Lab 3: Subqueries in the WHERE clause

Using the University Admissions database write the following queries.

1. The ID and names of the students who have applied for computer science degree in some university. (Hint: Use the **in** construct on a subquery in the **where** clause that generates the IDs of all students that applied for computer science.)

2. The ID and names of the students who have applied for computer science degree in some university, but this time without using a subquery in the where clause.

3. The names of the students who have applied for computer science degree in some university without using a subquery in the where clause. (Warning: Be very careful using the **distinct** keyword. What would happen if 2 students with different IDs have the same name?)

4. The IDs and names of students that have applied for computer science but have not applied for electronic engineering. (Hint: Use subqueries in the where clause and the **in** and **not in** constructs.)

5. A list of all the universities that have another university in the same city. (Hint: Generate a list of universities that are in the same city and check that the list/set is not empty. Make sure you do not compare the university against itself!)

6. Without using the max operator, find the student with the highest score. (Hint: Use **not exists** in the subquery in the **where** clause.)

7. Without using the max operator, find the student with the highest score using the **all** operator (>= all (T), tests that the element is greater than or equal to all the elements in T).

8. The ID and names of all the students not from the smallest school. (The **any** construct is a companion to **all**, any (T) tests if any element satisfies a condition.)

Aside: Some SQL systems do not support **any** or **all** constructs. The same query can be written using **exists** and **not exists**.