

LAB 2

Project Background

Start by revisiting the case study for the customers' requirements for this project in order to identify all foreign keys. Create normalized schemas with all the attributes domains using the case study as part of Homework 2 and create an ER-diagram as part of homework 3. The queries only need to be able to execute in PostgreSQL, so there is no need to create a Graphic User Interface to visualize your data.

Rules

The installation guide provides students with the proper software needed to complete this lab.

1. The labs must be done in groups of exactly two people. No larger groups are allowed, and if you have extraordinary extenuating circumstances that force you to do the labs alone, you must obtain permission to do so from the course leader. Both students in a group must be able to present all of the lab for the group to pass. Lab assistants do not record partial labs.
2. You must present correct and valid solutions to all the given tasks in order to pass the lab.
3. Presenting P+ assignments are optional for a higher grade if the given tasks are completed and passed.
4. This is a PSQL lab. No other programming languages, either embedded in the database or external to it, are allowed.
5. Please refrain from creating any [functions](#) since this lab is designed to assess query programming languages. Usage of built in PSQL functions is acceptable.
6. You are not allowed to hard-code anything except that which has been explicitly given to you in the problems. In particular, this means that constructs like limit 1 or similar artificial ways of reducing the output are forbidden.
7. You must utilize nothing but a single top-level SELECT statement to answer every problem(although you are, of course, allowed to use any number of sub-selects required within that top level statement). You are specifically forbidden from referencing any temporary data structures like views or temporary tables from the solutions.

LAB 2

Lab Presentation

Course related terminology is expected during the lab presentation.

Make sure you have the following ready to be presented to the TA:

1. Thorough description of the database design using an ER-diagram as part of homework 3.
2. Code for the database creation showing keys, domains etc.
3. Simple executable select statements to show the contents of the database.
4. Motivations for how the solution for each requested queries are sensible. (Not all tasks have one query as an answer, but they have to be good enough for e.g. a client to accept.)

Altering the database

OBS! The database is required at this stage to be in BCNF, alter the database to fulfil this requirement as necessary. Also alter and add foreign keys to all the tables that are necessary for the LMS.

Tips:

Name the constraints appropriately.

Foreign keys require the existence of a primary key to reference to.

Existing data may conflict with the foreign key constraint, if so then update or delete the conflicting data, but make sure there still is data left as according to the previous lab requirements.

Querying the database

The following statements for querying a database are to be executed and should show a satisfactory result.

1. Find a book and display only its title and author.

Expected output format:

```
title | author
-----+-----
Book  | Someone
```

2. Display all books published after a specific date and order by published date.

Expected output format:

```
title | published
-----+-----
```

3. Count all the books currently borrowed.

Expected output format:

```
count
-----
```

4. Calculate and display the sum of all paid fines.

LAB 2

Expected output format:

```
sum
----
```

5. Find all the books from one genre and display only its title, author, genre and a relevant column of your choice.

Expected output format:

```
title | author | genre | choice
-----+-----+-----+-----
```

6. Find all unpaid fines and display the late book, the fine amount, the fine owner and a relevant column of your choice.

Expected output format:

```
title | fine | name | choice
-----+-----+-----+-----
```

LAB 2

1. P+

Rewrite all previous queries from this lab using [CTE:s](#).

During the presentation you are expected to present:

- All queries ready to be executed with the exact same result as the original query.

2. P+

Using the [recursive method](#), sum all odd numbers between 1 and 50.

During the presentation you are expected to present:

- All queries ready to be executed.
- Motivations for your solution.