Installation and Activation of the

TI I3 IOT Application

For the Sensor Data Storage

And Live Processing

By

Feng Mo

September 25, 2017

Contents

[Overview 3](#_Toc481738294)

[Principle of operation 4](#_Toc481738295)

[Install the Bluemix I3 IOT application 5](#_Toc481738296)

* [Create an IBM account to access the IBM Bluemix service 5](#_Toc481738297)
* [Create a database 7](#_Toc481738298)
* [Create an IOT template application to host the IOT I3 application later 9](#_Toc481738299)
* [Create an IBM Watson MQTT device to receive the live sensor data and publish to the Bluemix MQTT broker 11](#_Toc481738300)
* [Import the I3 IOT application into the IOT template app created 20](#_Toc481738301)
* [Configure the database nodes to access the external Cloudant database 39](#_Toc481738302)
* [Configure the MQTT output node to send the live sensor data to the Watson MQTT broker 42](#_Toc481738303)
* [Modify the web app on the I3 Gateway to send data to the Bluemix I3 app 48](#_Toc481738304)
* [Display live sensor data on the IBM Watson Cards 52](#_Toc481738305)

# Overview

The TI I3 IOT application is a web server developed and run in the IBM Bluemix platform. It enables the TI I3 gateways to connect to it and send the sensor data. The sensor data is stored in an Cloudant database and can be retrieved later for processing. The data can also be displayed live on the IBM Watson IOT platform. This document describes the procedure to install the I3 IOT application on the IBM Bluemix platform, configure the I3 gateways to send sensor data to the I3 IOT application, and to display the live sensor data in the IBM Watson IOT platform.

To install and run the I3 IOT application on the IBM Bluemix platform, you will need to have an IBM Bluemix account. You can obtain a trial account online. You will need to access the Node-RED visual programming tool to create a template application which will be transformed later into the I3 IOT application through source code import. You will need a database service to create a Cloudant database for storing the sensor data. You will need a Watson IOT service to display the live sensor data and analysis. This document describes the steps necessary to accomplish each of these tasks.

The purpose of the I3 IOT App is to serve as a framework for developing more specialized and advanced data analysis and presentation applications.

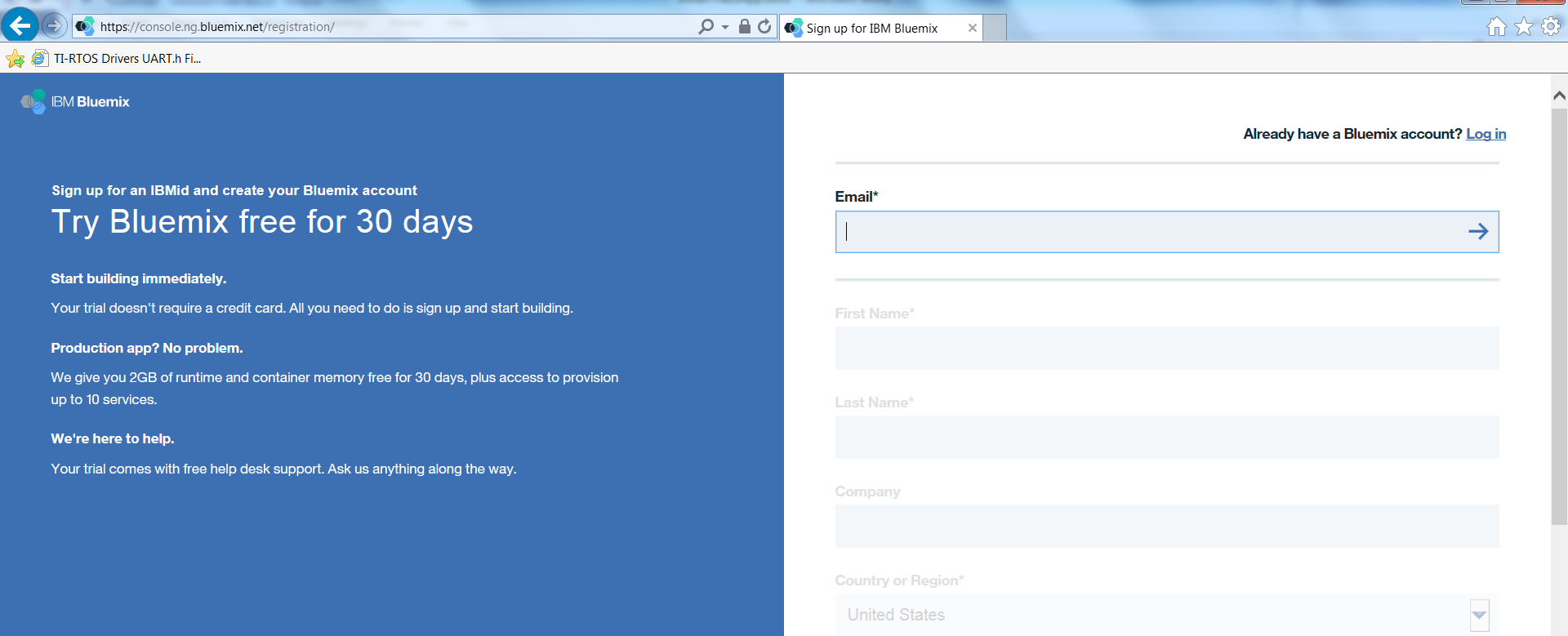
# Principle of operation

The sensor data is sent from the TI I3 gateways to the I3 IOT App using the HTTP/POST method. The I3 IOT App stores the received sensor data in a database. It also publishes the live sensor data to an MQTT broker running in the Watson IOT service of the Bluemix platform. The MQTT broker sends the data to the visualization panels called Cards created in the Watson IOT service.

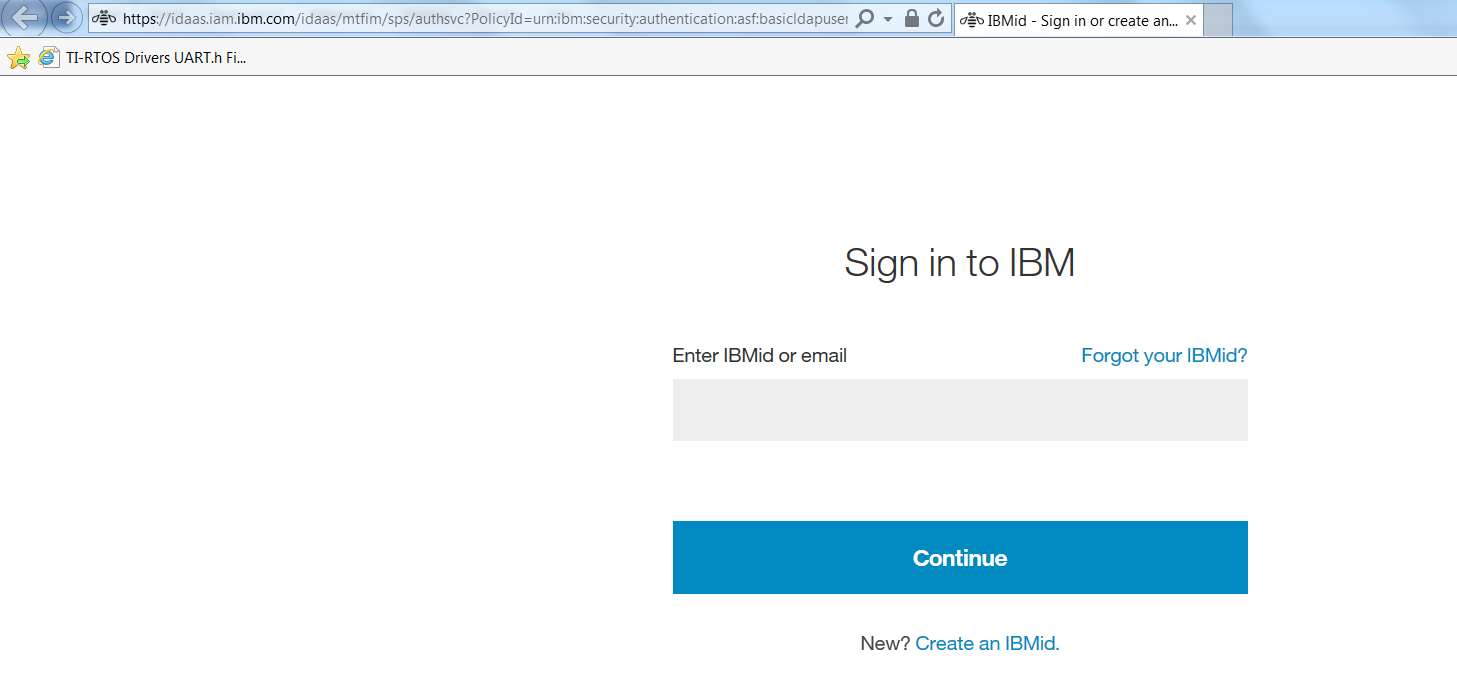
# Install the Bluemix I3 IOT application

## Create an IBM account to access the IBM Bluemix service

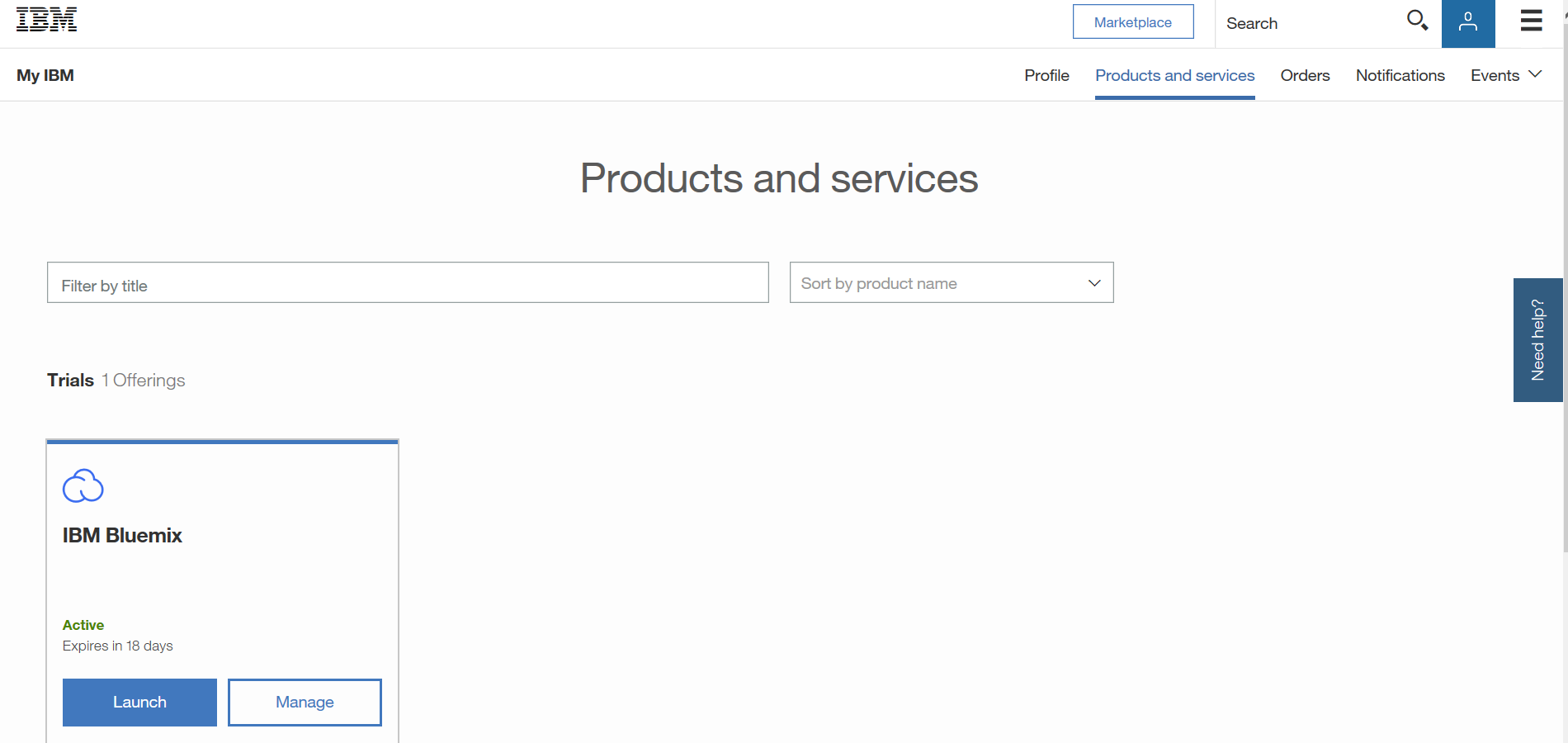
1. Register a 30-day trial IBM Bluemix account:

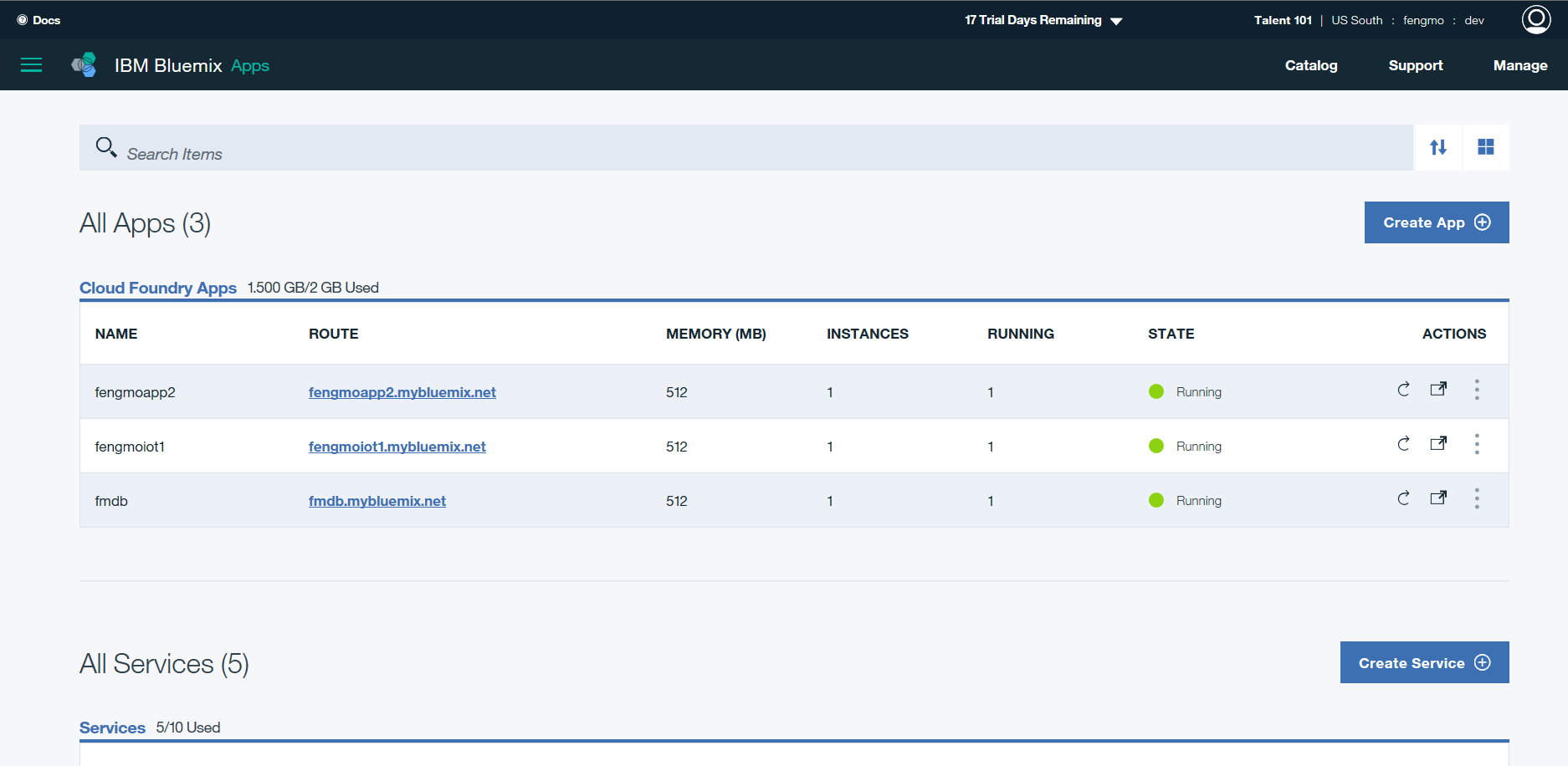


1. Log into the IBM account:



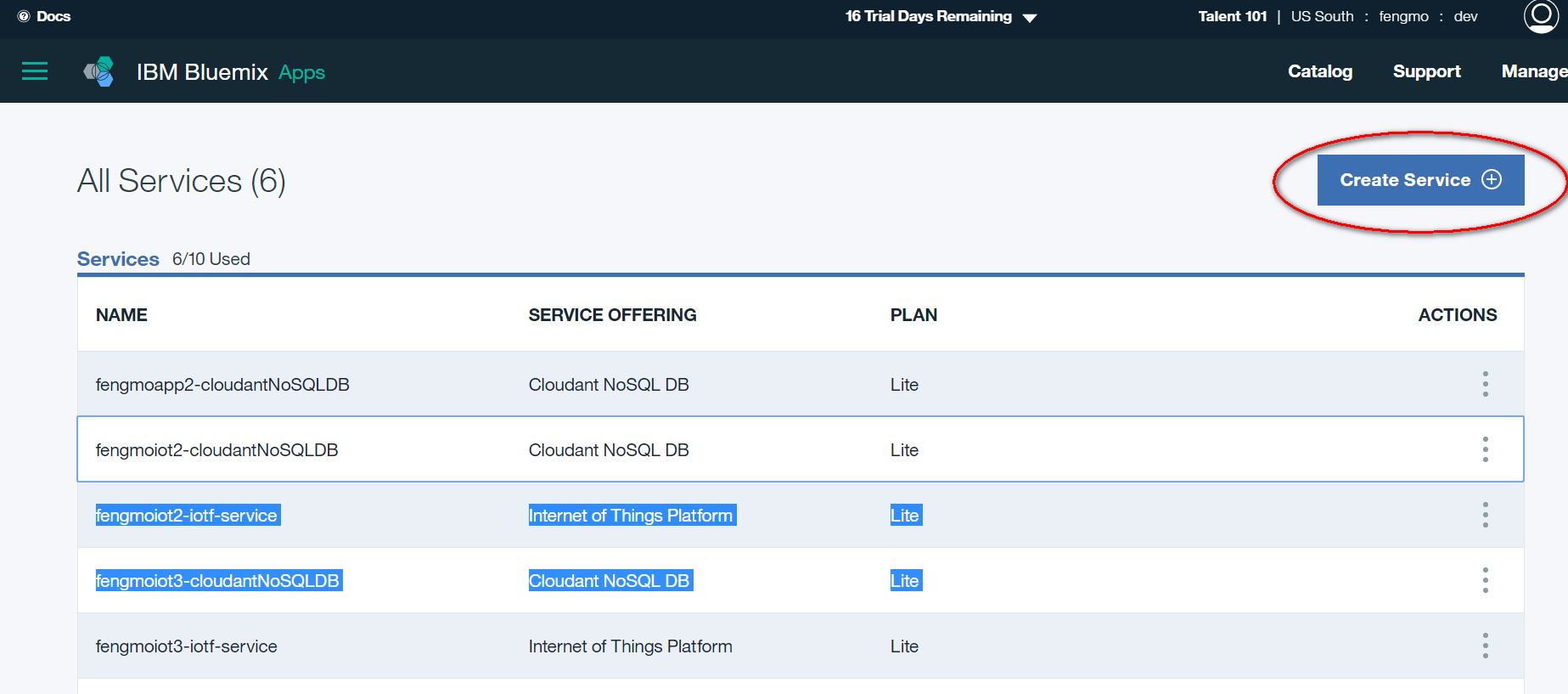
1. Launch the Bluemix



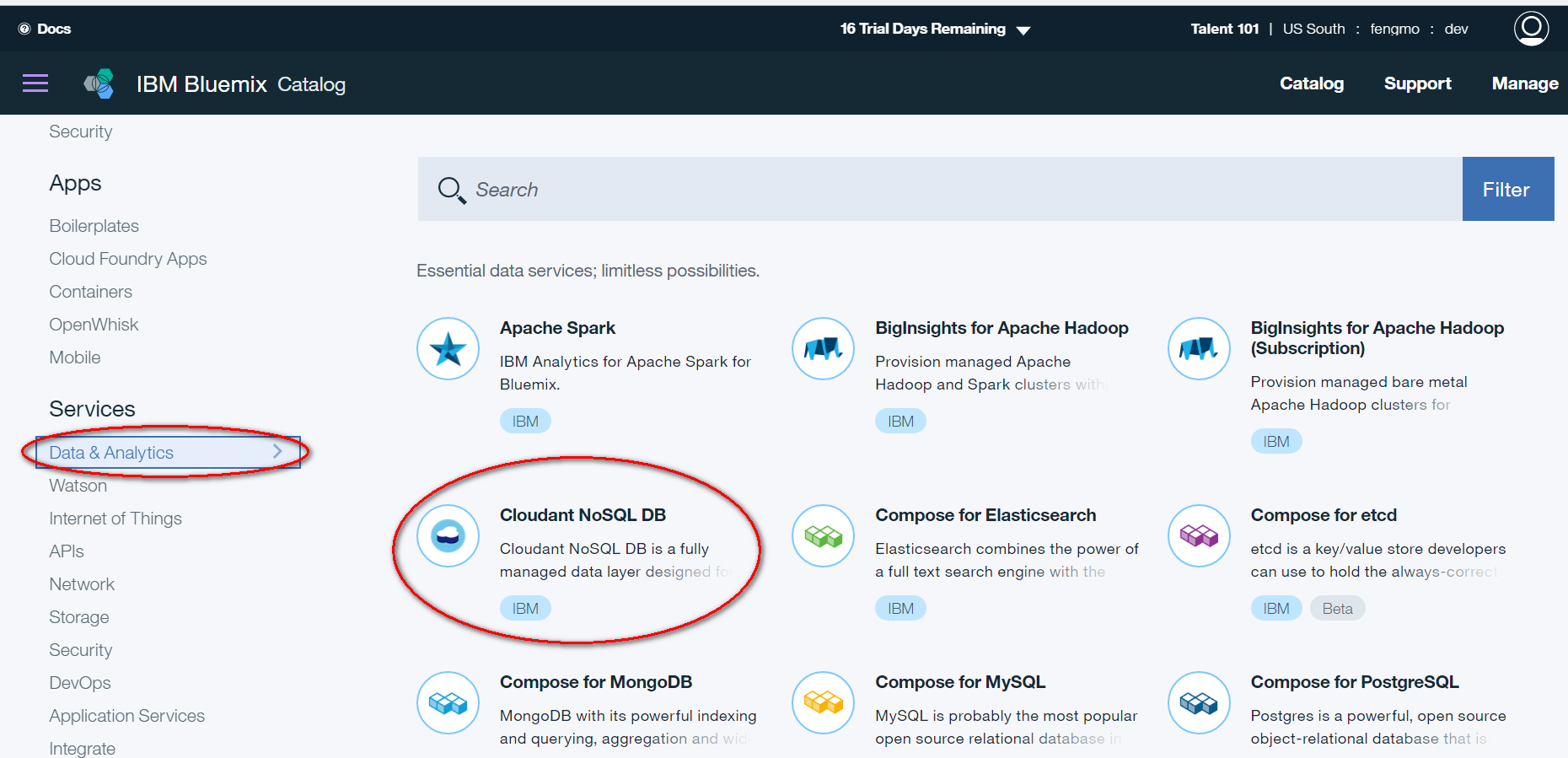


## Create a database

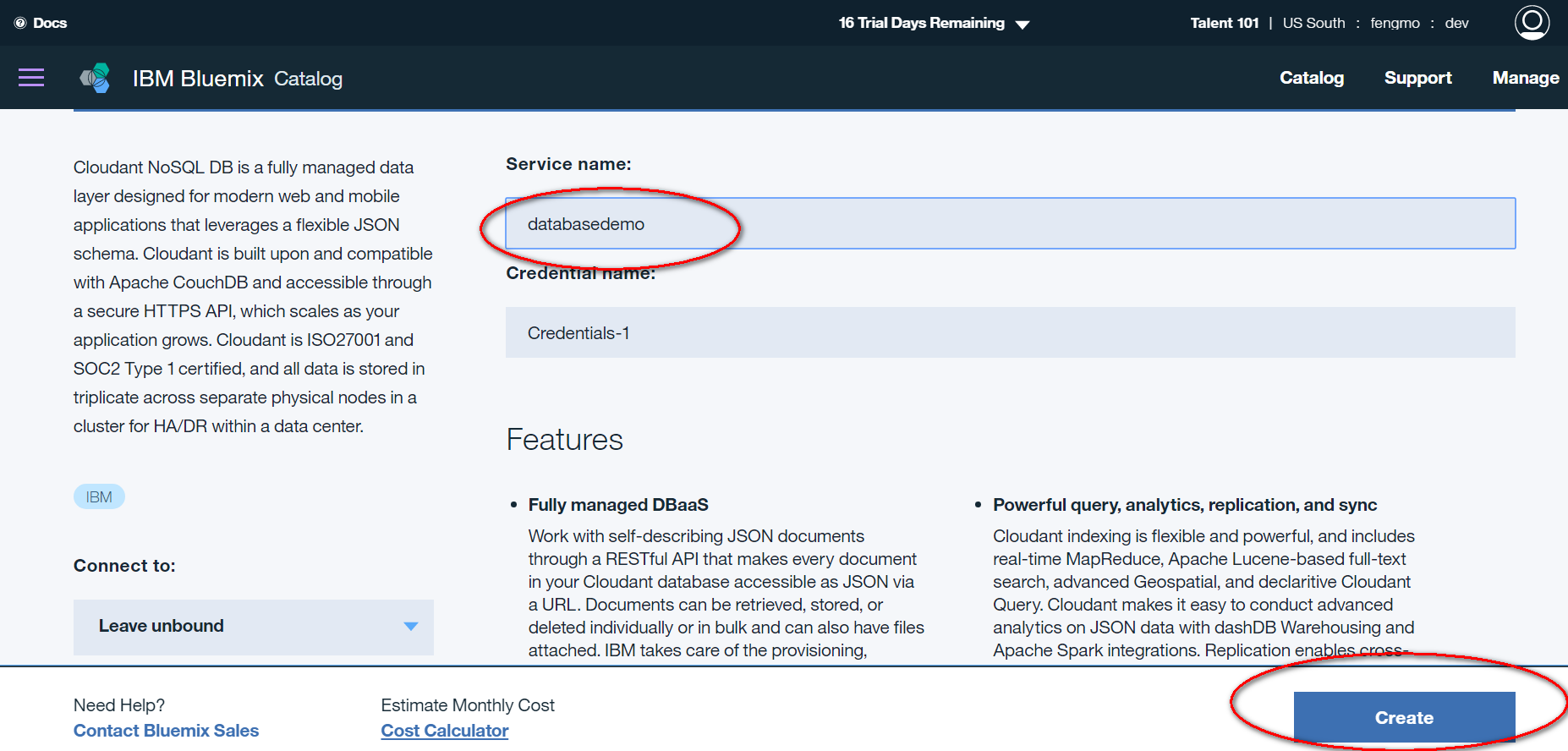
1. Click the “Create Services” button



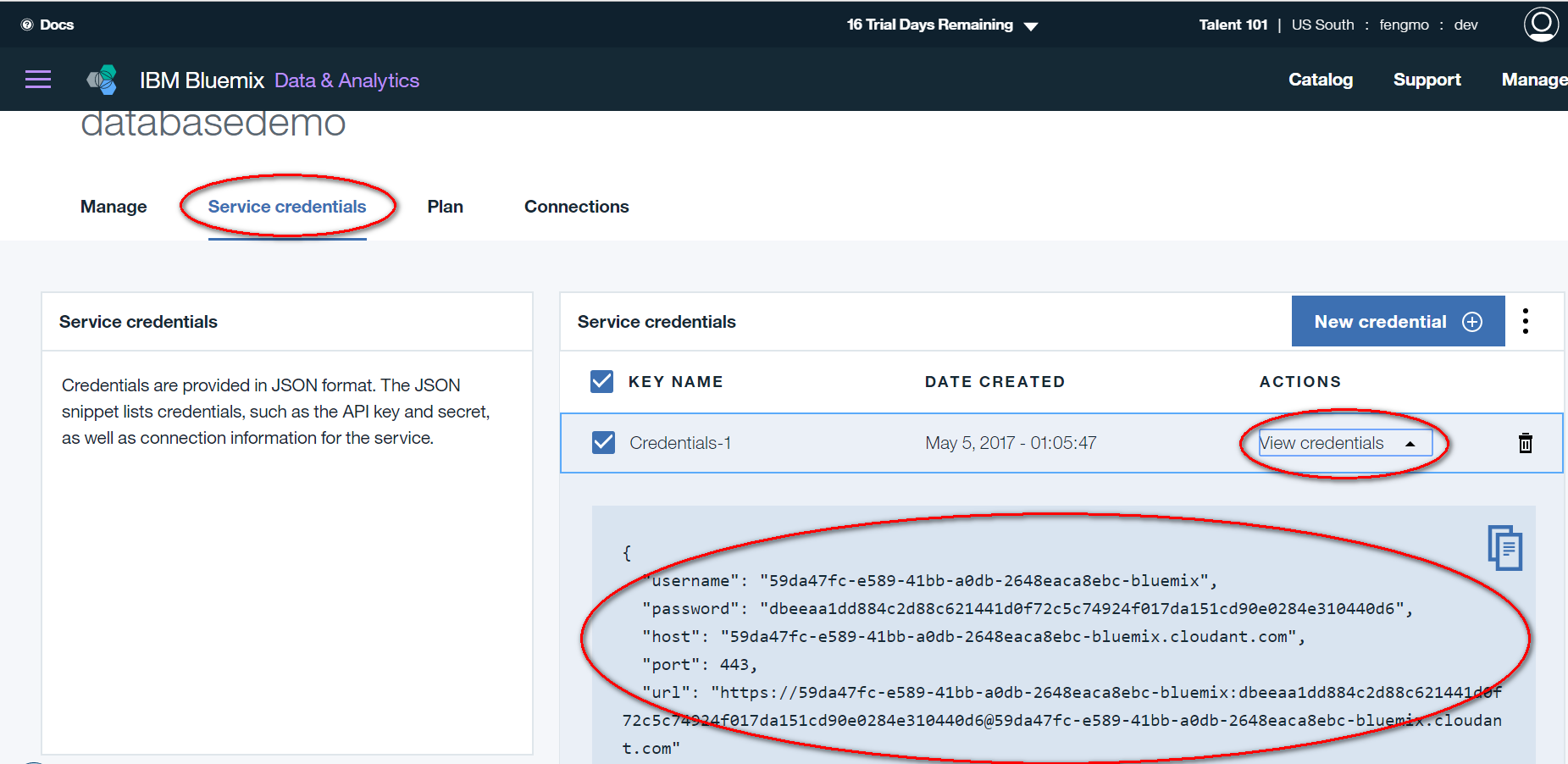
1. Click the “Data & Analysis” service and the “Cloudant NoSQL DB”



