{ Lab }

**IoT application in IBM Cloud**

**with Node-RED and IBM Watson**



June 2019 – v1.0

Authors: B. Marolleau, C. Bacle, C. Lalevée

June 2018 - v2.2

© Copyright IBM Corp. 2018

Materials may not be reproduced in whole or in part without the prior written permission of IBM

**Agenda**

[Before Starting 4](#_Toc9191845)

[Hands-on presentation 5](#_Toc9191846)

[Section 1. Overview 5](#_Toc9191847)

[Section 2. Prerequisites 5](#_Toc9191848)

[Cloud: Create Node-RED application and Login 6](#_Toc9191849)

[1. Cloud: Create a virtual sensor and a new flow 10](#_Toc9191850)

[Section 1. Sensors & IoT 10](#_Toc9191851)

[Section 2. Node-RED flow: creation & import 11](#_Toc9191852)

[2. Node-RED on PI:   MQTT Simulator 14](#_Toc9191853)

[3. Cloud: Visual Recognition 15](#_Toc9191854)

[4. Cloud: Camera & Dashboard 16](#_Toc9191855)

[5. Node-RED on PI: Camera & Dashboard 17](#_Toc9191856)

[6. Node-RED on PI: Grovepi Hat & Sensors (Optional #1) 18](#_Toc9191857)

[7. Node-RED on PI: Insert IoT Data in Cloudant DB (Optional #2) 19](#_Toc9191858)

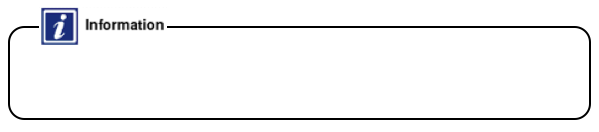
[8. Node-RED on PI: Process IoT Data with Watson Language Translator (Optional #3) 21](#_Toc9191859)

[9. Create a dashboard application in Node-RED (optional #4) 23](#_Toc9191860)

[Section 1. Import Node-RED Dashboarding capability 23](#_Toc9191861)

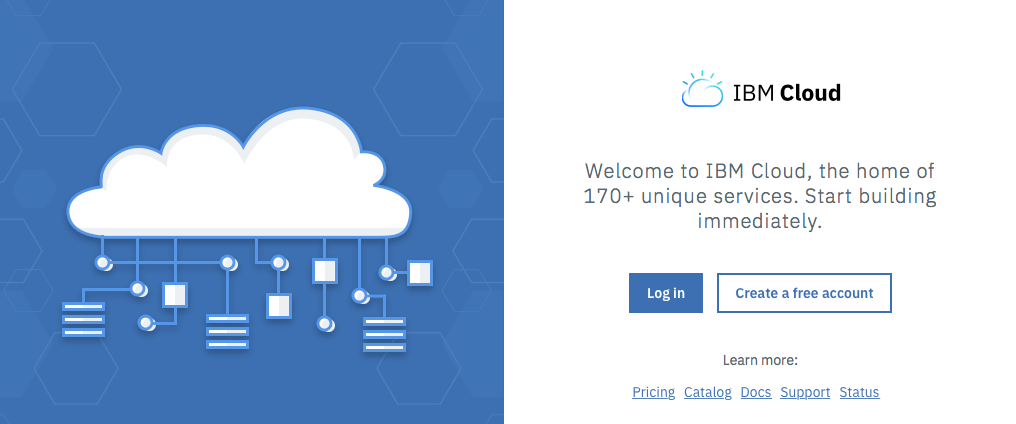
[Section 2. Create a simple Node-RED Dashboard 26](#_Toc9191862)

[Section 3. Add voice alert on dashboard 29](#_Toc9191863)

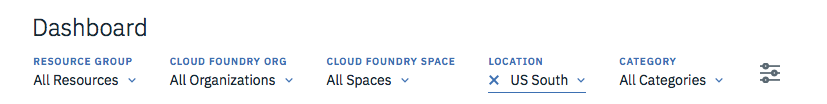
Before Starting

This hands-on required to have an IBM Cloud account. If you don’t, you can create one here: <http://bluemix.net/>.

* Open a browser and access to IBM Cloud: <https://console.bluemix.net>.
* If you have an IBM Cloud account, click **Log in**, and enter your IBM ID credentials.  
  If you don’t have an IBM Cloud account, click **Create a free account**. Enter your email address, and additional information required. You will receive an email with activation link. Once activated, you could use your new free IBM cloud account: log in.



* Select organization, location and space to use during this lab.



* If needed, free resources (GB / #Services) in your IBM Cloud Organization & Spaces to run the lab exercises.   
  If you encounter a resource contention (error message saying you are out of resources), clean up your spaces by deleting existing Apps or Services.

Hands-on presentation

* + 1. Overview

In this hands-on session, you will create a Node-RED application in IBM Cloud to collect, store and display virtual sensor data.

Node-RED (<https://nodered.org/>) is a flow-based programming tool, originally developed by the IBM Emerging Technology Services team (in early 2013) and now a part of JS Foundation.

Traditional development can be very technical, but Node-RED enables you to concentrate on the logic of your workflow and allows fast prototyping.

Node-RED consists of a Node.js-based runtime with a flow editor accessed through a web browser. Within the browser, you create your application by dragging nodes from a customizable palette into a workspace and start to wire them together. With a single click, the application is deployed back to its runtime.

Session objectives are:

* Create & modify an application using Node-RED in IBM Cloud and on a device (Raspberry)
* Discover new services & Node-RED to consume or create services (IoT / AI /Database)
* Discover Watson services
* Discover how to integrate Cloud Services with a Raspberry PI or any device using Node-RED and create prototype rapidly.
  + 1. Prerequisites

2 kinds of exercises: in the Cloud (IBM Cloud) and local (Node-RED on a Raspberry PI).

The necessary hardware; Raspberry PI 2 with an internet access and Node-RED installed. A PC connected to the Raspberry PI network.

Setup & Connectivity check

1/ connect the raspi - usb power

2/ from the PC - go to <http://raspberry314-pot.eu-de.mybluemix.net/home> .

then click on your team picture and get the IP of your raspi on the network.

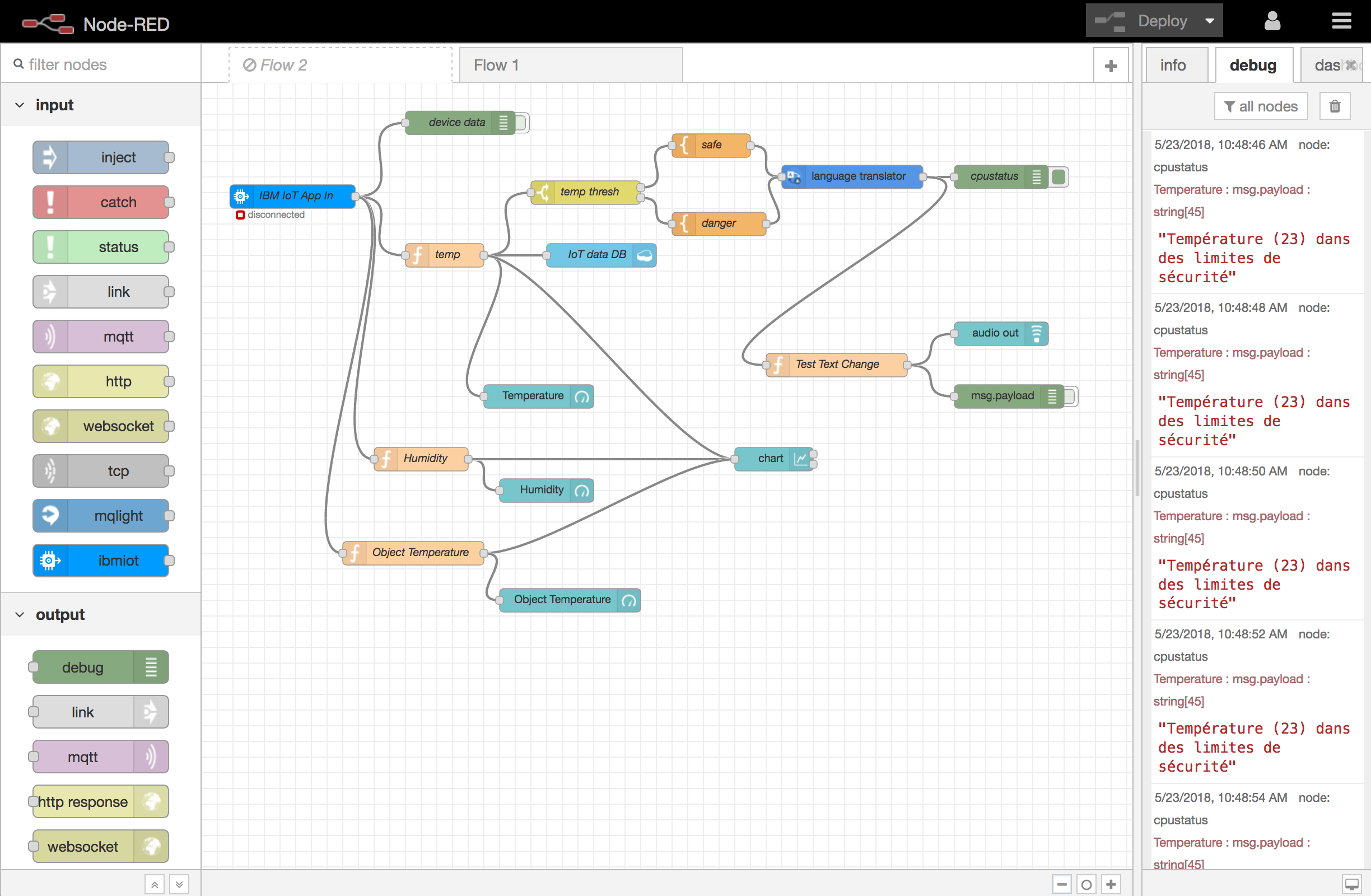
3/ On your PC, connect to [http://YOUR-RASPI-IP:1880](http://your-raspi-ip:1880/) from a Browser.

