INF1343, Winter 2012, Week 3 Exercises

Exercise 1

"Eatr" is a small company that wants to launch a website that would allow people to get information about restaurants in Toronto. Eatr has already done some preliminary user experience research and decided that the website will offer the following features:

1. A visitor to the site, "Alice," should be able search for restaurants that match some combination of:

    - The name of the restaurant or a part of a name (e.g., "Maroli").
    - Neighborhood ("the Annex" or "Kengsington Market").
    - Tags added by users (e.g., "pizza", "uighur", "andhra", "russian", "burger," "vegetarian").

    The search should result in a list of restaurants with a name of each restaurant and its average rating, sorted by rating.

2. Alice should be able to click on a restaurant from the list and see the following information about it:

    - The name of the restaurant.
    - Associated tags (see above).
    - The address and phone number.
    - Price range (from "$" to "$$$$$").
    - The average of visitor ratings.
    - A list of comments from visitors, which would include a rating, from one to five stars.

3. Alice should be able to create lists of restaurants, assign restaurants to those lists, and later see restaurants that she added to a particular list. For example, she might have a list called "Cool new places to check out," "Fancy places to go with parents," and "Late night places." Alice should be able to remove restaurants from her lists and to rename the lists.

4. Users should be able to rate restaurants or leave comments. Additionally, when looking at a comment by "Bob", Alice should be able to click on Bob's name to see Bob's profile and the list of all Bob's comments and ratings. The list should show which restaurant each comment or rating refers to.

5. Comments on restaurants should be "threaded." That is, Alice should be able to post a comment *in response* to a comment by Bob. The web interface should somehow show which comments are in response to other comments.

6. Alice should be able to mark Bob's comment as "helpful" or "unhelpful." When applicable, comments would be shown with a note saying something like "24 out of 32 people found this helpful."

The database does not need to support any features beyond those outlined above.

Draw an ER diagram for Eatr's database. The first version of your diagram should *use many-to-many relationships* where appropriate and do *not* use associative entities. Make sure your diagram shows cardinality and that all relationships are labeled. Once you are done with the first version, modify your ER diagram to get rid of all many-to-many relationships, replacing them with associative entities.

Exercise 2

Madame Z runs a fortune-telling business, employing a number of fortune tellers. The business has many customers, some of whom come back on a regular basis. Before a visit customers can brows the list of fortune-tellers by name or method used (tarot, astrology, etc.) and can schedule a session with a specific fortune-teller. Repeated customers are usually assigned to one person as their "primary" fortune-teller. When customers come in for a session, a number of predictions can be made, which are saved for future reference. (This ensures that this information is available when making predictions in later sessions.) The customer is billed for the services provided during the session, which can include multiple billable items. They can either receive an invoice after the completion of the session to be paid immediately, or have an invoice sent to them by mail at the end of the month, in which case the invoice would include all services for that month. Customers who receive an invoice by mail can pay it by mailing a check. The payments are recorded when they are received.

