

COMS 363 Fall 2022

Assignment II

Percentage in your final grade: 10%

Maximum score for the assignment: 100 points

Learning objectives:

1. Practice conceptual database design using the Entity Relationship (ER) model.
2. Practice designing a relational database from an ER diagram.

Instructions: The homework is individual work. Collaboration with others who are not teaching staff is not allowed. Use software to draw your ER diagram. Your ER diagram must be legible. We will deduct 10 points for the submission of a hand-drawn ER diagram. Use only the ER notations discussed in class. Other ER notations are not accepted for this assignment.

Referenced book:

- [1] Ramakrishnan and Gehrke. Database Management Systems 3rd edition, McGraw-Hill Higher Education, 2003.

Submission requirements

Submit A zip file named <netid>HW2.zip with the following files. Replace <netid> with your IASTATE netid.

<netid>Q1.pdf

<netid>Q2a.sql

<netid>Q2b.sql

<netid>Q2c.txt

Each file begins with a comment -- followed by "Author" and followed by your first name and last name, for example,

-- Author Wallapak Tavanapong

Testing of your answers: We will run each of your scripts using MySQL Workbench on MySQL Community Server 8. Make sure that your script runs without errors.

If your script gives an error, a zero score will be given. Contact your grading TA within seven days after you get your graded assignment back to explain your code to receive partial credits. After that period, there is no further change in your assignment score.

Questions

1. (50 points) This question was modified from Exercise 2.6 from [1].

Suppose you are to design a database for an imaginary Ames County Airport to store the following information.

- Each airplane is uniquely identified by a registration number and must be of a specific model. Each plane has a unique name.

- Each plane model is identified by a model number (e.g., DC-10) and has a capacity, weight, and amount of full-tank gas.
- Technicians, managers, and traffic controllers are airport employees. A social security number uniquely identifies an employee. The information about an employee to be kept is name, address, phone number, and a union member number. The union member number is unique for each employee. The database should store the education degree of a technician. Each plane technician is an expert on at least one plane model. For a traffic controller, the database should keep the latest exam date the controller passed. A manager's attribute also includes a benefit. No employees are allowed to be in more than one subclass. There can be other types of employees.
- Employees may have dependents. Names of dependents distinguish the dependents within each employee, but different employees may name their dependents the same. Each dependent must be associated with exactly one employee. The information about the dependents to be kept is their relationship with an employee and date of birth.
- A test type has a Federal Aviation Administration (FAA) test number, a name, and a maximum possible score. The FAA test number is unique for each test type.
- FAA requires keeping track of each time a technician tests an airplane using a given test type. Each testing event has the date, the test type, the number of hours spent on the test, and the test score. The same technician is allowed to do the same test (i.e., same FAA test number) on the same airplane on different dates. The client wants to be able to retrieve information about past testing events.
- The client wants to keep track of which manager monitors which test event. A manager can monitor multiple test events, and several managers can monitor a test event.
- An FAA test number FA50 is not allowed for the DC10 airplane model.

Draw an ER diagram that captures all the above requirements. You can add a constraint if your constraint does not conflict with the existing requirements. But you must specify the additional constraints you add to the database requirement to avoid point deduction. Otherwise, the teaching staff does not know whether your solution is wrong or you add other constraints.

Submit your drawing in pdf format as <netid>Q1.pdf.

Checklist to help avoid point deduction:

- The primary key of each entity set must be indicated.
- Relationship sets do not have underlined attributes because such a notation is not allowed. Each relationship can already be uniquely identified by the primary key values of the entities participating in the relationship.
- If there is any candidate key, write the name of the attribute(s) that form the candidate key and the name of the entity set at the corner of your ER diagram. For example, University ID: Candidate key of Student.
- All the given constraints need to be modeled in the ER diagram. For any constraint that ER is not capable of modeling, note it down in a separate sheet.
- Make sure that arrows (if any) are pointed toward the corresponding relationship set. No rectangles are connected without a diamond shape in the middle. No diamond shapes are directly connected.

Other information: You can use draw.io, Microsoft PowerPoint, Google Slides, or any other drawing software that can draw the notations we use in class. UML database notations are somewhat different

from the notations we use in this course. UML database diagrams allow specification of indexes which are the results of physical design (not the conceptual design that we are currently doing). Relationships are limited between two entity sets only, whereas the ER notations we use support relationships with two or more entity sets. ER model was originally designed to capture data and relationships at a conceptual level, which is independent of the underlying implementation. Therefore, we stay with the ER notations used in class.

2. (50 points) Given the ER diagram in Figure 1.