4/ logon IBM Cloud as mentioned in “Before Starting” section above.

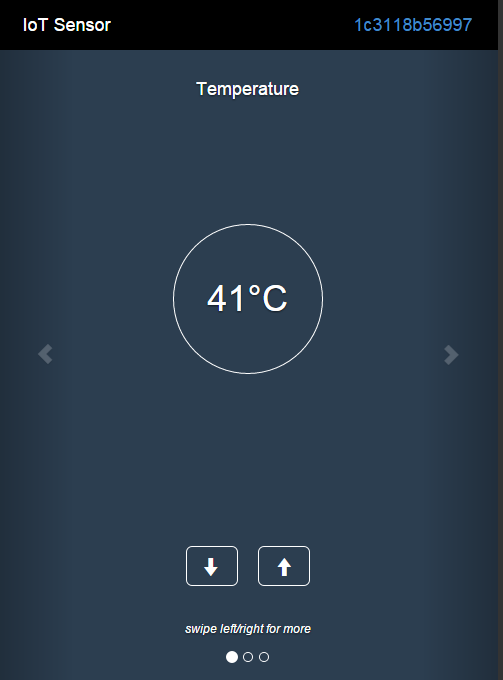
Cloud: Create Node-RED application and Login

Expected results are:

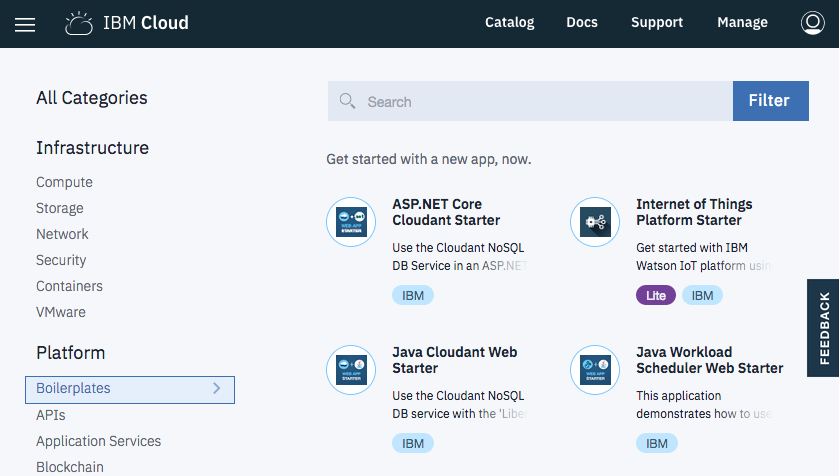
* Your Node-RED application is operational (using Node.js runtime), accessing Cloudant & IoT platform (QuickStart)



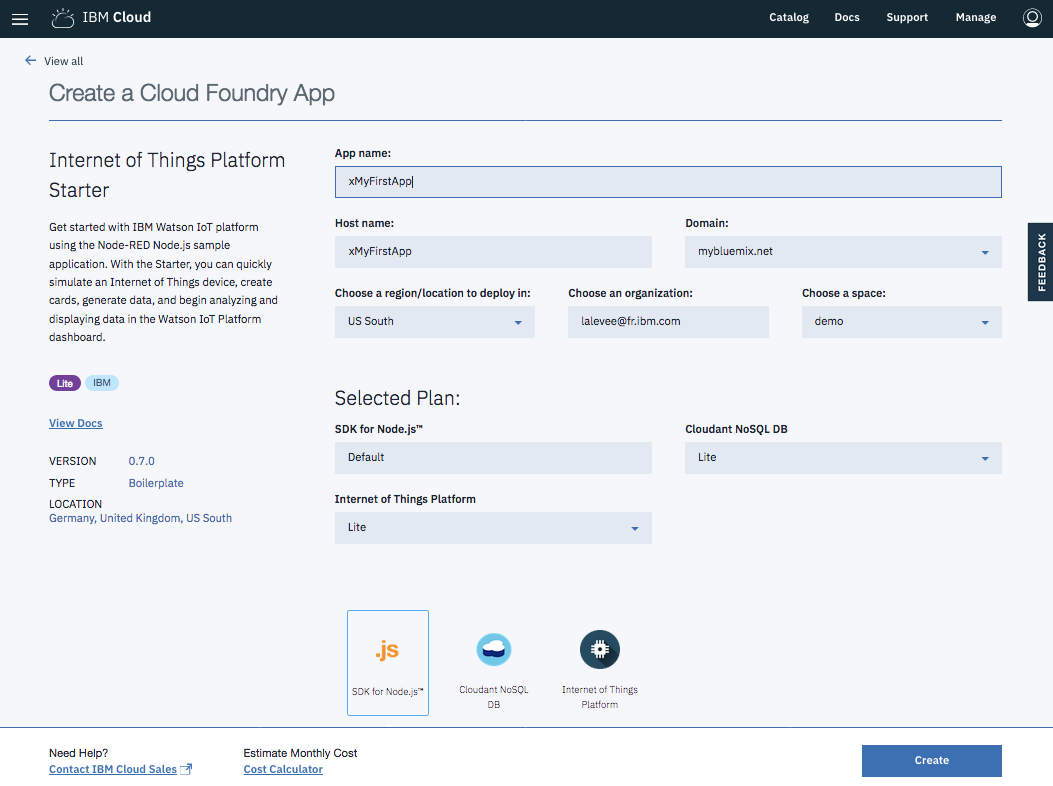
* Your Node-RED app is online (reachable from the Internet), & will be connected to a temperature simulator (sensor)



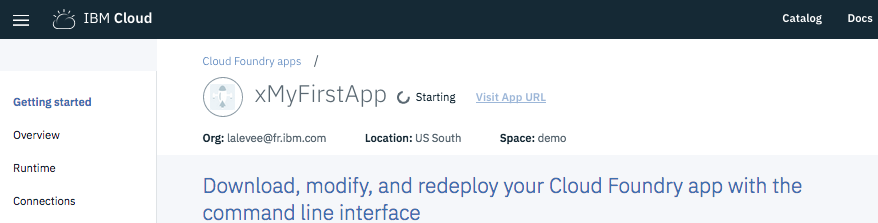
1. In IBM Cloud Catalog, choose **Boilerplate** category



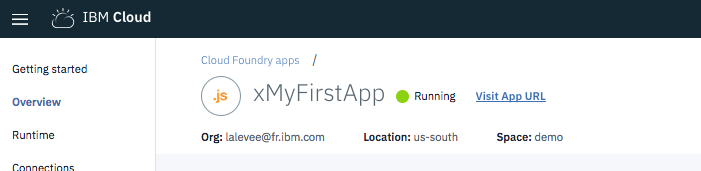
1. Click on **Internet of Things Platform Starter** to create an instance:   
   Fill in the App Name & Host name fields.  
   Note: Node-RED is a Node.js based application: using this boilerplate will instantiate a Node.js runtime + a Cloudant (NoSQL DB) service.



1. Click **Create**.   
   Wait for the environment to be created & the app to start (~4 minutes).

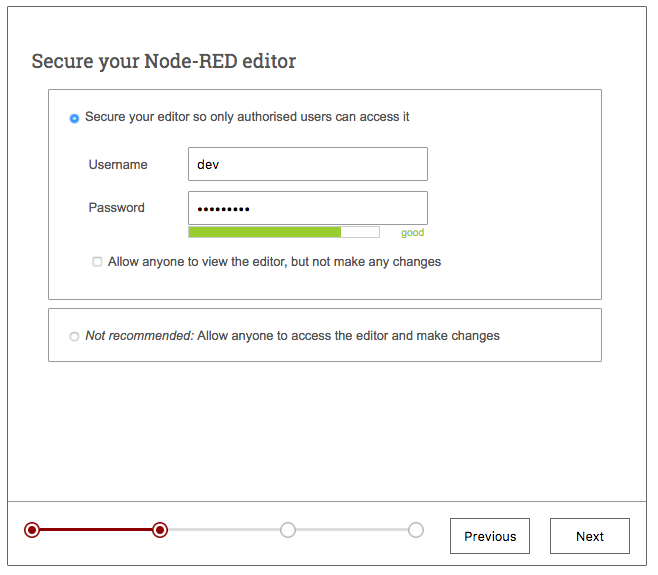


1. By clicking **Visit App URL**, access the Node-RED application

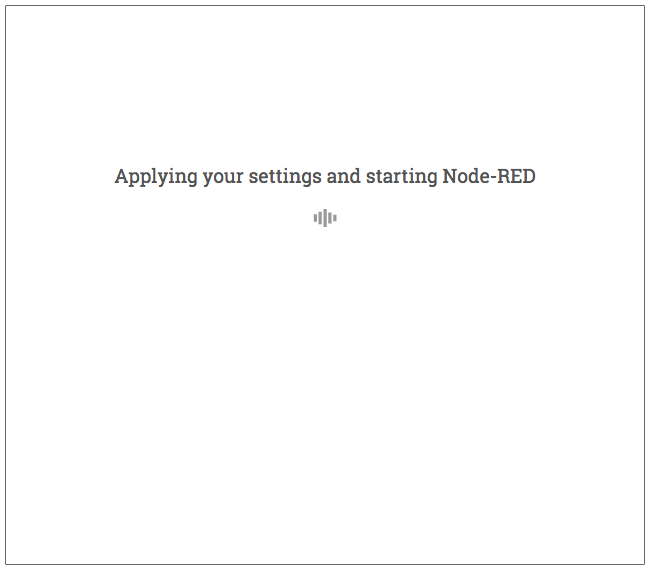


1. Run the wizard to configure authentication: secure your editor with your own credentials so only authorized users can access it (Node-RED has its own authentication system).

