

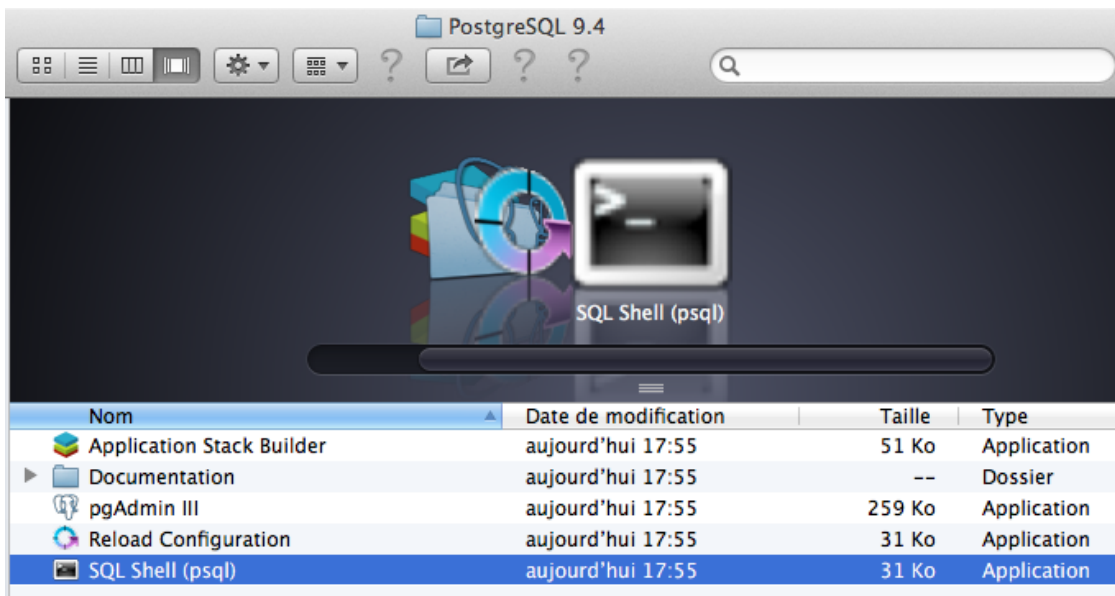
Bases de données<sup>1</sup>

Le langage SQL  
 TD3 BIS  
Sur Postgres

Dans votre compte rendu, chaque requête doit être accompagnée de son résultat (capture d'écran).

## Installation de PostGress

- Choisir la version PostgreSQL 9.4
- Choisir celle qui correspond au système d'exploitation que vous utilisez (Unix, Windows, Mac).
- Lors de son installation, il vous sera demandé un mot de passe. Mettez un mot de passe simple que vous pouvez retenir facilement.
- Après l'installation, vous devriez logiquement avoir un répertoire dont les éléments sont montrés dans la figure suivante :



- Cliquer sur SQL Shell et vous devriez avoir quelque chose qui ressemble à la fenêtre de la figure suivante :

```

daachi — psql — 80x24
Last login: Mon Mar 16 20:57:16 on ttys001
ihaveadreem:~ daachi$ /Library/PostgreSQL/9.4/scripts/runpsql.sh; exit
Server [localhost]:
Database [postgres]: postgres
Port [5432]:
Username [postgres]:
Password for user postgres:
psql (9.4.26)
Type "help" for help.

postgres=#

```

Pour ce TP, seuls les champs Database et password choisis lors de l'installation sont à renseigner comme indiqué dans cette figure. Ne rien mettre pour les autres champs.

1. Année universitaire 2018-2019.

- Pour créer votre propre base de données, vous y connecter, créer une table et voir ses champs avec leur types, vous pouvez faire comme dans la figure suivante.

```
postgres=# create database base_exemple;
CREATE DATABASE
postgres=# \c base_exemple;
You are now connected to database "base_exemple" as user "postgres".
base_exemple=# CREATE TABLE EMP(
base_exemple(# EMPNO INTEGER NOT NULL,
base_exemple(#      ENAME VARCHAR(15),
base_exemple(#      JOB VARCHAR(14),
base_exemple(#      MGR INTEGER,
base_exemple(#      HIREDATE DATE,
base_exemple(#      SAL REAL,
base_exemple(#      COMM REAL,
base_exemple(#      DEPTNO INTEGER);
CREATE TABLE
base_exemple=# \d emp
          Table "public.emp"
  Column |          Type          | Modifiers
-----+-----+-----
 empno   | integer                | not null
  ename  | character varying(15)  |
  job    | character varying(14)  |
  mgr     | integer                |
 hiredate | date                   |
  sal    | real                   |
  comm   | real                   |
  deptno | integer                |

base_exemple=#
```

N'hésitez pas à poser des questions pendant et après la séance du cours/TD.

### Exercice 1 Création de tables

- Créer la table EMP dont la clé est NoEMP et les attributs ont pour types :

Attribut	Type	Signification
NoEMP	NUMBER(4)	nombre sur au maximum 4 chiffres
NomEMP	VARCHAR2(10)	chaîne de caractères d'au maximum 10 caractères
EMPLOI	VARCHAR2(15)	chaîne de caractères d'au maximum 15 caractères
MGR	NUMBER(4)	nombre sur au maximum 4 chiffres
DateEMB	DATE	
SAL	NUMBER(7, 2)	nombre sur au maximum 7 chiffres dont deux après la virgule
COMM	NUMBER(7, 2)	nombre sur au maximum 7 chiffres dont deux après la virgule
NoDEPT	NUMBER(2)	nombre sur au maximum 2 chiffres

- Remplir cette table par les données suivantes :<sup>2</sup>

NoEMP	NomEMP	EMPLOI	MGR	DateEMB	SAL	COMM	NoDEPT
7369	SERGE	FONCTIONNAIRE	7902	17-DEC-1980	800		20
7499	BRAHIM	VENDEUR	7698	20-FEB-1981	1600	300	30
7521	NASSIMA	VENDEUR	7698	22-FEB-1981	1250	500	30
7566	LUCIE	GESTIONNAIRE	7839	2-APR-1981	2975		20
7654	MARTIN	VENDEUR	7698	28-SEP-1981	1250	1400	30
7698	BENJAMIN	GESTIONNAIRE	7839	1-MAY-1981	2850		30
7782	DAYANE	GESTIONNAIRE	7839	9-JUN-1981	2450		10
7788	ARIJ	ANALYSTE	7566	09-DEC-1982	3000		20
7839	MAYAR	PRESIDENT		17-NOV-1981	5000		10
7844	ROI	VENDEUR	7698	8-SEP-1981	1500	0	30
7876	VIRGINIE	FONCTIONNAIRE	7788	12-JAN-1983	1100		20
7900	LYNA	FONCTIONNAIRE	7698	3-DEC-1981	950		30
7902	ASMA	ANALYSTE	7566	3-DEC-1981	3000		20
7934	SIMONE	FONCTIONNAIRE	7782	23-JAN-1982	1300		10

2. INSERT INTO EMP VALUES (7369, 'SERGE', 'FONCTIONNAIRE', 7902, TO\_DATE('17-DEC-1980', 'DD-MON-YYYY'), 800, NULL, 20);

3. Créer la table **DEPT** dont la clé est **NoDEPT** et les attributs ont pour types :

Attribut	Type
NoDEPT	NUMBER(2)
NomDEPT	VARCHAR2(14)
LOC	VARCHAR2(13))

4. Remplir cette table par les données suivantes :

NoDEPT	NomDEPT	VILLE
10	COMPTABILITE	BREST
20	RECHERCHE	RENNES
30	VENTES	DINAR
40	GESTION	DINAN

5. Faire en sorte que la clé étrangère soit correctement définie, après avoir défini et rempli les deux tables.

## Exercice 2 Requêtes SQL simples

- Afficher tous les tuples de la table **EMP**
- À partir du contenu de la table **EMP** afficher les noms et les emplois de tous.
- Afficher les tuples correspondant au département 20 dans la table **EMP**.
- Rechercher et afficher les noms et salaires des employés exerçant en tant que vendeur et ayant une commission.
- Afficher tous les numéros de département. Est-il possible d'améliorer le contenu d'affichage ? Commenter.
- Lister les gestionnaires. On ne s'intéressera qu'à leurs noms, leurs salaires et leurs commissions.
- Déterminer la ville du département de **ASMA**.
- On s'intéresse aux employés gagnant entre 1000 € et 3000 €. Proposer une requête assurant cette tâche.
- Lister les noms des départements dont le nom de ville commence par **R** ou **D**.
- Lister les noms des départements dont le nom de ville contient un **R** ou un **D**.
- Dans l'ordre décroissant des salaires, lister les noms, les fonctions et les salaires des employés gagnant moins de 3000 €.
- Dans l'ordre décroissant des numéros des départements ensuite dans l'ordre croissant des salaires, lister les noms, les fonctions, les salaires des employés et les numéros des départements.
- Dans l'ordre décroissant des noms des départements ensuite dans l'ordre croissant des salaires, lister les noms, les fonctions, les salaires des employés et les noms des départements.

## Exercice 3 Création et mise à jour

- On souhaite revaloriser le salaire de tous les employés de 200€ chacun. Proposer l'instruction permettant de le faire sur la table **EMP**.
- Insérer un nouveau tuple dans la table **DEPT**. Le contenu du tuple est laissé à votre choix.
- Écrire et exécuter l'instruction de création de la table **COMMISSION** dont les attributs sont :
  - NomEmp** de type chaîne de caractères (**VARCHAR(10)**)
  - Emploi** de type chaîne de caractères (**VARCHAR(10)**)
  - Salaire** de type réel (**number(7,2)**)
  - ValComm** de type réel (**Number(4,2)**)
- Insérer plusieurs tuples dans cette table.
- Supprimer de la table **COMMISSION**, le tuple correspondant à un nom d'employé (au choix).
- Insérer dans cette table les données de la table **EMP** dont les employés ont une commission.
- Vider la table **COMMISSION** ensuite supprimer là.

8. Recréer la table `COMMISSION` en y chargeant directement les données de la table `EMP` dont les employés ont une commission.

**Exercice 4 Fonctions d'agrégat, ANY, ALL, GROUP BY, HAVING, etc.**

1. Lister le nombre d'employés gagnant plus que le minimum de tous les salaires
2. Afficher le nombre de départements différents dans la table `EMP`
3. Afficher le salaire moyen dans la table `EMP`
4. Lister tout employé dont le salaire est supérieur à au moins un salaire du département 30.
5. Lister tout employé du département 10 dont le salaire est supérieur à tous ceux du département 20.
6. Lister les salaires minimal et maximal pour chaque département
7. Lister le salaire moyen par emploi (`EMPLOI`)
8. Lister les salaires minimal et maximal pour chaque département ayant au moins 2 employés
9. Lister les départements ayant le minimum des salaires supérieur à la moyenne des salaires des employés de bureau (`FONCTIONNAIRE`)
10. Donner les noms des employés par ordre alphabétique.
11. Donner les noms des employés par ordre alphabétique inversé.
12. Lister les noms des employés dont le nom de manager commence par un H ou un S ou un un M.
13. Lister les noms des employés dont le nom de manager se termine par un E ou un M ou un un A.
14. Donner les informations sur les employés par ordre décroissant de date d'embauche puis par ordre alphabétique de nom.
15. Présenter les employés par groupes de même valeur de salaire, et par ordre décroissant de ces valeurs.
16. Représenter les employés groupés selon les noms de département ordonnés dans l'ordre alphabétique inverse.
17. Représenter les employés groupés selon le mois et l'année de leur date d'embauche (ordre chronologique).
18. Afficher le salaire moyen par job.
19. Afficher les noms des employés groupés par job puis par nom de département.
20. Afficher les nombres d'employés dirigés directement par chaque manager (ici on comprend manager comme un employé ayant son numéro dans la colonne mgr).
21. Afficher la ville et le nom des départements qui ont plus de quatre employés.
22. Afficher le nom des départements où les employés gagnent en moyenne plus de 2500€.
23. Afficher le nombre d'employés par département et par profession.
24. Donner pour chaque département, son nom, le salaire minimum de ses employés, le salaire maximum, le salaire moyen, le nombre d'employés, le nombre d'employés touchant une commission (éventuellement nulle) et le nombre de jobs différents exercés dans ce département.

**Exercice 5 Autres aspects de SQL**

1. Créer une vue `EMPDEPT` qui contient la jointure naturelle des deux tables `EMP` et `DEPT`
2. En utilisant cette vue, afficher les noms d'employés touchant plus que 2000 € avec leurs villes
3. Vider la table `EMP`
4. Supprimer la table `EMP`
5. Créer à nouveau la table `EMP` avec les mêmes attributs et les mêmes contraintes d'intégrité
6. Supprimer la colonne `COMM`
7. Renommer la colonne `COMM` en `BONUS`
8. Ajouter une colonne à la table `DEPT`