

Databases EX2: Relational Algebra & SQL

Due date: November 15, 23:55

Exercise Submission

There are two submission links for this exercise. One is for the relational algebra questions and one for the SQL. Please make sure you use the appropriate link for each part. Read the following instructions carefully so as to submit each part of the exercise correctly.

Relational Algebra Submission

The relational algebra part should be submitted as a PDF. In the PDF, include the Id numbers and usernames of the students submitting. *When working in pairs, the relational algebra part should be submitted by one member of the pair.*

SQL Submission

Submit your SQL answers as a zip file containing a `.sql` file for each question. The name of the `.sql` files are specified in the questions. Note that on the course homepage you are provided with a schema for testing your SQL queries.

In addition, your zip file should contain a README, as in Ex1. Please run your code one more time just before submission to check that it works on a lab computer! *When working in pairs, the SQL part should be submitted by both member of the pairs.*

1 Relational Algebra Queries

You are given the following relations from an online shopping website.

```
Customer (cid, cname, rating, budget)
Item (iid, iname, itype, price)
Purchase (cid, iid, pdate, quantity)
```

where

- cid is the id of a customer
- cname is the name of a customer
- iid is the id of an item
- iname is the name of an item
- pdate is the date of the purchase; you can assume that this attribute is a string in the following format: 2018-10-14
- quantity is at least 1

Write the following queries in relational algebra.

1. Return the ids of all items that were purchased on 'Black Friday' of 2017 (2017-11-24).
2. Return names of customers who have purchased at least 2 smartphones (an item of itype 'smartphone') in a single order.
3. Return the names of all customers who have purchased both a laptop and a smartphone.
4. Return all pairs of cid and iid such that the customer did not purchase the item.

5. Return the names of customers whose budget is at least 7,000 and who have also purchased all items that cost at most 300.
6. Return the iids of items with price more than 400, which were bought by precisely 3 customers (no less, no more).

2 SQL Queries

In your solution to the following queries you should:

- Use SELECT DISTINCT for all queries to ensure no duplicates in the result.
- Return results sorted in ascending order, by using ORDER BY x ASC, where x is the name(s) of the output column(s).

Test your queries using the `create.sql` file available on the course homepage. Note that the Serial datatype is an automatically increasing number. Thus, when inserting tuples into the tables (to test your queries) you do not need to specify a value for this field. The first row inserted will automatically get the value 1, the second will get the value 2, and so forth. Note also that character varying is a different syntax for the varchar datatype. They are identical.

Write the following queries in SQL. The name of the file in which the solution should be written appears at the beginning of each question. (Note that the first six questions are the same as in the previous section, but are repeated for your convenience.)

1. (`q1.sql`) Return the ids of all items that were purchased on ‘Black Friday’ of 2017 (2017-11-24).
2. (`q2.sql`) Return names of Customers who have purchased at least 2 smartphones (an item of itype ‘smartphone’) in a single order.
3. (`q3.sql`) Return the names of all customers who have purchased both a laptop and smartphone.
4. (`q4.sql`) Return all pairs of cid and iid such that the customer did not purchase the item.
5. (`q5.sql`) Return the names of customers whose budget is at least 7,000 and who have also purchased all items that cost at most 300.
6. (`q6.sql`) Return the iids of items with price more than 400, which were bought by precisely 3 customers (no less, no more).
7. (`q7.sql`) For each item itype, return the itype, average price, minimum price and maximum price. Order the results ascending according to itype.
8. (`q8.sql`) Return the item iid with the maximal average of budgets of the customers who bought it. (If there are several suitable items return all those items.)
9. (`q9.sql`) Update the table Item by dividing by 2 the price of all items that less than 3 customers bought.
10. (`q10.sql`) Delete all customers that did not buy any item costing more than 500.