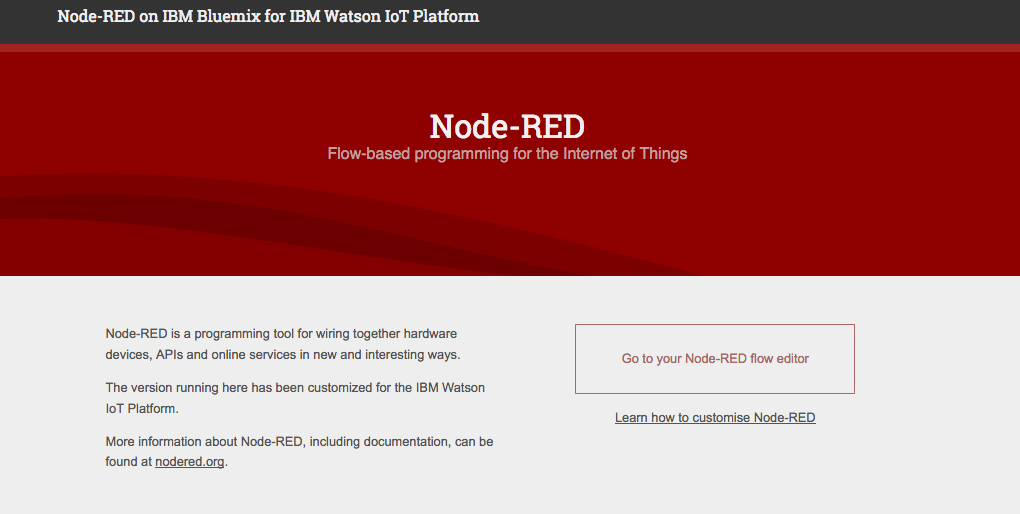
Don’t check **Allow anyone to view the editor, but not make any changes** and **Allow anyone to view the editor**.



1. On wizard last step screen, click on **Finish** to start Node-RED.



1. Node-RED is a browser-based editor that makes it easy to wire together flows that can be deployed to the runtime. In the case of IoT, Node-RED is really powerful to quickly test all the possibilities that IBM Cloud offers with different kind of services.   
   Your Node-RED app has a public URL like any web app (you defined it in step 2).  
   Click on **Go to your Node-RED flow editor** and use the credentials provided before.



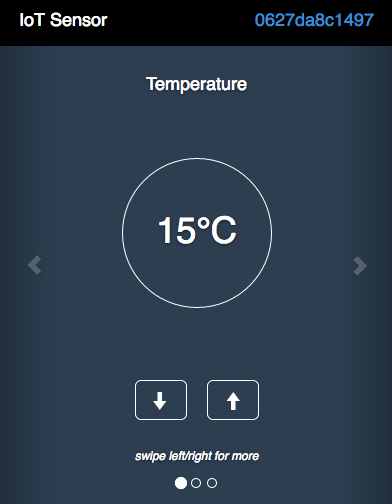
1. You now have access to the Node-RED UI.  
   Keep the existing default flow and create a new flow: click **+**. You will use it in next exercise.



* 1. Cloud: Create a virtual sensor and a new flow
     1. Sensors & IoT

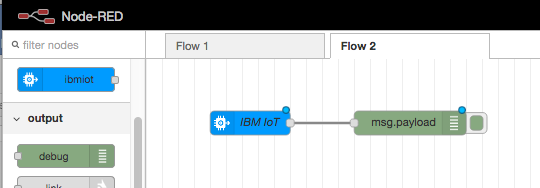
1. Open a new window or tab in your browser.
2. To create a sensor simulator, connect to <http://ibm.biz/iotsensor>  
   There are 3 simulated sensors:
   * Object temperature
   * Temperature
   * Humidity

The simulator (from IBM Cloud IoT Quickstart) connects automatically and starts publishing data.  
It must remain connected to visualize the data.  
Use the simulator buttons to change the simulated sensor readings. Data is published periodically.

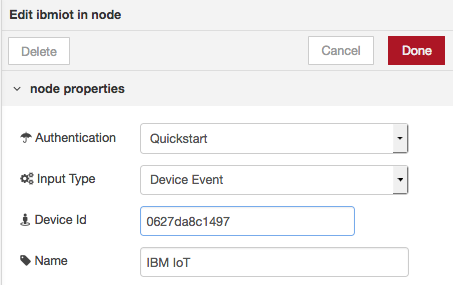


Note: Instead of using your desktop browser, you can use your smartphone.

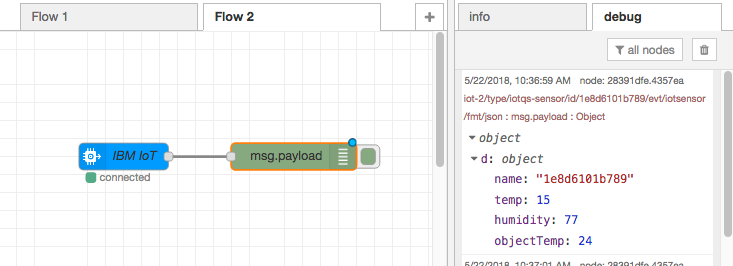
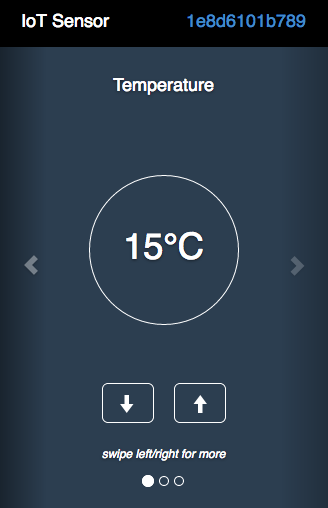
1. Identify your virtual device ID (top right corner) : copy it. You will use it in next section.  
   Warning: if you reload this page, the device ID changes.  
   * 1. Node-RED flow: creation & import
2. Go back to Node-RED window
3. From left panel, drag and drop nodes to the workspace
   * Chose the Input node **ibmiot**
   * Add an output **debug** node
   * Link them



1. Configure IBM IoT by double clicking on it :
   * Authentication: Quickstart (means it is a simple authentication – for demo purposes)
   * Device ID : <The value from Section 1, step 3 - Generated by the Simulator>



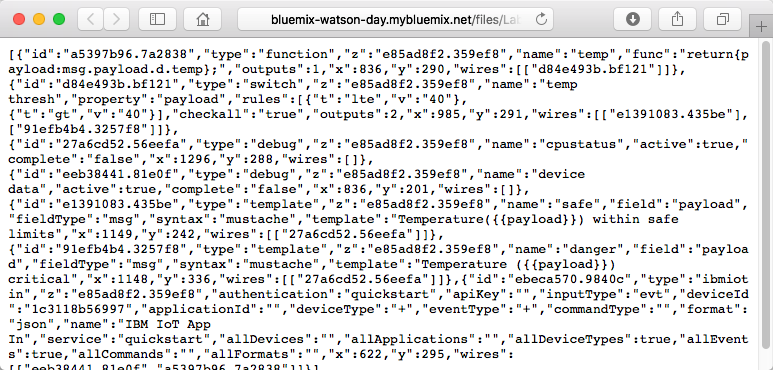
1. Click **Done** & deploy your flow by clicking the **Deploy** button (top right).
2. Check the **Debug** Panel on the right side while you are playing with the sensor simulator. You should receive Device (sensor = web app. you opened in other window) data as the IBM IoT Node subscribed to this particular Device topic.



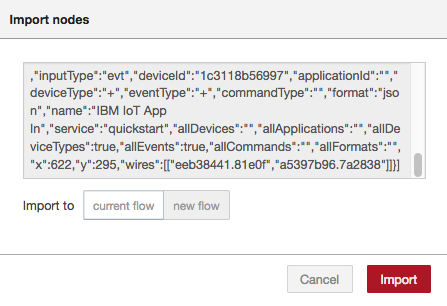
1. Delete the whole flow by selecting all the nodes & pressing the ‘Delete’ key.



1. Now import a new flow. A flow can be exported and imported using JSON file.  
   Open link <http://ibmcloud-watson-day.mybluemix.net/files/Lab3_IoT_Flow.v2.1.json> to display code in JSON format. You can also open the file you downloaded previously.



