

CS 122A: Introduction to Data Management – Fall 2021

Homework 1: E/R Modeling (100 points)

Due Date: Wednesday, Oct 6 (6:00 PM)

Partner Policy

The partner policy for this HW (and subsequent) HW assignments will be that your CS122A partner is your “brainstorming buddy”. You can discuss the HW with them freely w/o violating any rules! But - in the end, *this is an individual assignment, and each of you must turn in your own work*. For this particular HW assignment, you should **attempt the assignment alone first**, and then talk to each other, and then write down some notes about the key differences between yours and your partner’s initial ER diagrams. If you do not have a partner, you will need to find someone to talk to after completing ER design on your own - i.e., students who opt to work solo will still need to find some other team to compare diagrams with in order to get credit for the last part of this assignment (see below).

HW Submissions

This HW assignment must be submitted online, as a **PDF** file, through the associated assignment in **Gradescope** (HW1 in this case). See the table below for HW 1’s submission opportunities. Note that after the second deadline, Thursday, October 7th, no further HW 1 submissions will be accepted at all. We cannot accept **any** assignments after that time since we will be publishing a correct solution at that time. *Plan to turn your work in on time!* If possible, save your one dropped assignment for the end of the term when you are most likely to want or need it, or for a time when something unanticipated crops up.

Date / Time	Grade Implications
Wednesday, Oct 6 (6:00 PM)	Full credit will be available
Thursday, Oct 7 (6:00 PM)	10 points will be deducted

E-R Schema Design [90 pts]

After a year of using many different —and often confusing— online platforms to take and manage your courses, you and a group of your fellow CS122a students have decided to create an encompassing online course management system called **SWOOSH (Superior Web-Optimized Online Student Habitat)**, an application that can be used across various learning settings —universities, teach-for-dollars companies, corporate tech online training programs, etc. Many students and instructors alike have struggled to use online services not just to hold and record remote lectures, but also to manage follow-up discussions and clarify confusions; your goal is to help them out by providing a database-backend service that integrates teleconferencing with a forum-type, discussion-centric platform. To centralize the management of their courses, instructors can schedule, hold, and record meetings on SWOOSH, after which the recordings will be uploaded automatically. Students can then choose to attend meetings or watch recordings of the meetings, after which they can create posts about meetings to start follow-up discussions or ask questions, and reply to already created posts.

In an interview with the SWOOSH thought leaders, they shared the following initial thoughts on their vision for the data model needed for this new service:

1. Each user will be assigned a unique user id by SWOOSH when they first join the service. Users will also have to provide an email address and their name consisting of a first name and last name.
2. A given SWOOSH user will be a student or an instructor, or possibly both. (Instructors can also be students in SWOOSH, e.g., an instructor might wish to take another instructor's course!) Instructors can optionally have information about their stature and experience: An instructor can have a title and a set of education records, where each education record consists of a degree, major, school and graduation year. A student, on the other hand, could optionally specify their occupation. (Remember that this is a general-purpose platform, not just a platform for university students.)
3. A course is taught by an instructor; multiple instructors can teach the same course, and an instructor can teach multiple courses. Courses should have a unique course id, a course name, and an optional description of the course. A course can be taken by multiple students, and a student may take multiple courses. The enrollment of a student in a course will be timestamped with the date of enrollment.
4. Central to SWOOSH is its support for meetings. A course can be associated with multiple meetings, but each meeting can only be for one specific course. Meetings should have a unique meeting id, a duration, a start time, a meeting name, an indicator for whether participants should join muted or not, and optionally a passcode for the meeting. Meetings are hosted by an instructor, but multiple instructors cannot host the same meeting. Students can attend meetings, and many students can be attending the same meeting.
5. Some meetings can be recurring, so recurrence information about such meetings should be captured in a Recurrences descriptor. Recurrences contain a system-generated unique id, the day that the meeting repeats on (for simplicity's sake, we will assume that meetings recur one day a week), and the end date of the recurrence. Each recurrence specification will be associated with multiple meetings; once a recurring meeting is scheduled, all of the implied actual individual meetings are automatically generated (i.e., eagerly scheduled).
6. Meetings can be recorded automatically and stored in SWOOSH's database. Recordings should include the following information: a unique recording id and optional start and end times for the recording. The duration of a given recording is also of interest and can be computed based on the recording's start and end times. Multiple recordings can be made for a single meeting (like if there is an interruption!), but any one recording cannot be about different meetings. The recordings of a meeting do not overlap each other in time.
7. Students can choose to watch recordings fully or partially and may do so multiple times; while watching recordings, SWOOSH stores the duration of the student's watching sessions, which contains the start and end times of the recording segment that the student has watched.
8. The forum-type functionality of SWOOSH is mainly implemented through support for Posts. Posts should have a unique post id, a post type, a creation time, a set of topics that the post is about (i.e. tags), and optionally a body. Both students and instructors may create posts, but multiple users may not create the same post.
9. Users can mark a post as a good post by giving it a "thumbs up". A user can mark multiple posts as good posts and a post can be marked good by multiple users.
10. A student's number of good posts is of interest and may be computed based on its association with other entities (i.e., their posts and whether those have been marked as a good post).
11. Posts are always associated with meetings. Users can create multiple posts for a single meeting, but a given post can only be about one meeting. SWOOSH also offers the option to reply to a post. As a result, a post might be a reply to another post, and a post might have multiple reply posts, but a given reply post may only be associated with one (original) post.

