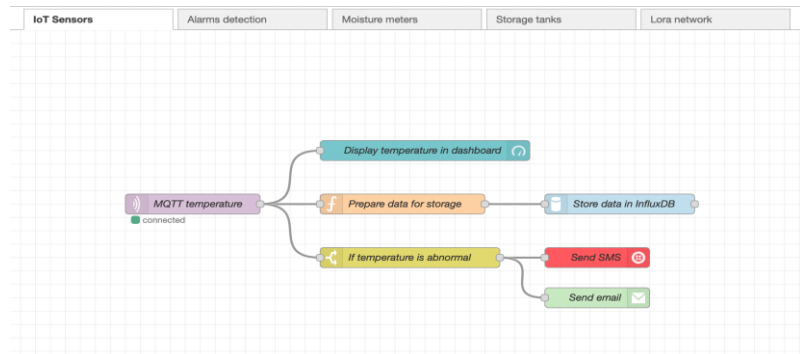


ACTIVITE 2

Prise en main de Node-Red

Qu'est-ce que Node-Red

Node-RED est un outil open-source de développement visuel conçu par IBM et est désormais maintenu par la communauté open-source. Node-RED est principalement utilisé pour créer des applications d'automatisation et d'Internet des objets (IoT).



Les points forts de Node-Red

Node-RED utilise un modèle de programmation visuelle basé sur des nœuds (nodes) et des connexions. Les développeurs peuvent créer des flux (flows) en reliant des nœuds qui représentent différentes actions ou fonctions.

Nœuds

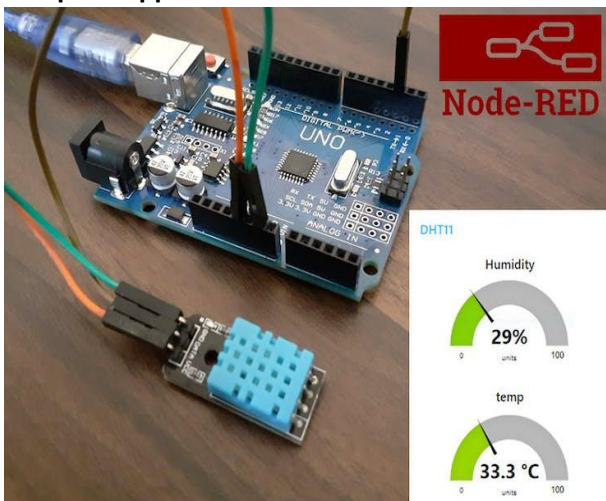
Les nœuds sont des blocs de construction fonctionnels, chaque nœud effectue une tâche spécifique, comme la lecture d'un capteur, la manipulation de données, l'envoi de courriels, etc.

Protocoles et services

Node-Red prend en charge de nombreux protocoles et services, ce qui facilite l'intégration avec des appareils IoT, des bases de données, des services web, etc.

Interface Web : Node-RED est accessible via un navigateur web.

Exemple d'application



Node-Red communique avec une carte Arduino (Serial, Bluetooth, WIFI, ...) pour récupérer les informations d'un capteur de température et d'humidité.

Ces valeurs seront transmises à un cloud et aussi affichées sur un tableau de bord graphique.

Sur le tableau de bord, on pourra ajouter des données venant d'autres capteurs.

Node-Red permet aussi de traiter les données (calculs, tris, ...)

Installer Node-Red sous Windows (sauf si déjà installé)

Les instructions suivantes concernent seulement l'installation de Node-Red sous Windows 10 ou 11.

Etape 1

Téléchargez la dernière version LTS de Node.js depuis la page : <https://nodejs.org/en/>

Exécutez le fichier MSI téléchargé. L'installation de Node.js nécessite des droits d'administrateur local.

Etape 2

Ouvrez "Invite de Commande" (CMD) et exécutez la commande suivante :

node --version && npm --version

```
Administrateur : Invite de commandes
Microsoft Windows [version 10.0.22631.2792]
(c) Microsoft Corporation. Tous droits réservés.
C:\Windows\System32>node --version && npm --version
```

node --version

Nous donne la version de "Node" et donc nous confirme son installation correcte.

npm --version

Nous donne la version de "npm" et donc nous confirme son installation correcte.