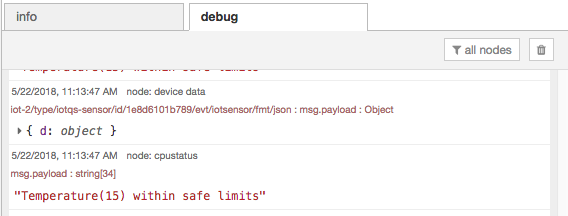
1. Select all and copy it
2. Click on the top right button near **Deploy**.  
   Select **Import**, **Clipboard** & copy/paste the content of the JSON file in **Current flow**.



1. Click on workspace to paste imported nodes
2. Fill in the **Device ID** field in the **IBM IoT App In** node.



1. Click **Deploy** to deploy the new Flow.   
   Modify the Device Temperature & check the **Debug** logs.



* 1. Node-RED on PI:   MQTT Simulator

To do Yves

* 1. Cloud: Visual Recognition

To do Yves

* 1. Cloud: Camera & Dashboard

To do Yves

* 1. Node-RED on PI: Camera & Dashboard

To do Yves

* 1. Node-RED on PI: Grovepi Hat & Sensors (Optional #1)

To do Benoit

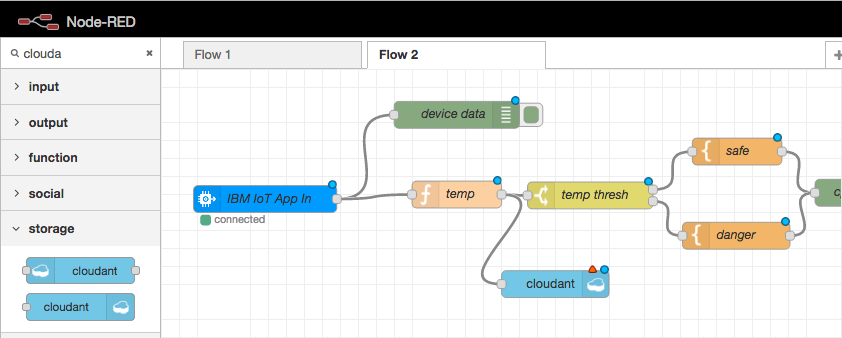
Flow on github

* 1. Node-RED on PI: Insert IoT Data in Cloudant DB (Optional #2)

Let’s insert the event data coming from the Device sensors in a Cloudant database!

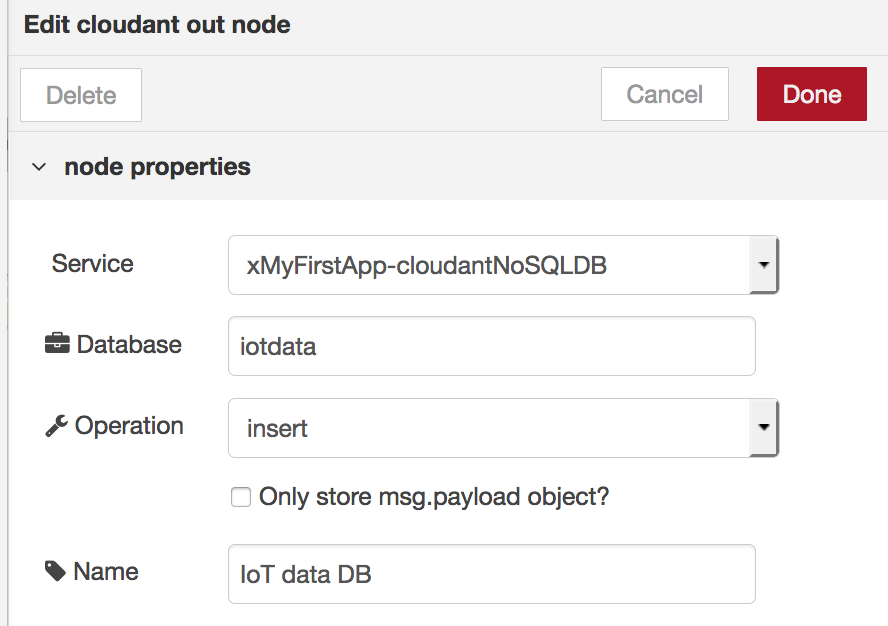
Remember that you already have a Cloudant service deployed for Node-RED. You will use it to store your data.

1. Add a Cloudant Node (**Cloudant OUT** node in the **Storage** Category) & link it to the **temp** function node

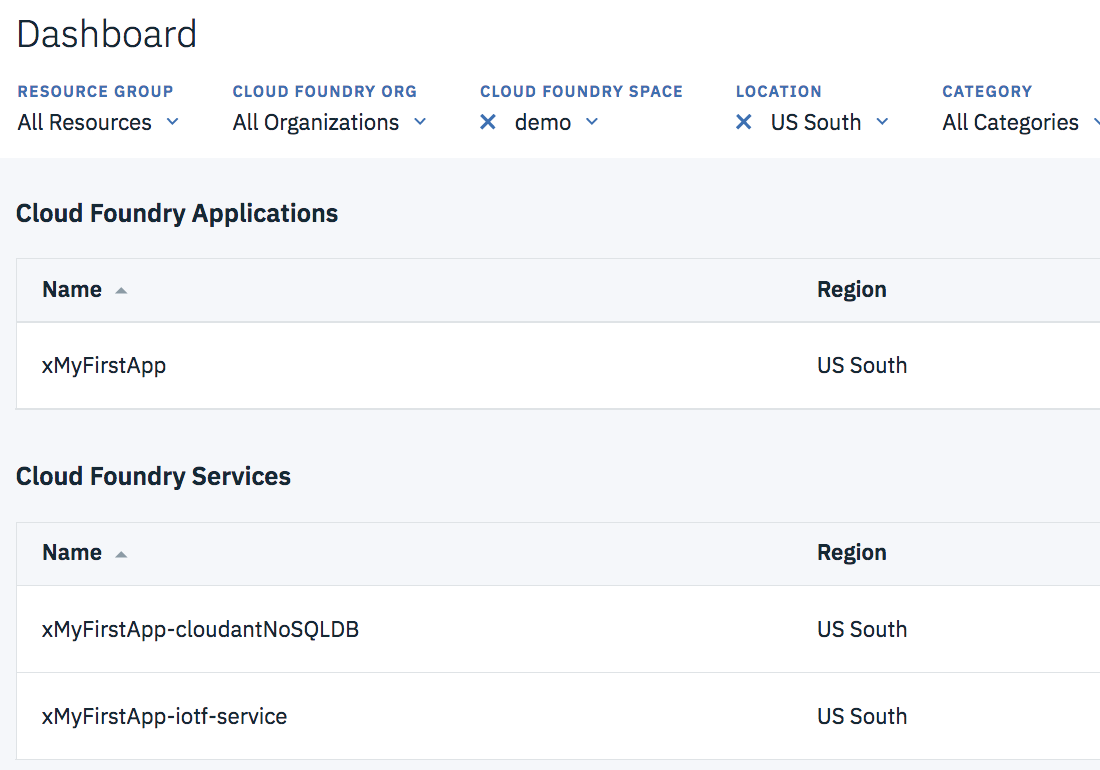


1. Configure it:
   * Service : Cloudant service name bound to your Node.js runtime.   
     *As Node.js is already bound to a Cloudant Service, the service name should appear in the Drop-down list.*
   * Database: name of your choice (lower case)
   * Name (node): name of your choice

Click **Done**.



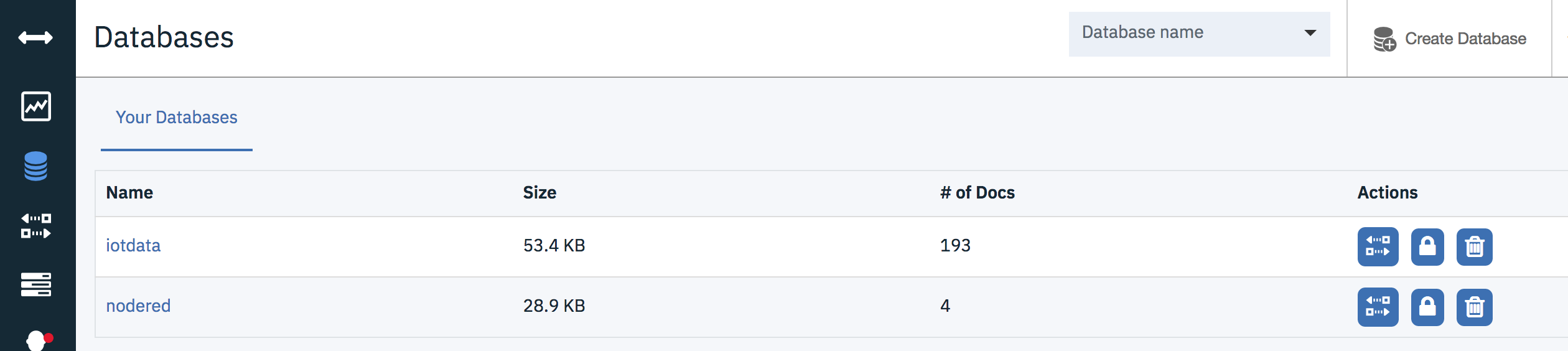
1. Deploy your new flow (**Deploy** button)
2. From your IBM Cloud Dashboard (IBM Cloud window of your browser), start the Cloudant dashboard by clicking on the line of Cloudant service



