Assignment: SQL #3

Avraham Leff

COM3580: Fall 2020

1 Assignment-Specific Packaging

The general packaging is unchanged from the basic "Homework Requirements" (see slides from first lecture and "Homework Policies for COM 3563" on Piazza).

This assignment's "DIR" **must be named** *SQL3*. Your report **must be named** *$DIR/SQL3.pdf*.

2 Database Installation

Some of the exercises in this assignment are based on the Silberschatz textbook's *University database*. You are already familiar with this database, but to make the results more interesting you are going to use the "large database" version of this schema.

Use the scripts from this git repository to create and populate the "large university database".

> The one difference between the "small" and "large" versions of the database is that table names in the small version have an **_s** suffix. For example: *student_s* rather than *student*.

2.1 Advice

To avoid confusion between "small" and "large" versions of the database, you may want to install the large version in a different database from the (probably "default") database that currently store the small version's tables. When writing your SQL scripts, make sure to \connect to the correct database.

For Exercise 4.18, **do not use "SQL1_Populate.sql" script** from this repository. Instead: use the "SQL2_Populate.sql" script because it represents *employees with no manager as employees with a NULL value for manages_id*. (You may want to compare & contrast to the approach taken in the "SQL1_Populate.sql" script.)

3 Requirements

Note: in all such exercises, you are responsible for (re)initializing the database before executing your SQL commands. All subsequent interactions with the database must execute the specified steps, **in the specified order**!

- Although the assignment references the textbook exercises, **pay attention** to any changes that I've made to the textbook version!

- For non-programming exercises, make sure to keep your answers short and on target! For *natural language* queries, you must

  1. Show me a formatted SQL "translation" of that query
  2. Show a **clear snapshot** of your POSTGRESQL executing that query and the results therefrom. Your snapshot should include the "number of rows" in your result set.

- Note: most, if not all, your grade depends on "correctness": i.e., the result of your query!

1. Textbook, Exercise 4.14 (**no more than two sentences!**)

2. Textbook, Exercise 4.19

3. Exercise 4.18 (a) “using an outer join” …

   Order your results by ascending employee id.

4. Exercise 4.18 (b) “this time, not using an outer join at all” …

   Order the results by ascending employee “id”.

5. Textbook, Exercise 4.2(a)

   - Order by ascending instructor id
   - Display only the first five results

6. Textbook, Exercise 4.2(b)

   - Order by ascending instructor id
   - Display only the first five results

7. Textbook, Exercise 4.2(c), **but modified** to:

   - The year of interest is **2008**, and do not further qualify the query by a specific semester
   - Display year, semester, course id, section id, instructor id, instructor name
   - Order by ascending year, then by ascending course id