1. Enter a name for the database and click the “Create” button, the name used in the application is databasedemo.

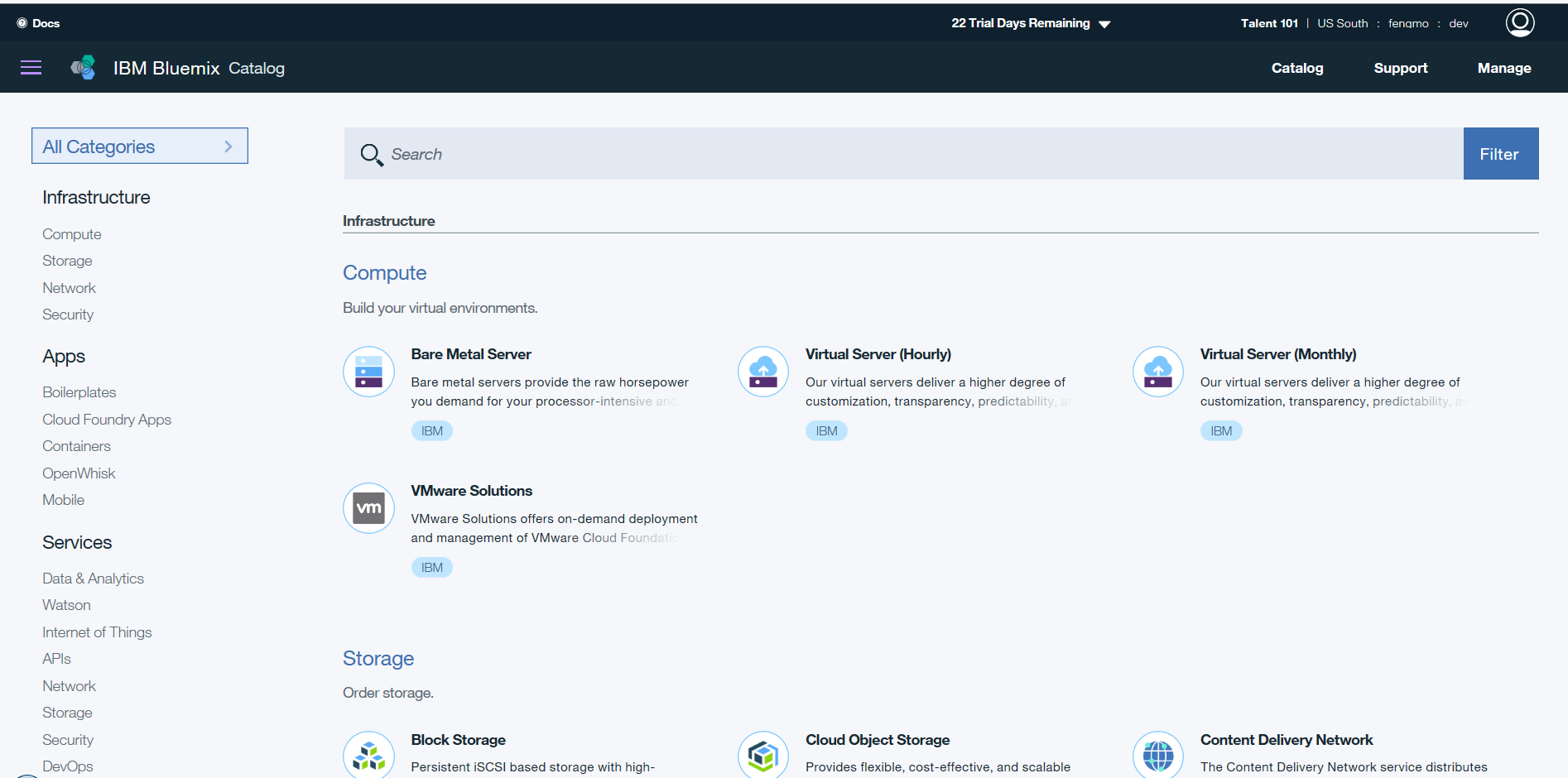


1. Note down the credential values for the database for later use in the application:



## Create an IOT template application to host the IOT I3 application later

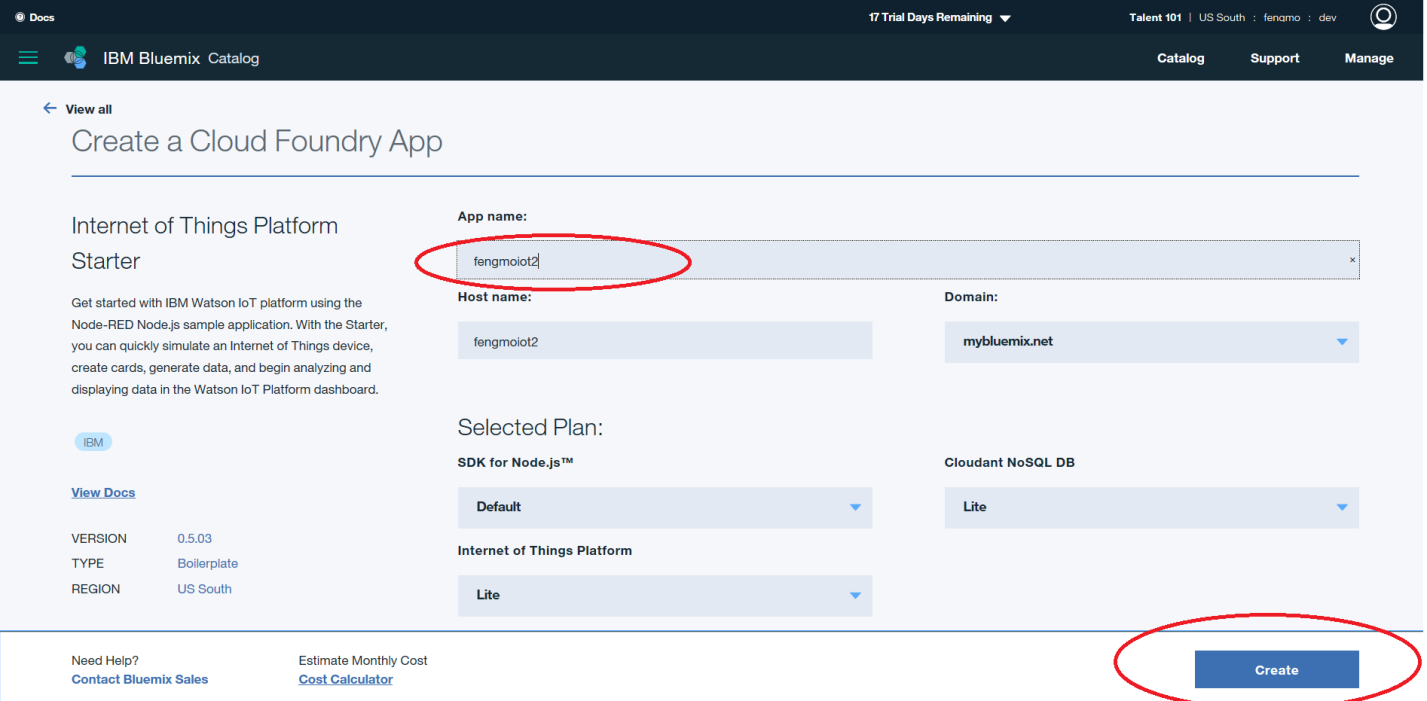
1. Go to the “IBM Bluemix Catalog” page:



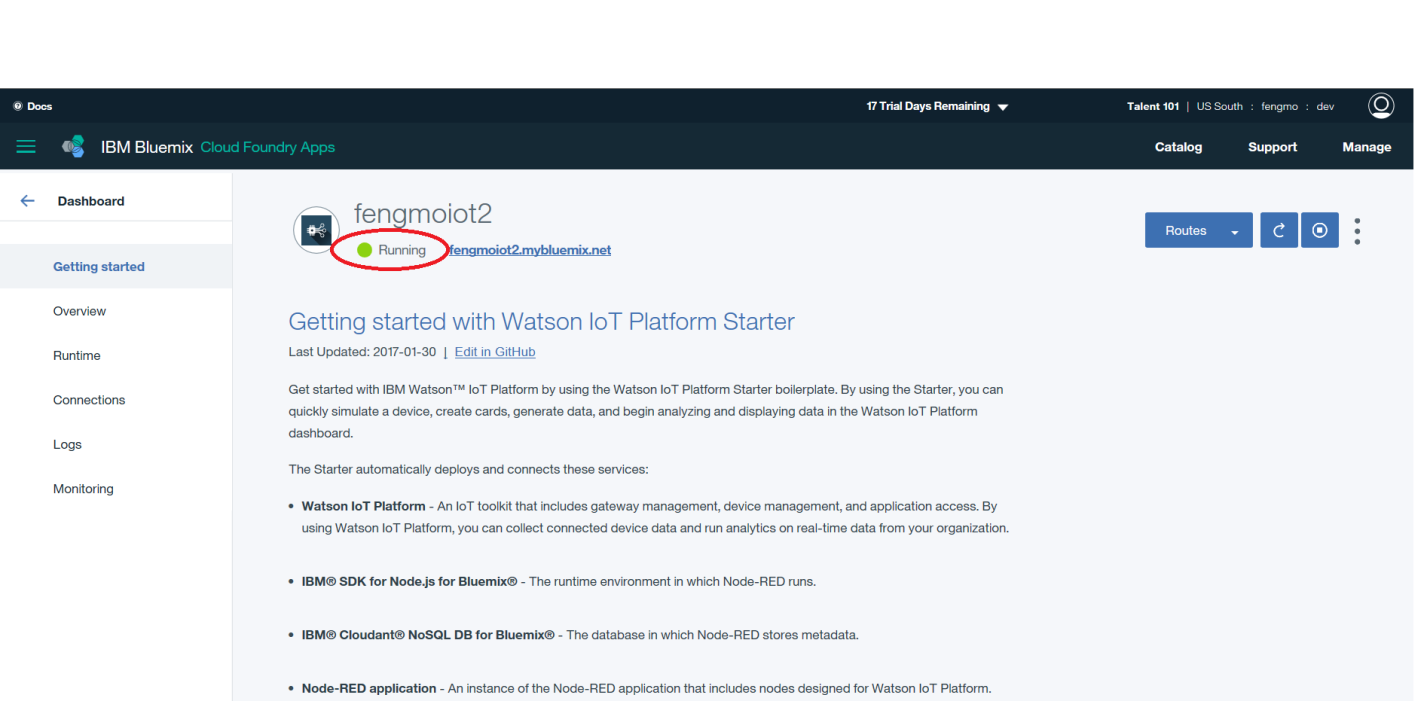
1. Select the “Boilerplates” in the “Apps” list on the left side and select the Internet of Things Platform Starter:



1. Enter a name (the example is fengmoiot2, you should give a different name to avoid conflict) for the application and click the “Create” button:

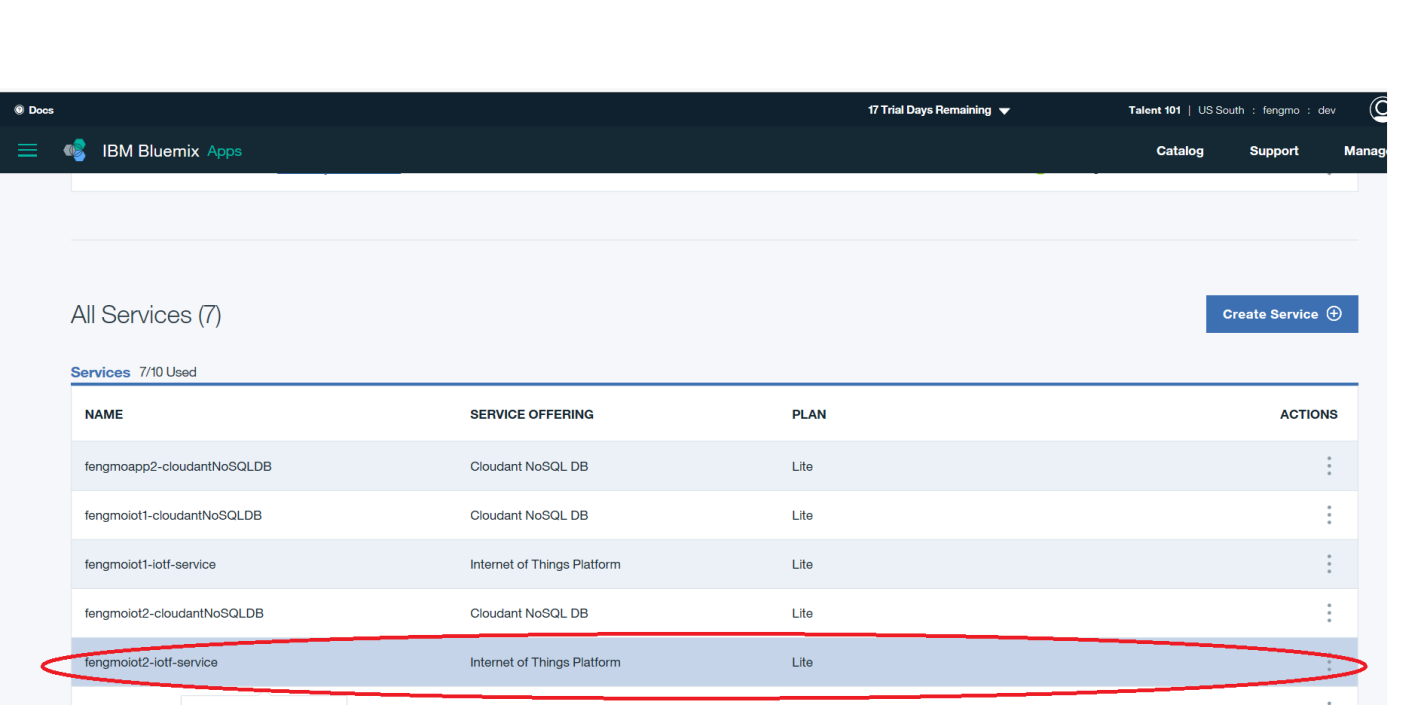


1. Wait for the app to start (may take a long time):



## Create an IBM Watson MQTT device to receive the live sensor data and publish to the Bluemix MQTT broker

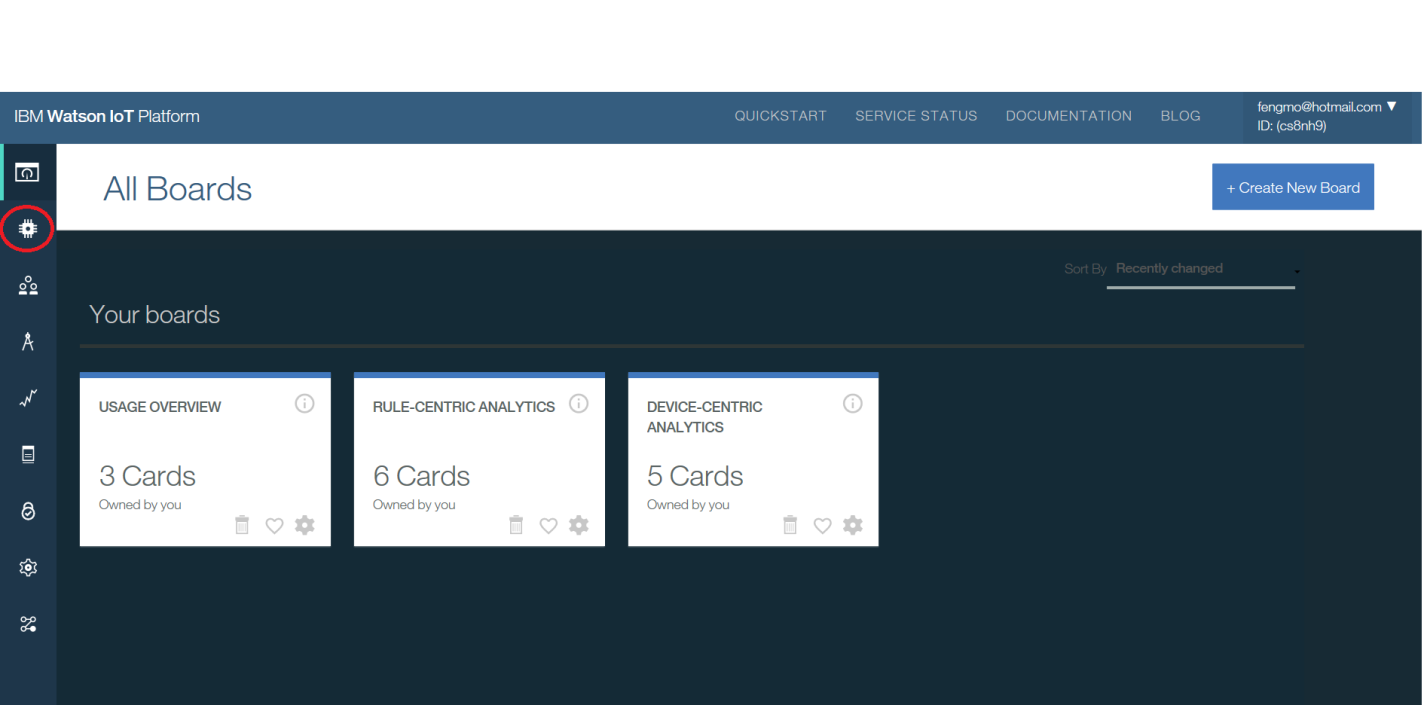
1. Go to the “Dashboard” and select the IOT service created as part of the new app:



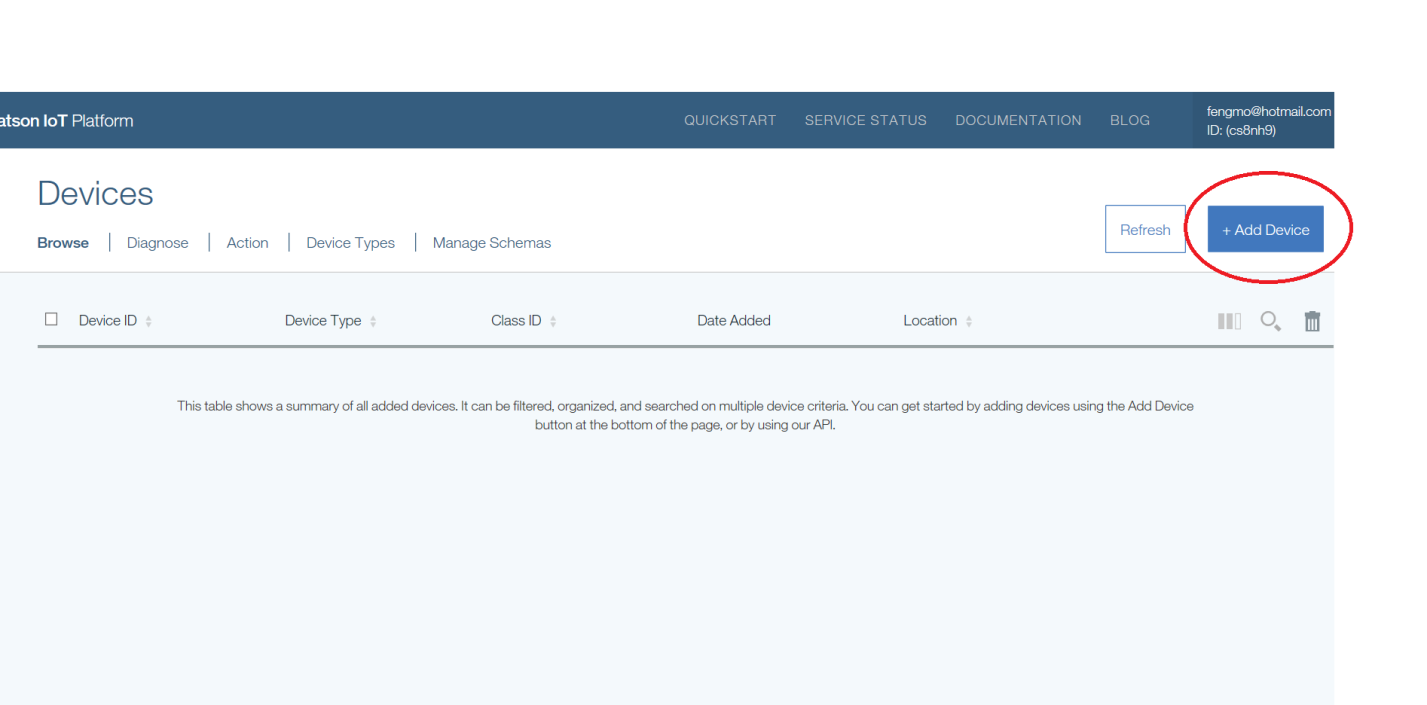
1. After a pause (can be long), you should see this screen. Click the “Launch” button



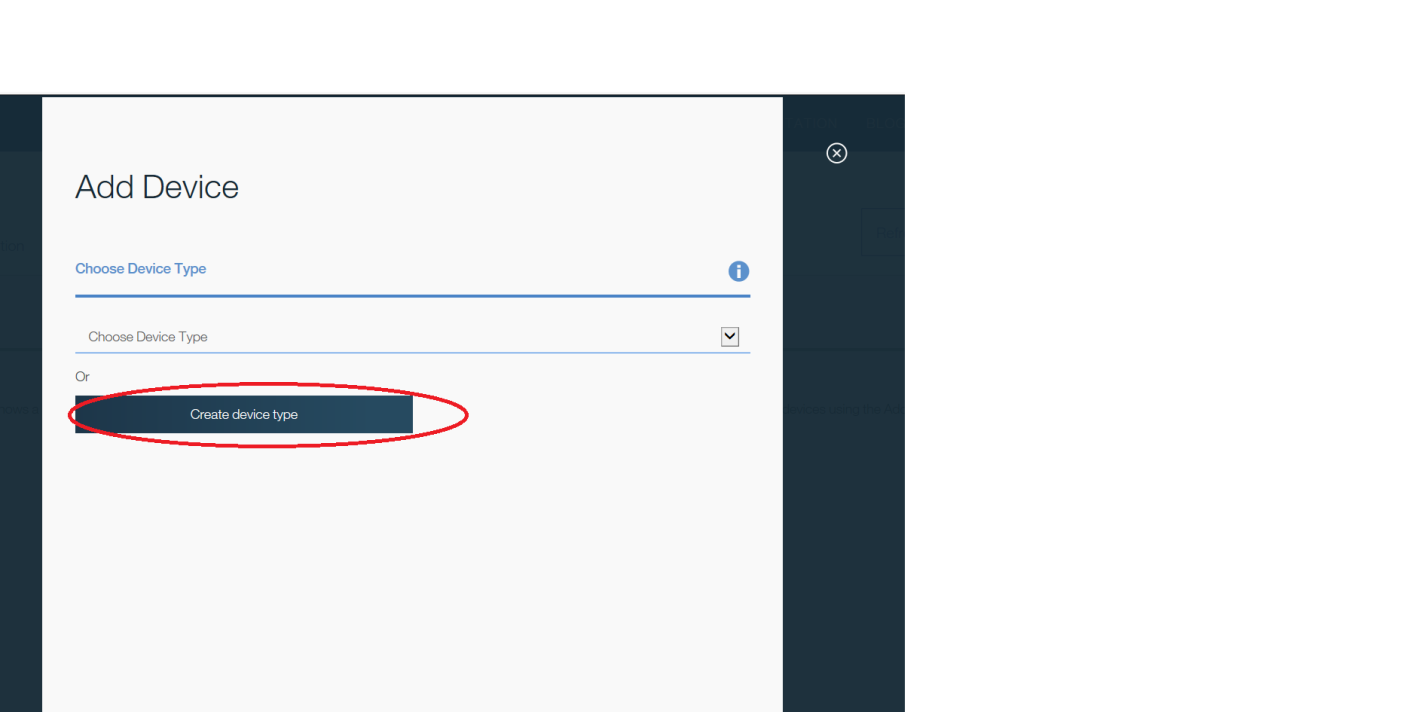
1. You should see this screen. Click the device button on the left side. If you don’t see this screen, instead, see a screen to log into the Watson platform, log in first. Then go back to step above to launch again.



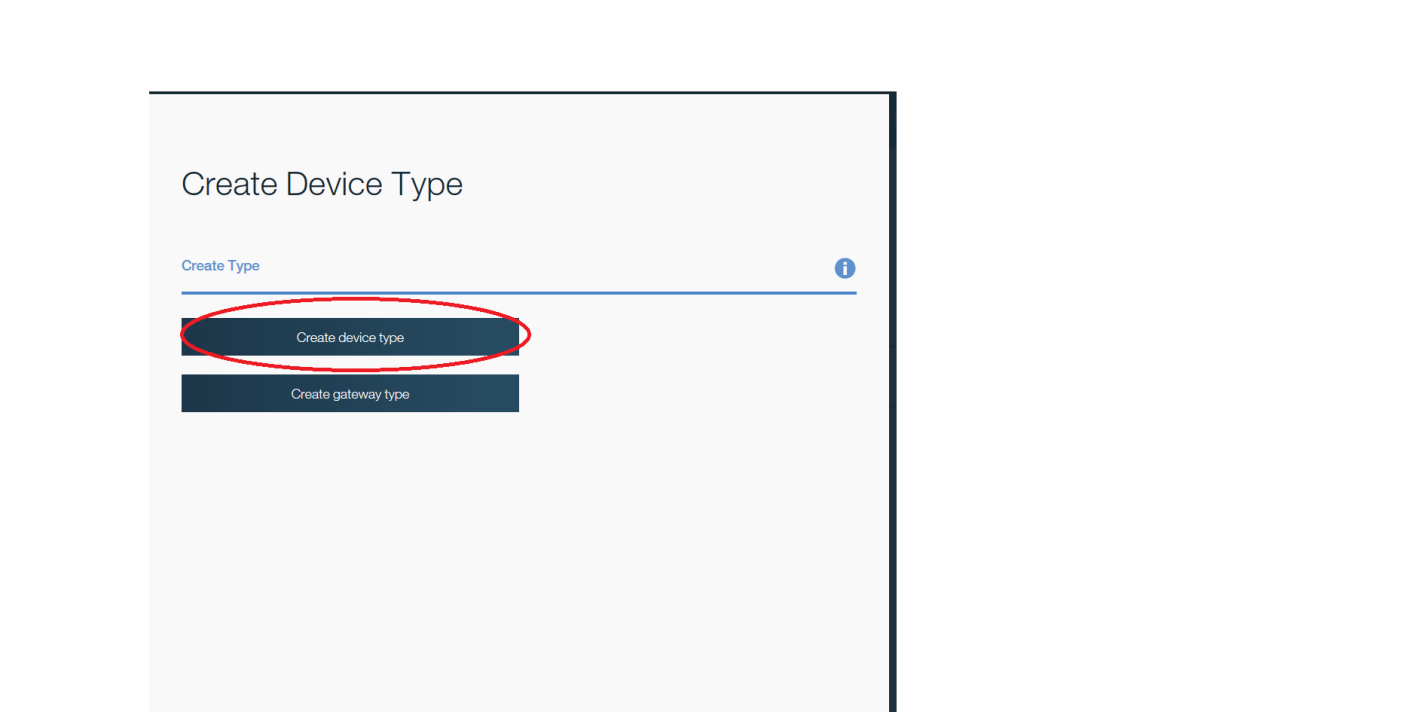
1. You should see this screen. Click the “+Add Device” button.



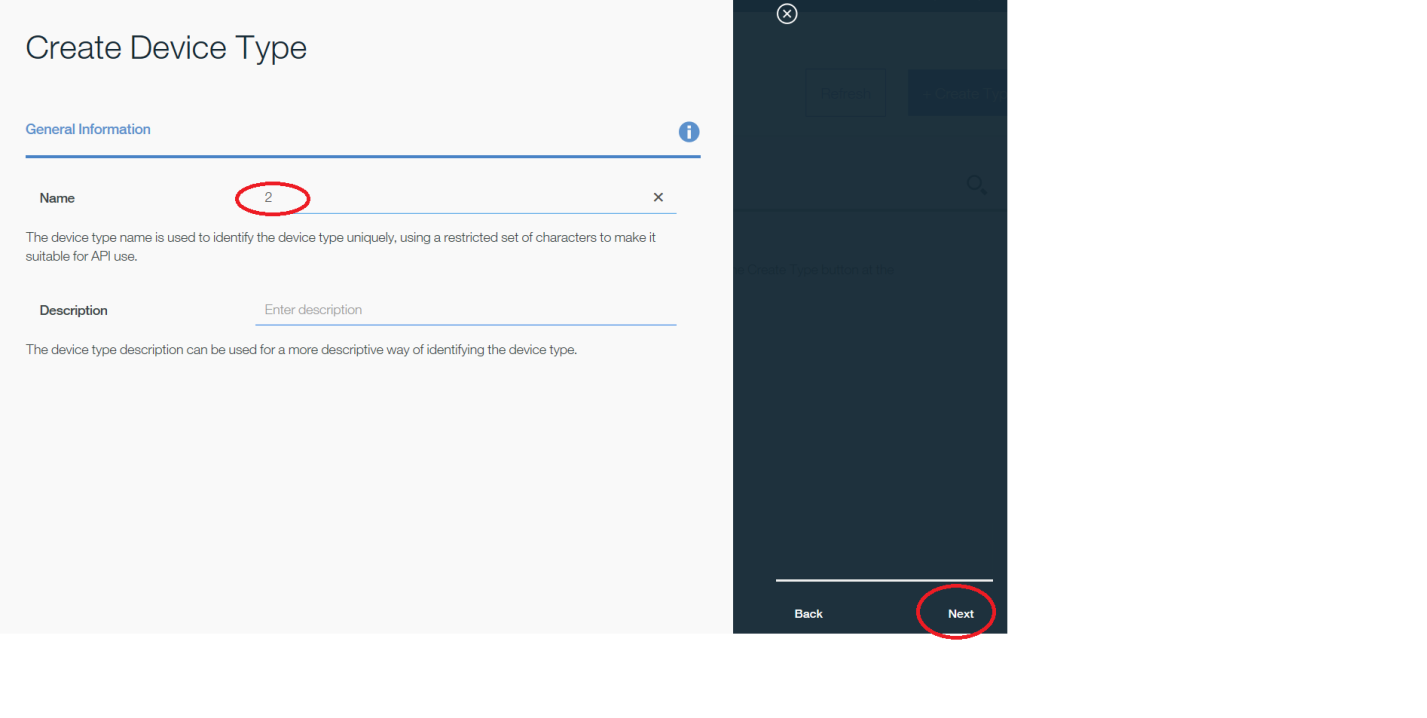
1. Click the “Create device type” button



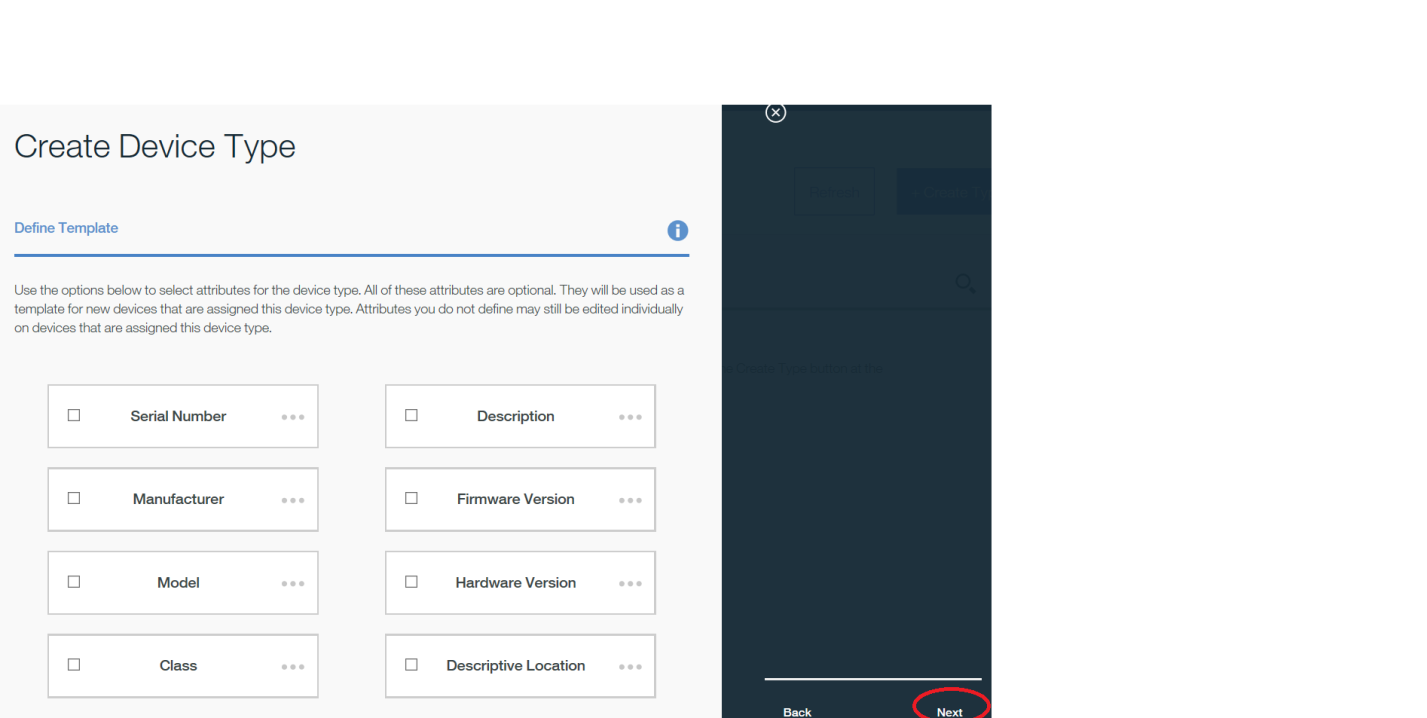
1. Click “Create device type” button



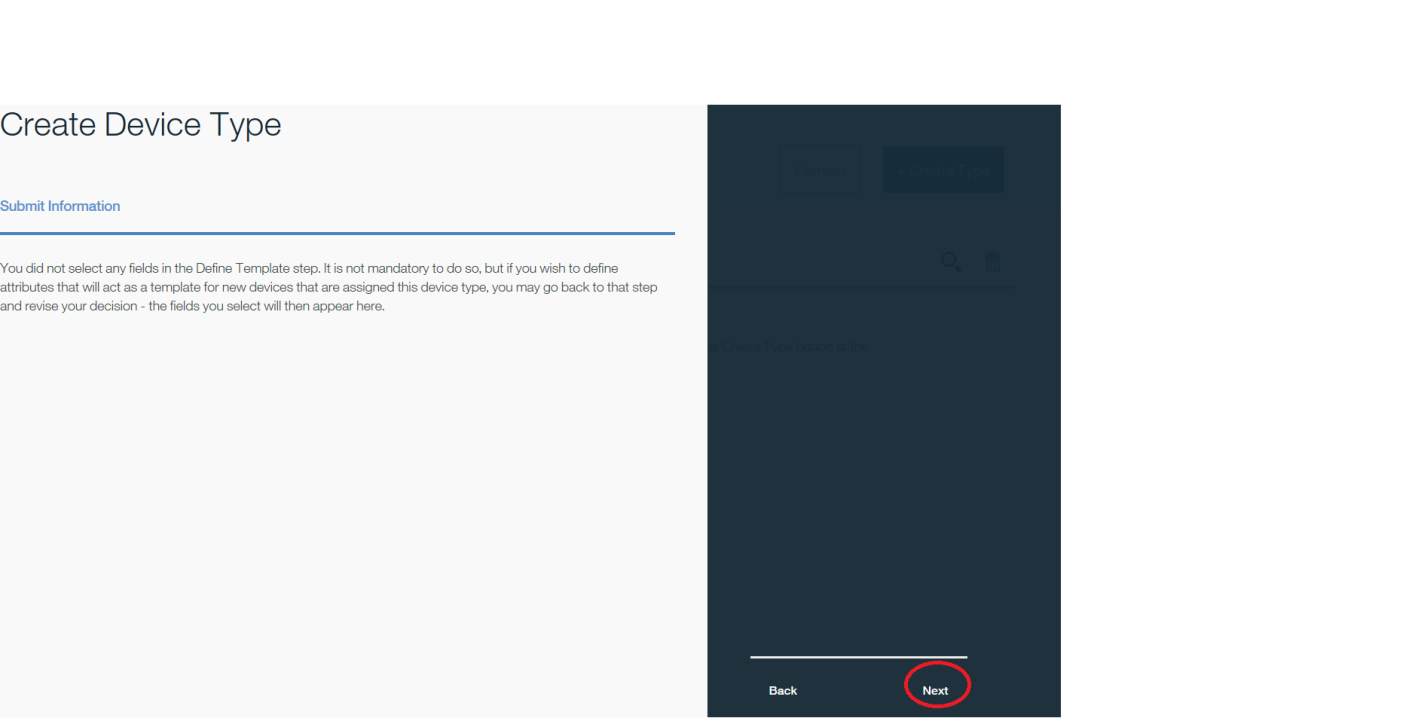
1. Enter 2 as the device name and click the “Next” button



