NYU Tandon School of Engineering
Computer Science and Engineering
CS-UY 3083, Introduction to Database Systems, Fall 2018 Prof Frankl

HOMEWORK #1

Instructions: You may work alone or with a group of up to 5 people. Hand in your solutions via Gradescope as a single pdf file with ONE ANSWER PER PAGE. Follow the Gradescope instructions to mark your solution to each problem or subproblem as indicated in the outline; otherwise the grades will not be able to find them. If you're working with a group, use GradeScope's group submission feature to indicate all members of your group.
Note: You may find it useful to use *draw.io* or another drawing tool to draw ER diagrams. Alternatively, you may draw them *neatly* by hand.

Problem 1

1. Draw an ER diagram, similar to those shown in class, with two entity sets, Person and TVseries and relationship set Watch. A Person has a unique name, composed of a first name and a last name; a class (such as Freshman, sophomore, etc); and some (zero, one, or multiple) phone numbers. A TVseries has a title, a year (the year it started), a number indicating how many seasons its been running and a number indicating the (max) number of episodes per season. The title and year, together, of a TVseries are unique; in other words it's possible for two different TVseries to have the same title or the same year, but no two TVseries have *both* the same title and the same year. A Person who Watches a TVseries can indicate how much they like it with a number of stars.

2. Consider the list of TV series posted in the Resources section on NYU Classes and consider yourself and a few other real or imaginary friends.

   - List at least three elements of the entity set Person.
   - List at least five elements of the relationship set Watches.

   This does not have to be true data ... it's OK to lie about what you or your friends watch, etc. In this list, you can identify people by their first and last name and identify movies by their title and release year.

3. Re-draw the ER diagram from part (1), adding attributes, entity sets, and/or relationship sets to model the following; indicate cardinality constraints for any relationship sets that are not many-to-many:

   - Each Person's date of birth
   - The frequency with which a person watches a TVSeries (e.g. indicating whether they watch it often or rarely, etc.)
   - actors, each of whom has a name (first name, last name), and a gender;
   - which actors appear (with regular roles) in which TVserries, along with the role they play.
   - A person's favorite actor
   - Which TVseries are spin-offs of which other TVseries.
   - Individual Episodes of a series. Each has a season number and an episode number that is unique for that particular TVseries. However episodes of different TVseries may have the same (season number, episode number). An Episode also has a synopsis.
   - Episodes have guest actors who appear in that episode, even if they're not a regular in the TVSeries.

   Draw *one* ER diagram that includes the entity sets and relationship sets from part 1 and all of the stuff listed in part 3. Where possible, use the nouns, verbs, and prepositions fromt he description above to name your entity sets and relationship sets.

Problem 2 Suppose you're designing a database for a restaurant. The restaurant manager says they want to keep track of customers who are currently seated, dishes (e.g. "hamburger", "fried chicken", etc). Customers are identified by their table number and their seat number (at the table), e.g. Table 5, Seat 2. Each dish has a unique name, a description, a category (e.g. appetizer, main dish, dessert) and a price. The database will keep track of which dishes were ordered by which customer, along with the status (e.g. "in preparation", "ready","served, finished", etc).

1. Draw an ER diagram modeling this information. It should have an entity set representing customers, an entity set representing dishes, and one relationship set.

2. While reviewing this ER diagram with you, the restaurant manager realizes that some of the dishes have different sizes with different prices (e.g., "small tomato soup for $3.00 and large tomato soup for $5.00). Modify the ER diagram to deal with this. Hint: use a weak entity set. Think about which entity sets participate in the relationship set representing orders.

3. You did such a good job on the database for "eat-in" orders, that the restaurant has hired you to design a database for their online orders. In this case, customers do not have table numbers and seat numbers, but each customer has a unique phone number and an address, composed of a building number, street name, and apartment number. In this scenario, a customer may order multiple servings of the same dish (e.g. three hamburgers). The database will keep track of how many of which dish was ordered by which customer, along with its status (e.g. "in preparation", "ready","out for delivery", etc). The database only keeps track of current orders and you may assume that a customer can't change an order once it's in the system.

4. Now the manager decides that the restaurant would also like to keep track of the date and time on which each order was made and keep historical data, so that they'll know which customers have ordered which dishes in the past. Note that adding date and time attributes to the ordered relationship set is not sufficient, as this still will not allow the a customer to order the same dish at different date/times. (Why not?). Instead, you can take one of the following approaches:

    - use a ternary relationship set, involving an additional entity set representing dates/time;
    - or, add another entity set to represent orders, with relationship sets indicating who the order is by and what item is ordered). You may represent the orders with a strong entity set, adding an OrderNumber or with a weak entity set that has *two* identifying strong entity sets.'

Problem 3 Consider the bookstore E-R in Figure 7.29 (also shown in slide 38 in the chapter 7 powerpoints). Submit one E-R diagram with the following modifications:

1. Modify the E-R diagram to indicate that a book is published by at most one publisher.

2. Modify the E-R diagram to indicate every book has at least one author.

Problem 4
Consider The E-R diagram in Fig. 7.29 (before the modifications from Problem 3). Make the following changes:

- omit the warehouse entity set and the stocks relationship set.
- omit the publisher entity set and the published_by relationship set
- Change the cardinality constraints on the basket_of relationship set to indicate that every shopping basket is associated with exactly one customer.
- Change the phone number attribute of customer to a multi-valued attribute.
- Change the name attribute of the customer entity set to a composite attribute with components first_name and last_name.

Following the rules we studied, derive the corresponding relational schema from the E-R diagram. Show your answer in the form of a schema diagram, in the style of figure 2.8. (That is, For each relation, draw a rectangle that lists the relation's attributes; underline the primary key(s); draw arrows to indicate foreign key contstraints.)

Problem 5:

- Draw an ER diagram with two entity sets Person and Homework, each with some appropriate attributes, and a relationship set WorkedTogether. Use the lower bound .. upper bound notation to specify the constraint that between 1 and 5 people should work together.
- List all the elements of WorkedTogether for your group for this assignment (HW1)