# Assignment 1

Chalkiopoulos Georgios | $p3352124$

November 20, 2021

# Part 1. company Database

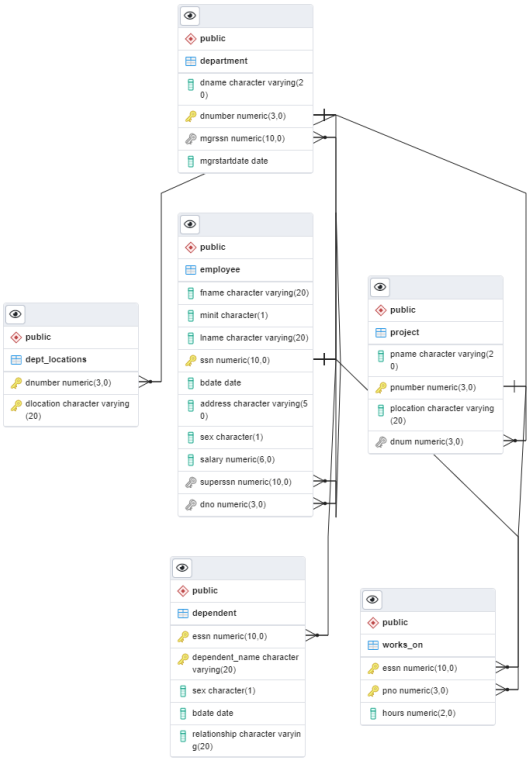


**Figure 1:** DB Schema for Company Database.

**Q1.** *For each department with max salary > 42000 get the department name as well as the number of employees.*

Using the employee table we find the distinct departments that have at least one salary greater than 42k (subquery 1). Then we join the employee table with the department, and we bring all employees with the department names, for which the department number (dno) is in the "subquery 1" we previously had.

**Q2.** *Get the names of all employees from the department that the lowest paid employee is in.*

Here we only use the Employee table, three times. First we calculate the min salary, then we select the department that the minimum salary is cointained and, finally, we bring all employees (fname, lname) that are in this department

**Q3.** *Find all employees who gain at least 5000 more than the average salary of the research department*

We use a subquery to fetch all employees (employee table) whose salary is greater than the salary calculated in the subquery. The subquery finds the average salary (added by 5K) of the research department).

**Q4.** *Create a view with the name of the department, the name of the director and the salary of the director*

The mgrssn gives the director of each department. We simply join the department and employee tables, to bring the directors, and add the selection to a view called Director.

**Q5.** *Create a view with the name of the department, the name and surname of the director the number of employees in the department and the number of project that are being controlled by it*

For this view (named question5) we selected the department table, joined with the employee table, to get the director name, joined with a table (e) that calculated the number of employees per department and, finally, joined with a table (p) that calculated the number of projects per department.

**Q6.** *Create a view with the name of the project, the name of the department, the number of employees the number of males employees, the number of female employees the total number of hours in the project*

First we need to find the number of males and females per project. There are the temp tables "male", "female". The by using the project folder we do the following:

- select the project table.
- join with the department table to get the department name.

- join with the male and female tables, calculated previously, to get the number of male and female employees. In the selection we will add the counts to get the total number.

- join the works_on table, using a subquery (w) that will fetch the total_hours per project.

All of this is added to the question6 view.

**Q7.** *Find the top 3 employees who have worked the most in the company project's, the name of the departments they belong and group the total number of hours per project*

There are two steps to solve this question. First we use a subquery (t) that uses the works_on table, joined with the employee and groups the number of hours each employee worked. and we pick the top 3 employees. Then we join the works_on, department and project tables to get all projects each employee worked on.

