

1 De la bonne utilisation des différents outils

Différents outils permettent de se connecter à la base de données PostgreSQL du cours et d'y effectuer des requêtes SQL. Nous en présentons deux ici : `psql` et `pgAdminIII`. Le premier est un outil utilisable avec la console (ou terminal) et ne nécessite pas d'être installé sur votre machine (on peut utiliser celui du serveur). Cela peut paraître austère à première vue mais il possède certaines fonctionnalités très pratiques. Le second propose une interface graphique plus sympathique mais doit être installé et bien configuré sur votre machine pour pouvoir accéder au serveur.

Choisissez celui que vous voulez, le tout étant d'être capable d'effectuer les requêtes SQL demandées dans les différents TDs. Sachez tout de même que la prise en main de `psql` est plus rapide (il suffit juste de se connecter au serveur).

1.1 psql

Présentation `psql` est un outil en console et est déjà installé sur la machine contenant la base de données. Il vous suffit donc de vous connecter en `ssh`¹ sur le serveur `etu-pgsql.math.univ-paris-diderot.fr` en entrant la commande suivante dans un terminal (sous Ubuntu, faire `CTRL+ALT+T` pour ouvrir un terminal) :

```
ssh entid@etu-pgsql.math.univ-paris-diderot.fr
```

où `entid` est votre identifiant ENT et d'entrer votre mot de passe ENT pour vous connecter à la machine. Sous Mac et Linux, aucun problème normalement pour utiliser `ssh`, ces outils sont installés par défaut. Sous Windows, cela se complique. On pourra utiliser un logiciel comme PuTTY qui permet d'utiliser `ssh`.

Une fois connecté, vous pouvez utiliser le programme `psql` installé sur le serveur et vous connecter à la base de données du cours nommée BD4-2015 en tapant la commande :

```
psql BD4-2015
```

Il faudra entrer le mot de passe distribué lors de la première séance.

Premier contact Si tout se passe bien, voici ce qui devrait s'afficher à l'écran :

```
psql (9.2.9)
Type "help" for help.
BD4-2015=#
```

Dès lors vous pouvez taper vos premières commandes SQL. Pour accéder aux tables du schéma `atp`, par exemple `Player_big`, il faudra taper `atp.Player_big`. Par exemple, vous pouvez entrer la commande SQL suivante :

```
SELECT * FROM atp.Player_big;
```

Remarque 1

- La liste des joueurs s'affiche mais est trop longue. Vous pouvez naviguer avec les flèches haut/bas pour voir les autres entrées.
- Pour quitter cet écran, appuyer sur la touche `q` comme "quitter".
- La liste des joueurs s'affiche mais c'est illisible? Mettez votre console en plein écran (F11 sur Linux) pour gagner en largeur.

1. `ssh` est un programme qui permet d'ouvrir une console sur une machine distante de façon sécurisée. Concrètement, on tape `ssh login@server` dans une console depuis chez soi ou une salle informatique et on se retrouve avec une console comme si on était devant la machine nommée `server` : vous verrez les fichiers de la machine `server` et non plus les vôtres, vous aurez accès aux programmes installés sur la machine `server` etc.

- BD4-2015-# s'affiche mais pas la liste des joueurs ? Vous avez oublié le point virgule (ça vous arrivera un jour ou l'autre) ! Écrivez un point-virgule et validez.

Essayez alors d'autres commandes SQL par exemple :

```
SELECT * FROM atp.Player_big WHERE last_name='Federer';
```

Cette fois-ci, la réponse est suffisamment courte pour rentrer entièrement dans l'écran donc pas besoin de naviguer avec les flèches ou de quitter avec q.

Vous avez aussi la possibilité d'utiliser des commandes propres à `psql`. Ces commandes ne servent pas à effectuer des requêtes sur la base de données mais permettent d'obtenir des informations sur `psql`, notamment la structure des tables. La commande

```
\d atp.Player_big
```

affichera la description de la table `atp.Player_big`. La commande `help` vous permettra aussi d'obtenir de l'aide sur les autres commandes. Essayez-la. En tapant `\?` par exemple, vous apprendrez qu'en tapant `\d atp.` vous aurez un récapitulatif des différentes tables de `atp`.

Trucs et astuces Nous donnons ici une liste de trucs et astuces pour vous faciliter l'utilisation de `psql`. Cela peut vous faire gagner énormément de temps :

- Pour ne pas taper `atp` devant chaque nom de table, on peut indiquer à `psql` qu'il faut aller chercher dans `atp` par défaut en tapant

```
SET search_path TO atp;
```

- Vous pouvez retrouver les commandes que vous avez taper précédemment en appuyant sur la flèche du haut.
- Même en plein milieu d'une expression, vous pouvez appuyer sur la touche entrée. Cela permet de revenir à la ligne lorsque votre requête est un peu longue afin de l'aérer un peu. La requête ne sera pas considérée comme terminée tant que vous n'avez pas écrit de point-virgule.
- `psql` permet l'*autocomplétion*. Si vous commencez à écrire une commande SQL ou le nom d'une table, vous pouvez appuyer sur la touche `Tab` pour compléter automatiquement. Par exemple, si vous avez déjà tapé `SELE` et que vous appuyez sur `Tab`, ce sera automatiquement complété en `SELECT`. Si vous avez tapé `Playe` et appuyez deux fois sur `Tab`, vous verrez apparaître `Player_big` et `Played_in_big` à l'écran, qui sont les deux seules manières de compléter ce que vous avez commencé à taper.

1.2 pgAdminIII

`pgAdminIII` est un outil graphique pour gérer les bases de données `Postgresql`. Il permet d'exécuter vos propres requêtes SQL sur une base de données tout comme `psql` mais propose aussi des outils graphiques pour visualiser vos requêtes ou explorer la base de données. Nous présentons ici brièvement la configuration de `pgAdminIII` ainsi que quelques-unes de ses fonctionnalités. Pour les plus curieux, on pourra se référer à l'aide fournie avec le logiciel (Help → Help Contents).

Téléchargement Contrairement à `psql`, vous devez installer `pgAdminIII` sur votre machine (`pgAdminIII` est déjà installé sur les machines de l'université). Les sources ou binaires de `pgAdminIII` peuvent être trouvées à l'adresse <http://www.pgadmin.org/download/>. Vous pouvez aussi l'installer par les dépôts si vous êtes sous Linux.

à vérifier

Configuration Après avoir lancé `pgadminIII`, il va falloir le configurer pour lui indiquer à quelle base de données il doit se connecter. Vous pouvez bien entendu configurer plusieurs bases de données différentes. Nous expliquons ici comment ajouter la base de données utilisée pour le cours. Si vous voulez en ajouter d'autres, le principe est le même.

Pour ajouter un nouveau serveur, cliquez sur **File** → **Add Server**. Une fenêtre s'ouvre. Renseignez les champs suivants de l'onglet **Properties** :

- **Name** : choisissez le nom que vous voulez donner à ce serveur, c'est celui qui s'affichera ensuite dans **pgadminIII**. Par exemple "Serveur Université".
- **Host** : c'est en général l'adresse du serveur. Cependant ici, nous allons accéder au serveur par **ssh** donc il faudra entrer **localhost**.
- **Port** : laissez à 5432.
- **Username** : votre identifiant ENT.
- **Password** : le mot de passe distribué la dernière fois.

Malheureusement cela ne suffit pas pour la connection car le serveur n'est pas directement accessible, il faut passer par un tunnel SSH. Allez dans l'onglet **Tunnel SSH**, cliquez sur la case **Use an SSH tunnel** et renseignez les champs suivants :

- **Host** : **etu-pgsq1.math.univ-paris-diderot.fr**
- **Username** : votre login ENT
- **Authentification** : cochez **Mot de passe**
- **Password/Passphrase** : votre mot de passe ENT.

Validez. Si tout se passe bien, vous verrez "Serveur Université" sur la gauche de la fenêtre. Vous pouvez dérouler le menu et chercher la base de données BD14-2015. Cliquez dessus. Désormais, vous pouvez explorer cette base de données en cliquant sur "Schéma". Cherchez le schéma "atp" que nous utiliserons au cours du semestre. Vous pouvez le dérouler aussi pour avoir des informations sur le schéma, par exemple, on peut voir ses tables.

Pour effectuer des requêtes SQL, cliquez sur le bouton SQL dans la barre en haut, vers le milieu de la fenêtre. Vous pouvez taper votre première requête dans la fenêtre qui s'affiche, par exemple :

```
SELECT * FROM atp.Player_big;
```

Appuyez sur F5 pour l'exécuter. Le résultat s'affiche en bas de l'écran dans un tableau que vous pouvez observer. Pour éviter de taper "atp." devant chaque nom de table, exécutez la requête

```
SET search_path TO atp;
```

Vous pouvez désormais exécuter des requêtes comme :

```
SELECT last_name FROM Player_big WHERE code='FRA';
```