IBM Watson IoT – T2 – Platform – NodeRED flow

Commissioning task 2

1. Introduction

In the commissioning task 1 you signed for a user id into IBM Cloud environment, created a device authentication and wrote device data either with MQTT application or micro controller.

You can look for the details on what your account in the IBM Cloud will offer you: Browse to IBM Documentation and further to subtitle IBM Watson IoT.

https://www.ibm.com/docs/en/watson-iot-platform

Please select ”Getting Started, Product overview”. Most probably you will end up in page

https://www.ibm.com/docs/en/watson-iot-platform?topic=product-overview-features

You learned what is the Lite Plan on the cloud and what it will offer you.

Now you are ready to begin with this commissioning task 2.

In the commissiong task 2 we process device data in the Watson IoT platform.

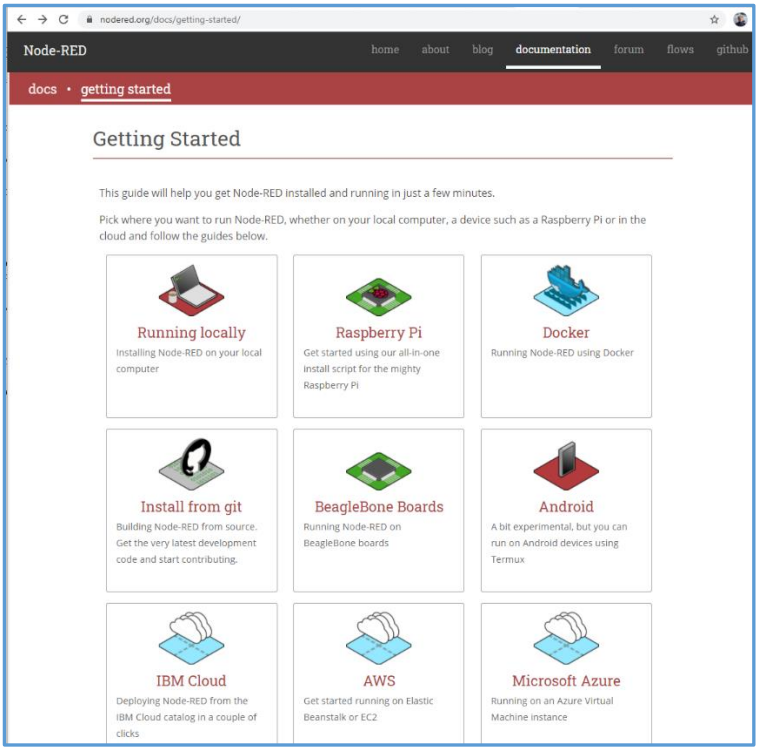
Please note that the IBM is continuously updating and developing their services. When you will be reading this the operations introduced in these instructions might already look different or exist on different web addresses. Anyway, the commissioning tasks introduced in these instructions are those that you would always need to complete.

2. Processing the device data. A device simulated in the Node-RED development environment.

Node-RED is an easy to use visual tool that can be used to develop applications, connections between applications and connections to external services.

You can have a look on options on building with the Node-RED at
https://nodered.org/docs/getting-started/

Fig. 2.1 The Node-RED / https://nodered.org/docs/getting-started/ 1.10. 2020 and 4.1.2022



In the next steps we will import into your Watson IoT platform a packet which includes the Node-RED installation and a Cloudant NoSQL database used by Node-RED.

An introduction to NodeRED you can find even on IBM hosted page
https://developer.ibm.com/components/node-red/

Click the “Node-RED Essentials” and watch the first 1 minute video.

A link to the installation package can be found in the IBM Cloud Catalog.

Navigate to the IBM Cloud Catalog.

https://cloud.ibm.com/catalog

Look for software ”Node-RED App” .

Fig. 2.2 IBM Cloud Catalog. Node-RED App. / https://cloud.ibm.com/catalog#software 4.1. 2022

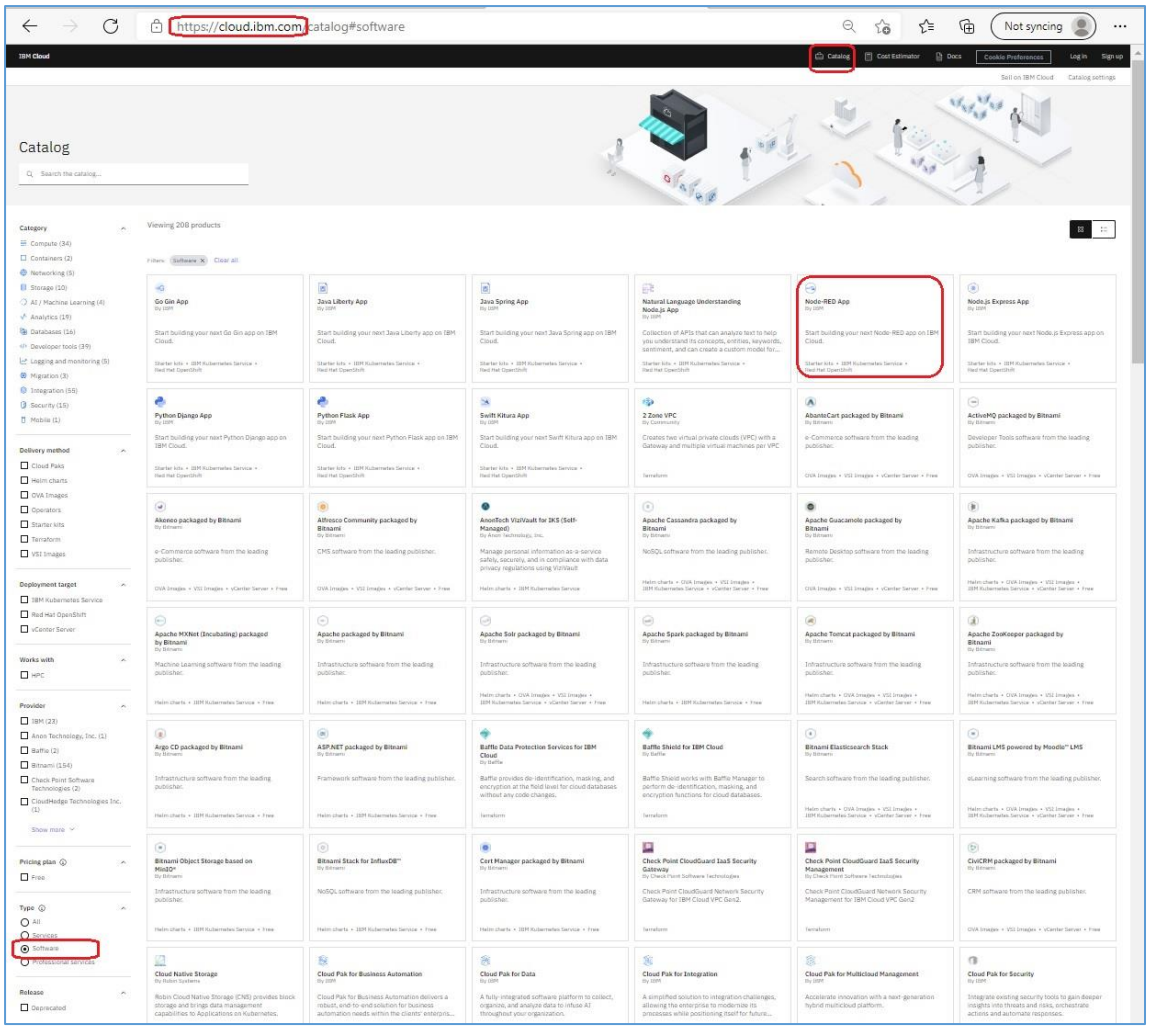
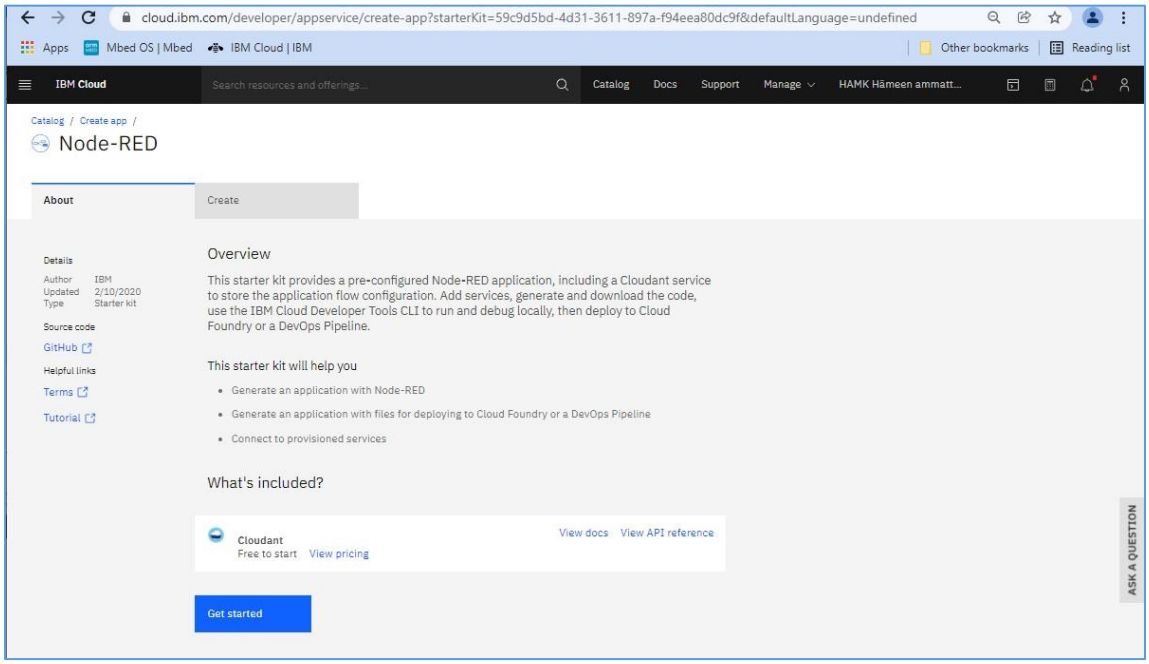


Fig 2.3 NodeRED. / 4.1. 2022



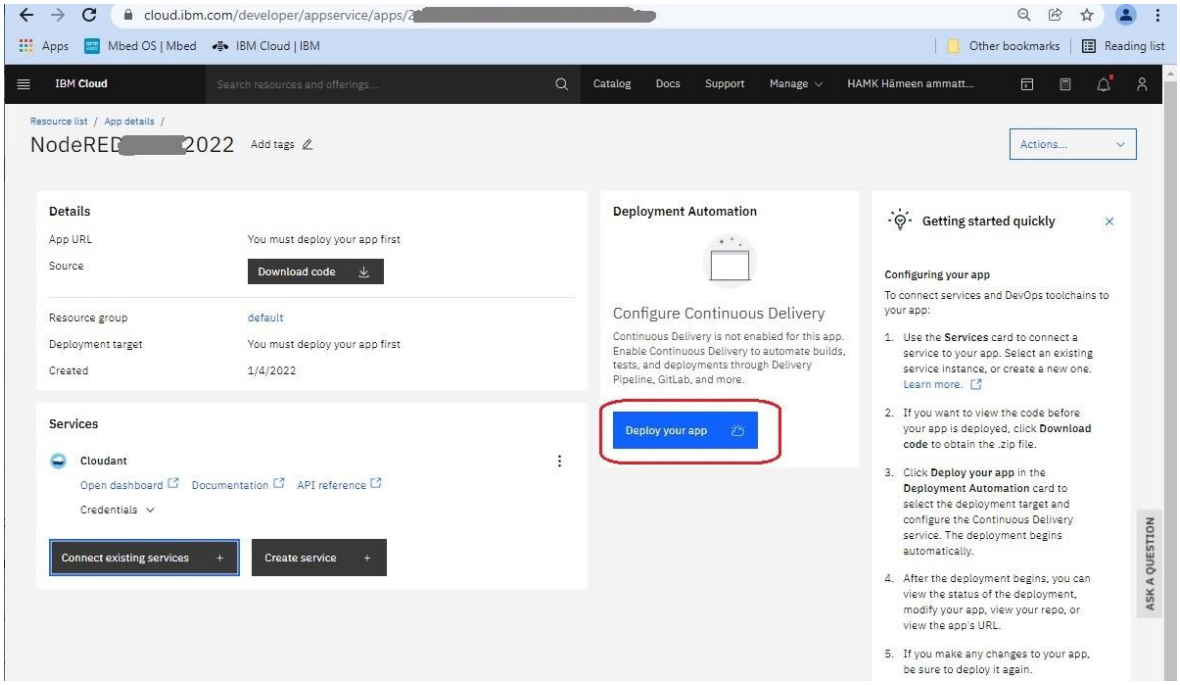
Click the Get started. Now you need to login with your own IBM Id.

