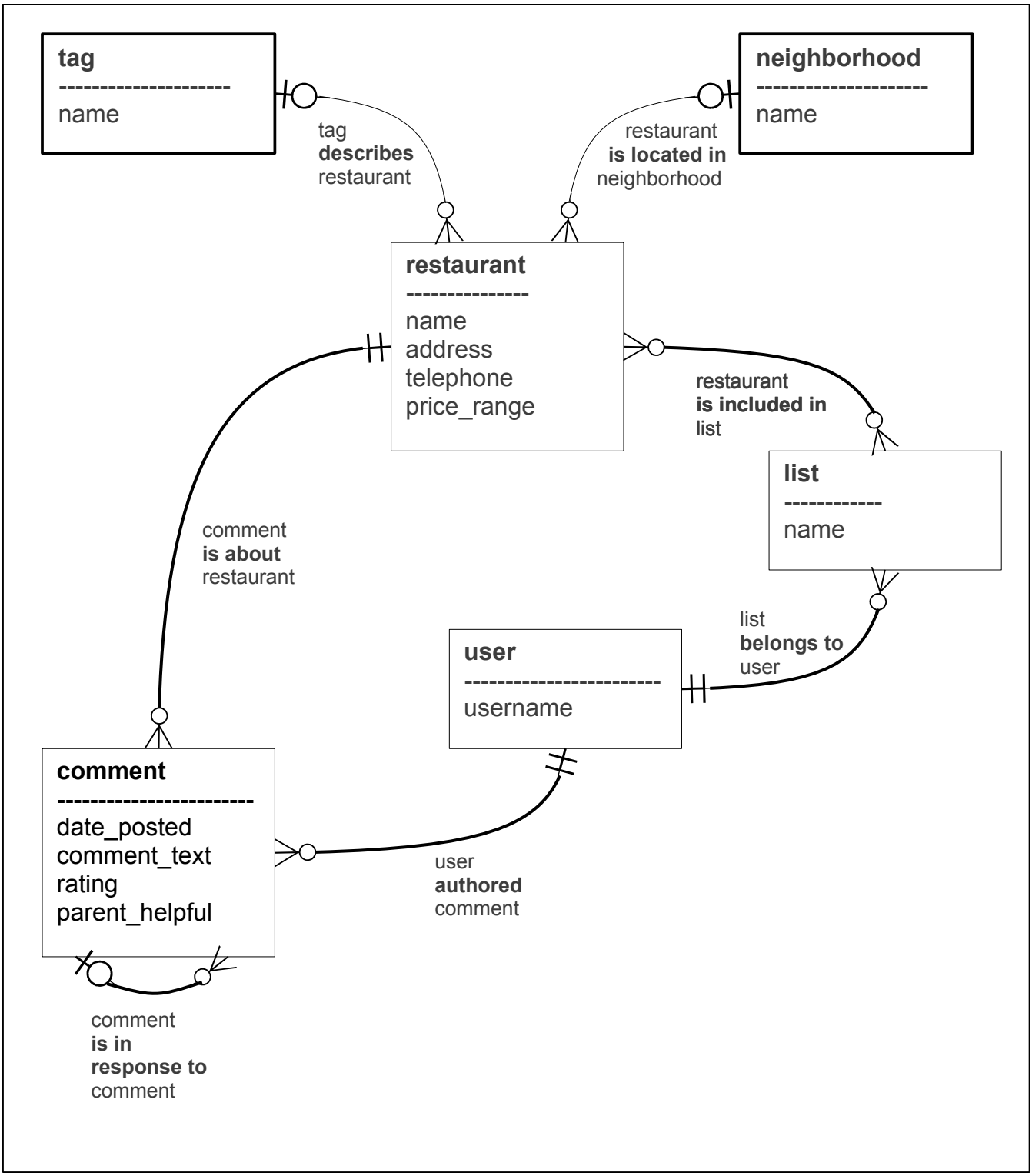
INF1343, Winter 2012, Answers to Week 3 Exercises

Exercise 1

An ER diagram with one M:M relationship:



Notes:

- In the first analysis, neighborhood and tags could be handled as attributes. Eventually, however, we would need to turn them into entities, so we might as well do this upfront.
- Ratings and comments can be handled with a single entity, since they act quite similarly to each other. So, this diagram has one entity that is called "comment" but can also be used for storing rating, since it has an attribute for this. The same entity can also be used for rating comments as helpful/unhelpful but adding a boolean attribute parent_helpful.

