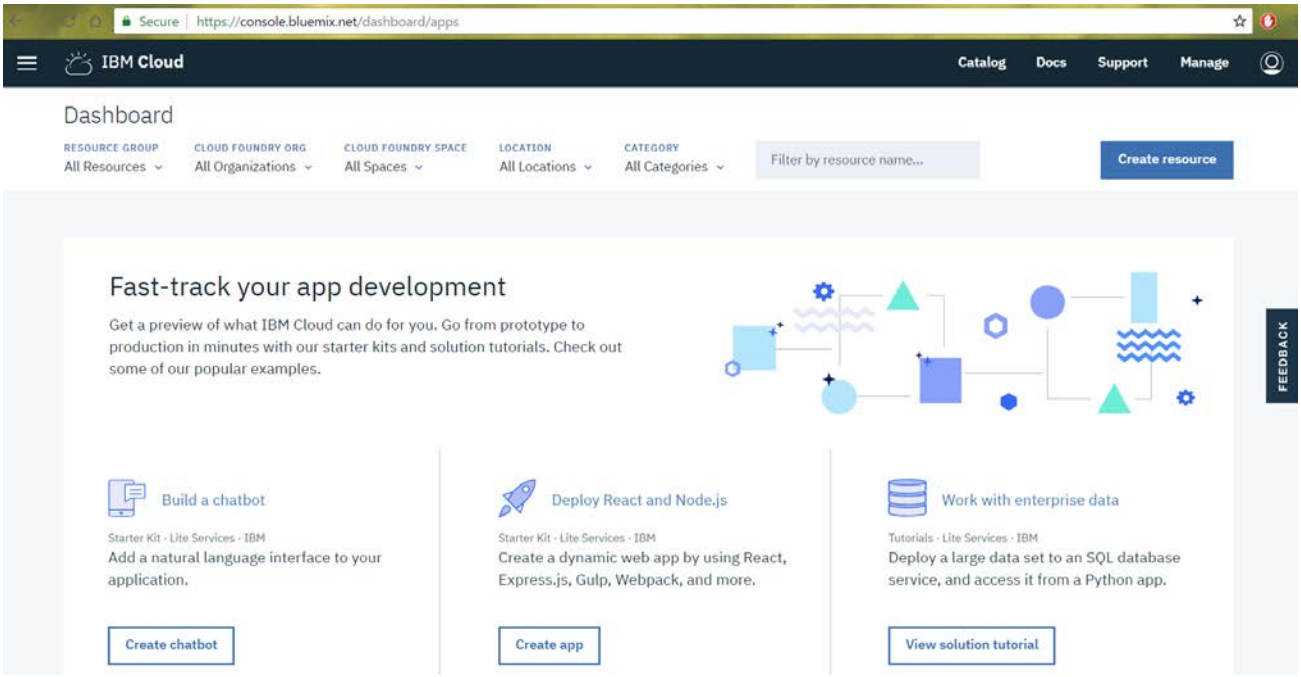
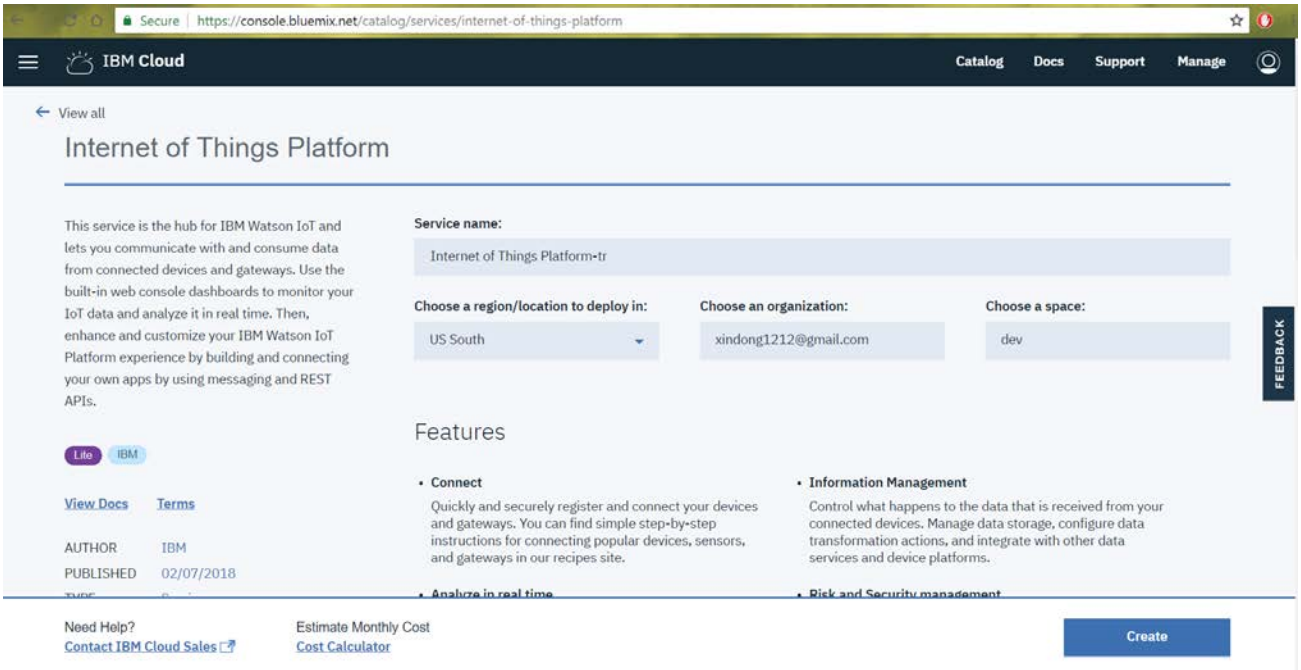
Go to https://console.bluemix.net/ to “Create a free account”.

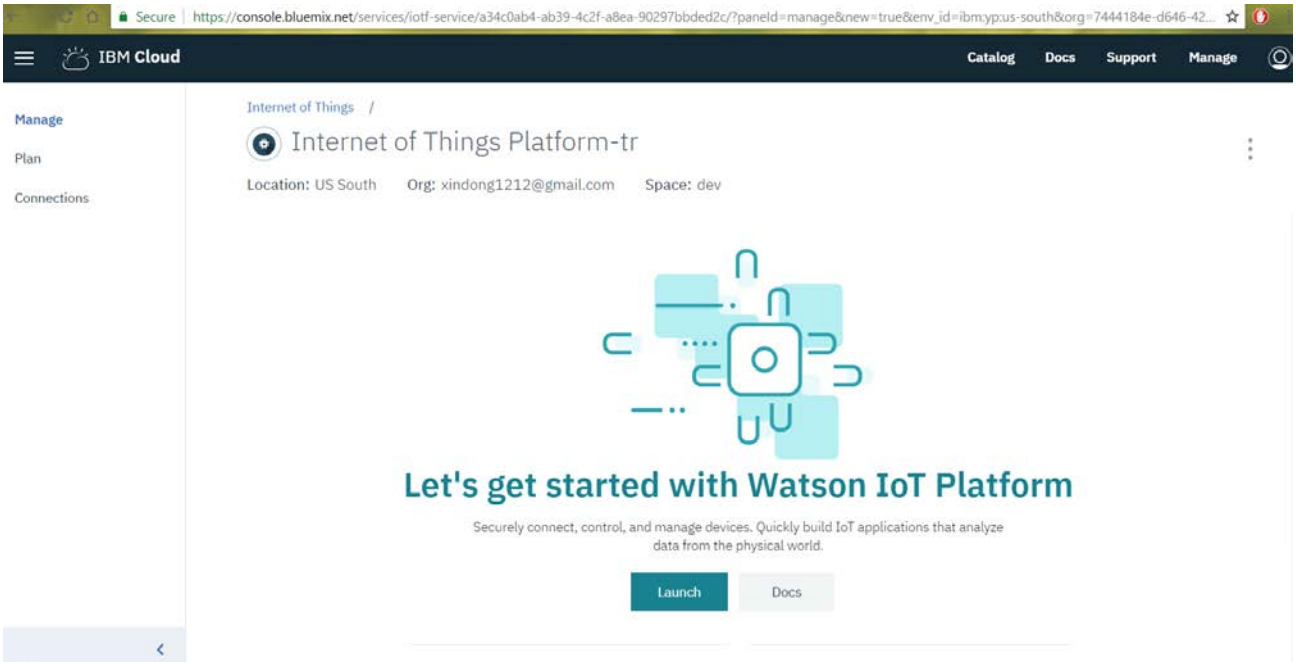


“Catalog” -> “Platform” -> “Internet of Things”

Create an “Internet of Things Platform”.

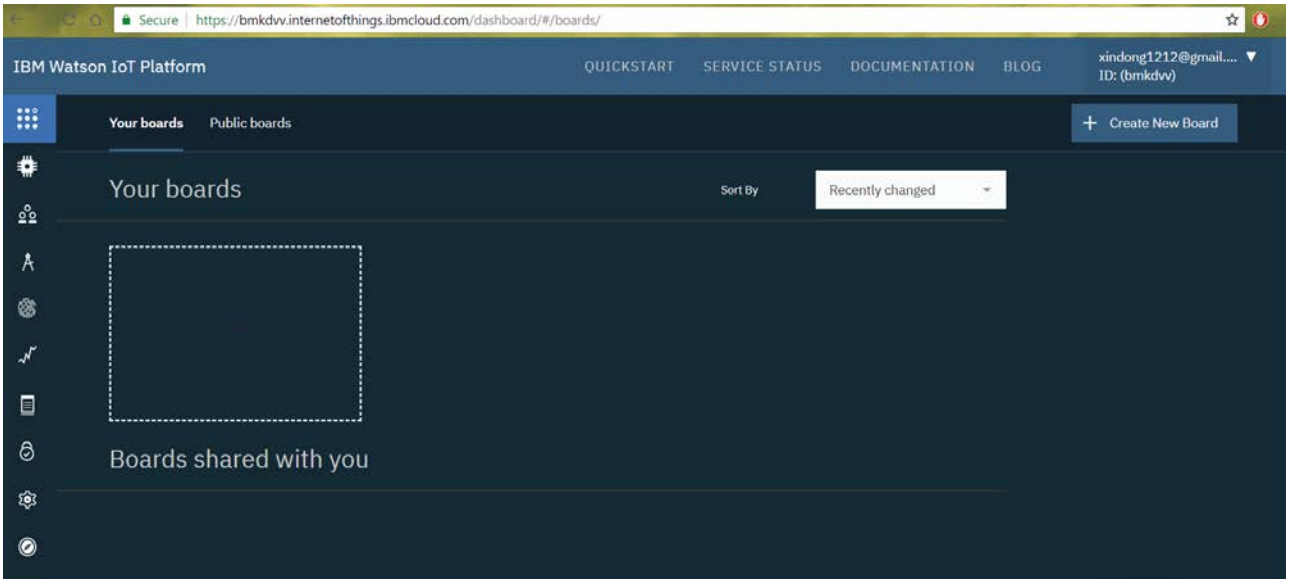


Leave everything as default and “Create”.

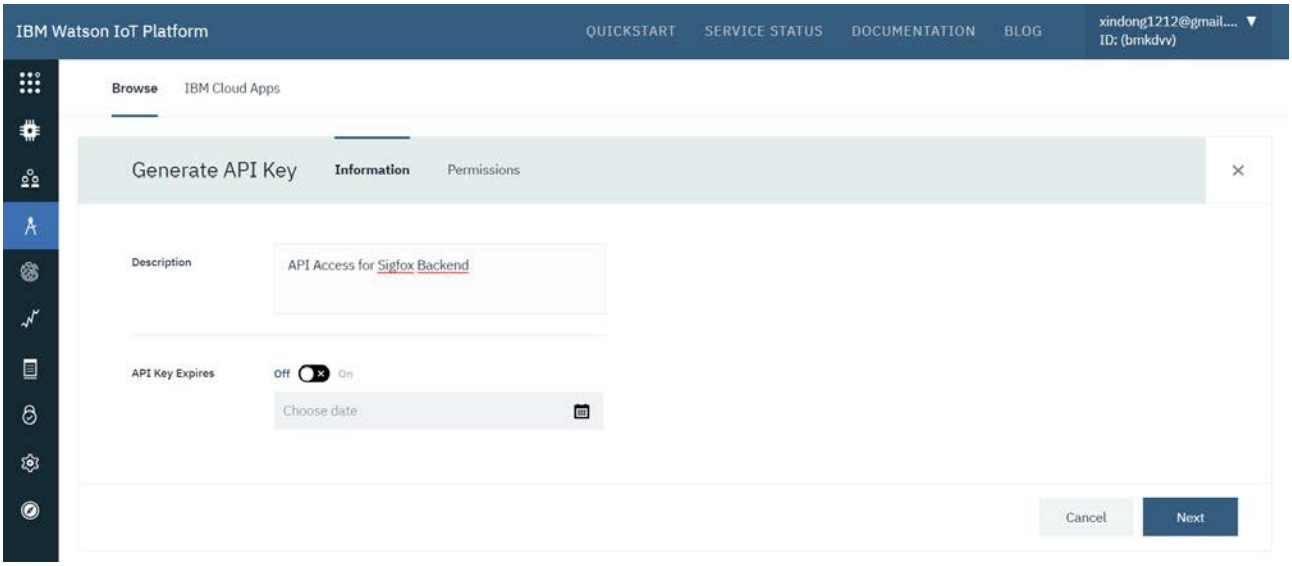


“Launch”.

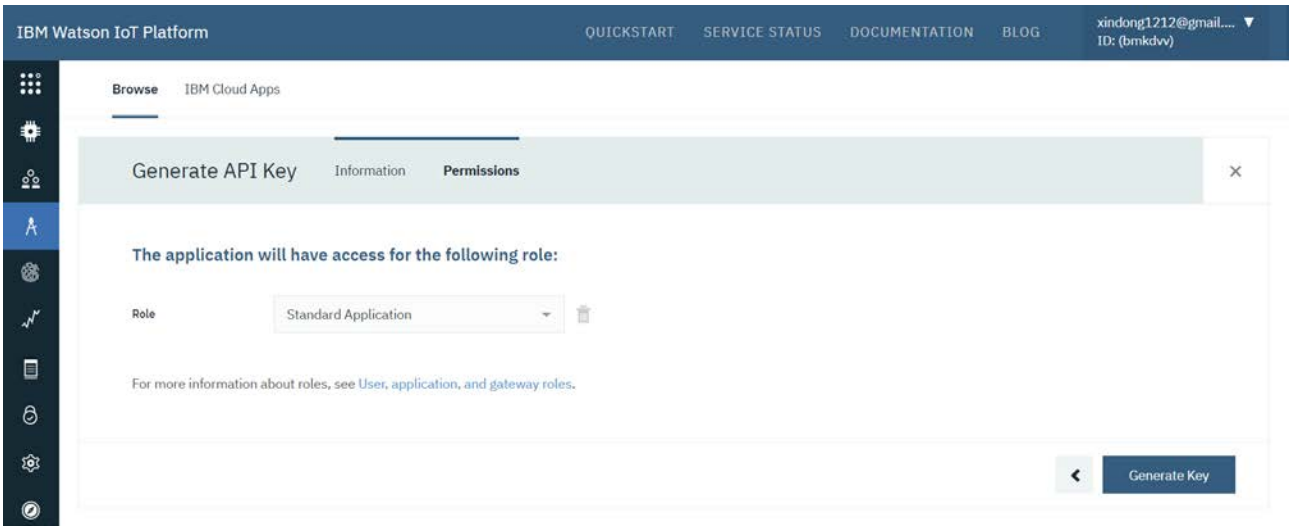
“IBM Watson IoT Platform” opens in a new tab.



