

Develop The Web Application Using Node-RED

TEAM ID	PNT2022TMID41673
PROJECT TITLE	REAL -TIME RIVER WATER QUALITY MONITORING AND CONTROL SYSTEM

Step 1: Sending data to the IBM Watson

The screenshot displays the IBM Watson IoT Platform interface. The top navigation bar includes 'Browse', 'Action', 'Device Types', and 'Interfaces'. A sidebar on the left contains various icons for navigation. The main content area shows a table of device information for device ID 123435, which is in a 'Disconnected' state. Below the table, the 'Recent Events' tab is selected, showing a list of events with columns for Event, Value, Format, and Last Received. The events are JSON objects containing Turbidity and pH values. A status bar at the bottom indicates '1 Simulation running'.

Event	Value	Format	Last Received
data	{\"Turbidity\":203,\"pH_value\":8}	json	a few seconds ago
data	{\"Turbidity\":597,\"pH_value\":12}	json	a few seconds ago
data	{\"Turbidity\":110,\"pH_value\":10}	json	a few seconds ago
data	{\"Turbidity\":608,\"pH_value\":9}	json	a few seconds ago
data	{\"Turbidity\":393,\"pH_value\":10}	json	a few seconds ago

Step 2: Configure the IBM IOT in the Node-red

The screenshot shows the Node-RED interface with a flow diagram on the left and the configuration panel for the 'ibmiot' node on the right. The flow diagram includes an 'IBM IoT' node connected to 'ph' and 'turbidity' nodes, which are then connected to a 'Cloudant' node. The configuration panel for the 'ibmiot' node is open, showing fields for Name, API Key, API Token, Server-Name, Scalable, Application ID, Keep Alive, and Use Clean Session.

Properties

- Name: lot api
- API Key: a-jj64y3-w7zkatzbl9
- API Token:
- Server-Name: jj64y3.messaging.internetofthings.ibmcloud.com
- Scalable: ☐
- Application ID:
- Keep Alive: 60 Seconds
- Use Clean Session: ☒

Step 3: Configure the Device in IOT in Node-red

The screenshot shows the Node-RED interface with a flow named 'Flow 1'. The flow includes an 'IBM IoT' node (connected), a 'Cloudant' node, and two function nodes labeled 'ph' and 'turbidity'. Below these, there is a '[get] /sensor_data' node and an 'http req' node. The 'Edit ibmiot in node' configuration panel is open on the right, 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

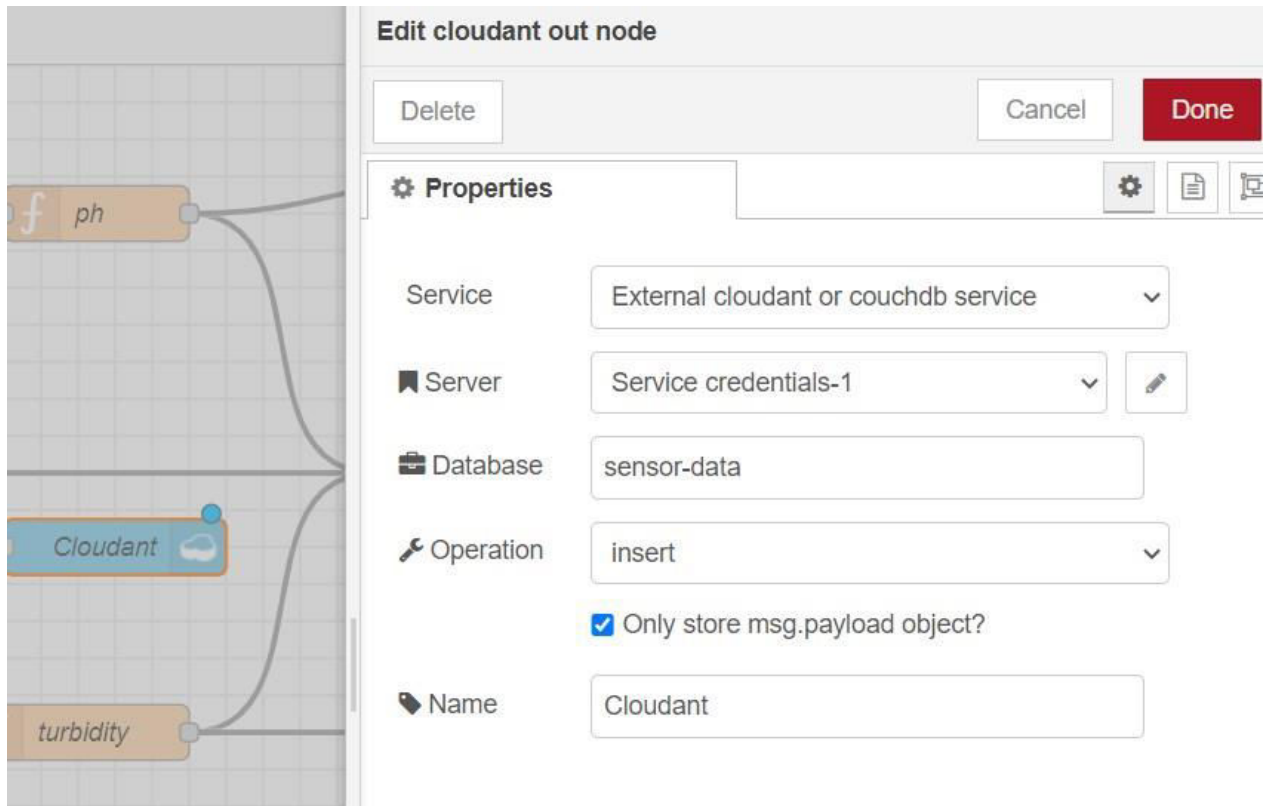
A note at the bottom of the panel states: "Use the Input Type property to configure this node to receive Events sent by IoT Devices, Commands sent to IoT Devices, Status Messages referring to IoT Devices, or Status Messages referring to IoT Applications."

