

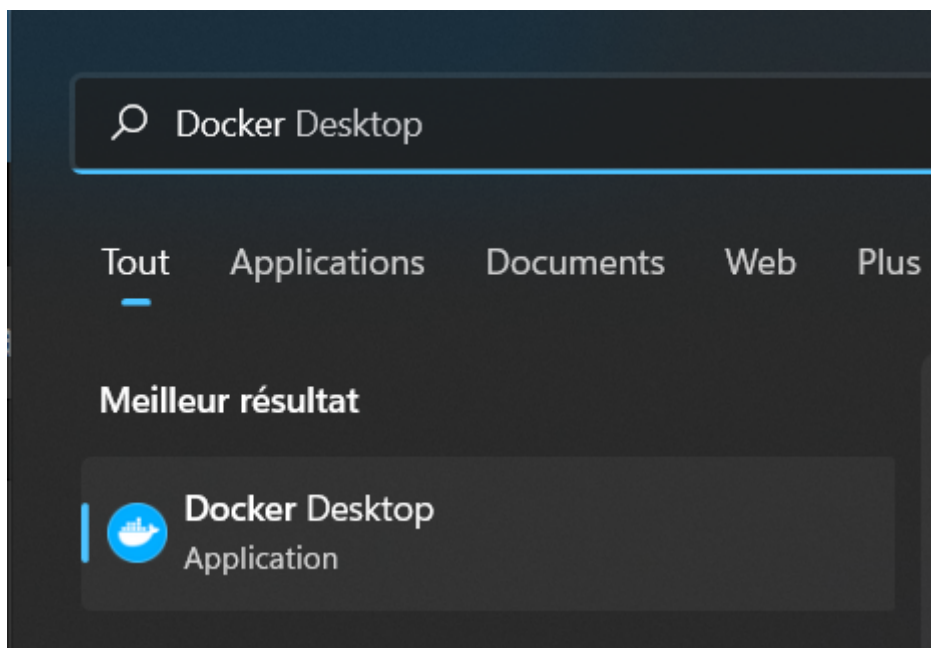
Installer Oracle via docker et s'y connecter avec Datagrip

Prérequis

- [Installer DataGrip](#)
- [Installer Docker](#)

Lancer une image Docker pour Oracle

Lancer le Docker engine en exécutant l'application Docker Desktop :



Exécuter la commande Docker suivante :

```
docker run -d -p 1521:1521 --name mon-oracle quay.io/maksymbilenko/oracle-12c
```

“ Cette image est particulièrement grosse (environ 4 Go) donc ça peut prendre quelques minutes selon votre connexion internet