“Apps” -> “Generate API Key”

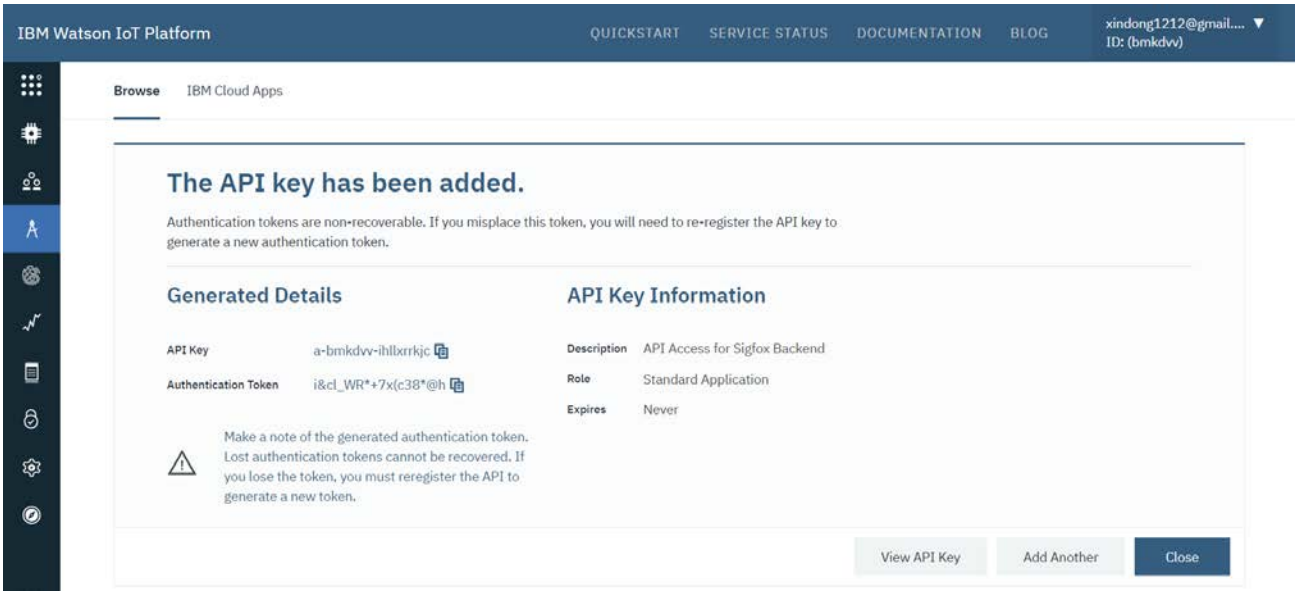


“Next”.



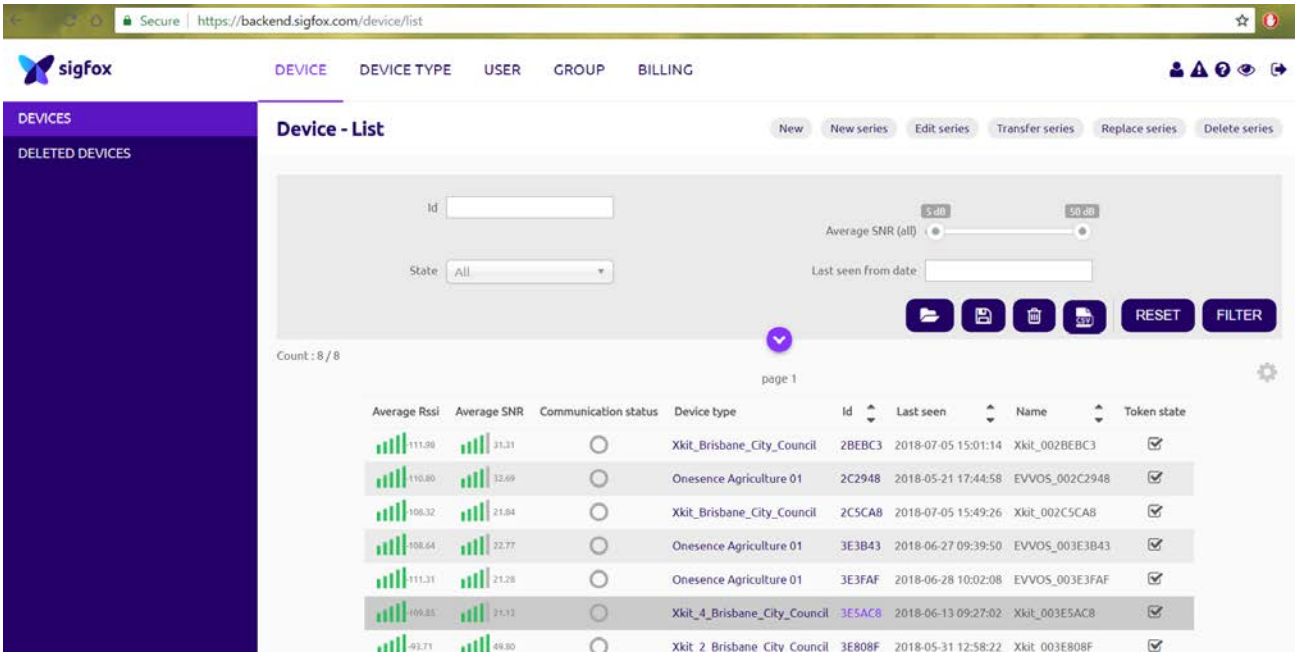
“Generate Key”.

IMPORTANT: Take a screenshot of the “Authentication Token”.

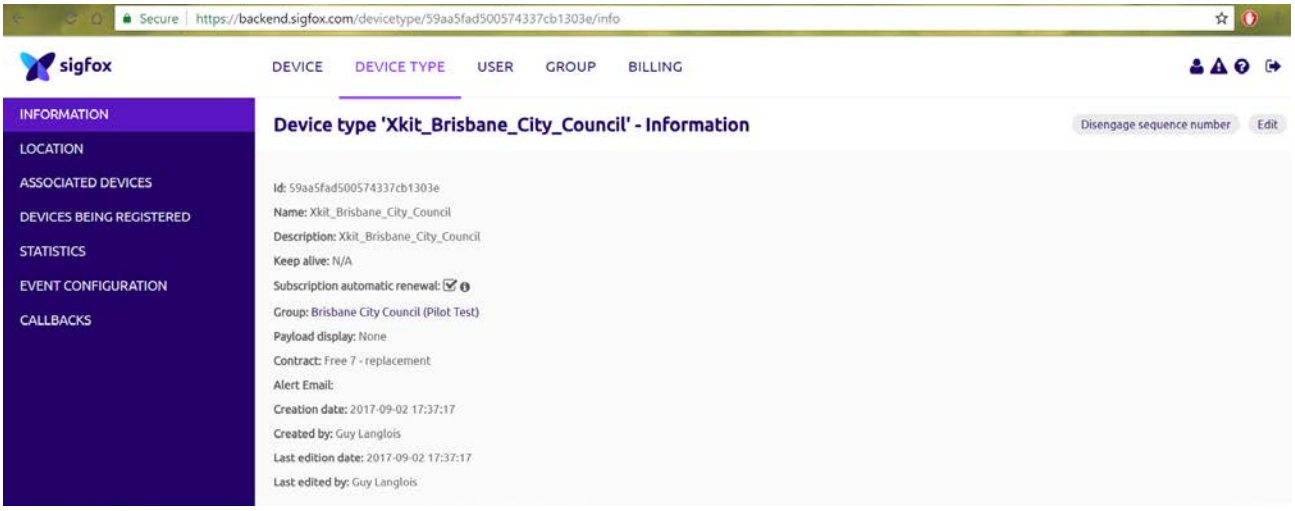


Go to https://backend.sigfox.com.

“Device”.

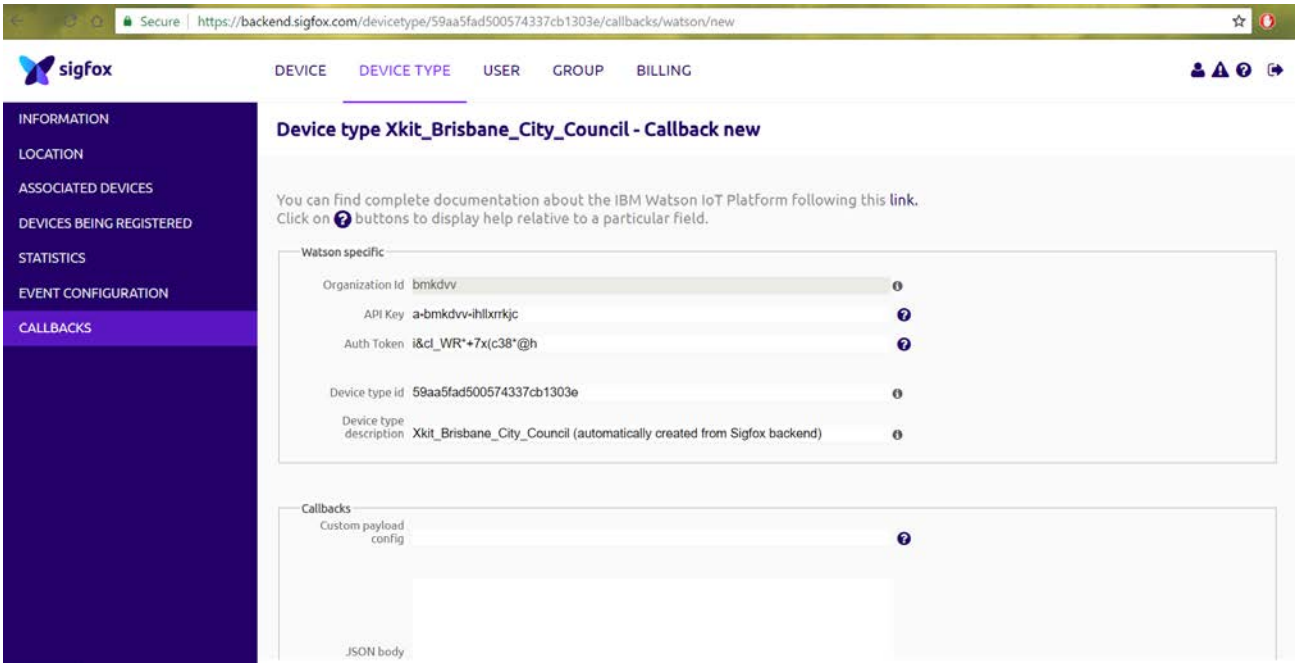


Click “Xkit_Brisbane_City_Council” Device type for Id “2BEBC3”.

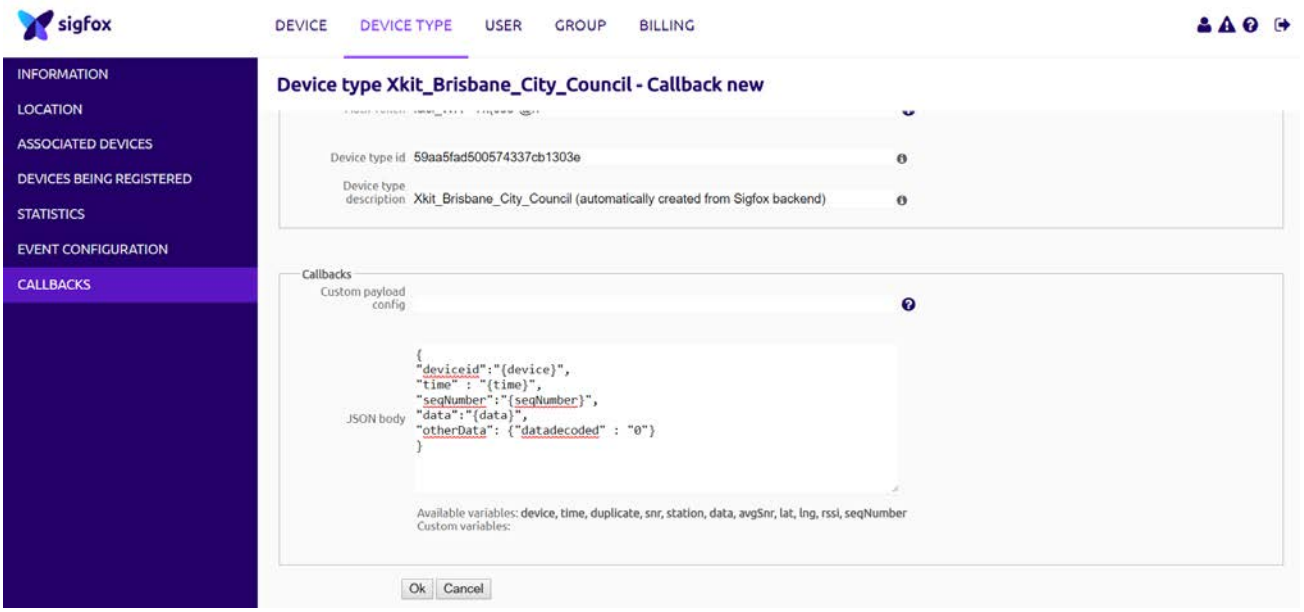


“CALLBACKS” -> “New” -> “IBM Watson IoT Platform”

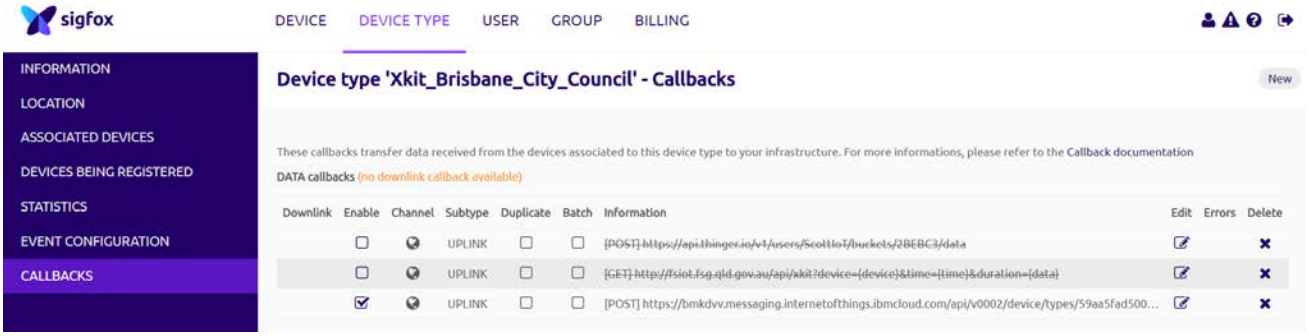
Enter “API Key” and “Auth Token” for the API Key “a-bmkdvv-ihllxrrkjc” created earlier in the Watson IBM Watson IoT Platform.



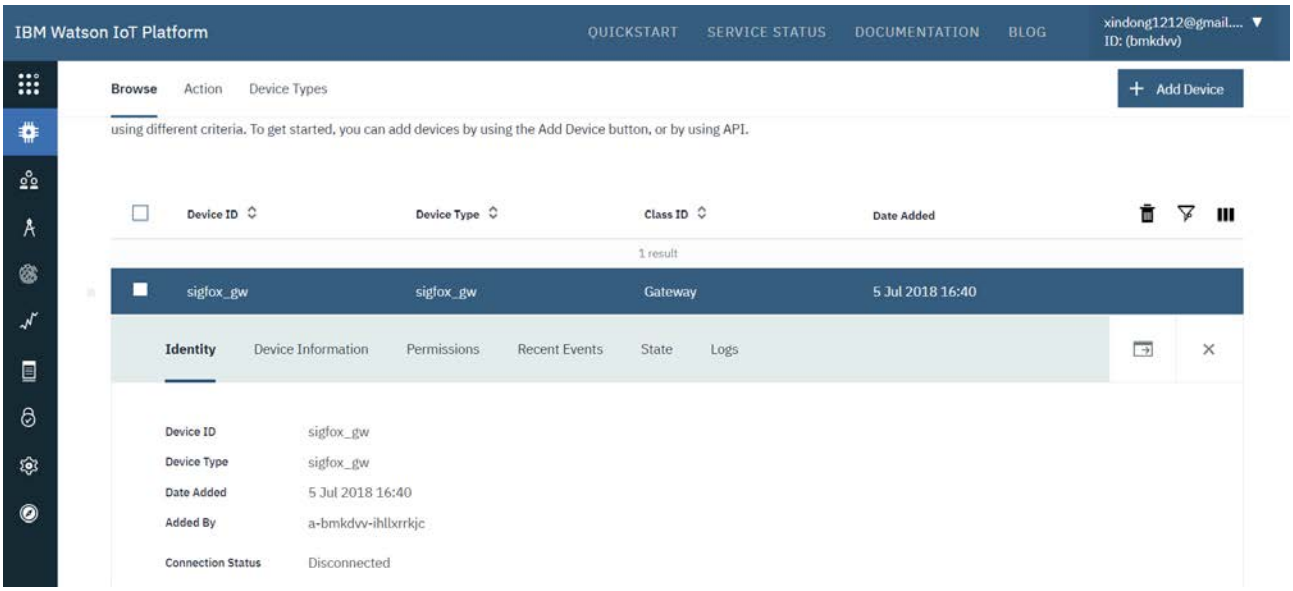
Define “JSON body”.



