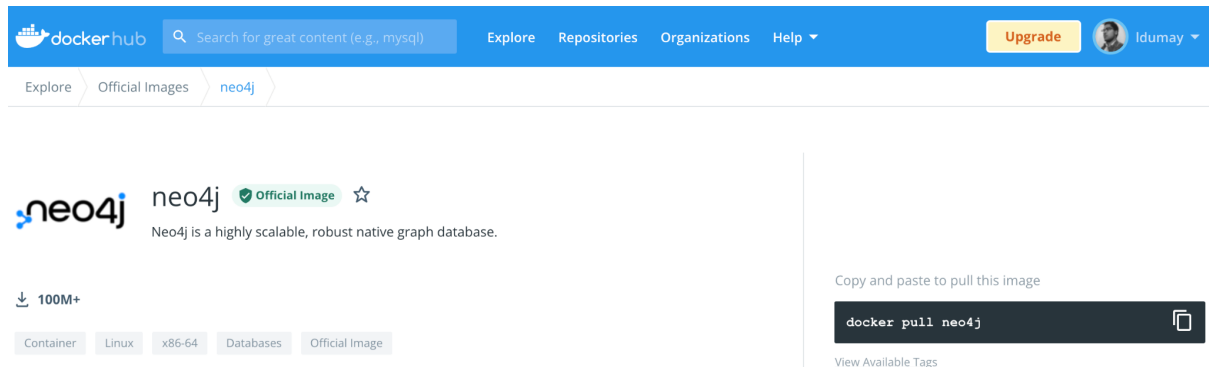


Neo4j - Installation via Docker

Nous avons choisi d'effectuer une installation légère via [Docker](#).

Pour ce faire, nous nous adaptons selon le petit guide d'installation disponible sur le [Hub de Docker](#).



1. Récupération de l'image officiel de Neo4j

Afin de récupérer l'image officiel de Neo4j, il suffit de faire la commande ci-dessous.

[Commande=>] `docker pull neo4j`

```
mtl@MacBook-Pro-LDumay ~ % docker pull neo4j
Using default tag: latest
latest: Pulling from library/neo4j
7d63c13d9b9b: Already exists
225be9814eda: Already exists
c78f8a9ed884: Pull complete
f7c9c563b3d6: Pull complete
1041dfc330bb: Pull complete
45569075e79f: Pull complete
cf4e8b14b6bc: Pull complete
d6d142cd8f86: Pull complete
Digest: sha256:58a9c0b15dc58fbc9881c88d69a0f4344b34e3c0a26b794547d9dfbc7b145928
Status: Downloaded newer image for neo4j:latest
```

2. Création d'un conteneur Neo4j

Pour créer un nouveau conteneur à partir de l'image Neo4j, voici la commande ci-dessous qui permet à la fois de créer un nouveau conteneur de Neo4j et de le démarrer avec les ports nécessaires au fonctionnement général de Neo4j.

[Commande=>] `docker run --publish=7474:7474 --publish=7687:7687 neo4j`

```
mtl@mbp-ldumay ~ % docker run --publish=7474:7474 --publish=7687:7687 neo4j
2021-11-04 11:34:42.823+0000 INFO Starting...
2021-11-04 11:34:46.382+0000 INFO ===== Neo4j 4.3.6 =====
2021-11-04 11:34:49.267+0000 INFO Initializing system graph model for component 'security-users' with version -1 and status UNINITIALIZED
2021-11-04 11:34:49.280+0000 INFO Setting up initial user from defaults: neo4j
2021-11-04 11:34:49.282+0000 INFO Creating new user 'neo4j' (passwordChangeRequired=true, suspended=false)
2021-11-04 11:34:49.296+0000 INFO Setting version for 'security-users' to 3
2021-11-04 11:34:49.300+0000 INFO After initialization of system graph model component 'security-users' have version 3 and status CURRENT
2021-11-04 11:34:49.307+0000 INFO Performing postInitialization step for component 'security-users' with version 3 and status CURRENT
2021-11-04 11:34:49.707+0000 INFO Bolt enabled on 0.0.0.0:7687.
2021-11-04 11:34:51.389+0000 INFO Remote interface available at http://localhost:7474/
2021-11-04 11:34:51.393+0000 INFO Started.
```

3. Vérification et renommage du conteneur Noe4j

Enfin, pour plus de simplicité d'utilisation, voici la méthode pour renommer le conteneur Neo4j.

[Commande 1=>] `docker ps`

[Commande 2=>] `docker rename <ecstatic_moore>* noe4j`

[Commande3 =>] `docker ps`

**<ecstatic_moore> : à remplacer par le nom aléatoire du conteneur créer.*

```
mtl@mbp-ldumay ~ % docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED     STATUS      PORTS                               NAMES
7dbd36fa3e5a   neo4j     "/sbin/tini -g -- /d..." 59 seconds ago Up 58 seconds 0.0.0.0:7474->7474/tcp, 0.0.0.0:7687->7687/tcp ecstatic_moore

mtl@mbp-ldumay ~ % docker rename ecstatic_moore noe4j
mtl@mbp-ldumay ~ % docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED     STATUS      PORTS                               NAMES
7dbd36fa3e5a   neo4j     "/sbin/tini -g -- /d..." About a minute ago Up About a minute 0.0.0.0:7474->7474/tcp, 0.0.0.0:7687->7687/tcp noe4j
```

4. Accès à page web du conteneur

Pour accéder à la page web du conteneur, il vous suffit d'aller sur le lien ci-dessous.

[Accès à Neo4j =>] <http://localhost:7474/>