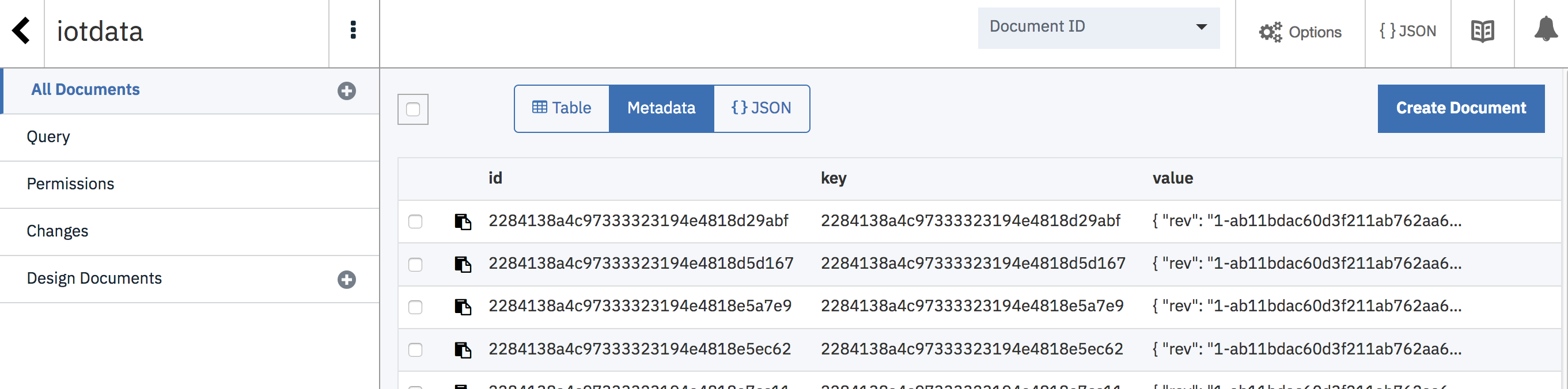
1. Click **Launch** button to start Cloudant console



1. Select **Database** icon in left panel, then your database name (defined in step 2).

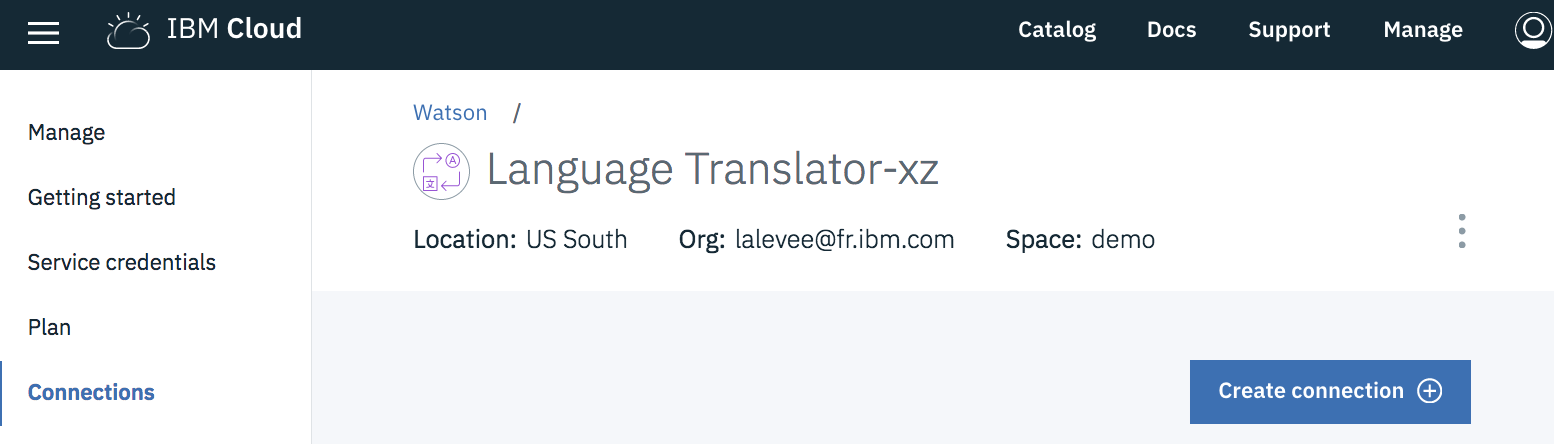


1. Have a look to the inserted data in the database. Click on record to see content.

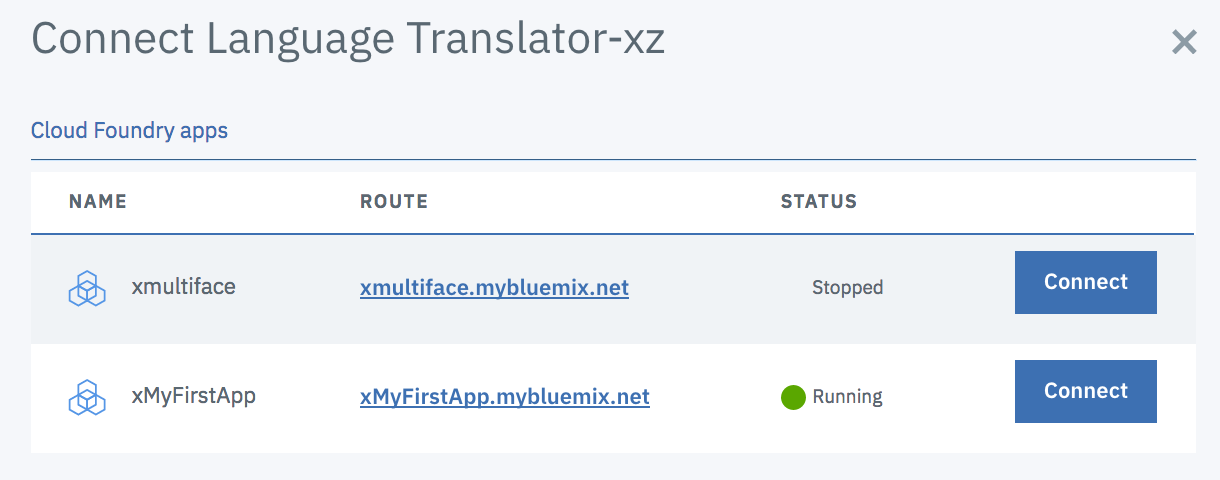


* 1. Node-RED on PI: Process IoT Data with Watson Language Translator (Optional #3)

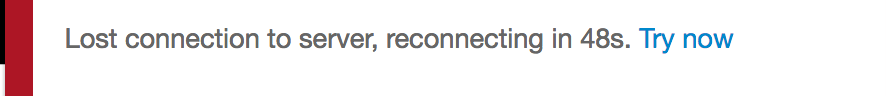
1. In IBM Cloud windows, add a « Watson Language Translator » service to your existing Node-RED application. From **Catalog**, click on **Language Translator** in Watson category.  
   Select **Lite** plan, then click **Create**. Service is deployed.
2. From Language Translator dashboard, **Connections** menu, click on **Create connection**.



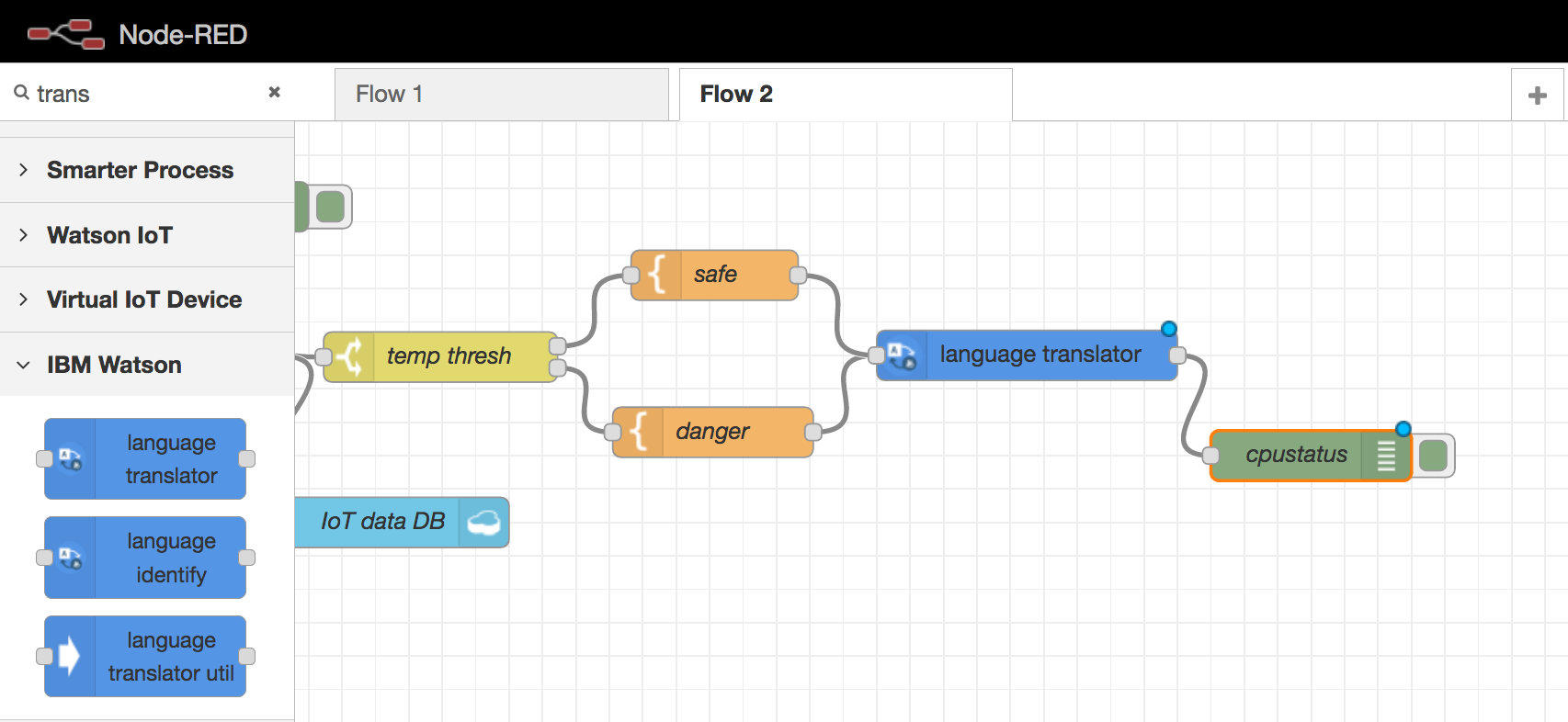
1. Connect service to your Node-RED application



