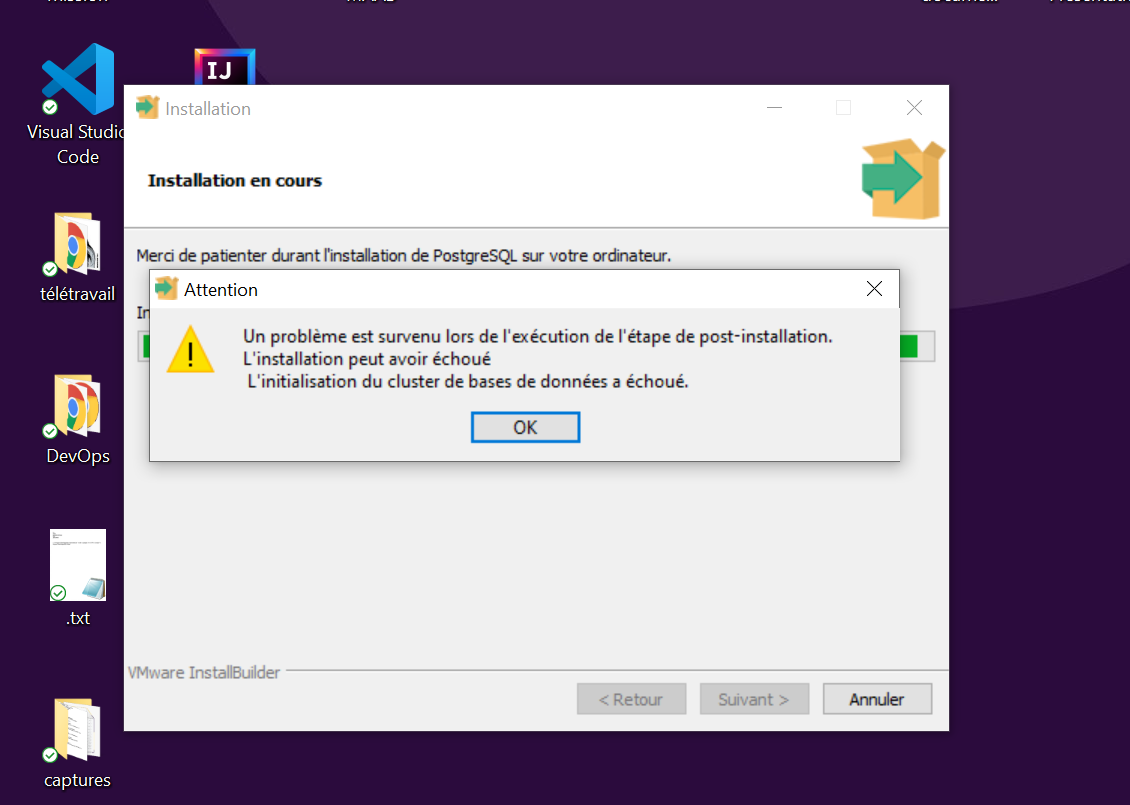
Installation de l’environnement technique – Gateway

# Installation des outils

* Installation de Java 11/HotSpot/Temurin 11 : <https://adoptopenjdk.net/>
* Installation IDE IntelliJ : <https://www.jetbrains.com/fr-fr/idea/>
* BDD : Installation de PostgreSql v13.6 : <https://www.enterprisedb.com/downloads/postgres-postgresql-downloads>

**En cas d'erreur sur la variable d’environnement ComSpec :** vérifier que la variable système “ComSpec” existe et a la valeur “C:\Windows\System32\cmd.exe” (attention à la casse), sinon corriger, redémarrer et réessayer.

**En cas d'erreur dans l'installation de bdd** : « L'initialisation du cluster de base de données a échoué » :



Si lors de l'installation, le message suivant est apparu : "l'initialisation du cluster de base de données a échoué"

Alors la base de données Postgres ne sera pas fonctionnelle.