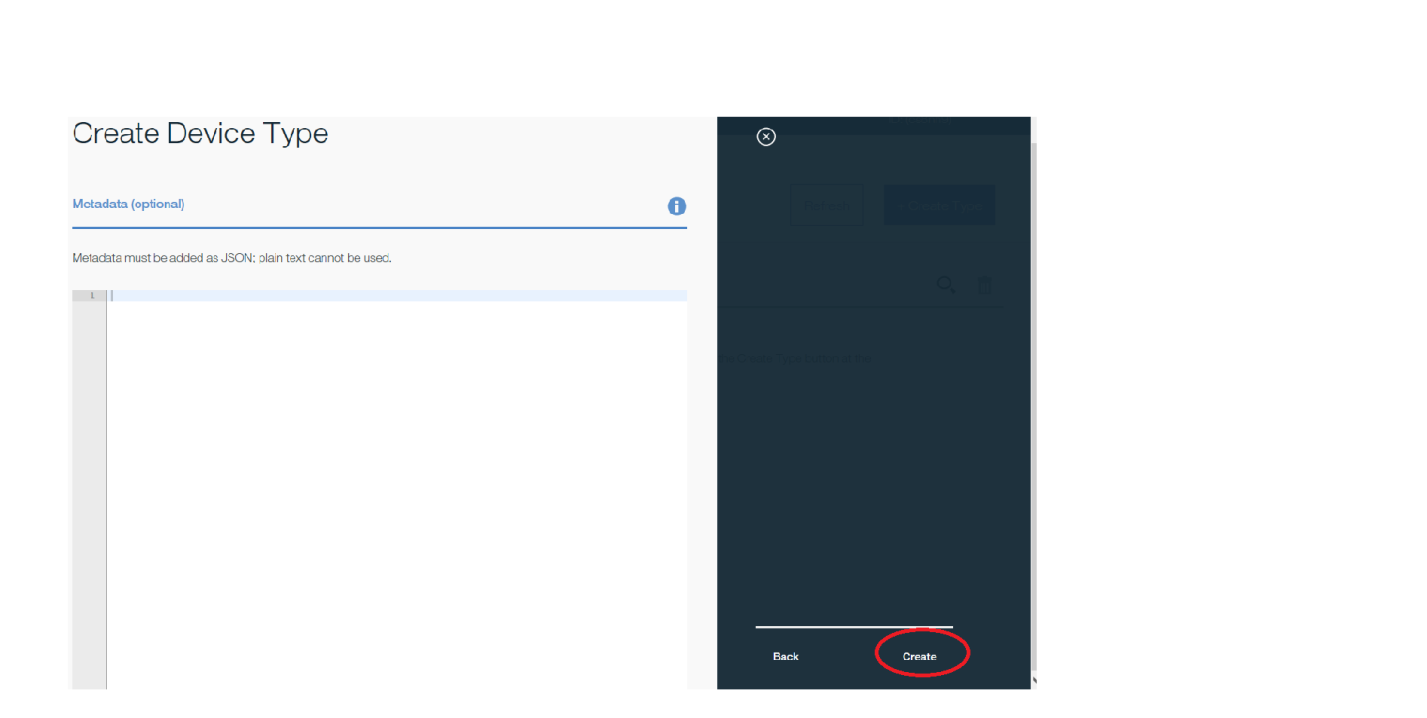
1. Click the “Next” button



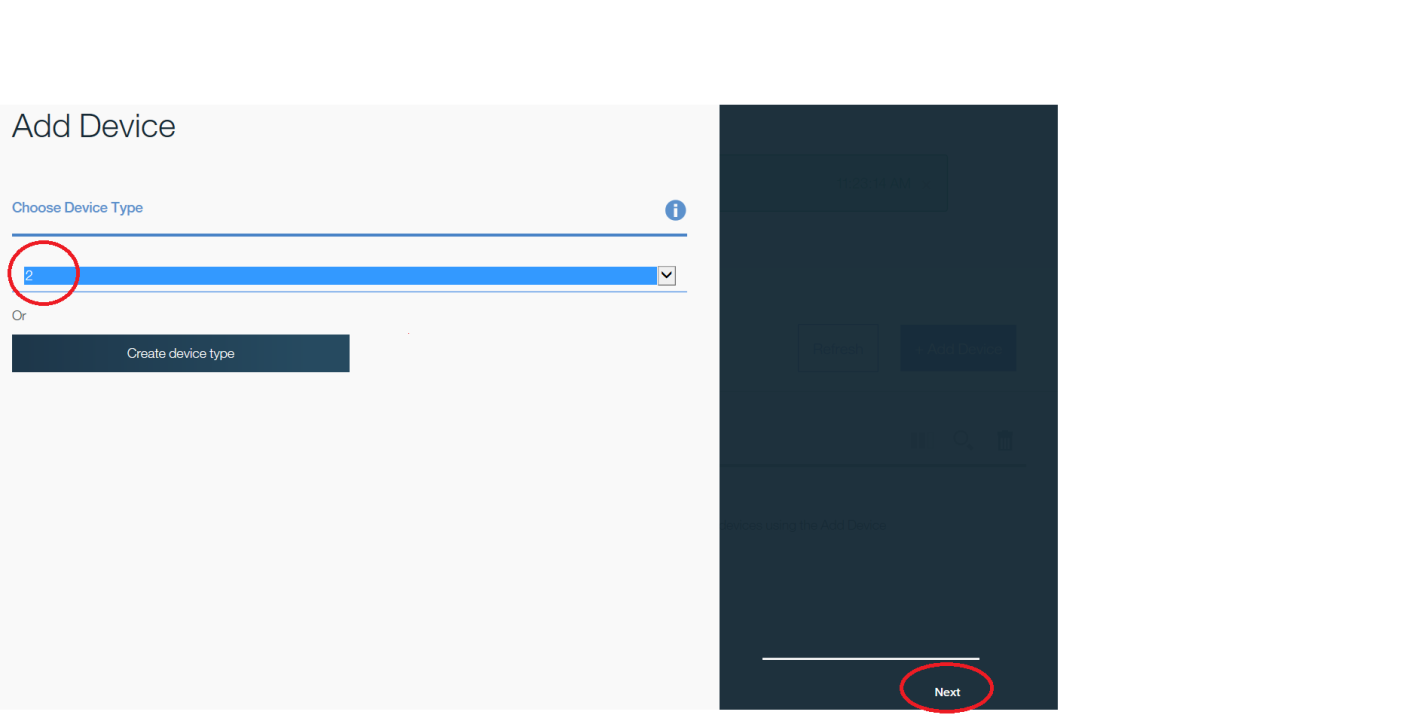
1. Click the “Next” button



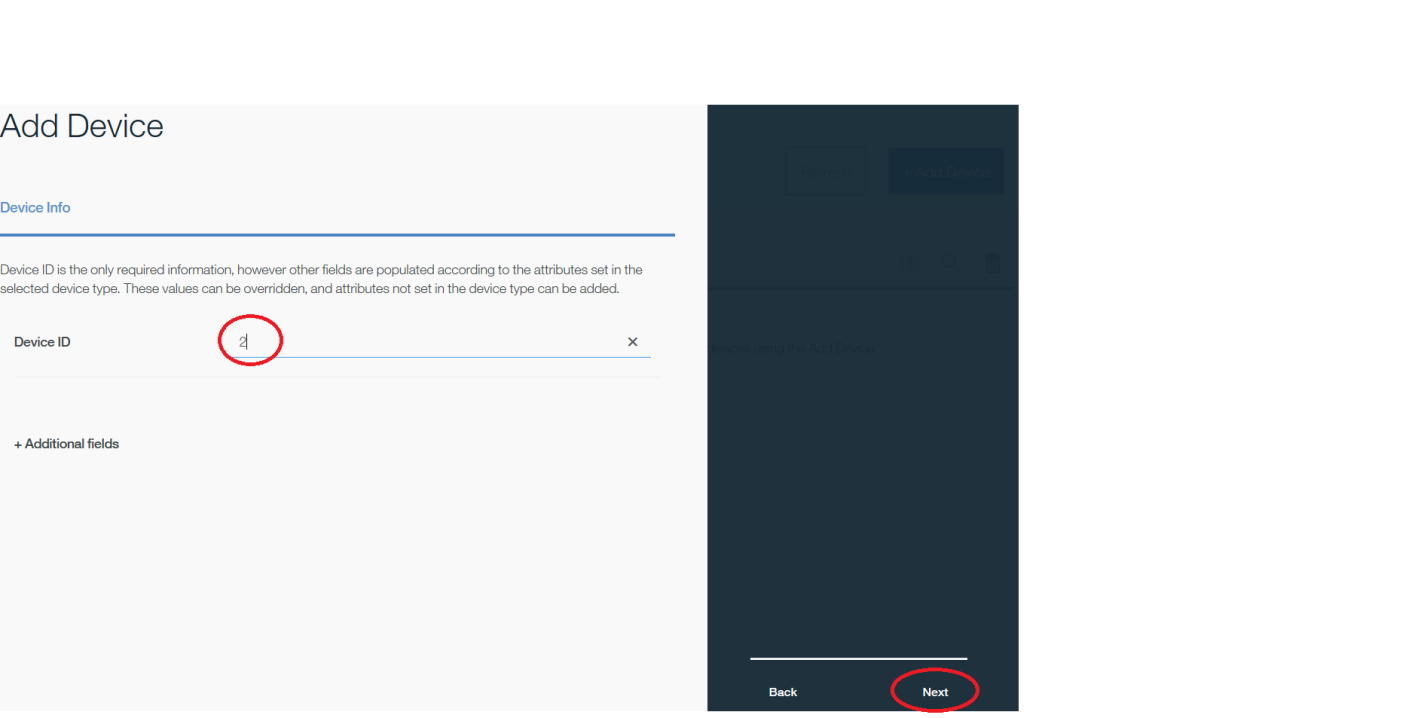
1. Click the “Create” button



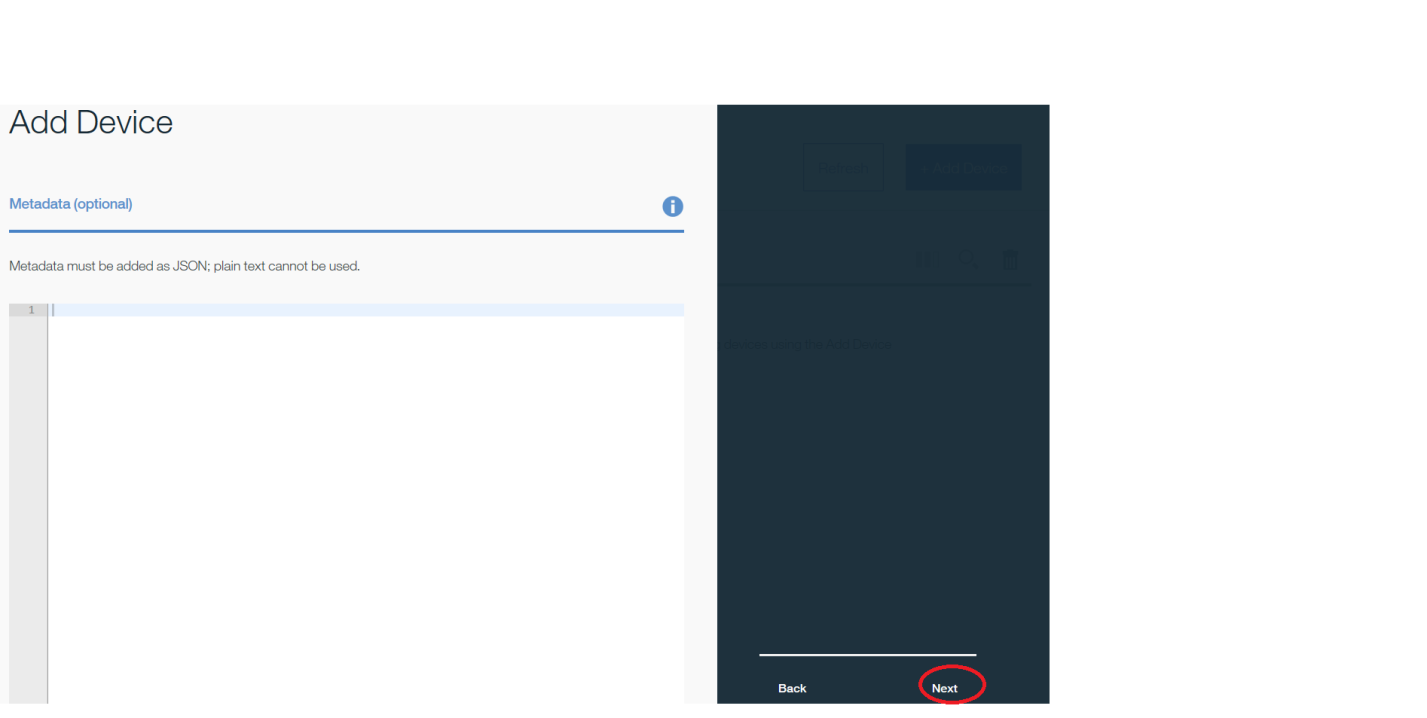
1. Select the device type “2” and click the “Next “ button



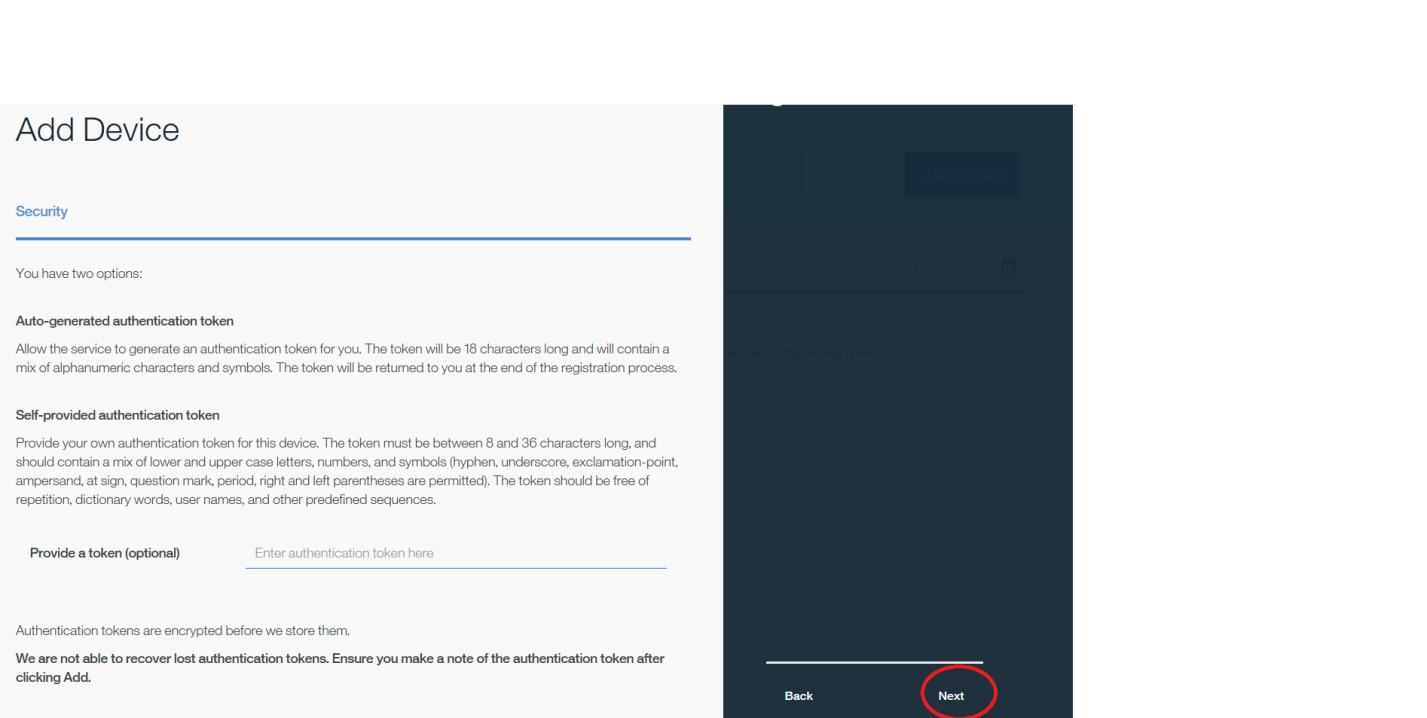
1. Enter 2 as the Device ID and click the “Next” button



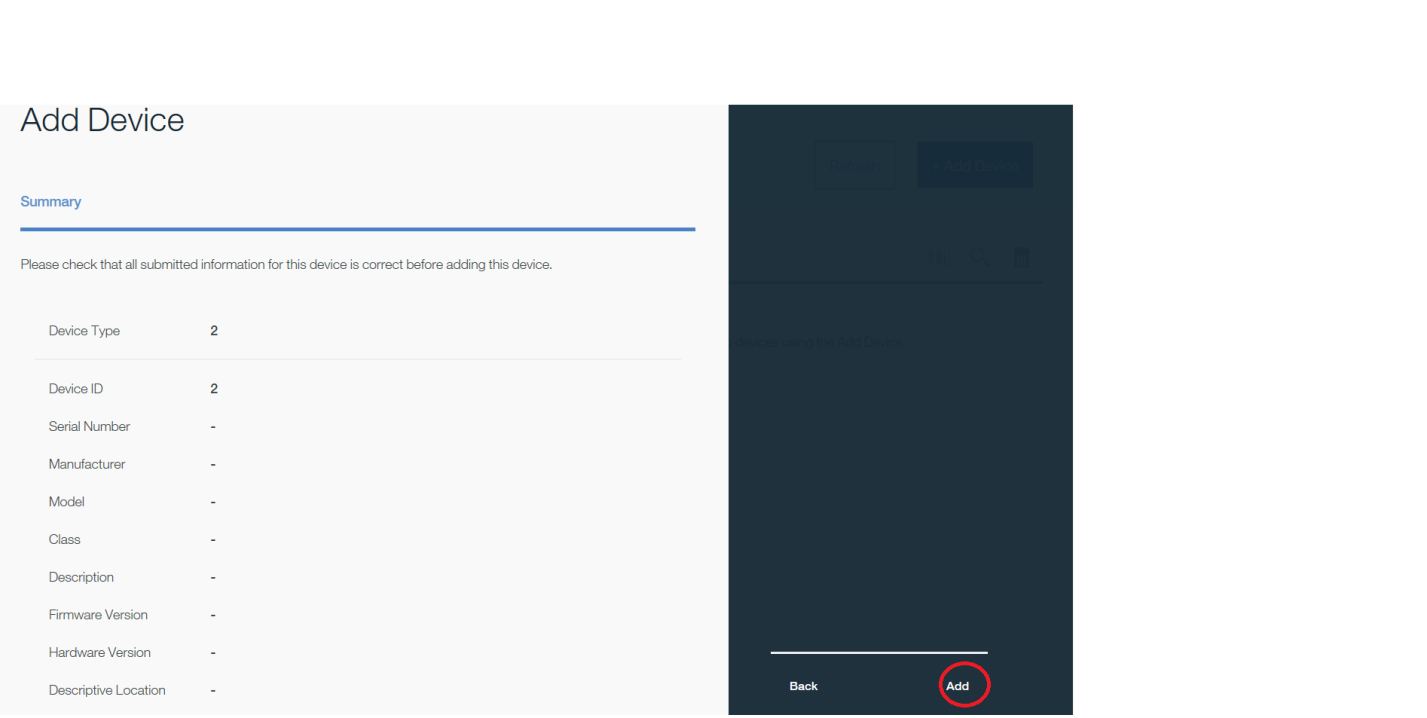
1. Click the “Next” button



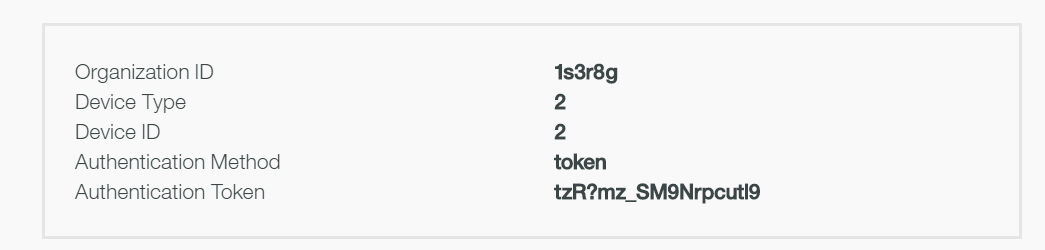
1. Click the “Next” button



1. Click the “Add” button



1. Copy down the device credentials into a text file and save it:



For this example, the text file should have the following lines:

*“*

Organization ID 1s3r8g

Device Type 2

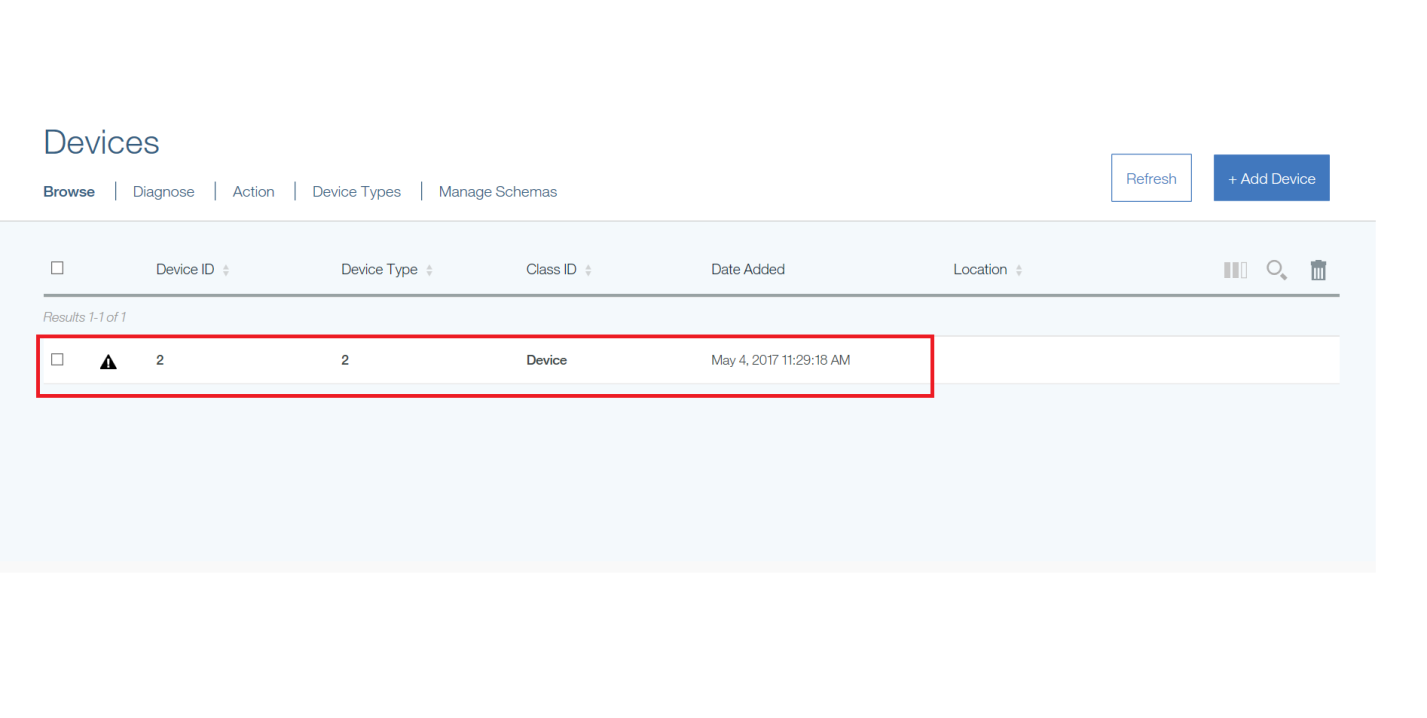
Device ID 2

Authentication Method token

Authentication Token tzR?mz\_SM9Nrpcutl9

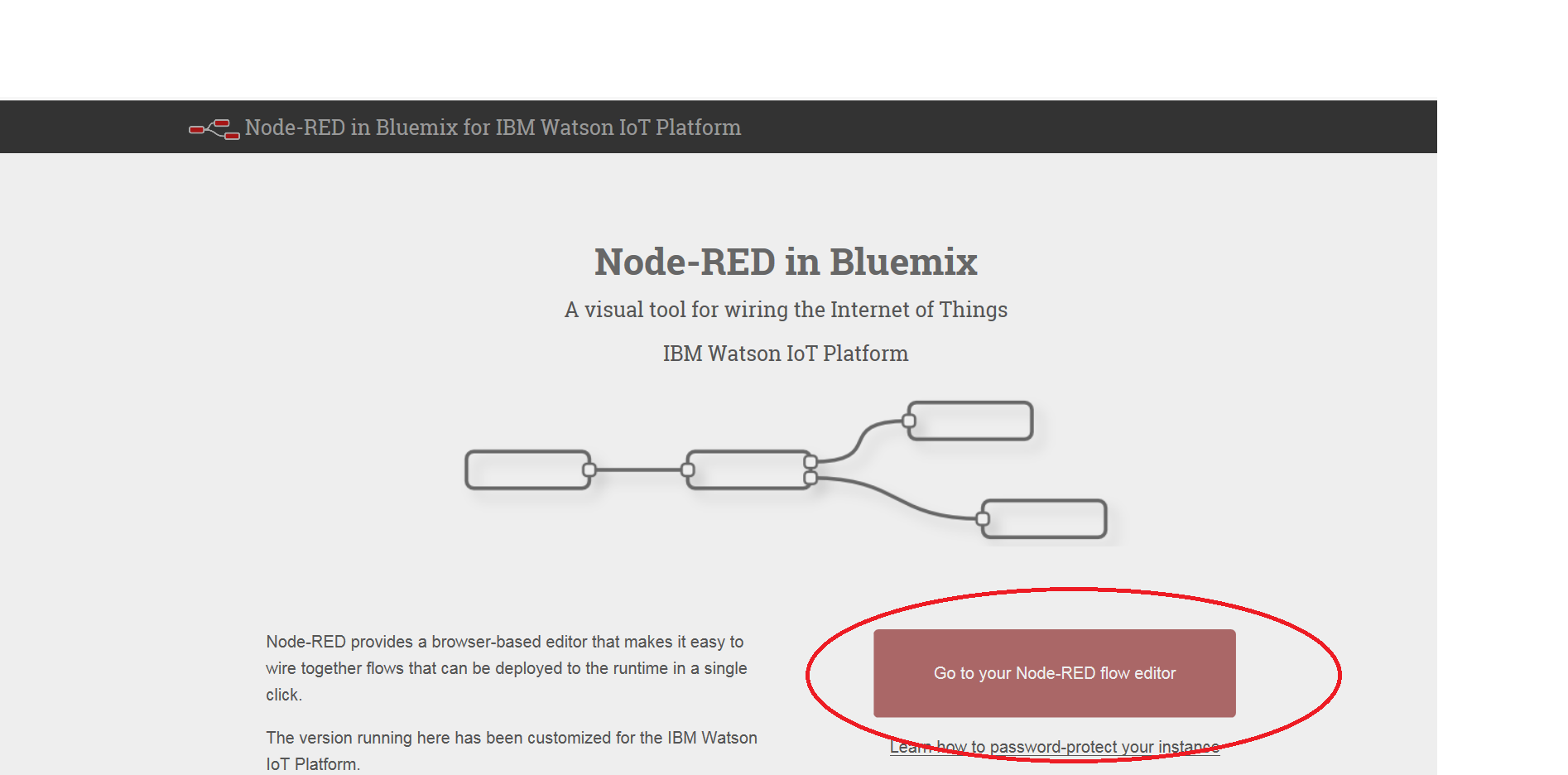
“

1. Click the close button on the Add Device screen, you should see your device:

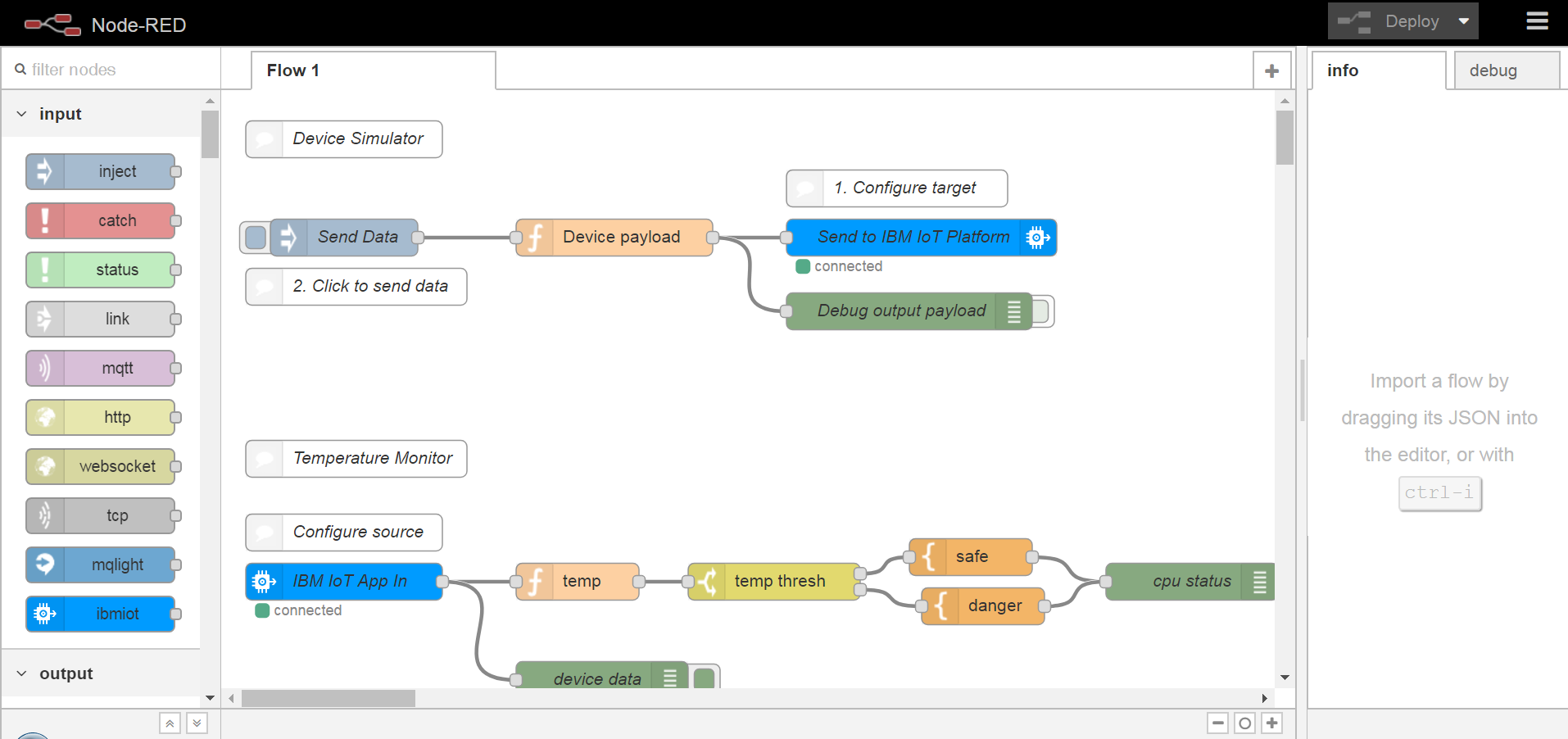


## Import the I3 IOT application into the IOT template app created

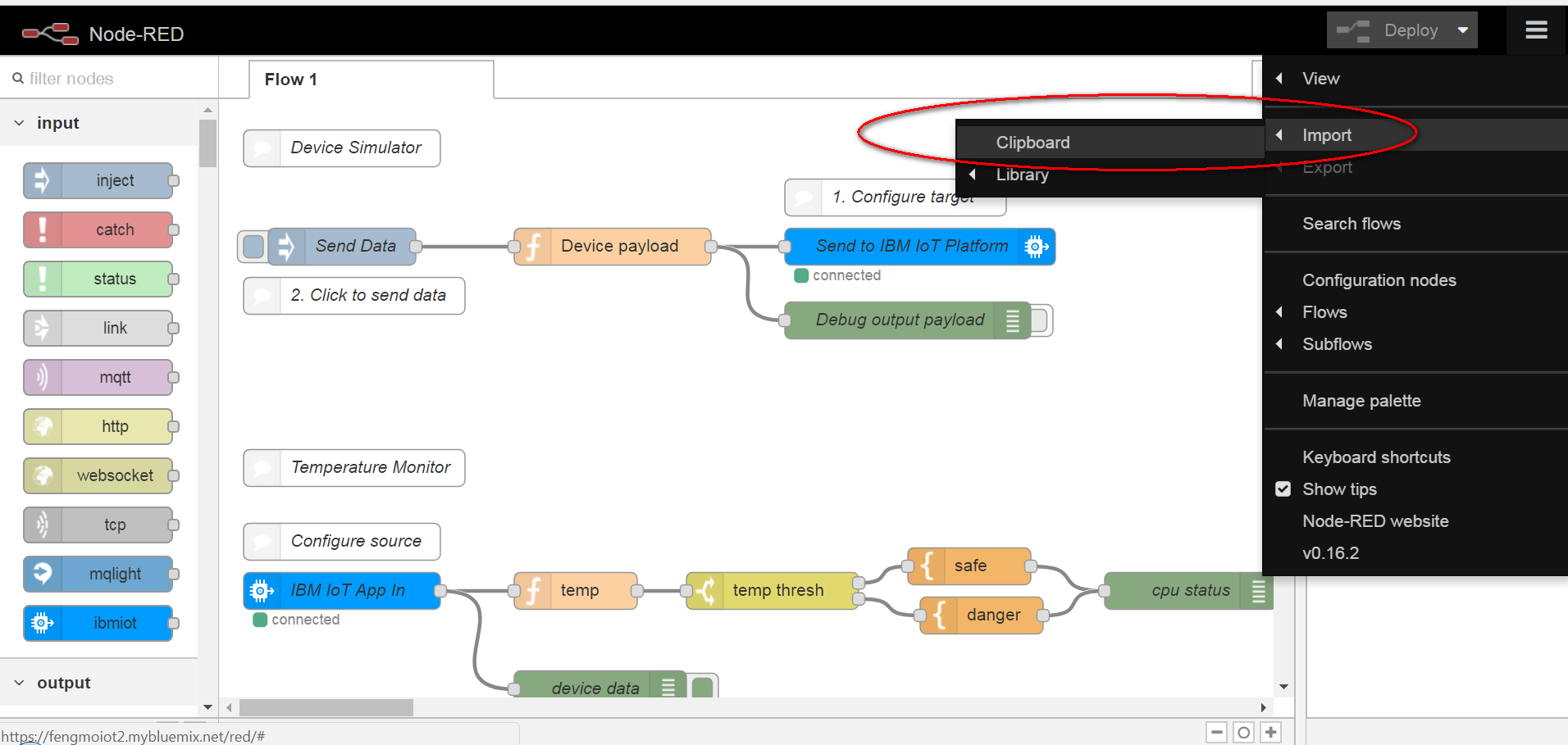
1. Go to the Node-RED graphical editor for your app by entering the URL in the browser, the example is <http://fengmoiot2.mybluemix.net> .
2. Click the red button to go to the Node-RED visual programming editor:

Note: You may not be able to do this inside the TI firewall. You will need to do it externally. 

