

## Problem Set #1 (due 9/29)

Note: In this homework, you do not need to create tables and execute queries using an actual DBMS. A written solution is sufficient.

1. Suppose you have a database modeling an online marketplace such as Amazon or eBay, where many merchants can offer products for purchase. It is given by the following highly simplified schema:

CUSTOMER (cid, cname, ccity, cstate)  
MERCHANT(mid, mname, mcity, mstate)  
PRODUCT (pid, pname, pdescription)  
OFFERS(pid, mid, pprice)  
PURCHASE(cid, pid, mid, puprice, putime)

In this schema, merchants offer products for sale at a certain price. Many merchants can offer the same product at different prices. Whenever a customer buys a product, we need to store which product they bought, from which merchant, at what price, and at what time and date. For both customers and merchants, we also store the city and state where they are located.

(a) Identify suitable foreign keys for this schema.

(b) In the PURCHASE table, why is (cid, pid, mid, putime) the primary key? What happens if we remove *mid* from the key? What if we remove *putime*?

(c) Write statements in SQL for the following queries.

- I. Output the *cid* and *cname* of any customer who has bought an item for more than \$50.
- II. Output the *cid* and *cname* for any customer who bought an item from a merchant located in the same state that they live in.
- III. For each merchant, output the *mid* and their total sales in \$ for the year 2015.
- IV. Output the *cid* of the customer(s) who made the most expensive purchase ever.
- V. Output the *cid* of the customer(s) that made the largest number of purchases.
- VI. For the merchant with *mname* "ABC Sneakers", output the total sales in \$ for each state.
- VII. Output the *mid* and *mname* of any merchant who has sold an item in every state in which there is at least one registered customers in the database.

(d) Write expressions in Relational Algebra for the above queries.

(e) Write either DRC or TRC queries for the above queries. Or explain the reason why you think a particular query cannot be done in DRC or TRC.

2. In this problem, you have to design a relational schema for a website that models a service similar to Airbnb, where users can rent accommodations (rooms or entire apartments) from other users and can also offer accommodations to other users for rent. For each user, you need to store a unique user name, plus their real name, email, and city of residence. Users may offer a number of accommodations ("properties") for rent, where each property has an ID, a host (the user offering it), a title (e.g., "Sunny

bedroom close to Prospect Park” or “Beautiful 3BR in Crown Heights”), an address, a maximum number of people that can sleep in it, and a longer description that may also contain various house rules and conditions. (The details of the description do not matter, and you do not have to model how to store photos, which a real website would of course support.)

A user may offer several properties. For each property and for each date in the calendar, the user can choose if the property is *available for rent*, *rented out*, or *unavailable*. (A property could be *unavailable* if the user needs it for herself, or if there are other reasons such as renovations or repairs taking place.) For each date that a property is available, the user can choose what price to charge for this day. Note that the price may be different from one day to the other; however, for simplicity we assume that once the user chooses a price for a particular property and date, this cannot be changed.

A reservation for a property can be for any contiguous range of dates when the property is still available. Once a reservation is done, the property becomes *rented out* for those dates, and for simplicity we assume that no cancellations are possible. For each reservation, you need to store the property, the range of dates, the user renting it, and the cost of the reservation, where the cost of the reservation consists of three parts: (1) the total rental price, which is simply the sum of the prices for the dates of the reservation, (2) a one-time cleaning fee for each reservation that depends on the property but not on the date or duration, and (3) an agency fee of 10% of the total rental price that goes to the web site offering the service (e.g., Airbnb). Finally, users who have rented a property can write a review and give a rating from zero to five stars for the property.

(a) Design a relational database schema for this application that supports the above functionality. Specify all primary and foreign key constraints, and state any assumptions you are making. You can decide which exact attributes make sense for this schema.

(b) Write SQL statements for the following queries. If your schema does not support these, you need to modify it appropriately. (For this first homework, you may use informal expressions such as `year(ts) = '2016'`, where `ts` is a timestamp, to check if the year is 2016.)

- I. Suppose a user is looking for a property in Chicago to rent from April 27 to May 6, 2017. Write a query to output the property ID, title, and total cost for any property that is fully available during these dates.
- II. For each property, output the property ID and the total utilization during 2015, defined as the ratio of the number of days it was rented out, and the number of days it was either available or rented out.
- III. For each property that has received at least 5 ratings, output the property ID and the average rating it has received.
- IV. For each city, output the name of the city and the total amount of fees the web site has earned from reservations for properties in that city.