**Vérifier l'origine de l'erreur**

C:\Users\<USERNAME>\AppData\Local\Temp\install-postgresql.log

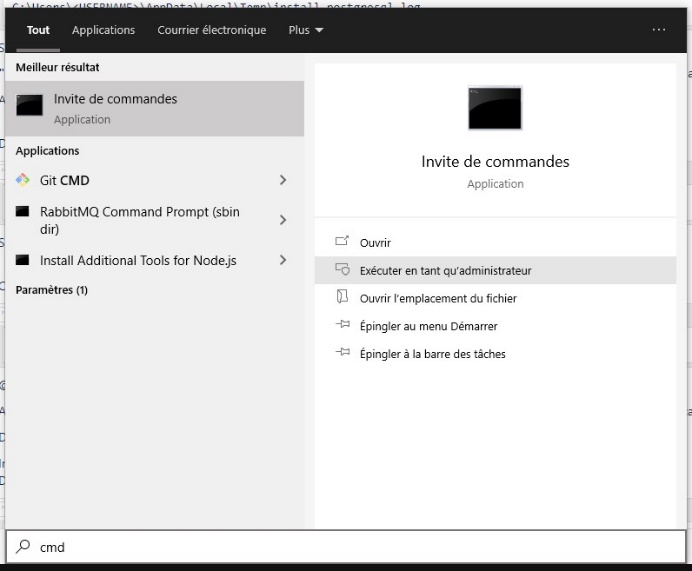
Si dans les dernières lignes du fichier il y a le message suivant

"lancement du script bootstrap...L'exécution du serveur PostgreSQL par un utilisateur dot‚ de droits d'administrateur n'est pas permise."

Alors il s'agit d'un problème d'utilisateur et de droit.

# Vérifier l'existence de l'utilisateur

Lancer en invite de commande comme administrateur



Dans l'invite de commande en tant que **administrateur** :

@admin> net user

Si l'utilisateur **postgres** n'apparait pas, alors il faut le créer.

# Créer l'utilisateur "postgres"

# Dans l'invite de commande en tant que administrateur:

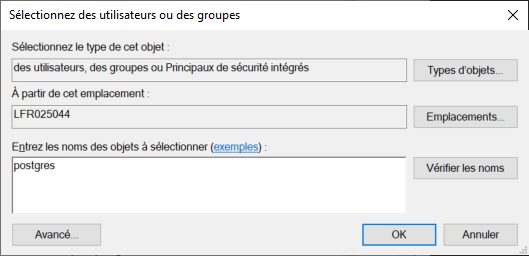
@admin> net user postgres <PASSWORD> /add

Attention, <PASSWORD> ne doit pas contenir le nom d'utilisateur et doit compter un certain nombre de majuscules, chiffres et caractères spéciaux.

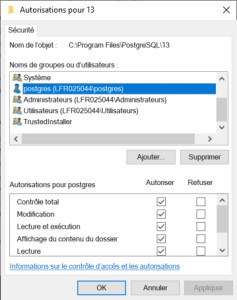
# Donner les droits à l'utilisateur "postgres"

Dossier "C:\Program Files\PostgreSQL" > Clique droit > Propriétés > Sécurité > Modifier

Vérifier si l’utilisateur « postgres » est présent dans la liste « Noms des groupes ou utilisateurs ». Sinon, cliquez sur ajouter :



Donner les droits "**Contrôle tota**l" sur le dossier d'installation de postgres à l'utilisateur "postgres"



**Import : il est également nécessaire de reproduire la même opération sur le dossier ‘data’**

# Initialiser la base de données

Vérifier que le dossier : "C: \Program Files\PostgreSQL\13\data" est bien vide et le supprimer.

Dans l'invite de commande en tant que **administrateur :**

@admin> runas /user:postgres cmd

Dans l'invite de commande créée en tant que **postgres :**

@postgres> "C:\Program Files\PostgreSQL\13\bin\initdb.exe" -A md5 -U postgres -W -E UTF8 --no-locale "C:\Program Files\PostgreSQL\13\data"

\Program Files\PostgreSQL\13\data"

# Lancer le service

# Dans l'invite de commande en tant qu’administrateur :

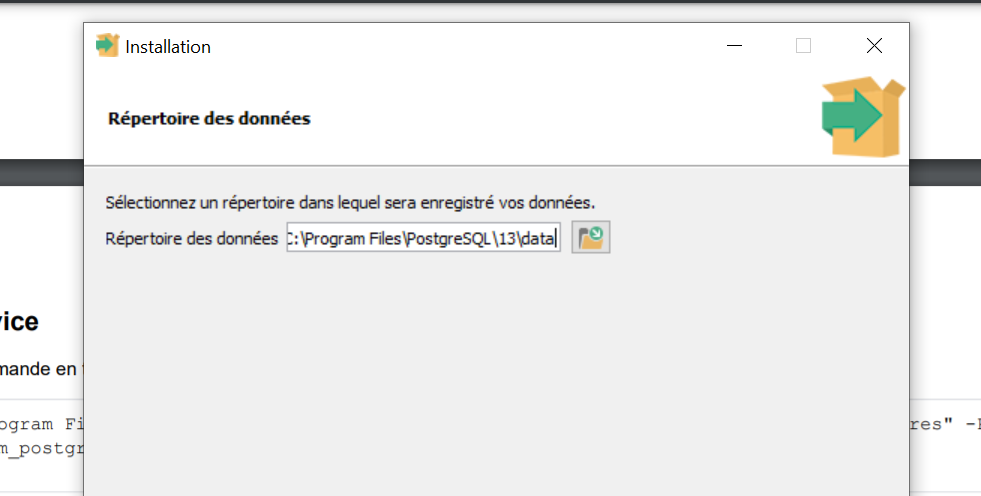
@admin> "C:\Program Files\PostgreSQL\13\bin\pg\_ctl.exe" register -N "PostgreSQL 13" -U ".\postgres" -P mdp\_user\_system\_postgres -D "C:\Program Files\PostgreSQL\13\data" -w