1. Accept the restage step to actually bind the service to the app.
2. While it is restaging (~3 minutes), take a look at **Service Credentials**. This information is useful if you want to invoke your Watson Service from any program (running in IBM Cloud or outside IBM Cloud)
3. Go back to Node-RED window.  
   If you get a connection error message, your application restaging is not finished. Wait. Try to refresh page.



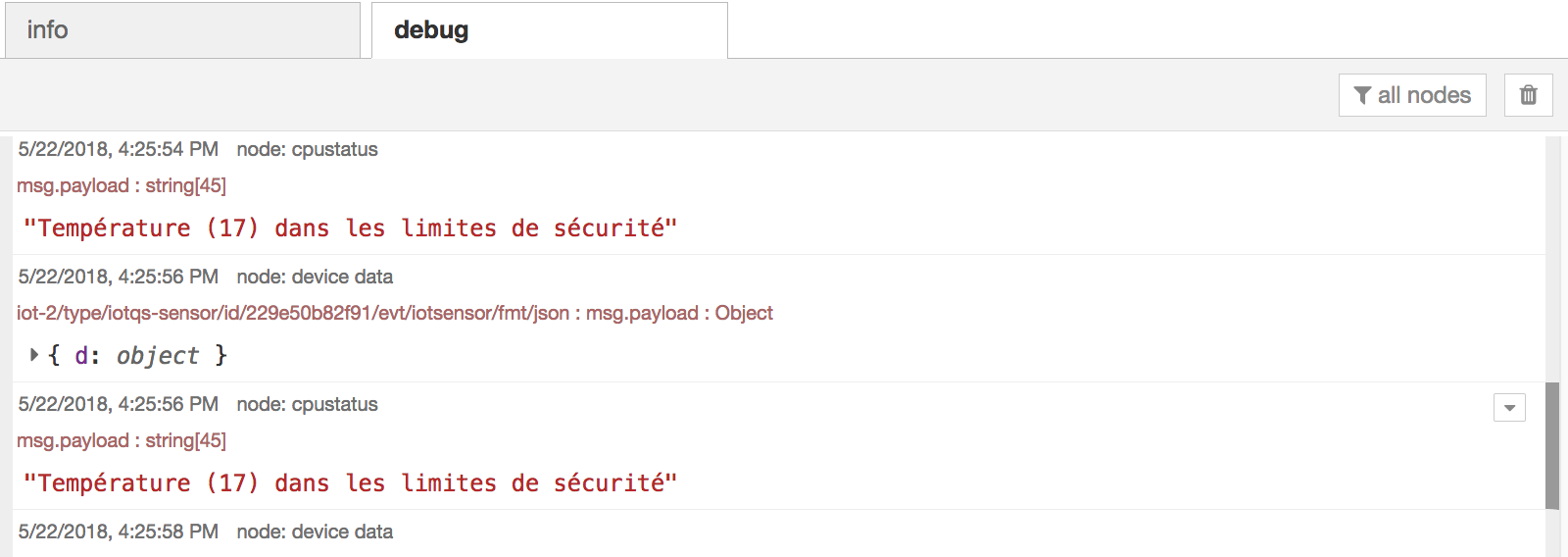
1. Go back to the Node-RED environment.  
   Add **Language translator** node and link it between the template (*safe & danger) &* debug nodes (*cpustatus*)*.*



1. Configure the Watson language translator node:
   * Name (of your choice)
   * Mode: **Translate**
   * Domains: **Conversational**
   * Source: **English**
   * Target: **French** (or Spanish, Portuguese & Arabic)

Note: The user/password fields are not necessary & do not appear in the node settings if a Watson Language Translator service is properly bound to Node-RED application.

1. **Deploy** your flow & check the logs (**debug** tab).

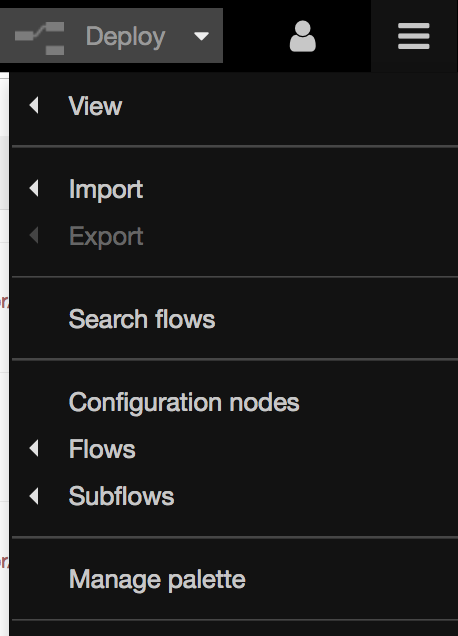


* 1. Create a dashboard application in Node-RED (optional #4)
     1. Import Node-RED Dashboarding capability

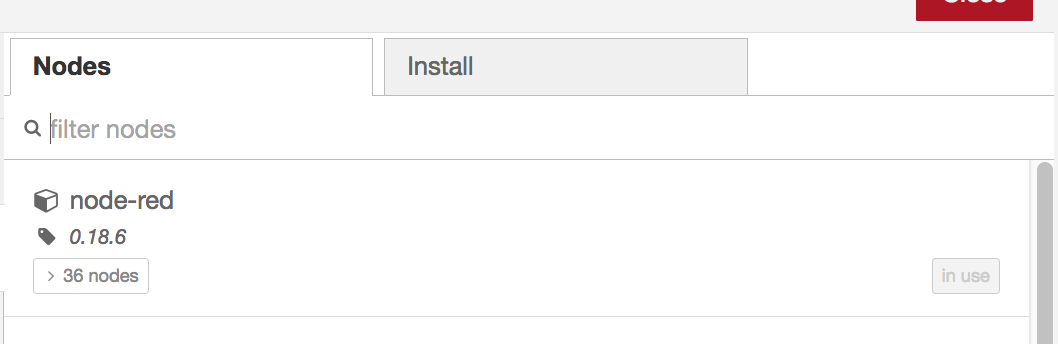
1. At the top right-hand side of the page, click the ‘burger’ menu:



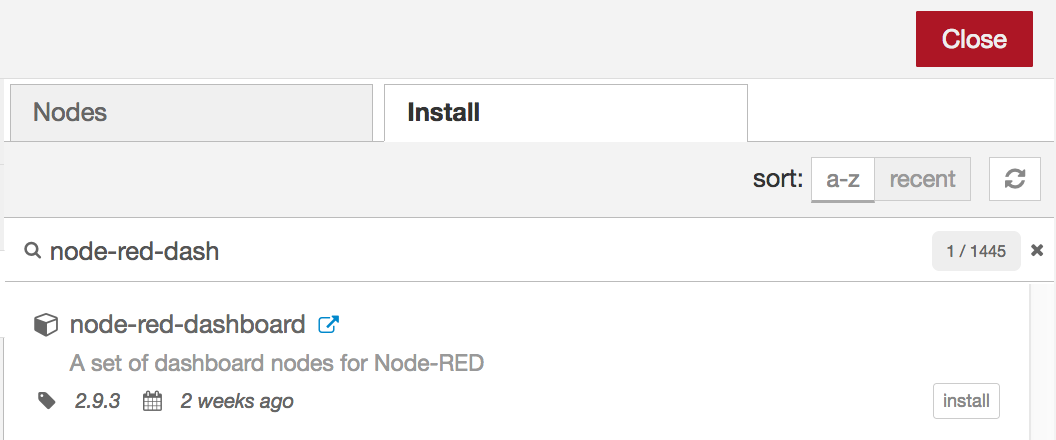
1. Click **Manage palette**:



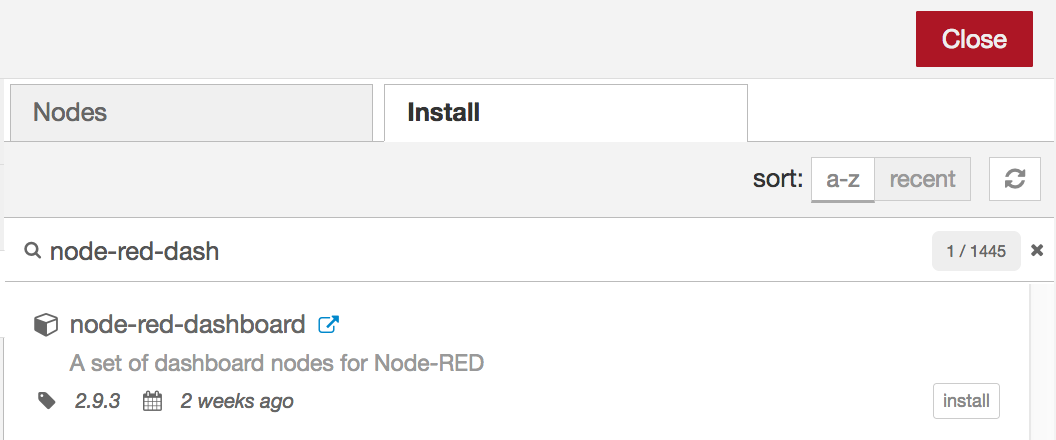
1. In the sidebar that appears on the left-hand side of the page, click the Install tab:



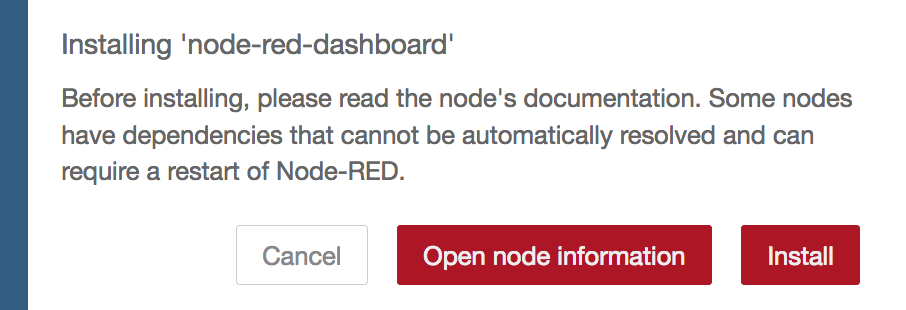
1. In the search field, type **node-red-dash**:



1. Next to the nod-red-dashboard result, click **install**



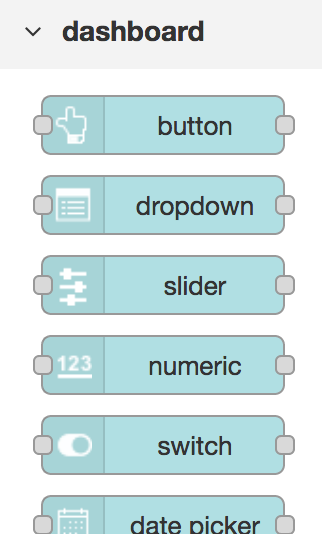
1. Click **Install**



1. Wait for the new nodes to be deployed.   
   A popup shows which nodes have been installed.



1. Click **Close**.
2. Note the additional dashboard nodes on the palette : **Dashboard** category.



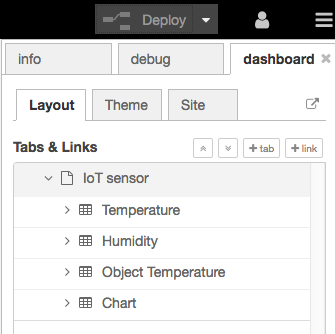
1. Note also that there is a new dashboard tab in the right-hand sidebar:



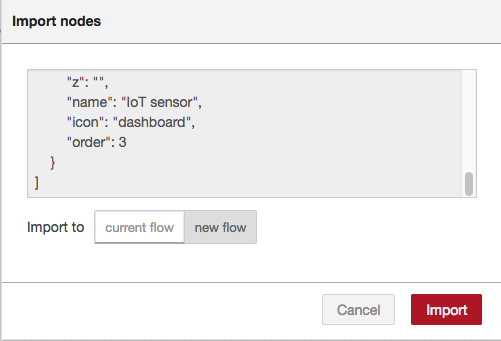
*TIP: This dashboard tab may be used to add new tabs, menus etc. to the visualization dashboard. There are also two available themes – light and dark.*

* + 1. Create a simple Node-RED Dashboard

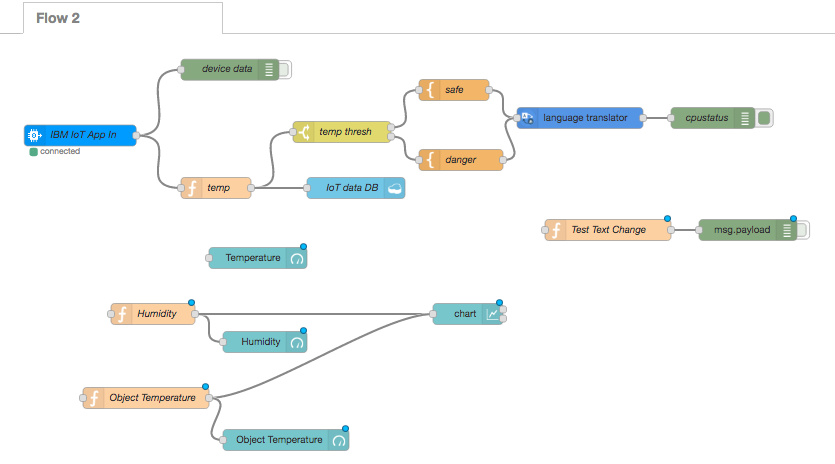
In this section, you will create a simple dashboard for sensor data using new dashboard nodes installed.



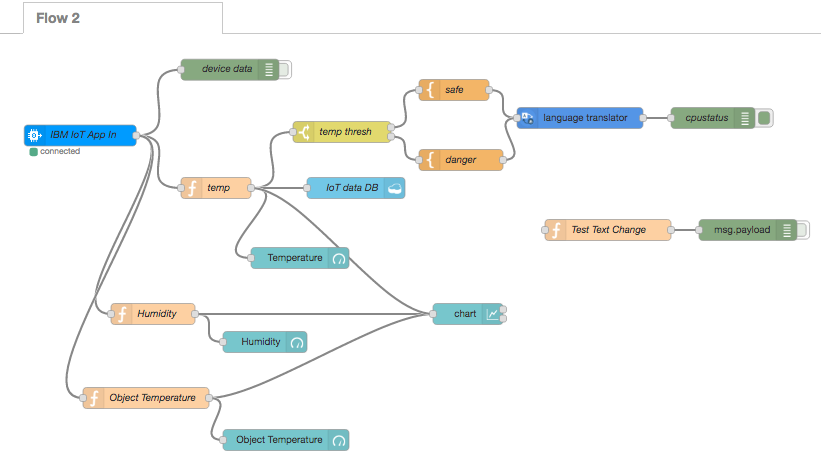
1. In the current Node-RED tab, import the file named **Lab3\_IoT\_Dashboard.v2.0.txt** (previously downloaded): **Menu** > **Import** > **Clipboard**.   
   Click **current flow** then **Import**.



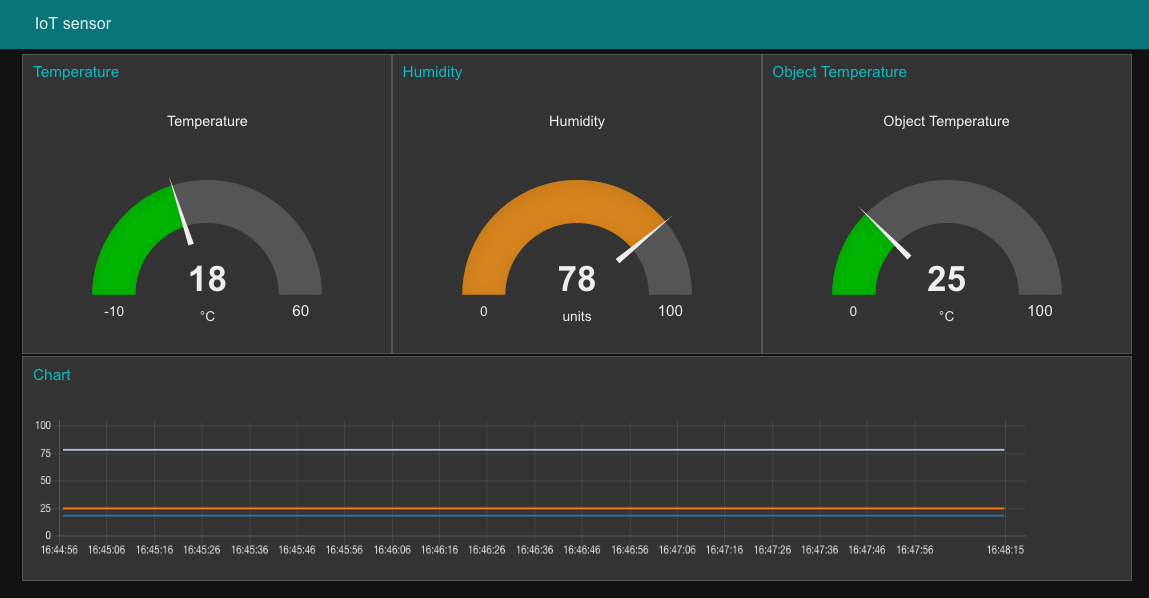
1. Click on workspace to paste new nodes.



