Tutorial 2 exercises

The exercises are for the following database (defined by schema):

Students(first_name,last_name,birthday,s_id)
Lecturers(first_name,last_name,birthday,l_id)
Courses(code,name, department) - For overall properties that does not depend on year
CoursesInYear(code,year,l_id) - For year depended properties
Enrolment(code,s_id,grade,year) - for simplicity we only got 1 grade per course

PART 1: Simple queries (1 table + aggregates):

- Find your average grade (you can use s_id=1)
- How many years have COMP207 been taught?
- How many students are in COMP207 this year?

Simple joins:

- What lecturers have taught you? – sort by their birthday
- What are the names of the courses you have attended? Sort by name
- What are the names of the courses I, Rasmus Ibsen-Jensen, have taught? (my name is unique…) - sort by code

Unions:

- What names (of students, lecturers or courses) exists?
- List all lecturers and students, ordered by their first and last name

How would you do the above simple queries for each student or each course, depending (instead of you and COMP207) – If you got the point of GROUP BY, this is hopefully reasonably clear and easy to do, otherwise you will hopefully get it from the answer… How about the simple joins?

(if you have time): Try to the simple joins and unions in relational algebra,

> Union is done by writing U between two queries in relational algebra

PART 2: More complex queries (I imagine these will take too long for many, but still, it seems like a good exercise)

- Find the students you have shared a course with
- Find the lecturers who has *not* taught you
    - You can use your previous query for who taught you but maybe not the way you would immediately think of
- Find the staff-student ratio (i.e. how many students per staff) in the computer science department this year
    - for simplicity a lecturer/student is in computer science if they have taught/studied a course there this year
    - Note: You have to be a bit careful here, since it is easy to end up counting the number of enrolments instead of the number of students…
    - Maybe use some views for this and the latter queries…
- What day(s) in the year are the most common birthdays for students?
    - You can use DAYOFMONTH(birthday) to get the day of month of birthday and MONTH(birthday) to get the month and CURRENT_DATE to get today
    - Try to figure out why I do not mention DAYOFYEAR – it does exist (hint: 1st of March is a different day in the year for 2021 as compared to 2020)
- Sort all students so the first is the one with the soonest birthday – you can sort the ones with the same birth day by name (i.e. if you had your birthday yesterday you should be last, if you have it today you should be first, ignoring others with the same birthday)