Npm est un "logiciel" qui va permettre d'effectuer des mises à jour de Node (ajout de nœuds, ...).

Etape 3

Finalisez l'installation à l'aide de cette commande :

npm install -g --unsafe-perm node-red

Nous pouvons maintenant utiliser Node-Red.

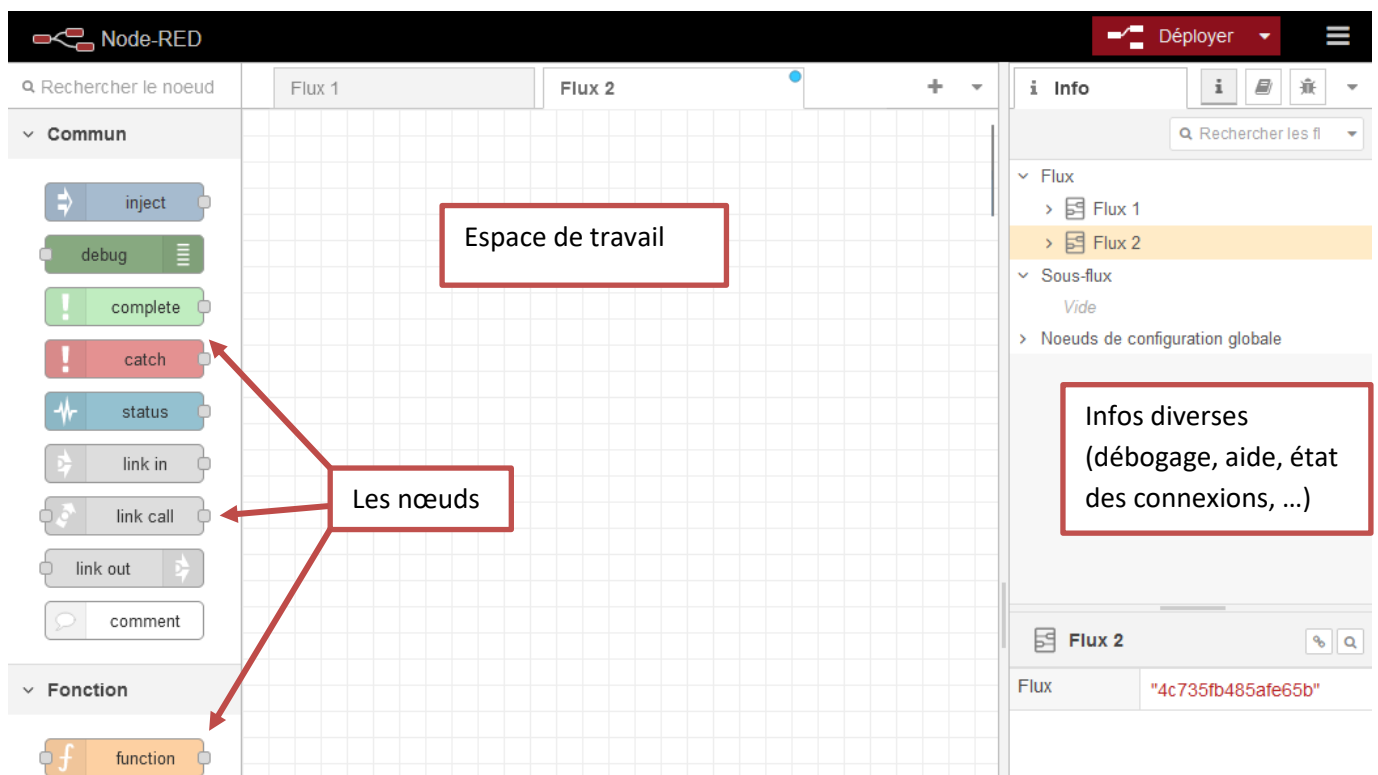
Etape 4

Dans "Invite de Commande" (en mode administrateur) tapez : **node-red**

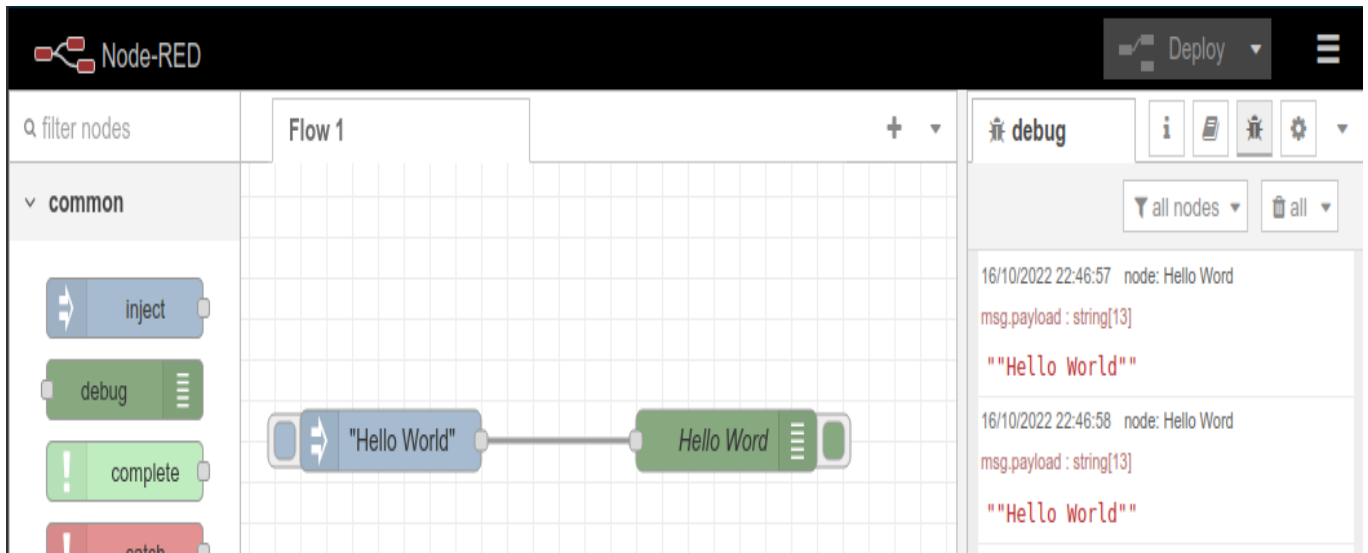
Le serveur Node-Red démarre, la fenêtre doit rester ouverte durant toute l'activité.

Pour vous connectez à ce serveur, tapez l'adresse suivante dans un navigateur : **127.0.0.1:1880**. Cette adresse IP correspond à l'adresse et au port utilisé par le serveur qui est précisée dans "Invite de Commande".

Prise en main de l'interface Node-Red



Programme 1 : Bonjour le monde

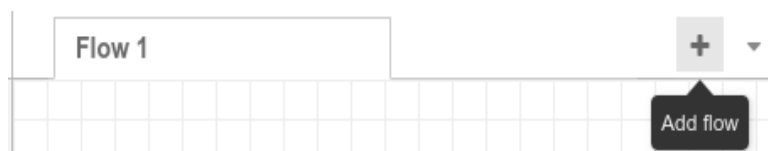


1 – Affichage d'un "Bonjour le monde" dans la console de débogage. Pour cela, vous devez effectuer les cinq étapes suivantes :

- Utilisez un nœud "inject".
- Modifier le nœud pour créer une chaîne de caractère "Bonjour le monde".
- Ajoutez un nœud debug.
- Terminez la config pour afficher le message "Bonjour le monde" dans la console de débogage.
- N'oubliez pas de cliquer sur le bouton "Déployer".
- Testez. A chaque clic sur le bouton d'inject, un message "Bonjour le monde" doit s'afficher.

Programme 2 : passage de valeur

2 – Créez un nouveau projet (flux) en ajoutant un onglet dans l'interface Node-Red :



Remarque

Pour supprimer un "flow", allez dans le [menu hamburger](#) → [Flows](#) → [Delete](#).

3 – Dans ce projet, il s'agit de relier à nouveau un nœud "inject" de type horodatage à "debug" et d'afficher la valeur de l'horodatage dans la console de débogage.



