Midterm Prep: Relational Algebra and SQL

Given the following db schema:

- *Customers* (`cid`: int, `name`: string, `city`: string, `state`: string, `age`: int)
- *Visit* (`cid`: int, `mid`: int, `visitday`: date)
- *Museums* (`mid`: int, `mname`: string, `mcity`: string, `mstate`: string, `mtype`: string)

A customer is identified by the `sid`. A customer also has a `name`, a `city` a `state`, and an `age`. A museum is identified by `mid`. A museum also has a name (attr. `mname`), a city (attr. `mcity`). A state (attr. `mstate`) and a type (attr. `mtype`; e.f.: history). Customers visit museums. When a customer visits a museum, a record is inserted into the *Visit* relation, with the `cid` of that customer, the `mid` of that museum and the day of visit (attr. `visitday`).

Question 1

Using the db schema from above:

1. Write the relational algebra to find the name and city of all customers from MA.

$$\pi_{name,city}\left(\sigma_{state='MA'}Customers\right)$$

2. Write the relational algebra to find information about all museums of type history or science.

$$\sigma_{(mtype='history')\vee(mtype='science')}Museums$$

3. Write the relational algebra to find the id and name of customers and the names of museums they visit.

$$\pi_{cid,name,mname}\left(Customers \bowtie Visit \bowtie Museums\right)$$

4. Write the relational algebra to find the names of customers who visited only museums in NY state.

$$\rho\left(CustVisitNY, \pi_{cid,name}\left(Customers \bowtie Visit \bowtie \left(\sigma_{mstate='NY'}Museums\right)\right)\right)$$
$$\rho\left(CustVisitOther, \pi_{cid,name}\left(Customers \bowtie Visit \bowtie \left(\sigma_{mstate\neq'NY'}Museums\right)\right)\right)$$
$$\pi_{cid,name}\left(CustVisitNY - CustVisitOther\right)$$

5. Write the relational algebra to find the names of museums that were visited by both customers from Boston, MA and Burlington, MA.

$$\rho\left(BostonMACustMuseums, \pi_{mid,mname}\left(\left(\sigma_{(city='Boston')\wedge(state='MA')}Customers\right) \bowtie Visit \bowtie Museums\right)\right)$$
$$\rho\left(BurlingtonMACustMuseums, \pi_{mid,mname}\left(\left(\sigma_{(city='Burlington')\wedge(state='MA')}Customers\right) \bowtie Visit \bowtie Museums\right)\right)$$
$$\pi_{mname}\left(BostonMACustMuseums \cap BurlingtonMACustMuseums\right)$$

Question 2

Using the db schema from above:

1. Write the SQL statement to find how many museums we have in the database.
2. Write the SQL statement to find how many museums from Boston, MA we have in the db.
3. Write the SQL statement to find the average age of customers for each state. Show the results only for those states that have at least 5 customers.
4. Write the SQL statement to find all customers who are from MA and are either younger than 30 or older than 40.
5. Write the SQL statements to create all tables from the db schema. Do not forget about the key constraints. Write a second version of `CREATE` statement for the table Customer to only allow customers older than 5 years.
6. Find the id and the name of customers who visited a museum from MA in year 2018. Sort the result by name in ascending order.
7. Write an insert statement for each table. The statements should be written in an order such that it won't cause an error.
8. Find the id and name of customers who visited museums in 2019 and did not visit any museum in 2020.
9. Find the id and the name of customers who visited museums both in 2020 and 2022.
10. Explain the `NOT NULL` constraint on a table column. Give an example of violation.

    The `NOT NULL` constraint ensures that a column is entered whenever data is inserted into the table. An example of this is given a table with the following schema:

```
CREATE TABLE Customers (
    cid     INT PRIMARY KEY,
    name    VARCHAR(40) NOT NULL,
    city    VARCHAR(50),
    state   VARCHAR(50),
    age     INT
);
```

    The following `INSERT` statement would fail because a name isn't inserted.

```
INSERT INTO Customers (cid, city, state, age) VALUES (1, 'Boston', 'MA', 22);
```

11. Write the SQL statement to extract information of customers who visited museums of type history.
12. Write the SQL statement to find information about museums that had at least 100 visitors in year 2021.
13. Explain the difference between the following `SELECT` statements assuming that `mtype` column can contain `NULL` values.

```
SELECT COUNT(*) FROM Museums;
SELECT COUNT(mtype) FROM Museums;
```

With the first statement, it counts the total rows in the *Museums* table disregarding any `NULL` data values. The second statement would count the number of rows in the `mtype` column while disregarding any rows whose value in the column is `NULL`.

14. Write the SQL Statement to find information about all museum that have a type.
15. Write the SQL query to find the id and name of customers who visited all museums.
16. Write the SQL query that finds the id and names of the youngest customers.
17. Write the SQL query to find all customers who visited all museums from MA. Extract the id and name of the customers. Sort the result by the name in descending order.
18. Write the SQL query to find the id and name of customers who visited both history and science museums. Remove the duplicates.