In this stage there might be parameters to fill in. Please fill in the name for your application. It is preferable to write the name by using no spaces and using only normal letter and number characters. Please use a name that you can later easily recognise as a name for this application.

If possible, select the same region as you have selected in the earlier Watson IoT exercises. The pricing plan is Lite.

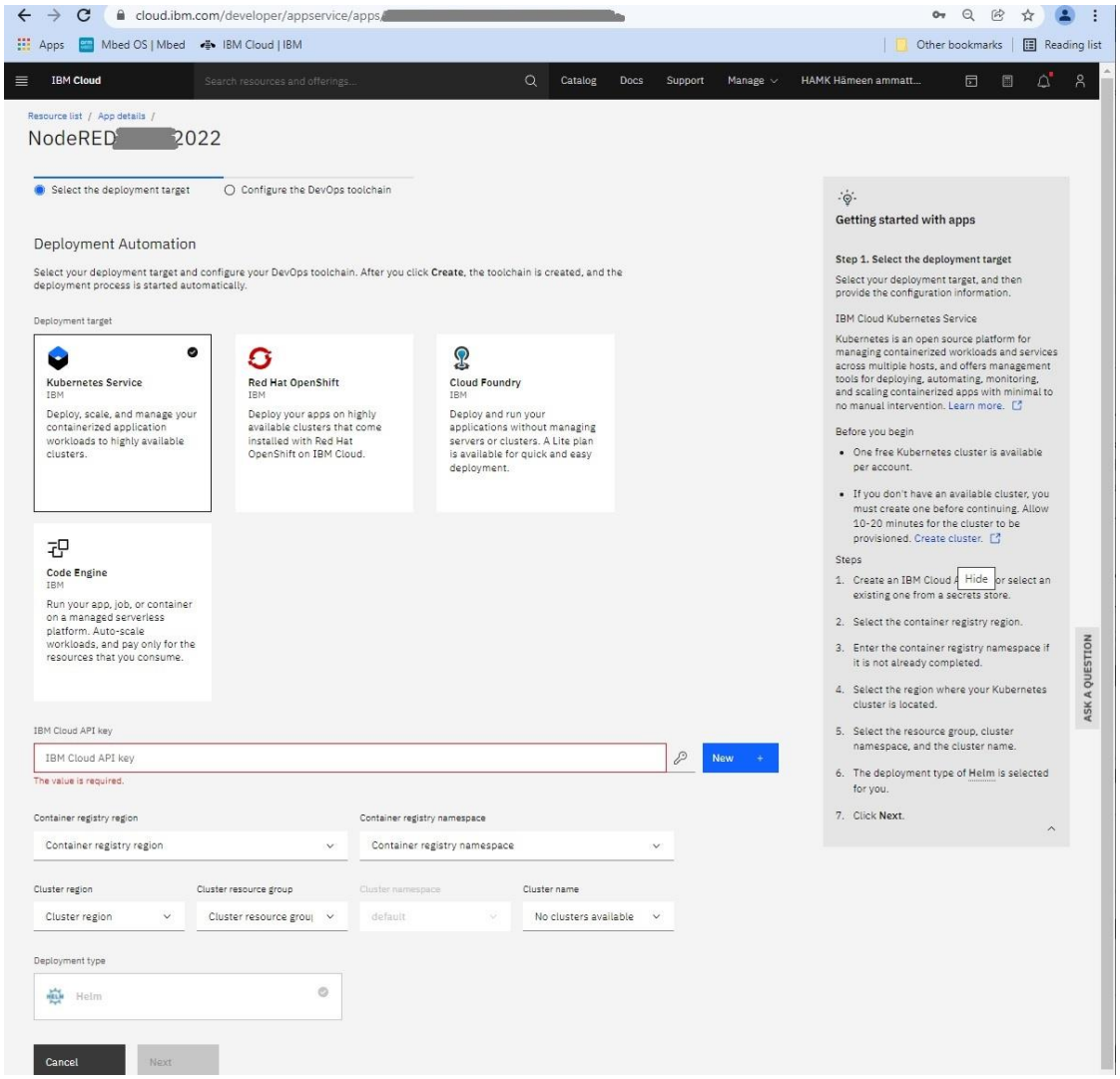
In the opening view click then Deploy in the Deployment automation.

Fig. 2.4 Deploy your app with Deployment automation. / 4.1.2022



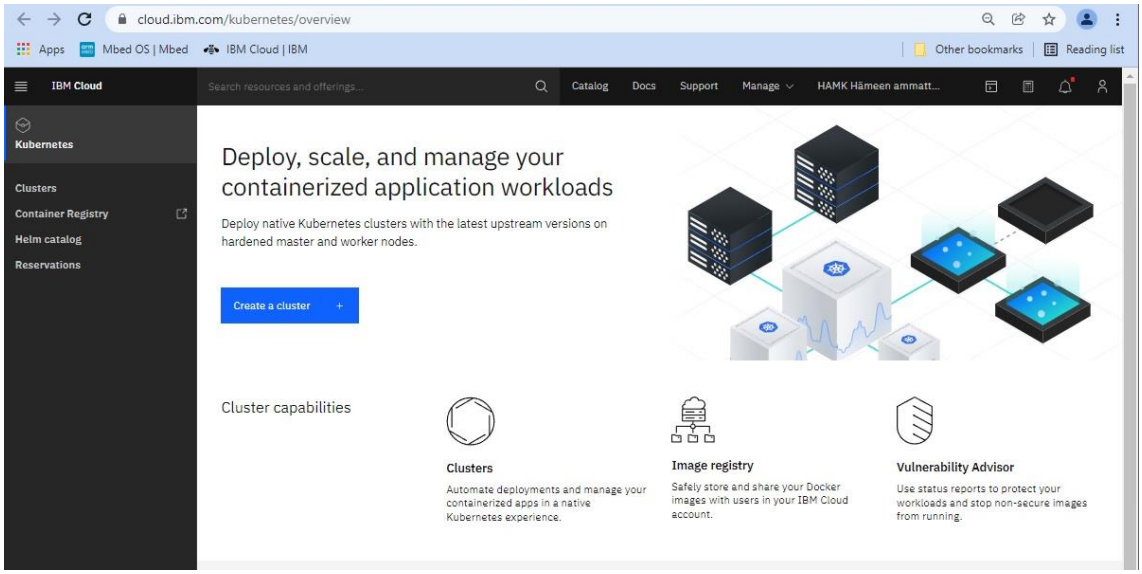
In the opening view select Kubernetes Services.

Fig. 2.5 Kubernetes Services. / 4.1. 2022



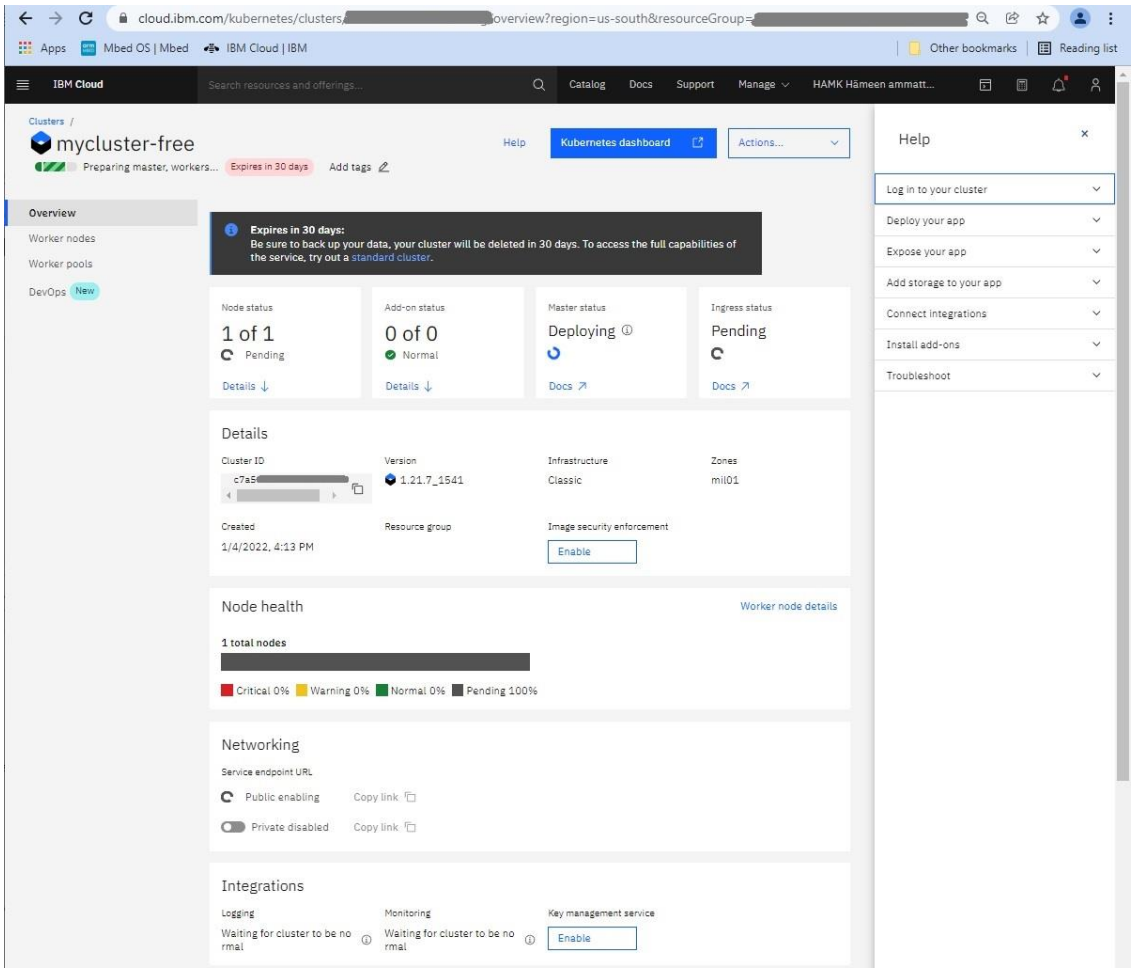
On the right right-click the Create Cluster, open in new tab and further select the plan Free.

Fig. 2.6 Creating the cluster. / 4.1. 2022



Click the Create cluster. The following view will open.