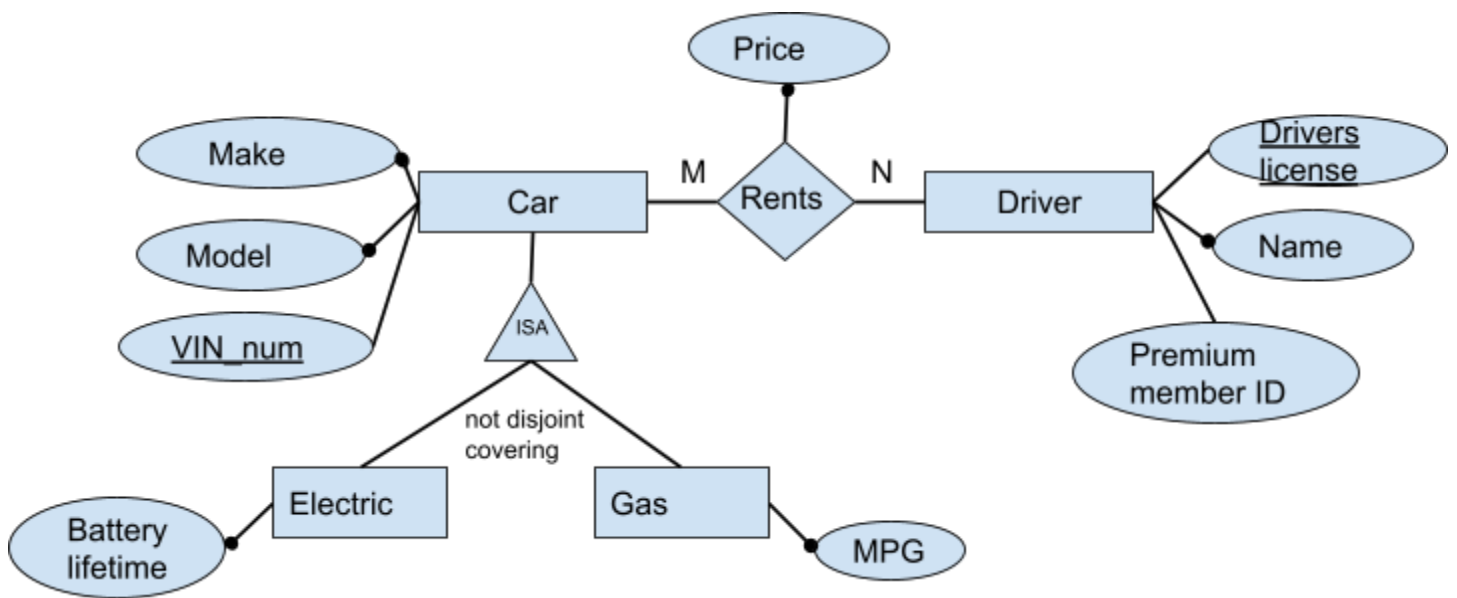
Develop an E-R schema to represent the required information and express its design in the form of an E-R diagram. (Please use the E-R notation from the lectures; do NOT invent your own notation or use UML. You can find a decent E-R model tutorial on the web at http://www.tutorialspoint.com/dbms/er_model_basic_concepts.htm, but beware of its different ISA notation, which you should not use here.) Be sure that your E-R design captures all of the implications of the planned SWOOSH user model and business model, including all aspects below ((a)-(b)), and be sure to make use of advanced E-R modeling features such as weak entities, ISA relationships, composite attributes, set-valued attributes, and mandatory attributes where appropriate. Please use the ISA notation from the lectures and be sure to think about overlap and covering constraints when detailing your design. NOTE: Do NOT attempt to design your E-R schema by making just one pass over the bullets above!!! You will need to carefully analyze all of the requirements in order to identify the available attribute and relationship commonalities for ISA purposes; attempting a one-pass design is likely to lead you to miss important (inheritance-based) sharing opportunities and lead you to a much messier-than-necessary design. Cleanliness matters!

