



Database Management Systems, January 2024

Homework 1: SQL

by Hildur Davíðsdóttir & Björn Þór Jónsson

Introduction

The goal of this homework assignment is to train you in writing SQL queries against a previously unknown database. To that end, we give you a database script and a description of 10 queries, and you must produce two files, one with the SQL queries, and the other with the output of the SQL queries, produced using `psql`. Since this project description document has much detail, both regarding the database and your submission, we recommend **reading it carefully**. If you have any questions, please ask us on Piazza.

Database description

In this homework you will work with a Gym Management database for the *Awesome Gym* franchise. The database contains information on gyms, members, instructors, classes, types of classes, equipment and all their relationships. The database represents data for the gym's operations in January 2023. The data is artificially and randomly generated and may, or may not, reflect reality.

The following is the **database schema** and **ER-diagram** of the database:

Instructor (*ID*, *name*, *phone*)

Member (*ID*, *name*, *phone*, *start_date*, *quit_date*, *IID*)

Gym (*ID*, *address*, *email*)

Type (*ID*, *name*, *capacity*)

Equipment (*ID*, *name*, *price*)

Class (*ID*, *IID*, *TID*, *GID*, *date*, *minutes*)

Attends (*MID*, *CID*, *rating*)

Needs (*TID*, *EID*, *quantity*)

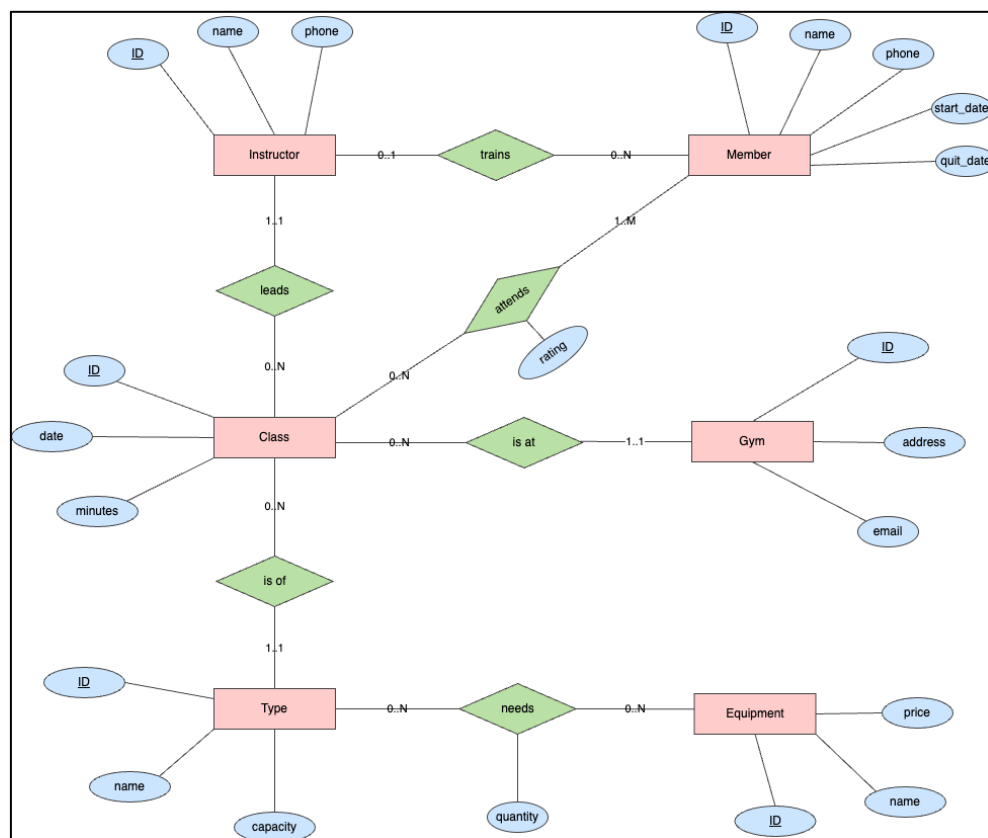


Figure 1: ER Diagram for the Gym Management database.

To better understand the database and its tables, you should study the DDL commands (the CREATE TABLE statements are at the top of the HW1-DB.sql file), study the database schema and ER-diagram in *Figure 1* and inspect the table data using SQL queries. Primary and foreign keys are defined in the DDL. Entities, attributes, and relationships are largely self-explanatory, but further details are below.