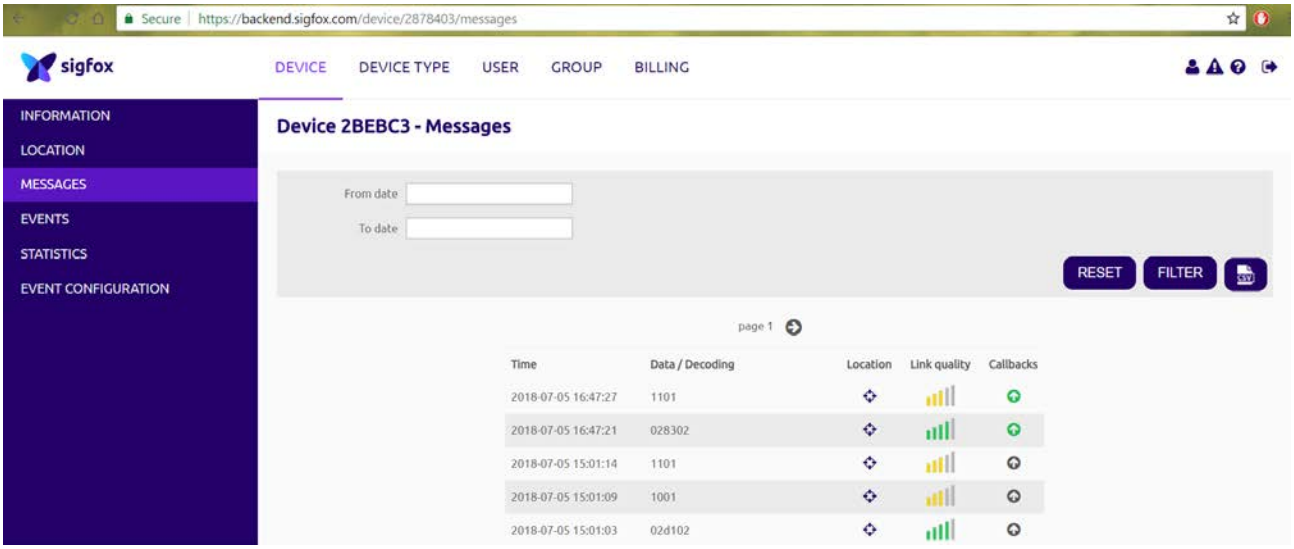
“Ok”.

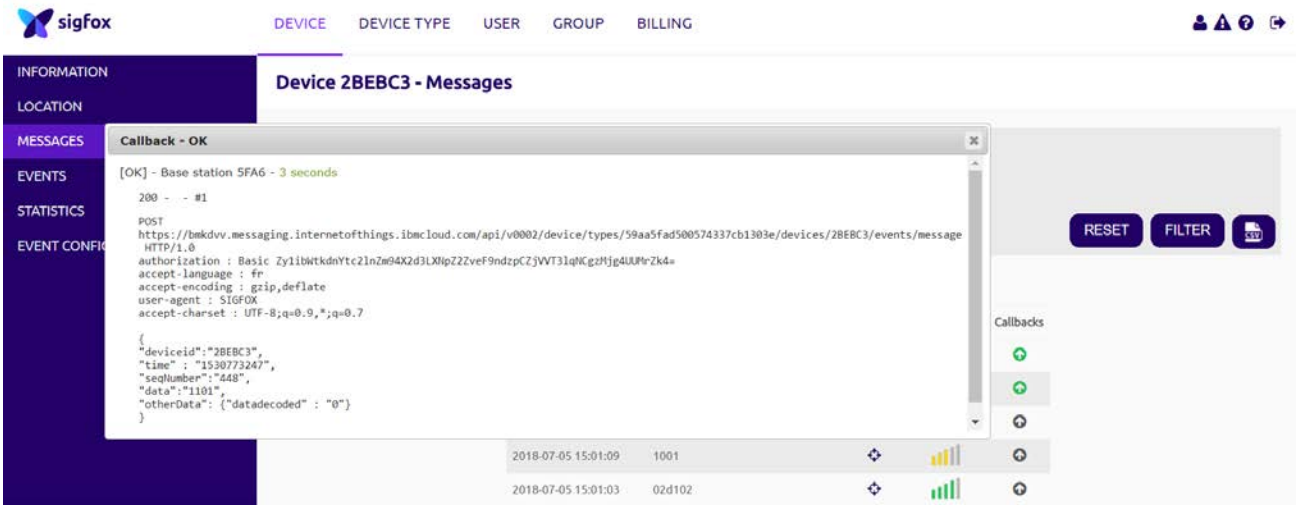


Go back to IBM Watson IoT Platform, check “Devices”, a “Gateway” has been automatically created from Sigfox backend.

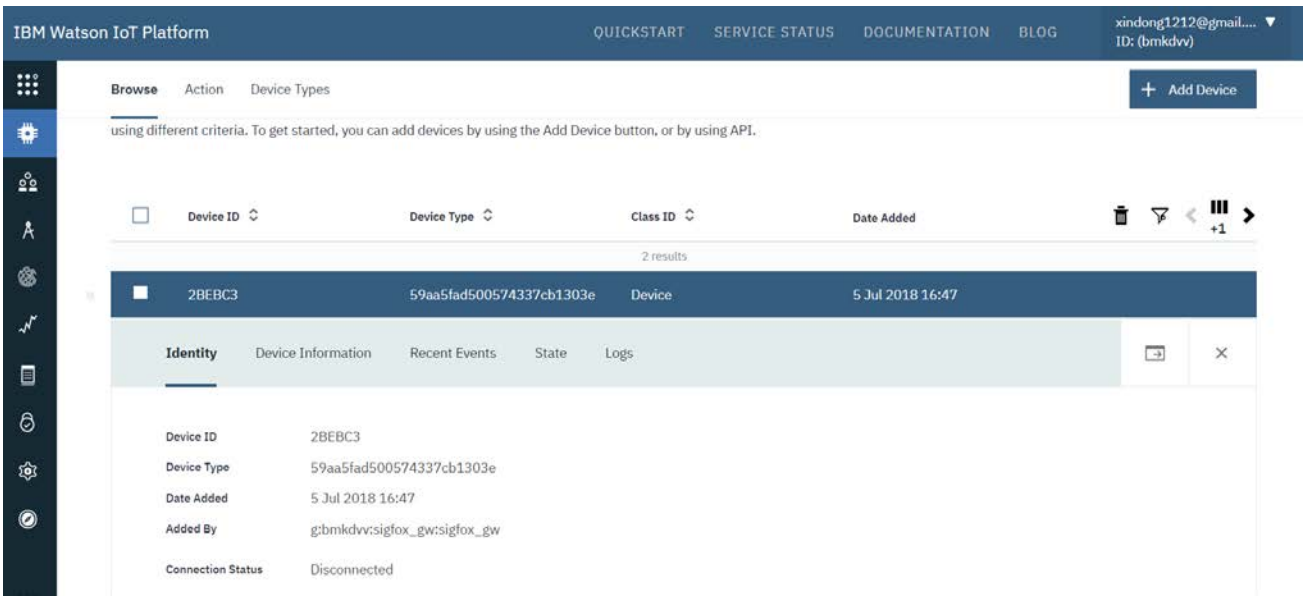


Send a Sigfox message using device “2BEBC3”, “Callback - OK”.



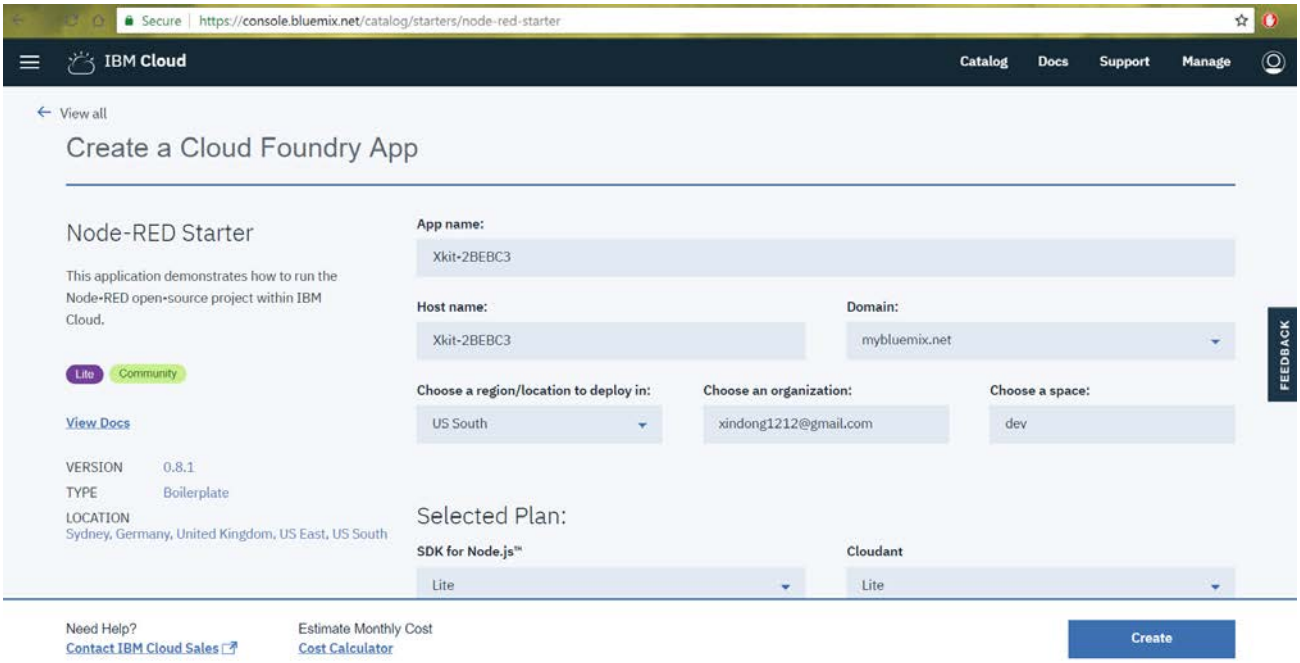


Check that Device “2BEBC3” has been added by the “sigfox_gw” Gateway.



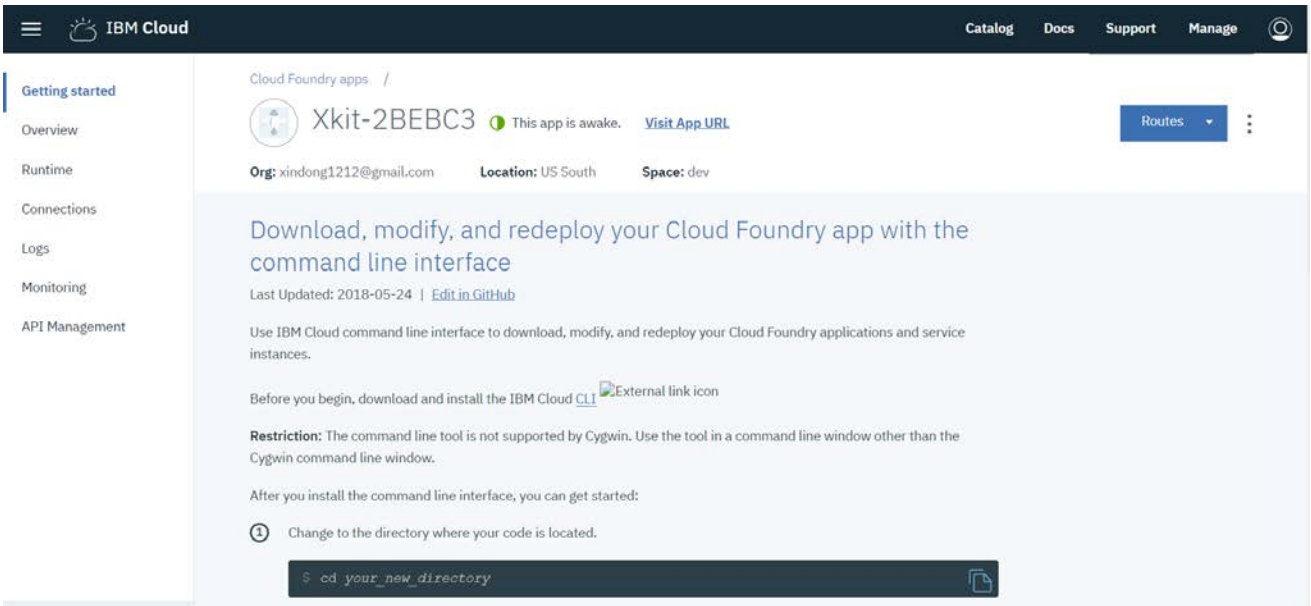
Go back to “IBM Cloud” “Dashboard”, go to “Catalog” and search for “Node-RED”.

Create a “Node-RED Starter” app, enter app name as “Xkit-2BEBC3” and leave all other fields as default.

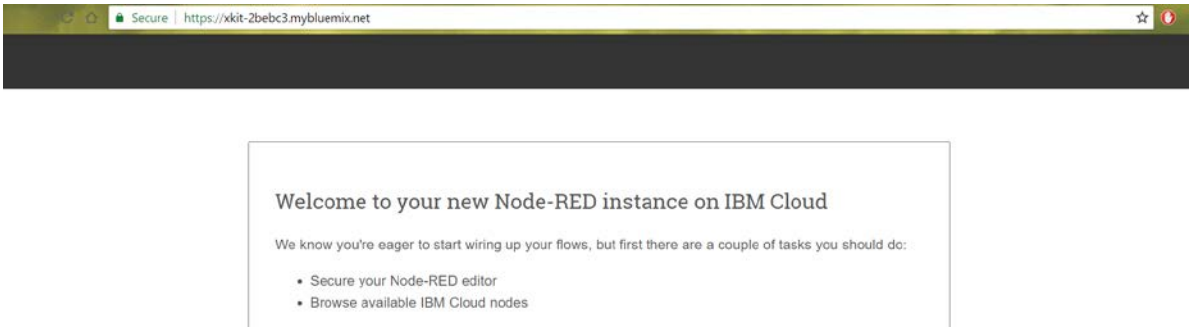


“Create”.

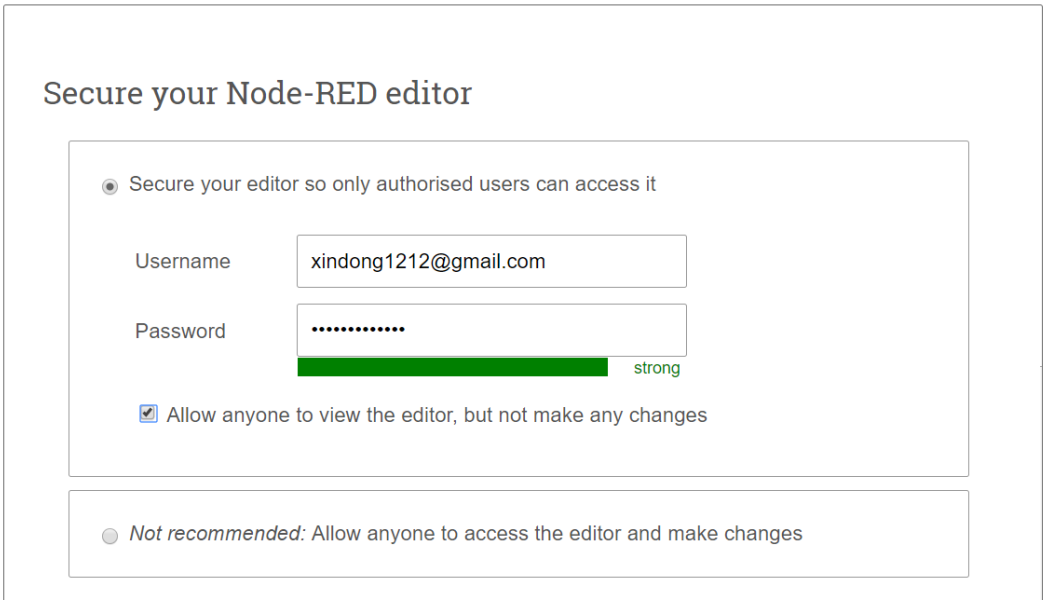
Wait for the app to start.