# Installation du serveur

Relancer le fichier postgresql-3\*-windows-x64.exe d’installation de postgresql.

Le fichier d’installation demandera de sélectionner un répertoire de données, pointez alors vers le fichier

"C : \Program Files\PostgreSQL\13\data" comme ci-dessous :



Si une fenêtre Stack Builder s’ouvrira à la suite de l’installation, la fermer.

# Lancer le serveur

Dans l'invite de commande en tant que **administrateur :**

@admin> runas /user:postgres cmd

*En cas d’erreur :* “1327 : Des restrictions de compte d’utilisateur empêchent cet utilisateur de se connecter. Des raisons possibles sont des mots de passe vides n’étant pas autorisés, des restrictions sur les heures de connexion ou une restriction de stratégie a été appliquée”

-> cela signifie que le mot de passe de l’utilisateur postgres a expiré, pour le modifier dans l’invite de commande :

net user postgres <new password>

Dans le nouveau terminal lancé, en tant que user **postgres**:

"C:\Program Files\PostgreSQL\13\bin\pg\_ctl.exe" start -D "C:\Program Files\PostgreSQL\13\data"

Le serveur est lancé

* Ouvrir pgAdmin et vérifier si le server postgres a bien été installé.

**Installation de Maven2**

<https://maven.apache.org/download.cgi>

Télécharger le fichier bin.zip pour windows

Décompresser le dossier et copier le dossier dans C:\Program Files

Rechercher dans la barre de recherche « variable » et cliquer sur Modifier les variables d’environnements

Une fenetre s’ouvre cliquer sur Variables d’environnements

Créer ou modifier **une variable système** nommé MAVEN\_HOME avec comme valeur :

C:\Program Files\apache-maven-3.8.5

**Installation WSL2 / Docker**

Configuration de l’utilisateur

* Install sudo

*apt install sudo*

* S’assurer que votre utilisateur fait partie des utilisateur root

*grep -E 'sudo|wheel' /etc/group*

Vous devriez voir le résultat suivant :

*sudo: x:27:myusername*

Si le résultat n’est pas celui attendu, merci de vous référer à la doc (Partie - **Is your user a "sudoer" ?**): <https://dev.to/_nicolas_louis_/how-to-run-docker-on-windows-without-docker-desktop-hik>

Mise à jour des packages

echo "nameserver 8.8.8.8" | sudo tee /etc/resolv.conf > /dev/null

* Mise à jour des packages

*sudo apt update && sudo apt upgrade*

* Enlever des anciennes installations docker

*y*

* Installation des dépendances

*sudo apt install --no-install-recommends apt-transport-https ca-certificates curl gnupg2*

Configuration du package repository

* Set variables OS

*source /etc/os-release*

* Être sur que apt fait confiance au repository

*curl -fsSL https://download.docker.com/linux/${ID}/gpg | sudo apt-key add -*

* Update les informations du repository

*echo "deb [arch=amd64] https://download.docker.com/linux/${ID} ${VERSION\_CODENAME} stable" | sudo tee /etc/apt/sources.list.d/docker.list*

*sudo apt update*

Installation de docker et ajout de l’utilisateur au groupe docker

* Installation docker

*sudo apt install docker-ce docker-ce-cli containerd.io*

* Ajout de l’utilisateur au groupe docker

*sudo usermod -aG docker $USER*

Fermer le terminal wsl et relancez le

Configuration du démon docker

* Création d’un répertoire partagé pour les sockets docker et ajout des permissions

*DOCKER\_DIR=/mnt/wsl/shared-docker*

*mkdir -pm o=,ug=rwx "$DOCKER\_DIR"*

*sudo chgrp docker "$DOCKER\_DIR"*

* Créer un fichier daemon.json dans le dossier /etc/docker avec les informations suivantes :