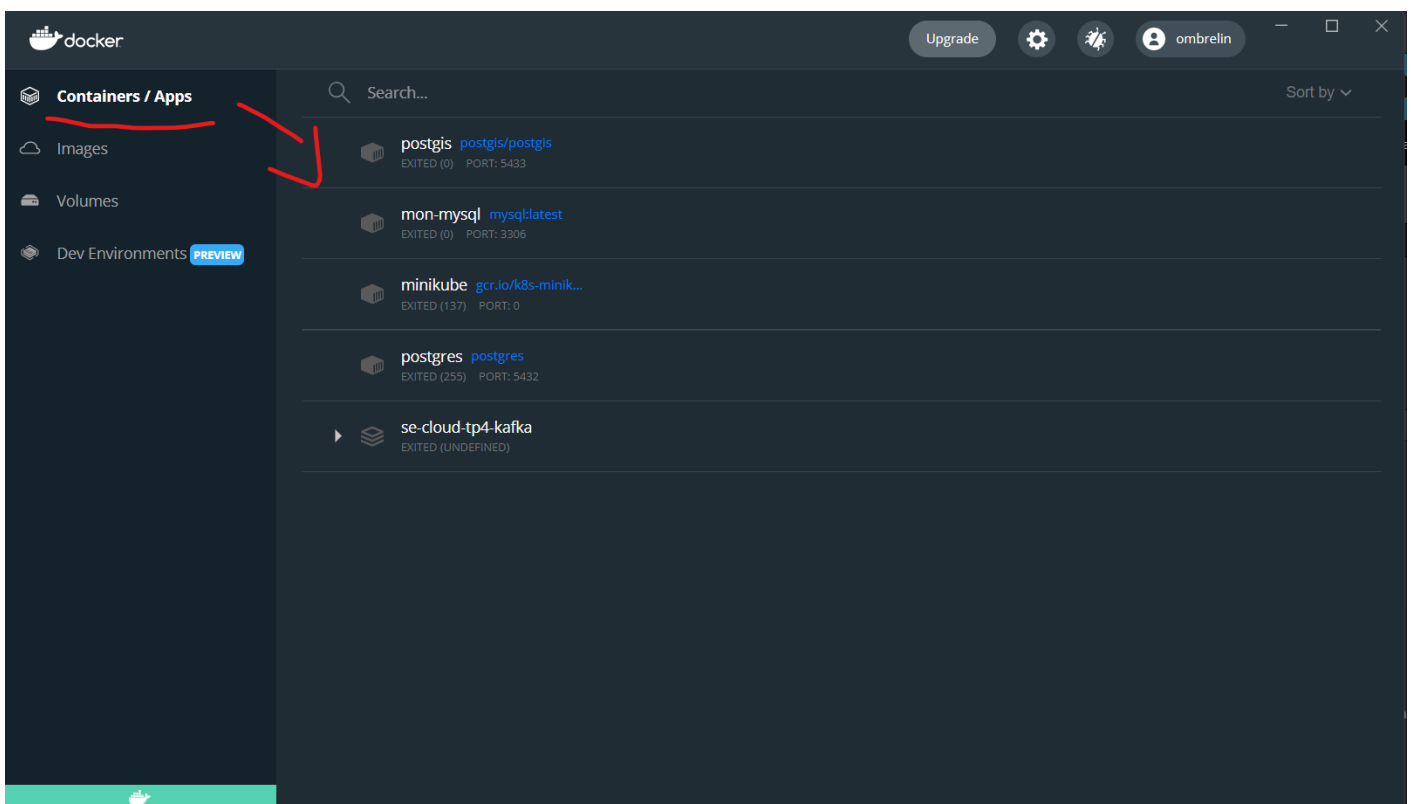
Explication des options :

- `-d` : lancer le conteneur en arrière-plan
- `-p 1521:1521` : forwarder le port 1521 du conteneur vers le port 1521 de la machine hôte. C'est le port utilisé pour se connecter à la base
- `--name mon-oracle` : on nomme notre conteneur pour permettre de le manipuler avec un nom simple
- `quay.io/maksymbilenko/oracle-12c` : emplacement de l'image que l'on veut appliquer à notre conteneur