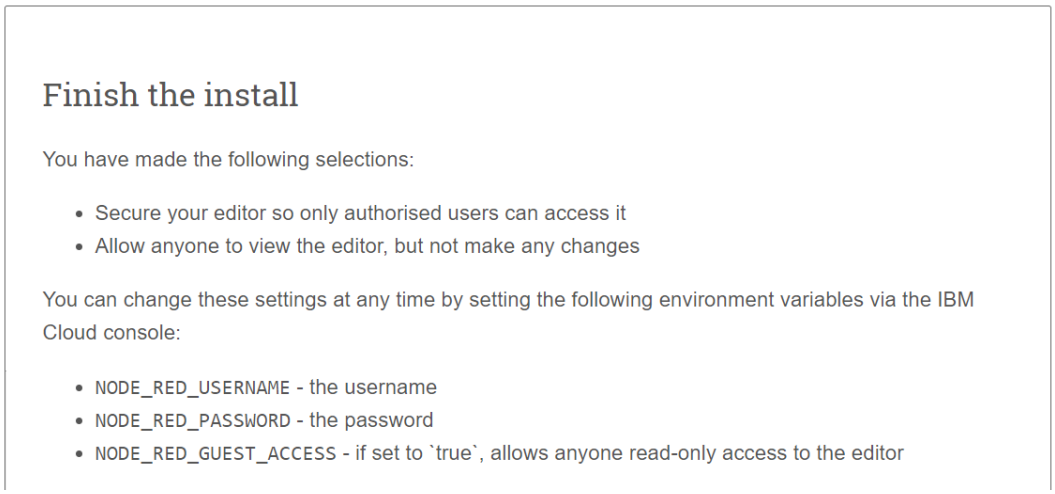


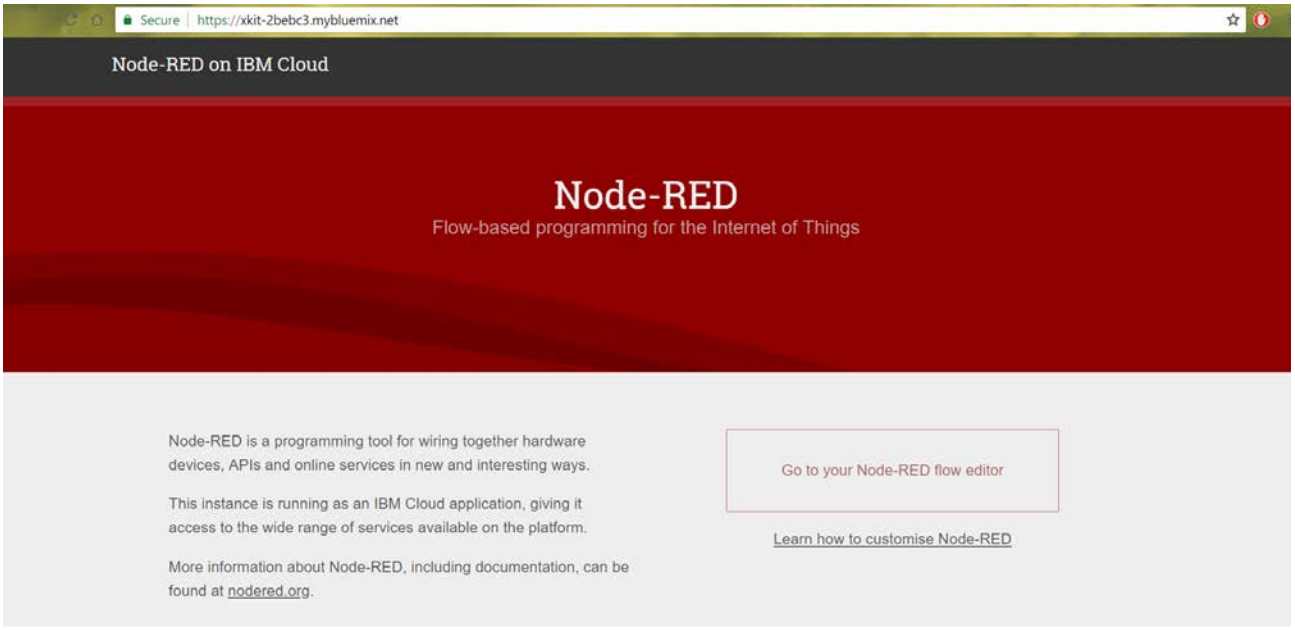
“Visit App URL”.



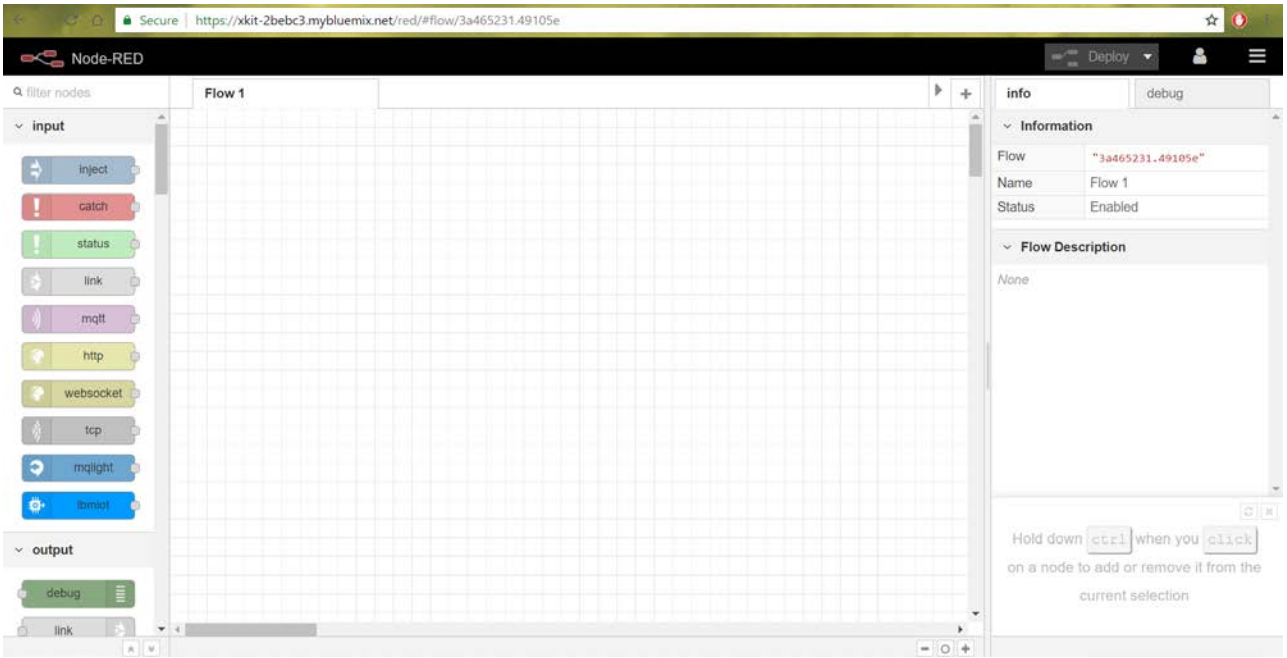
“Allow anyone to view the editor, but not make any changes.”



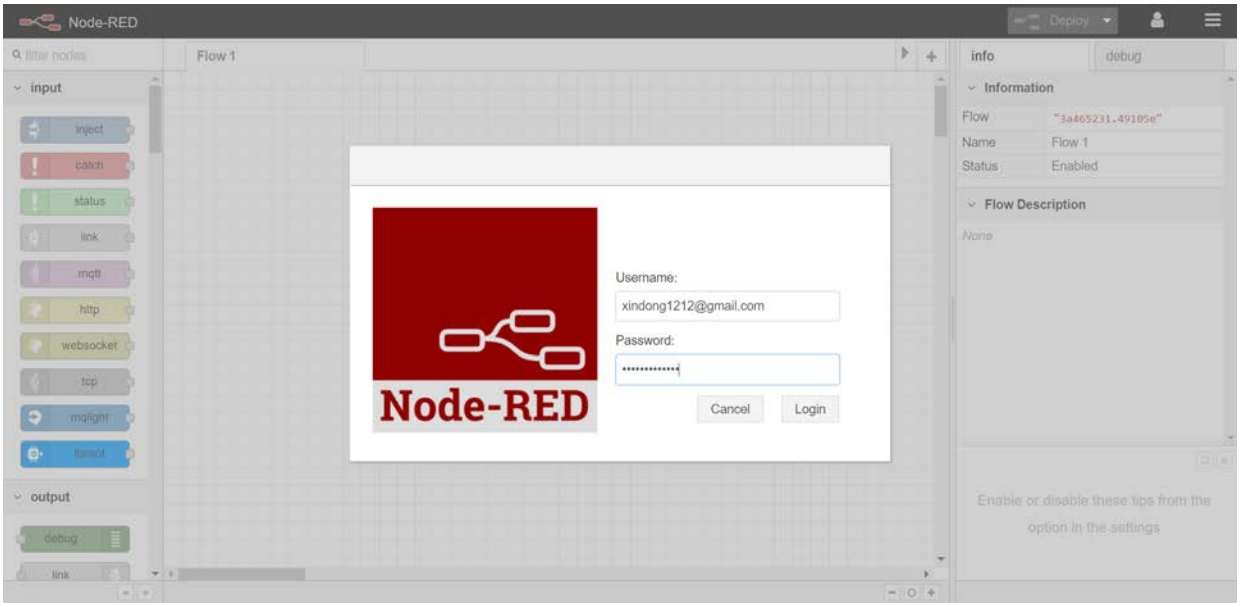




"Go to your Node-RED flow editor".



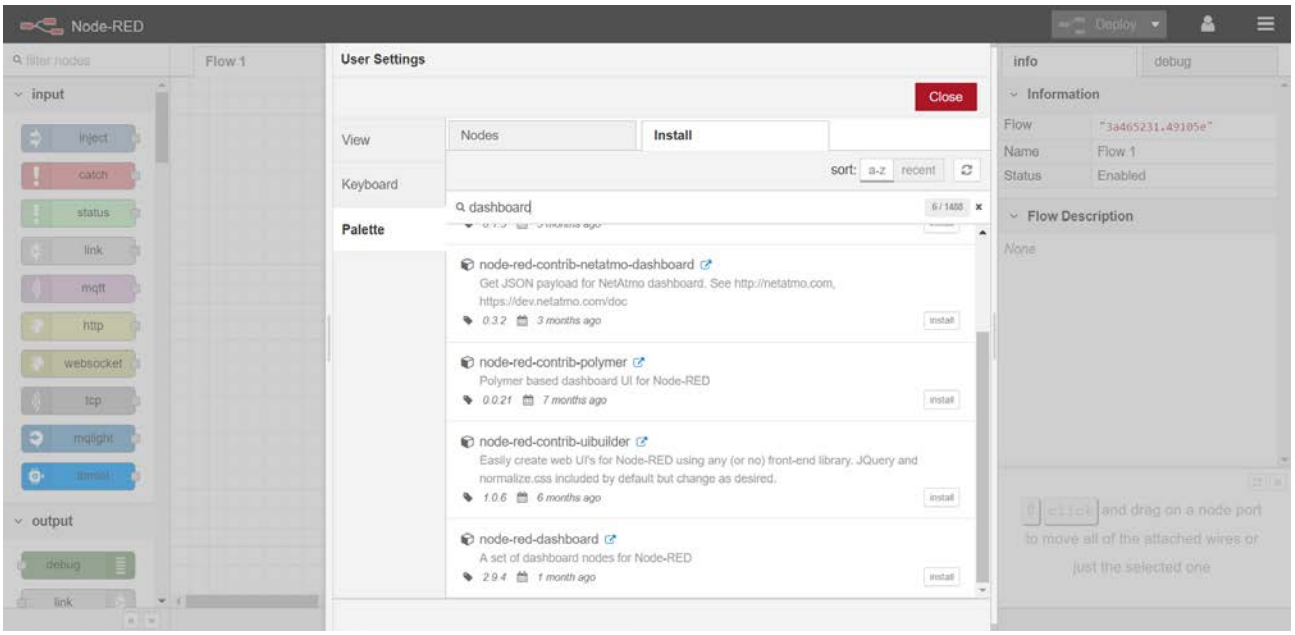
“Login”.



“Manage palette”.

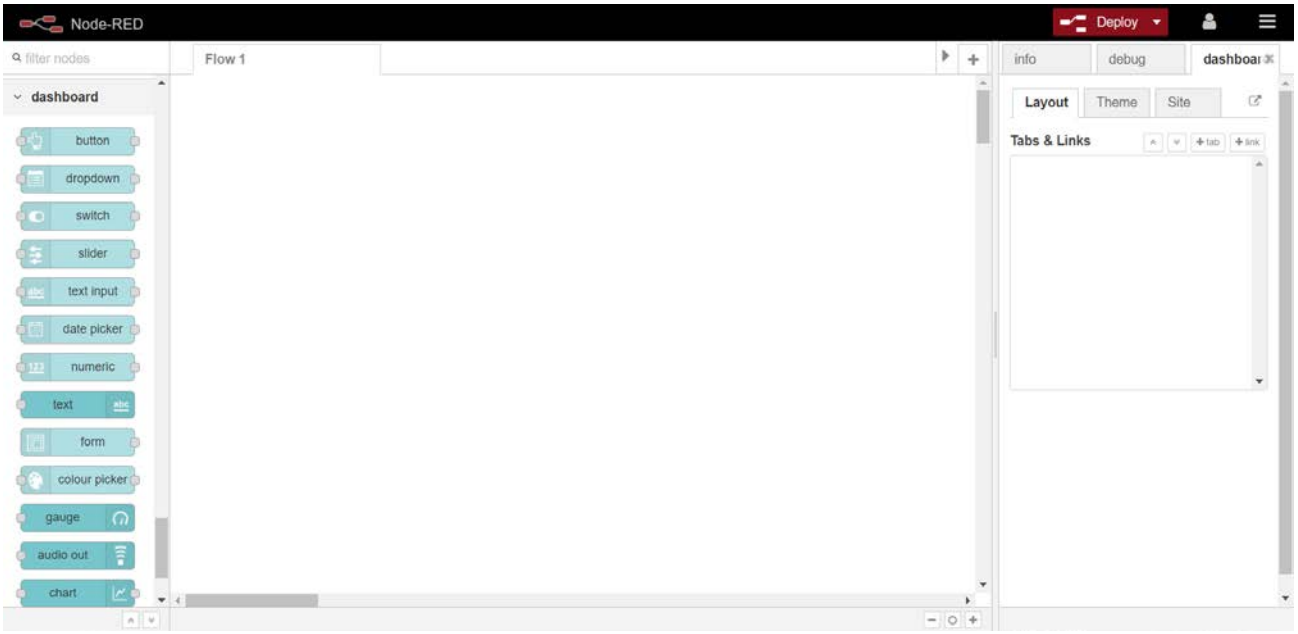
“Palette” -> “Install”, search for “dashboard”.

Install “node-red-dashboard”.

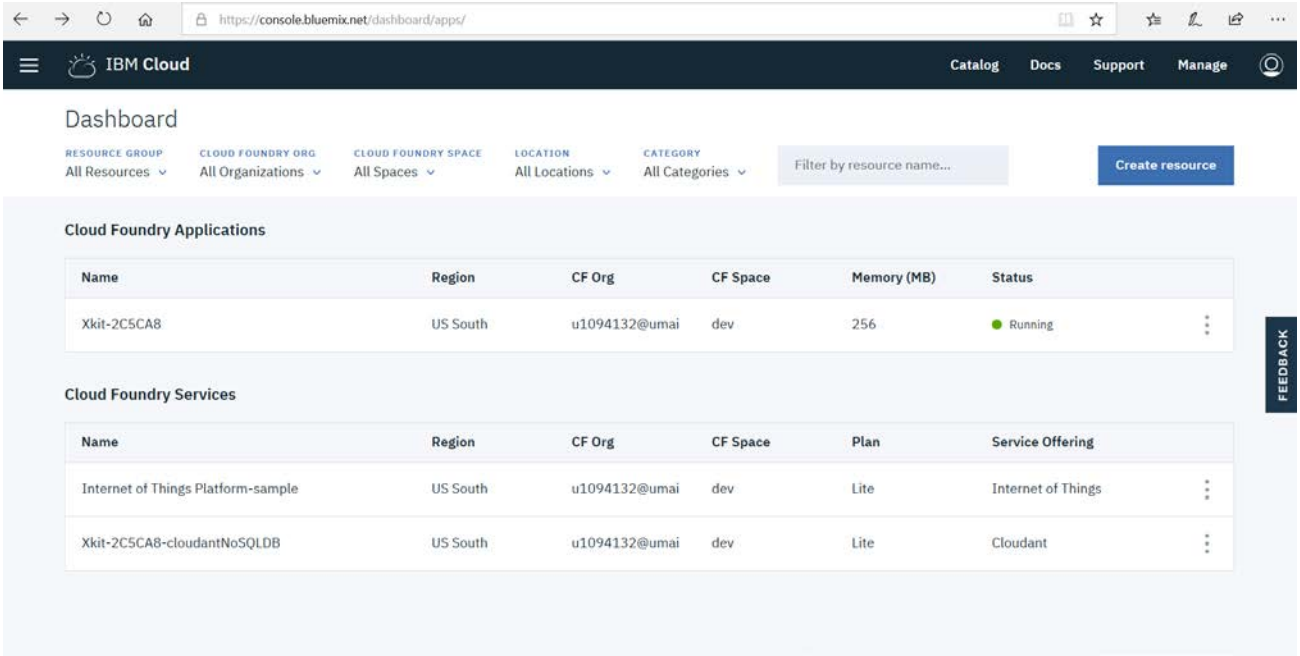


“Close”.

