

Node-Red

TP : INITIATION A NODE-RED ET MONITORING IoT

Objectif : Apprendre à installer Node-RED, manipuler des flux de données (JSON/Payload) et concevoir une interface de monitoring (Dashboard).

1. Installation de l'environnement

Nous utilisons Docker pour garantir un environnement isolé et persistant.

- **Action :** Exécutez la commande suivante dans votre terminal:

```
docker run -it -p 1880:1880 -v node_red_data:/data --name mynodered nodered/node-red:latest
```

- **Vérification :** Une fois le conteneur lancé, ouvrez votre navigateur à l'adresse <http://localhost:1880>.

2. Concepts de base (Flux "Hello World")

Après avoir visionné la **Vidéo 1** : À télécharger sur <https://github.com/bouhenic/FormationIOT/journée1/TpNodered>

Reproduisez le flux et **répondez** aux questions suivantes pour valider votre compréhension :

Compréhension du flux

Question	Réponse
A quoi sert le nœud d'injection ?	
A quoi sert le nœud de debug ?	
Que signifie la présence d'un point bleu sur un nœud ?	
Comment traduit-on « payload » en français.	
Quelle signification a la couleur grise du bouton « Deploy » ?	

Défi rapide : Créez un flux qui affiche votre **Nom et Prénom** dans l'onglet debug toutes les **3 secondes**

3. Traitement de données avec Moment.js

L'objectif est de transformer un temps Unix (timestamp) en une date lisible. Utilisez les ressources suivantes :

- **Vidéos 2 & 3** : À télécharger sur <https://github.com/bouhenic/FormationIOT/journée1/TpNodered>
- **Documentation Moment.js** : Utilisez les "Tokens" (jetons) pour formater votre texte.

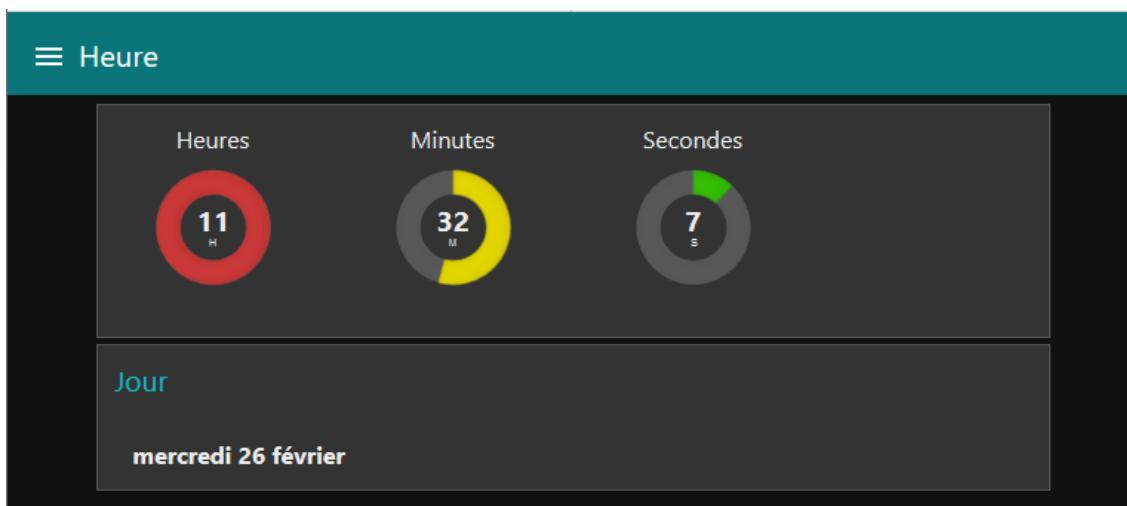
Node-Red

4. Création du Dashboard (IHM)

Modifiez votre interface pour qu'elle soit professionnelle et sombre (Style **Dark**).

Étapes de réalisation :

1. **Structure** : Ajoutez un groupe nommé "**Jour**" et insérez-y un nœud de type **Text**.
2. **Configuration** : Paramétrez le nœud **Moment** pour qu'il renvoie la date sous la forme : mercredi 26 février.
3. **Visualisation** : Ajoutez trois jauge pour les **Heures**, **Minutes** et **Secondes** comme illustré dans l'exemple.



Node-Red

Aide au formatage

	Token	Output
Month	M	1 2 ... 11 12
	Mo	1st 2nd ... 11th 12th
	MM	01 02 ... 11 12
	MMM	Jan Feb ... Nov Dec
	MMMM	January February ... November December
Quarter	Q	1 2 3 4
	Qo	1st 2nd 3rd 4th
Day of Month	D	1 2 ... 30 31
	Do	1st 2nd ... 30th 31st
	DD	01 02 ... 30 31
Day of Year	DDD	1 2 ... 364 365
	DDDo	1st 2nd ... 364th 365th
	DDDD	001 002 ... 364 365
Day of Week	d	0 1 ... 5 6
	do	0th 1st ... 5th 6th
	dd	Su Mo ... Fr Sa
	ddd	Sun Mon ... Fri Sat
	dddd	Sunday Monday ... Friday Saturday
Day of Week (Locale)	e	0 1 ... 5 6
Day of Week (ISO)	E	1 2 ... 6 7
Week of Year	w	1 2 ... 52 53
	wo	1st 2nd ... 52nd 53rd
	ww	01 02 ... 52 53
Week of Year (ISO)	W	1 2 ... 52 53
	Wo	1st 2nd ... 52nd 53rd
	WW	01 02 ... 52 53
Year	YY	70 71 ... 29 30
	YYYY	1970 1971 ... 2029 2030
	Y	1970 1971 ... 9999 +10000 +10001 Note: This complies with the ISO 8601 standard for dates past the year 9999
Week Year	gg	70 71 ... 29 30

Node-Red

	gggg	1970 1971 ... 2029 2030
Week Year (ISO)	GG	70 71 ... 29 30
	GGGG	1970 1971 ... 2029 2030
AM/PM	A	AM PM
	a	am pm
Hour	H	0 1 ... 22 23
	HH	00 01 ... 22 23
	h	1 2 ... 11 12
	hh	01 02 ... 11 12
	k	1 2 ... 23 24
	kk	01 02 ... 23 24
Minute	m	0 1 ... 58 59
	mm	00 01 ... 58 59
Second	s	0 1 ... 58 59
	ss	00 01 ... 58 59
Fractional Second	S	0 1 ... 8 9
	SS	00 01 ... 98 99
	SSS	000 001 ... 998 999
	SSSS ... SSSSSSSS	000[0..] 001[0..] ... 998[0..] 999[0..]
Time Zone	z or zz	EST CST ... MST PST Note: as of 1.6.0, the z/zz format tokens have been deprecated from plain moment objects. Read more about it here. However, they *do* work if you are using a specific time zone with the moment-timezone addon.
	Z	-07:00 -06:00 ... +06:00 +07:00
	ZZ	-0700 -0600 ... +0600 +0700
Unix Timestamp	X	1360013296
Unix Millisecond Timestamp	x	1360013296123