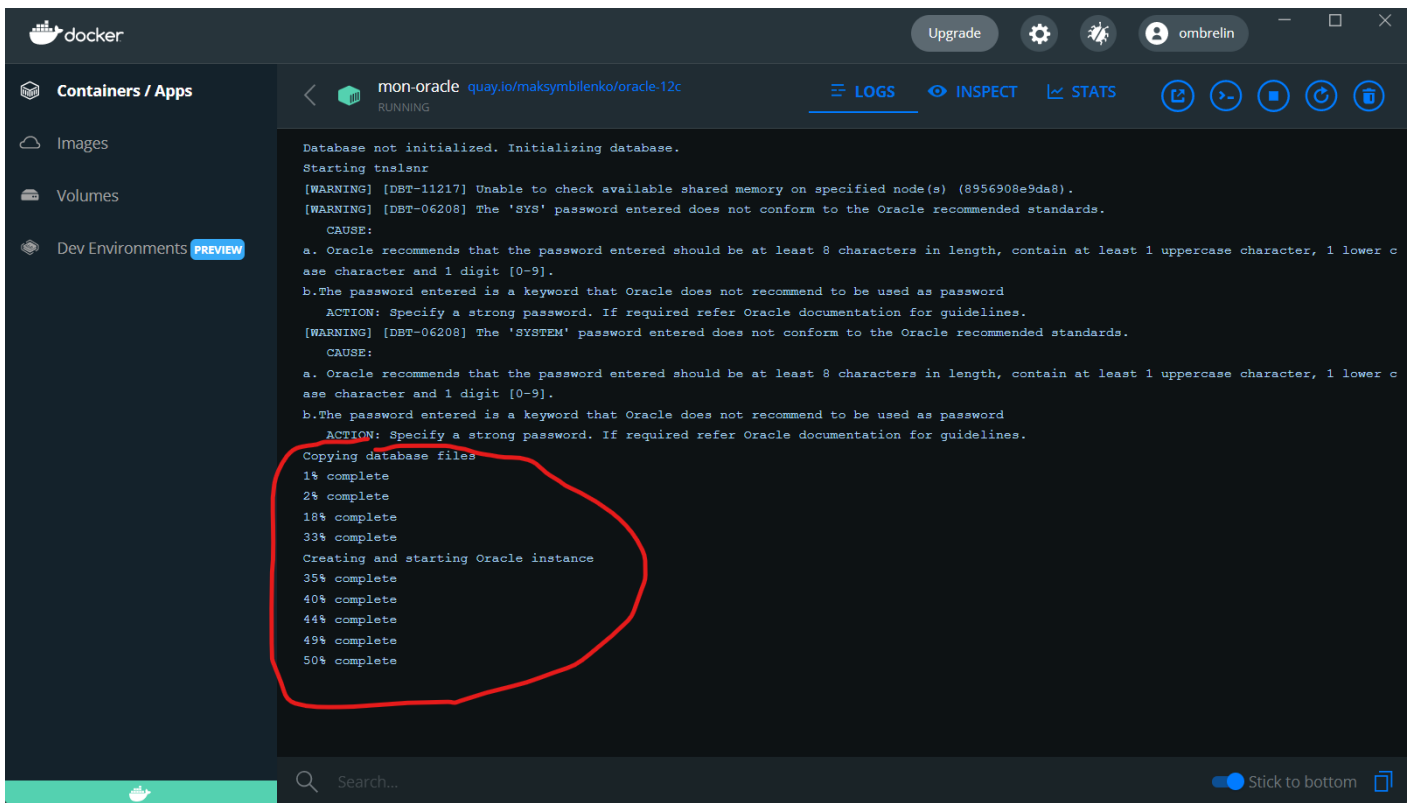
Pour voir l'état de votre conteneur, vous pouvez accéder à la GUI Docker Desktop, pour l'ouvrir, cliquez sur la petite icône de baleine en bas à droite de votre écran :



Cela vous ouvre l'interface, ainsi quand vous cliquez sur "Containers / Apps" vous voyez une liste des conteneurs que vous avez créés :



