

Use Dashboard Nodes For Creating UI(Web App)

TEAM ID	PNT2022TMID48692
PROJECT TITLE	REAL TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM

Step 1: Sending data to the IBM Watson

IBM Watson IoT Platform

Device ID: 12345, Status: Connected, Device Type: Nodered, Class ID: Device, Date Added: Nov 9, 2022 10:02 AM

Recent Events

Event	Value	Format	Last Received
status	{"ph":13,"turbidity":4}	json	a few seconds ago
status	{"ph":7,"turbidity":1}	json	a few seconds ago
status	{"ph":9,"turbidity":2}	json	a few seconds ago
status	{"ph":1,"turbidity":3}	json	a few seconds ago
status	{"ph":13,"turbidity":0}	json	a few seconds ago

Step2: Configure the IBM IOT in the Node-red

Flow 1

IBM IoT (connected)

Cloudant

turbidity

ph

[get] /sensor_data

http req

Edit ibmiot in node > Edit ibmiot node

Delete Cancel Update

Properties

Name: lot api

API Key: a-ij64y3-w7zkatzbl9

API Token:

Server-Name: ij64y3.messaging.internetofthings.ibmcloud.com

Scalable: ☐ Application ID:

Keep Alive: 60 Seconds ☒ Use Clean Session

Step 3: Configure the Device in IOT in

The screenshot shows the Node-RED interface with a flow named "Flow 1". The flow consists of an "IBM IoT" node (with a "connected" status) connected to three function nodes: "ph", "Cloudant", and "turbidity". Below this, there is a "[get] /sensor_data" node connected to an "http req" node. The "Edit ibmiot in node" dialog is open, showing the following properties:

- Authentication: API Key
- API Key: lot api
- Input Type: Device Event
- Device Type: All or +
- Device Id: All or device id e.g. ab12cd231a21
- Event: All or +
- Format: All or json
- QoS: 0
- Name: IBM IoT
- Service: registered

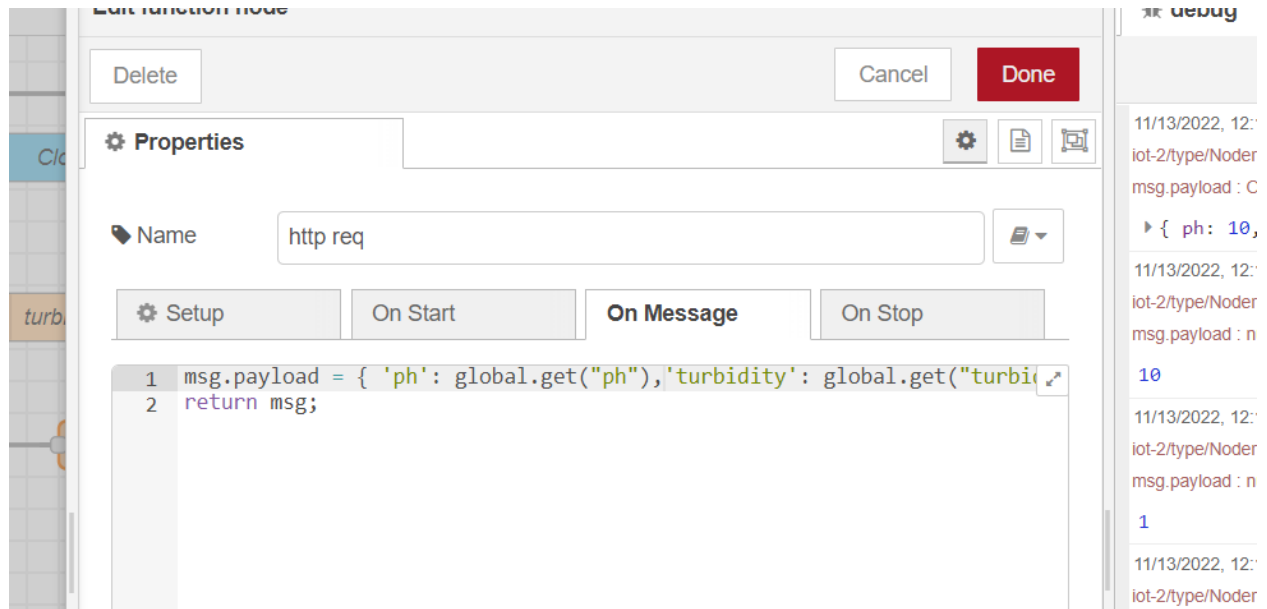
A note at the bottom of the dialog states: "Use the Input Type property to configure this node to receive Events sent by IoT Devices, Commands sent to IoT Devices, Status Messages referring to IoT Devices, or Status Messages referring to IoT Applications."

