

Implementing Product Stock Sensors

Difficulty Level: Easy

Objective

Implement simulated temperature and humidity sensors for the product store in a smart factory scenario.

Achievements

The skills to be acquired at the end of this module:

- Working with multiple tabs (flows) in Node-RED
- Exporting and importing flows in NodeRED
- Editing the properties of existing flows and nodes

1. Using the Existing Warehouse Flow as a Baseline for New Sensor Implementation

In this module, the product stock sensors will actually be the same as the sensors we developed in the previous module for the warehouse sensors, except that the temperature and humidity values should be published under the MQTT topics **stock/temperature** and **stock/humidity**, respectively.


Therefore, instead of creating this functionality from scratch, we will use the output of the previous module (“Implementing Warehouse Sensors”) and see different ways of copying and editing that NodeRED flow.

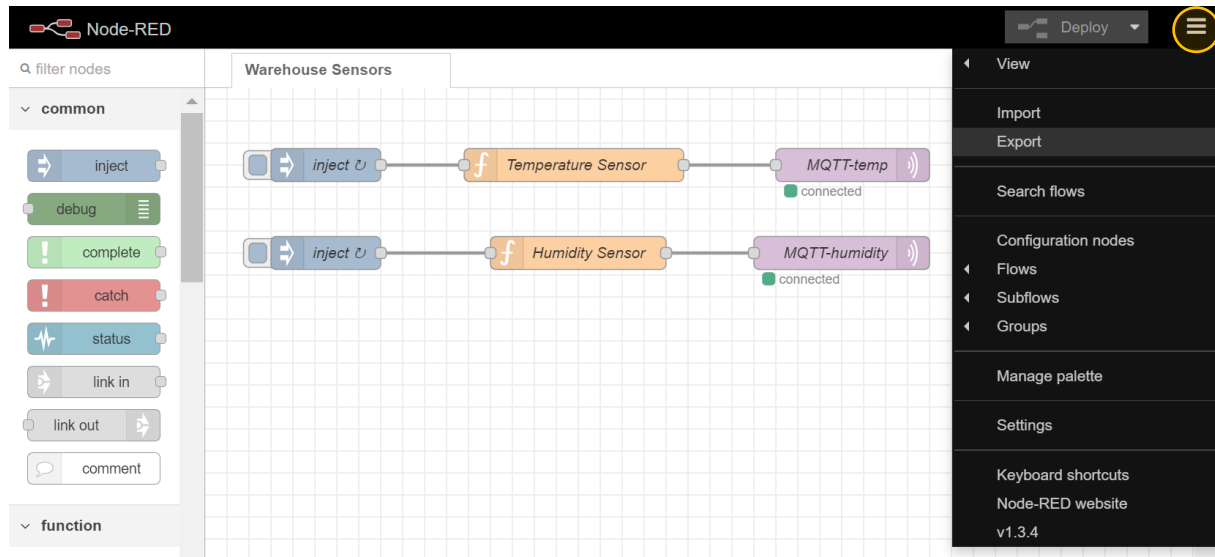
2. Exporting a flow in NodeRED

The process of “exporting” generally refers to the act of producing a transferable / sharable file out of a computer software, so that another user can then “import” that file to create the same environment on another software or in another copy of the same software.

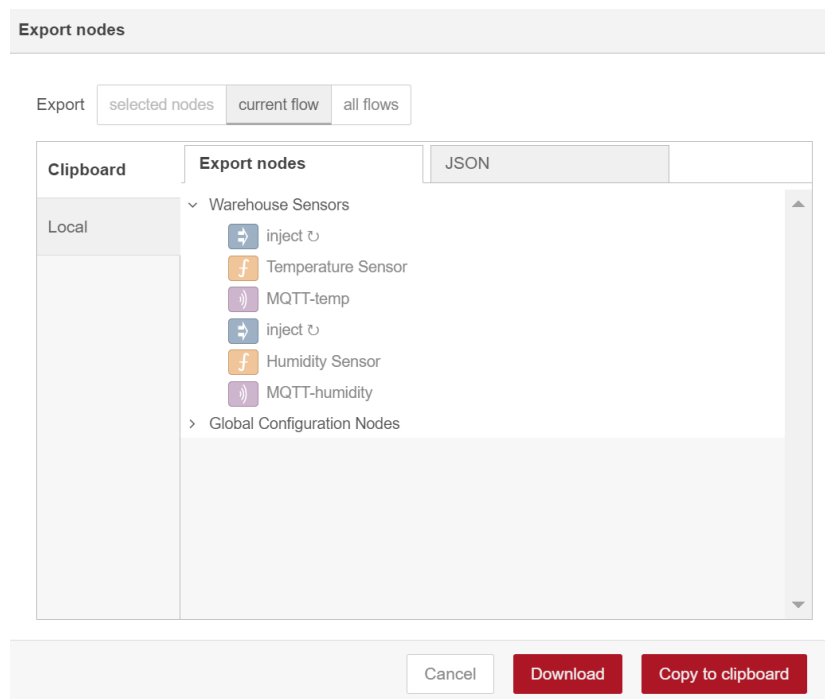
In our case, the “export” functionality of NodeRED is an important feature that allows you to share your application code easily with others as a single JSON file. You can also use this feature to reuse your own code in another application or in other parts of your same application.