Madame Z wants to start using a database for tracking the relevant information, so she hired a consultant who produced the ER shown below:
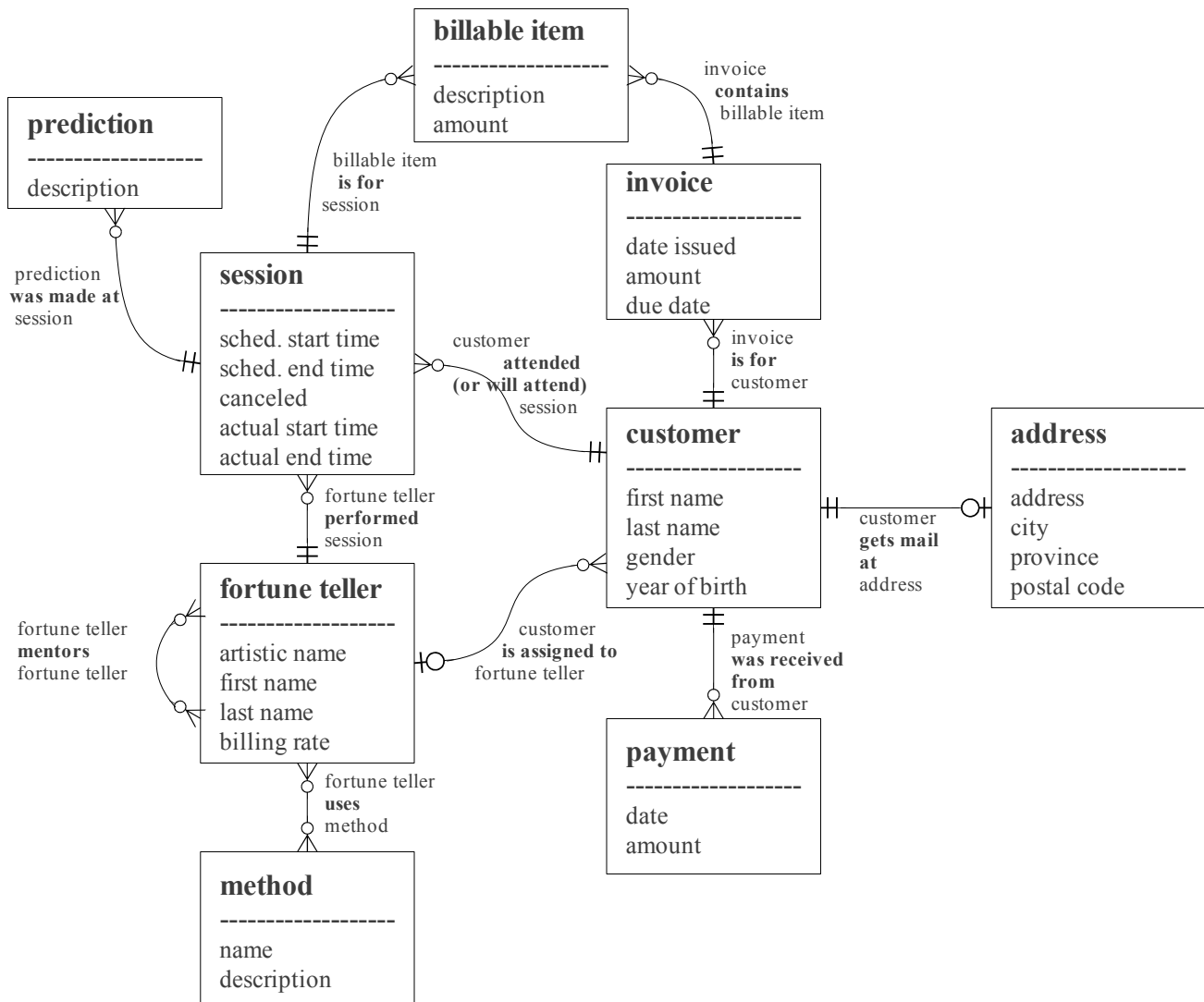

Figure 1. An ER diagram for Madame Z's Fortune-Telling Center.

Answer the following questions using the diagram.

a) Does the database allow payments to **not** be associated with a customer? Can a single payment pay for two customers' invoices?

b) Can a fortune teller have multiple mentors? Can they have no mentors at all? Can a fortune teller be listed as his / her own mentor?

c) Can a session have no predictions associated with it?

d) Can a session involve two customers? What about two fortune tellers?

e) Can a customer have two addresses on file? Can they have no address?

f) Looking at a particular prediction, is it possible to know who made it, what client it was for, and what method of fortune-telling was used for it? Explain.

g) Is it possible to identify to which session a particular payment applies? Explain.

h) Invoices are issued at the end of each month. When preparing an invoice for a particular customer, how would the system determine the amount to bill? Would it be possible to send customer an invoice specifying to what session each item corresponds?

i) Madame Z pays her fortune tellers based on how much money each of them brought in. Will she be able to figure out how much of her revenue comes from a particular fortune teller? Suppose that Madame Z wants to only pay fortune tellers when the customers whom they served actually pay their bills. Will she be able to do that?

j) Does the diagram contain any relations that would need to be broken-up with associative entities? If so, which ones? Explain.

Please make sure to answer the questions *based on the diagram*. E.g., for question (a), what is asked is not whether it is a good idea to allow payments to not be associated with the customer, but rather whether this will be possible *according to the diagram.*