Step 4: Add a Http in and Http response node, Configure the Http in node and set the Method as GET and create an HTTP API for communicating with the mobile application

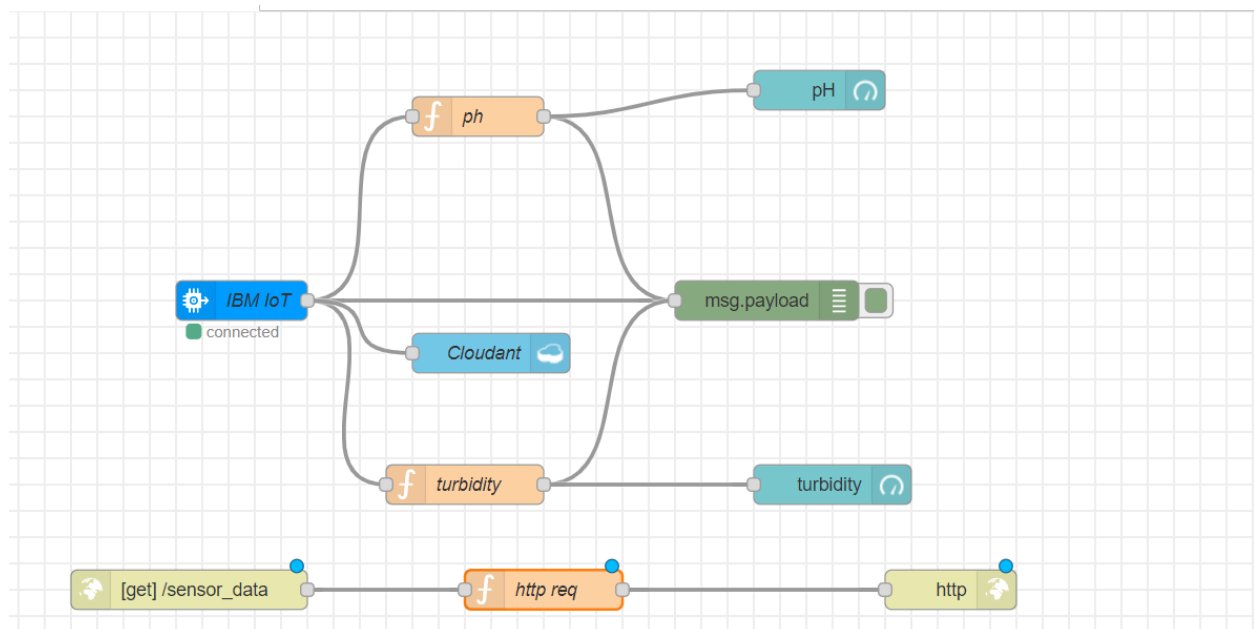
The screenshot shows the Node-RED interface with the same flow as in Step 3. The "Edit http in node" dialog is open, showing the following properties:

- Method: GET
- URL: /sensor_data
- Name: Name

Step 5: Add a function node to combine the pH and Turbidity value



Step 6: Configure all nodes and Deploy the Node-red



Step 7: Go to the API URL in node red and now you can able to view the data from the IBM Watson

