SQL Bridge – Sample Final

Your SQL Bridge Final Exam is due by end of day on Sunday July 26*. You should post a single .SQL script to GitHub that creates and populates the tables, and joins the information as described below. You are also expected to make a short (3 to 5 minute) presentation during our last Meetup on July 28th.*

Please note:

(1) *You may substitute a SQL project of your own choosing that demonstrates your understanding of the three capabilities highlighted below. You will need to provide a requirement document similar to what is below if you work with your own data and requirement. You may work in a small group if you take on this more ambitious alternative final.*

(2) *If you are also taking the R bridge and or the Data Science Math bridge, you may instead design a final project of your own choosing that incorporates what you have learned in both or all three of your bridge courses. For example, you might design and populate tables in SQL, analyze the exported data in R, then include some conditional probability in your analysis. This will require some forethought on your part. You will only be required to present once if you choose to combine finals.*

This project is where you show off your ability to (1) translate a business requirement into a database design, (2) design a database using one-to-many and many-to-many relationships, and (3) know when to use LEFT and/or RIGHT JOINs to build result sets for reporting.

An organization grants key-card access to rooms based on groups that key-card holders belong to. You may assume that users below to only one group. Your job is to design the database that supports the key-card system.

There are six *users*, and four **groups**. *Modesto* and *Ayine* are in group "**I.T.**" *Christopher* and *Cheong Woo* are in group "**Sales**". There are four ***rooms***: "***101***", "***102***", "***Auditorium A***", and "***Auditorium B***". *Saulat* is in group "**Administration**." Group "**Operations**" currently doesn't have any users assigned. I.T. should be able to access Rooms 101 and 102. Sales should be able to access Rooms 102 and Auditorium A. Administration does not have access to any rooms. *Heidy* is a new employee, who has not yet been assigned to any group.

After you determine the tables any relationships between the tables (One to many? Many to one? Many to many?), you should create the tables and populate them with the information indicated above.

Next, write SELECT statements that provide the following information:

- All groups, and the users in each group. A group should appear even if there are no users assigned to the group.
- All rooms, and the groups assigned to each room. The rooms should appear even if no groups have been assigned to them.
- A list of users, the groups that they belong to, and the rooms to which they are assigned. This should be sorted alphabetically by user, then by group, then by room.