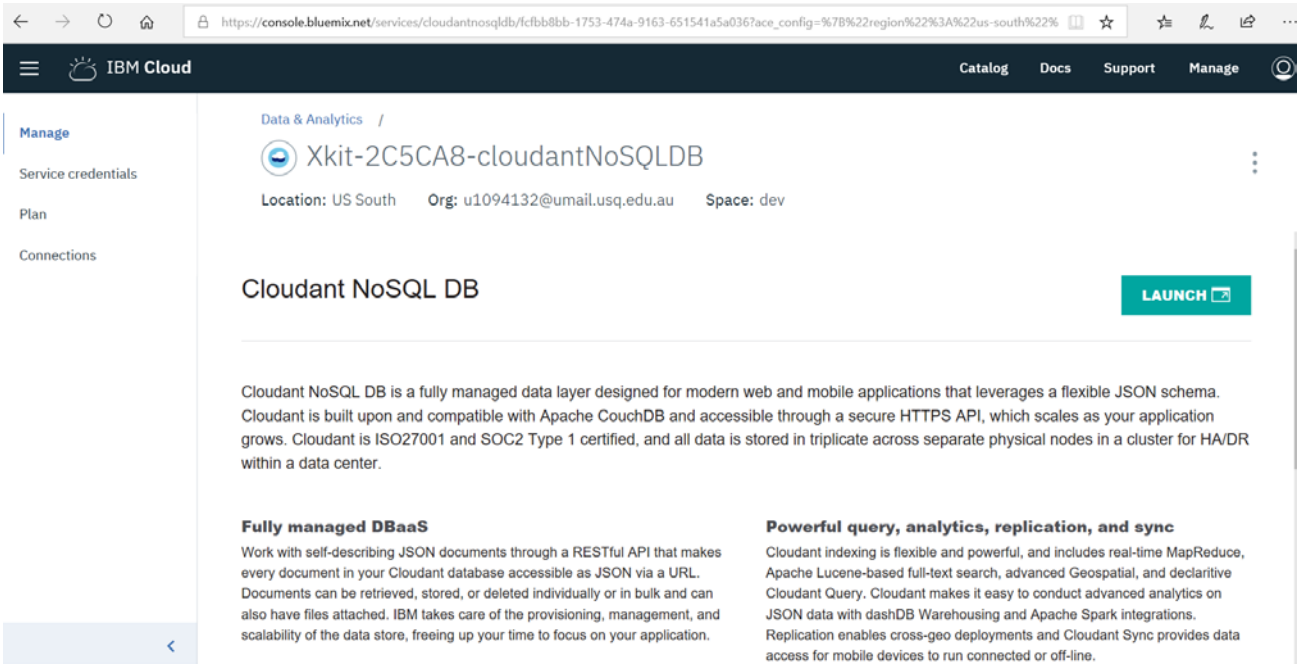
Observe that dashboard nodes and a dashboard tab have been added.



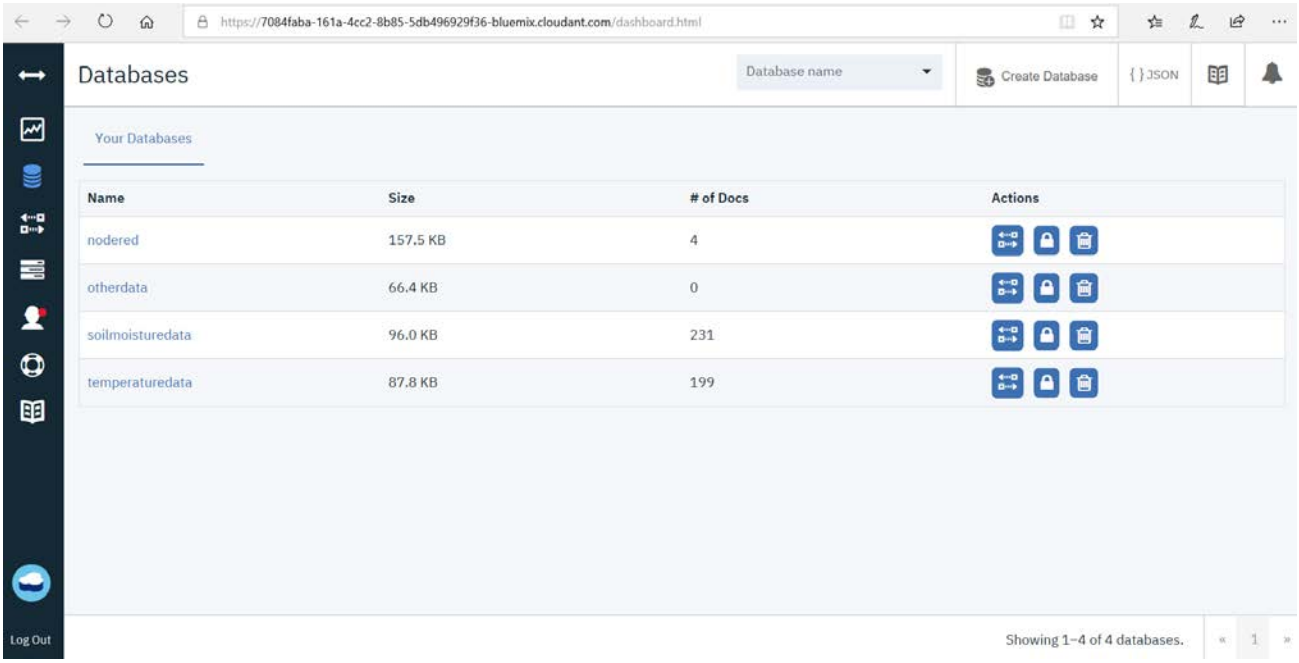
Go to https://console.bluemix.net and login as u1094132@umail.usq.edu.au.



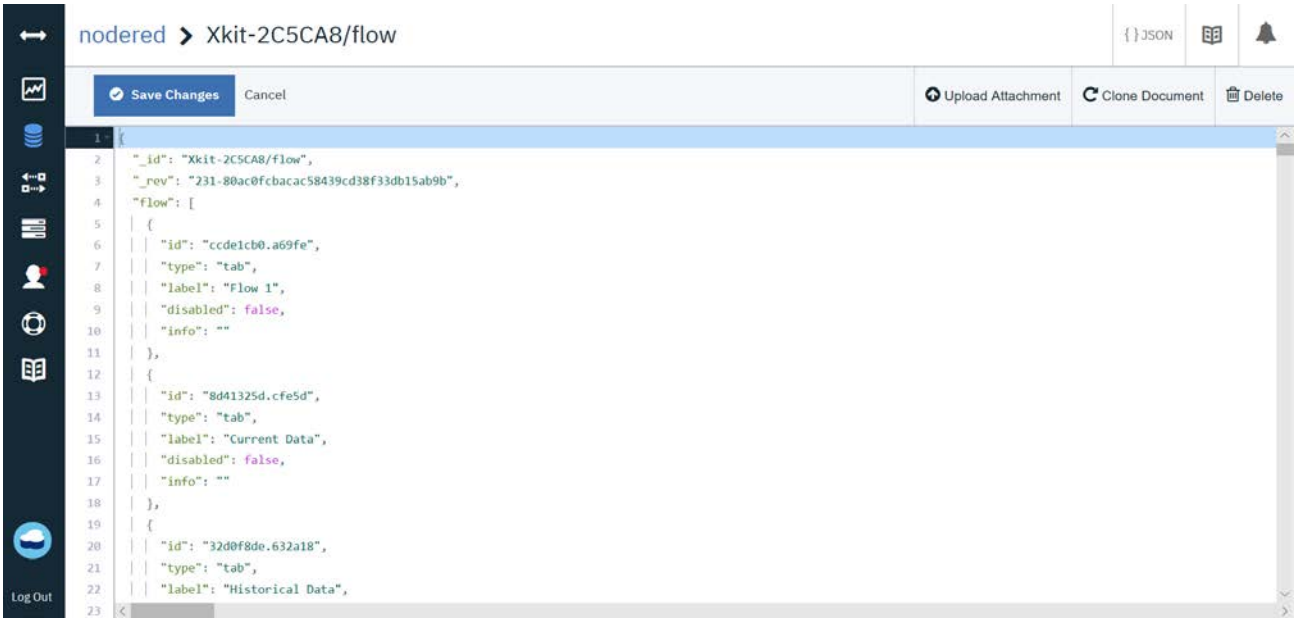
Open “Xkit-2C5CA8-cloudantNoSQLDB” Cloud Foundry Service.



“LAUNCH” in a new tab.



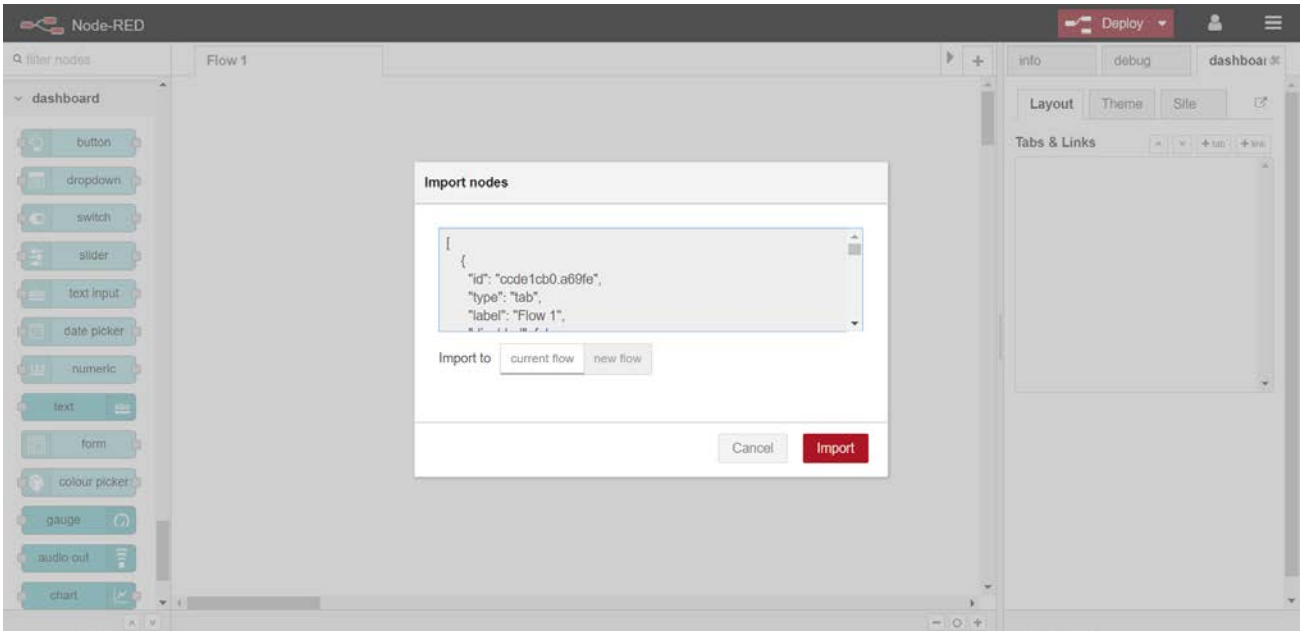
Open “nodered” database and the save the contents of “Xkit-2C5CA8/flow” to a JSON file.



Go back to https://xkit-2bebc3.mybluemix.net/red

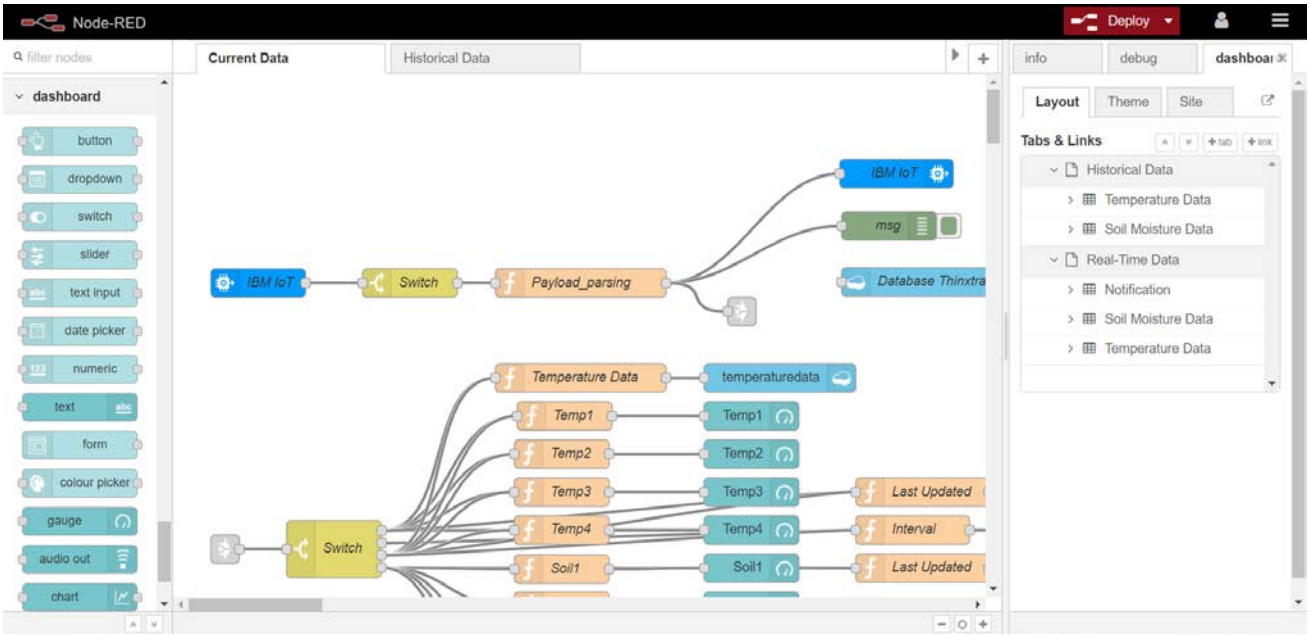
“Import” -> “Clipboard”

Import the copied “Xkit-2C5CA8/flow” to the “current flow”.



