

Professeure : Nastaran Fatemi Assistant : Christopher Meier

# Installation PostgreSQL + DataGrip

# 1 Introduction

Les labos de BDR vont utiliser le SGDB PostgreSQL. Pour simplifier son installation on va utiliser Docker.

# 2 Installation

#### 2.1 Docker

Docker est une plateforme de containérisation qui permet le lancement simple d'application et de services.

Les utilisateurs Windows et Mac installent *Docker Desktop*.

Les utilisateurs Linux installent *Docker engine* et *docker-compose* avec le gestionnaire de paquet de leur distribution.

### 2.2 Postgresql

Télécharger le fichier docker-compose.yml depuis Cyberlearn et placer le dans un dossier.

Depuis un terminal, naviguer vers le dossier, puis exécuter la commande suivante :

### \$ docker-compose up

### 2.3 DataGrip

DataGrip est un IDE pour les base de données créé par JetBrains. En tant qu'étudiant, vous avez accès gratuitement à une license pour tous les produits JetBrains.

# 3 Configuration

# 3.1 DataGrip

Aller sur File → New → Data Source → PostgreSQL pour créer une connexion vers Postgres (c.f. figure 1).

Utiliser la configuration suivante (c.f. figure 2):

Host localhost

Port 5432

User bdr

Password bdr

Database bdr

Il est aussi possible d'utiliser le rôle administrateur afin d'avoir un accès total au serveur PostgreSQL:

User postgres

Password root



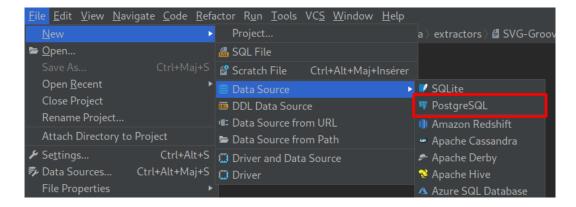


Figure 1: Capture - Creation d'une connexion

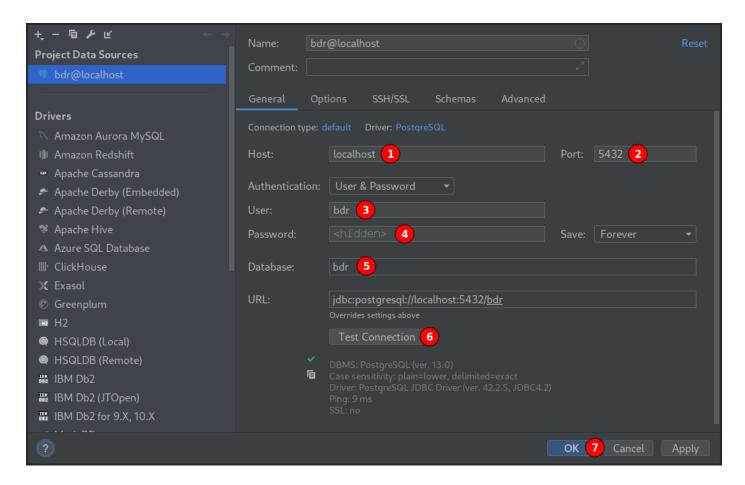


Figure 2: Capture - Configuration d'une connexion



Professeure : Nastaran Fatemi Assistant : Christopher Meier

# 4 Utilisation

### 4.1 Postgres

Les commandes suivantes doivent être executées depuis un terminal dans le dossier avec le fichier docker-compose.yml

Pour démarrer postgres:

\$ docker-compose start

Pour arrêter postgres:

\$ docker-compose stop

Pour supprimer postgres (attention vous allez perdre toutes les données):

\$ docker-compose drop

### 4.2 psql

L'application *psql* est un client en ligne de commande pour Postgresql.

\$ docker-compose exec -e PGPASSWORD=bdr postgresql psql --username=bdr

Vous pouvez executer un script SQL dans la base de données 'bdr' avec la commande suivante:

\$ docker-compose exec -e PGPASSWORD=bdr -T postgresql psql -U bdr -f - bdr < script.sql</pre>

### 4.3 DataGrip

Pour executer du code SQL:

- 1. Ouvrir une console
- 2. Selectionner le schéma
- 3. Entrer les commandes SQL
- 4. Executer la commande actuelle avec <Ctrl> + <Enter>



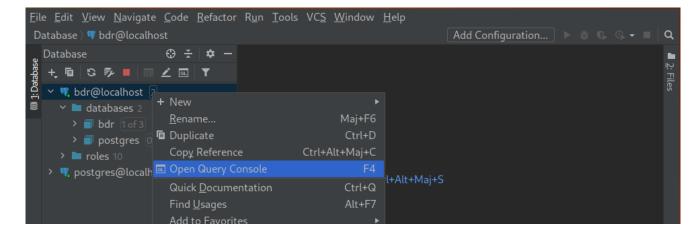


Figure 3: Capture - Ouvrir une console

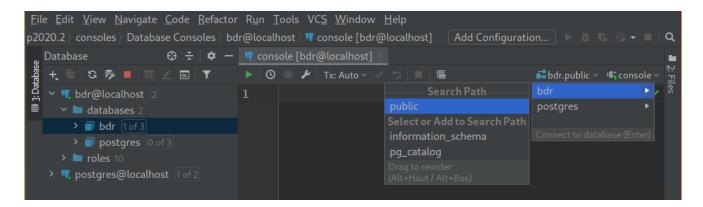


Figure 4: Capture - Selectionner le schéma



```
+ 6 5 秒 ■ Ⅲ ∠ 때 ▼
                              ▶ 🕓 🕩 🗡 | Tx: Auto 🗸 🗸 🗏 🚟
                                                                         🚅 bdr.tutoriel 🗸 📭 console 🤊
🗸 嘴 bdr@localhost 📵
                                    create schema tutoriel;
    > | bdr 2 of 4
                                     set search_path to tutoriel;
    > postgres 0 of 3
                                     create table test(
> ¶ postgres@localhost 1 of 2
                                         attr1 integer,
                                         attr2 varchar(20),
                                         primary key (attr1)
                                     insert into test values (2, 'hello world!');
                                     select * from test;
Tx 호 중 | 태 用 | 七
                       ■ Output ■ bdr.tutoriel.test
   bdr> create schema tutoriel
                      [2020-10-08 15:08:26] completed in 3 ms
       ■ schema
                       bdr.public> set search_path to tutoriel
                                                                                           ÷
                                                                                           ŧ
                      [2020-10-08 15:09:12] completed in 1 ms
         ronsole 26 ms
                       odr.tutoriel> create table test(
   > 📭 postgres@localhost
                                          attr1 integer,
                                          attr2 varchar(20),
                                          primary key (attr1)
                      [2020-10-08 15:10:06] completed in 8 ms
                        dr.tutoriel> insert into test values (2, 'hello world!')
                       [2020-10-08 15:10:41] 1 row affected in 12 ms
                                      select * from test
                      [2020-10-08 15:11:03] 1 row retrieved starting from 1 in 19 m
≡ TODO ▶ <u>4</u>: Run 9 <u>6</u>: Problems

♠ 8: Services

                                                                                    94 Event Log
Connected (2 minutes ago)
```

Figure 5: Capture - Entrer les commandes SQL



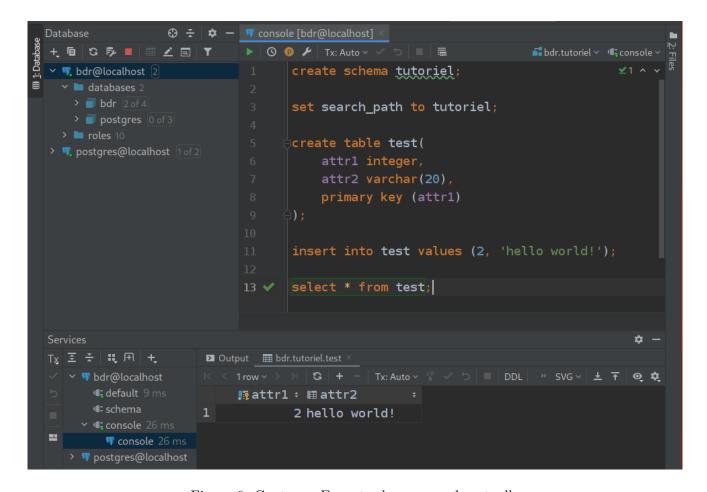


Figure 6: Capture - Executer la commande actuelle