1. After a pause, you should this screen:



1. Import the I3 IOT app source code into the editor by selecting “Import”->”Clipboard”:



1. Paste the following I3 IOT app (all lines in red) source code into the box:

**[**

**{**

**"id": "7698f79a.cd11b",**

**"type": "tab",**

**"label": "Flow 1"**

**},**

**{**

**"id": "9a007402.30b4c8",**

**"type": "cloudant",**

**"z": "",**

**"host": "7514bf55-276f-4b67-876d-54b0657fb4f3-bluemix.cloudant.com",**

**"name": ""**

**},**

**{**

**"id": "378b0ead.471c92",**

**"type": "cloudant",**

**"z": "",**

**"host": "7514bf55-276f-4b67-876d-54b0657fb4f3-bluemix.cloudant.com",**

**"name": ""**

**},**

**{**

**"id": "87c9260b.b082a8",**

**"type": "mqtt-broker",**

**"z": "",**

**"broker": "qv1pl7.messaging.internetofthings.ibmcloud.com",**

**"port": "1883",**

**"clientid": "d:qv1pl7:2:2",**

**"usetls": false,**

**"compatmode": true,**

**"keepalive": "60",**

**"cleansession": true,**

**"willTopic": "",**

**"willQos": "0",**

**"willRetain": "false",**

**"willPayload": "",**

**"birthTopic": "",**

**"birthQos": "0",**

**"birthRetain": "false",**

**"birthPayload": ""**

**},**

**{**

**"id": "dc278c87.c9f05",**

**"type": "cloudant in",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"cloudant": "9a007402.30b4c8",**

**"database": "databasedemo",**

**"service": "\_ext\_",**

**"search": "\_all\_",**

**"design": "",**

**"index": "",**

**"x": 316.5999450683594,**

**"y": 187.29000854492188,**

**"wires": [**

**[**

**"4e773d16.d2ef34",**

**"d4b632e2.d8e0d"**

**]**

**]**

**},**

**{**

**"id": "3a2edce9.a97bd4",**

**"type": "inject",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"topic": "",**

**"payload": "{ query: \"\", limit: 100} ",**

**"payloadType": "str",**

**"repeat": "",**

**"crontab": "",**

**"once": false,**

**"x": 83.18997192382812,**

**"y": 206.75997924804688,**

**"wires": [**

**[**

**"dc278c87.c9f05"**

**]**

**]**

**},**

**{**

**"id": "4e773d16.d2ef34",**

**"type": "debug",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"active": true,**

**"console": "false",**

**"complete": "payload",**

**"x": 784.1998901367188,**

**"y": 165.1300048828125,**

**"wires": []**

**},**

**{**

**"id": "42cc0cc6.df58f4",**

**"type": "http in",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"url": "/sites",**

**"method": "post",**

**"swaggerDoc": "",**

**"x": 127.39999389648438,**

**"y": 336.1600341796875,**

**"wires": [**

**[**

**"1f91bc06.bac5c4",**

**"5eb08126.2c30f",**

**"bbc84f32.87a37"**

**]**

**]**

**},**

**{**

**"id": "631283bc.0fba2c",**

**"type": "cloudant out",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"cloudant": "378b0ead.471c92",**

**"database": "databasedemo",**

**"service": "\_ext\_",**

**"payonly": false,**

**"operation": "insert",**

**"x": 833.7899856567383,**

**"y": 277.8500328063965,**

**"wires": []**

**},**

**{**

**"id": "94b90d8.69db87",**

**"type": "debug",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"active": false,**

**"console": "false",**

**"complete": "true",**

**"x": 810.5999145507812,**

**"y": 381.94000244140625,**

**"wires": []**

**},**

**{**

**"id": "b34dce14.bd5578",**

**"type": "change",**

**"z": "7698f79a.cd11b",**

**"name": "addtimestamp",**

**"rules": [**

**{**

**"t": "set",**

**"p": "timestamp",**

**"pt": "msg",**

**"to": "",**

**"tot": "date"**

**}**

**],**

**"action": "",**

**"property": "",**

**"from": "",**

**"to": "",**

**"reg": false,**

**"x": 695.3999328613281,**

**"y": 335.8599853515625,**

**"wires": [**

**[**

**"94b90d8.69db87",**

**"62690881.b304e4",**

**"631283bc.0fba2c"**

**]**

**]**

**},**

**{**

**"id": "1f91bc06.bac5c4",**

**"type": "change",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"rules": [**

**{**

**"t": "delete",**

**"p": "req",**

**"pt": "msg"**

**}**

**],**

**"action": "",**

**"property": "",**

**"from": "",**

**"to": "",**

**"reg": false,**

**"x": 319.3900146484375,**

**"y": 336.3999938964844,**

**"wires": [**

**[**

**"4372873.b16c0f8"**

**]**

**]**

**},**

**{**

**"id": "4372873.b16c0f8",**

**"type": "change",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"rules": [**

**{**

**"t": "delete",**

**"p": "res",**

**"pt": "msg"**

**}**

**],**

**"action": "",**

**"property": "",**

**"from": "",**

**"to": "",**

**"reg": false,**

**"x": 512.2000122070312,**

**"y": 336.4099884033203,**

**"wires": [**

**[**

**"b34dce14.bd5578"**

**]**

**]**

**},**

**{**

**"id": "bb0d5fc4.6a07f",**

**"type": "http in",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"url": "/start",**

**"method": "get",**

**"swaggerDoc": "",**

**"x": 125.80001831054688,**

**"y": 489.75999450683594,**

**"wires": [**

**[**

**"82c8a82b.526068"**

**]**

**]**

**},**

**{**

**"id": "728be92e.cf47c",**

**"type": "http response",**

**"z": "7698f79a.cd11b",**

**"name": "home page",**

**"x": 693.7899169921875,**

**"y": 481.12998962402344,**

**"wires": []**

**},**

**{**

**"id": "60b2d9f0.85f1c4",**

**"type": "function",**

**"z": "7698f79a.cd11b",**

**"name": "compose home page",**

**"func": "msg.payload = `<html>\n<head>\n</head>\n<body>\n<form action=\"http://fengmoiot2.mybluemix.net/sites\" method=\"post\">\n<fieldset>\n<legend>Sensor Data:</legend>\nGateway:<br>\n<input type=\"text\" name=\"gwid\" value=`+msg.gwid+`><br>\nSensor:<br>\n<input type=\"text\" name=\"sensorid\" value=`+msg.sensorid+`><br>\nTemperature:<br>\n<input type=\"text\" name=\"temp\" value=`+msg.temp+`><br>\nHumidity:<br>\n<input type=\"text\" name=\"hum\" value=`+msg.hum+`><br>\nPressure:<br>\n<input type=\"text\" name=\"pres\" value=`+msg.pres+`><br><br>\n<input type=\"submit\" value=\"Submit\">\n</fieldset></form>\n</body>\n</html>`;\nreturn msg;",**

**"outputs": 1,**

**"noerr": 0,**

**"x": 494.98992919921875,**

**"y": 463.66004943847656,**

**"wires": [**

**[**

**"728be92e.cf47c",**

**"48a0e1ab.d56554"**

**]**

**]**

**},**

**{**

**"id": "82c8a82b.526068",**

**"type": "function",**

**"z": "7698f79a.cd11b",**

**"name": "set defaults",**

**"func": "msg.gwid = '\"kilby GW 1\"';\nmsg.sensorid = '\"kilby S1\"';\nmsg.temp = 24;\nmsg.hum = 40;\nmsg.pres = 1010;\nreturn msg;",**

**"outputs": 1,**

**"noerr": 0,**

**"x": 285.7900085449219,**

**"y": 486.7799835205078,**

**"wires": [**

**[**

**"60b2d9f0.85f1c4"**

**]**

**]**

**},**

**{**

**"id": "5eb08126.2c30f",**

**"type": "function",**

**"z": "7698f79a.cd11b",**

**"name": "extract data",**

**"func": "msg.gwid = '\"'+msg.payload.gwid+'\"';\nmsg.sensorid = '\"'+msg.payload.sensorid+'\"';\nmsg.temp = msg.payload.temp;\nmsg.hum = msg.payload.hum;\nmsg.pres = msg.payload.pres;\nreturn msg;",**

**"outputs": 1,**

**"noerr": 0,**

**"x": 282.5899963378906,**

**"y": 439.57997131347656,**

**"wires": [**

**[**

**"60b2d9f0.85f1c4"**

**]**

**]**

**},**

**{**

**"id": "c7e4f5a6.00bd48",**

**"type": "inject",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"topic": "",**

**"payload": "{ query: \"\", limit: 100 } ",**

**"payloadType": "str",**

**"repeat": "",**

**"crontab": "",**

**"once": false,**

**"x": 93.39993286132812,**

**"y": 104.15995788574219,**

**"wires": [**

**[**

**"e66f839b.7487d"**

**]**

**]**

**},**

**{**

**"id": "800d03ac.b44cd8",**

**"type": "cloudant out",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"cloudant": "9a007402.30b4c8",**

**"database": "databasedemo",**

**"service": "\_ext\_",**

**"payonly": false,**

**"operation": "delete",**

**"x": 708.9899291992188,**

**"y": 87.23997497558594,**

**"wires": []**

**},**

**{**

**"id": "e66f839b.7487d",**

**"type": "cloudant in",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"cloudant": "9a007402.30b4c8",**

**"database": "databasedemo",**

**"service": "\_ext\_",**

**"search": "\_all\_",**

**"design": "",**

**"index": "",**

**"x": 300.18994140625,**

**"y": 96.83995056152344,**

**"wires": [**

**[**

**"a055889f.17e1d",**

**"13d6b65b.3ee13e"**

**]**

**]**

**},**

**{**

**"id": "a055889f.17e1d",**

**"type": "debug",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"active": true,**

**"console": "false",**

**"complete": "true",**

**"x": 448.18988037109375,**

**"y": 54.73997497558594,**

**"wires": []**

**},**

**{**

**"id": "13d6b65b.3ee13e",**

**"type": "function",**

**"z": "7698f79a.cd11b",**

**"name": "delete document",**

**"func": "if (msg.payload.length > 0)\n{\n msg.payload = {\_id:msg.payload[0].\_id,\n \_rev:msg.payload[0].\_rev};\n}\nreturn msg;",**

**"outputs": 1,**

**"noerr": 0,**

**"x": 509.7899475097656,**

**"y": 107.83995056152344,**

**"wires": [**

**[**

**"800d03ac.b44cd8"**

**]**

**]**

**},**

**{**

**"id": "a18aacf7.56dde",**

**"type": "comment",**

**"z": "7698f79a.cd11b",**

**"name": "Delete documents one at a time:",**

**"info": "",**

**"x": 169,**

**"y": 44.649993896484375,**

**"wires": []**

**},**

**{**

**"id": "cdf4d161.fb07a8",**

**"type": "comment",**

**"z": "7698f79a.cd11b",**

**"name": "Query the database:",**

**"info": "",**

**"x": 128.19000244140625,**

**"y": 164.63999938964844,**

**"wires": []**

**},**

**{**

**"id": "51d8acd5.512a9",**

**"type": "comment",**

**"z": "7698f79a.cd11b",**

**"name": "Add data to the database:",**

**"info": "",**

**"x": 144.99002075195312,**

**"y": 297.44000244140625,**

**"wires": []**

**},**

**{**

**"id": "d4b632e2.d8e0d",**

**"type": "function",**

**"z": "7698f79a.cd11b",**

**"name": "graph\_sensor\_type\_0",**

**"func": "var xydata=\"[\";\nfor (i = 0; i < msg.payload.length-1; i++)\n{\n\txydata += \"{x:\" + msg.payload[i].timestamp/10000 + \", y:\" + msg.payload[i].payload.sensor\_0.data.channel + \"},\";\n}\nxydata += \"{x:\" + msg.payload[i].timestamp/10000 + \", y:\" + msg.payload[i].payload.sensor\_0.data.channel + \"}]\";\n\nmsg.payload = `<!DOCTYPE HTML>\n<html>\n<head>\n <script type=\"text/javascript\">\n\n \n var graphData = {\n title:{\n text: \"Channel\"\n },\n\n data: [\n {\n type: \"line\",\n dataPoints:`+xydata+`\n }\n ]\n };\n\nwindow.onload = function () \n{\n var chart = new CanvasJS.Chart(\"chartContainer\",graphData);\n\n chart.render();\n}\n</script>\n<script src=\"http://canvasjs.com/assets/script/canvasjs.min.js\"></script>\n</head>\n<body>\n<script type=\"text/javascript\">\nsetInterval(function(){window.location.reload(true);}, 2000);\n</script>\n <div id=\"chartContainer\" style=\"height: 800px; width: 100%;\">\n </div>\n</body>\n</html>`;\nreturn msg;",**

**"outputs": 1,**

**"noerr": 0,**

**"x": 660.5999755859375,**

**"y": 225.98004150390625,**

**"wires": [**

**[**

**"73db5ef5.3918d8"**

**]**

**]**

**},**

**{**

**"id": "2a984dd5.da4726",**

**"type": "function",**

**"z": "7698f79a.cd11b",**

**"name": "query",**

**"func": "msg.payload='{ query: \"\", limit: 100 } ';\nreturn msg;",**

**"outputs": 1,**

**"noerr": 0,**

**"x": 196.19000244140625,**

**"y": 254.08999633789062,**

**"wires": [**

**[**

**"dc278c87.c9f05"**

**]**

**]**

**},**

**{**

**"id": "922351a5.d5d56",**

**"type": "http in",**

**"z": "7698f79a.cd11b",**

**"name": "query",**

**"url": "/query",**

**"method": "get",**

**"swaggerDoc": "",**

**"x": 66.58999633789062,**

**"y": 251.44000244140625,**

**"wires": [**

**[**

**"2a984dd5.da4726"**

**]**

**]**

**},**

**{**

**"id": "73db5ef5.3918d8",**

**"type": "http response",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"x": 870.5999755859375,**

**"y": 214.39999389648438,**

**"wires": []**

**},**

**{**

**"id": "2b7f73b1.7b488",**

**"type": "function",**

**"z": "7698f79a.cd11b",**

**"name": "filter and sort",**

**"func": "function sortFunc(a,b)\n{\n return(a.timestamp > b.timestamp);\n}\n\nvar i = 0;\nwhile (i < msg.payload.length)\n{\n if (msg.payload[i].payload.type != \"sensor\_type\_0\")\n {\n if (i < msg.payload.length - 1)\n {\n msg.payload[i] = msg.payload[i+1];\n i--;\n }\n \n msg.payload.length--;\n }\n i++;\n}\n\nvar sortedMsg = msg.payload; //msg.payload.sort(sortFunc);\n\nmsg.payload = sortedMsg;\n\nreturn msg;",**

**"outputs": 1,**

**"noerr": 0,**

**"x": 522.5899047851562,**

**"y": 157.3199920654297,**

**"wires": [**

**[]**

**]**

**},**

**{**

**"id": "48a0e1ab.d56554",**

**"type": "debug",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"active": false,**

**"console": "false",**

**"complete": "true",**

**"x": 619.3900146484375,**

**"y": 532.3299560546875,**

**"wires": []**

**},**

**{**

**"id": "bbc84f32.87a37",**

**"type": "debug",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"active": false,**

**"console": "false",**

**"complete": "true",**

**"x": 101.80000305175781,**

**"y": 391.53997802734375,**

**"wires": []**

**},**

**{**

**"id": "73f20af4.6aa6cc",**

**"type": "function",**

**"z": "7698f79a.cd11b",**

**"name": "mqtt",**

**"func": "var data={};\ndata.d = {timestamp:msg.payload};\nmsg.payload = data;\nreturn msg;",**

**"outputs": 1,**

**"noerr": 0,**

**"x": 598.5999755859375,**

**"y": 609.4199829101562,**

**"wires": [**

**[**

**"2fe5265.cc36d5a",**

**"f89eed9c.631828"**

**]**

**]**

**},**

**{**

**"id": "622bb528.7f2f44",**

**"type": "inject",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"topic": "",**

**"payload": "",**

**"payloadType": "date",**

**"repeat": "",**

**"crontab": "",**

**"once": false,**

**"x": 398.5899658203125,**

**"y": 605.3599853515625,**

**"wires": [**

**[**

**"73f20af4.6aa6cc"**

**]**

**]**

**},**

**{**

**"id": "2fe5265.cc36d5a",**

**"type": "debug",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"active": false,**

**"console": "false",**

**"complete": "true",**

**"x": 708.989990234375,**

**"y": 710.4999694824219,**

**"wires": []**

**},**

**{**

**"id": "62690881.b304e4",**

**"type": "function",**

**"z": "7698f79a.cd11b",**

**"name": "demux",**

**"func": "var omsgStr = '{\"'+msg.payload.gateway\_0.name+'\":';\nomsgStr += '{\"'+msg.payload.sensor\_0.\_id+'\":';\nomsgStr += JSON.stringify(msg.payload);\nomsgStr += '}}';\nmsg.payload = omsgStr;\n\nreturn msg;",**

**"outputs": 1,**

**"noerr": 0,**

**"x": 849.5,**

**"y": 483.45001220703125,**

**"wires": [**

**[**

**"1ec47009.cee9b",**

**"f89eed9c.631828"**

**]**

**]**

**},**

**{**

**"id": "1ec47009.cee9b",**

**"type": "debug",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"active": false,**

**"console": "false",**

**"complete": "true",**

**"x": 998.5,**

**"y": 431.29998779296875,**

**"wires": []**

**},**

**{**

**"id": "f89eed9c.631828",**

**"type": "mqtt out",**

**"z": "7698f79a.cd11b",**

**"name": "",**

**"topic": "iot-2/evt/timestamp/fmt/json",**

**"qos": "",**

**"retain": "",**

**"broker": "87c9260b.b082a8",**

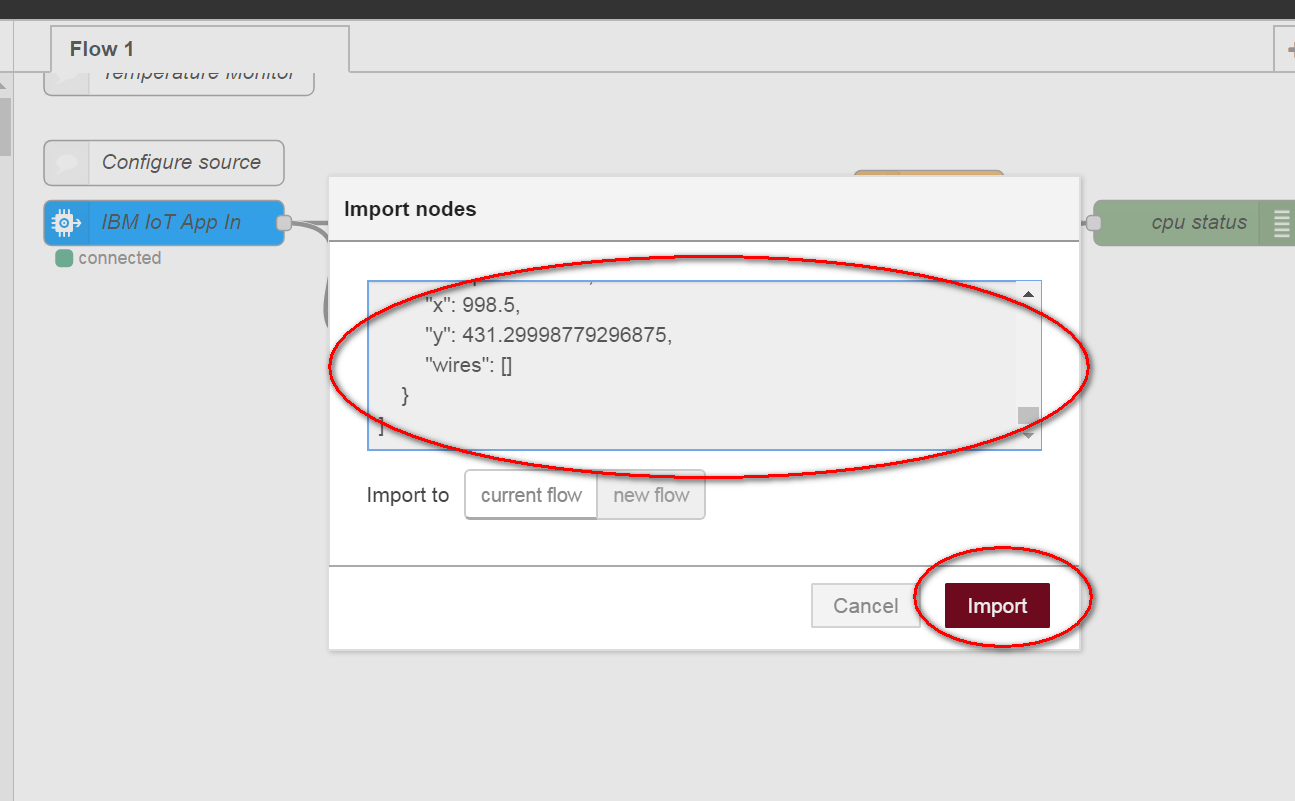
**"x": 1000.5,**

**"y": 587.2999877929688,**

**"wires": []**

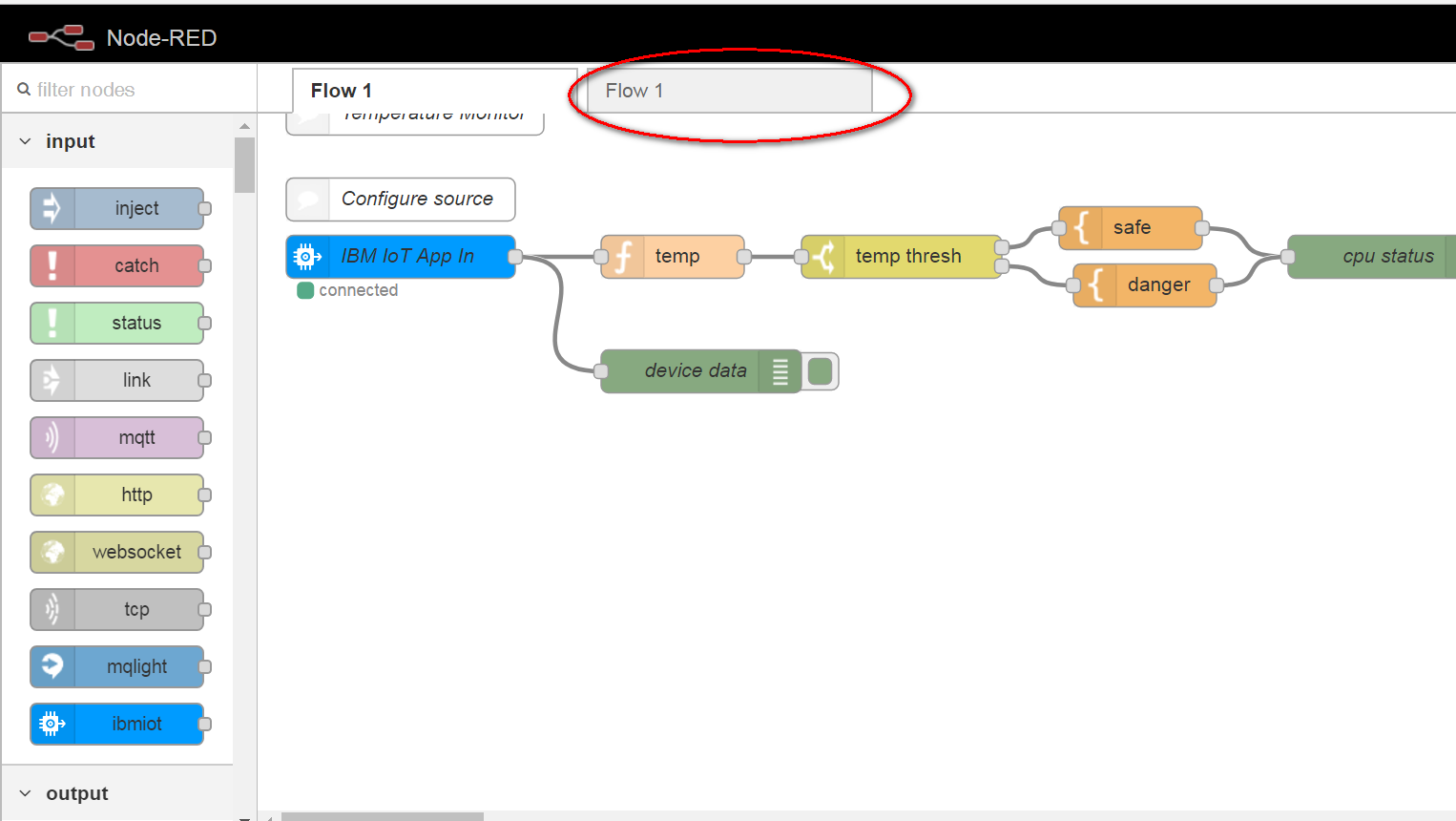
**}**

**]**

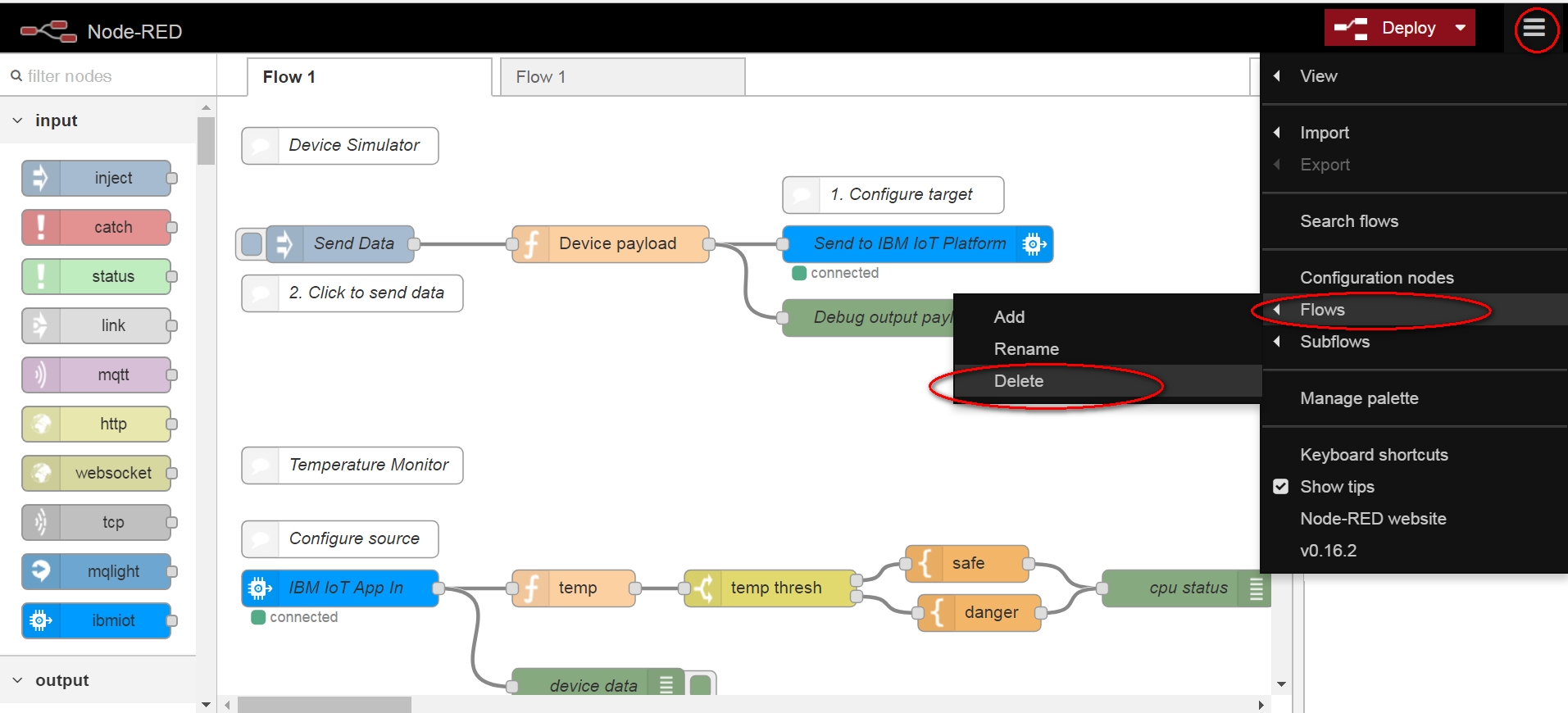


Click the “Import” button

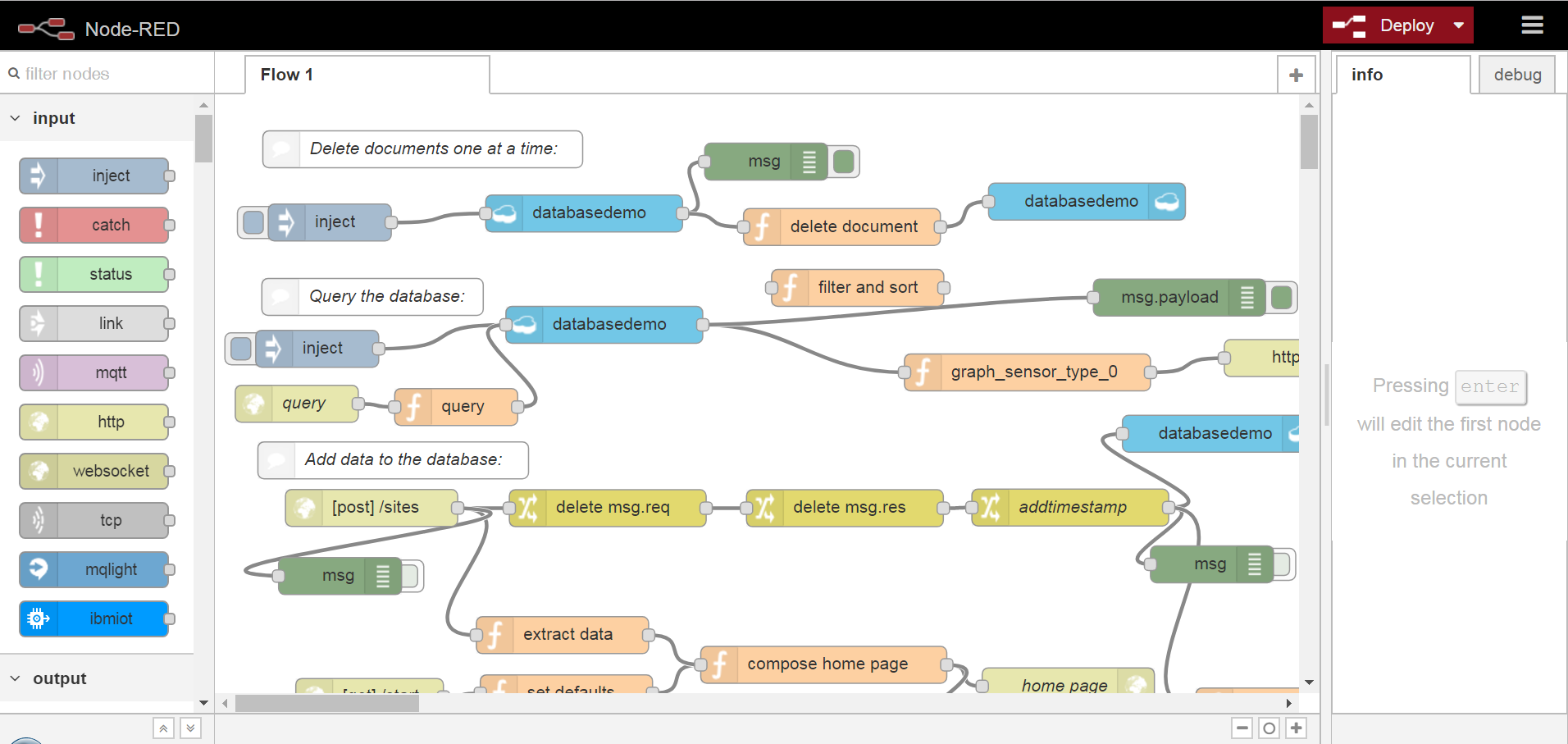
1. You should see a second “Flow 1” tab:



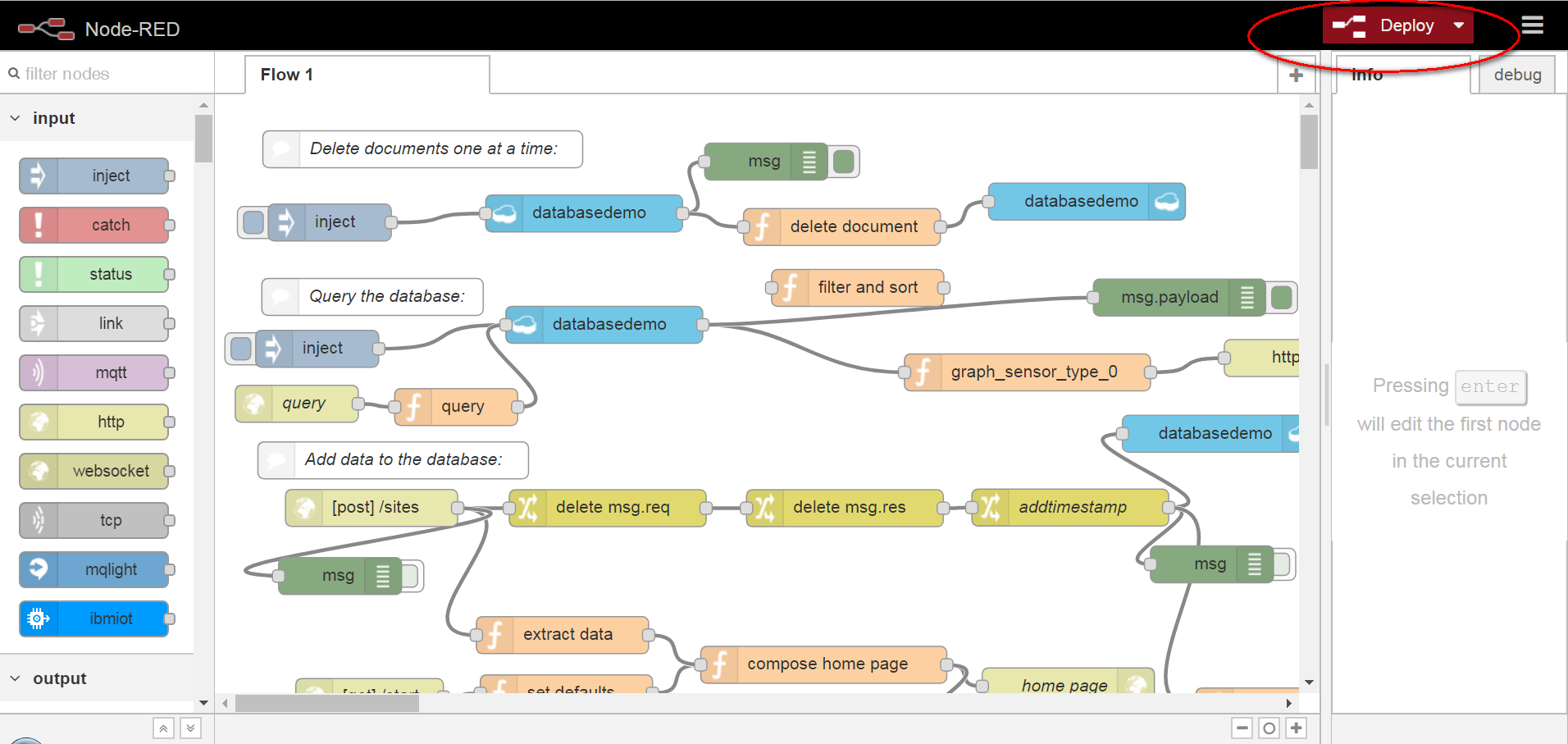
1. Delete the original “Flow 1”:



