

Assignment 3 – due Friday, September 20 (hard copy due in class)

Collaboration policy: *This is an individual assignment. You are allowed to discuss the assignment with your colleagues, but your submission should reflect your own work. Sharing or copying code is not permitted. We reserve the right to divide the grade between any students who violate this policy. In addition, you must identify all those that you collaborate with on your assignment cover sheet. Consult your instructor if you need further clarification.*

1. a) Design an E/R diagram for a baseball database application described as follows:

- Teams have a TID and a name
- Players have a name and an age
- Pitchers are a type of Players. Each pitcher has attributes W (win games), L (loss games), and ERA (earned run average)
- Batters are a type of Players. Each batter has attributes AVG (batting average) and HR (home runs)
- Each player plays for exactly one team; a team can have many players
- Each player has exactly a contract relationship with his team, and a contract contains years and salary.
- Games have a date (on which the game was played) and a score (e.g., “3:2”)
- A game has exactly a winning team and has exactly a losing team; A game can be uniquely identified by its winning team, losing team, and the date.

If you feel that you must make any additional assumptions, please state them clearly in your solution. Remember to indicate the key for each entity, as well as the multiplicity of each relationship using the appropriate notation. (Same for the next exercise.)

b) Translate the E/R diagram to a relational schema. Try to minimize the number of relations your solution has, and merge relations where appropriate. Don't forget to specify the keys. In translating a subclass hierarchy, use the E/R style translation.

2. Your next task is to design a database for an online video service that offers hit TV series.

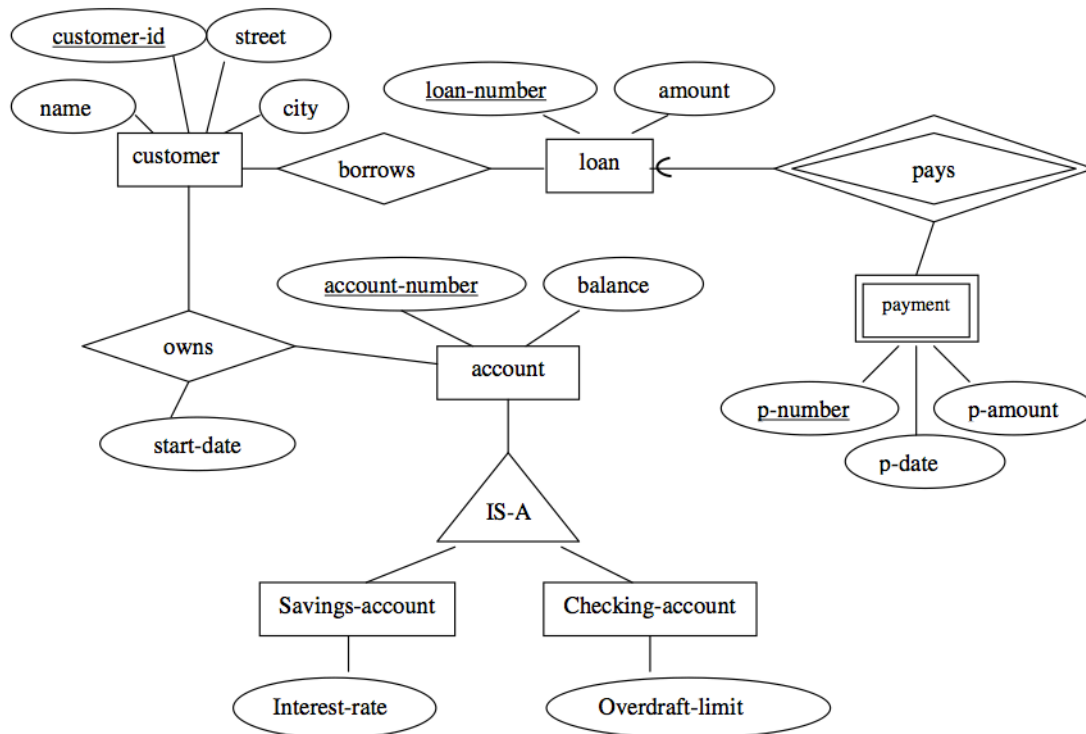
a) As a first step, design an E/R diagram for your database. The following is the description of the application:

- Each series has a name and a description.
- We would like to record information on the regular cast of the series. In other words, for each main actor/actress in the series, we want to record his/her name and address. You may assume that the cast doesn't change.
- Each series has many episodes. Each episode has an episode number, the year and date it was first aired, and the length of the episode in minutes. The episode number uniquely identifies an episode with respect to the series, but two different series can have the same episode number.
- Registered viewers can comment on any episodes. For each comment, we want to record its post date and content.
- For each registered viewer, we want to record his/her userID and password.

Clearly specify any additional assumptions you might make.

b) Next, translate your diagram to a relational schema. Try to minimize the number of relations your solution has, and merge relations where appropriate. Don't forget to specify the keys.

3. Consider the following E/R diagram from a bank accounts application domain.



a) Describe in plain English the application modeled by the diagram.

b) Translate the E/R diagram to a relational schema. Try to minimize the number of relations your solution has, and merge relations where appropriate. Don't forget to specify the keys. In translating a subclass hierarchy, use the E/R style translation.

c) Write SQL "CREATE TABLE" statements that create the tables (relations) in your schema. Specify the necessary key, foreign key, NOT NULL and uniqueness constraints (justify your choices).