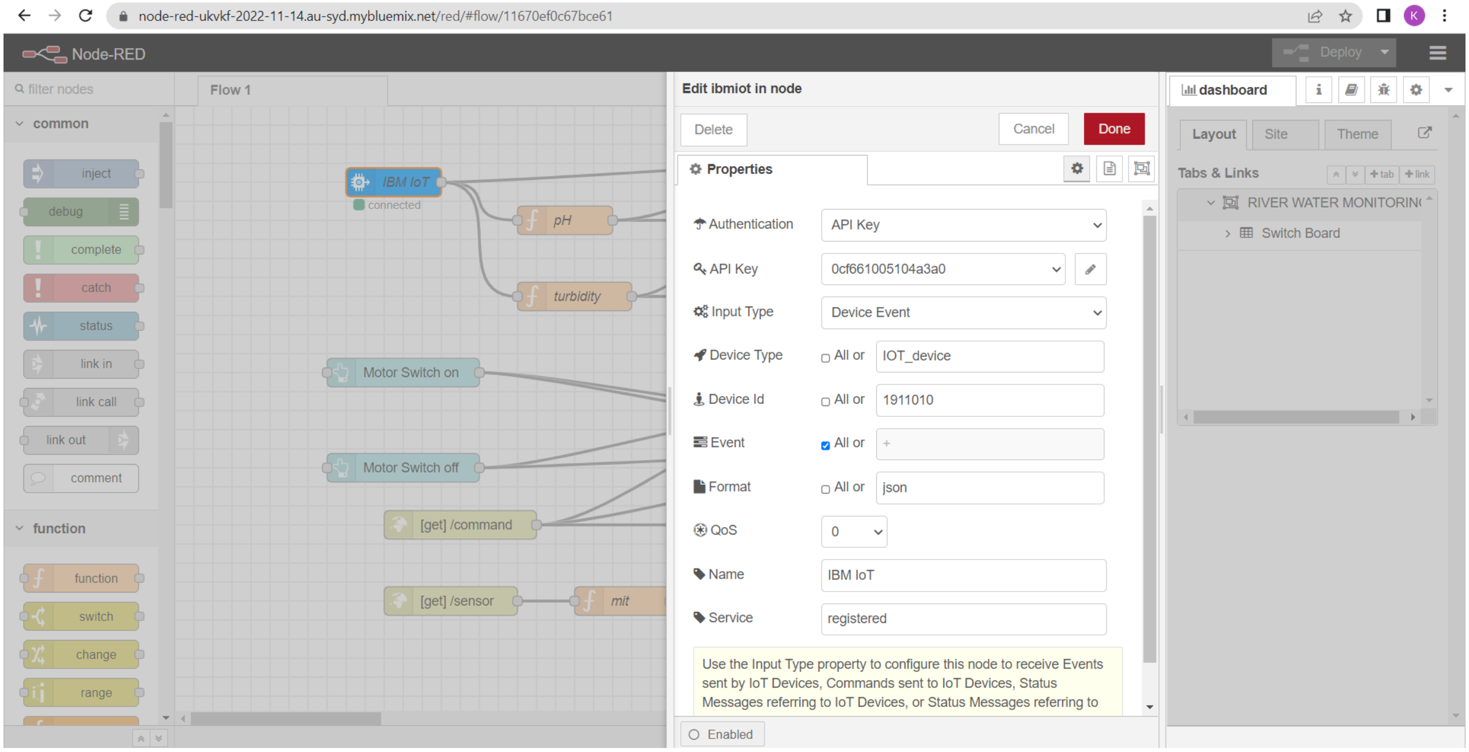
Device ID 1911010
Device Type IOT_device
Date Added 12 Nov 2022 11:54
Added By navinprasathtn72@gmail.com
Connection Status **Disconnected**
Last Connected: 22 Nov 2022 00:13
Client Address: 106.195.37.186 SecureToken
Duration: 6 minutes
Data Transferred: 111.8 KB

Items per page 50 | 1–1 of 1 item

1 of 1 page 1

Configuration of Node-Red to collect IBM cloud data

The node IBM IoT App In is added to the Node-Red workflow. Then the appropriate device credentials obtained earlier are entered into the node to connect and fetch device telemetry to Node-Red.



node-red-ukvkf-2022-11-14.au-syd.mybluemix.net/red/#flow/11670ef0c67bce61

Edit ibmiot in node

Properties

Authentication: API Key
API Key: 0cf661005104a3a0
Input Type: Device Event
Device Type: □ All or IOT_device
Device Id: □ All or 1911010
Event: ☑ All or +
Format: □ All or json
QoS: 0
Name: IBM IoT
Service: registered

Use the Input Type property to configure this node to receive Events sent by IoT Devices, Commands sent to IoT Devices, Status Messages referring to IoT Devices, or Status Messages referring to

Once it is connected Node-Red receives data

from the device Display the data using debug

node for verification

Connect function node and write the Java script code to get each reading separately.