Using the schema template that we provide, draw your E-R diagram, including:

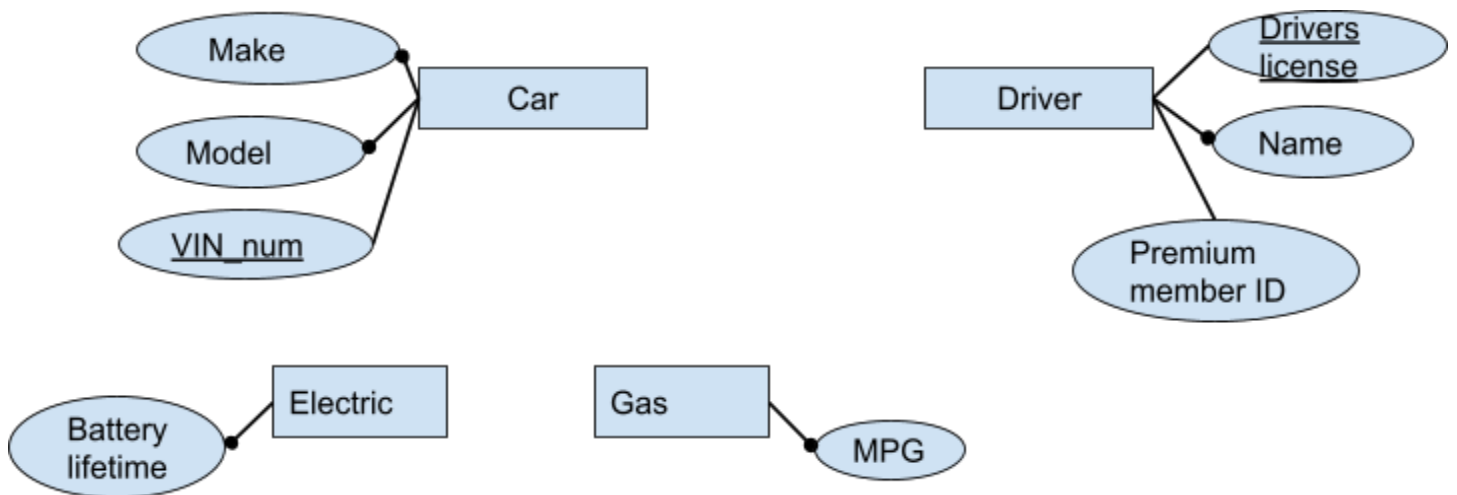
- (a) [45 pts] All of the relevant entities and their associated attributes, including their keys. (For Entities, you may **ONLY** use “User”, “Student”, “Instructor”, “Courses”, “Meetings”, “Recurrence”, “Recordings”, “Posts”. Do **NOT** introduce any additional entities or use fewer entities than these. Note that the template is not intended to indicate strong vs. weak entities, so you should add that level of detail if/as appropriate.)
- (b) [45 pts] All of the relevant relationships and their associated relationship attributes, if any. For the relationships, each one must also have appropriate cardinality and participation constraints indicated on your diagram.

Be sure to grab the HW #1 Google Slides schema template from the CS122a web page and use that as the basis for drawing the E-R schema that you turn in. You can make a copy of the Google Slides file and edit that. (Your solution will **NOT** be accepted if you do not use our provided physical layout for your diagram!!! Without such standardization, it is too difficult to grade everyone’s answers with sufficient thoroughness.) Please use some sort of software to draw your E-R diagram - again, please do not change the spatial placement of the entities - instead of just drawing it by hand.