The following list states noteworthy features of the database as well as some limitations (simplifications of the real world). Read this **carefully**.

- Instructors can both lead classes (such as yoga and spinning) and have members in personal training (see ‘trains’ relation).
- Classes can only have one instructor, be of one type and be located at one gym. This is why there is an Instructor ID, Type ID and Gym ID in the Class table; these are a One-To-Many relationships.

Instructions

To start working with the database, you must first create it, and then run the database script *HW1-DB.sql* against the database using psql on your laptop, as discussed in class.

Write SQL queries to retrieve the information requested below from your PostgreSQL database. The queries should not return anything except the requested answers. The queries should also be as simple and readable as possible and consistently formatted, as SQL queries are generally part of your code base. To make the most of the homework assignment, you should take time to not only solve the queries, but test variants.

While subqueries are fine (and sometimes necessary), you should not produce a sequence of several queries that allow you to answer the question, you should only write a single query for each question. Queries should avoid system-specific features, including the LIMIT keyword and you should not use magic constants. Note that occasionally, some incorrect variants of some queries could return the same results as a correct query. The fact that a query returns the correct answer thus does not necessarily mean that the query itself is correct. In short, the query must work for *any instance of the schema*.

It is very useful to insert additional data to test your queries adequately. Of course, if you add test data, you can always return to the original data by re-running the database installation script. It is also very useful to use the **hints** provided in some of the problems to check whether you are on the right track. If you are unable to complete a query you can still include your **attempt**, along with a brief description in comments, as it may be given partial points.

Many queries ask you to count the rows that satisfy some criteria. We highly recommend starting by creating a query to return all the rows satisfying the given criteria. Once your query is ready, you can then turn it into a counting query, either by replacing the output columns with a *count(*)* statement, or by enveloping the query with a counting query as shown in *Figure 2*:

```
select count(*)
from (
    select ...
) tmp;
```

Figure 2: A counting query enveloping another query.

In some queries, you may need to use PostgreSQL built-in operators. Here is a list of some useful operators. You should study the functionality of the operators to understand where they may be useful.

| | | |
|---------|-----------|---------|
| avg() | extract() | round() |
| count() | lower() | |
| sum() | max() | |

Deliverables

This project is a group project, with **3** students per group (you must form your own groups). Individual submissions will not be accepted, unless you have been granted permission by the teachers. The deadline is at **23:59 on Sunday, February 4th**. Late submissions will **not be accepted**.

Each group must submit two files to **Gradescope** (See submission instructions on Canvas):

- An sql file named **QUERIES.sql**, containing all the queries used to retrieve the requested data. The queries should be in the correct order, with a **comment** preceding each query with the query number (A...J) **as well as 1-3 sentences explaining your solution**. It must be possible to run the **QUERIES.sql** file using **psql** without errors. We recommend using the provided template for your submission, but remember to rename your file to **QUERIES.sql**.
- A txt file named **RESULTS.txt**, containing the results of all queries, produced by running **QUERIES.sql** with **psql**.

Queries

- A. How much does the most expensive equipment cost? Return only the price.
- B. 792 members have started the gym in April (of any year). How many members have started the gym in January (of any year)?
- C. 154 classes were held with 'burn' somewhere in their type name. How many classes were held that have 'fit' somewhere in their type name? (Note that your query should be case-**ins**sensitive, i.e. classes with 'fiT' and 'Fit' in their type name should also be counted).
- D. How many different instructors have led at least one class in which a member that they have in personal training attended?
- E. Return the name of every class type along with the average rating that all classes of the type have received. The result should be rounded to the nearest integer and ordered from highest to lowest. Name the column with the average rating "Average Rating".
- F. How many members have not attended any classes and do not have a personal instructor?
- G. 43 instructors have led 15 or more classes. How many instructors have led 10 or more classes?
- H. For how many members is it true that there exists at least one other member with the same start date and quit date as them? (Note that if that is true for John and Mary, they should be counted as two results. Note also that two people that have not quit cannot be considered as having the same quit date).
- I. How many classes were held in gyms in Reykjavik and have a capacity of either 30 or 40 people, but the capacity was not used fully?
- J. Return the ID and name of the member(s) that attended classes for the longest total time (in minutes) of all members.