1. You should now have the new “Flow 1” tab screen:



1. Click the “Deploy” button (upper right)



## Configure the database nodes to access the external Cloudant database

In this example, we have a Cloudant database created in a separate Cloudant database service. The information about the database is needed to configure the database nodes:

**{**

**“database name”: “databasedemo”,**

**"username": "7514bf55-276f-4b67-876d-54b0657fb4f3-bluemix",**

**"password": "b824a67c1496ac11d286e81b5cfe4f5f27e6b61eb11972fca2650ddb8a1f64ec",**

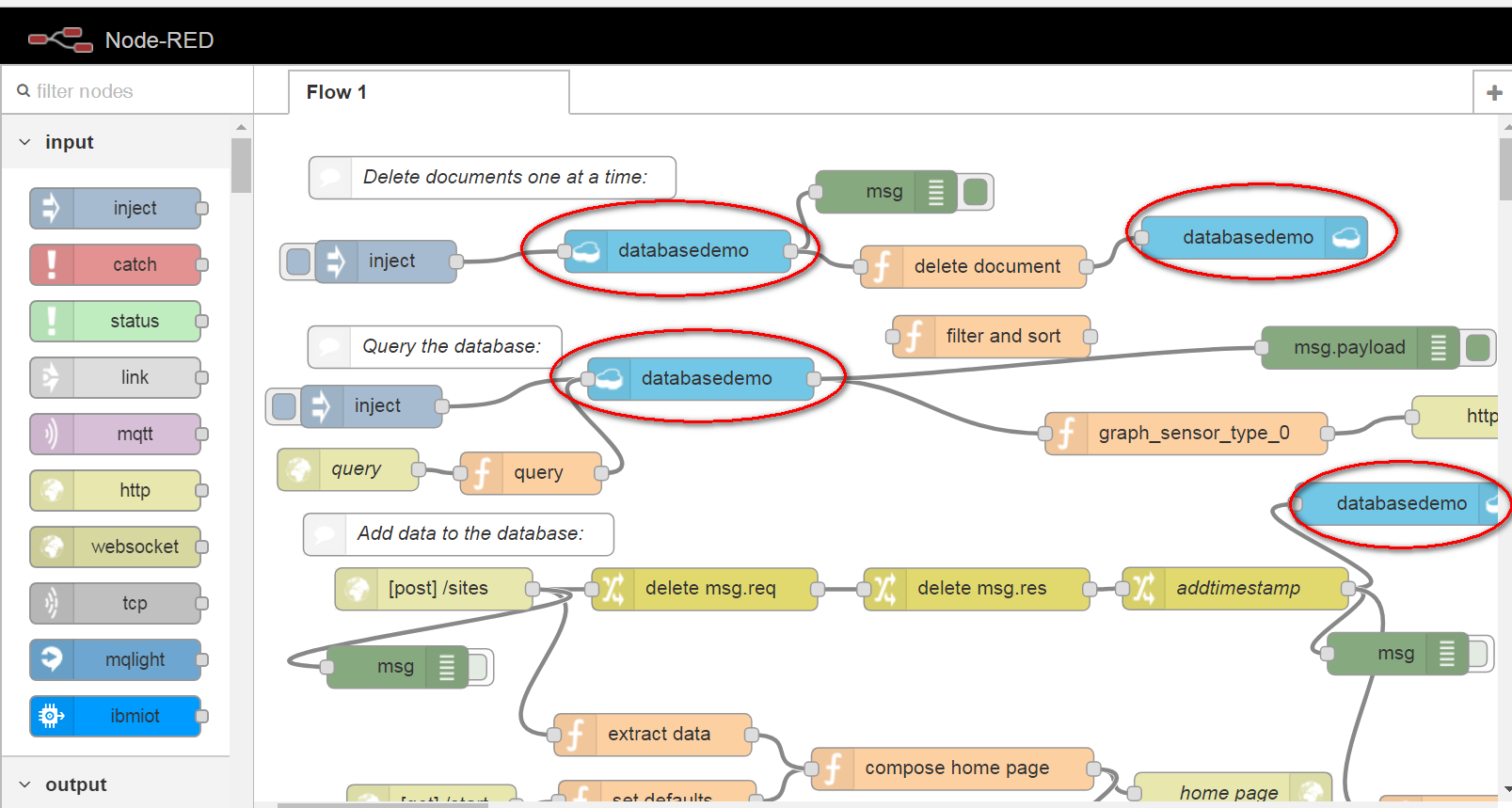
**"host": "7514bf55-276f-4b67-876d-54b0657fb4f3-bluemix.cloudant.com",**

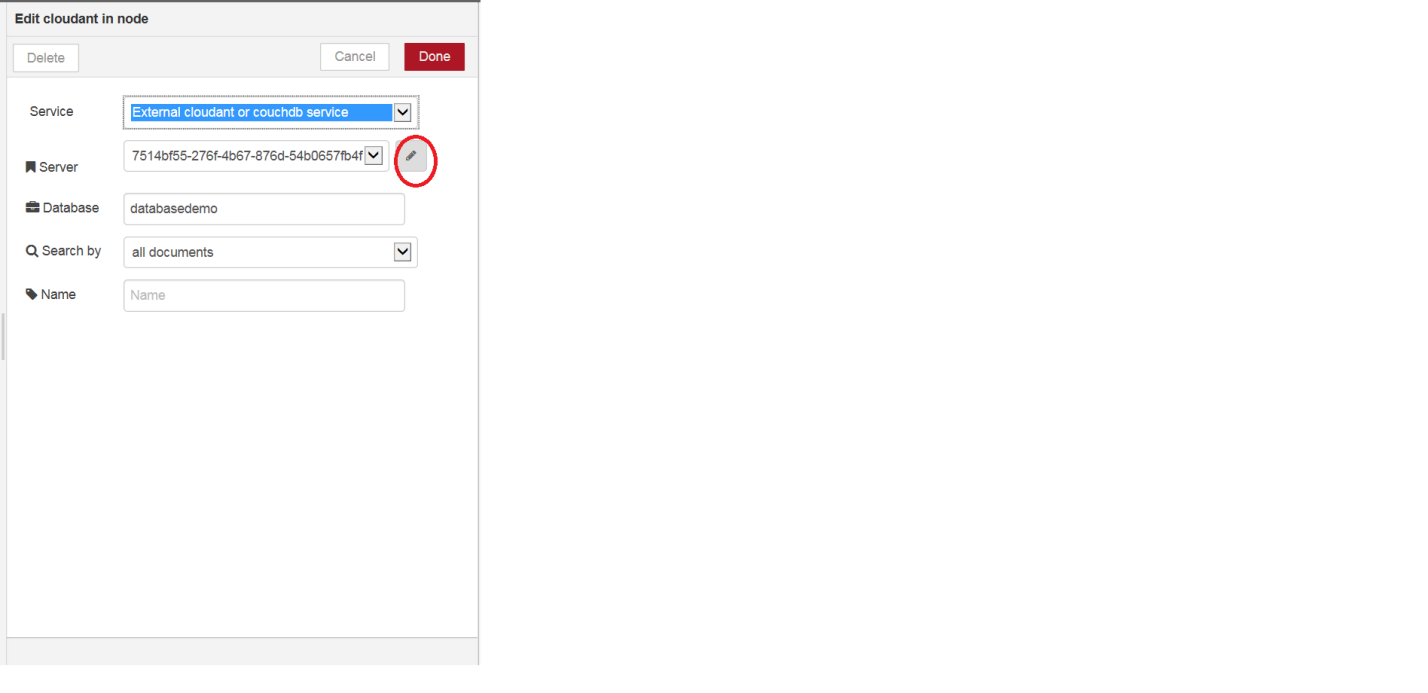
**"port": 443,**

**"url": "https://7514bf55-276f-4b67-876d-54b0657fb4f3-bluemix:b824a67c1496ac11d286e81b5cfe4f5f27e6b61eb11972fca2650ddb8a1f64ec@7514bf55-276f-4b67-876d-54b0657fb4f3-bluemix.cloudant.com"**

**}**

You can create your own Cloudant database in a database service. In which case, you need to note down the information about the database and replace the information used here with the actual database information, Host, Username, Password, and the name of the database.

1. Click one of the four blue “databasedemo” nodes 
2. and click the pencil button next to the “Server” box



1. enter the following database credentials (should match the database you created):

Host:

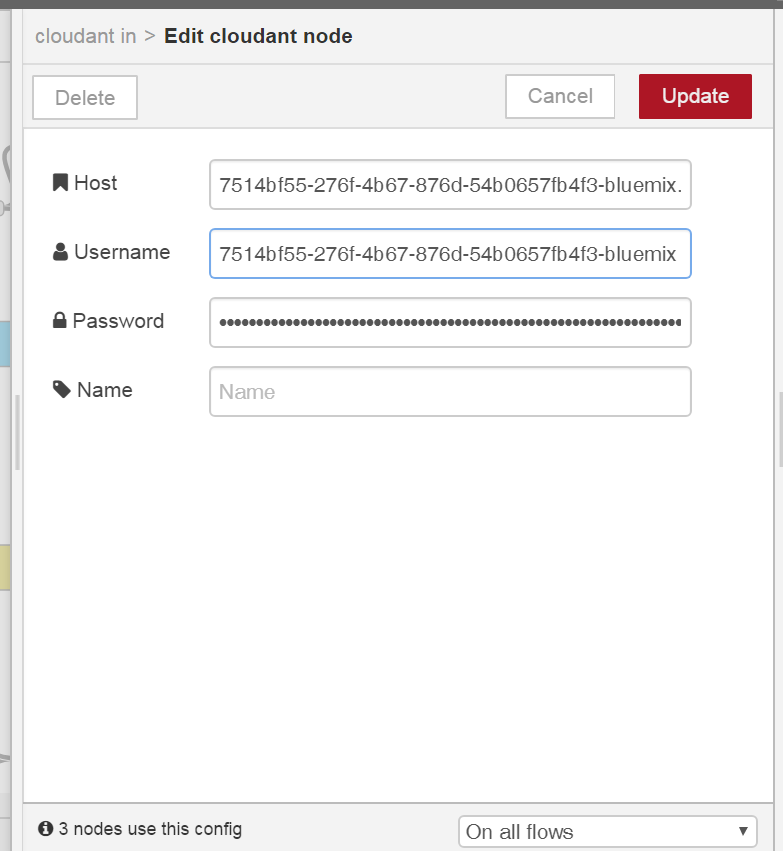
7514bf55-276f-4b67-876d-54b0657fb4f3-bluemix.cloudant.com

Username:

7514bf55-276f-4b67-876d-54b0657fb4f3-bluemix

Password:

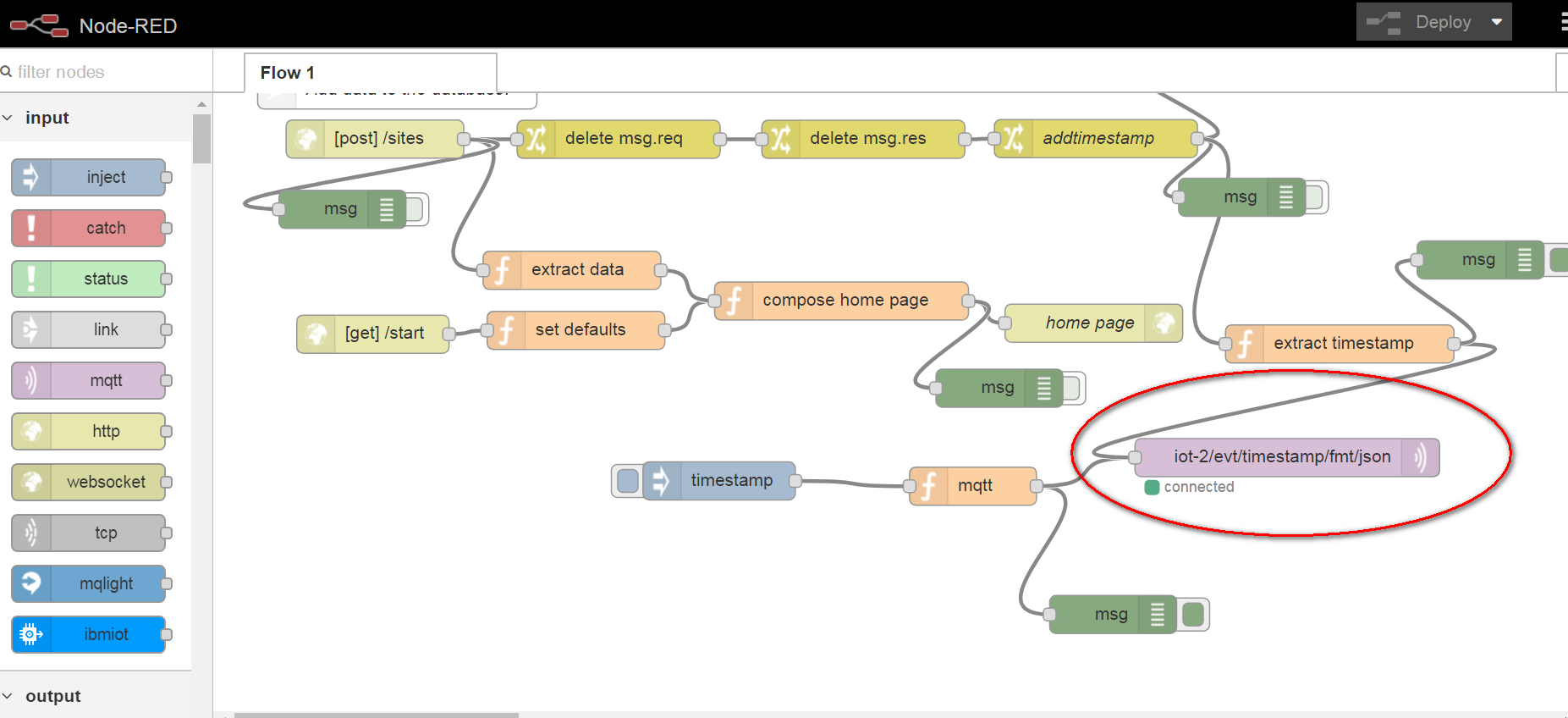
b824a67c1496ac11d286e81b5cfe4f5f27e6b61eb11972fca2650ddb8a1f64ec



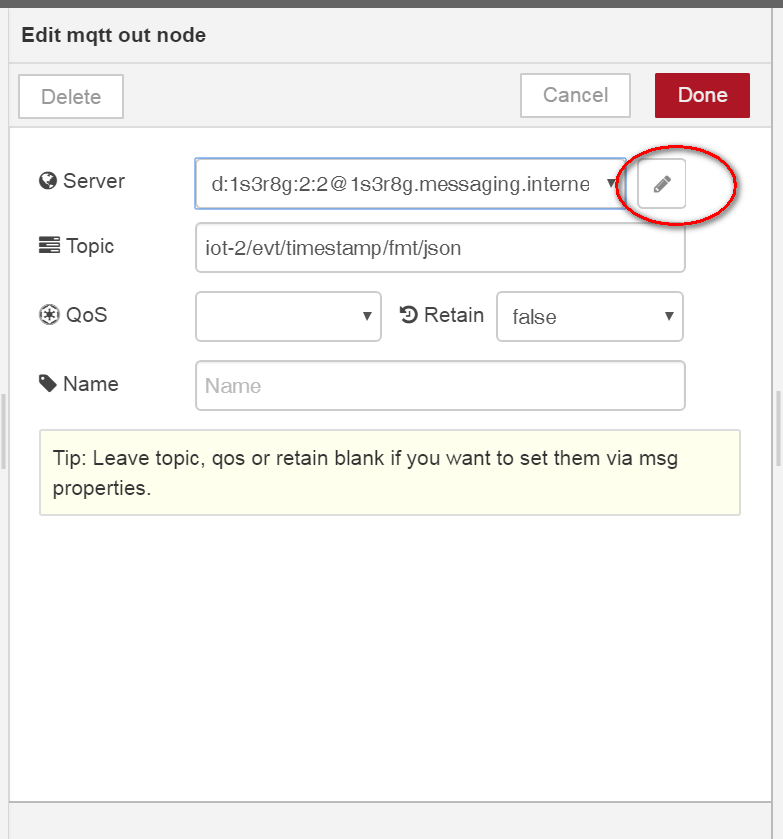
1. Click the “Update” button
2. Click the “Done” button
3. Repeat steps 15-18 for all the other three blue “databasedemo” nodes

## Configure the MQTT output node to send the live sensor data to the Watson MQTT broker

1. Click the purple “mqtt output” node



1. Click the edit button next to the “Server” box

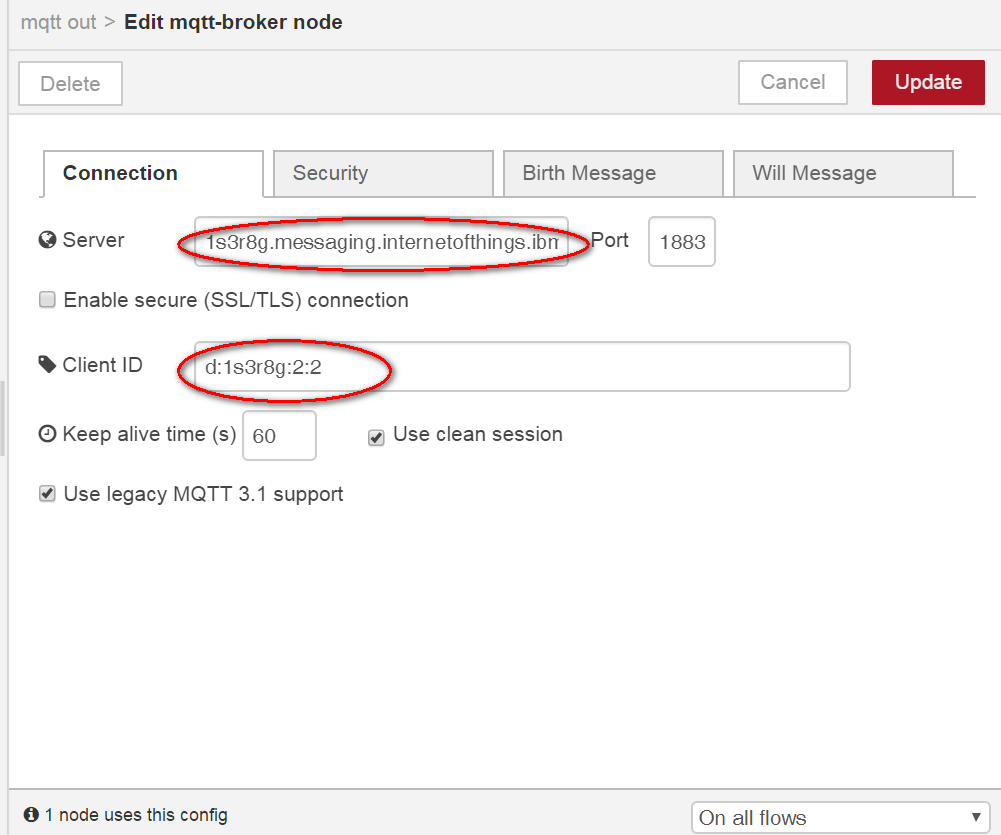


1. Enter the server name into the server box. It should be

*org\_id*.messaging.internetofthings.ibmcloud.com

where *org\_id* should be replace by the actual Organization ID of the device created above. In this example, the server name is

**1s3r8g.messaging.internetofthings.ibm**



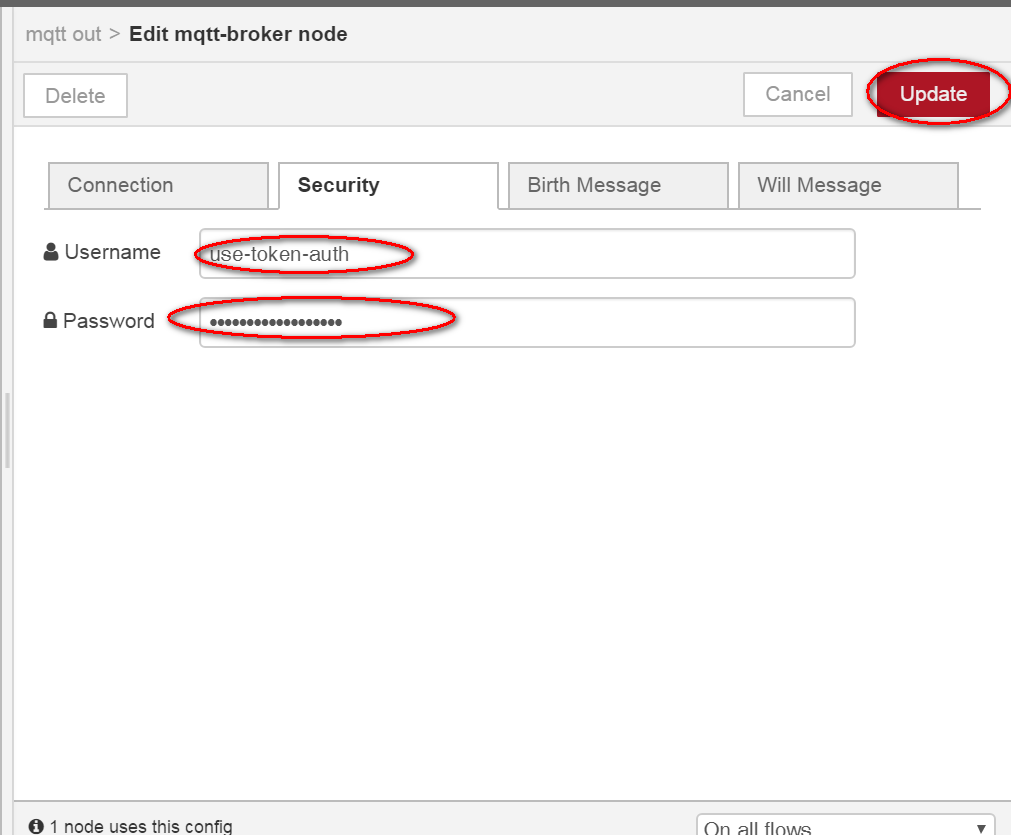
1. Enter the client ID into the Client ID box. It should be

d:*org\_id*:*device\_type*:*device\_id*

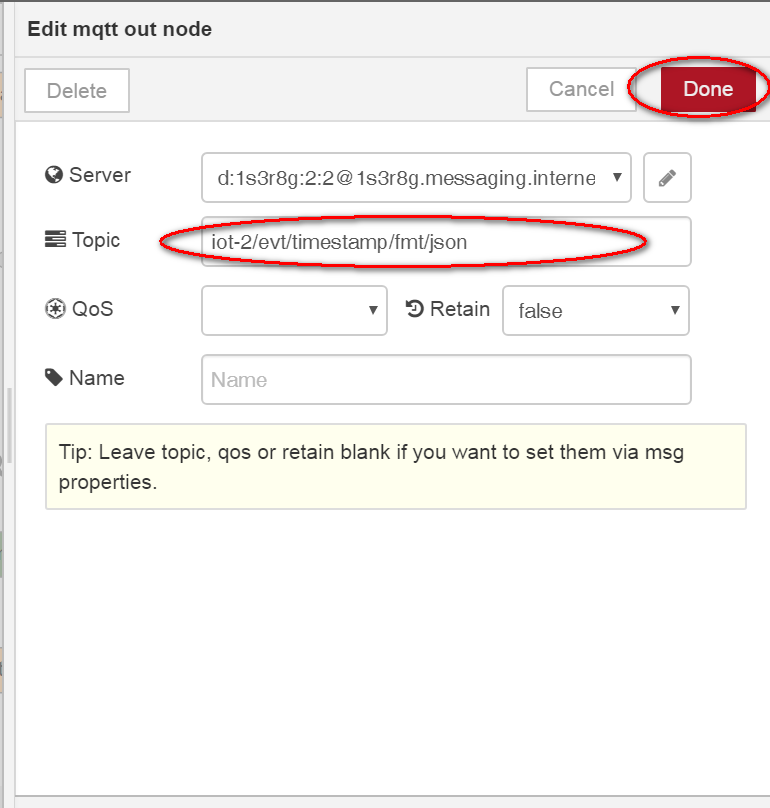
Where *org\_id* should be replaced by the actual Organization ID , *device\_type* by the actual device type, and *device\_id* by the actual device ID of the device created above. In this example the Client ID should be

**d:** **1s3r8g:2:2**

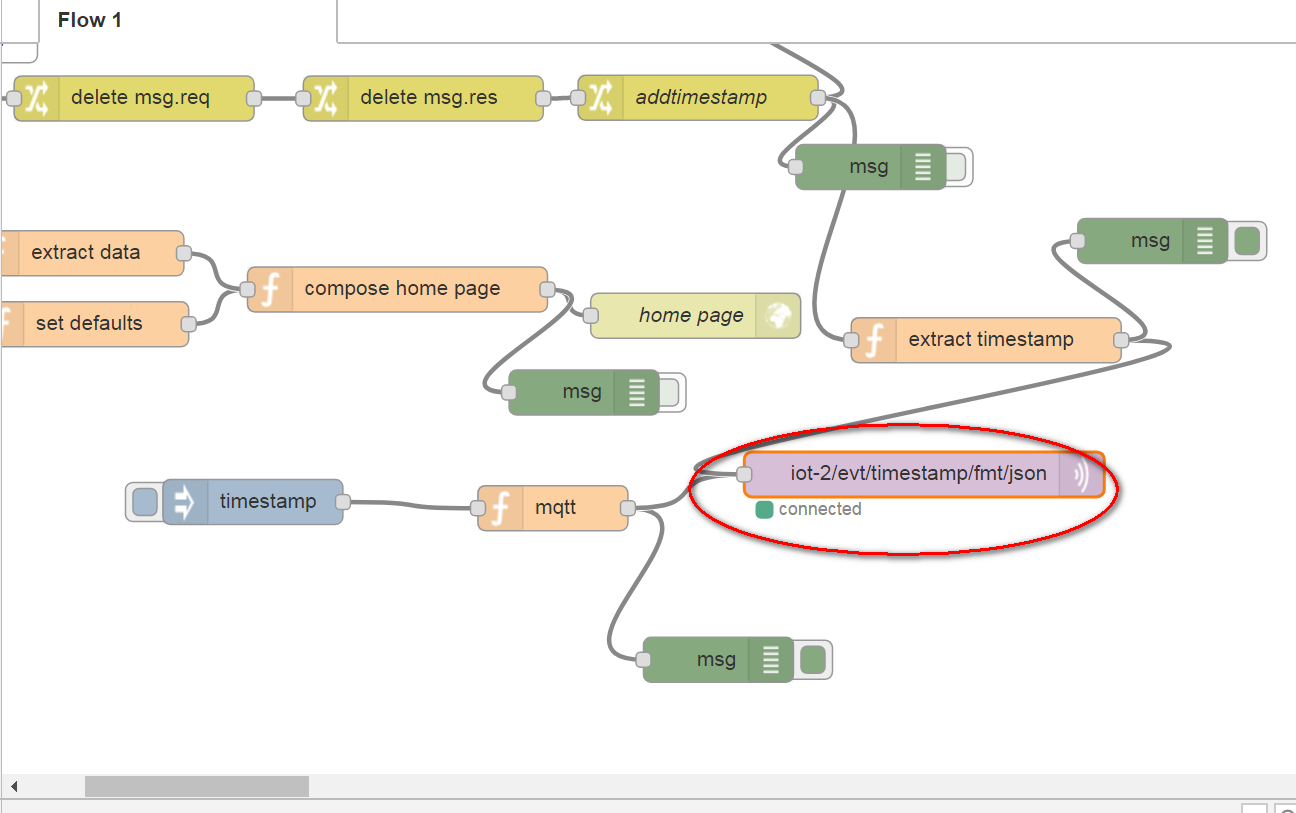
1. Click the security tab. Enter “use-token-auth” into the Username box and the Authentication Token (tzR?mz\_SM9Nrpcutl9 in this example) of the device into the Password box

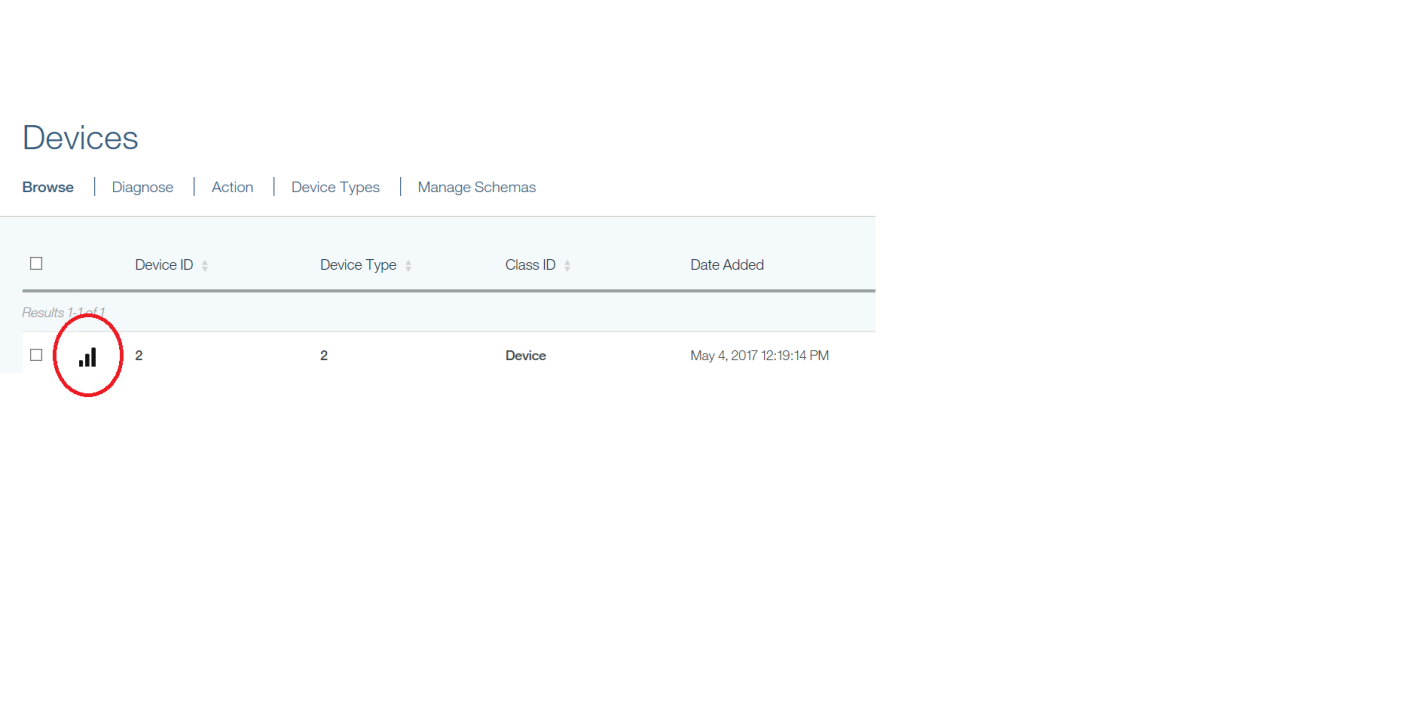


1. Click the “Update” button
2. Enter iot-2/evt/timestamp/fmt/json into the Topic box



1. Click the “Done” button
2. Click the “Deploy” button
3. You should see the “mqtt” node displays the “connected” status. And the device also shows the same:





## Modify the web app on the I3 Gateway to send data to the Bluemix I3 app

1. Modify the cloud.json to send data to the I3 app:

**If you need a webproxy:**

“

{

"host":"webproxy.ext.ti.com",

"port":80,

"path":"http://fengmoiot2.mybluemix.net/sites",

"data":{

"name":"I3-Kilby",

"address":"http://i3-kilby.sc.ti.com/",

"gps":[ 32.925167, -96.757725 ],

"color":"#FF0000"

}

}

“

Modify the highlighted parts to be the actual webproxy server address, the application name, and a unique gateway name. The gateway name is used to identify data from each gateway when there are multiple gateways sending data.

**If you don’t need a webproxy:**

“

{

"host":"fengmoiot2.mybluemix.net",

"port":80,

"path":"/sites"

"data":{

"name":"I3-Kilby",

"address":"http://i3-kilby.sc.ti.com/",

"gps":[ 32.925167, -96.757725 ],

"color":"#FF0000"

}

}

“

Again, modify the highlighted parts to be the actual application name, and a unique gateway name. The gateway name is used to identify data from each gateway when there are multiple gateways sending data.

