Questions to be discussed: 1, 2, 3(a)-(c)

1. This question considers a binary relational algebra operator called the **division operator** denoted by $/$[1].

   Consider two relations R and S where the set of attributes in the schema of R and S are $(A_1, \cdots, A_m, B_1, \cdots, B_n)$ and $(B_1, \cdots, B_n)$ respectively where $m \geq 1$ and $n \geq 1$. That is, the set of attributes in S is a *proper* subset of the set of attributes in R.

   Assume that the attributes that are in R but not in S are ordered as $(A_1, \cdots, A_m)$ in the schema of R and the schema of S is $(B_1, \cdots, B_n)$. Let L denote the list of attributes in the schema of R.

   The division of R by S (denoted by R/S) computes the largest set of tuples $Q \subseteq \pi_{A1,\cdots,A_m}(R)$ such that for every tuple $(a_1, \cdots, a_m) \in Q$,

   $$\pi_L(\{(a_1, \cdots, a_m)\} \times S) \subseteq R$$

   Q is also referred to as the **quotient** of R/S and its schema is $(A_1, \cdots, A_m)$. The following example illustrates R/S given two relations $R(A, B)$ and $S(B)$.

   R

A	B
a	1
a	2
b	1
c	1
c	2
c	3
d	2
d	3

   S

B
1
2

   R/S

A
a
c

   (a) Consider again Question 5 in Tutorial 1 to find the restaurants that sell all the pizzas that Maggie likes and don't sell any pizza that Ralph likes. Write a relational algebra expression for this query that uses the division and natural join operators.

   (b) Given relations $R(A, B)$ and $S(B)$, write a relational algebra expression to compute the division of R by S using only the basic relational operators (*i.e.*, σ, π, ρ, $\times$, $\cup$, $\cap$ and $-$).

---

[1] It is a "division" operation because it kind of reverses the "multiplication" operation $\times$.

2. Consider a relational database for a company that consists of the following two tables.

   - Offices(<u>office_id</u>, building, level, room_number, area)
   - Employees(<u>emp_id</u>, name, office_id, manager_id)

   The database satisfies the following constraints:

   1. Offices stores information about the office rooms in the company. Each room has a unique identifier office_id (*which is the primary key of Offices*) and information on its building name, floor level, room number and floor area.

   2. {building, level, room_number} is a *candidate key* of Offices.

   3. Employees stores information about the employees in the company.

   4. emp_id is the *primary key* of Employees.

   5. The name of each employee must be a non-NULL value.

   6. Each employee must be assigned to exactly one office identified by office_id.

   7. Each employee may be managed by at most one manager.

   8. If an employee is managed by someone, the emp_id of his/her manager is recorded in manager_id.

   9. A record in Offices cannot be removed if there's some employee assigned to that office.

   10. A manager in Employees cannot be removed if there's some other employee managed by that manager.

   11. Any modification to office_id in Offices is propagated to other database records.

   12. Any modification to emp_id in Employees is propagated to other database records.

   Write SQL statements to create the database schema with appropriate attribute domains and constraints.

3. Consider a relational database for an online shop that consists of the following five tables.

- Books(isbn, title, authors, year, edition, publisher, number_pages, price)
- Customers(cust_id, name, email)
- Carts(cust_id, isbn)
- Purchase(pid, purchase_date, cust_id)
- Purchased_items(pid, isbn)

The database satisfies the following constraints:

1. Books records information about the books available for sale in an online shop. Each book has a unique identifier isbn and information about its title, authors, publishers, publication year, edition, number of pages and selling price. The title and authors must have non-NULL values. The value of the edition must be non-NULL with one of the following values: *paperback*, *hardcover* or *ebook*. The selling price must have a positive value. If the number of pages in known, it must be a positive value.

2. Customers stores information about the shop's customers. Each customer has a unique identifier cust_id, a name and an email address. The name must have a non-NULL value.

3. Carts stores information about the books in customers' shopping carts. Each shopping cart record indicates a book that a customer is interested to purchase but hs not yet purchased.

4. When a customer decides to purchase their selected books, a new record for this purchase is recorded in the Purchase table which has the following information:

   (a) A unique identifier pid.
   (b) The date of the purchase.
   (c) The customer identifier.

   In addition, each book in the customers' shopping cart is added to the Purchased_items table and the customers' shopping cart is emptied.

(a) Write SQL statements to create a database schema with appropriate attribute domains and constraints.

(b) Suppose that the schema of Purchase is changed with purchase_date being replaced by purchase_timestamp, where each customer has at most one purchase at any timestamp. How would this change affect your answer for part (a)?

(c) For each of the following additional constraints on the database, state whether the constraint can be expressed using SQL constructs that you have learned. If it is possible, add this constraint to your database schema in part (a).

   (i) If a book is a hardcover edition, its selling price must be at least 30.

    (ii) If a book has both hardcover and paperback editions (*for the same book title and authors*), the selling price for the hardcover edition must be higher than the selling price for the paperback edition.

    (iii) If the number of pages in a book is more than 1000, the edition of the book must be an ebook or its price must be at least 100.

    (iv) All the books published by 'Acme' from 2010 onwards have only ebook edition.

(d) Consider the following additional constraints on the database:

    (i) If a customer is deleted from `Customers`, remove all the customers' records from `Carts` and `Purchase`.

    (ii) If a book is deleted from `Books`, remove all records from `Carts` that reference this book. Additionally, for each of the records in `Purchased_items` that reference this book, change its `isbn` value to the default value of 0.

    (iii) If a purchase is deleted from `Purchase`, remove all the records from `Purchased_items` that reference this purchase.

    (iv) Any modification of `cust_id` in `Customers` is propagated to other records in the database.

    (v) Any modification of `isbn` in `Books` is propagated to other records in the database.

    (vi) Any modification of `pid` in `Purchase` is propagated to other records in the database.

Add these additional constraints to your answer for part (a)