CS 486/586 Introduction to DBMS
Winter 2020

Assignment 1 – Basic SQL Queries
Due: Friday, January 17, 11:59PM on D2L

You may do this assignment individually or you may work with one partner. Please turn in your completed assignments on D2L. If you do the assignment with a partner, you need only turn in one assignment.
This assignment is based on the Spy relational database. Information about this database and information about how to access it is on the Database Info Page. (http://web.cecs.pdx.edu/~maier/db_resources/db.htm)

The final page of the homework contains some useful PSQL commands. Take a look!

Part One (25 Points)

1) Describe the following tables: agent, skill, skillrel, mission, securityclearance from the spy database schema in the format shown below. Underline the attributes making up the primary key of each table and describe foreign keys in a separate line as shown.
TableName1(Attribute1, Attribute2, Attribute3,…)
Attribute2 is a foreign key to Table3(Attribute2)
Or: Attribute2 -> Table3(Attribute2)

For 2a through 2h, please give the answer – you do not have to write SQL queries to answer these questions, you only have to examine the schema of the spy database.

2a) Can two languages have the same name? (Note: name refers to the attribute/column named 'language' in the relation languages)

2b) How many languages can an agent speak?

2c) How many security clearance levels can an agent have?

2d) How many agents can have a given skill?

2e) How many teams can participate in any given mission?

2f) Can two agents have different affiliation strengths to the same organization?

2g) Can one agent have different affiliation_strengths to two or more different organizations?

2h) Can an agent participate in more than one ongoing mission?

Part Two (5 points per item) Give the English request that could have resulted in each of the SQL queries below. (Don't just paraphrase the SQL into words.) Also include the first five rows of the result for each query (or fewer, the result is smaller), and the total number of rows returned.

3.
(a) SELECT * FROM Agent WHERE city = 'Istanbul' AND country = 'Turkey';
(b) SELECT city FROM Agent;
(c) SELECT DISTINCT city FROM Agent;
(d) SELECT agent_id, city, country FROM Agent WHERE salary > 100000 AND country != 'USA';

4.
(a) SELECT city FROM Agent WHERE salary > 90000;
(b) SELECT city FROM Agent
WHERE Agent.salary > 90000;
(c) SELECT city FROM Agent A WHERE A.salary > 90000;

5.
(a) SELECT DISTINCT sc_level FROM SecurityClearance;
(b) SELECT * FROM Agent A, SecurityClearance S
WHERE A.city = 'London'
AND (S.sc_level = 'Top Secret');
(c) SELECT * FROM Agent A, SecurityClearance S
WHERE A.city = 'London'
AND (S.sc_level = 'Top Secret')
AND A.clearance_id = S.sc_id;

6.
(a) SELECT A.agent_id, A.first, A.last, A.city, A.country
FROM Agent A, TeamRel TR, Team T
WHERE A.agent_id = TR.agent_id
AND TR.team_id = T.team_id
AND T.name = 'Giraffe';

(b) SELECT A1.first, A1.last, A2.city, A2.country
FROM Agent A1, Agent A2
WHERE A1.city = A2.city
AND A1.country = A2.country
AND A1.clearance_id > A2.clearance_id
AND A1.salary >= A2.salary;

Part Three (10 points each) Write a single SQL statement for each of the following queries. Show the first five rows of the result for each query (or fewer, if the result is smaller) and the number of rows returned. You should be able to write these SQL queries using only the features covered in the first lecture notes. Throughout this class (and in real life!) when writing queries, make sure you are using the data you are given. Eg. If you are asked to find all the agents who speak Hindi, your query should contain something along the lines of "language = Hindi" and not "lang_id = 19". **There are good reasons for this – ask on Slack if you're curious about what they are!**

7. What are the team ID and the meeting frequency for the ShowBiz team?

8. Which countries have agents with Classified or Magellon clearance? (**Your query shouldn't depend on what clearance IDs are used for these clearance levels, just the names of the levels.**)

9. What are the first name, last name, and city of all agents on at least two teams? (You can do this without COUNT.)

10. List the name and status of all missions that have at least one agent with the skill Kung Fu. Don't repeat missions in your result.

11. Which pairs of agents have the same first and last names? (List each pair only once.)

Useful PSQL Commands:

\dt - list all tables

\d schemaname.tablename - display info (schema) of a table

\? - help with 'backslash' commands

\q - quit psql

help - help with SQL commands (also can use help select for help on select)

Quotes:

In the syntax Type = "checking" AND Balance > 1000, why there are quotes around checking, but not around 1000 (or Type or Balance).

Type and Balance are attribute names, so do not need quotes,. 1000 is an integer or numeric value and doesn't need quotes. But checking is a string, and strings do need quotes. The other error in that syntax is that postgres uses single quotes, rather than double quotes use single quotes in postgres, not double quotes.

Temp Tables:

Postgres allows you to create temporary tables from other tables if necessary – it may be of some use in sanity checking some of the SQL queries you write for your homework assignments.

As an example of how this is done, the syntax:

SELECT first, last INTO agentname FROM agent;

will create a permanent table that only contains the first and last attributes from the agent table. This table will need to be deleted when you are done with it, otherwise it will persist indefinitely. If you want to create a temp table instead, which will be deleted automatically by Postgres when you end your session, you can use the following syntax:

CREATE TEMP TABLE agentname AS SELECT first, last FROM agent;