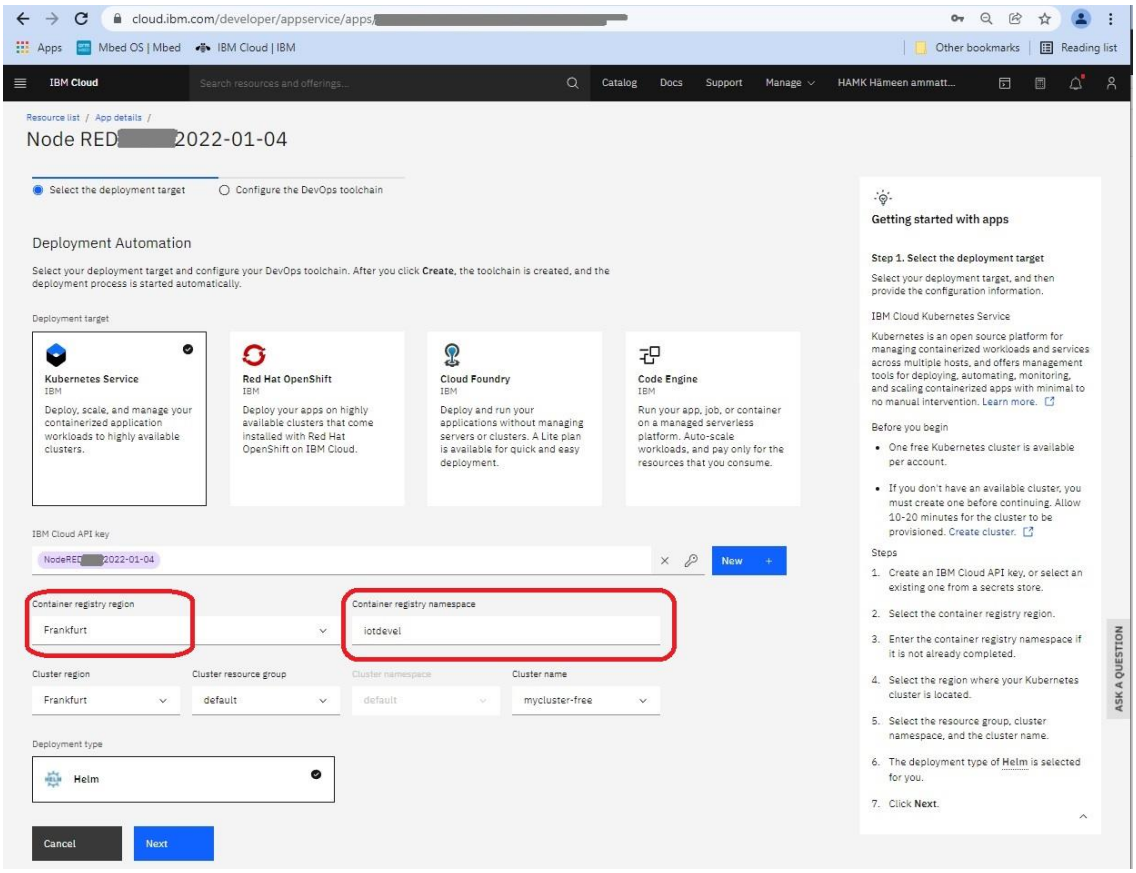
Fig. 2.7 In progress / 4.1. 2022



Please note that the deployment will take some time before the cluster will be ready. Finally, the Node health Normal will be 100% and the Pending 0%.

Get back to the NodeRed App page. Continue by creating a new API key. Click the IBM Cloud API key, New.

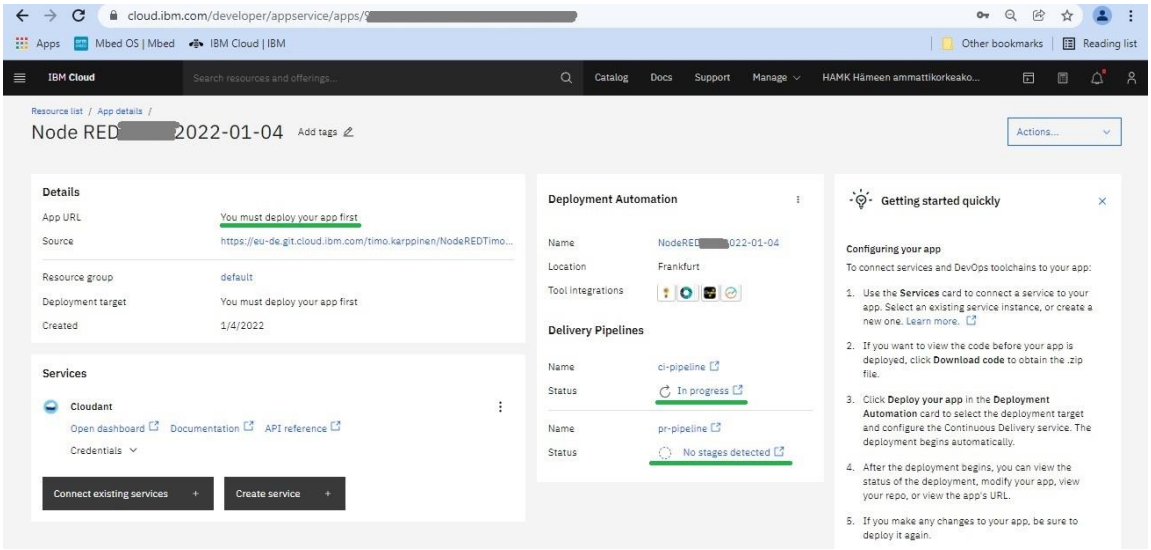
Fig 2.8. API key and the details / 4.1.2022



Select the same region as you have used earlier. You can edit some text on the Container registry namespaces.

Click the Next.

Fig 2.9. Deployment in progress / 4.1. 2022



It will take more than 15 minutes….. Finally the underlined text will change “No Stages detected”….”In Progress”….”Success”. The App URL will be finally visible.

As soon as I got the ci-pipeline success and the App URL visible I was able to continue to the NodeRED editor. Just copy the URL and browse to that IP address. Read the instructions on the page. Create your NodeRED username and password. Open the flow editor.

Finally the services created with NodeRED will be visible in the IBM Cloud Dashboard in your Resource list.

Fig 2.10 The Node-RED application with it’s resources in the IBM Cloud Resource list as Kubernetes Service App / 4.1. 2022

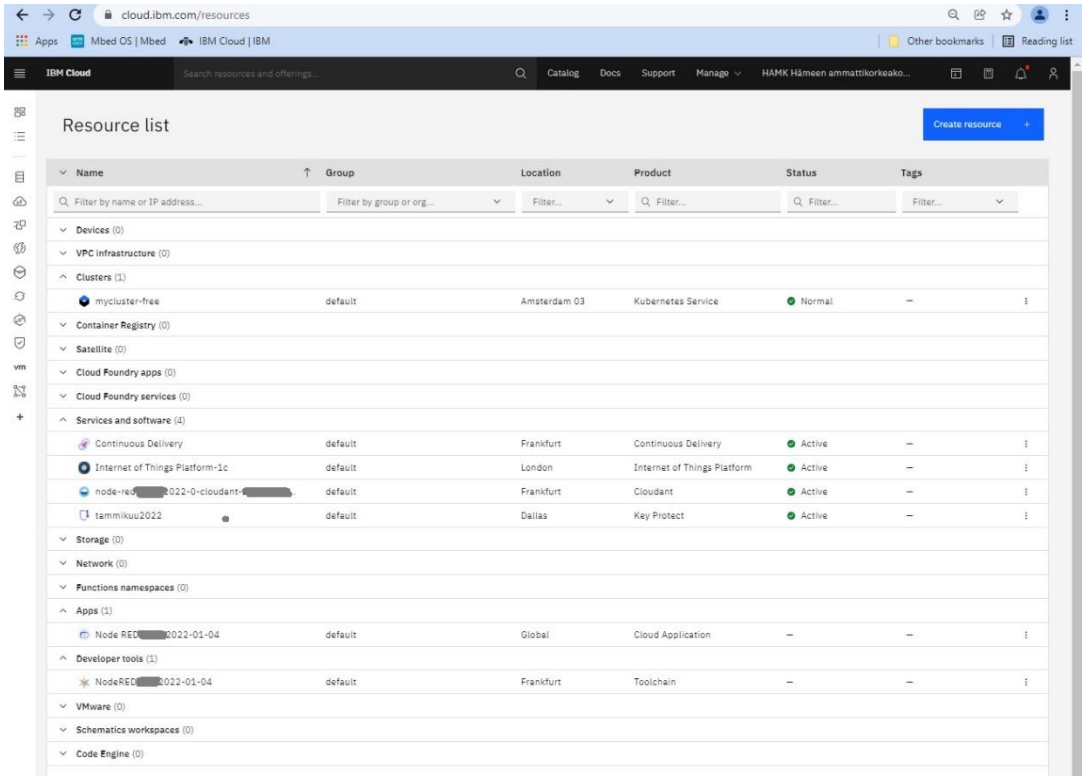
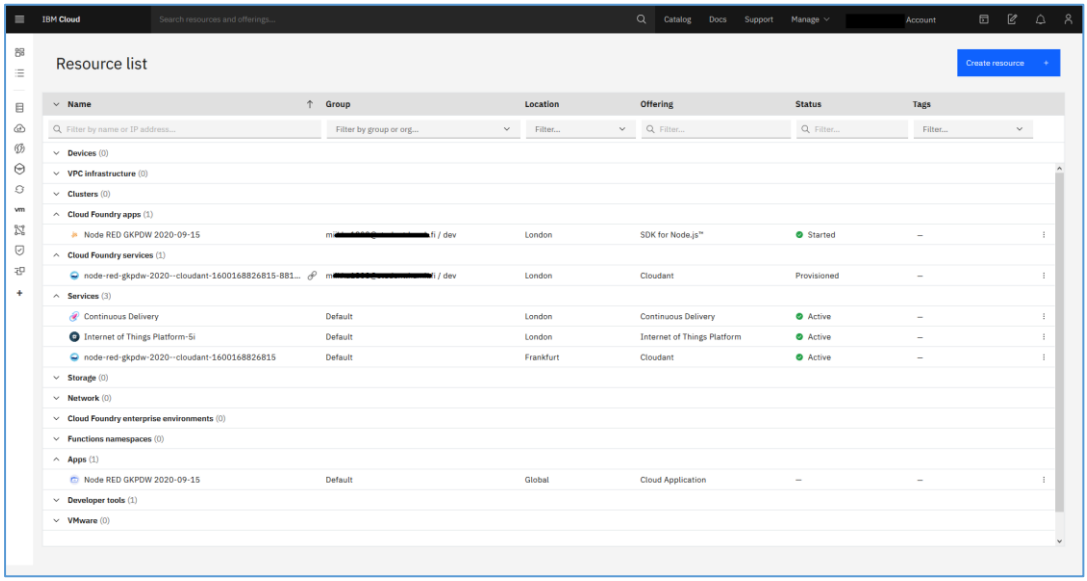


Fig 2.11 The Node-RED application with it's resources in the IBM Cloud Resource list as Cloud Foundry App. / 1.10. 2020



Please note that it is possible to create the NodeRED as a Cloud Foundry App or as a Kubernetes Service App. This time it was created as a Kubernetes Services App.

You can safely close your NodeRED flow editor. It can be started by selecting the Apps NodeRED in the Dashboard Resource list.