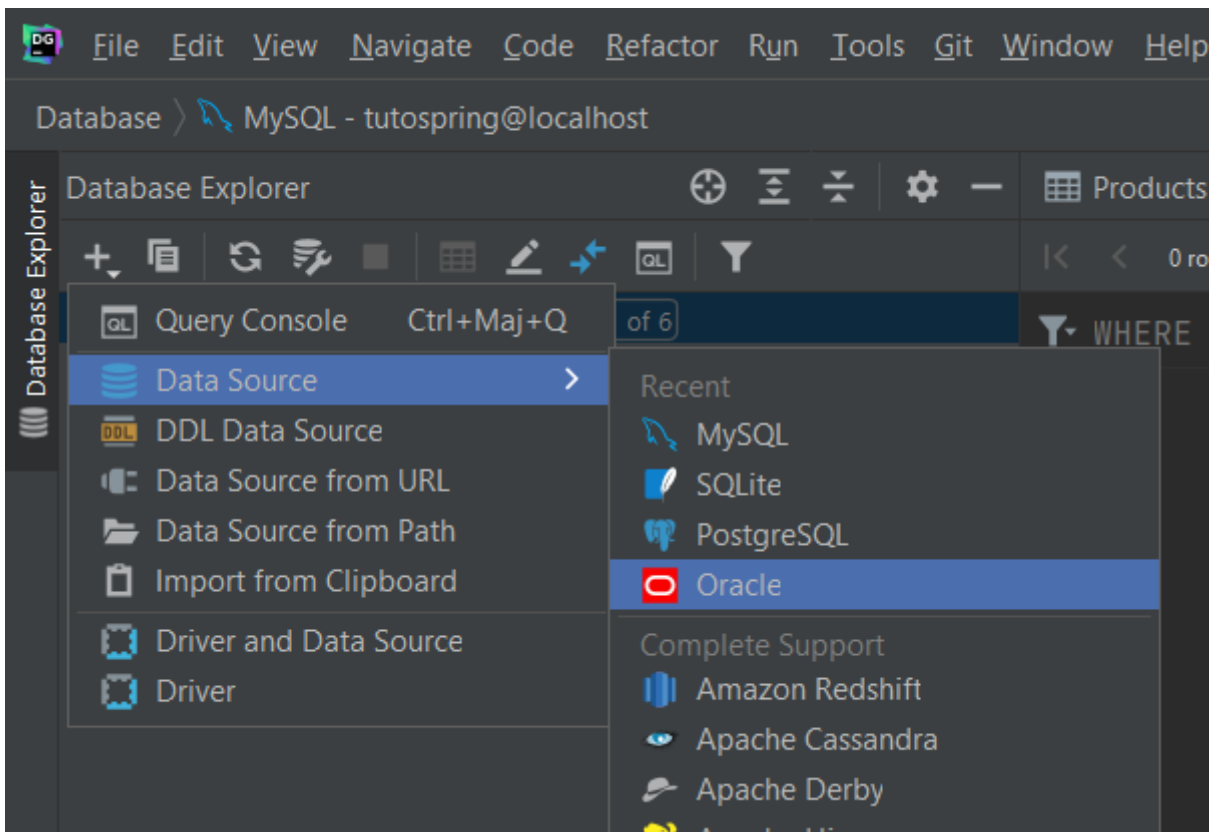
Vous pouvez cliquer sur la ligne de votre image pour voir ses logs. Dans ses logs vous pouvez voir la progression de l'initialisation de la base :



Attends que cette initialisation soit terminée (affichage du message `Database ready to use. Enjoy!`;)) avant de vous connecter avec Datagrip.

Se connecter à Oracle sur DataGrip

Ouvrez Datagrip, puis cliquez sur l'icone  pour ajouter une nouvelle source de donnée de type Oracle :



Vous arrivez ensuite à la page de connexion à la base. Voici les étapes pour se connecter :