4 – Quelle est la signification de la valeur affichée dans la console ? (Consultez l'aide pour en savoir plus)

Cela correspond à l'heure actuelle en millisecondes depuis le 1er janvier 1970.



👉 5 – Utilisez un nœud "**fonction**" pour convertir cette valeur en Année-Mois-Jour Heures-Minutes-Secondes. Changez le nom "fonction" par "conversion".



Par défaut, les objets génèrent des messages dans un objet au format JSON, nommé `msg`.
`msg.payload` contient donc le message injecté par l'objet "**inject**".

👉 6 – Programmez l'objet "conversion" puis testez votre solution.

Onglet [Message reçu], ajoutez :

```
let date = new Date(msg.payload);  
msg.payload = date.toString();  
return msg;
```

🗒 7 – Demandez à l'IA la signification des lignes précédentes.

<code>let date = new Date(msg.payload);</code>	Cette ligne crée une nouvelle variable <code>date</code> en utilisant l'objet <code>Date</code> intégré de JavaScript. L'objet <code>Date</code> permet de travailler avec des dates et des heures. La valeur de <code>msg.payload</code> est passée en argument de l'objet <code>Date</code> . Si <code>msg.payload</code> est un nombre, il sera interprété comme le nombre de millisecondes écoulées depuis le 1er janvier 1970 à minuit UTC. Si <code>msg.payload</code> est une chaîne de caractères, elle sera analysée comme une date et une heure.
<code>msg.payload = date.toString();</code>	Cette ligne remplace la charge utile du message (<code>msg.payload</code>) par une chaîne de caractères représentant la date et l'heure actuelles. La méthode <code>toString()</code> est appelée sur l'objet <code>date</code> pour convertir la date en une chaîne de caractères.
<code>return msg;</code>	Cette ligne renvoie le message (<code>msg</code>). Dans Node-RED, cela permet d'envoyer le message au nœud suivant dans le flux.

👉 8 – Testez cette solution. Vous devriez avoir la date et l'heure du jour.

Programme 3 : Météo

Dans l'activité qui suit, vous allez interroger un site web pour y récupérer les informations météo d'une ville (fichier au format JSON).

<https://www.prevision-meteo.ch/services/json/>

Par exemple vous pouvez consulter les informations concernant la ville de Lorgues à l'adresse suivante :

<https://www.prevision-meteo.ch/services/json/lorgues>

Votre objectif est de réaliser une application (Dashboard) qui affiche les informations générales de la ville (**city_info**) et les conditions météo actuelles (**current_condition**).

JSONDonnées brutesEn-têtes

EnregistrerCopierTout réduireTout développer

city_info:

name:

"Lorgues"

country:

"France"

latitude:

"43.4949989"

longitude:

"6.3602781"

elevation:

"200"

sunrise:

"06:47"

sunset:

"18:40"

forecast_info:

latitude:

null

longitude:

null

elevation:

"144.8"

current_condition:

date:

"13.03.2024"

hour:

"10:00"

tmp:

15

wnd_spd:

8

wnd_gust:

0

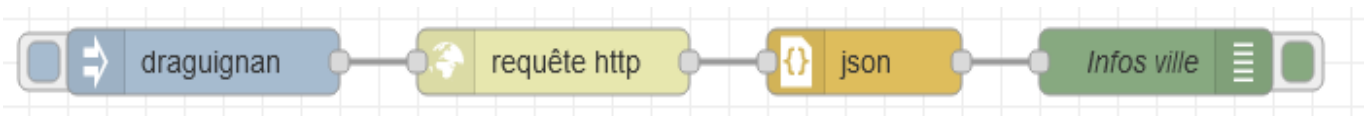
wnd_dir:

"E"

pressure:

1014.3

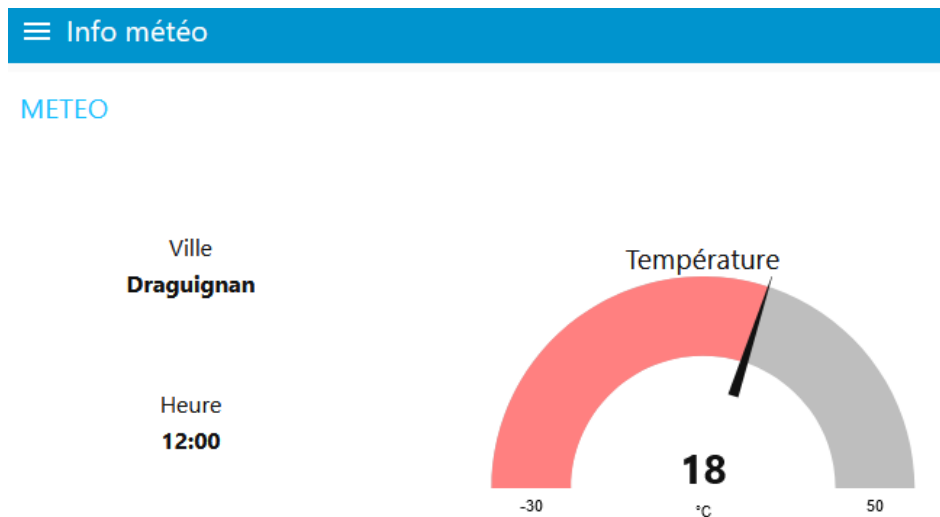
9 – Méthodologie :



- Utilisez un injecteur pour transmettre le nom de la ville (chaîne de caractère). L'injecteur génère une variable `{{payload}}` contenant le nom de la ville.
- Il faut transmettre la chaîne de caractère (le nom de la ville) à un nœud : **http request** pour générer une requête http. En retour, vous récupérerez un fichier json que vous devrez interpréter.
 - o Précisez URL.
 - o Ajoutez à la fin de l'URL la variable `{{payload}}` avec les doubles accolades.
- La réponse étant au format JSON, vous utiliserez un nœud **JSON** pour l'interpréter.
- Vérifiez que l'action du nœud **JSON** est bien sur "Convertir entre chaîne JSON et objet".
- Ajoutez un nœud **debug** pour afficher les informations de la ville (Sortie = `msg.payload.city_info`)

- 👉 10 – Testez votre solution.
- 👉 11 – **Modification 1** : utilisez un autre noeud **debug** pour afficher les conditions météo actuelles de la ville.
- 👉 12 – **Modification 2** : utilisez plusieurs injecteurs pour afficher les infos des villes : Iorgues, Draguignan et Nice.
- 👉 13 – **Modification 3** : Modifiez les nœuds **debug** pour afficher seulement le nom de la ville, l'heure et la température correspondante.

L'étape suivante consiste à présenter les résultats dans un "dashboard" comme ci-dessous.

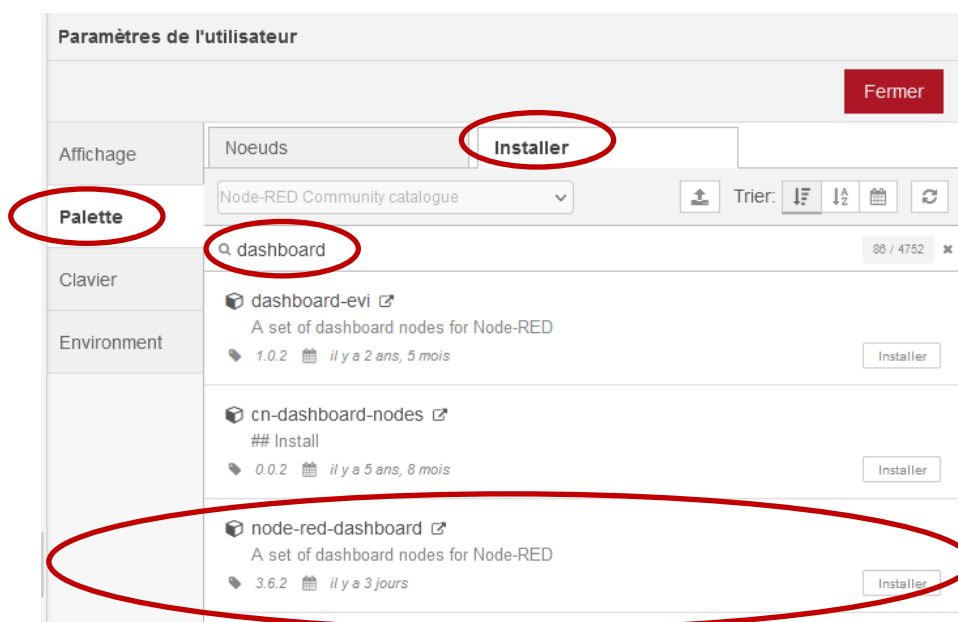


On va donc ajouter une palette de nœuds "dashboard" à Node-Red.

