CSC 3300

Homework #1, Spring 2021

Relational Algebra

**SKILLS YOU WILL LEARN:**

* Writing relational algebra expressions

**DESCRIPTION**

To complete this exercise, you will use the RelaX relational algebra calculator ([link)](http://dbis-uibk.github.io/relax/calc.htm). For all the questions, you will use the database named:

**UIBK – PS DATABASE SYSTEMS – EXERCISE SHEET 5 (PIZZA)**

For each of the questions below, you will provide an answer by

* Entering the corresponding relational algebra into RelaX and executing it.
* Taking a screenshot of the expression and result
* Naming the screenshot according to instructions on each question

In addition, you will save all the screenshots into a directory called homework1. This directory will be compressed into a zip file and uploaded to the Homework 1 submission folder on iLearn.

**For these questions, you will NOT use any aggregate functions, order by, group by, division, left outer join, right outer join, full outer join, left or right semi-join.** If you use any of these operators you will either lose points or receive zero points for the question. In addition, answer each questions as if you had no knowledge of what data is stored in the tables, although you CAN use knowledge obtained by investigating the databases structure. For example, answering

𝝅𝒑𝒓𝒊𝒄𝒆(𝝈𝒑𝒓𝒊𝒄𝒆=𝟏𝟐(𝑺𝒆𝒓𝒗𝒆𝒔))

would solve problem 15 but would result in zero points for the solution.

Make sure that you follow the directions below and use the filenames (all lower case) as instructed. You can use the screenshot program under the Accessories desktop menu of the Linux VM to produce the screenshot files. **Take a screenshot of the RelaX window only, do not take a screenshot of the entire Linux screen. Make sure that you capture all the pertinent output in your screenshot, i.e. the entire query and all of the results**. You can zoom out to get the whole window (using the Ctrl-minus key combination in Firefox on your VM). Regardless, make sure your screenshots are readable. Not following these directions will result in points being deducted from your grade.

Problems:

1. (**4 points**) For this problem, you will write two relational algebra queries that each show all information in the eats table. For the first query, use the projection operator. For the second query, do **NOT** use the projection operator. Include a screenshot for each query.

a. Screenshot file names: *problem1a.png, problem1b.png*

1. **2 points**) Write a relational algebra query that, when executed, results in the names of all adult people (18 years and above) that frequent at least one pizzeria. You may use any of the following operators, though not all are necessary: *selection, projection, natural join, difference*

a. Screenshot file name: *problem2.png*

1. (**2 points**) Write a relational algebra query that, when executed, results in the names of all children (17 years or below) that do not visit Pizza Hut pizzerias. You may use any of the following operators, though not all are necessary: *selection, projection, natural join, difference, union.* To test your answer, use the following two unions that contain inline relations in your expression:

𝑷𝒆𝒓𝒔𝒐𝒏  {𝒏𝒂𝒎𝒆: 𝒔𝒕𝒓𝒊𝒏𝒈,𝒂𝒈𝒆: 𝒏𝒖𝒎𝒃𝒆𝒓,𝒈𝒆𝒏𝒅𝒆𝒓: 𝒔𝒕𝒓𝒊𝒏𝒈

|  |  |  |
| --- | --- | --- |
| and | ′𝑱𝒐𝒆′,𝟏𝟑, | ′𝑴𝒂𝒍𝒆′} |

𝑭𝒓𝒆𝒒𝒖𝒆𝒏𝒕𝒔  {𝒏𝒂𝒎𝒆:𝒔𝒕𝒓𝒊𝒏𝒈,𝒑𝒊𝒛𝒛𝒂:𝒔𝒕𝒓𝒊𝒏𝒈 ′𝑱𝒐𝒆′,′𝑫𝒐𝒎𝒊𝒏𝒐𝒔′}

a. Screenshot file name: *problem3.png*

1. (**2 points**). Execute the following relational algebra query:

𝝈𝒑𝒓𝒊𝒄𝒆>𝟏𝟎 (𝝈𝒑𝒊𝒛𝒛𝒂=′𝒑𝒆𝒑𝒑𝒆𝒓𝒐𝒏𝒊′ (𝝅𝒑𝒊𝒛𝒛𝒂(𝑺𝒆𝒓𝒗𝒆𝒔)))

This query generates an error. You should be able to figure out what the query is supposed to do. Fix the query by using the comment characters (/\* and \*/). In other words, put in comments above your relational algebra expression what you would expect this query *should* return if it were correct.