Fig. 2.12 If your own NodeRED development environment is running as Kubernetes Service App, click the IP address visible as the App URL. / 4.1. 2022

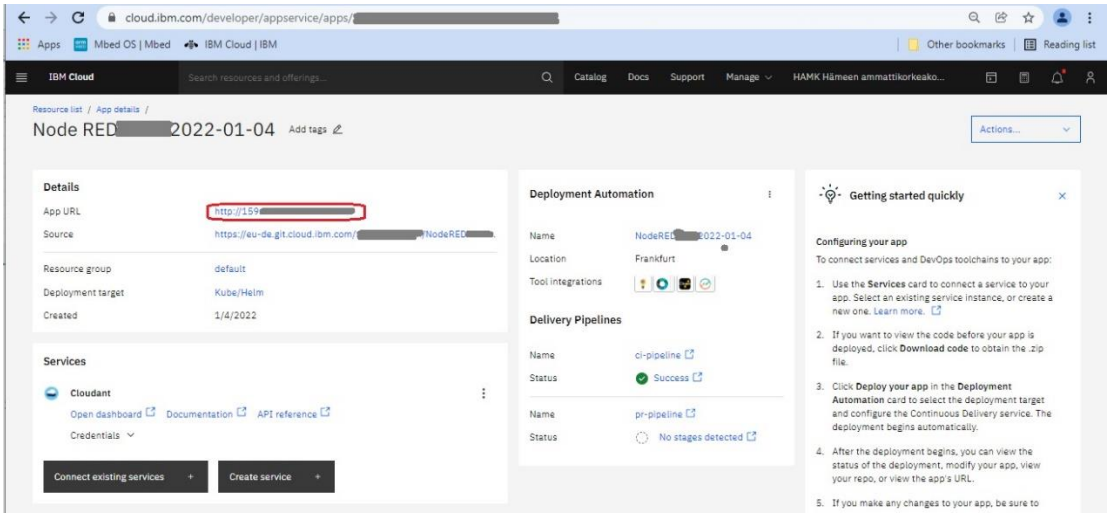
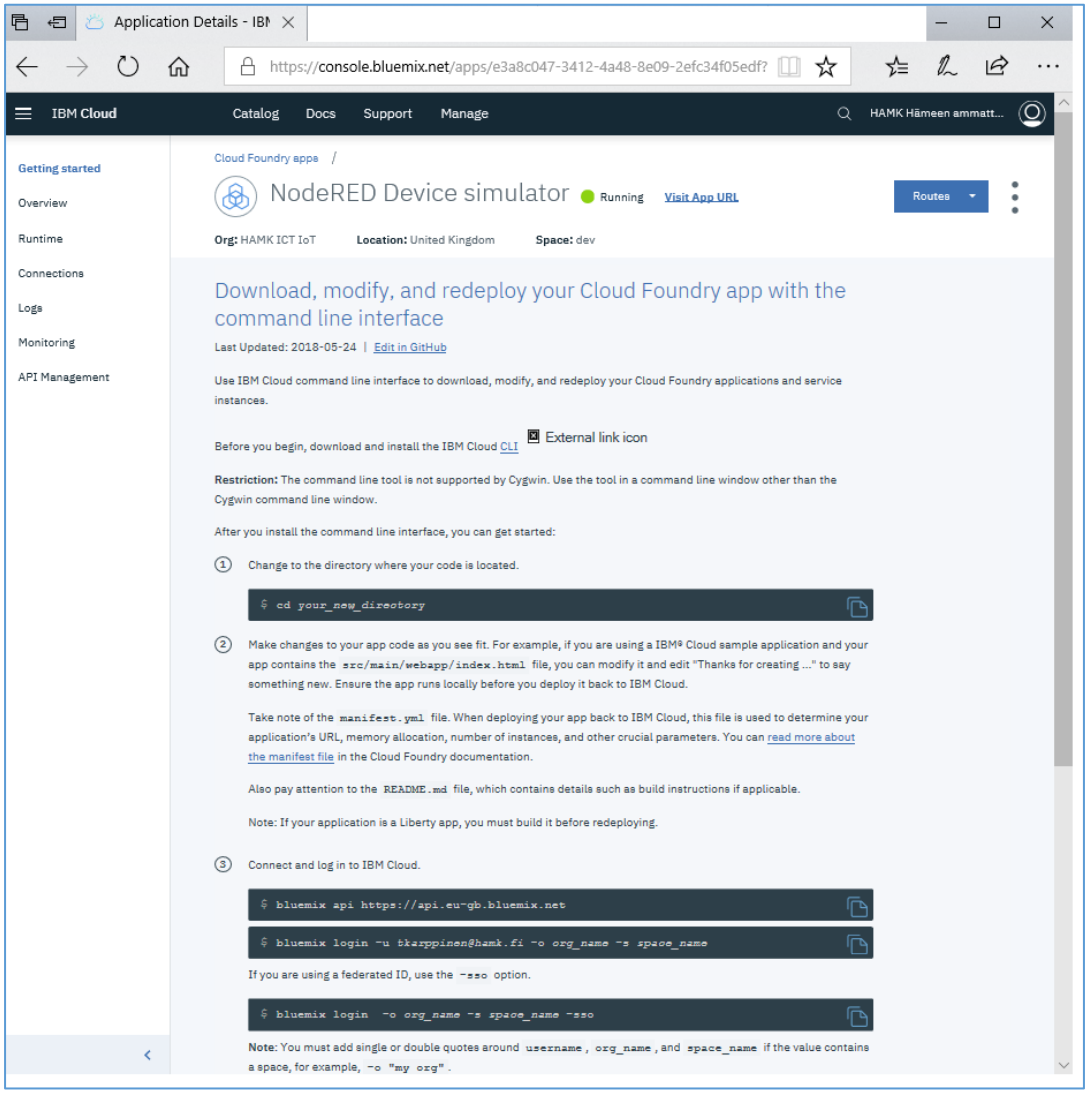
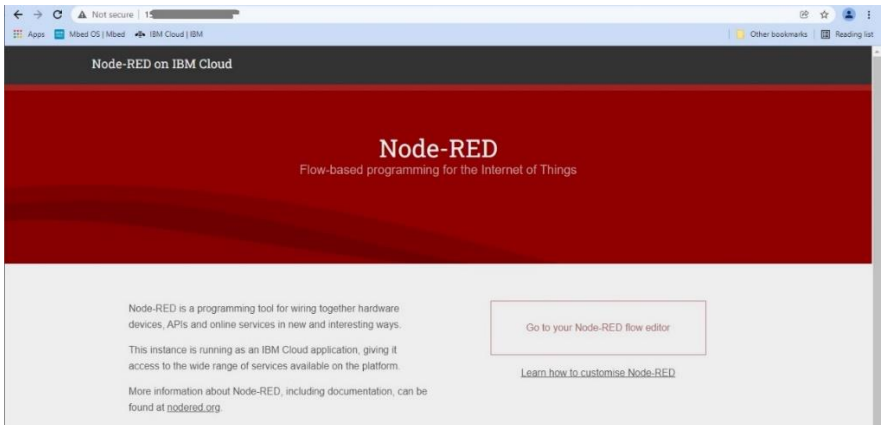


Fig. 2.13 If your own NodeRED environment is running as Cloud Foundry App. Click the Visit App URL / 2019



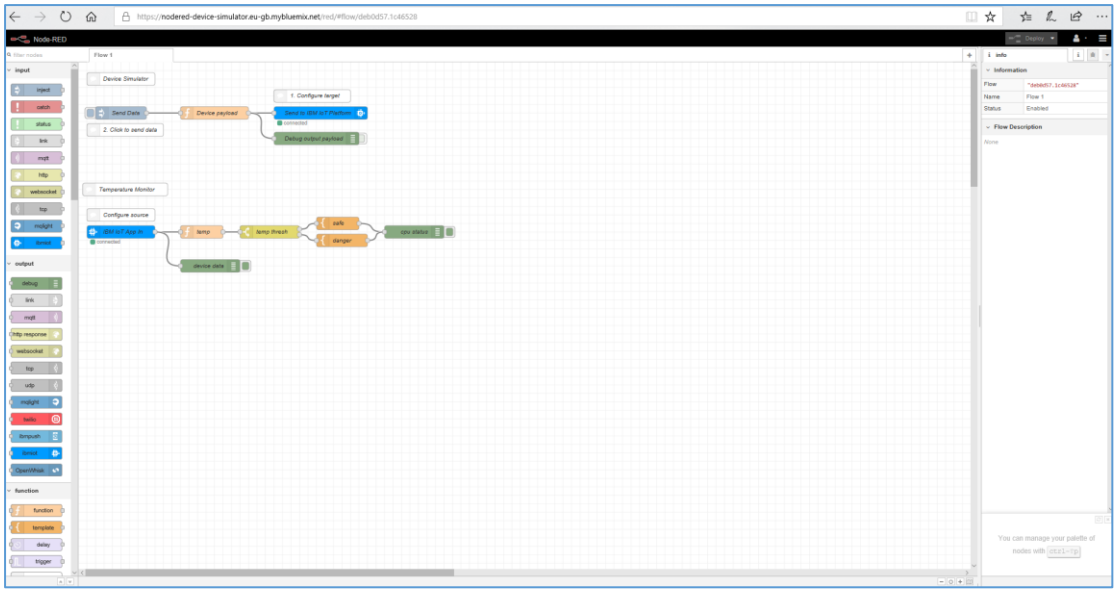
By clicking you will open a Node-RED editor. That looks like the red coloured Node-RED environment. For the editor you need to create a user name and a pass word.

Fig. 2.14 Opening your NodeRED editor. / 4.1. 2022



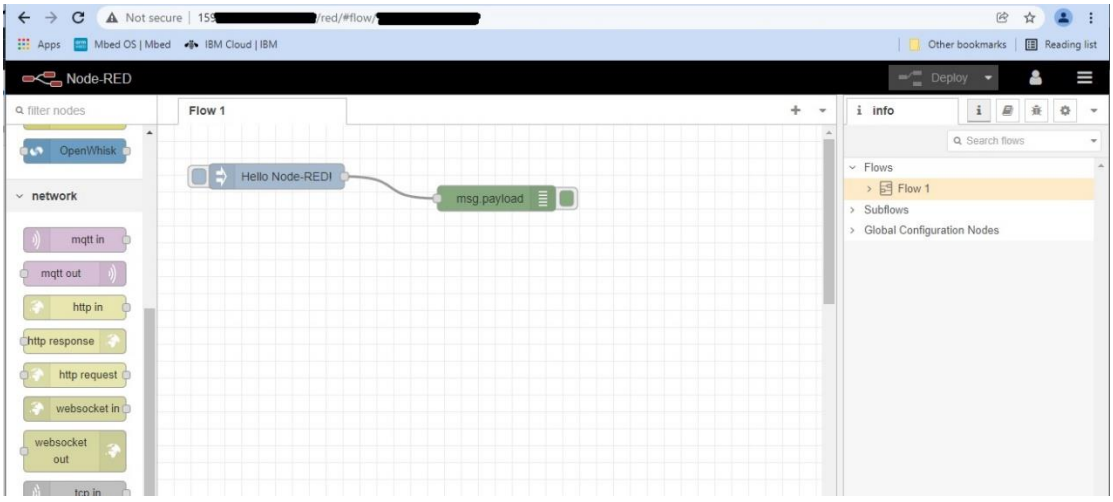
You might end up to a ready made flow in the Node-RED editor page.

Fig. 2.15 NodeRED flow editor. / 2019



Or the page could be almost empty with just two nodes.

Fig. 2.16 NodeRED flow editor. / 1.4. 2022



3. Working with the Flow Editor

When making changes to this so-called flow editor you need to press the “Deploy” in the top right corner to get these changes updated.

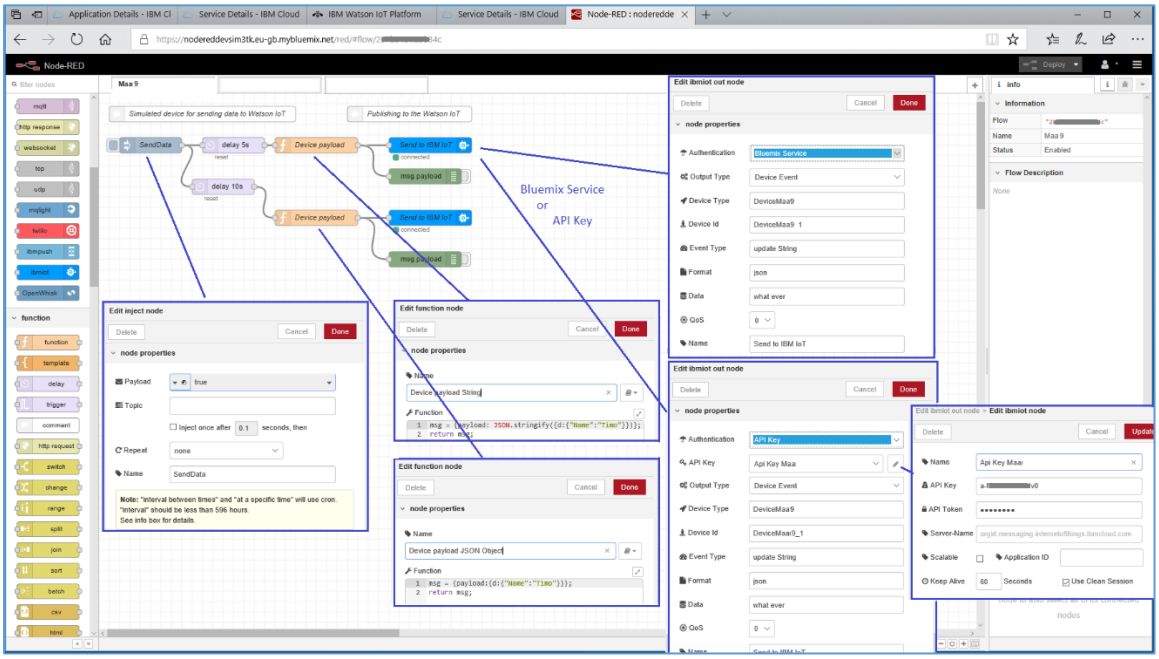
The type of each node can be recognised by the colour and symbol. The name text is often rewritten.