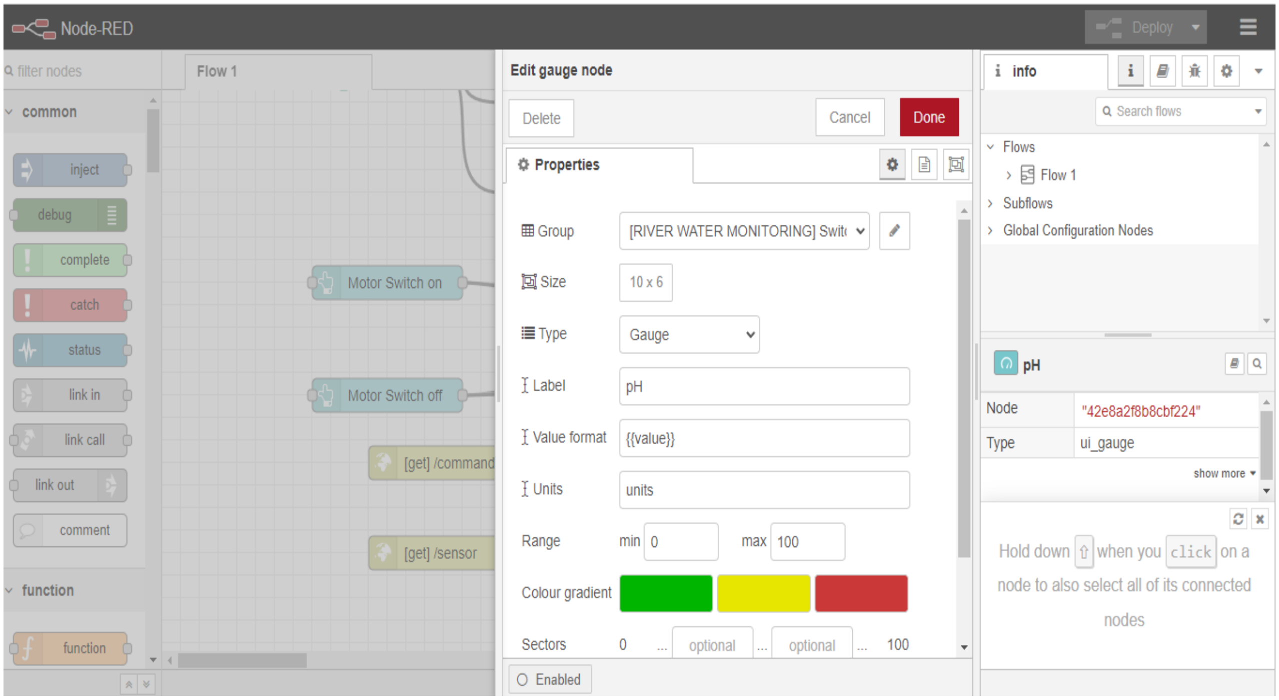
The Java script code for the

function node is:

msg.payload=msg.payload.d.pH

returnmsg;