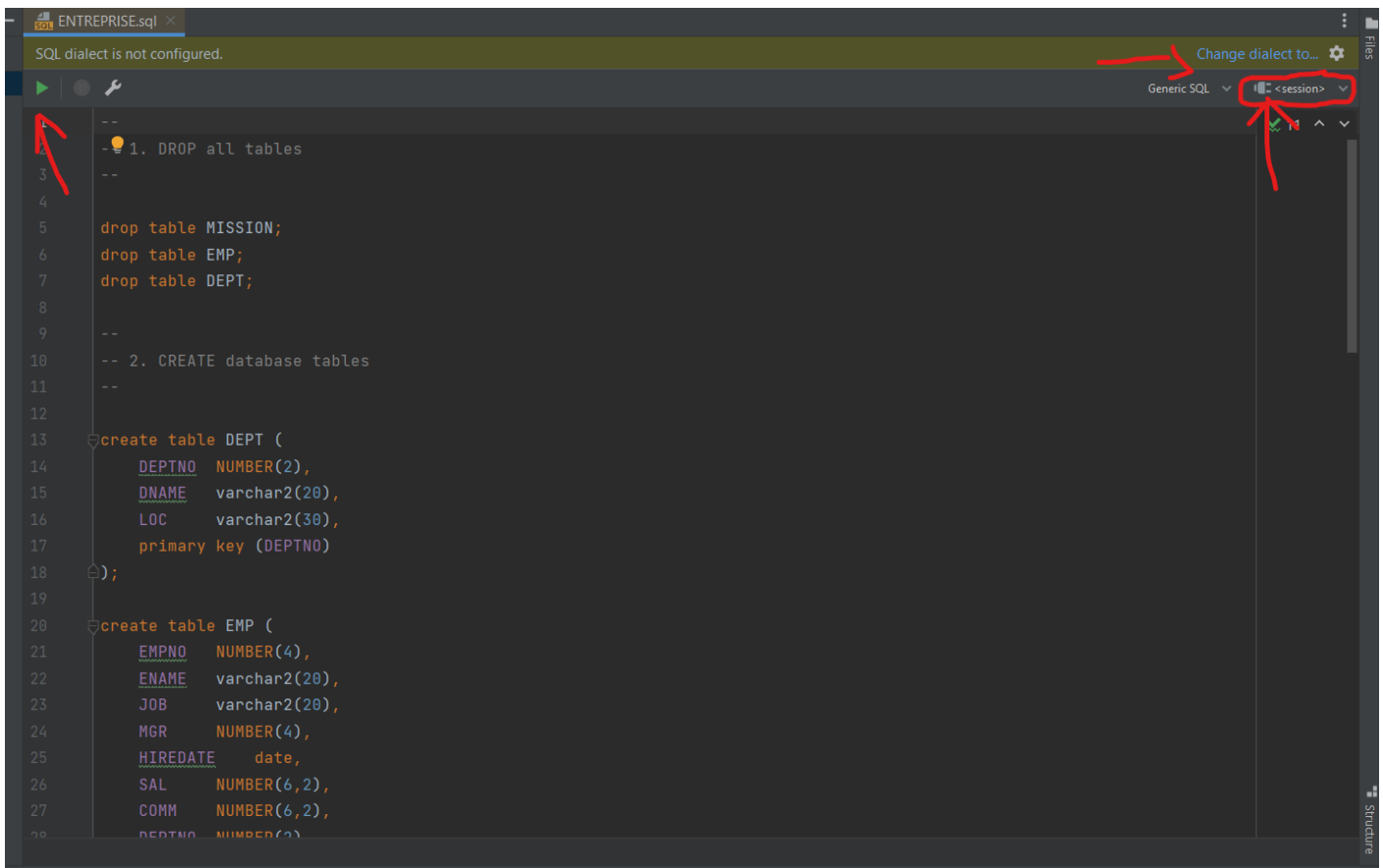
1. En bas à gauche du formulaire, cliquez sur "Download missing driver"
2. Dans "Name" renseignez le nom que vous voulez
3. Dans "Host" indiquez `localhost`
4. Dans "Port" indiquez `1521`
5. Dans "SID" indiquez `xe` (en minuscule, la valeur de base est XE mais en majuscule)
6. Dans "User" indiquez `system`
7. Dans "Password" indiquez `oracle`

Enfin cliquez sur "Test Connection" pour vérifier que tout est bon, si vous avez une popup verte cliquez sur "OK".

Patientez ensuite quelques seconde le temps que la base de donnée soit explorée par l'outil, puis faites clic droit sur votre nouvelle Datasource puis `New > Query Console`. Cela va vous ouvrir une invite de commande SQL avec une bonne autocompletion et coloration syntaxique.