1. Modify the app.js to send the data when available

Add two global variables:

“

//================================================================================

// MODULES AND GLOBALS

//================================================================================

**…**

**var** global\_data = {};

**var** newSensorData = 0;

“

Add the code:

“

**function obs\_parse(id,data){**

**…**

**ws.broadcast(JSON.stringify(msg));**

**global\_data = {type:"sensor\_type\_0", gateway\_0:cloud.data, sensor\_0: {\_id:id, data:obj\_data}};**

**newSensorData = 1;**

**sendDataToCloud();**

**}**

“

Change the last part of the code:

“

setInterval(**function**(){

**if**(cloud!=**null**&&

cloud.host!=**null**&&

cloud.port!=**null**&&

cloud.path!=**null**&&

cloud.data!=**null**&&

cloud.data.name!="FIXME"

){

…

);

“

**to**

“

**function sendDataToCloud()**

**{**

**if(cloud!=null&&**

**cloud.host!=null&&**

**cloud.port!=null&&**

**cloud.path!=null&&**

**cloud.data!=null&&**

**cloud.data.name!="FIXME"**

**){**

**if (newSensorData && !isEmpty(global\_data))**

**{**

**var options =**

**{**

**hostname: cloud.host,**

**path: cloud.path,**

**port: cloud.port,**

**method: 'POST',**

**headers:**

**{**

**'Content-Type': 'application/json',**

**}**

**};**

**var req = http.request(options, function(res)**

**{**

**res.on('data', function (body) {});**

**});**

**req.on('error', function(e) { });**

**// write data to request body**

**//req.write(JSON.stringify(cloud.data));**

**req.write(JSON.stringify(global\_data));**

**newSensorData = 0;**

**console.log("Post data:"+cloud.path);**

**req.end();**

**}**

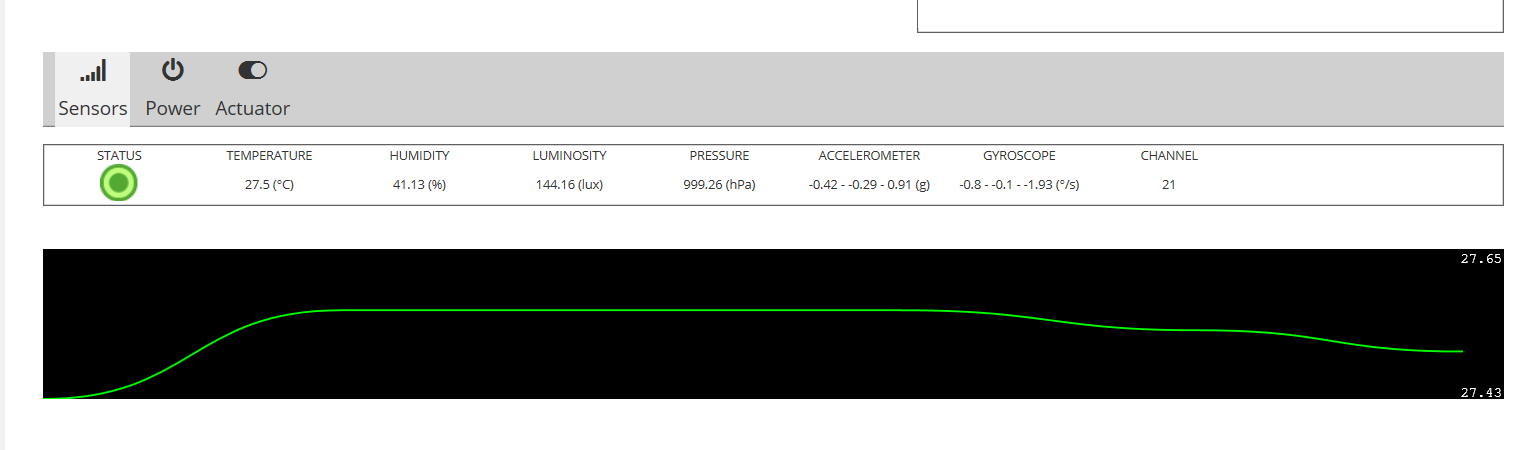
**}**

**}**

**//setInterval(sendDataToCloud,5000);**

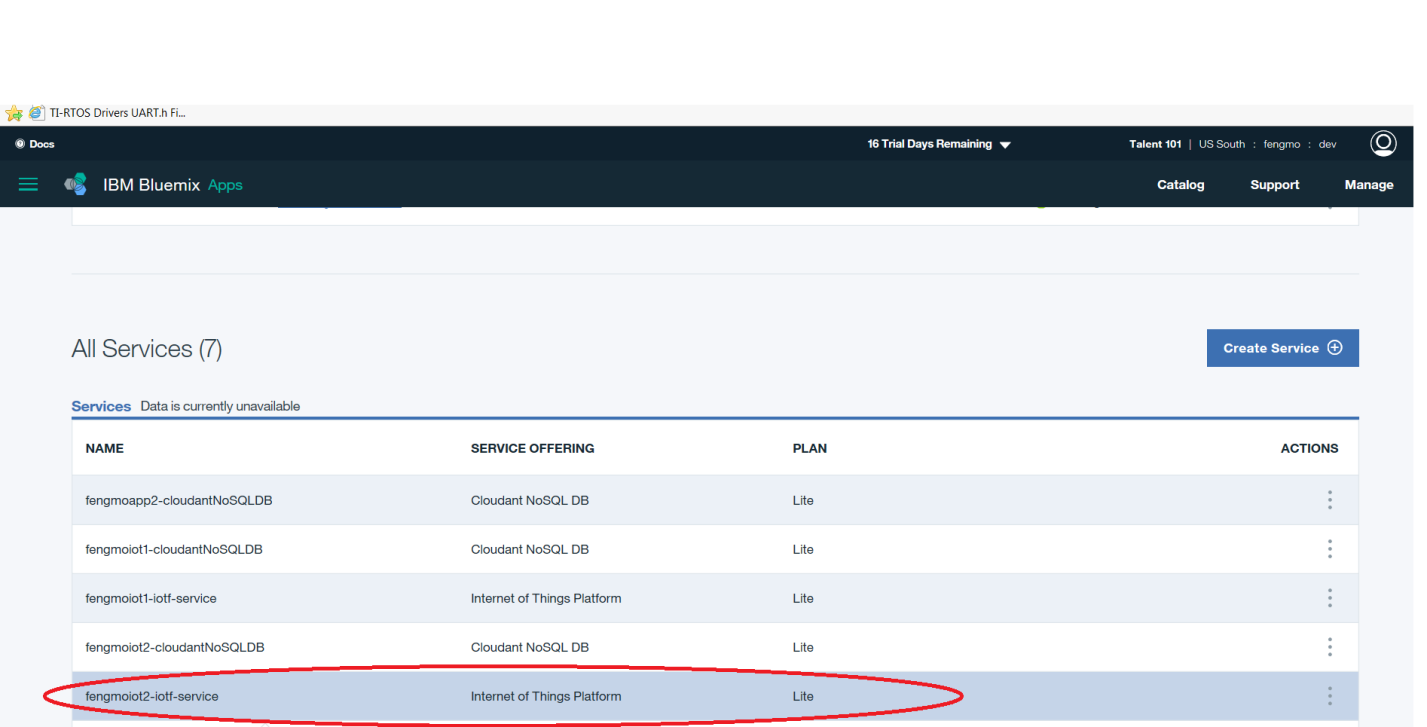
“

1. On the web app screen, select a node to send data. The data is passed on to the Bluemix I3 as well.



## Display live sensor data on the IBM Watson Cards

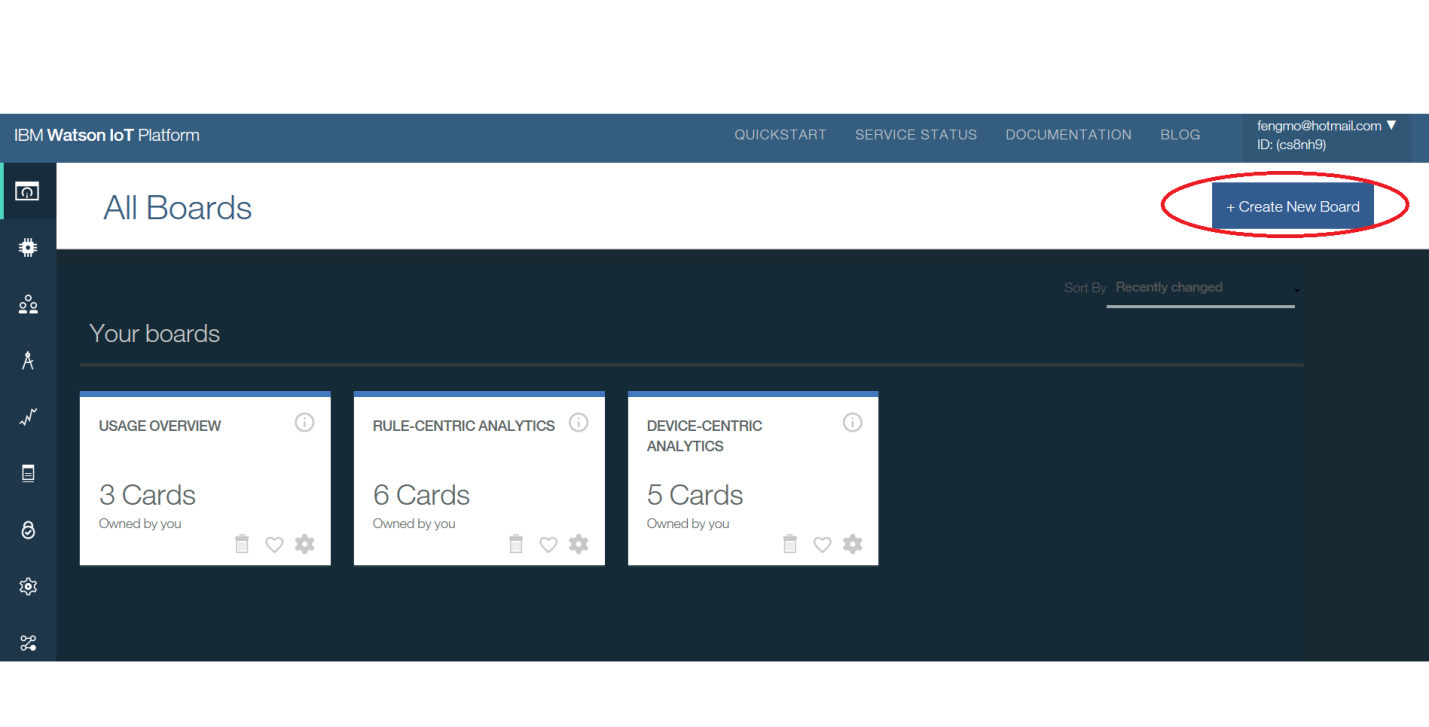
1. Goto the Bluemix Dashboard and click the IOT service created with the app



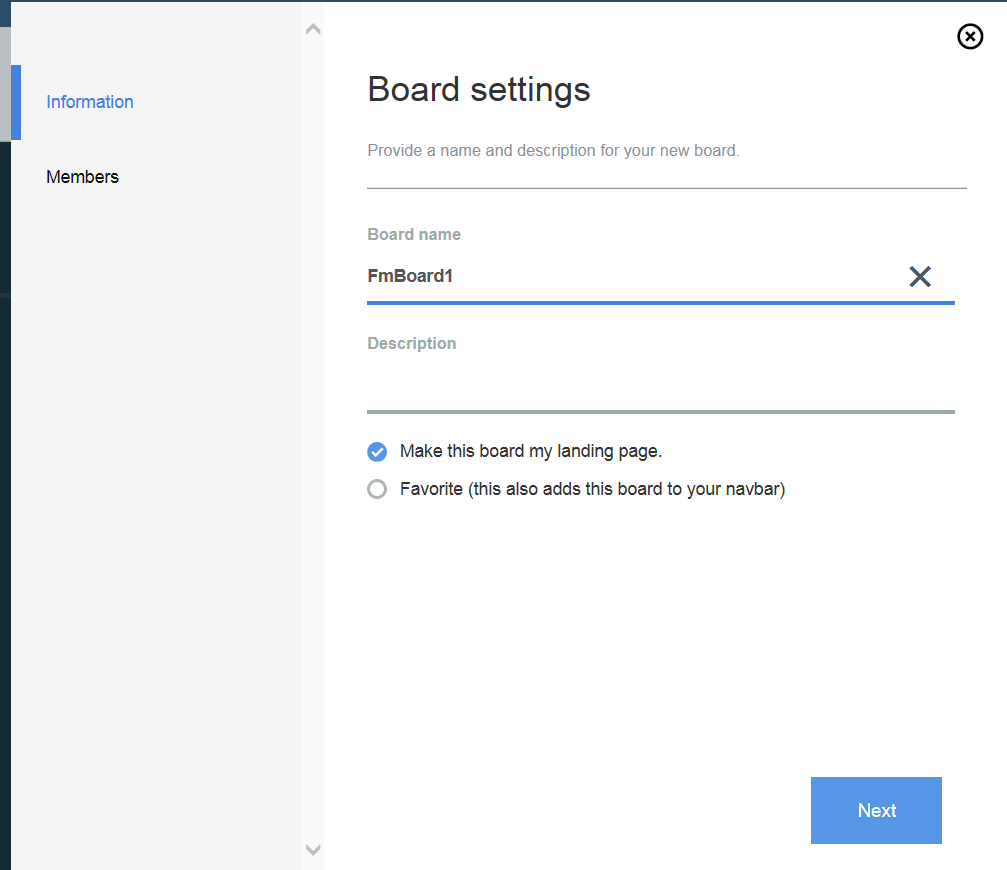
1. After a pause, you should see this screen. Click the “Launch” button



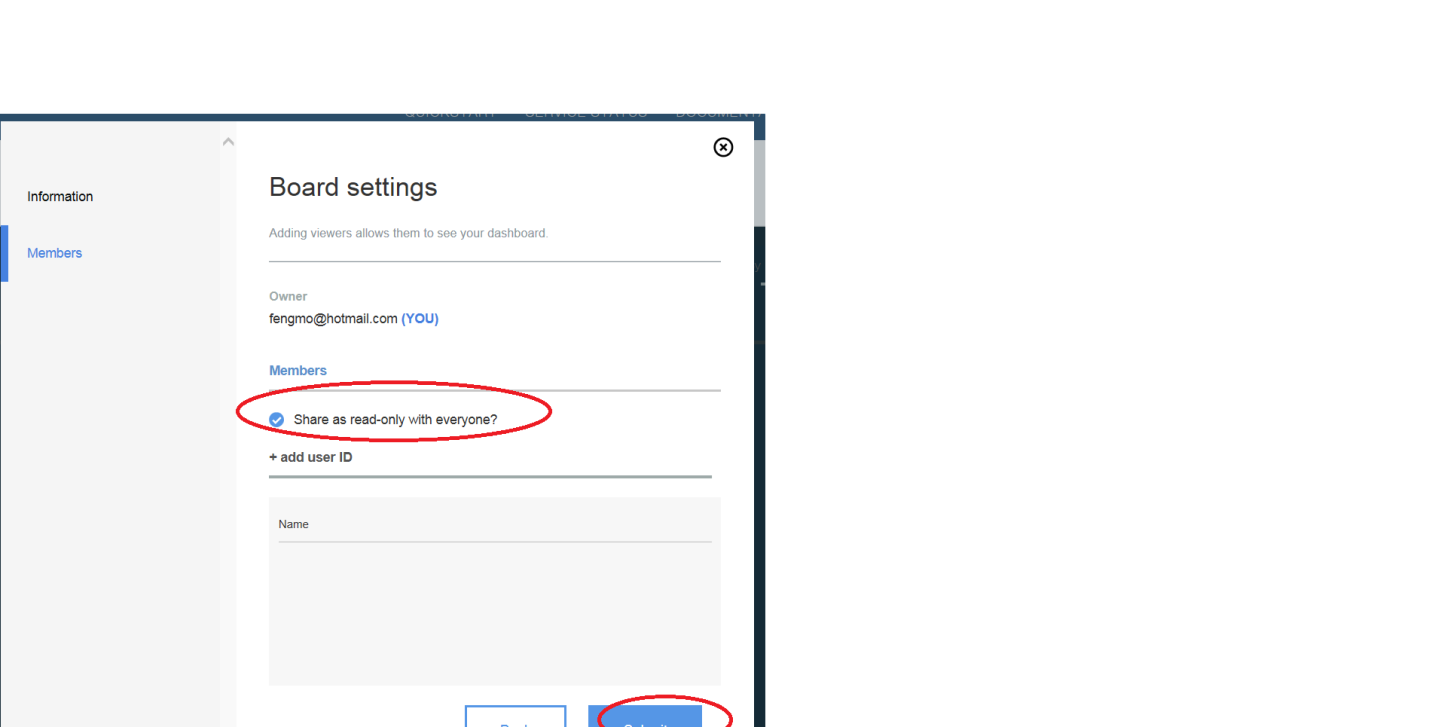
1. Click the “Create New Board” button



1. Give a name to the new board and click “Next”



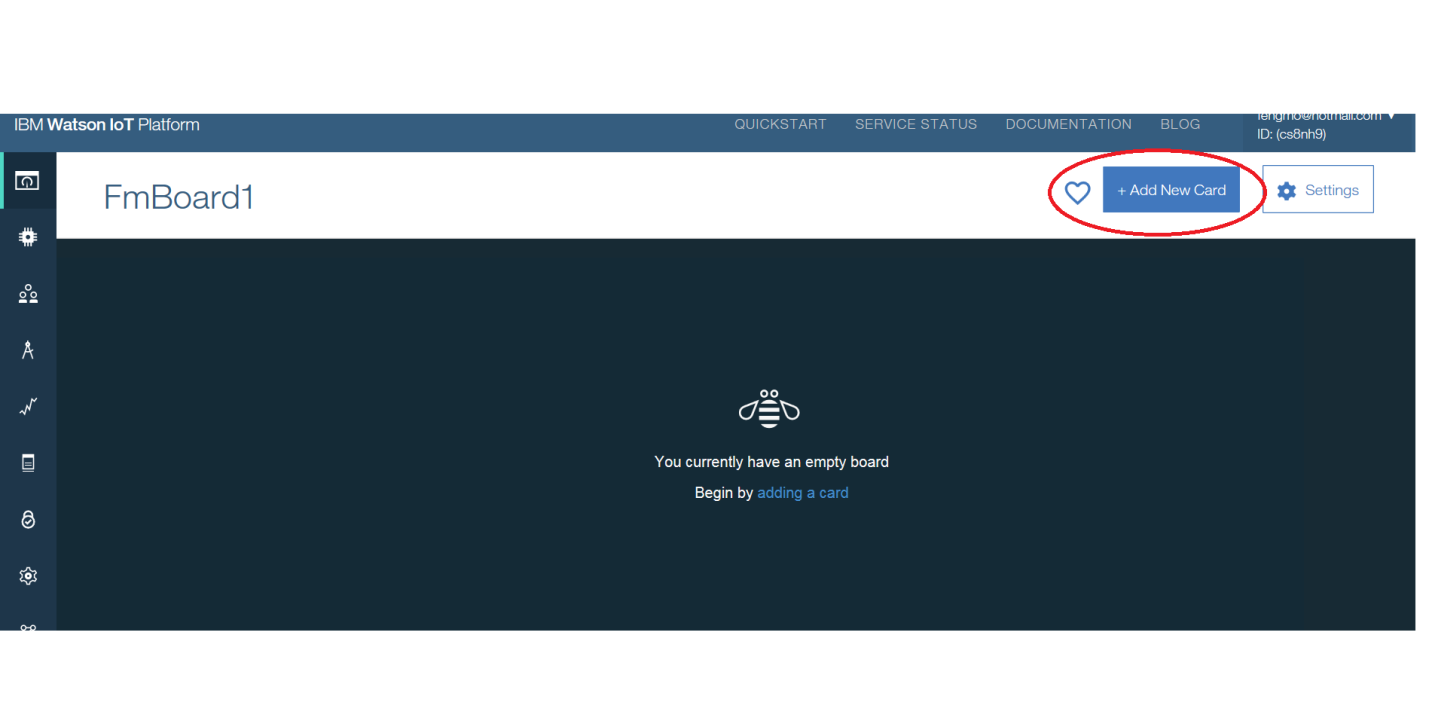
1. Check the share with every one box and click the “Submit” button



1. Click the newly create board



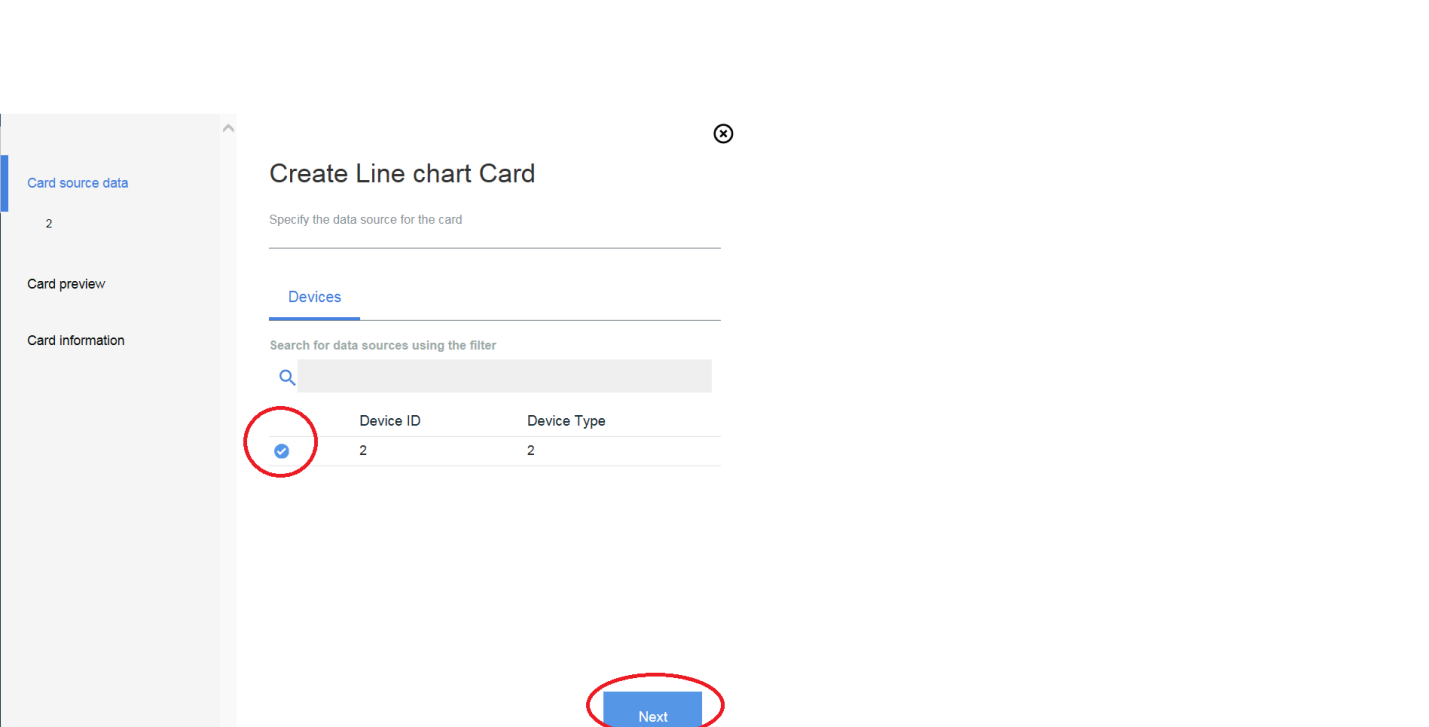
1. Click the “Add New Card” button



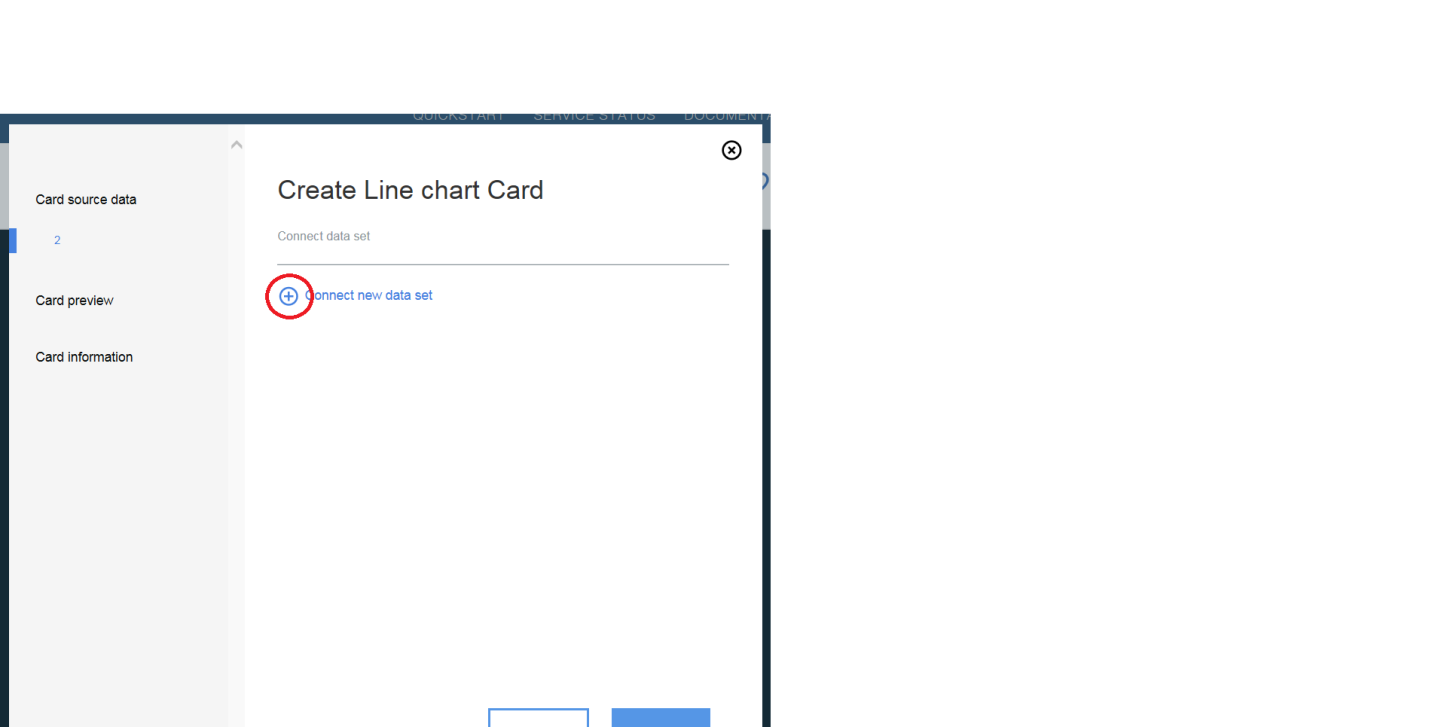
1. Select Line chart



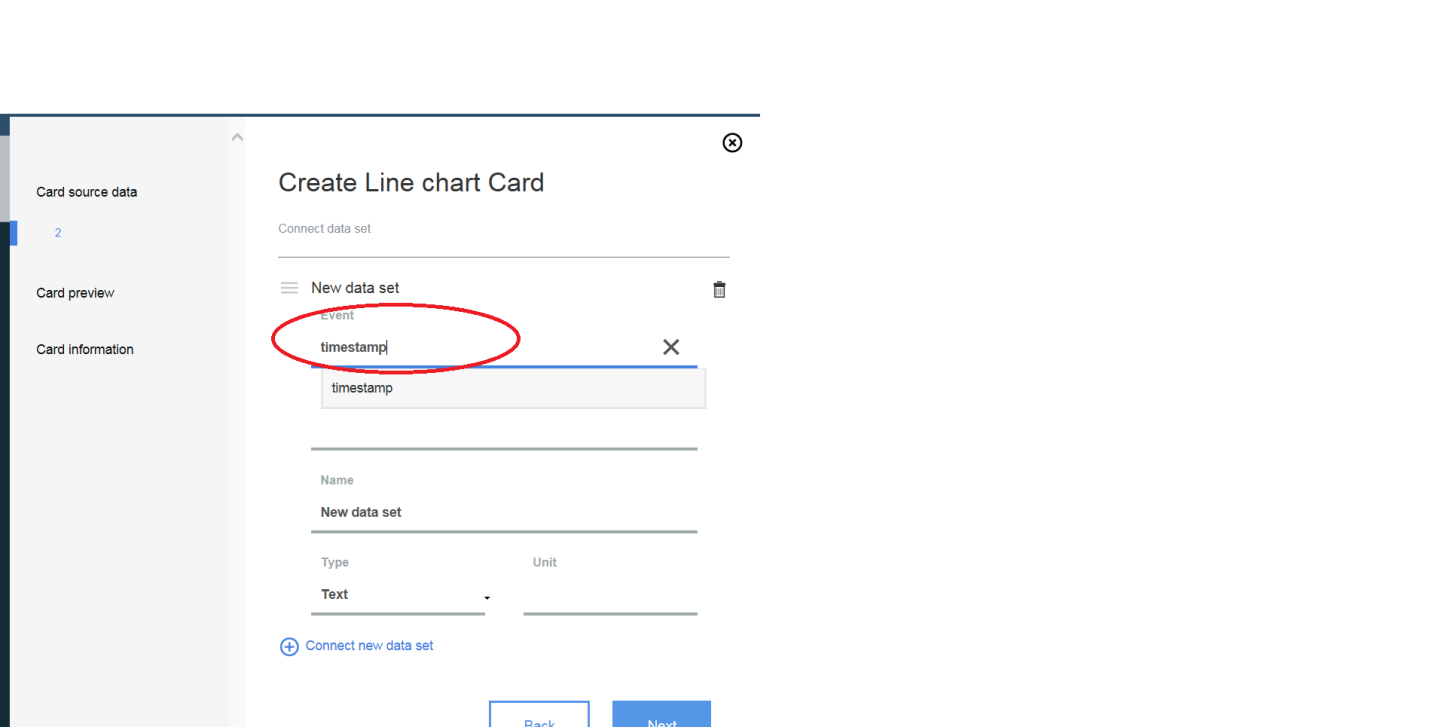
1. Check the device button and click the “Next” button



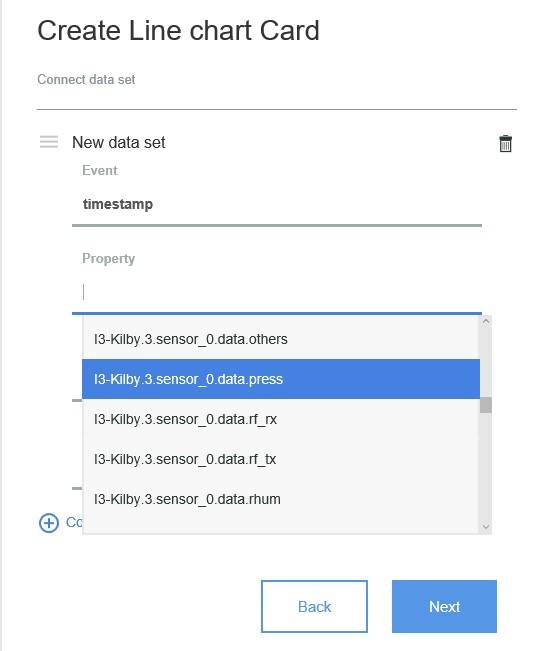
1. Click and expand the button “Connect new data set” button



1. Select the event



1. Select the pressure in Property

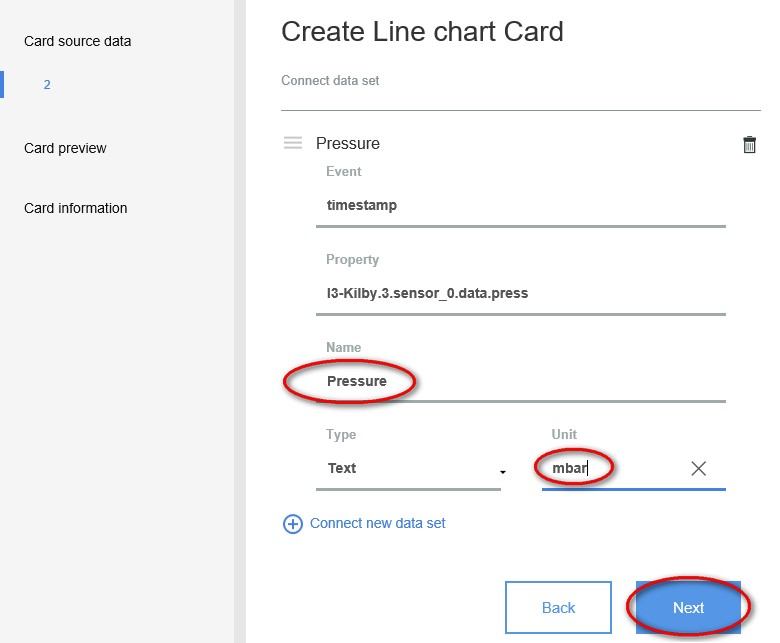


The sensor data name has the gateway name and sensor ID as the prefix:

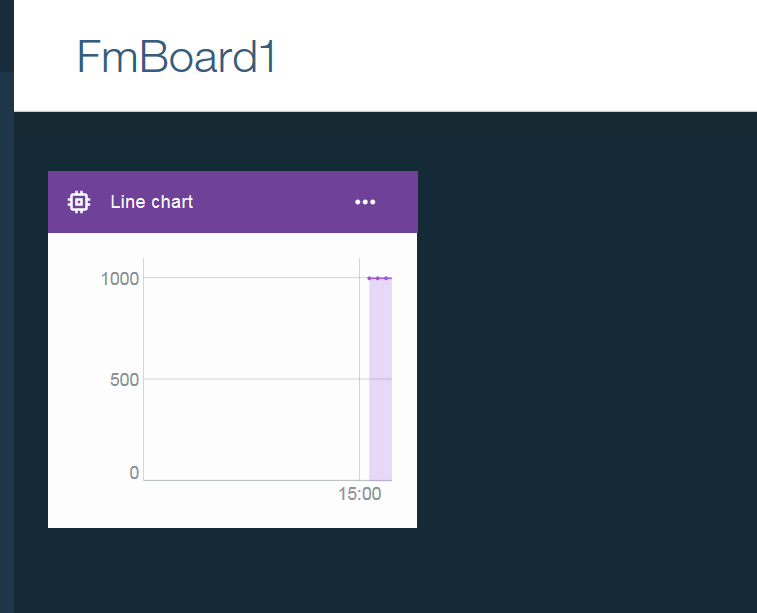
**gatawayname.sensor\_id.**

In the current example, the gateway name is **I3-Kilby**, the sensor ID is **3**.