# Part 2. university Database

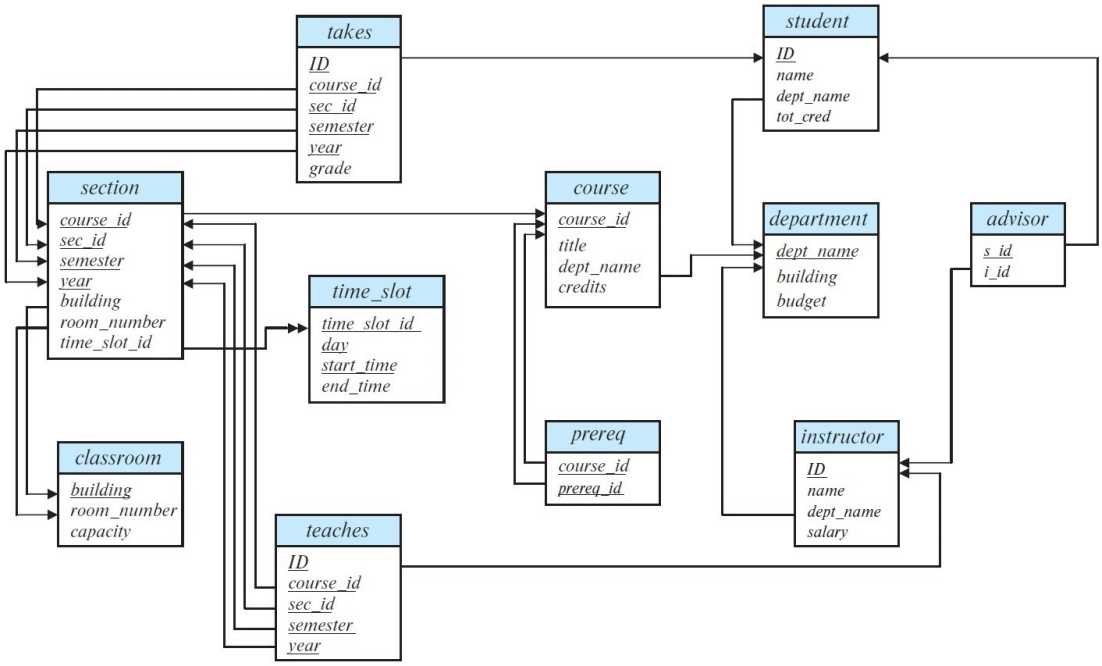


**Figure 2:** DB Schema for University Database.

**Q8.** *Order instrctors in ascending order based on the grade they have given to students. For each instructor add the total grades, A, B, C. Each line will consist of the Name of the instructor, the sum number of records for each grade.*

We will use the table takes, which has the grades per student, and join that with teached (to see which was the teacher that gave the grade, and also the inctructor table, that has the name of the instructor. We grouped grades from A-F calculating the plus or minus as the same grade (ex A+ is still A). We used the expression:

```
SUM(CASE WHEN grade like 'A%' then 1 else 0 END)
```

for each grade, which will count the number we need. Then we ordered each column from A-F in ascending order.

**Q9.** *Create a view with the Buuildings, the rooms (for each building) the total number of students and the total number of hours per semester*
*building*||*room_number*||*semester*||*tot_hours*||*tot_student_no*

For this question we used the classroom table, which contains the building and the room number. Then we join the section table, to get to the time_slot and takes tables. Regarding the takes table, we use a subquery (t1) to get the total number of students per course, semester, year. Finally, we take the differense of the starttime and endtime to calculate the duration and use all of this in a view named buildings.

**Q10.** *Create a view that will print the weekly schedule. The schedule will have for each semester/year each week of the day, along with the courses that are being tought that day (time, room, instructor) ordered by course start time*

For this question we have done the following:

- first we created the i1 table which gives a rank to the days of the week and the semesters, and also adds a zero in front of single digit hours and minutes (just to look pretty).
- We then combine the section, time_slot and course tables in order to get all the columns we need. We name this view, schedule.
- Finally we select everything from the schedule and we have added a filter (where statement) in case we only want to see information for one single department.