The messages from the Debug nodes will be visible after you click the bug image “debug” on the top right corner.

You might have an empty flow page if you did not create the Node-RED environment by installing the “Internet of Things Platform Starter”. That is no problem. You will still be able to create the same functions by creating a Node-RED flow page. Import the same nodes. Populate the nodes with the same content as seen in the next picture.

You can as well complete the flow from “Internet of Things Platform Starter” to have exactly the same nodes with the similar content.

Fig. 3.1 Flow with the nodes and the node parameters.



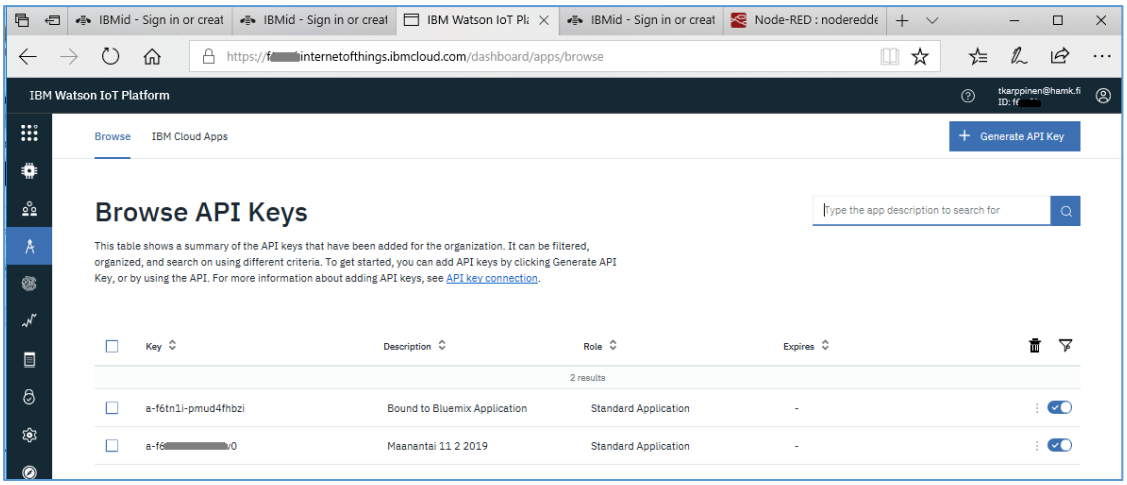
There is a reason for sending the message payload both as string and as object to the Watson IoT mqtt broker.

In the node “Send to IBM IoT” you have as Authentication options Quick Start and API Key. With my user rights it was possible to select Bluemix Service as seen on the picture.

In the node “Send to IBM IoT” there are parameters Device Type, Device Id, API Key and API Token. The same Device Type, Device Id, API Key and API Token needs to be defined similarly at the Watson IoT platform where the node would be connecting to.

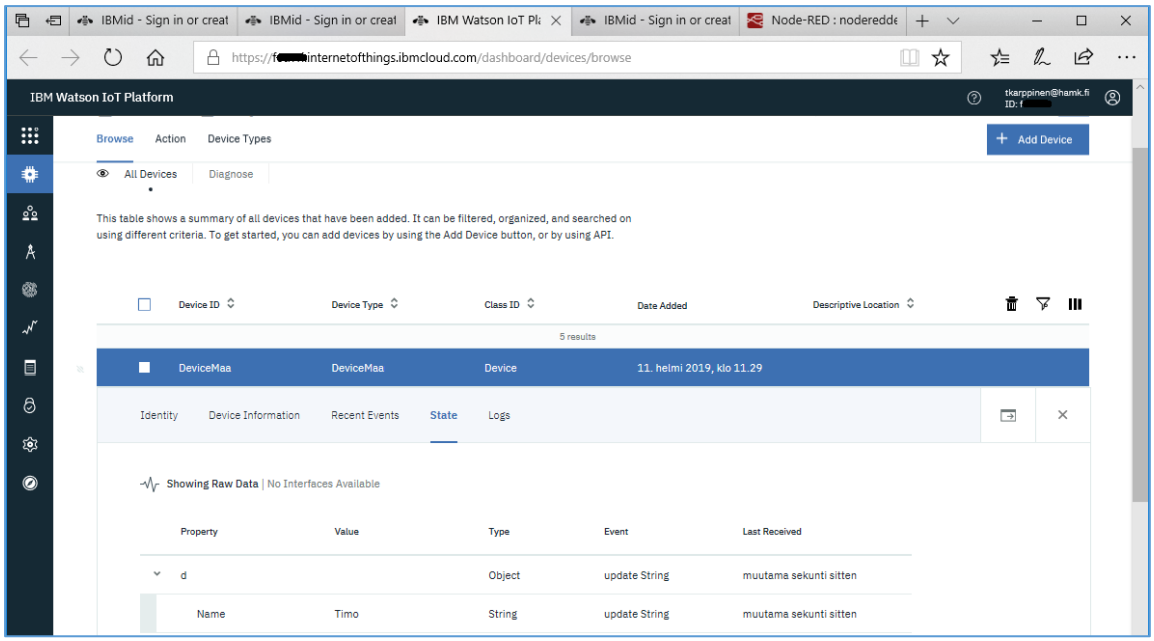
If the menu on left does not contain the blue coloured input node and output node “ibmiot”, please have a look on instructions 2 pages later in this document!

Fig. 3.2 API Key



For the API Key you can select the Standard Application. Please remember to save your new Token. Later you will not be able to get it visible in the page Browse API keys.

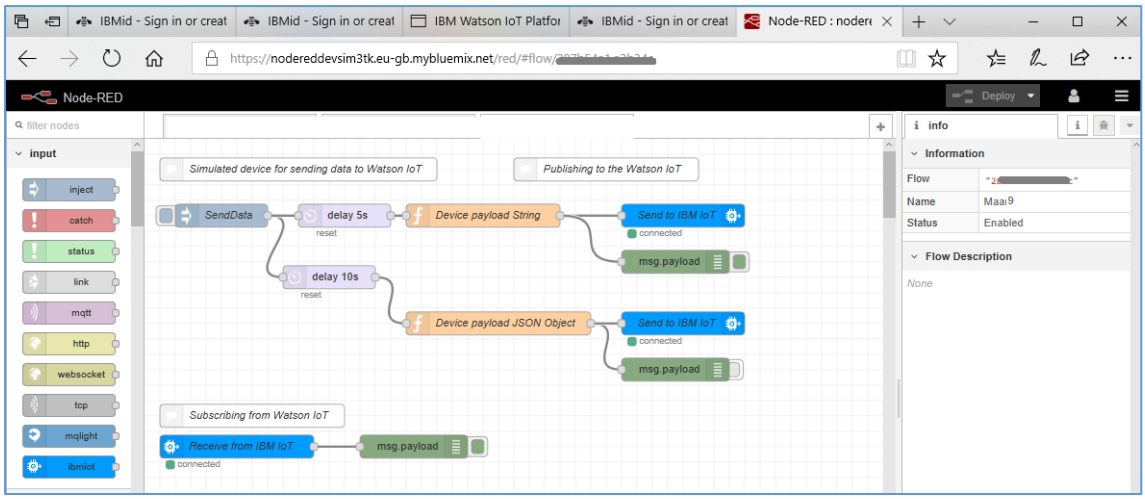
Fig. 3.3 The content of the message from the Node-RED flow visible on IoT platform.



The message and the message payload sent on the Node-RED flow will be visible on the Watson IoT platform page. In this document there is later an instruction how to create the necessary new device in the Watson IoT platform.

In the same flow editor we can create a node which will register as a subscriber to the mqtt broker in the Watson IoT and will read the mqtt messages.

Fig. 3.4 IBM IoT Subscriber.



Please import an ibmiot node from the input group. Similar parameters will be written as was earlier written for the ibmiot node imported from the output group.

Note 1 on year 2020, 2021, 2022 !

YOU PROBABLY DID NOT GET THE **IBM IOT** NODES IN THE NODE MENU ON THE LEFT ON YOUR NodeRED – the bright blue nodes in the picture above! We will correct the situation according to instructions published by IBM. This way we will save the available memory resource for the application.

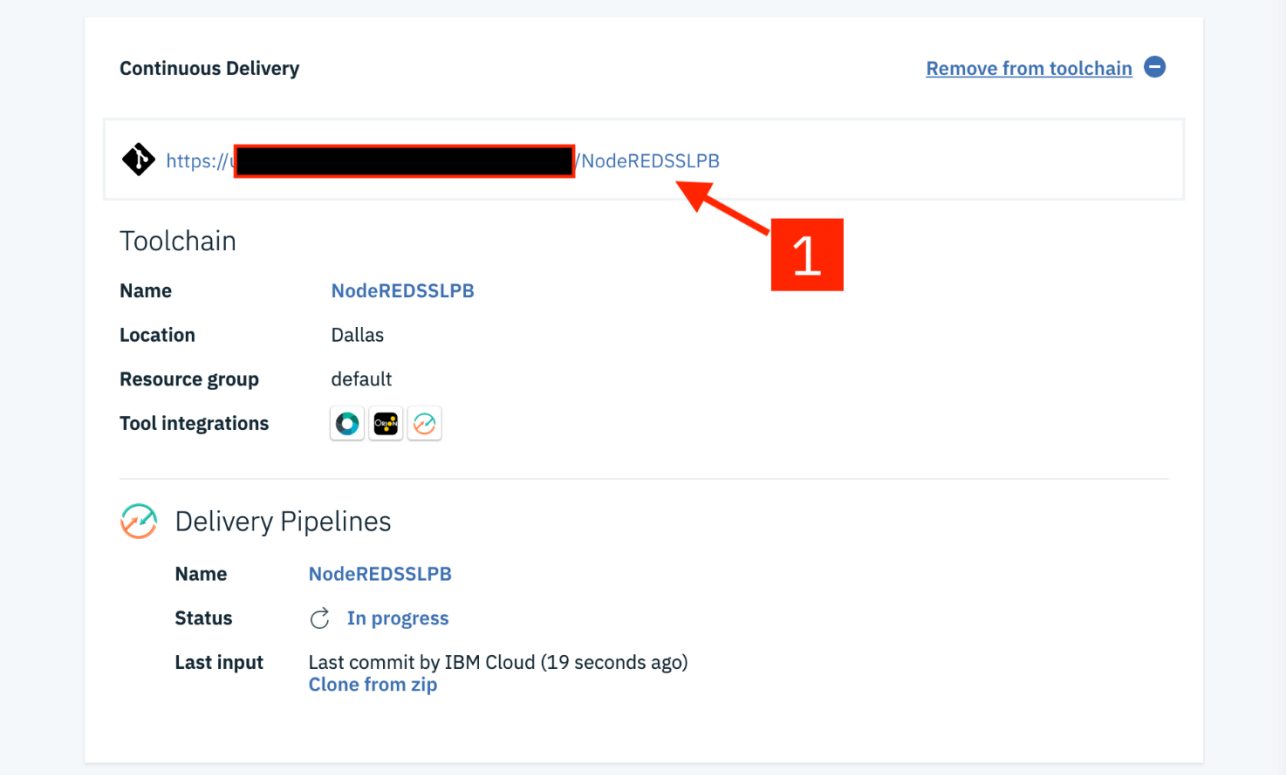
Add extra nodes to your Node-RED palette

Node-RED provides the palette manager feature that allows you to install additional nodes directly from the browser-based editor. This is convenient for trying nodes out, but it can cause issues due to the limited memory of the default Node-RED starter application.

