

Today: SQL. §§ 6.0–6.3.

Next class: Relational Design Theory. §§ 3.1–3.3.

Reminders: Homework. Newsgroup. Syllabus. Reading.

1. List the members of your group below. Underline your name.

2. Consider a database with relations `Students(id, name, year)`, `Courses(id, title, ta)`, and `Enrolls(student, course, credits)`. A tuple $(i, n, y) \in \text{Students}$ denotes a student with student-identifier i , name n , and year y . A tuple $(i, t, a) \in \text{Courses}$ denotes a course with course-identifier i , title t , and whose teaching assistant's student-identifier is a . A tuple $(s, c, r) \in \text{Enrolls}$ denotes the enrollment of the student with identifier s in the class with identifier c , for r credits.

We say student t is a TA of student s , for r credits, if s is enrolled for r credits in a course whose TA is t . We say a TA t is responsible for r credits if r is the sum of credits of all student enrollments in all courses whose TA is t .

Write a SQL statement to create a view that provides the names and IDs of the TAs who are the TAs of the maximum number of students for r credits, for each distinct value of r occurring in the database.

3. If your query for Question 2 uses subqueries in a *where* clause then provide an equivalent query that does not, else provide an equivalent query that uses such a subquery.