a. Screenshot file name: *problem4.png*

1. (**4 points**) For this question, you will write four relational algebra expressions, one for each table, that will result in four screenshots. Each expression should show all the values for the primary key for the particular table. Logical assumptions are
   * Peoples’ names are unique
   * A person may eat more than one kind of pizza
   * Pizzerias are unique
   * A pizzeria may serve more than one type of pizza
   * Pizzerias will not charge multiple prices for the same type of pizza

Do not allow any table to have duplicate rows.

a. Screenshot file names: *problem5\_person.png, problem5\_frequents.png, problem5\_eats.png, problem5\_serves.png*

1. **2 points**) You will create a new relation with the following expression in order to finish this problem: pizzeria={ pizzeria:string , address:string ‘Pizza Hut’ , ‘1000 N. Street’

‘Little Caesars’ , ‘1000 S. Street’

‘Dominos’ , ‘1000 E. Street’

‘Straw Hat’ , ‘1000 W. Street’

‘New York Pizza’ , ‘Sydney’

‘Chicago Pizza’ , ‘New York’

}

With the new relation above, this database has three foreign keys based on the assumptions of the primary keys in question 5. Write three relational algebra expressions, using three different screenshots that show the values of the foreign keys. Include a comment that describes what primary key each query references.

* 1. Screenshot file names: *problem6\_foreign1.png, problem6\_foreign2.png, problem6\_foreign3.png*

1. (**2 points**) Write a relational algebra expression that results in the names for all the people that may, possibly, eat pepperoni people at Pizza Hut. You may use any of the following operators, though not all are necessary: *selection, projection, natural join, difference.*
   1. Screenshot file name: *problem7.png*
2. (**2 points**) Write a relational algebra expression that results in the pizzerias that may, possibly, be eaten by females over the age of 20. You may use any of the following operators, though not all are necessary: *selection, projection, natural join, difference.*
   1. Screenshot file name: *problem8.png*
3. (**2 points**) Write a relational algebra expression that results in the names for all the people that eat at a pizzeria that serves a pizza that costs between seven and ten dollars, inclusive. You may use any of the following operators, though not all are necessary: *selection, projection, natural join, difference.*
   1. Screenshot file name: *problem9.png*
4. (**2 points**) Write a relational algebra expression that results in the names for all males over the age of 20 that do NOT eat cheese pizza. You may use any of the following operators, though not all are necessary: *selection, projection, natural join, difference.*
   1. Screenshot file name: *problem10.png*
5. (**2 points**) Write a relational algebra expression that results in, for each teenager, the name of the teenager, the pizzeria the teenager visits, the pizza the teenager eats at the pizzeria and the pizza’s price. You may use any of the following operators, though not all are necessary:

*selection, projection, natural join, difference.*

* 1. Screenshot file name: *problem11.png*

1. (**2 points**) Write a relational algebra expression that results in all of the people that frequent Little Caesars that eat pizza that Ian does NOT eat. You may use any of the following operators, though not all are necessary: *selection, projection, natural join, difference.*
   1. Screenshot file name: *problem12.png*
2. **2 points**) Write a relational algebra expression that results in the names for all the people that eat cheese pizza at a pizzeria that does not serve sausage pizza. You may use any of the following operators, though not all are necessary: *selection, projection, natural join, difference.*

a. Screenshot file name: *problem13.png*

1. (**4 points**) Write a relational algebra expression that results in all pizzerias that serve a pizza that is more expensive as the same type of pizza served by another pizzeria. For example, Little Caesars and New York Pizza both sell pepperoni pizza, but the pepperoni pizza is more expensive at Little Caesars. So the results list would have a row that lists Little Caesars in the first column, the price of the Little Caesars pepperoni pizza in the second column, New York Pizza in the third column, the price of the New York Pizza pepperoni pizza in the fourth column, and finally the type of pizza in the final column. You may use any of the following operators, though not all are necessary: *rename, cartesian product, selection, projection, natural join, difference.*

a. Screenshot file name: *problem14.png*

1. (**3.4 points extra credit – no partial credit**) Note: Your grade for this assignment can not exceed 100%. If you have answered every other question correctly, and also answer this question correctly, you will still only earn a 100%.

Write a relational algebra expression that results in the age of the oldest female person. You may ONLY use any of the following operators, though not all are necessary: *rename, cartesian product, selection, projection, intersection, difference.*

a. Screenshot file name: *problem15.png*

**SUBMISSION:**

Once you are finished with all the questions, create a zip of your directory of files into a single compressed archive and submit the file to iLearn under Homework 1.