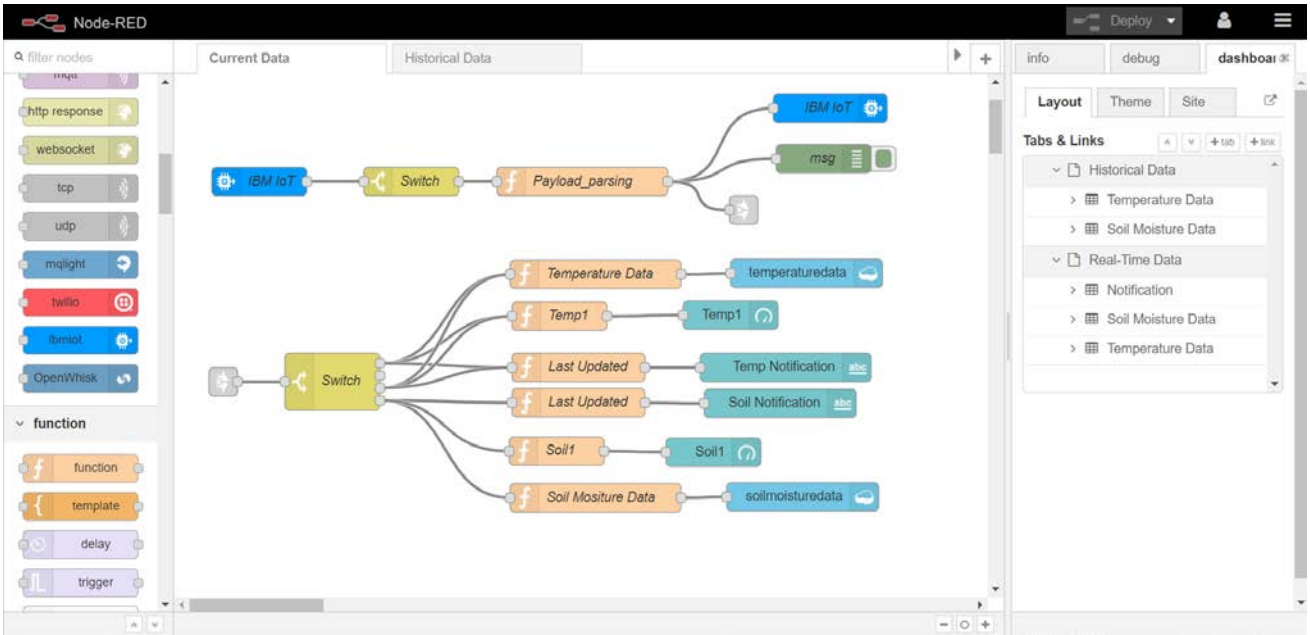
Select “Flow 1” tab, “Flows” -> “Delete”.

Do this for the other “Flow 1” tab as well.

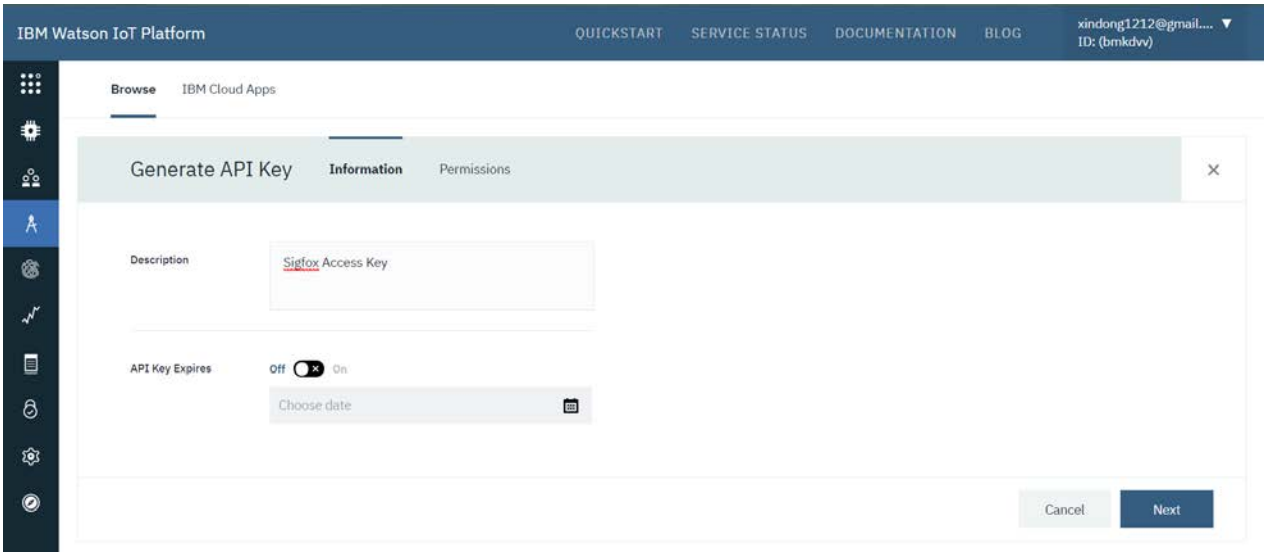


Delete these unused nodes in the “Current Data” flow tab:

“Database Thinxtra”, “Temp2 ” function, “Temp2” gauge , “Temp3 ” function, “Temp3” gauge , “Temp4 ” function, “Temp4” gauge , “Soil2 ” function, “Soil2” gauge , “Soil3” function, “Soil3” gauge , “Soil4” function, “Soil4” gauge , “Interval” function, “Interval Notification”.

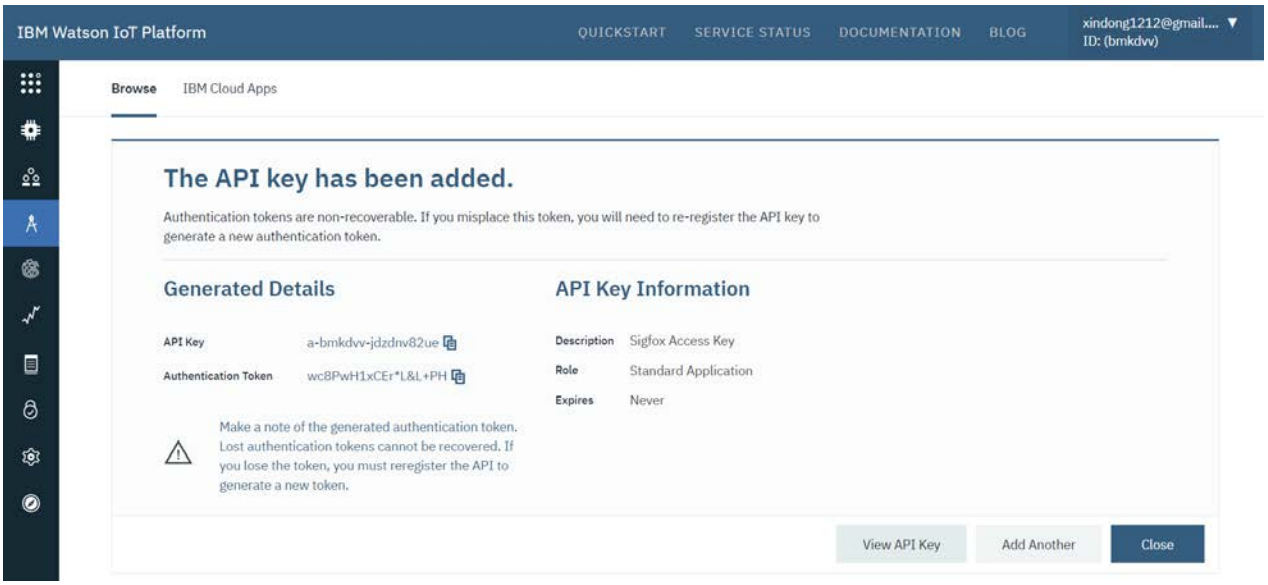


Go to IBM Watson IoT Platform and create another API Key.



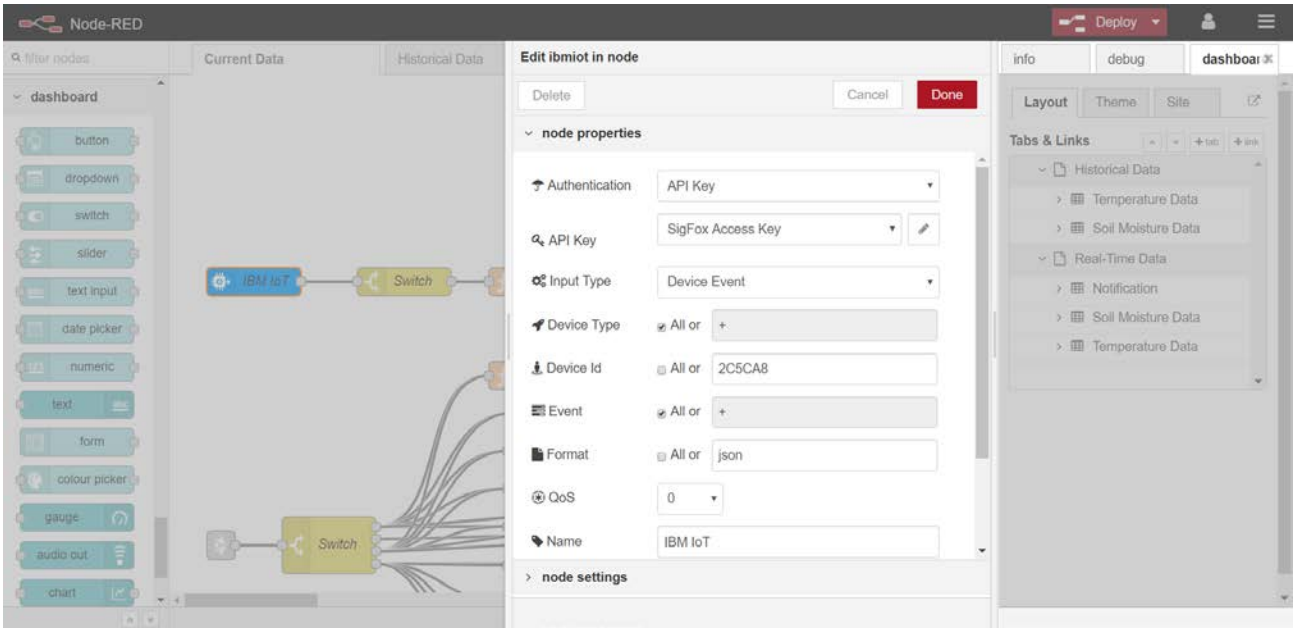
"Next".

IMPORTANT: Take a screenshot of the "Authentication Token".



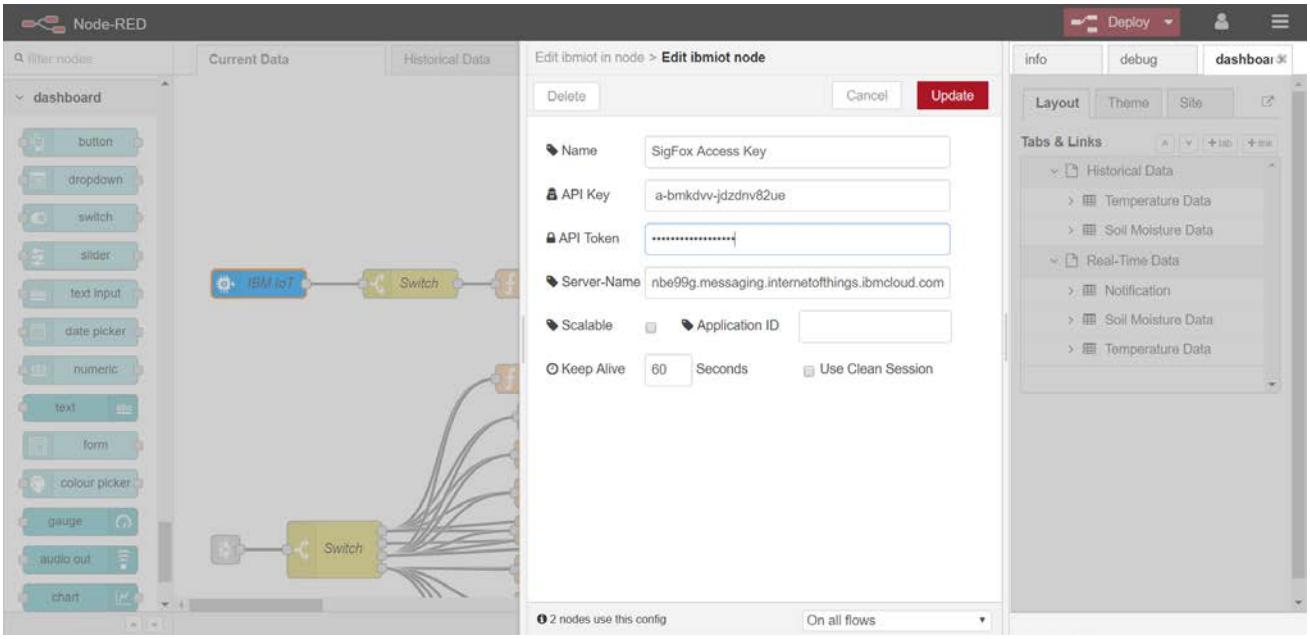
Go back to https://xkit-2bebc3.mybluemix.net/red

Edit “IBM IoT” input node.



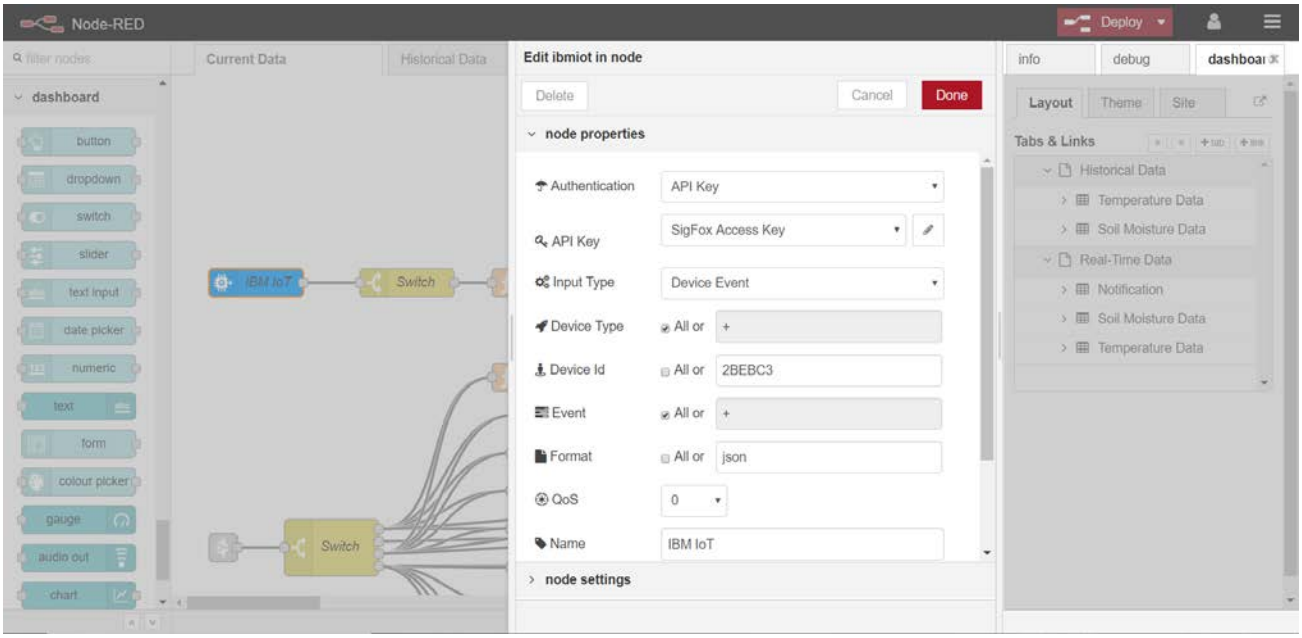
Edit “API Key”.

Enter “API Key” and “API Token” for API Key “a-bmkdvv-jdzdnv82ue”.



“Update”.

Change “Device Id” to “2BEBC3”.

