Lab 2. SQL Queries

- assigned: week 3; due: week 6

On the relational structure created for the first lab, write SQL statements that:

- insert data - for at least 4 tables; at least one statement should violate referential integrity constraints;
- update data - for at least 3 tables;
- delete data - for at least 2 tables.

In the UPDATE / DELETE statements, use at least once: {AND, OR, NOT}, $\{<, \leq, =, >, \geq, \neq\}$, IS [NOT] NULL, IN, BETWEEN, LIKE.

On the same database, write the following SELECT queries:

a. 2 queries with the union operation; use UNION [ALL] and OR;

b. 2 queries with the intersection operation; use INTERSECT and IN;

c. 2 queries with the difference operation; use EXCEPT and NOT IN;

d. 4 queries with INNER JOIN, LEFT JOIN, RIGHT JOIN, and FULL JOIN (one query per operator); one query will join at least 3 tables, while another one will join at least two *many-to-many* relationships;

e. 2 queries using the IN operator to introduce a subquery in the WHERE clause; in at least one query, the subquery should include a subquery in its own WHERE clause;

f. 2 queries using the EXISTS operator to introduce a subquery in the WHERE clause;

g. 2 queries with a subquery in the FROM clause;

h. 4 queries with the GROUP BY clause, 3 of which also contain the HAVING clause; 2 of the latter will also have a subquery in the HAVING clause; use the aggregation operators: COUNT, SUM, AVG, MIN, MAX;

i. 4 queries using ANY and ALL to introduce a subquery in the WHERE clause; rewrite 2 of them with aggregation operators, and the other 2 with [NOT] IN.

You must use:

- arithmetic expressions in the SELECT clause in at least 3 queries;
- conditions with AND, OR, NOT, and parentheses in the WHERE clause in at least 3 queries;
- DISTINCT in at least 3 queries, ORDER BY in at least 2 queries, and TOP in at least 2 queries.

Obs.

You can use views in at most 3 queries.

You can change the relational structure created for the first lab.

Your queries must be relevant to the problem domain.