

CS5200: Homework 2: Logical Modeling

Fall 2024

1 Overview

This assignment asks you to construct logical models given informal descriptions of data to be represented. By completing this assignment, you will practice the following skills:

- Understanding informal descriptions of data, such as you might receive from your users
- Drawing an entity-relationship diagram that describes relations and the relationships between them.
- Constructing a logical (relational) model from the informal description.
- Selecting appropriate primary keys for the relations in your model.
- Identifying violations of 1NF, 2NF, and 3NF
- Refactoring designs to satisfy the requirements of 3NF

For this assignment, you should submit an entity-relationship diagram for each problem. Please submit these as a ZIP file containing one PDF file per problem. (If you should need to re-upload your submission for any reason, be sure that the latest ZIP file contains all of your PDFs.)

Each entity-relationship diagram must satisfy the following requirements:

- Each relation must have a name and a primary key. Simple and compound keys are both permitted. Prefer natural keys where appropriate, but you may use artificial keys when necessary.
- List all appropriate attributes for each relation, and clearly identify foreign keys.
- Relationships between relations must be clearly indicated using the notation described in [lecture 3](#) (logical modeling), including all cardinality and modality annotations.
- Each relation must be in third normal form. You may either design the models in 3NF directly, or, if it is easier for you, you may construct a starting model and gradually refactor it into 1NF, 2NF, and finally 3NF, as I have done in lecture. If you produce your 3NF model by repeatedly refactoring a starting model, you should submit only the final version of the model; please do not include the intermediate stages.

1.1 Drawing Diagrams

To produce E-R diagrams, you may use a drawing tool or draw them by hand. If you draw the diagrams by hand and scan or photograph them, be sure that the diagrams are legible in the PDF that you upload.

For drawing tools, I recommend [draw.io](#), which is a good free tool that you can run in your browser; it has decent support for E-R diagrams. You may choose to use other tools if you like, as long as they can generate PDF diagrams that conform to the conventions we have discussed in class.

Sales Receipt

Date

01/31/2008

Sale No.

549

Sold To

John Q. Customer
1234 Anystreet Way
Sometown, OK 99999

Item	Description	Qty	Rate	Amount	Tax
OtherItem	Platinum-Coated Kanooter Valve	1	50.00	50.00	Tax
			0.00	0.00	
Customer Message			Tax (7.5%)	3.75	
			Total	53.75	

Figure 1: Sales receipt

2 Questions

Problem 1. Construct a logical model to represent data for sales receipts, as in the diagram in figure 1. Each receipt has the following properties:

- Customer name
- Customer address (street, city, state, zip code)
- Order date
- Order number (“Sale No.” in the figure)
- An optional customer message (blank in the included figure)
- Some number of items. Each item has the following properties:
 - An item number
 - An item description
 - The quantity of that item ordered
 - The unit price of that item (“Rate” in the figure)
 - The total price of the quantity ordered (“Amount”)
 - Whether the item is subject to sales tax
- A tax rate
- The total cost of the order

Question 2. Consider an education application, similiar to Canvas, that tracks university courses, people (who are either students or faculty), course registrations, assignments, and grades. All people have an ID number, a first name, and a last name. Additionally, students have an enrollment date, and faculty members are part of a department. Courses offered by the university are part of a department, and they have a catalog number (e.g., 'CS5200 '), a description, and

an optional lecturer, who must be a faculty member. (We allow missing lecturers for situations in which, for instance, a department is planning its schedule and hasn't yet determined which faculty members will teach which classes.) A department has a name and (optionally) a chairperson, who must be a faculty member. A student can register for one or more courses.

A course can have arbitrarily many assignments, each of which has a name, a due date, a body, and a weight (i.e., how many points students can earn on that assignment). Students may make arbitrarily many submissions for an assignment: each submission has a timestamp indicating when it was made and the text of the submission. Also, a student can have at most one grade for an assignment. Finally, it is possible for users of the system to send messages to other users. Each message should have a sender, recipient, title, body, and the time at which it was sent.