*{*

*"hosts": ["unix:///mnt/wsl/shared-docker/docker.sock"],*

*"iptables": false*

*}*

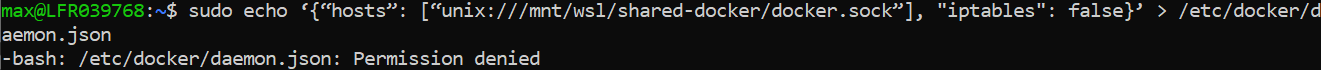
* Si le dossier /etc/docker n'est pas présent:

*sudo mkdir /etc/docker*

*sudo echo ‘{“hosts”: [“unix:///mnt/wsl/shared-docker/docker.sock”], "iptables": false}’ > /etc/docker/daemon.json*

Cas d’erreur :

Si malgré le sudo la permission est refusée, éditer le fichier à la main.



Sudo vi /etc/docker/daemon.json

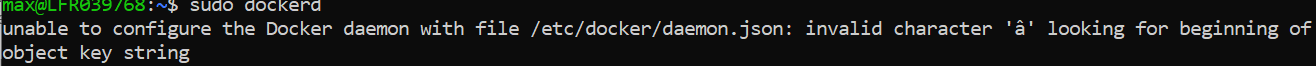
Appuyer sur Echap puis i pour entrer en mode edition et copier-coller le texte suivant :

{

"hosts": ["unix:///mnt/wsl/shared-docker/docker.sock"],

"iptables": false

}



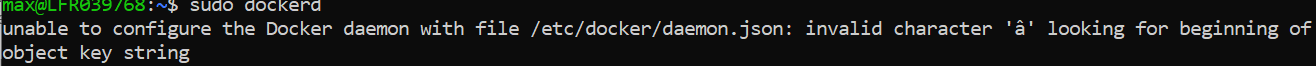
Pour vérifier le fonctionnement du démon docker, vous pouvez lancer la commande

*sudo dockerd*

Elle devrait se finir par **« API listen on /mnt/wsl/shared-docker/docker.sock »**

Si c’est le cas, vous avez réussi. Bravo. Youpi.

Cas d’erreur :

* 1.  Vérifier la bonne indentation du fichier json et son contenu.

* 1. failed to start daemon: pid file found, ensure docker is not running or delete /var/run/docker.pid

Solution : Lister les services en cours d’exécution de docker : ps -ef | grep docker

Tuer tous les services listés : kill –9 [PID]

Si le problème persiste vous pouvez essayer de supprimer les anciennes versions de docker :

sudo apt remove docker docker-engine docker.io containerd runc

Modification du .bashrc

Pour permettre au démon docker de se lancer sans avoir à ouvrir un terminal et à lancer une commande, vous pouvez modifier le fichier **.bashrc** en ajoutant les informations suivantes. **Attention à la bonne indentation**.

*DOCKER\_DISTRO="Ubuntu-20.04"*

*DOCKER\_DIR=/mnt/wsl/shared-docker*

*DOCKER\_SOCK="$DOCKER\_DIR/docker.sock"*

*export DOCKER\_HOST="unix://$DOCKER\_SOCK"*

*if [ ! -S "$DOCKER\_SOCK" ]; then*

*mkdir -pm o=,ug=rwx "$DOCKER\_DIR"*

*chgrp docker "$DOCKER\_DIR"*

*/mnt/c/Windows/System32/wsl.exe -d $DOCKER\_DISTRO sh -c "nohup sudo -b dockerd < /dev/null > $DOCKER\_DIR/dockerd.log 2>&1"*

*fi*

Si vous avez des pbs avec la DOCKER\_DISTRO, lancez la commande ***wsl -l -q*** dans un powershell pour si la liste des WSL distro sur votre pc.

Afin de ne pas avoir à renseigner de mdp, lancer la commande

*sudo vi sudo*

Et ajouter la ligne suivante (pour sauvegarder : nano = « crt + s », vim = « echape puis :wq! puis entrer)

*%docker ALL=(ALL)  NOPASSWD: /usr/bin/dockerd*

Fermer le terminal wsl et relancer le.

Pour plus d’informations, référez-vous à la doc suivante : <https://dev.to/bowmanjd/install-docker-on-windows-wsl-without-docker-desktop-34m9>

**Pensez à désinstaller Docker Desktop et à redémarrer votre ordinateur.**

Vous pourrez ensuite lancer vos commandes docker dans le terminal wsl.