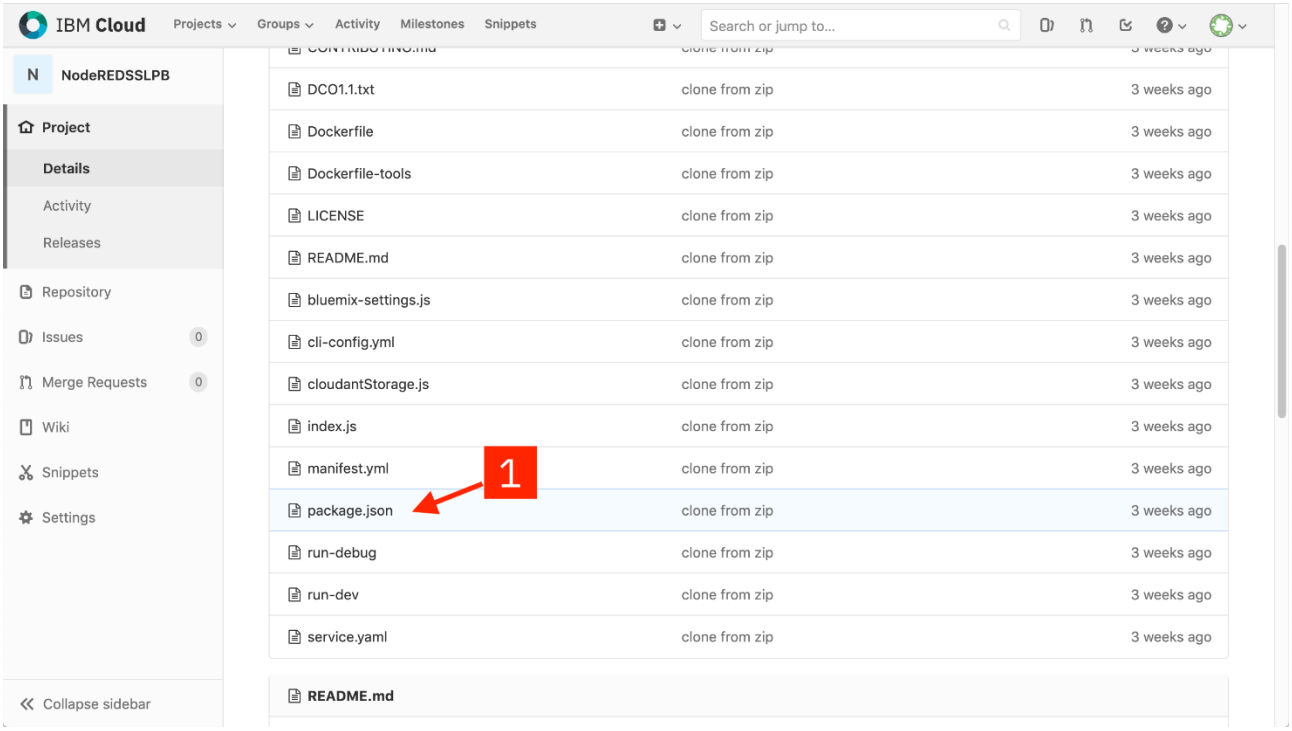
The recommended approach is to edit your application's package.json file to include the additional node modules and then redeploy the application.

This step shows how to do that in order to add the node-red-dashboard module and node-red-contrib-scx-ibmiotapp.

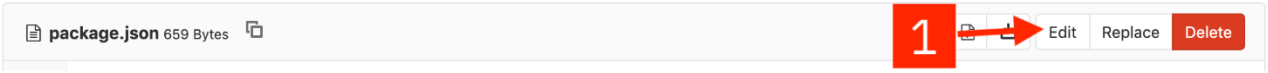
1. On your application's details page, click the url in the **Continuous Delivery** box. This will take you to a git repository where you can edit the application source code from your browser.



2. Scroll down the list of files and click on `package.json`. This file lists the module dependencies of your application.



3. Click the `Edit` button

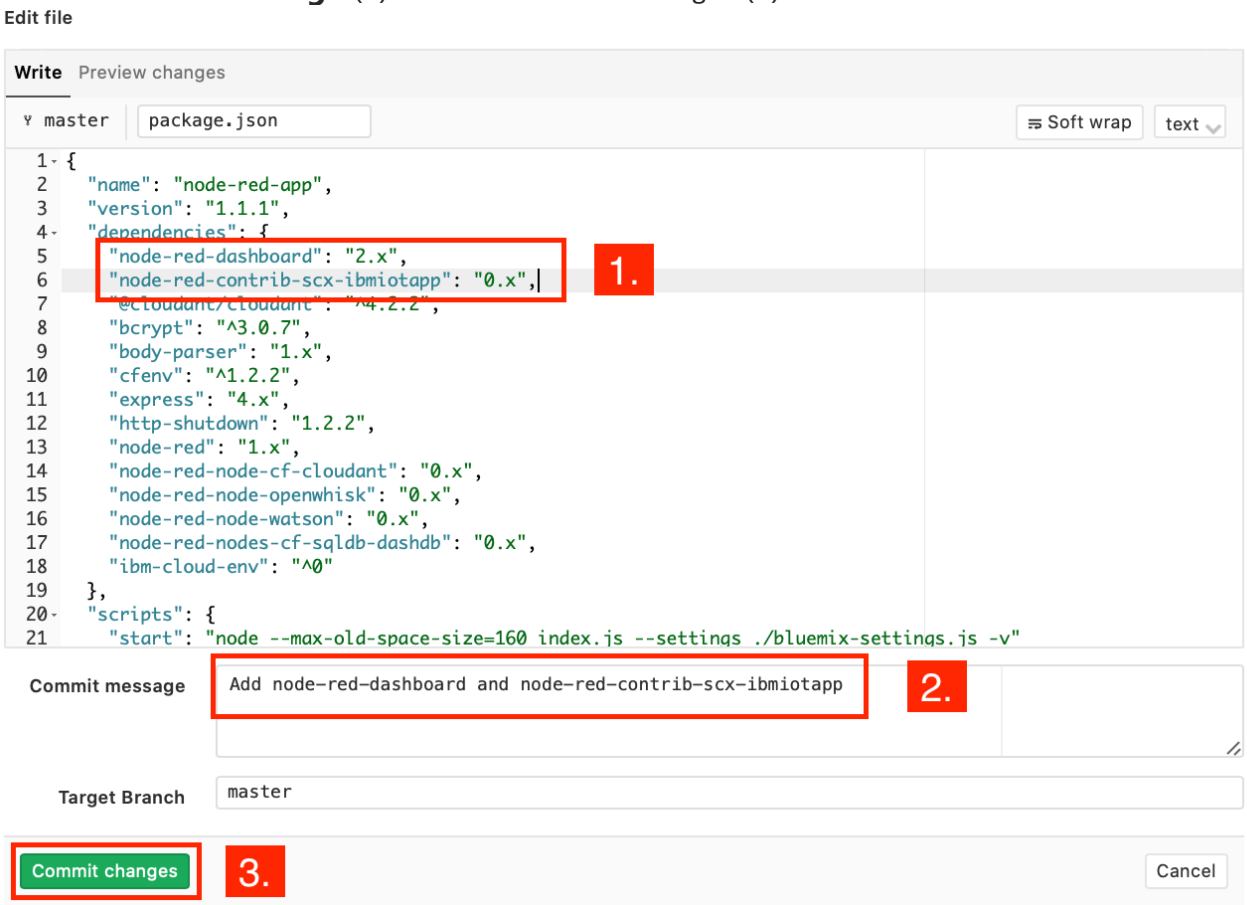


4. Add the following entry to the top of the dependencies section (1):

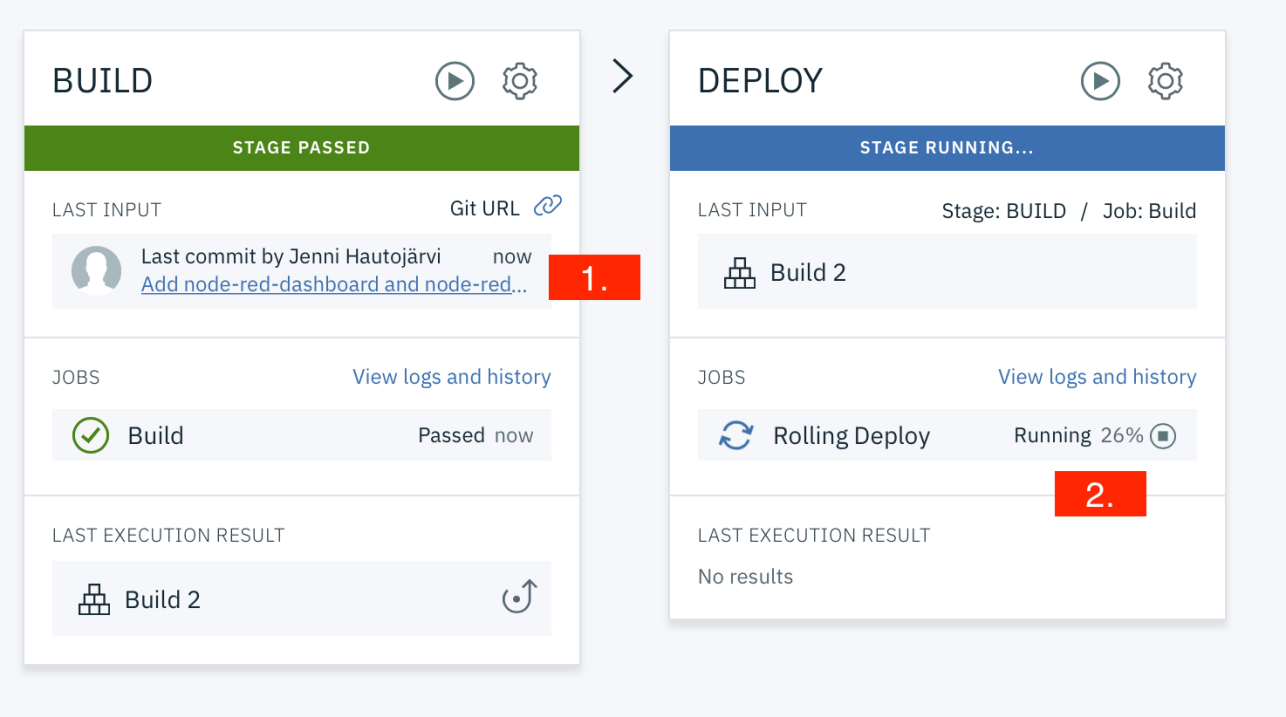
```
"node-red-dashboard": "2.x",
"node-red-contrib-scx-ibmiotapp": "0.x",
```

Note: Do not forget the comma (,) at the end of the line to separate it from the next entry.

Add a **Commit message** (2) and click `Commit changes` (3)



5. At this point, the Continuous Delivery pipeline will automatically run to build and deploy that change into your application. If you view the Delivery Pipeline you can watch its progress. The Build section shows you the last commit made (1) and the Deploy section shows the progress of redeploying the application (2).



6. Once the Deploy stage completes, your application will have restarted and now have the node-red-dashboard nodes preinstalled.

If the nodes are not showing in Node-red. Refresh browser. You might need to login to your application.

End of NOTE 1 ! / IBM Cloud 2020

Assignment "Commissioning 1"

Test the operation! Which one of the messages – JSON object or JSON string – will correctly authenticate in the IBM Watson IoT and will publish the payload?