Before we can implement this, however, we will need to break up the one M:M entity. We do this by inserting an associative entity "list_membership" between "restaurant" and "list".

Exercise 2

a) Does the database allow payments to **not** be associated with a customer?
— **No**, an invoice *must* be associated with a customer.
Can a single payment pay for two customers' invoices?
— **Yes**, a single payment can potentially apply to multiple invoices. Or, to be more precise, payments are not associated with invoices directly. Madame Z just records invoices for a customer and payments for this customer. Presumably, she knows that the customer owes her money if the total amount of payments is less than the total amount of invoices.

b) Can a fortune teller have multiple mentors?
— **Yes**, a fortune teller can have multiple mentors.
Can they have no mentors at all?
— **Yes**, a fortune teller can have no mentor at all.
Can a fortune teller be listed as his / her own mentor?
— There is nothing in this diagram that would suggest that a fortune teller cannot be marked as his / her own mentor. However, note that this is simply because there is no way to express such a prohibition.

c) Can a session have no predictions associated with it?
— **Yes**. (But notice that a prediction **must** be associated with a session.)

d) Can a session involve two customers?
— **No**, a session must be associated with one and only one customer.
What about two fortune tellers?
— **No**, a session must be associated with one and only one fortune teller.

e) Can a customer have two addresses on file?
— **No**, they can have at most one address.
Can they have no address?
— **Yes**, a customer can be without an 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.
— **Yes, yes, no.** We can always determine who made a prediction and who it was for since **prediction** is always associated with a specific **session**, which in turn is always associated with a specific **teller** and **client**. We cannot be certain about what method was used, however, since the relationship between **teller** and **method** is many-to-many.

g) Is it possible to identify to which session a particular payment applies? Explain.
— **No.** When looking at a **payment** we can determine which **customer** it was for, but **customer** can be associated with multiple instances of **session**. In fact, we can't even know if a payment is for a particular *set* of sessions. Judging by the ER diagram, customers just make payments whenever they feel

like it and Madam Z just credits their accounts for the amounts that they paid.

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?
— An **invoice** entity is associated with multiple instances of **billable_item**, which can be used to record specific things for which the customer is being billed on an invoice. So, when an actual invoice is generated for the customer, this invoice could include a list of specific things the customer is billed for, as well as the total, which could be calculated at this point. The specific items can be listed together with the corresponding sessions, since each **billable_item** is associated with a **session**. (Note that we can have bill for multiple sessions with one invoice and we can have multiple items for each session.) One thing to note, though: **billable_item** has a mandatory relationship with **invoice**. This means an instance of **billable_item** cannot be created before there is an instance of **invoice** that it would be tied to. This does not represent a practical problem, however. All this means is that a system would create an empty invoice after the customer's first visit during a particular month, then fill that invoice with billable items over the course of the month. When it's time to issue an invoice, the system will just find all the billable items associated with the invoice and all the sessions associated with those items.

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?
— **Sort of.** All charges are associated with specific sessions and each session is associated with a particular fortune teller. So, Madame Z can figure out what *charges* were brought by each fortune teller. For the *revenue*, though, see the next question.
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?
— **Not easily.** Payments are not associated with specific invoices or charges. So, if a customer pays half of their bill, it is impossible to determine which specific charges are unpaid.

j) Does the diagram contain any relations that would need to be broken-up with associative entities? If so, which ones? Explain.
— The relationship between **fortune_teller** and **method** will need to be broken up, since it is M:M. Same for the recursive relationship **fortune_teller**.