Finally connect Gauge nodes from dashboard to see the data in UI



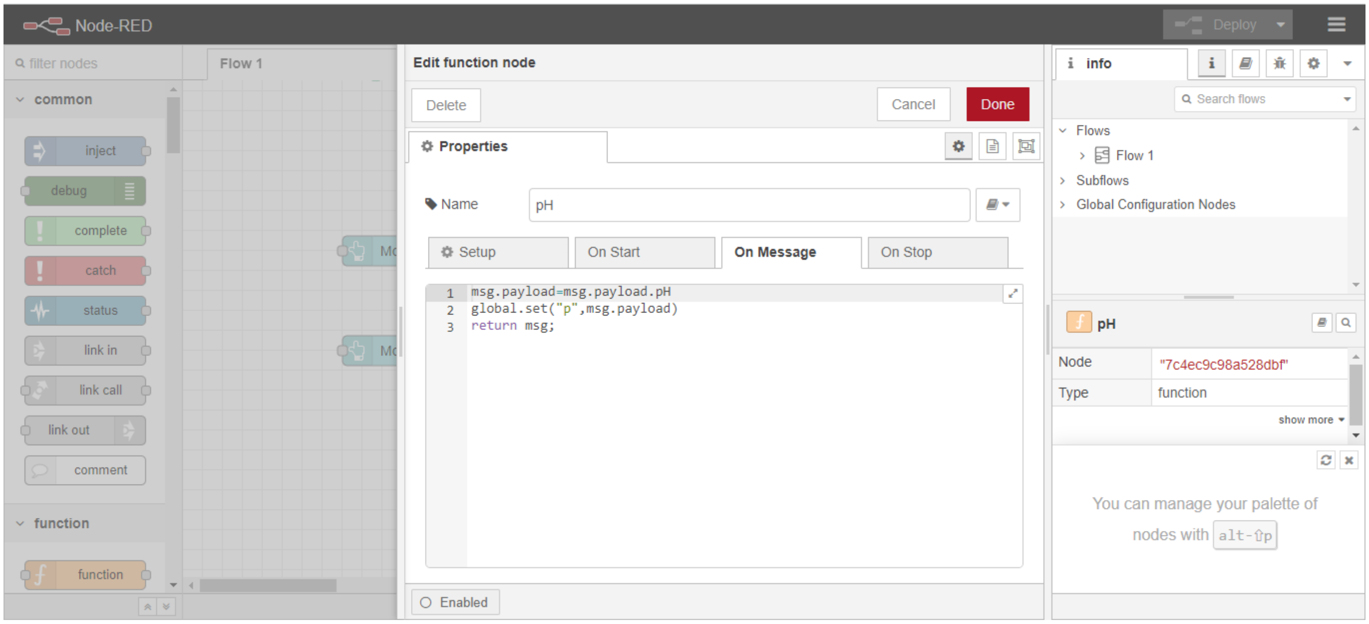
Edit gauge node

Group	[RIVER WATER MONITORING] Swit
Size	10 x 6
Type	Gauge
Label	pH
Value format	{{value}}
Units	units
Range	min 0 max 100
Colour gradient	🟩 🟨 🟥
Sectors	0 ... optional ... optional ... 100

pH

Node	"42e8a2f8b8cbf224"
Type	ui_gauge

Hold down ⇧ when you click on a node to also select all of its connected nodes



Edit function node

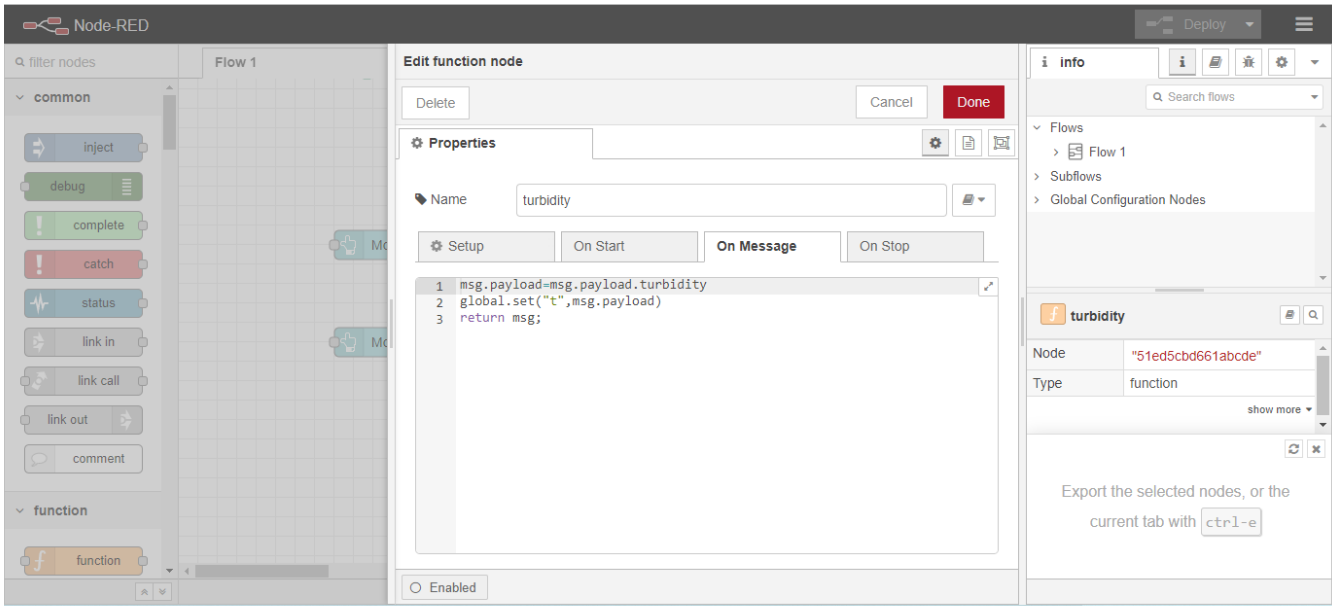
Name pH

```
msg.payload=msg.payload.pH
global.set("p",msg.payload)
return msg;
```

pH

Node	"7c4ec9c98a528dbf"
Type	function

You can manage your palette of nodes with alt-⇧p



Name: turbidity

```
msg.payload=msg.payload.turbidity
global.set("t",msg.payload)
return msg;
```

Node	"51ed5cbd661abcde"
Type	function