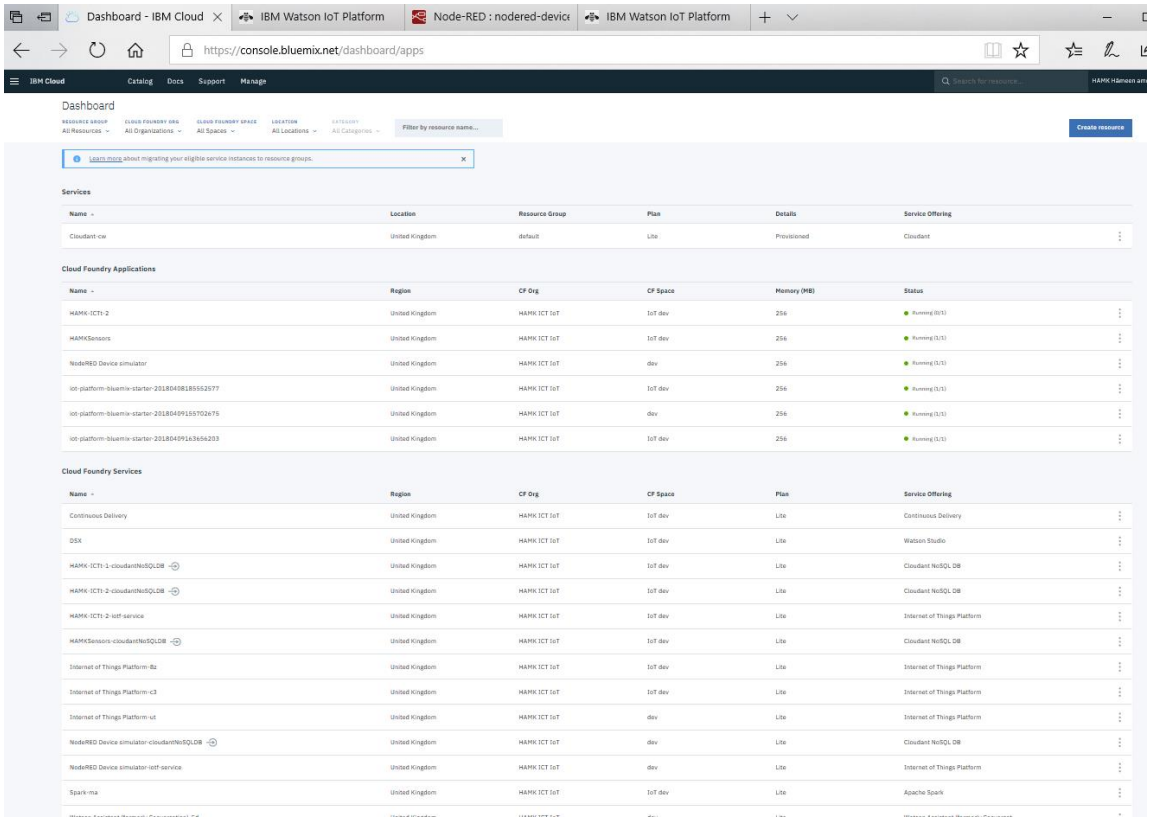
Finally – please save for yourself the address for your Node-RED flow. For example:

https://nodered-device-simulator.eu-gb.mybluemix.net/red/#flow/deb0d57.1c46528

Later you can access this flow editor page. But of course the access will be possible with correct user name and pass word only.

You have created a new service in the IBM Cloud platform. You will see the service instance by selecting the "three lines" in the top left corner of the IBM Cloud Dashboard page.

Fig 3.5 Console Dashboard.



You will recognise the service from the name you gave to it earlier in this commissioning task.

Assignment ”Commissioning 2”

If you created the Node-RED environment example by installing the “Internet of Things Platform Starter” you got as an example a room thermostat application. If you did not start from the “Internet of Things Platform Starter” you can still get the same function by importing a function node and filling it with the following java script code:

```
// Microcontrollers with sensors:
var area = ["Greenhouse1","Greenhouse2","Greenhouse3"];
// Array of pseudo random temperatures
var temp1 = [15,17,17.5,20,21.5,23,24,22.2,19,17];

// Array of pseudo random relative humidities
var humidity1 = [50,55,61,68,65,60,53,49,45,47];

// Counter to select from array.
var counter2 = context.get('counter2')||0;
counter2 = counter2+1;
if(counter2 > 2) counter2 = 0;
context.set('counter2',counter2);

// Counter to select from array.
var counter1 = context.get('counter1')||0;
counter1 = counter1+1;
if(counter1 > 9) counter1 = 0;
context.set('counter1',counter1);
```

```
// Create MQTT message in JSON
msg = {
  payload: JSON.stringify(
    {
      d:{
        "Area":area[counter2],
        "Temp" : temp1[counter1],
        "Humidity" : humidity1[counter1],
      }
    }
  )
};
return msg;
```

Please make to following changes in the application.

- Make the Send Data to send a new measurement value every two minutes.
- make some changes in the node **Device payload String** in the function written in javascript. The function is creating measurement values by selecting a value from a table.

Please add in an **ibmiot** type node - which is seen as the **Receive from IBM iot** node in the flow in the pictures above - a function for receiving the messages:

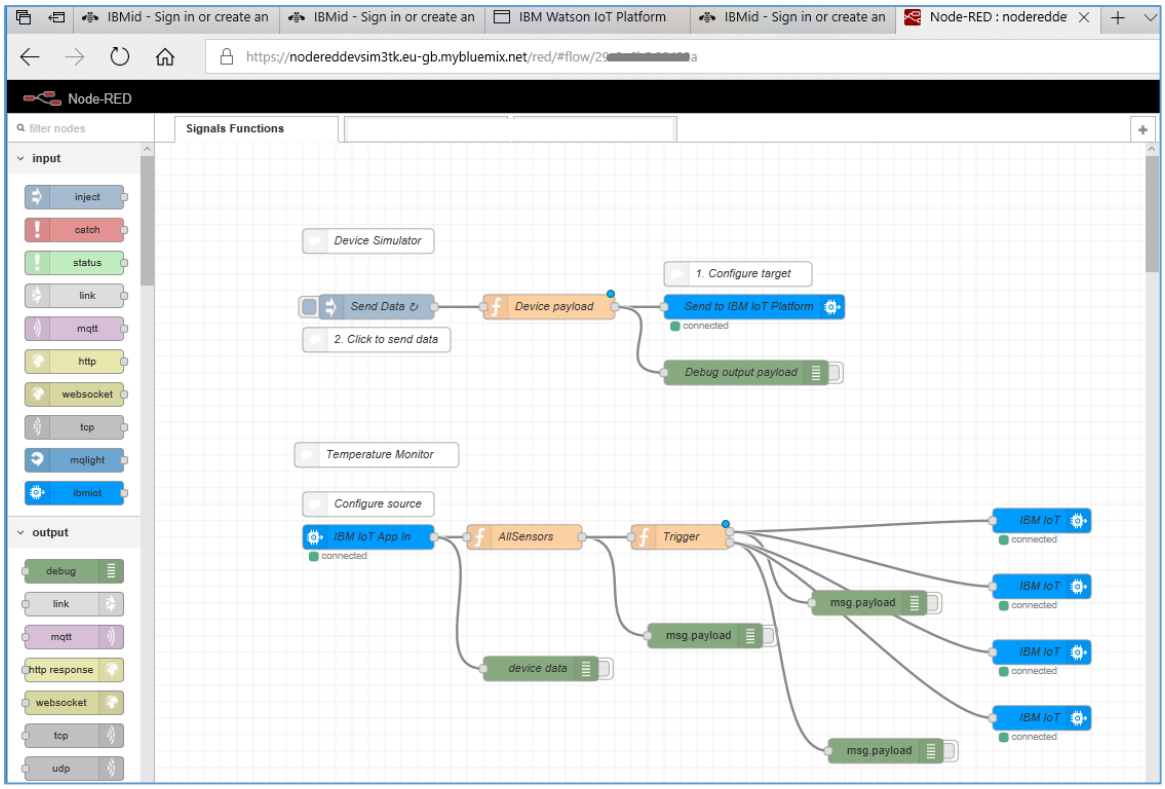
return
{payload:{"msgArea":msg.payload.d.Area,"msgTemp":msg.payload.d.Temp,"msgHum":msg.payload.d.Humidity}};

and after that an other **function** type node with a function:

```
var farea = msg.payload.msgArea;
var ftemp = msg.payload.msgTemp;
var fhum = msg.payload.msgHum;
var trigger = [false,false];
var msgOut = ["",""];

if (farea == "Greenhouse1"&&ftemp > 20)
{
    trigger[0] = true;
}
msgOut[0] = {payload:{"trcommand":trigger[0]}};

if (farea == "Greenhouse2"&&ftemp > 21)
{
    trigger[1] = true;
}
msgOut[1] = {payload:{"trcommand":trigger[1]}};

   return msgOut;
```

The flow can look like following:

Fig. 3.6 Modified flow.



4. A device created in the Watson IoT platform

We can create a device in the Watson IoT platform to correspond the device simulated in the Node-RED flow.

Please look at the subtitle Cloud Foundry Services. You will see there the new

- Cloudant NoSQL DB
- Internet of Things Platform

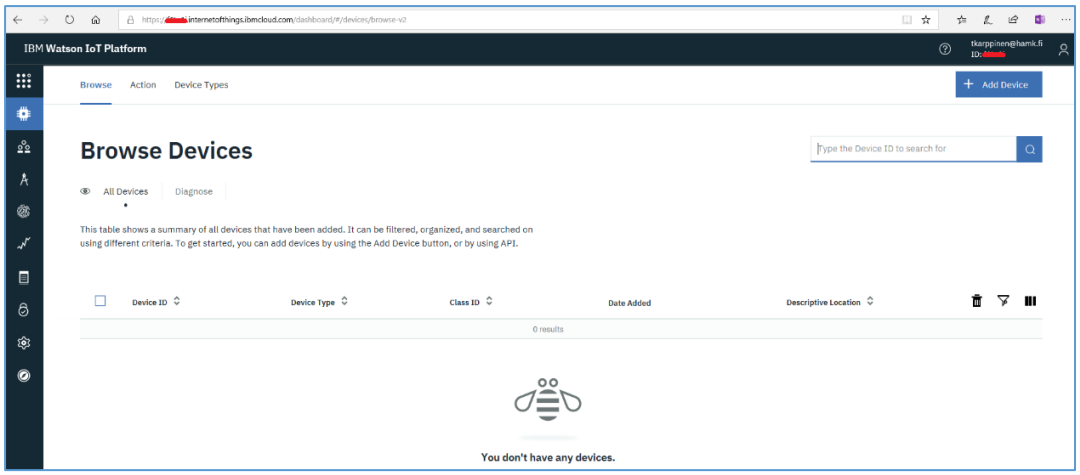
You can again recognise those form the name you gave earlier.

The database includes the NodeRED flow. The Internet of Things Platform is still an empty platform. Or if you added the NodeRED into IoT platforms from the earlier exercises you might have there already some device definitions.

Click the correct IoT platform. In the opening page please select Launch.

You will get an empty view or you might get a view with devices from earlier exercises.

Fig 4.1 Device view. Overwriten with red the ”Organization ID”.



Please write down or copy with copy paste the organization ID. In the picture this is overwritten with red colour.

Please create a new device with Add Device. Fill in the Device Type and Device ID fields exactly with the same text as in the Node-RED example.

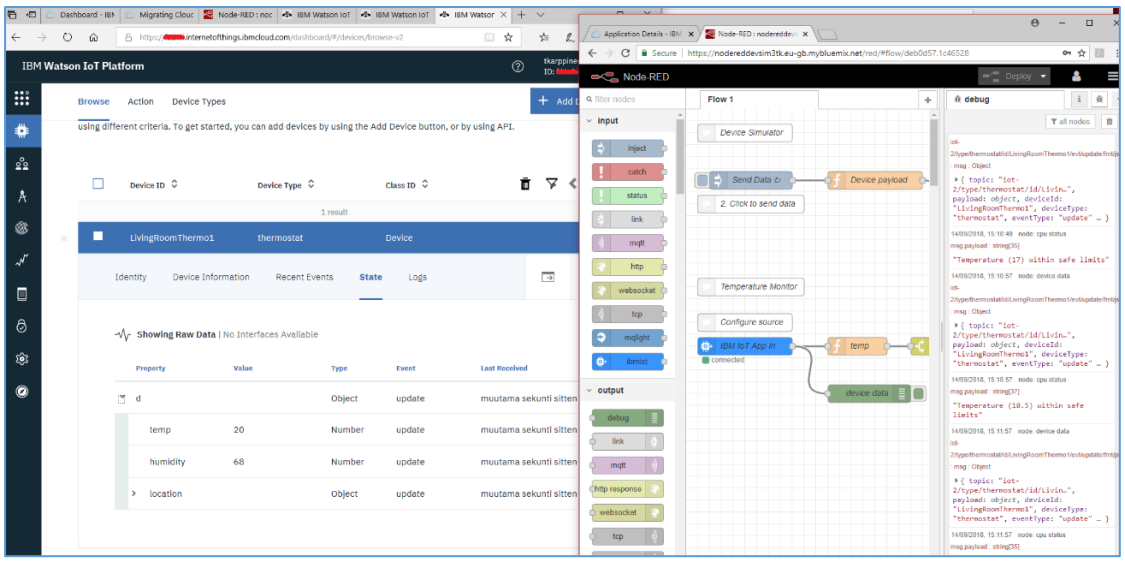
Device Type xxxxxxxxxxxxxxx

Device ID xxxxxxxxxxxxxxx_1

Continue with Next. You can leave the Device Information page as it is. Continue with Next. Let the Authentication Token be created automatically. Continue with Next and further with Done. Please write down in a text file the device information and the Authentication Token.