Création d'un tableau de bord

Pour créer un tableau de bord il faut d'abord ajouter une palette de nœuds "Dashboard".

- 👉 14 – Cliquez sur le menu hamburger (en haut à gauche) puis gérer les palettes. Installez la palette "node-red-dashboard".

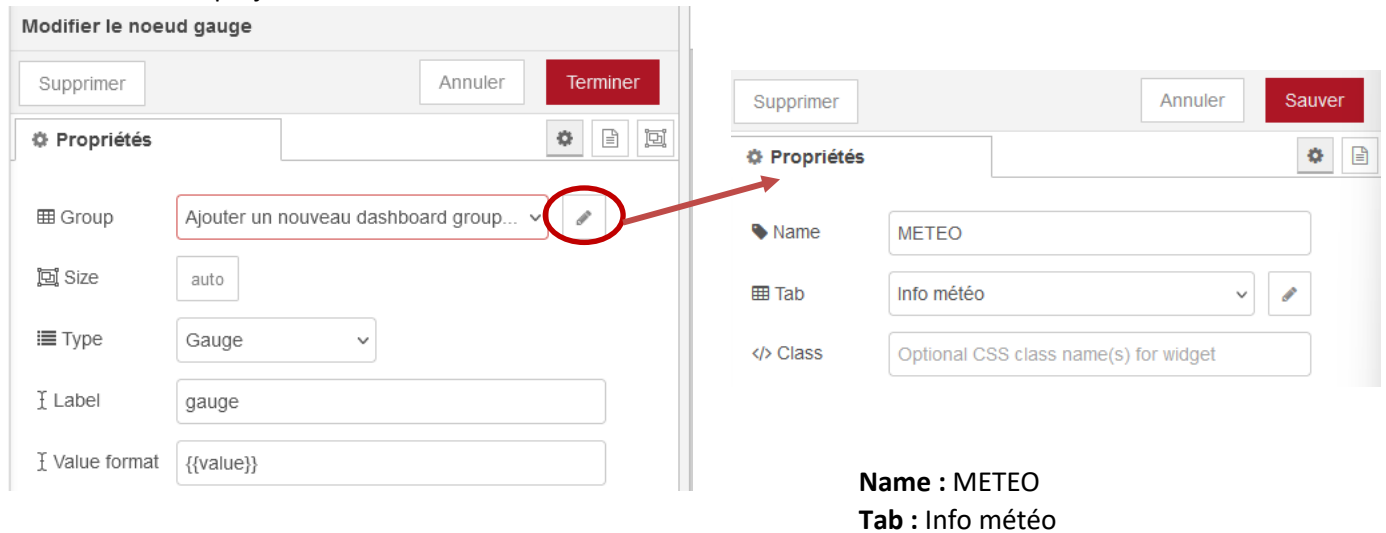


Vérifiez dans le menu "Nœuds" la présence des nœuds Dashboard.

15 – L'objectif est de visualiser la valeur de la température d'une ville. Placez un nœud "Gauges" dans l'espace de travail, reliez la sortie du nœud "json" avec le nœud "gauge". Cliquez sur déployer pour valider (et sauvegarder).

16 – Double cliquez sur le nœud "Gauge" pour le configurer.