Ouvrir et exécuter des fichiers de script SQL

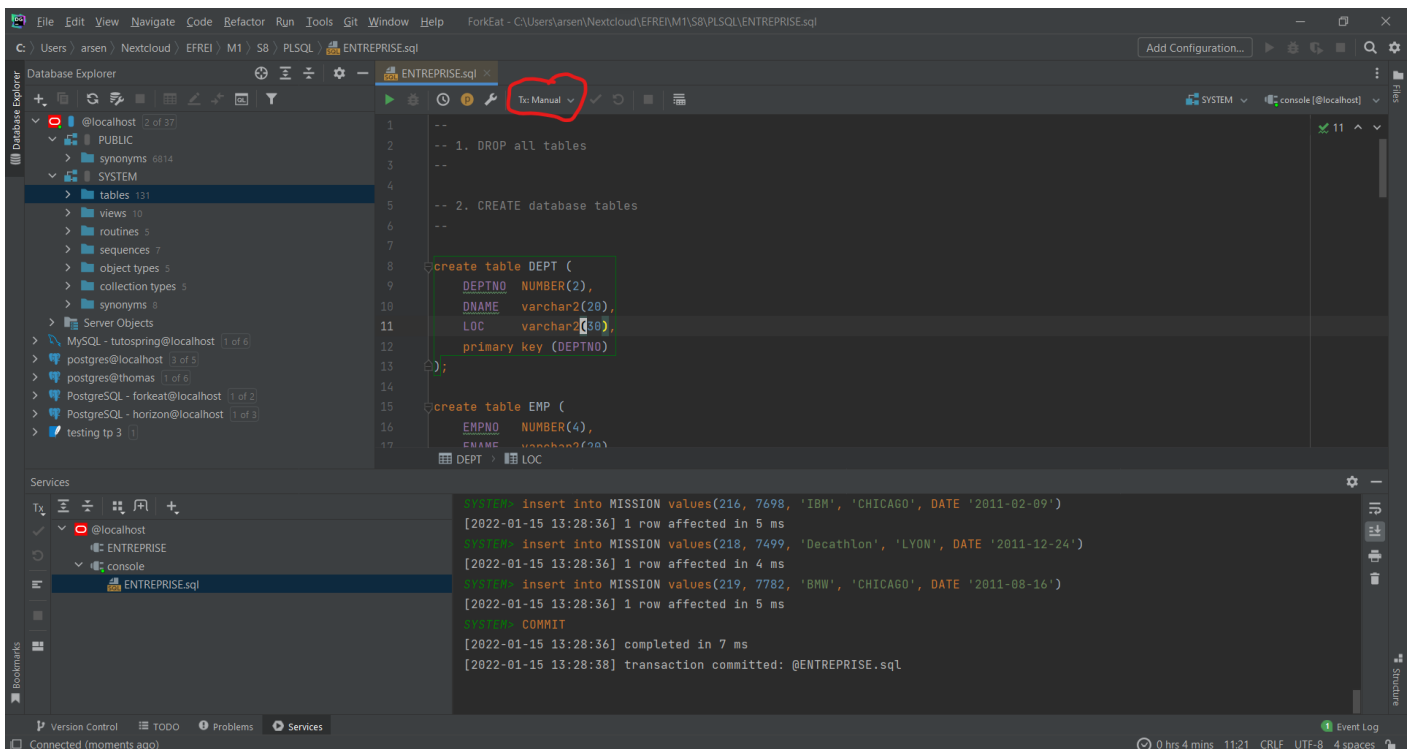
Pour ouvrir un script SQL avec Datagrip, vous pouvez simplement le Drag & Drop dans l'éditeur :



Infos à savoir :

- Avec le bouton "Change dialect to" vous pouvez régler l'éditeur pour lui dire de valider la syntaxe d'Oracle
- Avec la liste déroulante "Session", vous pouvez rattacher l'éditeur à une session de connexion (existante ou en créer une nouvelle) cela permettra à l'éditeur de fournir de la validation et de l'autocomplétion en étant conscient du contenu de votre base).
- Le bouton "Run" (à gauche) permet d'exécuter le script SQL sur la session courante, quand vous exécutez, il vous demande si vous voulez exécuter seulement le statement sur lequel est votre curseur, ou tout le fichier

Quand vous exécutez une comande sur une console ou un script, une console de log s'ouvre pour afficher le résultat de l'exécution :



J'ai aussi entouré la liste de déroulante "Tx: Manual", cela permet de changer de mode de transaction :

- "Manual" : n'autocommit pas, il faut appuyer sur la coche verte juste à droite pour commit la transaction
- "Auto" : autocommit les commandes

Par défaut mettez sur "Auto".

Revision #6

Created 2022-01-15 11:15:54 UTC by Arsène Lapostolet

Updated 2022-01-17 15:02:47 UTC by Arsène Lapostolet