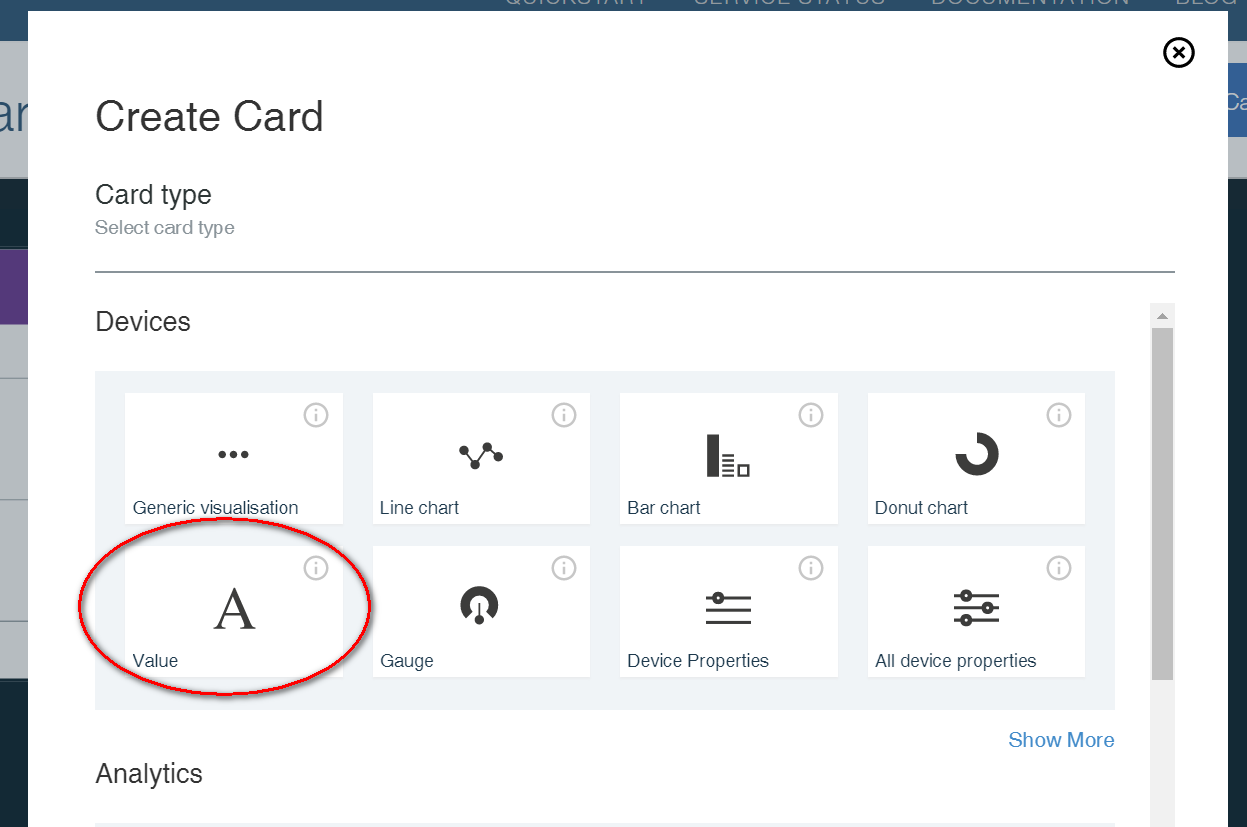
1. Set Name to “Pressure”, Unit to “mbar”, and click the “Next” button



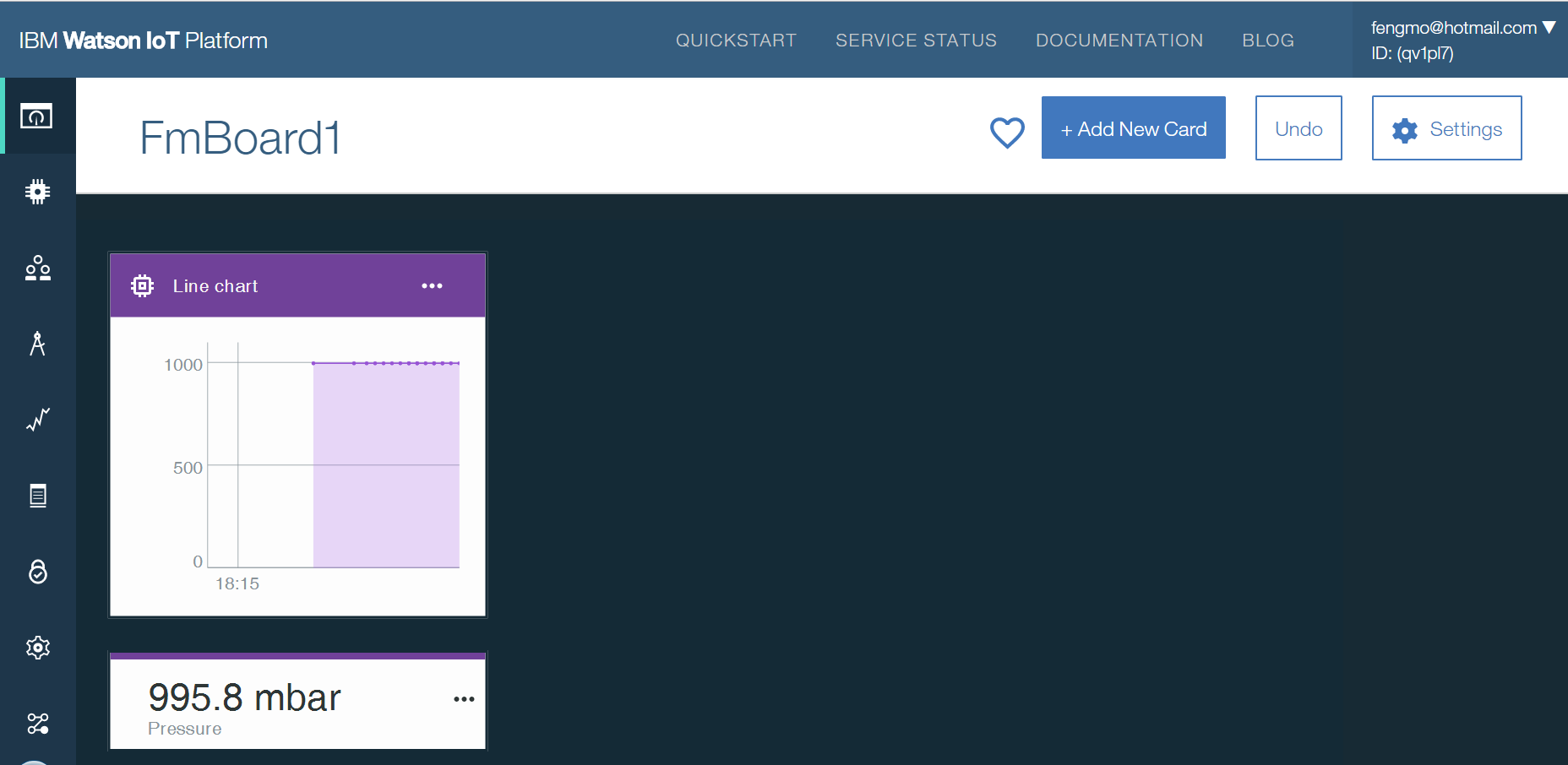
1. Click “Next”
2. Click “Submit”
3. You should see the live display of the pressure data from the sensor



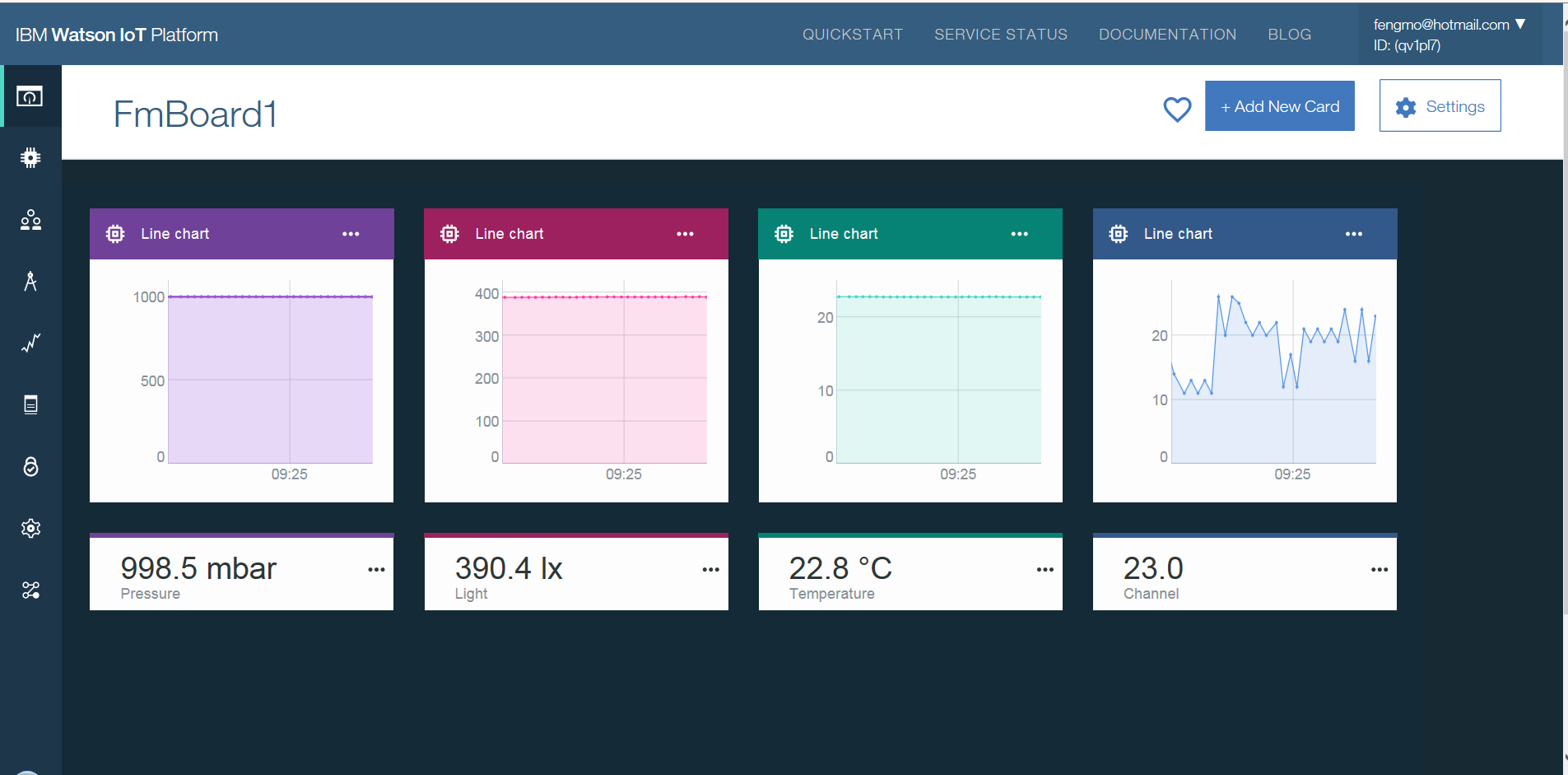
1. Add a second Value type card for display the current pressure sensor reading



1. Repeat the same steps as for the first card. You should see the second card:



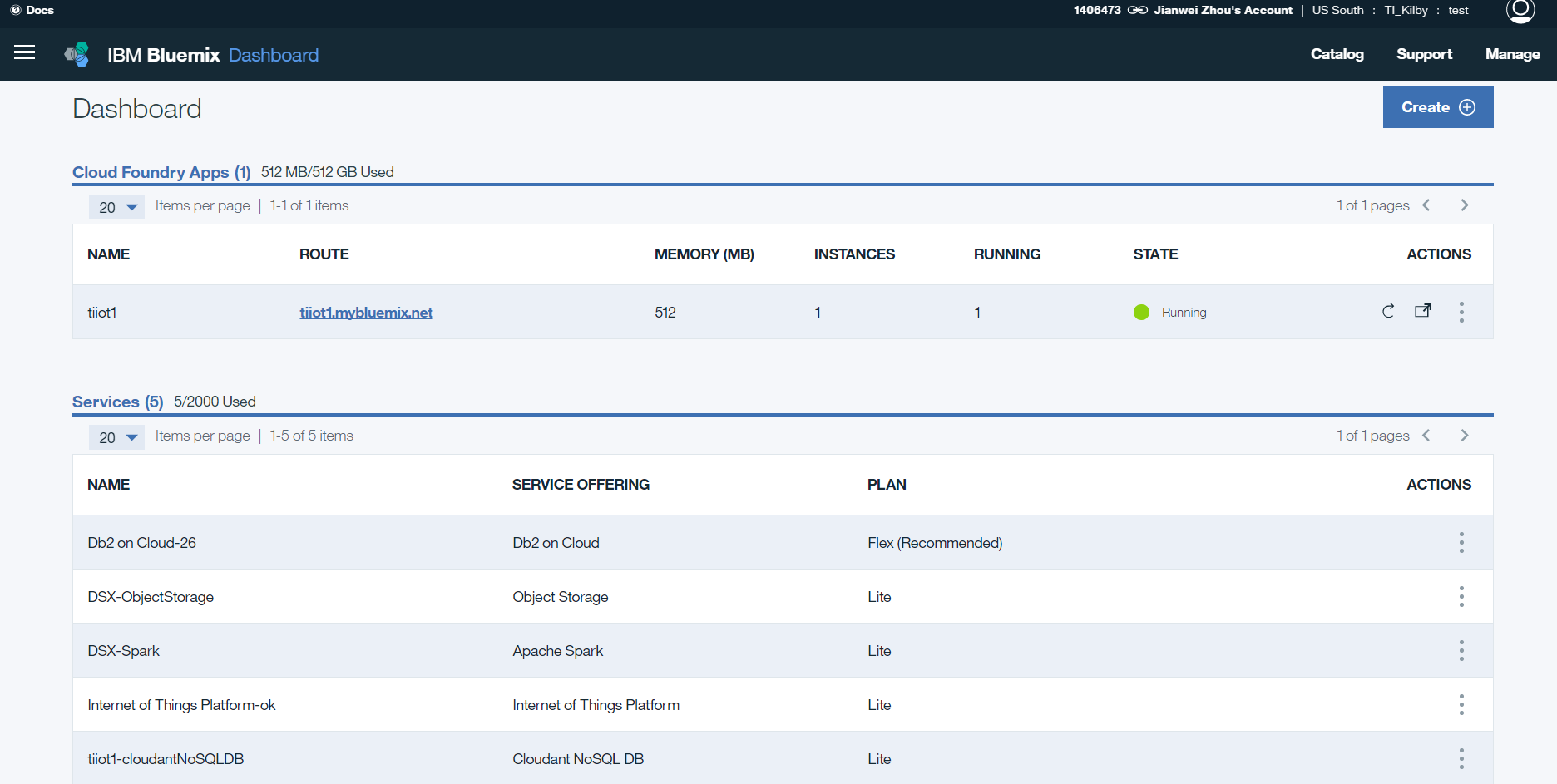
Here is an example with multiple displays



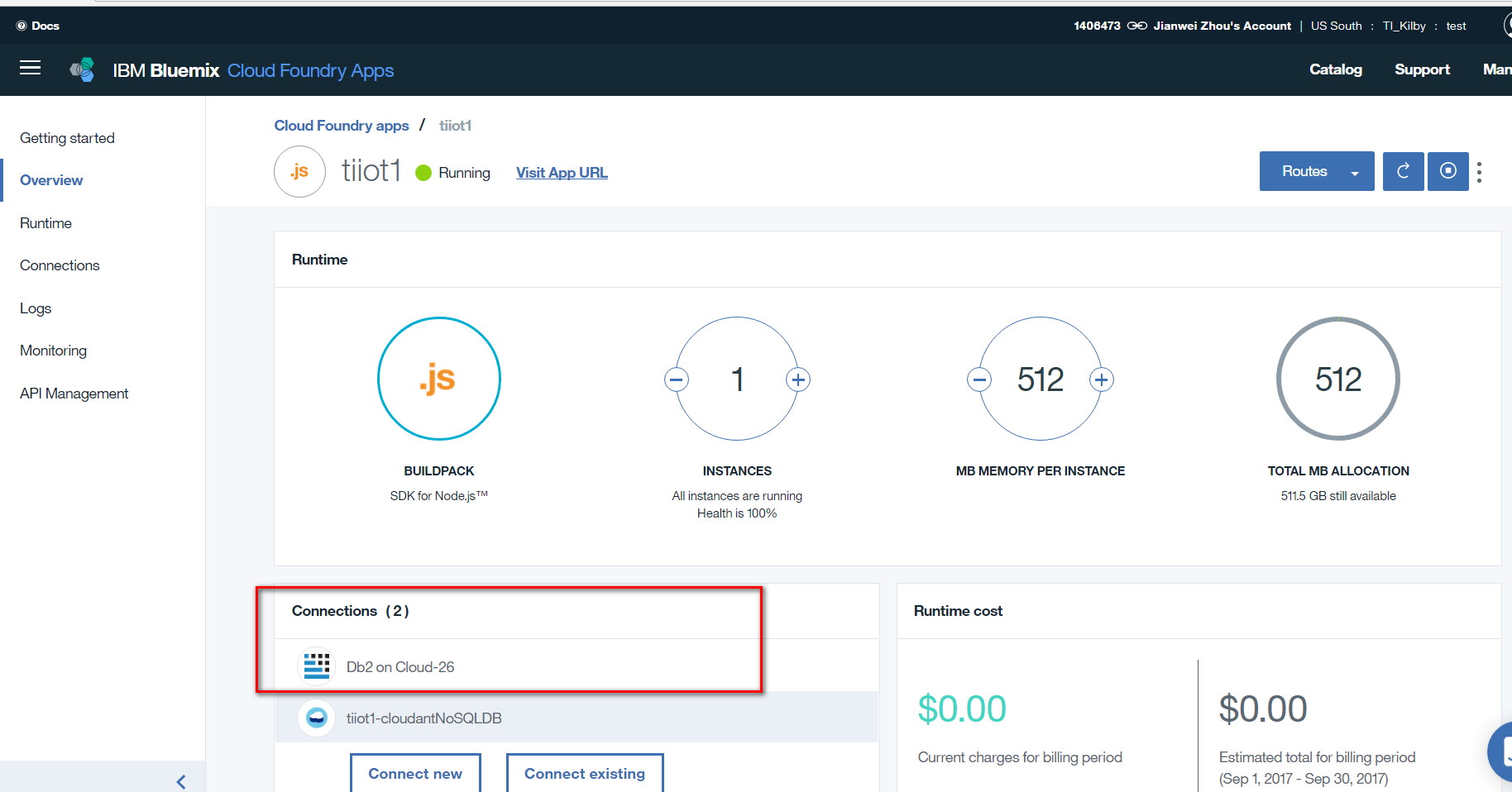
# Working with the SQL database

Make sure use the Chrome browser.

## Create a new “Db2 on Cloud-26 service” in the Dashboard

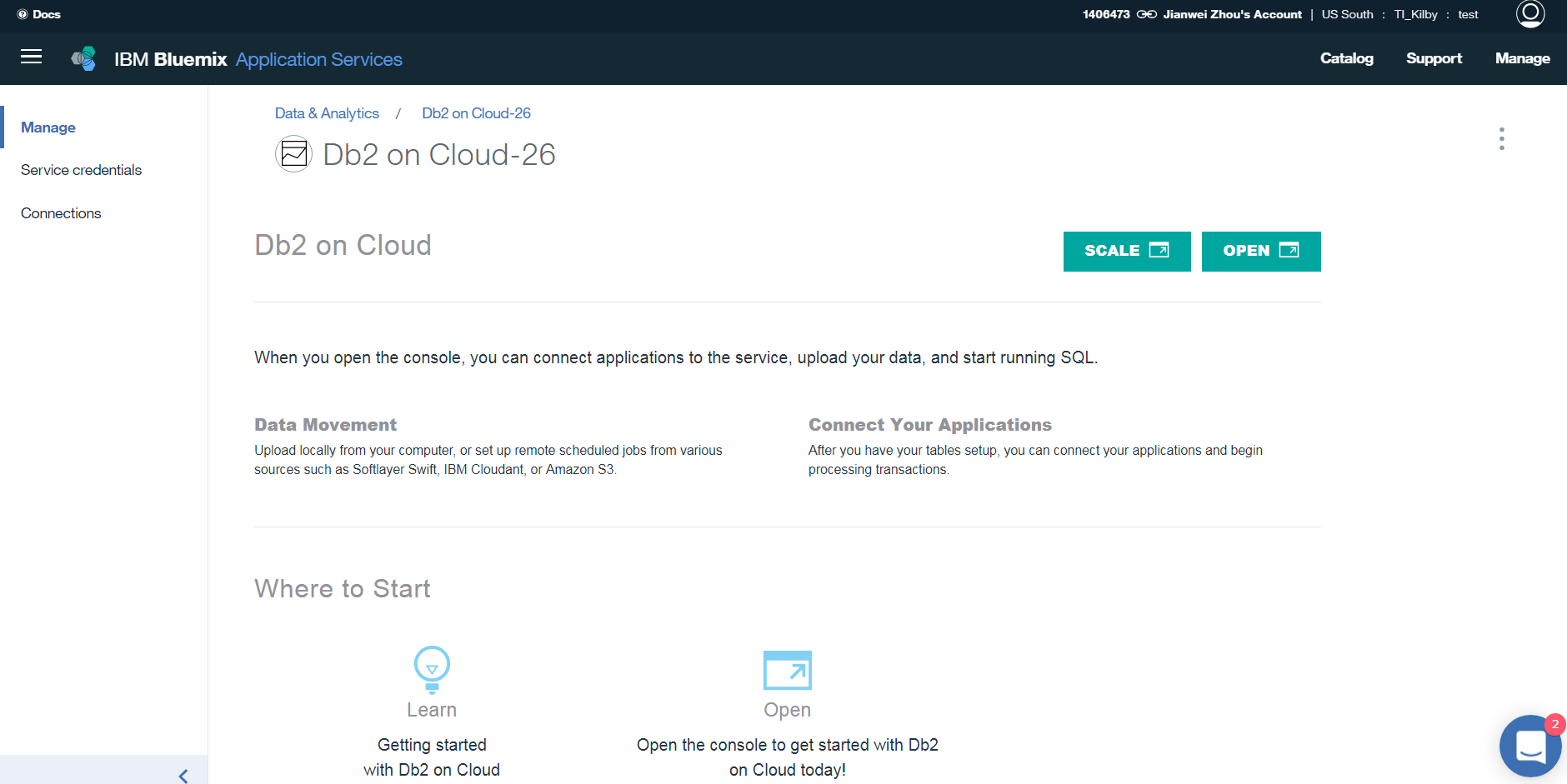


## Create a connection to the Db2 service from the application

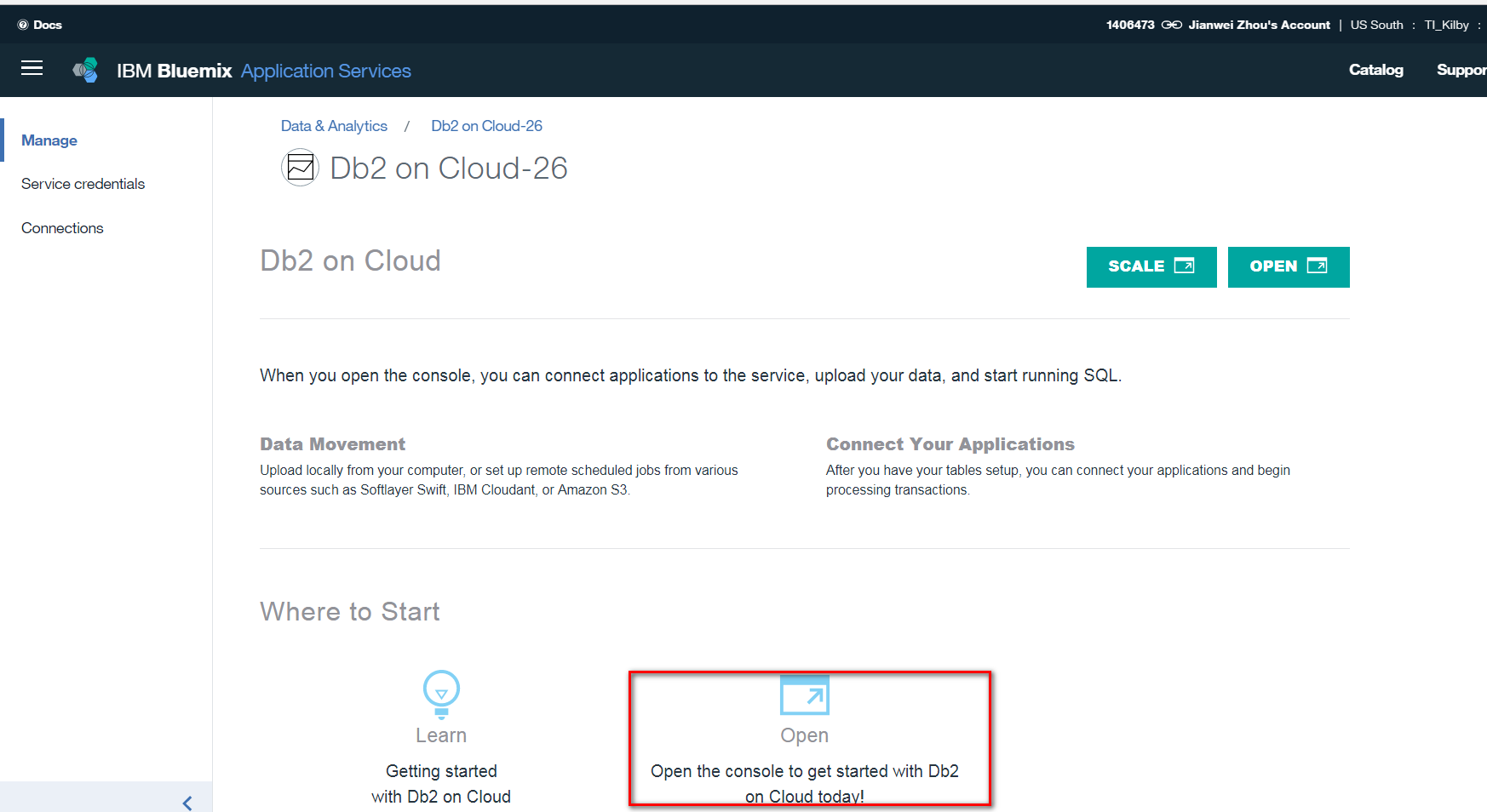


## Create the databases

Go back to the Dashboard and click the Db2 service

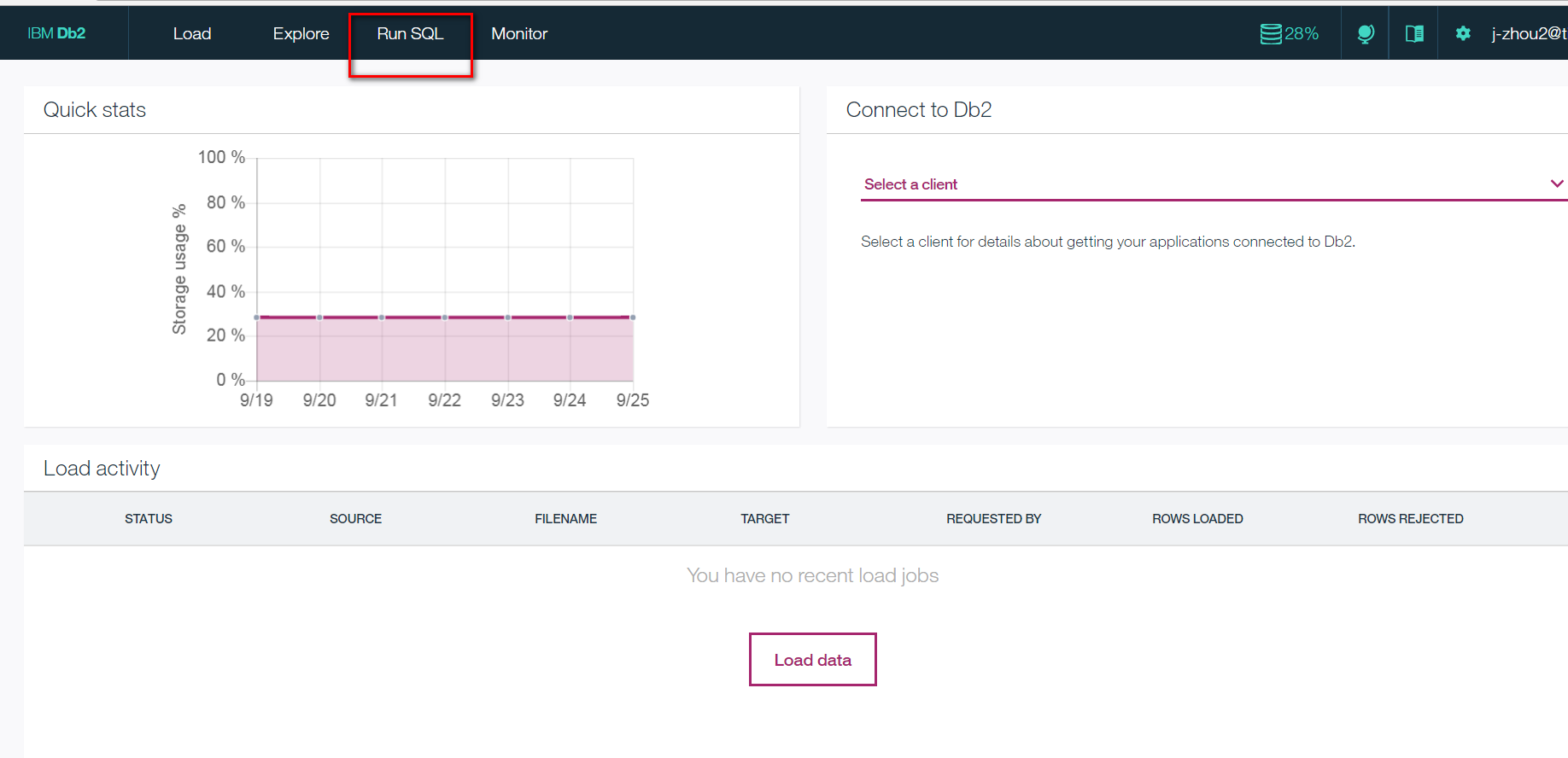


Click “Open”



(END)

Click the “Run SQL”



### Create the first SQL database ("I3 SENSOR DATA")

This database contains all the sensor data.

Paste the following data into the “SQL editor” box

“

CREATE TABLE "I3 SENSOR DATA"

(

"TIMESTAMP" DOUBLE PRECISION,

"DATE" CHAR(10),

"TIME" CHAR(8),

"GATEWAY\_NAME" VARCHAR(64),

"GATEWAY\_GPS\_LAT" FLOAT,

"GATEWAY\_GPS\_LONG" FLOAT,

"SENSOR\_ID" VARCHAR(32),

"ACCELX" FLOAT,

"ACCELY" FLOAT,

"ACCELZ" FLOAT,

"BAT" FLOAT,

"CC2650\_ACTIVE" FLOAT,

"CC2650\_SLEEP" FLOAT,

"CHANNEL" INT,

"EH" FLOAT,

"EH1" FLOAT,

"GPSEN\_ACTIVE" FLOAT,

"GPSEN\_SLEEP" FLOAT,

"GYROX" FLOAT,

"GYROY" FLOAT,

"GYROZ" FLOAT,

"LED" FLOAT,

"LUX" FLOAT,

"MSP432\_ACTIVE" FLOAT,

"MSP432\_SLEEP" FLOAT,

"PRESS" FLOAT,

"RF\_RX" FLOAT,

"RF\_TX" FLOAT,

"RHUM" FLOAT,

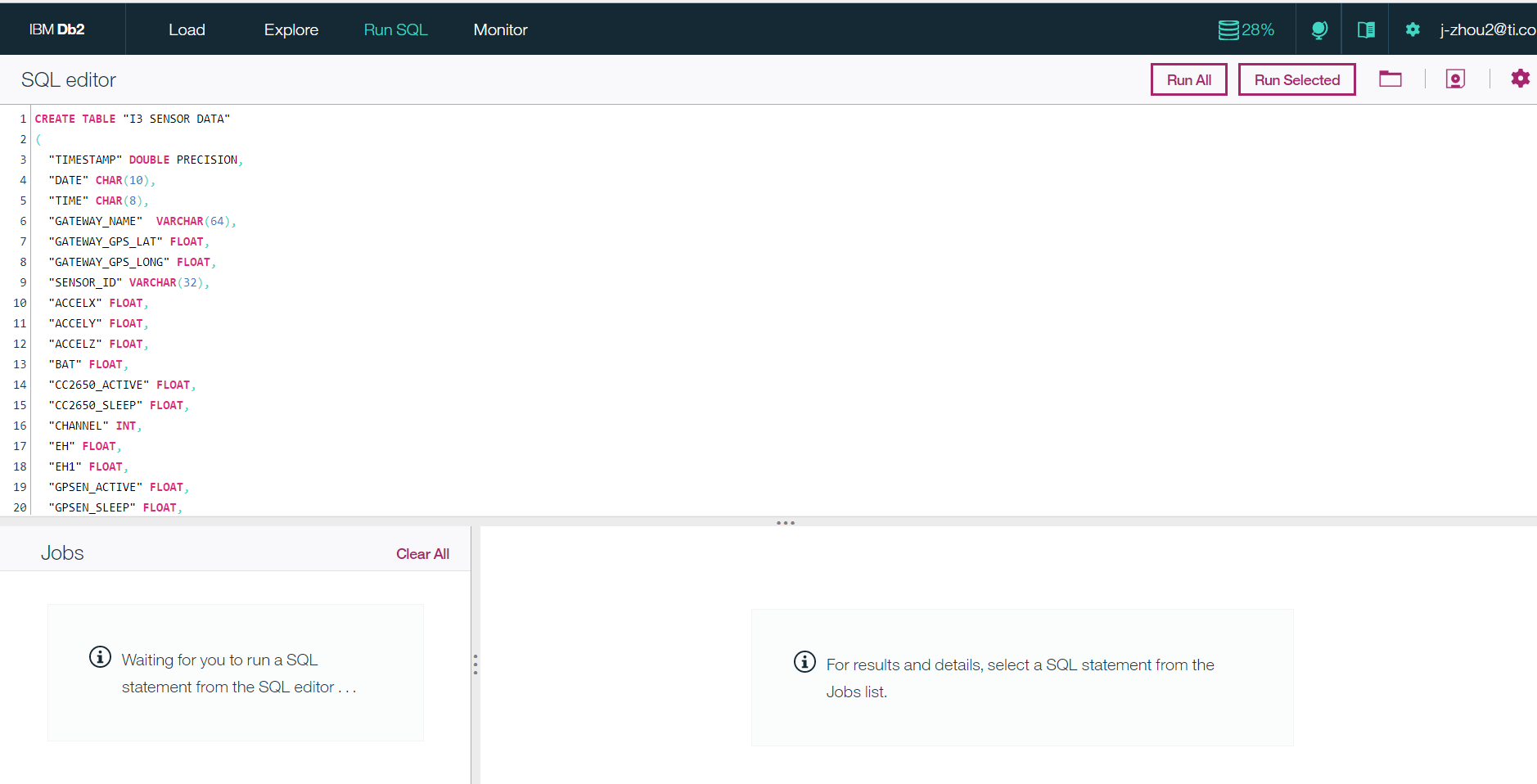
"SSM\_ACTIVE" FLOAT,

"SSM\_SLEEP" FLOAT,

"TAMB" FLOAT

);

“



Click the “Run All” button.

### Create the second SQL database ("I3 PERF DATA")

This database contains all the sensor performance measurements.