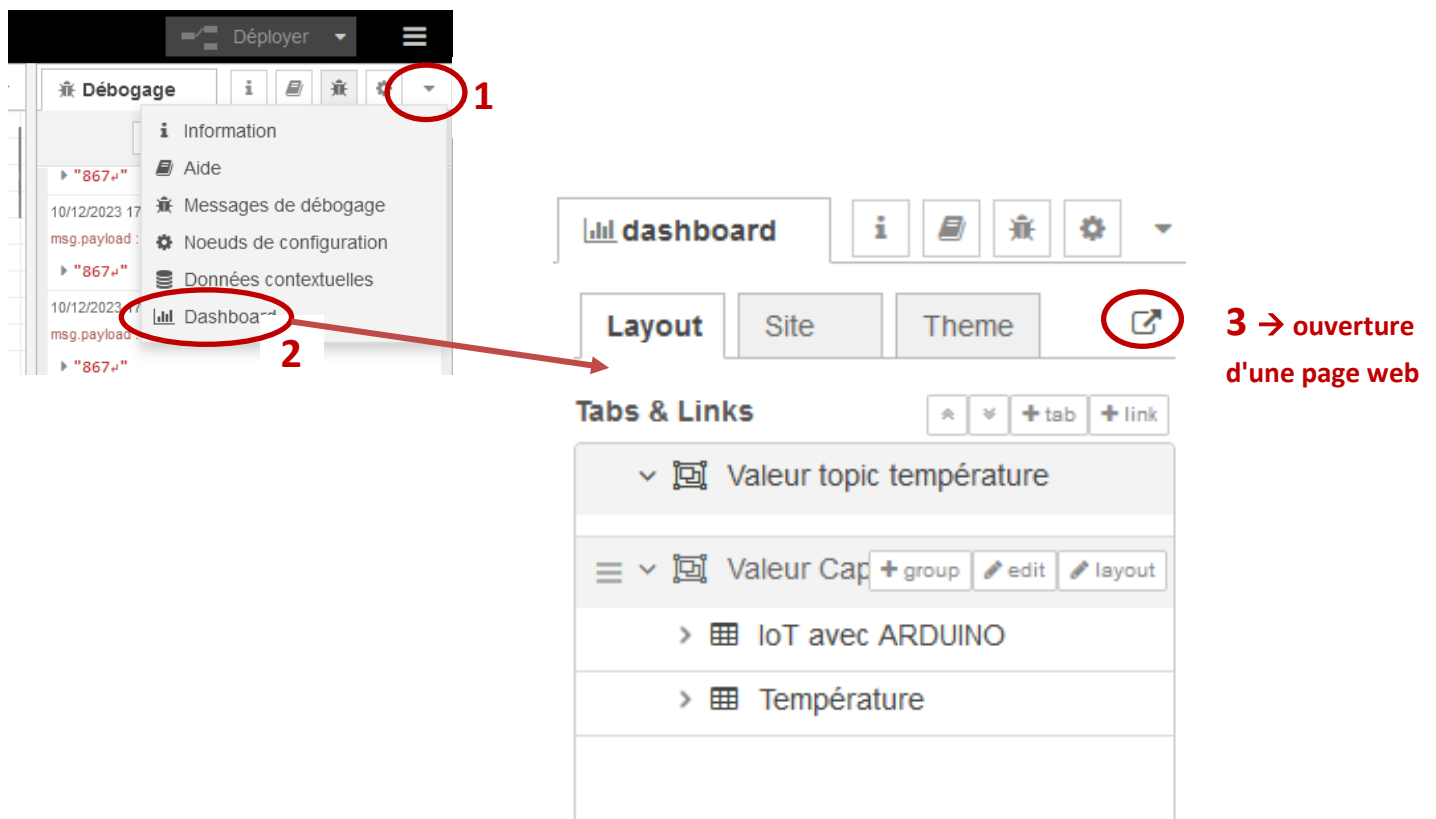
17 – La première étape consiste à définir un groupe (Tab) dans lequel on regroupera les gauges (Name) commun à notre projet.



Name : METEO
Tab : Info météo

Pensez à cliquer sur "Sauver", "Terminer" puis "Déployer".

18 – Allons voir le résultat.