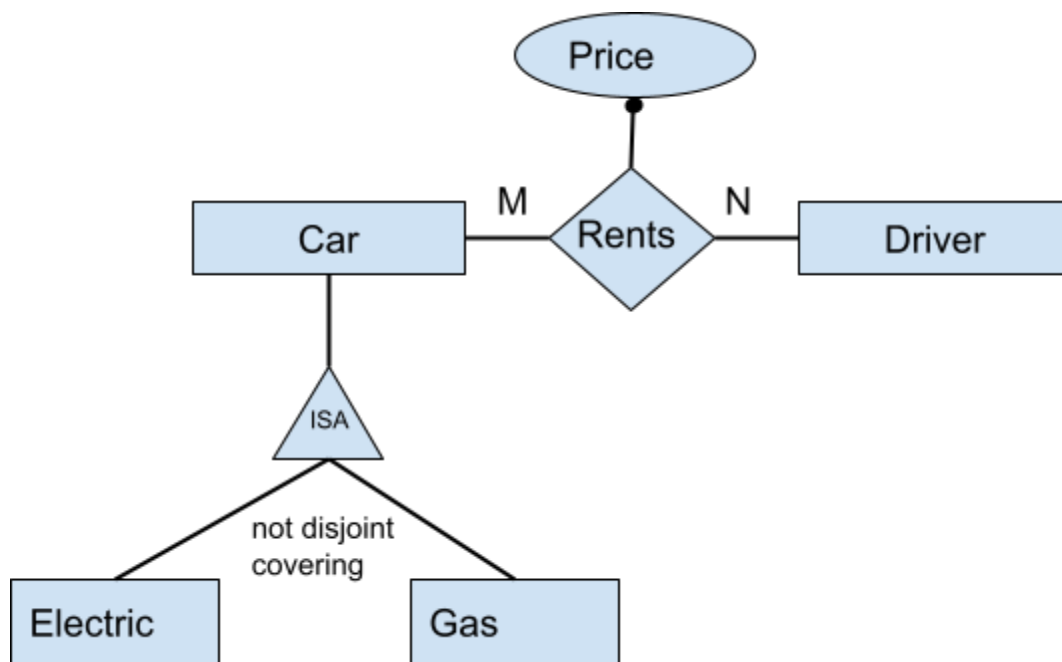
Important: As you will see in the template, there are actually two similar pages that have all necessary entities on them. **For the first one, just indicate the *attributes* for each entity without introducing any relationships. For the second page of your overall E-R diagram, just show the *relationships*, their *cardinalities*, their *attributes* (only the relationship attributes), and the *ISA constraints*.** Please do not show any entity attributes in the second E-R diagram. As an example, suppose that your final design for SWOOSH’s conceptual schema was the following E-R diagram:



In your first E-R diagram you should show all of the entities with their attributes - but no relationships and no IsA's. This diagram will be used to check that you have correctly identified all of the entities and all of their attributes as needed for SWOOSH.



Then, in your second E-R diagram, you should show the graph of all entities (without their attributes) and relationships (with their attributes if any):



This latter diagram will be used to check your overall design w.r.t. its relationships -- and it will give you a nice overview to refer back to later on in the project as you try to construct your relational queries. You will eventually want to print out the SWOOSH E-R diagram and take it to a tattoo parlor or else keep it sitting next to you as you work on each of the subsequent HW assignments!

Note: If the example pictures shown above are indeed the same as your final design for SWOOSH, you should probably read the SWOOSH founders' information again more carefully... (-:-)

E-R Design Comparison w/ Partner [10 pts]

Given the description and the requirements above, and even with the limitations imposed by having our template, there are still multiple different ways that one might interpret or represent the requirements! After completing a first design pass for this assignment **on your own**, talk to your brainstorming buddy and compare your initial E-R diagrams. Make some notes about the differences that you find between your initial E-R diagram and your partner's: Look carefully at your respective relationships, their cardinality and/or participation constraints, all the attributes, etc., and talk to each other about why your model looks the way it does. After this meeting, go ahead and update your own model based on anything you now want to model differently. To get full credit for this part of the assignment, the PDF that you turn in should include:

- Your final E-R diagram, *laid out using the provided template(s)*.
- A list of the main differences that you identified in your partner meeting and a few sentences summarizing what you learned from doing the comparison exercise.
- Your own initial E-R diagram, just for the record, *also laid out using the template(s)*, so we can see what you started with versus what you ended up with. (Note: The details of this first version will not be graded for correctness.)

Once you are done with all of the questions, you can download the template by going to File -> Download -> PDF. Make sure to submit the PDF file, otherwise Gradescope will not accept your submission!