

Database Management System (1)

Assignment #2

Chapter #3 SQL

Instructions:

- ✓ Deadline: 10-11-2022
- ✓ You must solve all the questions
- ✓ Cheating will give a zero grade for both students.
- ✓ Put the answers in one pdf file name it with 120201234-FirstName-LastName.pdf
- ✓ File size should not be more than 20 MB.
- ✓ SQL answers should be in the following form:

Query	Result																																
<pre>SELECT count(*) , semester , year FROM `takes` group by semester,year</pre>	<div>Showing rows 0 - 19 (20 total, Query took 0.0217 seconds.)</div> <div><div>SELECT count(*) , semester , year FROM `takes` group by semester,year</div><div><div>Show all</div><div>Number of rows: 25</div><div>Filter rows: Search this table</div><div>Sort by key: None</div></div></div> <div>+ Options</div> <table><thead><tr><th></th><th>count(*)</th><th>semester</th><th>year</th></tr></thead><tbody><tr><td><div><input type="checkbox"/> Edit Copy Delete</div></td><td>604</td><td>Fall</td><td>2001</td></tr><tr><td><div><input type="checkbox"/> Edit Copy Delete</div></td><td>2755</td><td>Fall</td><td>2002</td></tr><tr><td><div><input type="checkbox"/> Edit Copy Delete</div></td><td>1848</td><td>Fall</td><td>2003</td></tr><tr><td><div><input type="checkbox"/> Edit Copy Delete</div></td><td>856</td><td>Fall</td><td>2004</td></tr><tr><td><div><input type="checkbox"/> Edit Copy Delete</div></td><td>1239</td><td>Fall</td><td>2005</td></tr><tr><td><div><input type="checkbox"/> Edit Copy Delete</div></td><td>2428</td><td>Fall</td><td>2006</td></tr><tr><td><div><input type="checkbox"/> Edit Copy Delete</div></td><td>1773</td><td>Fall</td><td>2007</td></tr></tbody></table>		count(*)	semester	year	<div><input type="checkbox"/> Edit Copy Delete</div>	604	Fall	2001	<div><input type="checkbox"/> Edit Copy Delete</div>	2755	Fall	2002	<div><input type="checkbox"/> Edit Copy Delete</div>	1848	Fall	2003	<div><input type="checkbox"/> Edit Copy Delete</div>	856	Fall	2004	<div><input type="checkbox"/> Edit Copy Delete</div>	1239	Fall	2005	<div><input type="checkbox"/> Edit Copy Delete</div>	2428	Fall	2006	<div><input type="checkbox"/> Edit Copy Delete</div>	1773	Fall	2007
	count(*)	semester	year																														
<div><input type="checkbox"/> Edit Copy Delete</div>	604	Fall	2001																														
<div><input type="checkbox"/> Edit Copy Delete</div>	2755	Fall	2002																														
<div><input type="checkbox"/> Edit Copy Delete</div>	1848	Fall	2003																														
<div><input type="checkbox"/> Edit Copy Delete</div>	856	Fall	2004																														
<div><input type="checkbox"/> Edit Copy Delete</div>	1239	Fall	2005																														
<div><input type="checkbox"/> Edit Copy Delete</div>	2428	Fall	2006																														
<div><input type="checkbox"/> Edit Copy Delete</div>	1773	Fall	2007																														

- ☒ Show that, in SQL, \neq all is identical to **not in**
 - ☒ Show that, in SQL, $=$ some is identical to **in**
 - ☒ What is union-compatible mean? Mention all operations that require union-compatibility?
-
- ☒ Depending on the university Schema on the related files write the appropriate SQL statement then execute the SQL and show the data:
 1. Find the titles of courses in the Comp. Sci. department that have 3 credits.
 2. Find the IDs of all students who take courses in 2017 but not take any course in 2018 without duplication (solve using 2 ways)
 3. Find the highest salary of any instructor.
 4. Find all instructors earning the highest salary (there may be more than one with the same salary).
 5. Find the number of enrollments for each section that was offered in Fall 2017.
 6. Find the sections that had the maximum enrollment in Fall 2017.
 7. Increase the salary of each instructor in the Comp. Sci. department by 10%.
 8. Delete all courses that have never been offered (i.e., do not occur in the section relation).
 9. Insert every student whose tot cred attribute is greater than 100 as an instructor in the same department, with a salary of \$10,000.
 10. Find the ID and name of each student who has taken at least one Comp. Sci. course; make sure there are no duplicate names in the result.

11. For each department, find the maximum salary of instructors in that department.
You may assume that every department has at least one instructor.
12. Create a new **course** “CS-001”, titled “Weekly Seminar”, with 0 credits.
13. Create a **section** of this course in Fall 2017, with sec id of 1, and with the location of this section not yet specified.
14. Enroll every student in the Comp. Sci. department in the above section.
15. Delete enrollments in the above section where the student’s ID is 12345.
16. Delete all takes tuples corresponding to any section of any course with the word “advanced” as a part of the title; ignore case when matching the word with the title.
17. find the names of those departments whose budget is higher than that of Philosophy. List them in alphabetic order.
18. find the name and ID of each History student whose name begins with the letter ‘C’ order the result by total credit hours.
19. find the number of students in each section. The result columns should appear in the order “courseid, secid, year, semester, num”. You do not need to output sections with 0 students.

employee (ID, person_name, street, city)
works (ID, company_name, salary)
company (company_name, city)
manages (ID, manager_id)

Figure 1 Employee Schema

- ❖ Consider the employee database, where the primary keys are underlined. Give an expression in SQL for each of the following queries.
 1. Find ID and name of each employee who lives in the same city as the location of the company for which the employee works.
 2. Find ID and name of each employee who lives in the same city and on the same street as does her or his manager. (hint: use subqueries in **from** clause)
 3. Find ID and name of each employee who earns more than the average salary of all employees of her or his company.
 4. Find the company that has the smallest payroll.
 5. Give all employees of “First Bank Corporation” a 10 percent raise.
 6. Give all managers of “First Bank Corporation” a 10 percent raise.
 7. Delete all tuples in the works relation for employees of “BigBank”.
 8. Give an SQL schema definition for the employee database. Choose an appropriate domain for each attribute and an appropriate primary key for each relation schema. Include any foreign-key constraints that might be appropriate.

9. Write queries for creating employee table
10. Add department column to works relation.