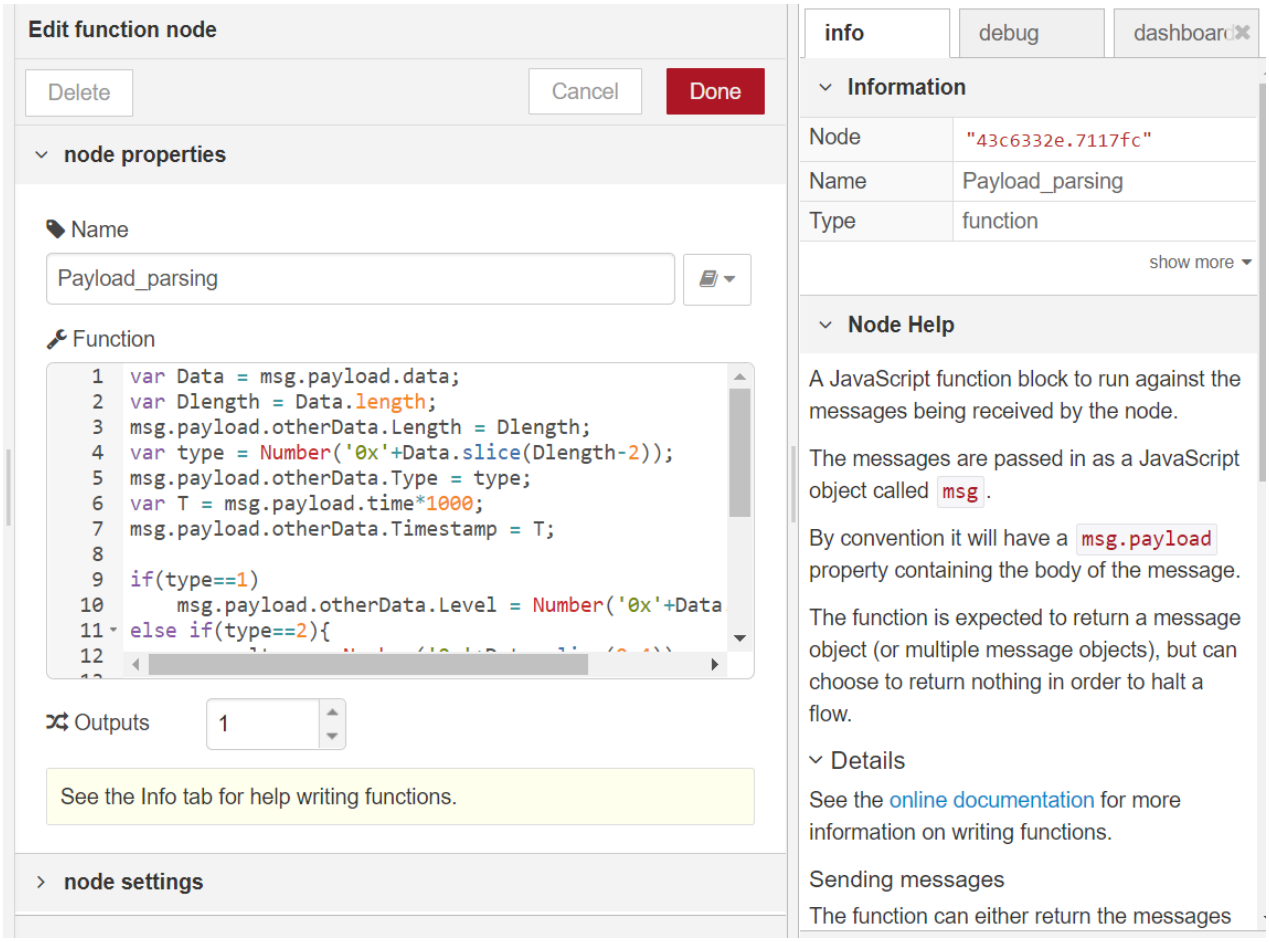


“Done”.

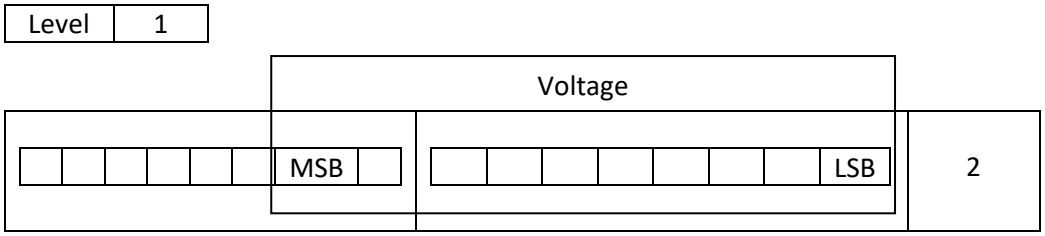
Update “Device Id” for the “IBM IoT” output node as well.

“Deploy”.

Now edit “Payload_parsing” function node.



The two types of Sigfox messages are defined as follows.



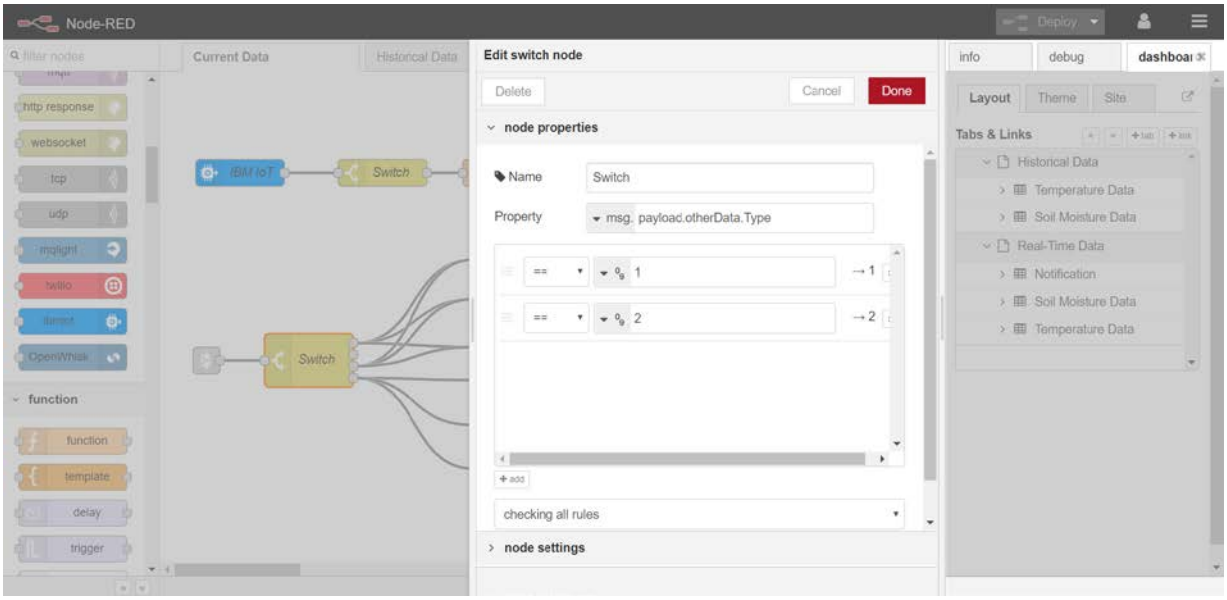
The JavaScript code below is the body of the function.

```
var Data = msg.payload.data;
var Dlength = Data.length;
msg.payload.otherData.Length = Dlength;
var type = Number('0x'+Data.slice(Dlength-2));
msg.payload.otherData.Type = type;
var T = msg.payload.time*1000;
msg.payload.otherData.Timestamp = T;


if(type==1)
   msg.payload.otherData.Level = Number('0x'+Data.slice(0,2));
else if(type==2){
   var voltage = Number('0x'+Data.slice(0,4));
   var height = 0;
   if(voltage!==0)
      height = 25.4 + (voltage+1)/1024*797.6;
   msg.payload.otherData.Height = height.toFixed(2);
}


msg.payload.otherData.datadecoded = '1';
return msg;
```

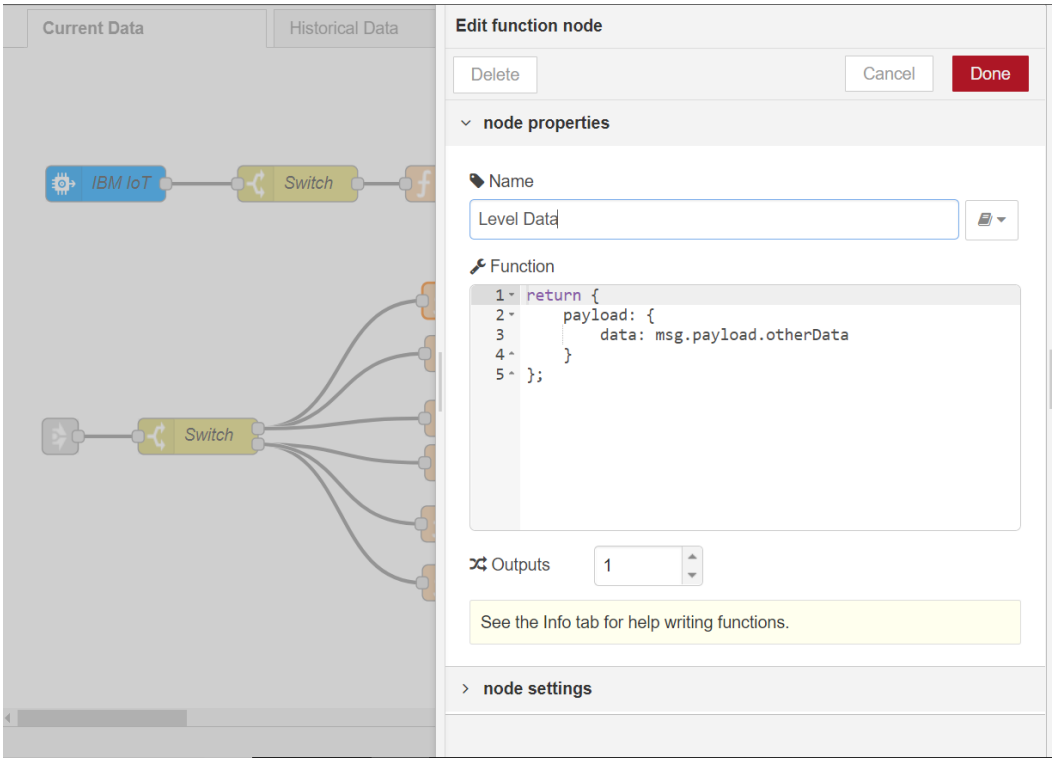
Edit “Switch” node.



“Done”.

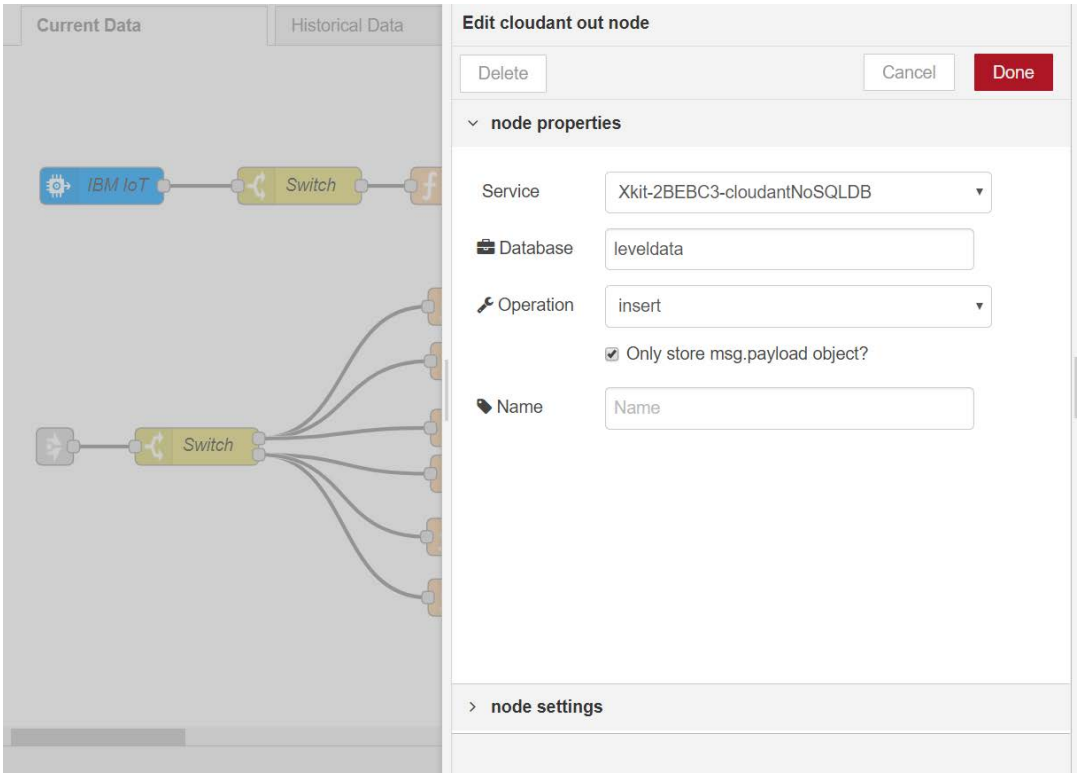
“Deploy”.

Change the name of the “Temperature Data” function to “Level Data”.

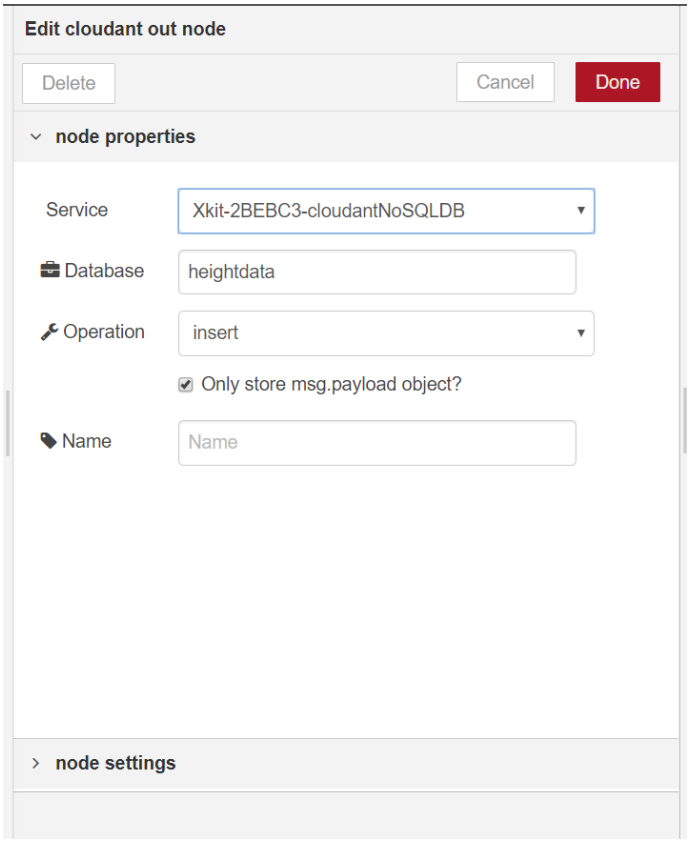


Change the name of the “Soil Moisture Data” function to “Height Data”.

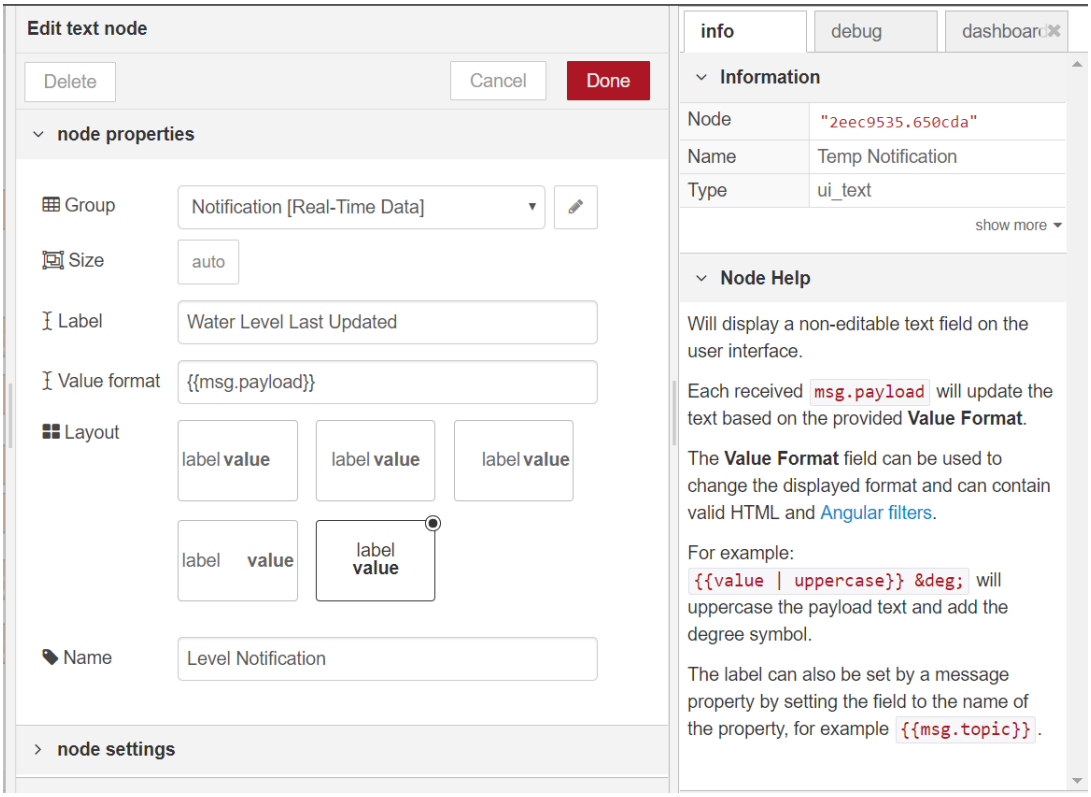
Edit the “temperaturedata” cloudant out node.