We will now see how we can export the *warehouse flow* from the previous module, so that we can use it later to create the product stock sensors. In case the you have not worked on the previous module, so that the *warehouse flow* is not available to you, you can follow the remaining steps with any given flow in NodeRED and then use the provided [warehouse_sensors.json](#) file in the next steps.

First click on the icon  at the top right of NodeRED window, and then select the “Export” option from the menu, as depicted below.



Then the “Export nodes” window pops up, as shown below.



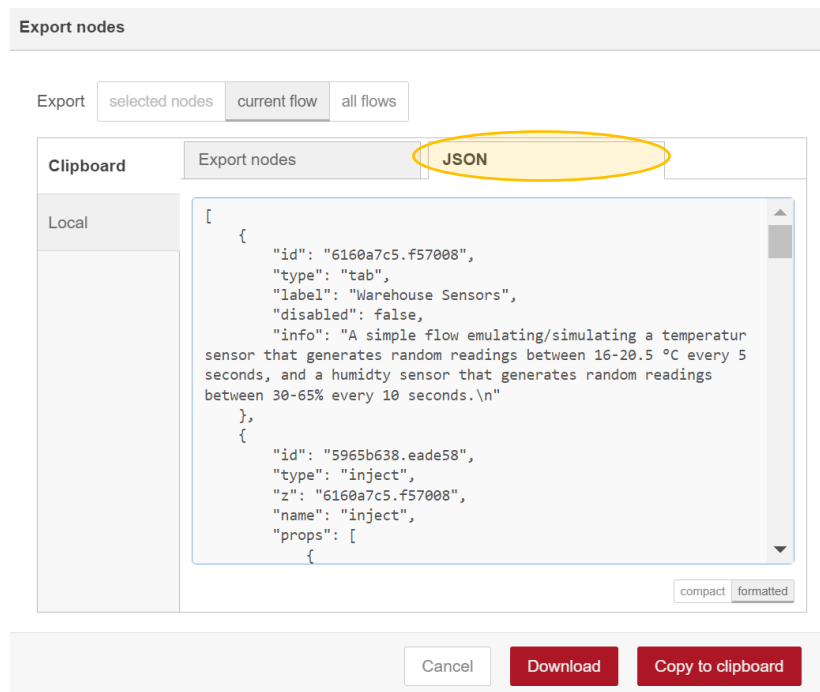
Here we notice several options:

- What to export: “selected nodes”, “current flow” or “all flows”
- Where to export: “Clipboard” or “Local”
- How to export: “Download” or “Copy to Clipboard”

The option of “Local” saves the flow in an internal “library” in NodeRED, instead of providing the option of copying the JSON document to the clipboard or download it. This library can then be used similarly

to import the saved flow. As the name implies, this method only works locally in the same NodeRED instance and cannot be used to share your application/flow with others.

In the default view of the export window, we see the list of nodes in the current flow that will be exported. We can also switch to the JSON view to preview the description of the flow and all nodes in json file format. This shows what exactly is copied into the clipboard or downloaded, if we choose one of those options:

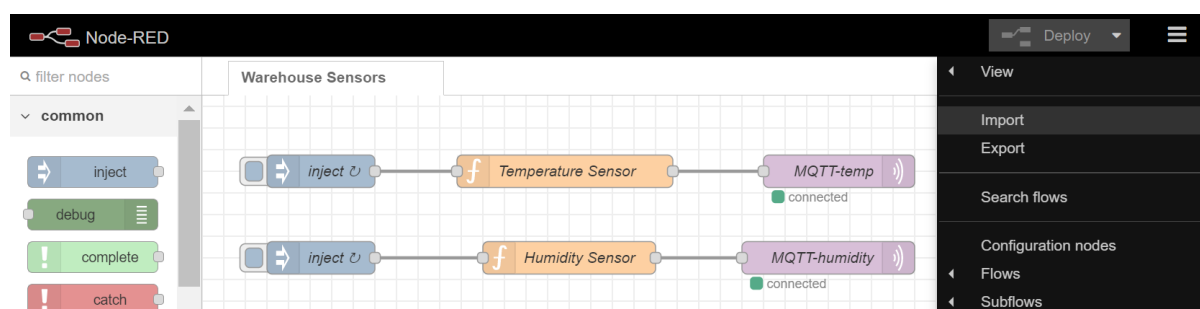


Now go ahead and click on “Download” or “Copy to clipboard” to export the current flow.