Step 4: Configure the Host and Username in the IBM Cloudant

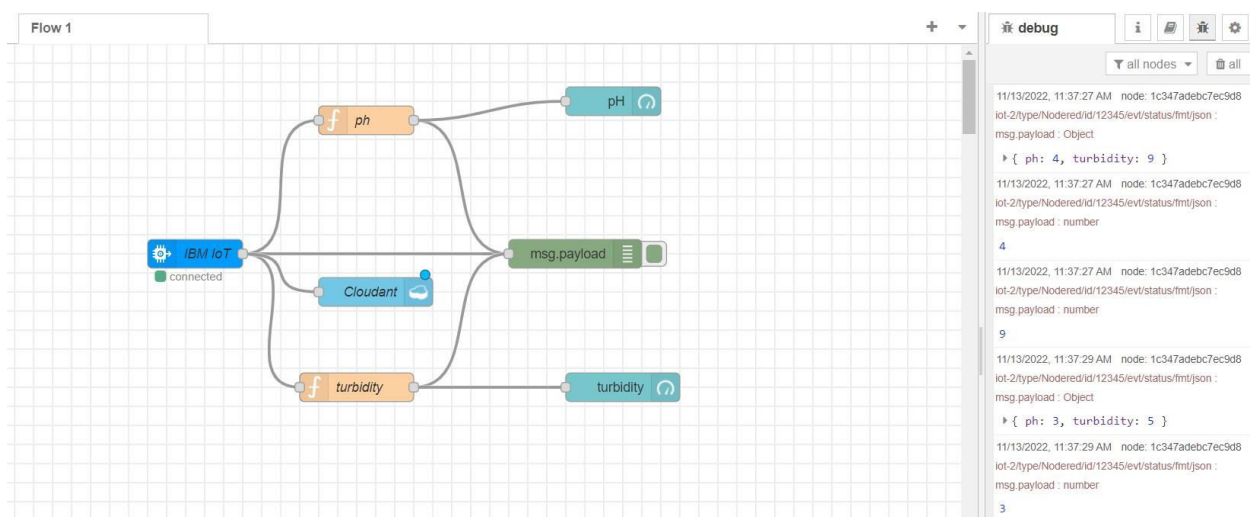
The screenshot shows the Node-RED interface with the same flow as in Step 3. The 'Edit cloudant in node' configuration panel is open on the right, showing the following properties:

- Host: https://apikey-v2-1qahn1go9eja4qf24hvp05tous
- Username: apikey-v2-1qahn1go9eja4qf24hvp05tousbized0:
- Password:
- Name: Service credentials-1

Step 5 : Edit the Cloudant out node and set the database where the data is need to be stored
















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



Step 7: Finally we can able to the store the sensor data and we can view in the data in the Cloudant Dashboard

sensor-data	
documents	
y	
issions	
ges	
n Documents	

Document ID			
Options			
<div> <input type="checkbox"/> <div> <div>Table</div> <div>Metadata</div> <div>{ } JSON</div> <div></div> </div> </div>			
_id	ph	turbidity	
<input type="checkbox"/>  010fe061191ab4d557357902c09708b6	4	5	
<input type="checkbox"/>  010fe061191ab4d557357902c0993384	3	7	
<input type="checkbox"/>  010fe061191ab4d557357902c09be7c4	2	1	
<input type="checkbox"/>  010fe061191ab4d557357902c09c5979	8	6	
<input type="checkbox"/>  010fe061191ab4d557357902c0a7e608	7	9	
<input type="checkbox"/>  010fe061191ab4d557357902c0b235e0	0	0	
<input type="checkbox"/>  010fe061191ab4d557357902c0bca96b	12	1	
<input type="checkbox"/>  010fe061191ab4d557357902c0bfd11b	6	4	
<input type="checkbox"/>  010fe061191ab4d557357902c0c453fe	3	2	
<input type="checkbox"/>  010fe061191ab4d557357902c0c7a3d9	6	7	
<input type="checkbox"/>  010fe061191ab4d557357902c0c841c3	9	0	
<input type="checkbox"/>  010fe061191ab4d557357902c0c85fde	0	10	
<input type="checkbox"/>  010fe061191ab4d557357902c0ce91bd	12	3	

sensor-data ➤ 010fe061191ab4d557357902c09708b6

✓ Save Changes

Cancel

1

2

"_id": "010fe061191ab4d557357902c09708b6",

3

"_rev": "1-7ec49537df45a74402eaf02e1ef6b84d",

4

"ph": 4,

5

"turbidity": 5

6