

Database Technology– FSS 2018

Exercise 3: SQL Part 2

3.1. SQL queries using university database

Write the following inserts, deletes or updates in SQL, using the university schema.

- a. Increase the salary of each instructor in the Comp. Sci. department by 10%.
- b. Delete all courses that have never been offered (that is, do not occur in the section relation).
- c. Insert every student whose tot_cred attribute is greater than 100 as an instructor in the same department, with a salary of \$10,000.
- d. Display a list of all instructors, showing their ID, name, and the number of sections that they have taught. Make sure to show the number of sections as 0 for instructors who have not taught any section. Your query should use an outerjoin, and should not use scalar subqueries.
- e. Write the same query as above, but using a scalar subquery, without outerjoin.
- f. Display the list of all departments, with the total number of instructors in each department, without using scalar subqueries. Make sure to correctly handle departments with no instructors.
- g. Create a new course “CS-001”, titled “Weekly Seminar”, with 0 credits.
- h. Create a section of this course in Autumn 2017, with sec_id of 1.
- i. Enroll every student in the Comp. Sci. department in the above section.
- j. Delete enrollments in the above section where the student’s name is Chavez.

3.2. Insurance database

Consider the following insurance database. Construct SQL queries for this relational database.

person(driver_id, name, address)

car (license, model, year)

accident (report_number, date, location)

owns (driver_id, license)

participated (report_number, license, driver_id, damage_amount)

- a. Find the total number of people who owned cars that were involved in accidents in 1989.
- b. Add a new accident to the database; assume any values for required attributes.
- c. Delete the Mazda belonging to “John Smith”.
- d. Update the damage amount for the car with the license number “AABB2000” in the accident with report number “AR2197” to \$3000.

3.3. Employee database

Consider again the employee database and create SQL expressions.

employee (employee_name, street, city)
 works (employee_name, company_name, salary)
 company (company_name, city)
 manages (employee_name, manager_name)

- a. Modify the database so that Jones now lives in Newtown.
- b. Give all managers of First Bank Corporation a 10 percent raise unless the salary becomes greater than \$100,000; in such cases, give only a 3 percent raise.

3.4. Left outer join

Suppose we have three relations $r(A, B)$, $s(B, C)$, and $t(B, D)$, with all attributes declared as not null. Consider the expressions

- r natural left outer join $(s$ natural left outer join $t)$
 - $(r$ natural left outer join $s)$ natural left outer join t
- a. Give instances of relations r , s and t such that in the result of the second expression, attribute C has a null value but attribute D has a non-null value.
 - b. Is the above pattern, with C null and D not null possible in the result of the first expression? Explain why or why not.

3.5. Constraints

Enhance your DDL definition from Exercise 2.2. by creating reasonable constraints.

3.6. Views

Create a view in the university database for Einstein which shows all the names and department names of students which were taught by instructor Einstein.