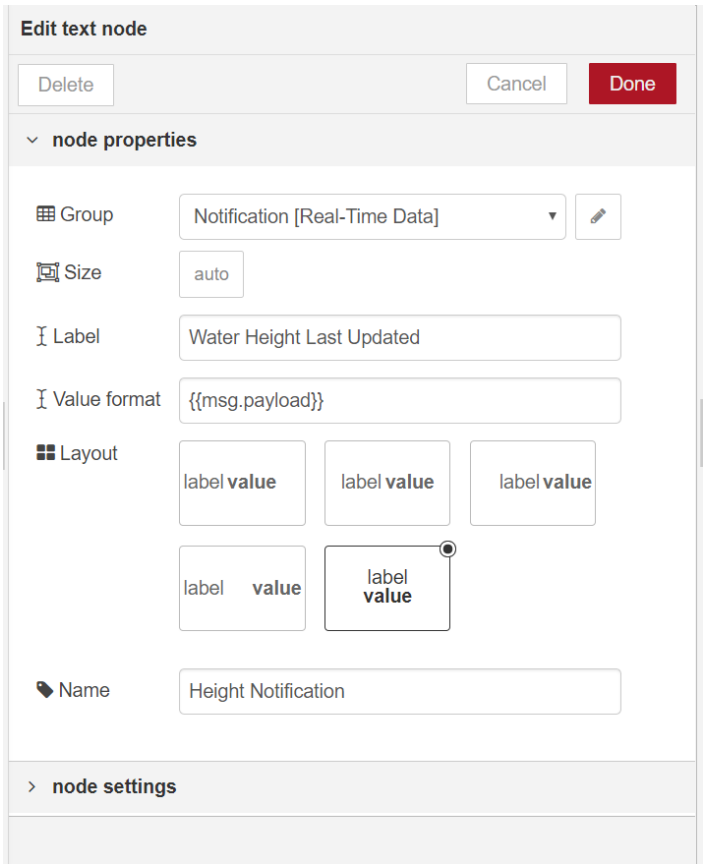
Edit the “soilmoisturedata” cloudant out node.



Edit the “Temp Notification” text node.



Edit the “Soil Notification” text node.

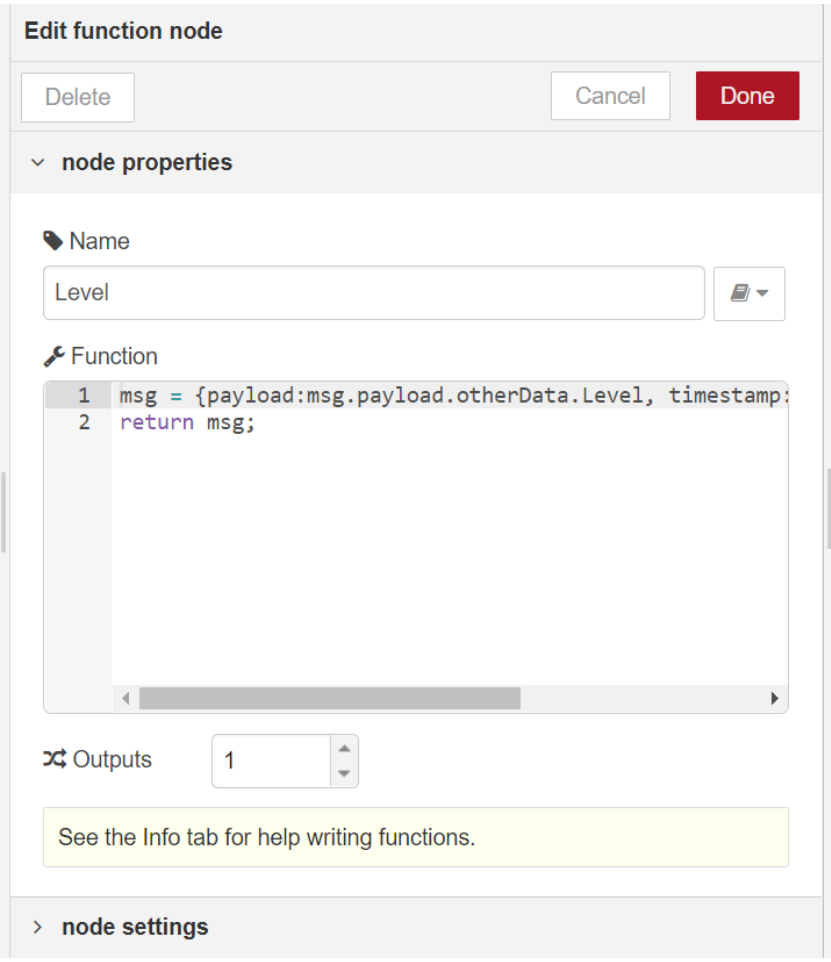


Edit “Temp1” function node.

The body of the function is:

```
msg = {payload:msg.payload.otherData.Level, timestamp:msg.payload.otherData.Timestamp};

return msg;
```



Rename “Soil1” function node to “Height”.

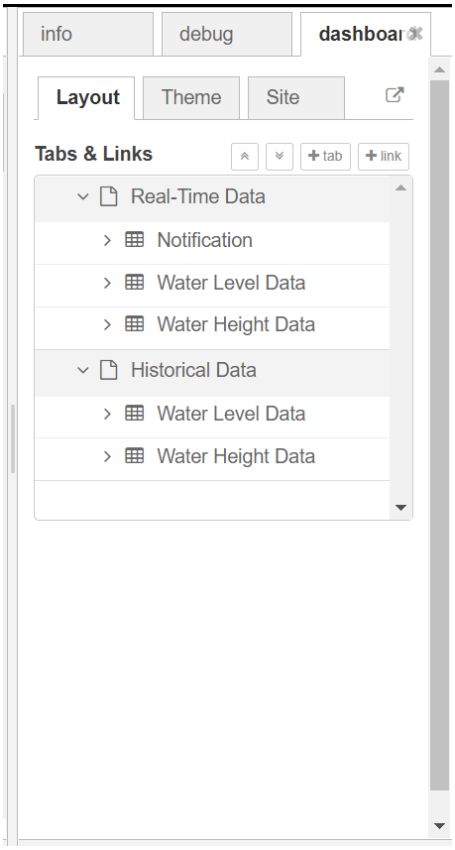
The body of the function is:

```
msg = {payload:msg.payload.otherData.Height, timestamp:msg.payload.otherData.Timestamp};

return msg;
```

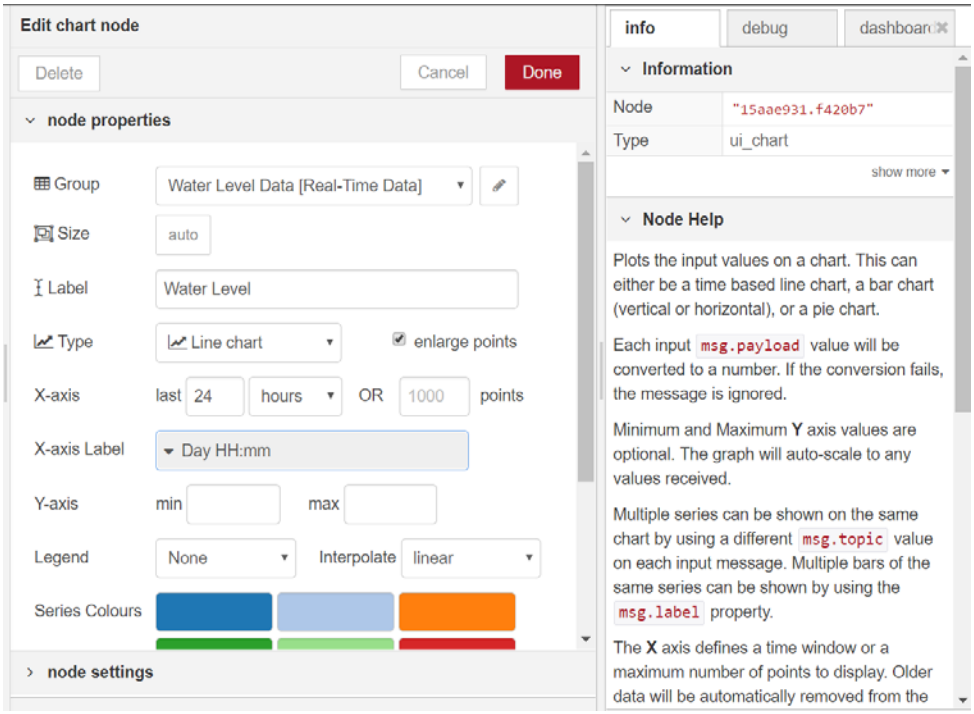
“Deploy”.

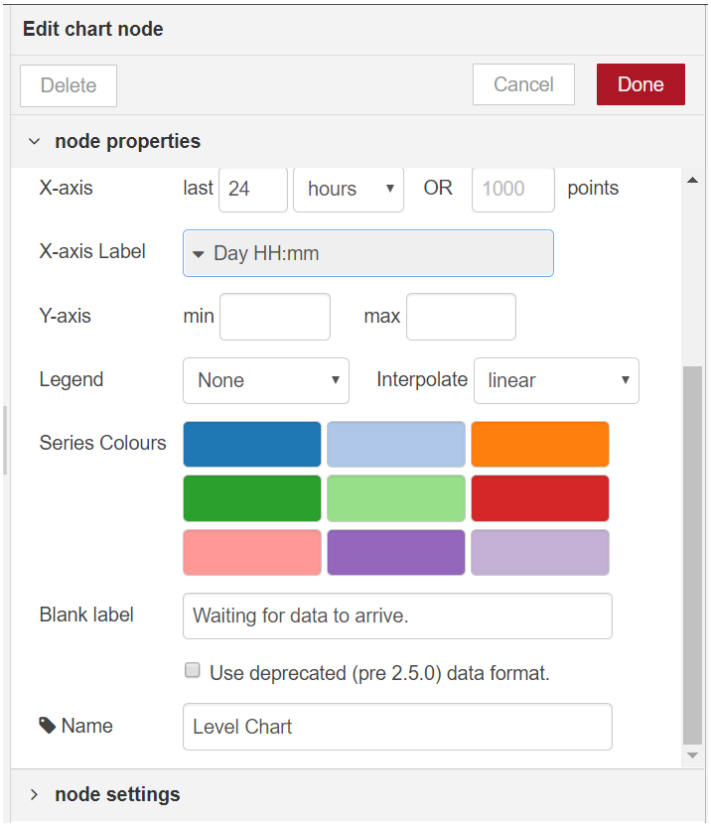
Go to “dashboard” “Layout”, rename the “Soil Moisture Data” and “Temperature Data” groups.



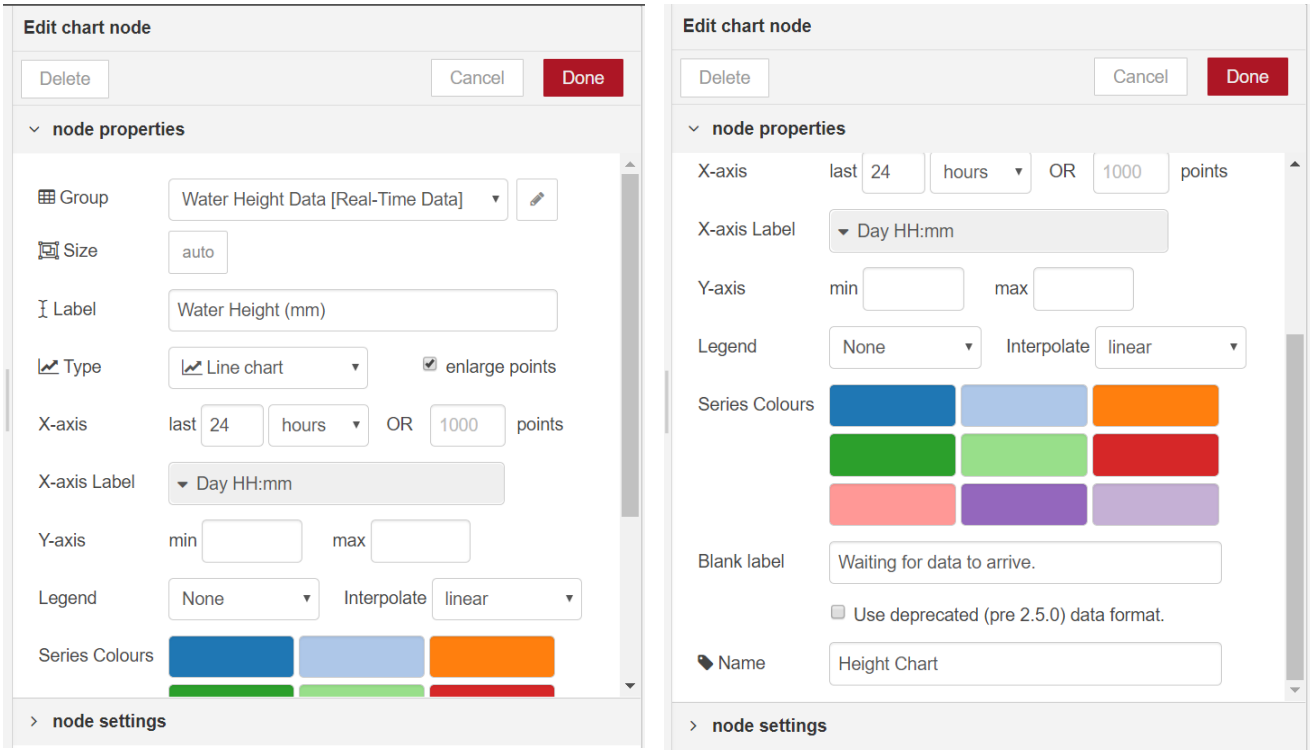
Replace “Temp1” gauge with a “Water Level” live chart.

Change “Group” to “Water Level Data [Real-Time Data]”.



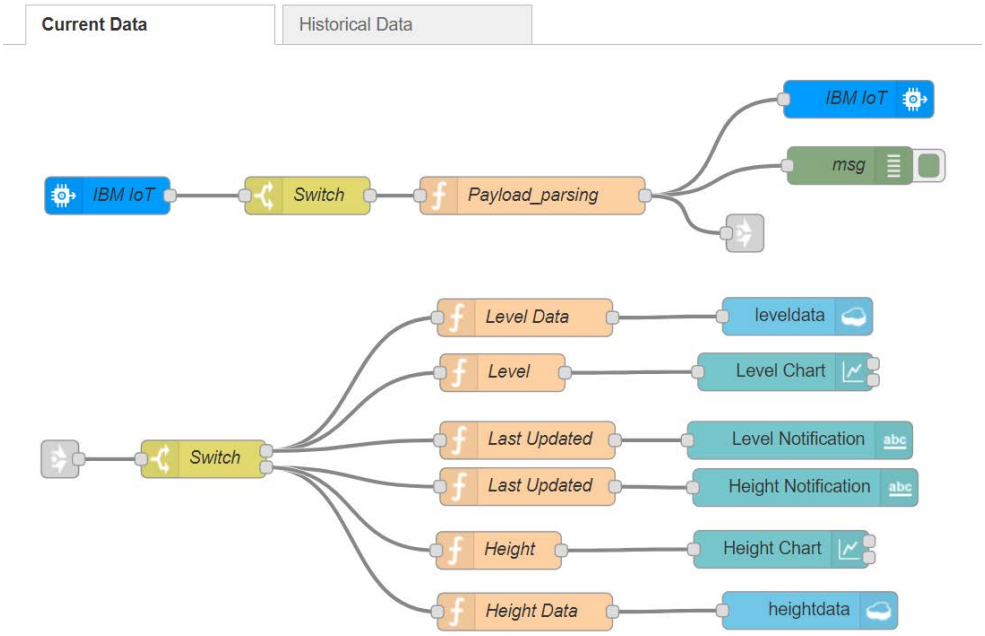


Replace “Soil1” gauge with a “Water Height (mm)” live chart.



“Deploy”.

The final “Current Data” flow:



The dashboards after initial set-up: