

# Advanced Database - TP 1

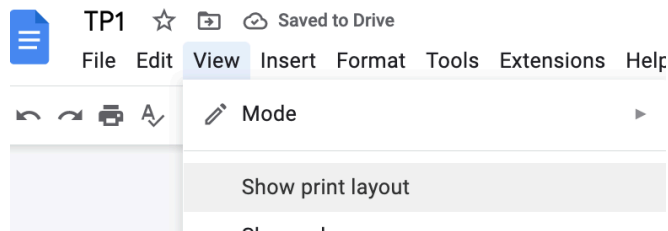
## SQL

### Install PostgreSQL and client

Follow instructions [here](#).

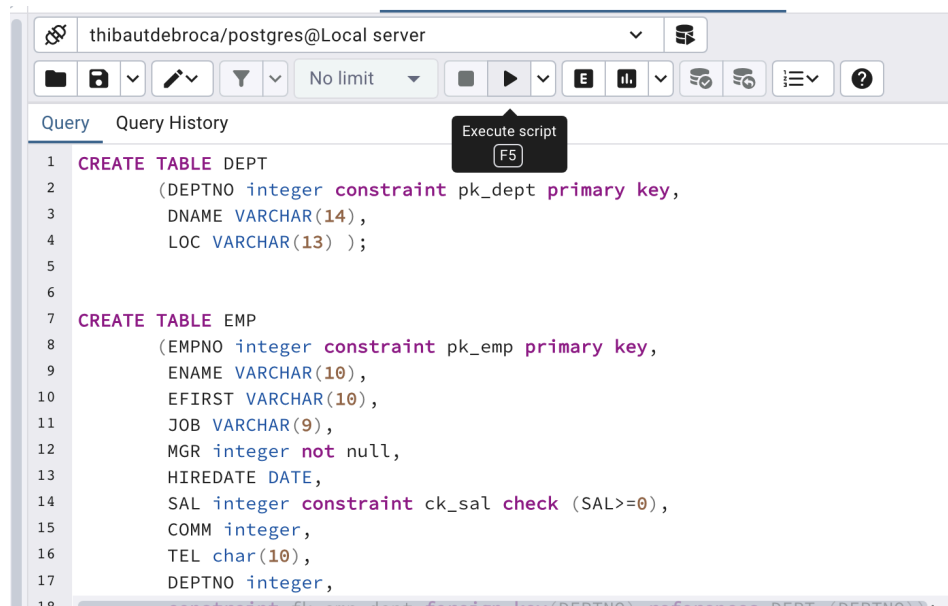
### Tip using Google Doc

In order to read more easily the google doc, I recommend you un-check this option in Google Doc:



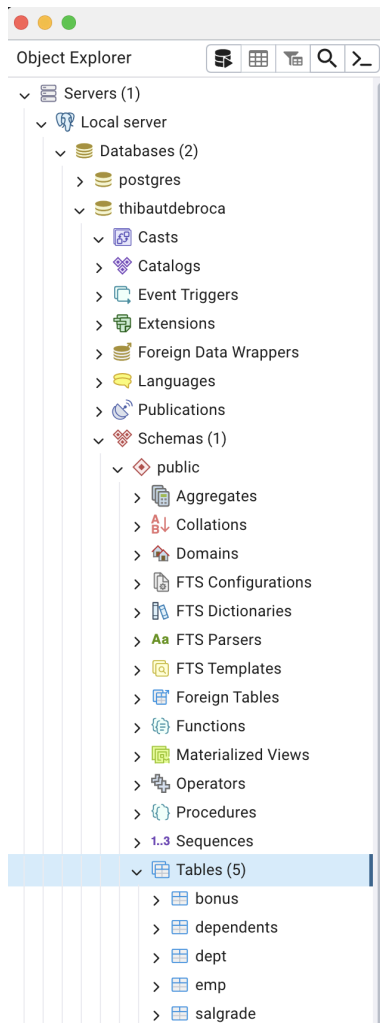
### Exercise 1: DDL queries

1. **Copy** the content of the script [creation.sql](#) and paste in a Sql editor in Postgresql.

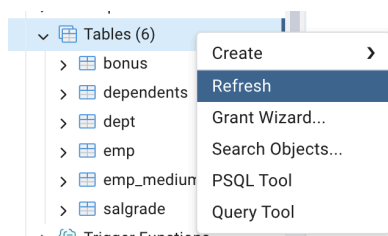


Run the creation script (F5).

Check if the tables appear in the list of tables:



If you can't see the tables, right click and click on "refresh":



**2. E/R diagram:** Identify the different tables created and integrity constraints to ensure data consistency. Draw the entity / relational model from the given database.

Note: Dependents mean a person who depends on another person to be affiliated to the social Security System. In that case a dependent can be a child whose father is an employee (defined

by empno link).

3. **First integrity Constraint:** Define an integrity constraint to prohibit the possibility of two employees to own the same firstname, lastname and phone number. Use the alter table statement.

4. **Second integrity Constraint:** More and more employees have two phone numbers: a fixed and a mobile phone. Unfortunately, the database allows us to store one phone number. Bring the necessary modifications to the database. Set up an integrity constraint in order to make sure that the entered mobile number begins with 06.

5. **Third integrity Constraint On Delete:** In practice, we realize that when one wants to delete an employee, it is necessary to remove all his dependents. How is it possible to make such removal dynamic? Do not forget to first remove the referential constraint fk\_dependent\_emp that already exists.

6. **Explain Errors for Integrity Constraint:** Fill the tables with the [insert.sql](#) script. Explain the errors that you could get and correct them.

7. **Define a sequence** to make easy creation of a new dependent. This sequence has to start with the value 8000 and an increment gap of 1.

The instruction to create a sequence:

```
CREATE SEQUENCE ma_sequence [START WITH] ... [INCREMENT BY]...  
[MINVALUE]...[MAXVALUE]...[CYCLE|NOCYCLE];
```

8. **Use Sequence:** Add some tuples to the DEPENDENT table using the previous sequence Use nextval('ma\_sequence') to get the values from the sequence

9. **Discuss** on the best way to do an auto-increment with postgres. Apart from Sequence, there is also the “serial” data object. But there is also the “identity” since postgres 10 (2017):

[https://wiki.postgresql.org/wiki/Don%27t\\_Do\\_This#Don.27t\\_use\\_serial](https://wiki.postgresql.org/wiki/Don%27t_Do_This#Don.27t_use_serial)

<https://stackoverflow.com/a/55300741/1029722>

## Exercise 2: DML queries Answer the following queries using SQL.

1. List the content of all tables to see the attributes names
2. Select employees whose commission is higher than their salary
3. Select employees earning between 1200 and 2400 (earning is sal + commision)
4. Select employees who are CLERK or ANALYST

5. Select employees whose name begins by M
6. Select employees whose name includes a L in second position
7. Select employees who are MANAGER or CLERK in the department 10 and whose salary is greater than 1500
8. Select employees whose commission is NULL
9. Select employees by ascending order (by hiredate)
10. Select employees ordered by job, and for each job, by decreasing salary
11. Select departments without employees
12. List employees indicating for each the name of his/her manager
13. List employees earning more than JONES
14. List employees displaying in the same column salary and commission
15. List department numbers which are both in table EMP and in table DEPT
16. List employees working in CHICAGO and having the same job than JONES
17. List employees who don't work in the same department than their manager
18. List employees working in a department having at least one CLERK
19. List employees of department 10 having the same job than someone from the department SALES
20. List employees having the same job than JONES or a salary greater than FORD's salary
21. List employees having a salary greater than all employees of department 20

### **Exercise 3: Join Table.**

**1. Create a Table for employee's Projects:** Each employee is working on one or several projects. Create a table "project" containing the number, name, starting date and budget of each project.

Write the corresponding SQL Request

Table name is: "project"

Attributes are: "projno", "pname", "startdate", "budget"

**2. Create the Join Table:** An Employee can work on many projects and a project can be affected by many employees. Create the necessary tools for that. Insert some elements in the tables.

Write the SQL query to create the corresponding table (name of the table is : "project\_emp"). Tip: don't forget foreign keys

Write some requests to insert elements in the table:

- You should have 4 projects in the project table
- In the join table 'project\_emp' you should have at least 20 lines.
  - One of the employee should be assigned to all the projects

### 3. List all employees (by empno) who work on all projects ?

Write 1 SQL query to answer this question, return only the employee number (empno)

### 4. Options on View creation: Explain the following instruction

```
CREATE VIEW sales_staff AS
SELECT empno, ename, deptno
FROM emp
WHERE deptno = 10 WITH CHECK OPTION
```

### 5. View Creation: Create the view and try the following queries, explain the result

INSERT INTO sales\_staff VALUES (7584, 'OSTER', 10);

INSERT INTO sales\_staff VALUES (7591, 'WILLIAMS', 30);

### 6. Analyze: Give the number of projects per employee and display only employees assigned to at least 2 projects.

### 7. Set Queries:

- Find employees who have worked on project number 1 and also on project number 2
- Find employees who have worked on project number 3 but never on project number 4

### 8. Find Top 3 employees per project by their salary.

**9.1** For each employee, display the percentage of projects he has been working on (for example, if there are 10 projects in the project table, and the employee is working on 3 projects, you should display 30%).

**9.2** Same query than before, but create another column 'scope\_size':

- If the percentage of projects is 0, indicate value 'Empty'
- If the percentage of projects is between 10 and 50, indicate value 'Small'
- If the percentage of projects is between 50 and 80, indicate value 'Medium'
- If the percentage of projects is between 80 and 100, indicate value 'Large'
- If the percentage of projects is 100, indicate value 'Total'

## Training links for little test

Test of 2023:

- [https://docs.google.com/document/d/13FFc\\_yTq-U-K5NIyu-J\\_OKYUAEvOOCHbb-MrsACUtHk/edit#heading=h.ihrs8dc471eg](https://docs.google.com/document/d/13FFc_yTq-U-K5NIyu-J_OKYUAEvOOCHbb-MrsACUtHk/edit#heading=h.ihrs8dc471eg)

Chess:

- [https://docs.google.com/document/d/1EkEYwghMCNQzAZ4H2YMb94CaSYUkqWgN-\\_0v4beyoew/edit](https://docs.google.com/document/d/1EkEYwghMCNQzAZ4H2YMb94CaSYUkqWgN-_0v4beyoew/edit)
- [https://docs.google.com/presentation/d/1DGu3c8bc7dt\\_R4Cbe37\\_0o-XGnx4hXU8/edit#slide=id.p1](https://docs.google.com/presentation/d/1DGu3c8bc7dt_R4Cbe37_0o-XGnx4hXU8/edit#slide=id.p1)
- [https://docs.google.com/presentation/d/1IJDwlbjukHXL9qS-n2cRXgDDhRCDsGbHusjDmOnmUec/edit#slide=id.gd45c7b0bb5\\_1\\_564](https://docs.google.com/presentation/d/1IJDwlbjukHXL9qS-n2cRXgDDhRCDsGbHusjDmOnmUec/edit#slide=id.gd45c7b0bb5_1_564)