3. Importing a flow in NodeRED

We will now import the flow that was exported in the previous step into our application, so that we create a copy of the sensors from the warehouse flow and edit those to create a new pair of sensors for the product stock.

Open the Node-RED menu again, as in the previous step, and then click on “Import” this time:



This brings up the “Import nodes” window, which looks similar to the export window:

Import nodes

Clipboard

Paste flow json or [select a file to import](#)

Local

Examples

Import to

current flow

new flow

Cancel

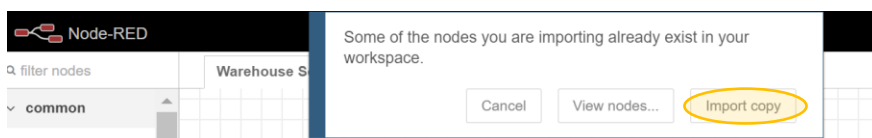
Import

Now, depending on whether you did in the previous step, follow either of the two:

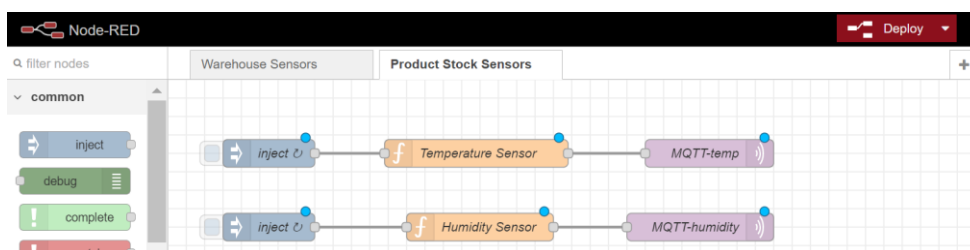
- If you downloaded a JSON file, click on “select a file to import” and choose that file;
- If you copied the JSON content to the clipboard, simply paste it into the red colored text area.

Then, click on “Import” at the bottom right. (Note that the option of “current flow” versus “new flow” is only relevant when we import individual nodes only. Since we are currently importing a whole flow, it will be added as a new flow in any case.)

You will get a warning message that the nodes you are importing already exist in your workspace. This is normal and you can simply proceed by clicking on “Import copy”.

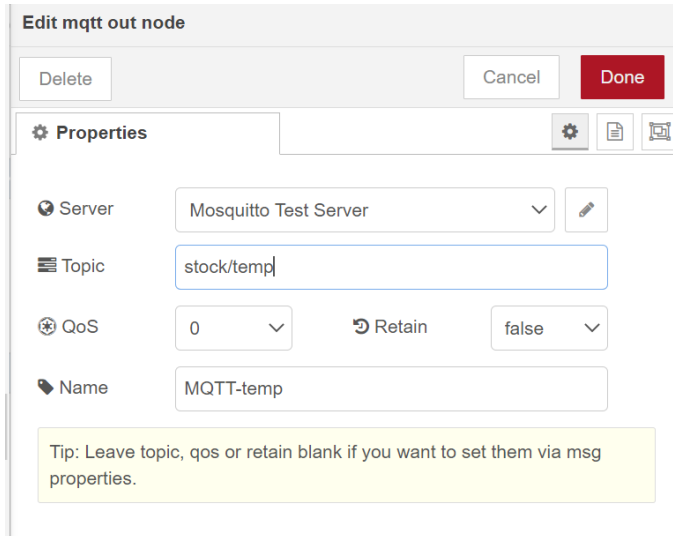


Now we should have a second flow in our NodeRED editor with the same name. Double click on the **second flow/tab** title to edit its settings. Change its name from “Warehouse Sensors” to “Product Stock Sensors”. Now your flows should look like the following:



The only change we need to do now is to edit the two MQTT-out nodes, so that the sensor readings are sent to the MQTT topics “stock/temp” and “stock/humidity”.

For this, double click on the “MQTT-temp” node and change the “Topic” field accordingly:



Edit mqtt out node

Delete Cancel Done

Properties

Server: Mosquitto Test Server

Topic: stock/temp

QoS: 0 Retain: false

Name: MQTT-temp

Tip: Leave topic, qos or retain blank if you want to set them via msg properties.

Similarly, double click on the “MQTT-humidity” node and change the “Topic” field to “stock/humidity”.

That is it, congratulations! When you click on “Deploy”, your two flows will both be running, periodically sending the temperature and humidity values in the parts warehouse and product store to the MQTT broker.

In the next module we will see how to read those sensor data and visualize them in a dashboard.