1 → Déployer

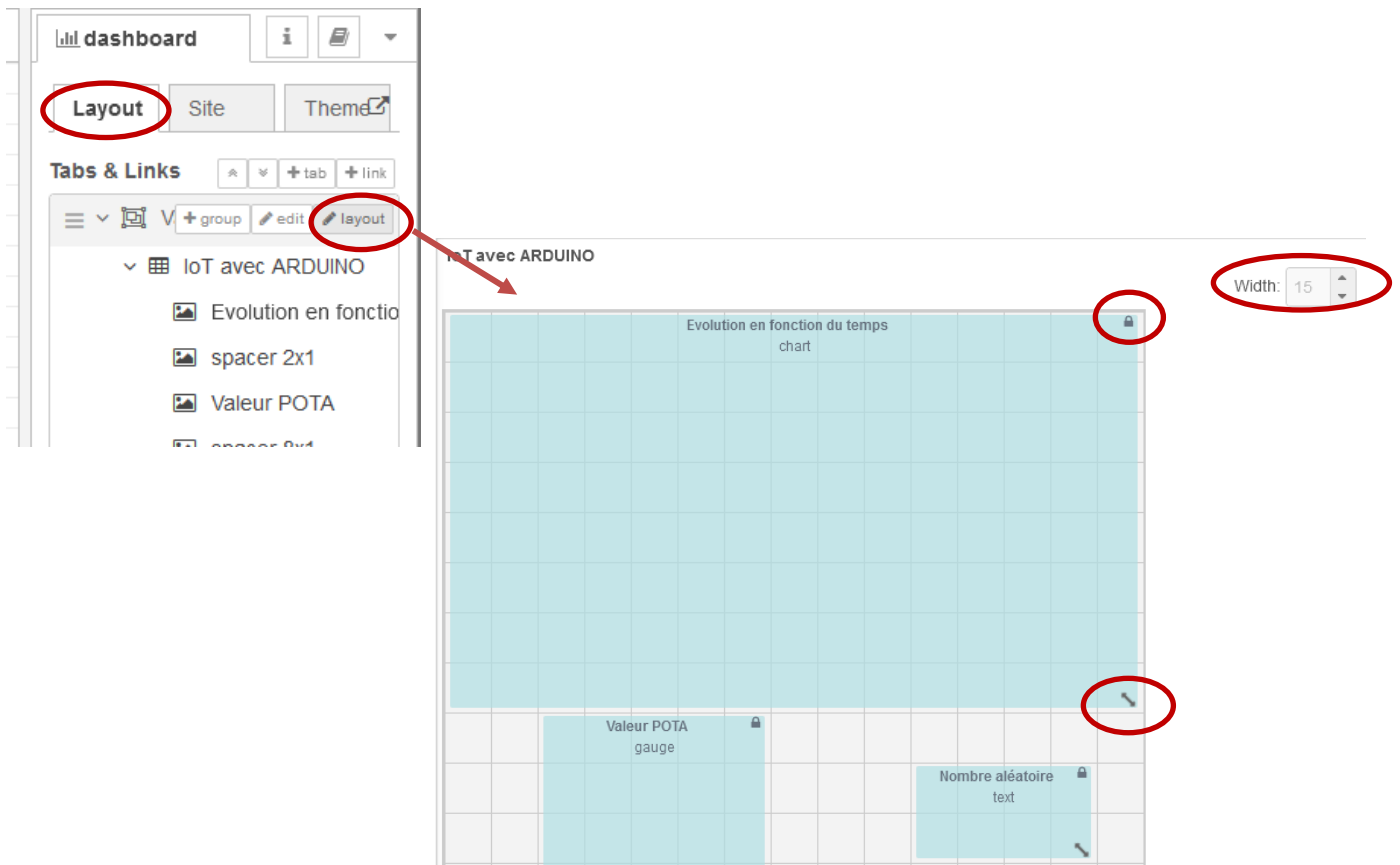
2 → Dashboard

3 → ouverture d'une page web

19 – Waouh, c'est beau !

Mais il y a un problème d'échelle : valeur à afficher = 22 et valeur max de la jauge : 10 !!
Allez modifier les paramètres de la "gauge" pour avoir un affichage correct. Ajoutez un titre, changez les couleurs, modifiez la taille, ...

20 – Modifiez la disposition du tableau de bord.



21 – **Modification 4** : Ajouter au niveau du Dashboard une zone texte pour entrer le nom de la ville et un bouton valider.

