SQL Exam No. 3 09.05.2024

For each of the following tasks, write appropriate SQL commands. We expect solutions in the form of one file containing **the content** of SQL commands, not the results of the query. Any queries with incorrect syntax will not be checked. Check your solution, for example, using `\i file.sql`! You can send the file multiple times; only the latest version will be checked. Remember that before executing a database-modifying operation, you can issue the command `BEGIN;` and then `ROLLBACK;` (if you want to revert all changes) or `COMMIT;` (if you want to save them).

Load the file `jjit.sql` into your database. Send your solutions through the form at `https://dbserv.stud.ii/`. Do this as often as possible! All data on your computers is erased after a restart. In case of any problems, be sure to contact the course instructor before restarting your computer.

The format of the first line of the solution: `-- group-firstname-lastname`, where `group` are the initials of the instructor leading your group (`jotop/mabi/plg/pwi`), e.g., `pwi-Jan-Kowalski`. The required format of the entire solution file:

```
-- group-firstname-lastname
-- Task 1
<query>

-- Task 2
<query>
...
```

You can earn a total of 15 points but the maximum includes 14.

Task 1 (2 points) Read the documentation section on materialized views 40.3. Materialized Views.

Create a materialized view:

```
skill_offer_mat_view(skill text, city text, permanent_offers int,
b2b_offers int)
```

For each skill and city, the view should store the numbers of offers that require that skill, broken down into permanent and B2B contracts, excluding pairs where both these numbers are 0.

Provide the command that refreshes the contents of the view.

The view returned by the reference query contains tuples: `SQL, Wrocław, 58, 79` and `SQL, Warsaw, 144, 146`.

Task 2 (1 point) Create a table:

```
skill_offer_table(skill text, city text, permanent_offers int,
b2b_offers int)
```

The primary key is (skill, city). For each skill and city, the table should store the numbers of offers requiring that skill, broken down into permanent and B2B contracts, excluding pairs where both these numbers are 0.

Write a query that will copy data into the skill_offer_table based on the current state of the database.

Task 3 (12 points) Write triggers that will take care of updating the table from the previous task when needed. Assume that the state of the database changes only by inserting, deleting, or updating tuples describing the required skills (table skill), where modifications do not change the offer_id attribute.

Write queries to test your solution.