Paste the following data into the “SQL editor” box

“

CREATE TABLE "I3 PERF DATA"

(

"TIMESTAMP" DOUBLE PRECISION,

"DATE" CHAR(10),

"TIME" CHAR(8),

"GATEWAY\_NAME" VARCHAR(64),

"GATEWAY\_GPS\_LAT" FLOAT,

"GATEWAY\_GPS\_LONG" FLOAT,

"SENSOR\_ID" VARCHAR(32),

"NUMTXFAIL" INT,

"NUMTXTOTAL" INT,

"CURPARENT" INT,

"NUMPARENTCHAN" INT,

"NUMSYNCLOST" INT,

"AVGDRIFT" INT,

"MAXDRIFT" INT,

"AVGBCNRSSI" INT,

"AVGACKRSSI" INT,

"NUMCANDIDATEP" INT,

"NUMBEACONLOST" INT,

"NUMBEACONTOTA" INT,

"TIMEASSOCIATI" INT,

"NUMOUTOFBUFFE" INT,

"BEACON\_INFO" VARCHAR(256),

"TS" DOUBLE PRECISION,

"RTT" FLOAT,

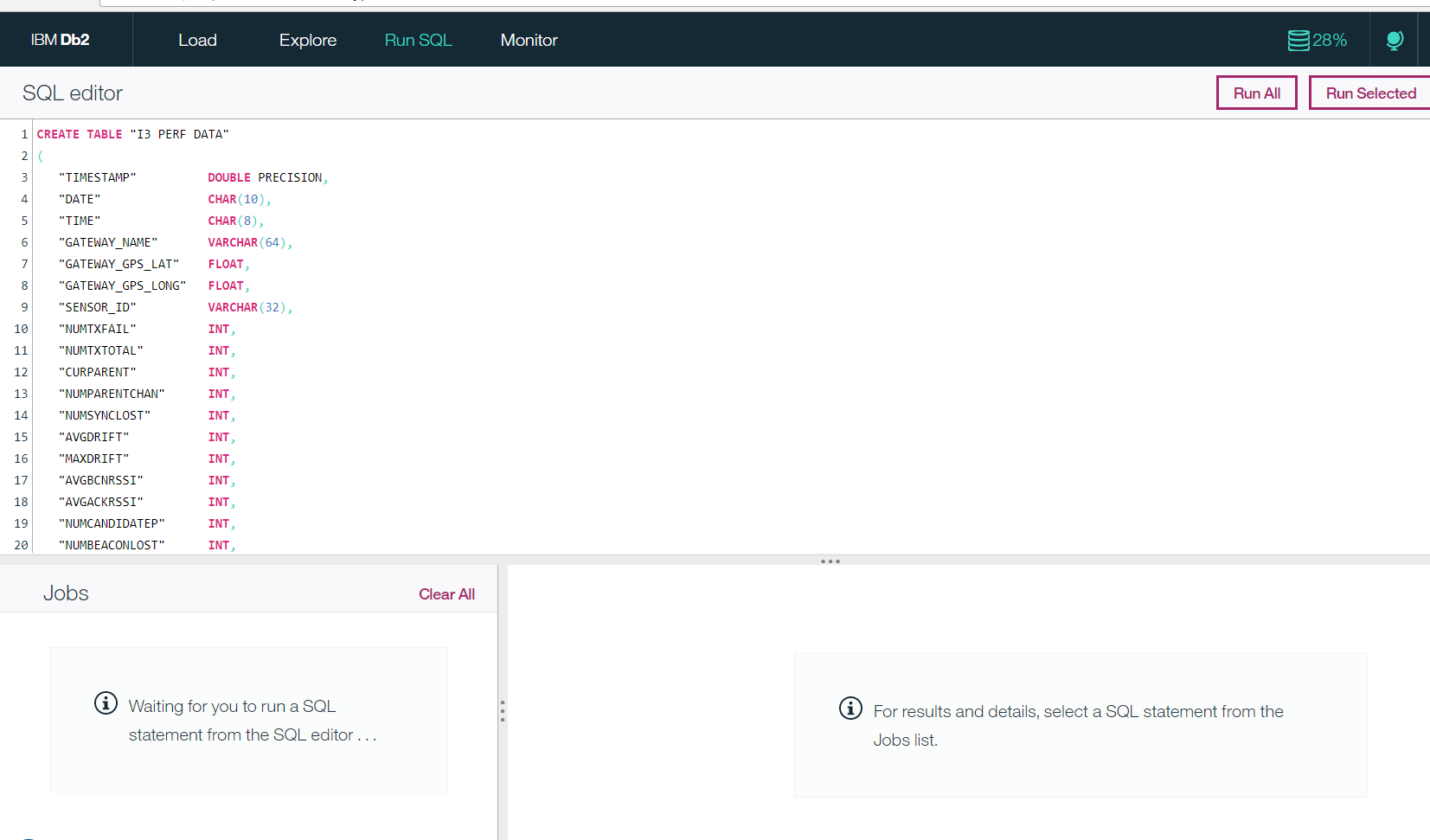
"APP\_PER\_LAST\_SEQ" FLOAT,

"APP\_PER\_SENT" FLOAT,

"APP\_PER\_LOST" FLOAT

);

“



## Update the node-red application source code

Replace the node-red application source code with the following text:

“

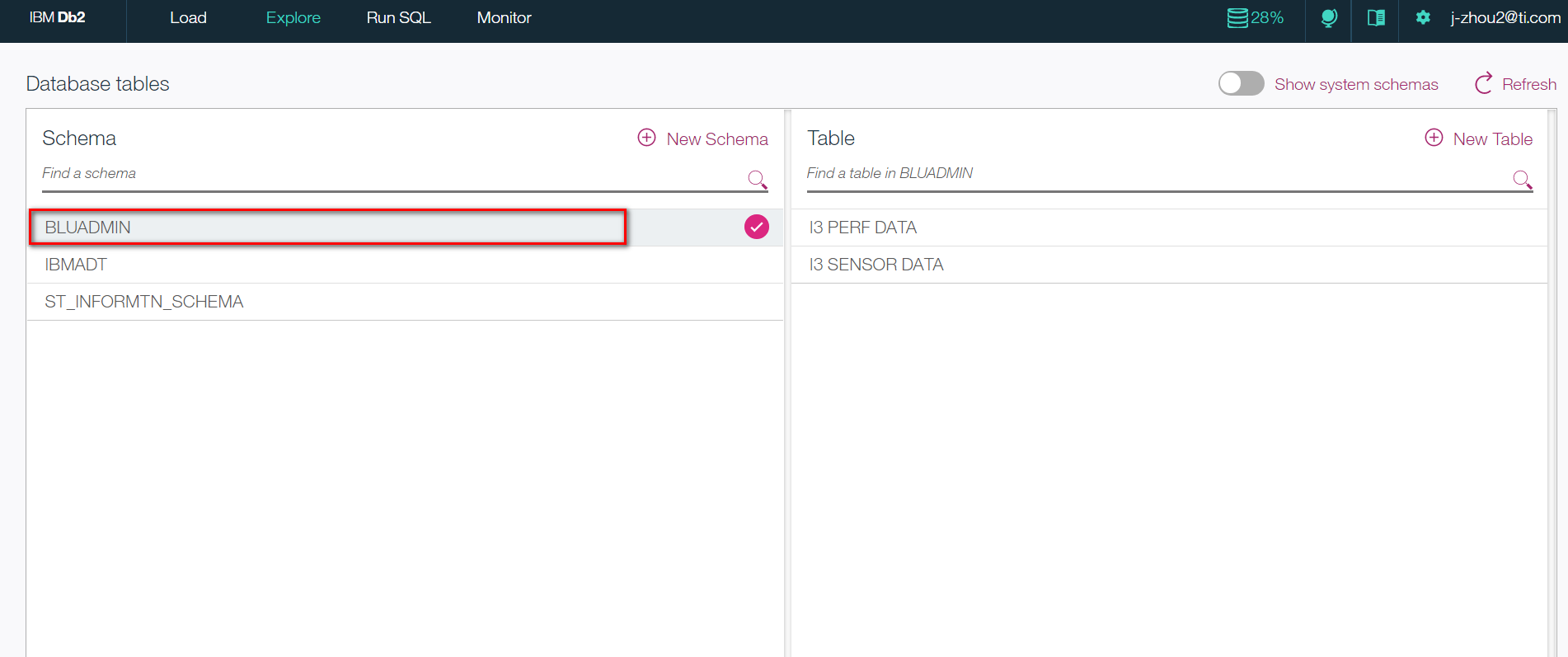
[{"id":"429d1a20.f1ed14","type":"dashDB in","z":"f5e4497.bab7cf8","dashDB":"","service":"dashDB for Analytics-iv","query":"SELECT RHUM FROM \"I3 SENSOR DATA\" WHERE GATEWAY\_NAME='I3-Kilby' AND DATE='2017-05-11' AND TIME > '18:34:00' AND TIME < '18:36:00' AND SENSOR\_ID=4;","params":"","name":"","x":553,"y":103.69999694824219,"wires":[["65e899cb.78a02","70e32053.228824"]]},{"id":"77b92694.8ef91","type":"http in","z":"f5e4497.bab7cf8","name":"","url":"/sites","method":"post","swaggerDoc":"","x":127.39999389648438,"y":398.1600341796875,"wires":[["552b76ee.9c2568","db2671ff.66c848","7f66f290.2cb284"]]},{"id":"f6781dd2.0e722","type":"debug","z":"f5e4497.bab7cf8","name":"","active":false,"console":"false","complete":"true","x":890,"y":440,"wires":[]},{"id":"258c7726.6bc8b8","type":"change","z":"f5e4497.bab7cf8","name":"addtimestamp","rules":[{"t":"set","p":"timestamp","pt":"msg","to":"","tot":"date"}],"action":"","property":"","from":"","to":"","reg":false,"x":695.3999328613281,"y":397.8599853515625,"wires":[["f6781dd2.0e722","12c3fc40.4af96c"]]},{"id":"552b76ee.9c2568","type":"change","z":"f5e4497.bab7cf8","name":"","rules":[{"t":"delete","p":"req","pt":"msg"}],"action":"","property":"","from":"","to":"","reg":false,"x":319.3900146484375,"y":398.3999938964844,"wires":[["323f668c.0d81e6"]]},{"id":"323f668c.0d81e6","type":"change","z":"f5e4497.bab7cf8","name":"","rules":[{"t":"delete","p":"res","pt":"msg"}],"action":"","property":"","from":"","to":"","reg":false,"x":512.2000122070312,"y":398.4099884033203,"wires":[["258c7726.6bc8b8"]]},{"id":"18c62aa7.f6cf41","type":"http in","z":"f5e4497.bab7cf8","name":"","url":"/start","method":"get","swaggerDoc":"","x":125.80001831054688,"y":551.7599945068359,"wires":[["7f66f290.2cb284"]]},{"id":"d09a55ef.0c6aa","type":"http response","z":"f5e4497.bab7cf8","name":"home page","x":700.18994140625,"y":527.1300048828125,"wires":[]},{"id":"b2bf3be8.58b02","type":"function","z":"f5e4497.bab7cf8","name":"compose home page","func":"msg.payload = `<html>\n<meta http-equiv=\"refresh\" content=\"2\" >\n<head>\n</head>\n<body>\n<form action=\"http://tiiot1.mybluemix.net/sites\" method=\"post\">\n<fieldset>\n<legend>Sensor Data:</legend>\nGateway:<br>\n<input type=\"text\" name=\"gwid\" value=`+msg.gwid+`><br>\nSensor:<br>\n<input type=\"text\" name=\"sensorid\" value=`+msg.sensorid+`><br>\nTemperature:<br>\n<input type=\"text\" name=\"temp\" value=`+msg.temp+`><br>\nHumidity:<br>\n<input type=\"text\" name=\"hum\" value=`+msg.hum+`><br>\nPressure:<br>\n<input type=\"text\" name=\"pres\" value=`+msg.pres+`><br><br>\n<input type=\"submit\" value=\"Submit\">\n</fieldset></form>\n</body>\n</html>`;\nreturn msg;","outputs":1,"noerr":0,"x":494.98992919921875,"y":525.6600494384766,"wires":[["d09a55ef.0c6aa","21bf1ee4.6ba3c6"]]},{"id":"7f66f290.2cb284","type":"function","z":"f5e4497.bab7cf8","name":"extract data","func":"if (!global.gwid)\n{\n global.gwid = 0;\n global.sensorid = 0;\n global.temp = 0;\n global.hum = 0;\n global.press = 0;\n}\n\nif (msg.payload.gateway\_0)\n{\n if (msg.payload.gateway\_0.name)\n global.gwid = '\"'+msg.payload.gateway\_0.name+'\"';\n if (msg.payload.sensor\_0.\_id) \n {\n global.sensorid = '\"'+msg.payload.sensor\_0.\_id+'\"';\n }\n \n if (msg.payload.sensor\_0.data)\n {\n if (msg.payload.sensor\_0.data.tamb)\n global.temp = msg.payload.sensor\_0.data.tamb;\n if (msg.payload.sensor\_0.data.rhum)\n global.hum = msg.payload.sensor\_0.data.rhum;\n if (msg.payload.sensor\_0.data.press)\n global.press = msg.payload.sensor\_0.data.press;\n }\n}\n\nmsg.gwid = global.gwid;\nmsg.sensorid = global.sensorid;\nmsg.temp = global.temp;\nmsg.hum = global.hum;\nmsg.pres = global.press;\nreturn msg;","outputs":1,"noerr":0,"x":282.5899963378906,"y":501.57997131347656,"wires":[["b2bf3be8.58b02"]]},{"id":"bf644ba6.8ee1f8","type":"comment","z":"f5e4497.bab7cf8","name":"Add data to the database:","info":"","x":144.99002075195312,"y":359.44000244140625,"wires":[]},{"id":"21bf1ee4.6ba3c6","type":"debug","z":"f5e4497.bab7cf8","name":"","active":false,"console":"false","complete":"true","x":678.5900268554688,"y":472.7299499511719,"wires":[]},{"id":"db2671ff.66c848","type":"debug","z":"f5e4497.bab7cf8","name":"","active":false,"console":"false","complete":"true","x":95.80000305175781,"y":472.53997802734375,"wires":[]},{"id":"b12bf950.d7e28","type":"mqtt out","z":"f5e4497.bab7cf8","name":"","topic":"iot-2/evt/timestamp/fmt/json","qos":"","retain":"true","broker":"971d9ea2.644a9","x":1160,"y":100,"wires":[]},{"id":"4ad86d8d.fd21","type":"function","z":"f5e4497.bab7cf8","name":"Demux","func":"\nvar omsgStr = '{\"'+msg.payload.gateway\_0.name+'\":';\nomsgStr += '{\"'+msg.payload.sensor\_0.\_id+'\":';\nomsgStr += JSON.stringify(msg.payload);\nomsgStr += '}}';\nmsg.payload = omsgStr;\n\nreturn msg;","outputs":1,"noerr":0,"x":920,"y":100,"wires":[["b12bf950.d7e28","ffe23438.59ff18"]]},{"id":"ffe23438.59ff18","type":"debug","z":"f5e4497.bab7cf8","name":"","active":false,"console":"false","complete":"false","x":1150,"y":60,"wires":[]},{"id":"c497e679.1feb1","type":"function","z":"f5e4497.bab7cf8","name":"formatSql","func":"var ts = new Date(msg.timestamp);\nvar a = JSON.stringify(ts.getMonth()+1);\nif (!msg.payload.sensor\_0.data.lux)\n{\n msg.payload.sensor\_0.data.lux = 0;\n}\n\nif (!msg.payload.sensor\_0.data.gyrox)\n{\n msg.payload.sensor\_0.data.gyrox = 0;\n}\n\nif (!msg.payload.sensor\_0.data.gyroy)\n{\n msg.payload.sensor\_0.data.gyroy = 0;\n}\n\nif (!msg.payload.sensor\_0.data.gyroz)\n{\n msg.payload.sensor\_0.data.gyroz = 0;\n}\n\nif (!msg.payload.sensor\_0.data.rhum)\n{\n msg.payload.sensor\_0.data.rhum = 0;\n}\n\nmon = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getDate());\ndat = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getHours());\nhr = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getMinutes());\nmin = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getSeconds());\nsec = (a < 10) ? ('0'+a) : a;\nvar dstr = JSON.stringify(ts.getFullYear())+'-'+mon+'-'+dat;\nvar tstr = hr+':'+min+':'+sec;\n\nvar sqlMsg = {\n TIMESTAMP : msg.timestamp,\n DATE : dstr,\n TIME : tstr,\n GATEWAY\_NAME : msg.payload.gateway\_0.name,\n GATEWAY\_GPS\_LAT : msg.payload.gateway\_0.gps[0],\n GATEWAY\_GPS\_LONG : msg.payload.gateway\_0.gps[1],\n SENSOR\_ID : msg.payload.sensor\_0.\_id,\n ACCELX : msg.payload.sensor\_0.data.accelx,\n ACCELY : msg.payload.sensor\_0.data.accely,\n ACCELZ : msg.payload.sensor\_0.data.accelz,\n BAT : msg.payload.sensor\_0.data.bat,\n CC2650\_ACTIVE : msg.payload.sensor\_0.data.cc2650\_active,\n CC2650\_SLEEP : msg.payload.sensor\_0.data.cc2650\_sleep,\n CHANNEL : msg.payload.sensor\_0.data.channel,\n EH : msg.payload.sensor\_0.data.eh,\n EH1 : msg.payload.sensor\_0.data.eh1,\n GPSEN\_ACTIVE : msg.payload.sensor\_0.data.gpsen\_active,\n GPSEN\_SLEEP : msg.payload.sensor\_0.data.gpsen\_sleep,\n GYROX : msg.payload.sensor\_0.data.gyrox,\n GYROY : msg.payload.sensor\_0.data.gyroy,\n GYROZ : msg.payload.sensor\_0.data.gyroz,\n LED : msg.payload.sensor\_0.data.led,\n LUX : msg.payload.sensor\_0.data.lux,\n MSP432\_ACTIVE : msg.payload.sensor\_0.data.msp432\_active,\n MSP432\_SLEEP : msg.payload.sensor\_0.data.msp432\_sleep,\n PRESS : msg.payload.sensor\_0.data.press,\n RF\_RX : msg.payload.sensor\_0.data.rf\_rx,\n RF\_TX : msg.payload.sensor\_0.data.rf\_tx,\n RHUM : msg.payload.sensor\_0.data.rhum,\n SSM\_ACTIVE : msg.payload.sensor\_0.data.ssm\_active,\n SSM\_SLEEP : msg.payload.sensor\_0.data.ssm\_sleep,\n TAMB : msg.payload.sensor\_0.data.tamb\n};\n \nmsg = {payload : sqlMsg};\nreturn msg;","outputs":1,"noerr":0,"x":860,"y":200,"wires":[["f22c12f5.751cc","81fb3ff7.b09238"]]},{"id":"f22c12f5.751cc","type":"debug","z":"f5e4497.bab7cf8","name":"","active":false,"console":"false","complete":"false","x":1024,"y":161.07998657226562,"wires":[]},{"id":"81fb3ff7.b09238","type":"dashDB out","z":"f5e4497.bab7cf8","dashDB":"","service":"Db2 on Cloud-26","table":"I3 SENSOR DATA","name":"I3 database","x":1024,"y":199.44998168945312,"wires":[]},{"id":"65e899cb.78a02","type":"debug","z":"f5e4497.bab7cf8","name":"","active":false,"console":"false","complete":"true","x":730,"y":100,"wires":[]},{"id":"54452b25.78a49","type":"inject","z":"f5e4497.bab7cf8","name":"","topic":"Query","payload":"true","payloadType":"bool","repeat":"","crontab":"","once":false,"x":325,"y":100.36000061035156,"wires":[["429d1a20.f1ed14"]]},{"id":"70e32053.228824","type":"ui\_chart","z":"f5e4497.bab7cf8","name":"","group":"36a90f63.5f1024","order":0,"width":0,"height":0,"label":"chart","chartType":"line","legend":"false","xformat":"HH:mm:ss","interpolate":"linear","nodata":"","ymin":"0","ymax":"20","removeOlder":1,"removeOlderPoints":"1000","removeOlderUnit":"3600","cutout":0,"colors":["#1F77B4","#AEC7E8","#FF7F0E","#2CA02C","#98DF8A","#D62728","#FF9896","#9467BD","#C5B0D5"],"x":744,"y":37.43999481201172,"wires":[[],[]]},{"id":"6e32697.d685f58","type":"inject","z":"f5e4497.bab7cf8","name":"","topic":"","payload":"","payloadType":"date","repeat":"1","crontab":"","once":false,"x":328,"y":46.36000061035156,"wires":[["ee1ddc6e.89d38"]]},{"id":"ee1ddc6e.89d38","type":"function","z":"f5e4497.bab7cf8","name":"dataGen","func":"msg.payload = '17';\nmsg.topic = 'junk';\nreturn msg;","outputs":1,"noerr":0,"x":504,"y":36.72999572753906,"wires":[[]]},{"id":"12c3fc40.4af96c","type":"switch","z":"f5e4497.bab7cf8","name":"dataRouter","property":"payload.type","propertyType":"msg","rules":[{"t":"eq","v":"sensor\_type\_0","vt":"str"},{"t":"eq","v":"sensor\_type\_perf\_data","vt":"str"},{"t":"eq","v":"network\_perf\_data","vt":"str"}],"checkall":"true","outputs":3,"x":674,"y":210.32000732421875,"wires":[["c497e679.1feb1","4ad86d8d.fd21"],["fd47f5aa.78482"],["c816a995.e1af58"]]},{"id":"fd47f5aa.78482","type":"function","z":"f5e4497.bab7cf8","name":"formatSqlPerfData","func":"var ts = new Date(msg.timestamp);\nvar a = JSON.stringify(ts.getMonth()+1);\nmon = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getDate());\ndat = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getHours());\nhr = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getMinutes());\nmin = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getSeconds());\nsec = (a < 10) ? ('0'+a) : a;\nvar dstr = JSON.stringify(ts.getFullYear())+'-'+mon+'-'+dat;\nvar tstr = hr+':'+min+':'+sec;\n\nvar beacon\_info = [];\nvar i;\nif (msg.payload.sensor\_0.data.msg.beacon\_info)\n{\n for (i = 0; i < msg.payload.sensor\_0.data.msg.beacon\_info.length; i++)\n {\n beacon\_info.push(\n {\n sa: msg.payload.sensor\_0.data.msg.beacon\_info[i].shortAddr,\n jp: msg.payload.sensor\_0.data.msg.beacon\_info[i].joinPriority,\n ri: msg.payload.sensor\_0.data.msg.beacon\_info[i].rssi,\n lq: msg.payload.sensor\_0.data.msg.beacon\_info[i].lqi\n });\n }\n}\nif (!msg.payload.sensor\_0.data.msg.timeAssociation)\n{\n msg.payload.sensor\_0.data.msg.timeAssociation = 0;\n}\nvar sqlMsg = \n{\n TIMESTAMP : msg.timestamp,\n DATE : dstr,\n TIME : tstr,\n GATEWAY\_NAME : msg.payload.gateway\_0.name,\n GATEWAY\_GPS\_LAT : msg.payload.gateway\_0.gps[0],\n GATEWAY\_GPS\_LONG : msg.payload.gateway\_0.gps[1],\n SENSOR\_ID : msg.payload.sensor\_0.\_id,\n NUMTXFAIL : msg.payload.sensor\_0.data.msg.numTxFail,\n NUMTXTOTAL : msg.payload.sensor\_0.data.msg.numTxTotal,\n CURPARENT : msg.payload.sensor\_0.data.msg.curParent,\n NUMPARENTCHAN : msg.payload.sensor\_0.data.msg.numParentChange,\n NUMSYNCLOST : msg.payload.sensor\_0.data.msg.numSyncLost,\n AVGDRIFT : msg.payload.sensor\_0.data.msg.avgDrift,\n MAXDRIFT : msg.payload.sensor\_0.data.msg.maxDrift,\n AVGBCNRSSI : msg.payload.sensor\_0.data.msg.avgBcnRSSI,\n AVGACKRSSI : msg.payload.sensor\_0.data.msg.avgAckRSSI,\n NUMCANDIDATEP : msg.payload.sensor\_0.data.msg.numCandidateParent,\n NUMBEACONLOST : msg.payload.sensor\_0.data.msg.numBeaconLost,\n NUMBEACONTOTA : msg.payload.sensor\_0.data.msg.numBeaconTotal,\n TIMEASSOCIATI : msg.payload.sensor\_0.data.msg.timeAssociation,\n NUMOUTOFBUFFE : msg.payload.sensor\_0.data.msg.numOutOfBuffer,\n BEACON\_INFO : JSON.stringify(beacon\_info),\n TS : msg.payload.sensor\_0.data.ts,\n RTT : msg.payload.sensor\_0.data.rtt,\n APP\_PER\_LAST\_SEQ : msg.payload.sensor\_0.data.app\_per.last\_seq,\n APP\_PER\_SENT : msg.payload.sensor\_0.data.app\_per.sent,\n APP\_PER\_LOST : msg.payload.sensor\_0.data.app\_per.lost\n}; \n \nmsg = {payload : sqlMsg};\nreturn msg;","outputs":1,"noerr":0,"x":930,"y":260,"wires":[["8f69bea4.ceacd","ade77a2.6d0dc08"]]},{"id":"ade77a2.6d0dc08","type":"debug","z":"f5e4497.bab7cf8","name":"","active":false,"console":"false","complete":"payload","x":1210,"y":280,"wires":[]},{"id":"8f69bea4.ceacd","type":"dashDB out","z":"f5e4497.bab7cf8","dashDB":"","service":"Db2 on Cloud-26","table":"I3 PERF DATA","name":"I3 perf database","x":1200,"y":240,"wires":[]},{"id":"c816a995.e1af58","type":"function","z":"f5e4497.bab7cf8","name":"NetworkPerfData","func":"var ts = new Date(msg.timestamp);\nvar a = JSON.stringify(ts.getMonth()+1);\nmon = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getDate());\ndat = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getHours());\nhr = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getMinutes());\nmin = (a < 10) ? ('0'+a) : a;\na = JSON.stringify(ts.getSeconds());\nsec = (a < 10) ? ('0'+a) : a;\nvar dstr = JSON.stringify(ts.getFullYear())+'-'+mon+'-'+dat;\nvar tstr = hr+':'+min+':'+sec;\n\nvar sqlMsg = \n{\n TIMESTAMP : msg.timestamp,\n DATE : dstr,\n TIME : tstr,\n GATEWAY\_NAME : msg.payload.gateway\_0.name,\n GATEWAY\_GPS\_LAT : msg.payload.gateway\_0.gps[0],\n GATEWAY\_GPS\_LONG : msg.payload.gateway\_0.gps[1],\n SENSOR\_ID : msg.payload.sensor\_0.\_id,\n NUMTXRETRY : msg.payload.sensor\_0.data.msg.numTxRetry,\n NUMTXTOTAL : msg.payload.sensor\_0.data.msg.numTxTotal,\n CURPARENT : msg.payload.sensor\_0.data.msg.curParent,\n NUMPARENTCHAN : msg.payload.sensor\_0.data.msg.numParentChange,\n NUMSYNCLOST : msg.payload.sensor\_0.data.msg.numSyncLost,\n AVGDRIFT : msg.payload.sensor\_0.data.msg.avgDrift,\n MAXDRIFT : msg.payload.sensor\_0.data.msg.maxDrift,\n AVGBCNRSSI : msg.payload.sensor\_0.data.msg.avgBcnRSSI,\n AVGACKRSSI : msg.payload.sensor\_0.data.msg.avgAckRSSI,\n NUMCANDIDATEP : msg.payload.sensor\_0.data.msg.numCandidateParent,\n NUMBEACONLOST : msg.payload.sensor\_0.data.msg.numBeaconLost,\n NUMBEACONTOTA : msg.payload.sensor\_0.data.msg.numBeaconTotal,\n NUMOUTOFBUFFE : msg.payload.sensor\_0.data.msg.numOutOfBuffer,\n UIPRXLOST : msg.payload.sensor\_0.data.msg.UIPuipRxLost,\n LOWPANTXLOST : msg.payload.sensor\_0.data.msg.lowpanTxLost,\n LOWPANRXLOST : msg.payload.sensor\_0.data.msg.lowpanRxLost,\n COAPRXLOST : msg.payload.sensor\_0.data.msg.coapRxLost,\n COAPOBSDIS : msg.payload.sensor\_0.data.msg.coapObsDis,\n RTT : msg.payload.sensor\_0.data.rtt,\n APP\_PER\_LAST\_SEQ : msg.payload.sensor\_0.data.app\_per.last\_seq,\n APP\_PER\_SENT : msg.payload.sensor\_0.data.app\_per.sent,\n APP\_PER\_LOST : msg.payload.sensor\_0.data.app\_per.lost,\n TS : ts\n}; \n \nmsg = {payload : sqlMsg};\nreturn msg;","outputs":"3","noerr":0,"x":930,"y":340,"wires":[["7b493cd1.117a64"],[],[]]},{"id":"7b493cd1.117a64","type":"debug","z":"f5e4497.bab7cf8","name":"","active":true,"console":"false","complete":"payload","x":1210,"y":340,"wires":[]},{"id":"971d9ea2.644a9","type":"mqtt-broker","z":"","broker":"ocwbfr.messaging.internetofthings.ibmcloud.com","port":"1883","clientid":"d:ocwbfr:2:2","usetls":false,"compatmode":true,"keepalive":"60","cleansession":true,"willTopic":"","willQos":"0","willRetain":"false","willPayload":"","birthTopic":"","birthQos":"0","birthRetain":"false","birthPayload":""},{"id":"36a90f63.5f1024","type":"ui\_group","z":"","name":"TII3","tab":"fa999557.5f0a88","disp":true,"width":"6"},{"id":"fa999557.5f0a88","type":"ui\_tab","z":"","name":"TII3","icon":"dashboard"}]

“

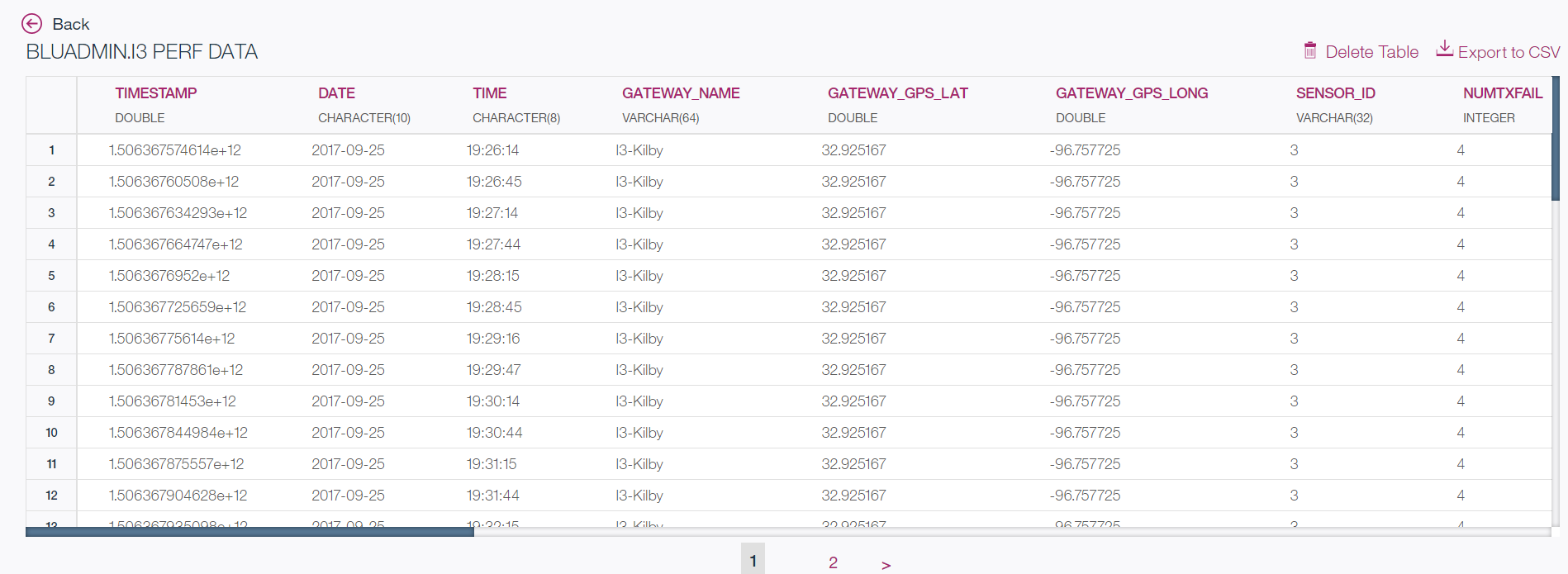
Launch the application.

## Access the SQL databases

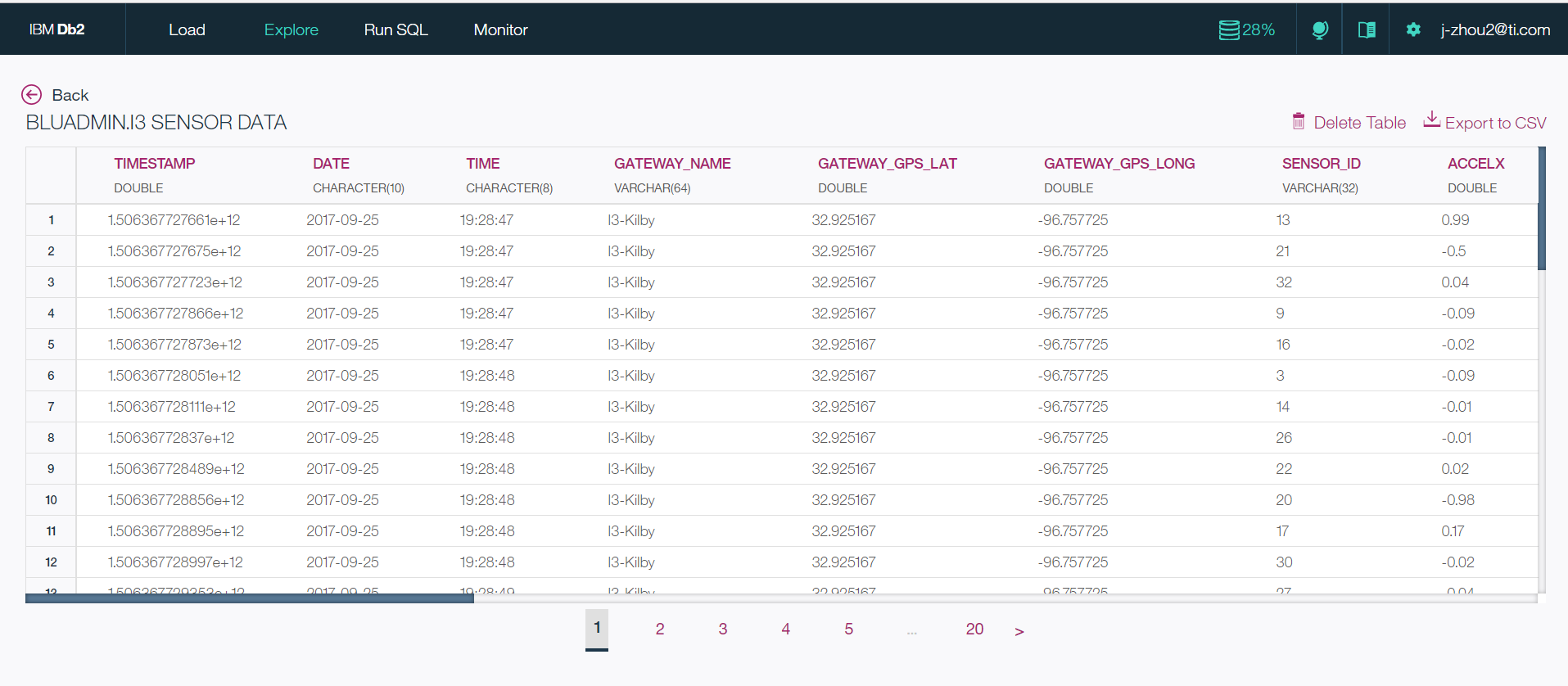
Select the Db2 service again and click ->Click “Explore”. Click “BLUEADMIN” :



Click “I3 PERF DATA”->”View Data”:



Click “I3 SENSOR DATA”-> “View Data”:



(END)