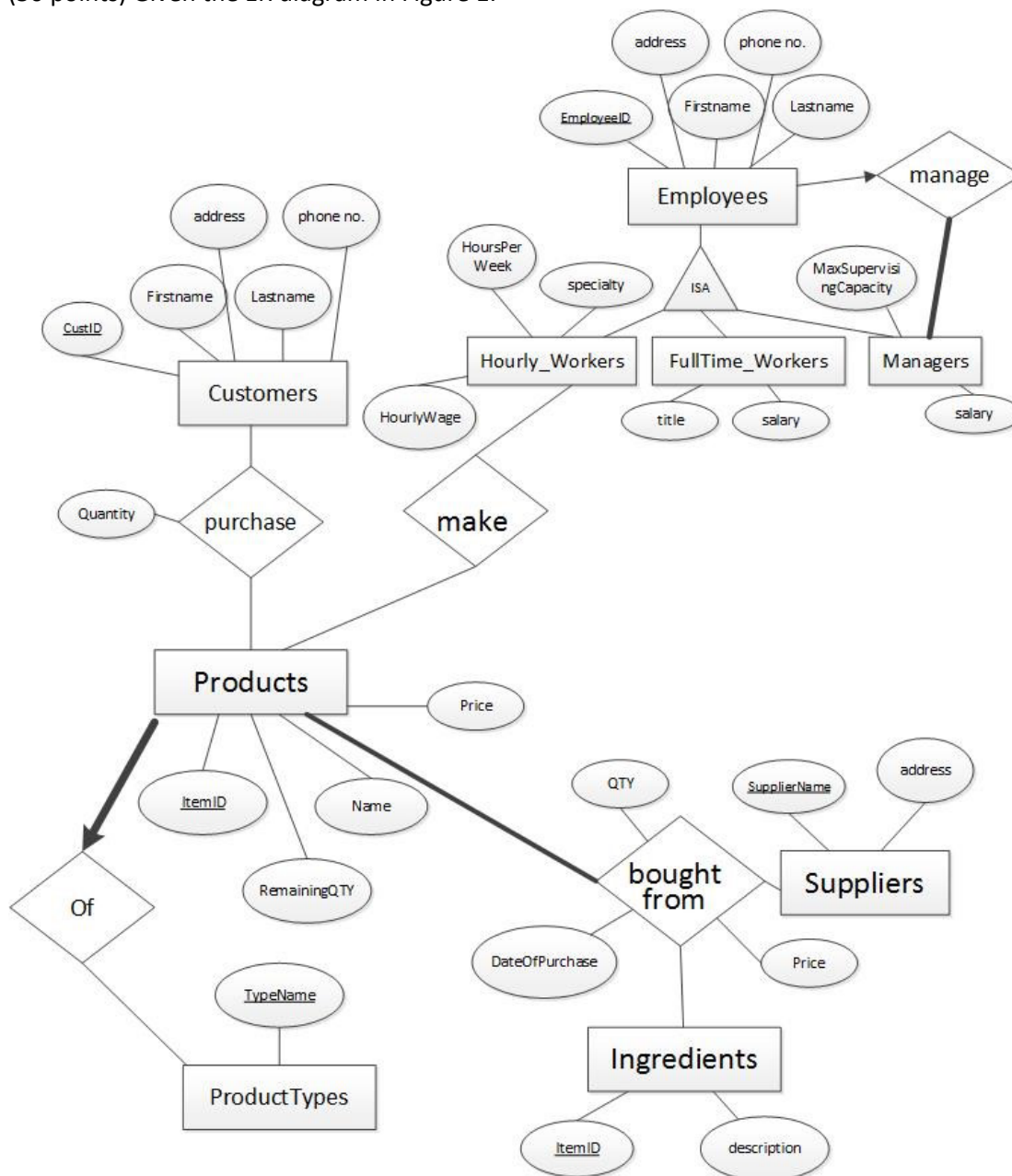


Figure 1: ER diagram of a hypothetical factory.

Design a corresponding relational database schema from the ER diagram in Figure 1. Use the method discussed in class to convert ER notations to relations while minimizing unnecessary

redundancy and unnecessary relations. Implement as many constraints given in the ER diagram as possible.

a) (30 points) Provide a script of SQL DDL statements that create relations under a database called <netid>hw2q2. The relation names and attribute names should be meaningful for ease of understanding and maintenance of the database over time. Name the script <netid>Q2a.sql. The script must drop any existing tables of the same names before creating the new ones.

b) (10 points) Provide a script of SQL DML statements to add at least four rows of data of your choice into each table, but the data must satisfy all the constraints. Name this script <netid>Q2b.sql. The script must delete any of the existing rows before the insertion.

We expect that your data be different from other students as the homework is individual work.

c) (10 points) Write down all the constraints in the ER diagram in Figure 1 that cannot be enforced using relational database design (i.e., the use of relation schemas, primary key, foreign key, not null, and unique constraints). Put your answer in Q2c.txt. We will be writing database triggers to enforce this type of constraints later on.