1. Connect new nodes to existing…
   1. **Temperature** to **temp**
   2. **Humidity** to **IBM Iot App In**
   3. **Object Temperature** to **IBM Iot App In**
   4. **chart** to **temp**

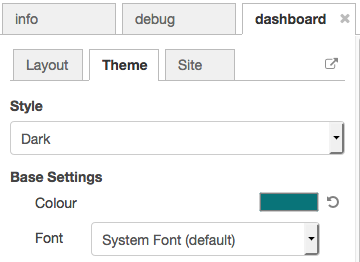


1. Deploy the flow: **Deploy**
2. Connect to **http://<YOUR\_APP\_HOSTNAME>/ui** (URL generated by Node-red-dashboard node) to see your new dashboard. Change value on virtual sensor app. to see impact on gauges and lines.

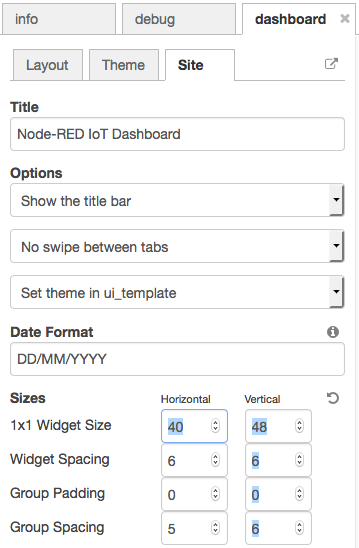


1. Customize your dashboard using **Dashboard** tab

* **Theme**: use Dark style



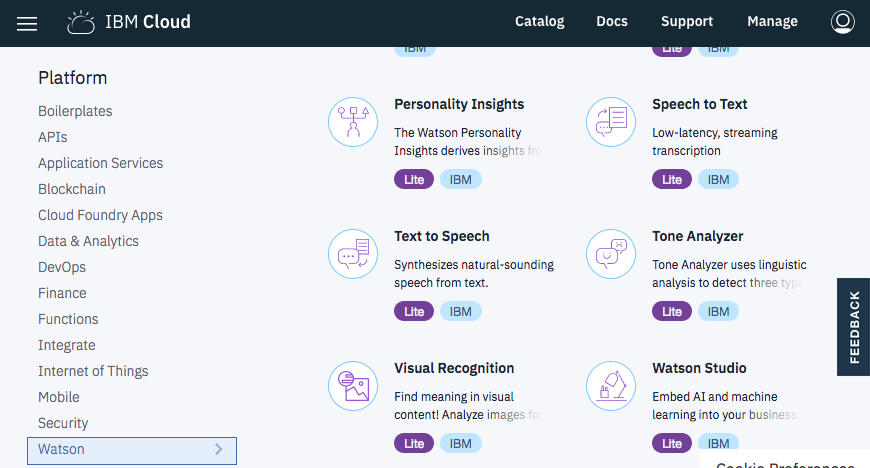
* **Site**: as below, change **Title**, **1x1 Widget Size**, **Group spacing**



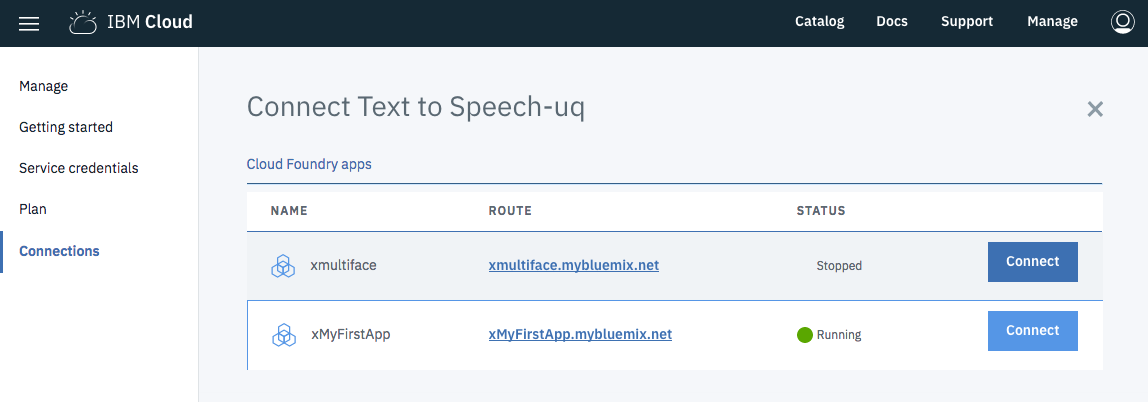
1. Deploy and check your dashboard.
   * 1. Add voice alert on dashboard

In this section, you will add a voice node allowing your app to tell say message when temperature change. To do that, you will deploy a new Watson service: Text to Speech.

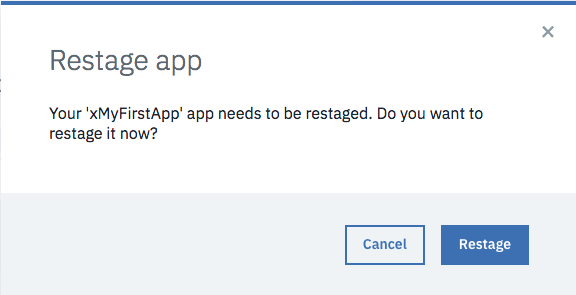
1. In the IBM Cloud window, click **Catalog** then **Watson** category.  
   Click on **Text to Speech** service to create your own instance.



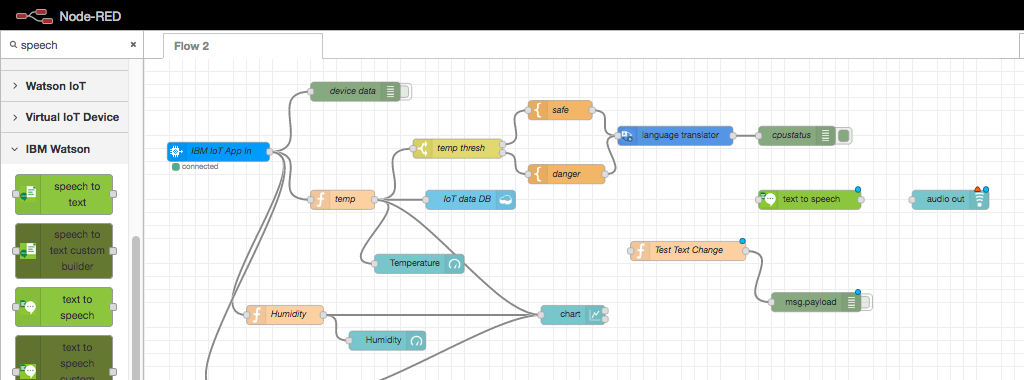
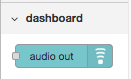
1. Enter a name for your service and click **Create**.
2. On new service dashboard, click **Connection**, and connect Text to speech service to your Node-RED application: **Connect**.



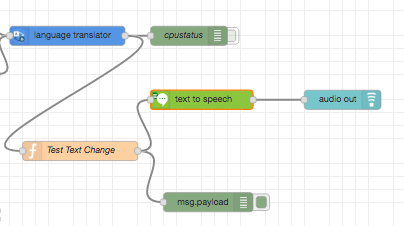
1. Accept to restage application and wait for your Node-RED application to restart.



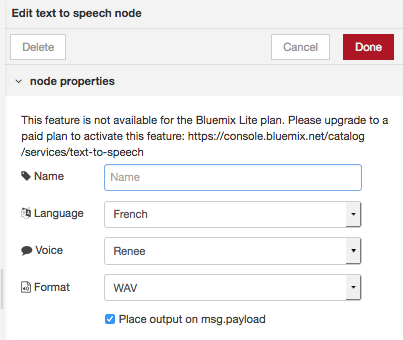
1. When restarted, in Node-RED environment, add an **Audio out** node (category **Dashboard**) and a **text to speech** node (category **Watson**) to your flow



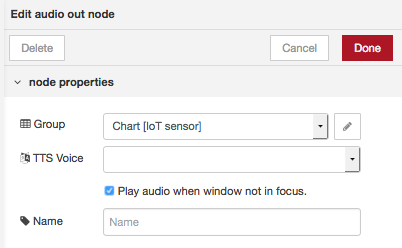
1. Connect nodes as below
   1. **Test text change** to **language translator**
   2. **Test text change** to **text to speech**
   3. **text to speech** to **audio out**



1. Configure **text to speech** node to
   * use the language your translated to (as configured before in **language translator** node)
   * place the output on msg.payload (check box)



1. Configure **audio out** node
   * Set **group** to **chart [IoT sensor]**
   * Check **Play audio when window not in focus**



1. Deploy the flow: **Deploy**
2. Do you hear something ?   
   Try to change temperature in virtual sensor app.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ END OF LAB \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_