Return to device view. Click the row of your device.

Fig. 4.2 Simulated values are transmitted to IoT platform to the device view.



Assignment ”Exercise 1”

Please change the Node-RED flow:

The application field is traffic counting. The sensor device is an intelligent device with capabilities on image processing. The device can make a difference between cars, cyclists and pedestrians. Every 15 minutes it produces numbers on how many cars, cyclists and pedestrians have passed the observation point.

Every 15 minutes there will be new message with: Location, time, calculated amounts of traffic. The message has a JSON structure. JSON was used in the original example as well.

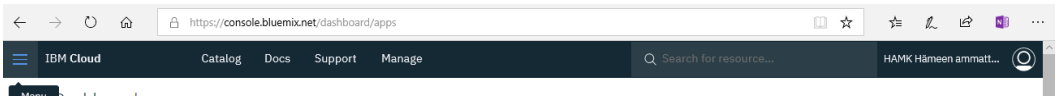
Please return: Screen capture with changes in the function in the node in the Node-RED flow. Screen capture with variables and values visible at the IoT platform device view.

5. How could you find again your Node-RED flow ?

To be able to continue the development you of course need to ne able to edit again your Node-RED flow.

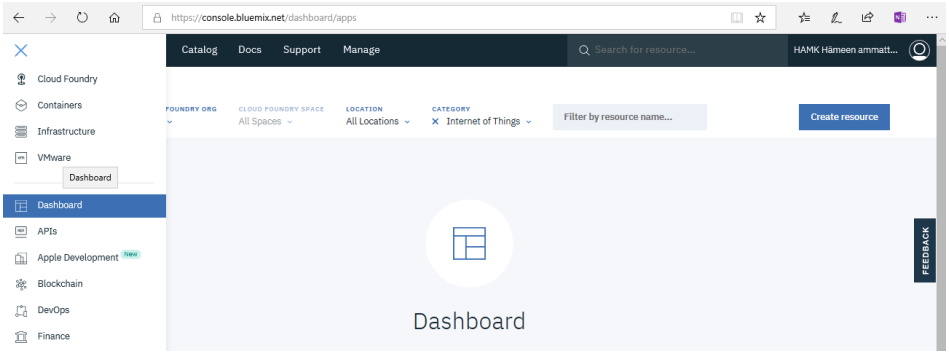
Please sign in into the **IBM Cloud**. There are a number of ways to find the right site to sign in. You can fore example search with word **IBM Cloud Catalog**. Or you can use the link https://cloud.ibm.com/catalog . And please sign in with your IBM ID.

Fig. 5.1 IBM Cloud Menu.



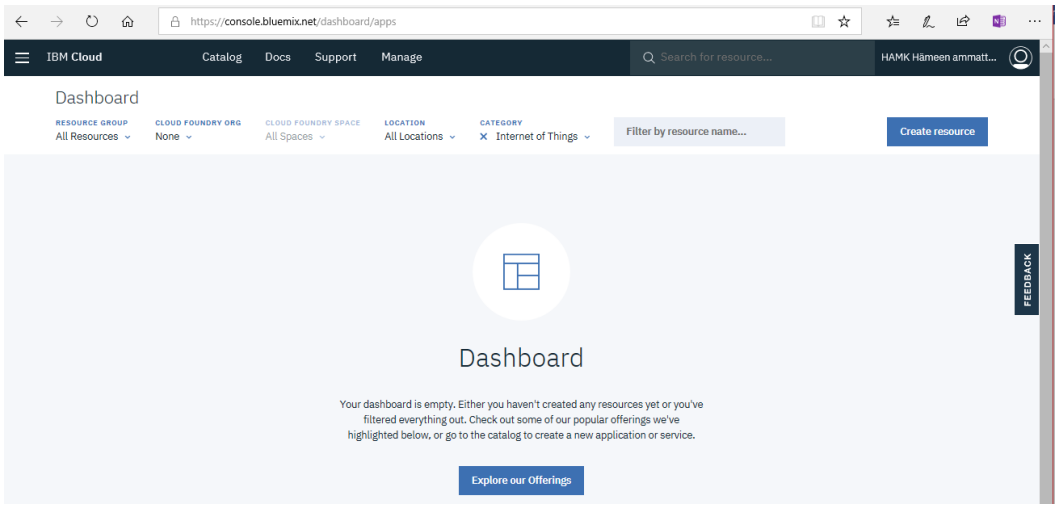
Please click the “**three lines**” visible on the top left corner.

Fig. 5.2 Dashboard



Select the **Dashboard** on the menu on left.

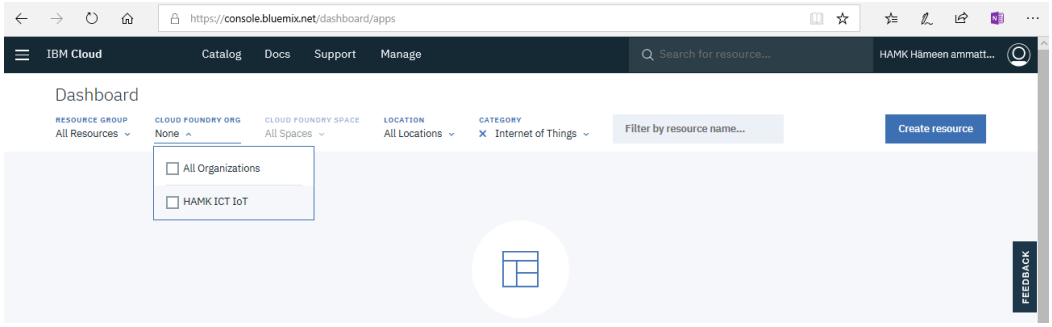
Fig. 5.3 Dashboard empty



Either you will get a full catalog of your development platforms or you might get an empty dashboard page.

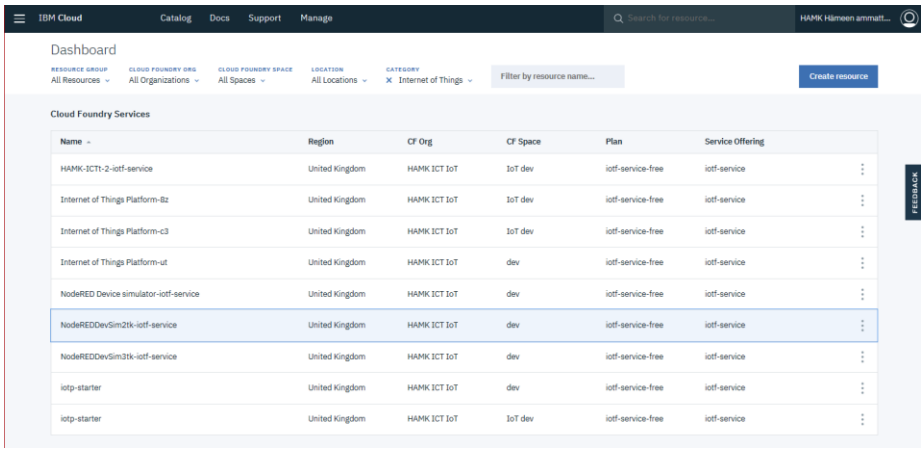
If you got an empty dashboard please select on the menu on top the **CLOUD FOUNDRY ORG** and further **the organisation id** where you developed the NodeRED example.

Fig. 5.4 Organization



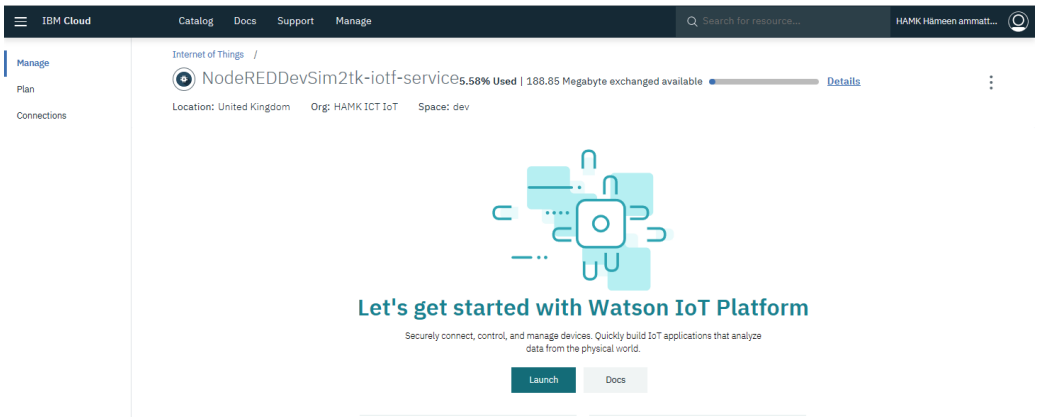
Now you should have your instances visible .

Fig. 5.5 Service instance



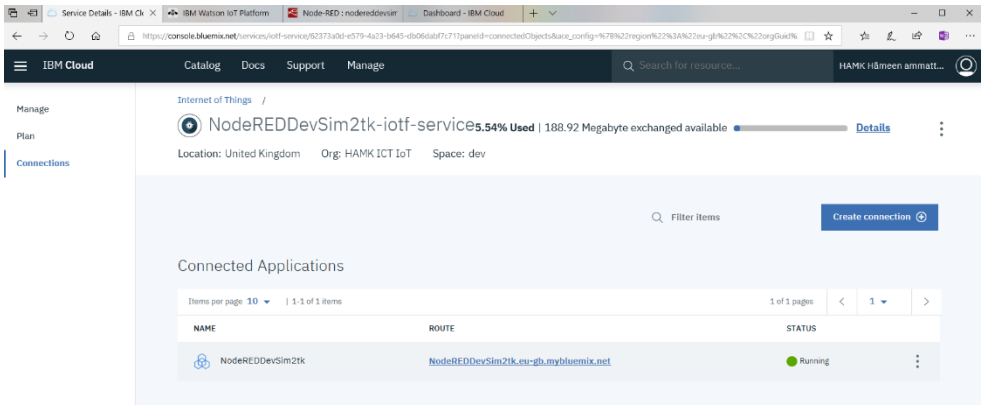
You can select the instance from the catalog and continue with **Launch**.

Fig. 5.6 The IoT platform is started.



On the subtitle **Connections** you can find your **NodeRED flow**.

Fig. 5.7 NodeRED flow



6. References in the GitHub

Good starting page for Node-RED related topics is

https://github.com/watson-developer-cloud/node-red-labs

There are introductions to different Node-RED nodes – Basic Examples.

https://github.com/watson-developer-cloud/node-red-labs/blob/master/basic_examples/README.md

There are exercises where nodes are combined and applications created – Advanced Labs.

https://github.com/watson-developer-cloud/node-red-labs/blob/master/advanced_examples/README.md

Application examples which you can use as starting point for your own development – Node-RED Starter Kits.

https://github.com/watson-developer-cloud/node-red-labs/blob/master/starter-kits/README.md

At the end of the page there is a definition for user rights and permissions to use the content in your own development.

The MIT License (MIT)

Copyright (c) 2015, 2016, 2017, 2018 IBM Corp.

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.