Lab #3

in-class assignment on data modeling, normalization, functional dependecies and some querying

Purpose: The purpose of this lab is to apply the data modeling concepts that you have covered so far to realistic cases.

Teams: You may work on this lab in teams of 3. All submitted work should be clearly written (or preferably submitted electronically) with every team member's name clearly printed at the top of each page.

Activities:

Map ER and EER diagrams to relations

Identify the functional dependencies in a schema

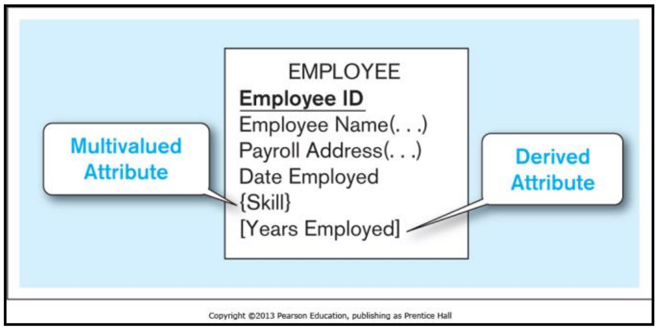
Decompose a relation into 1NF, 2NF and 3NF relations, identifying anoamlies and functional dependecies along the way.

Part I. Mapping of ER and EER diagrams to relations.

For each of the following ER and EER Diagrams:

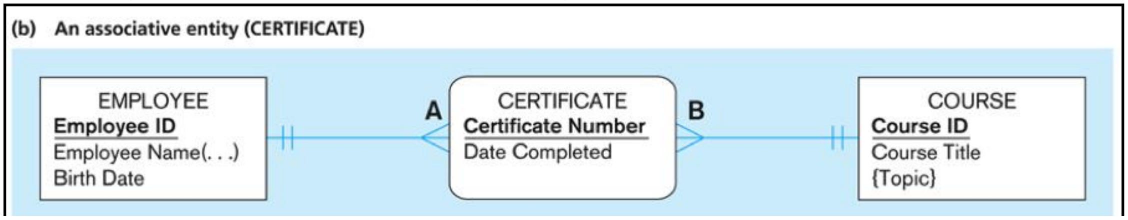
- ●Map the ER diagram to a dependency diagram (an example is Figure 15.3 your text) (only for the questions with as * next to them)
- ●List the relations. For instance, STUDENT(<u>Student_ID</u>, Student_Name, Student_GPA) FK's in italics
- ●List the functional depedencies, for instance Student_ID --> {Student_Name, Student_GPA} Be careful of partial dependencies. Your key is always the determinant. The dependent attributes are those that are fully dependent on that key.
- ●Make sure that your relations are in 3NF.

*1. Multivalued attribute--this is repeating, and if mapped directly would be a violation of 1NF. A derived attribute is calculated, and does not have to be stored.

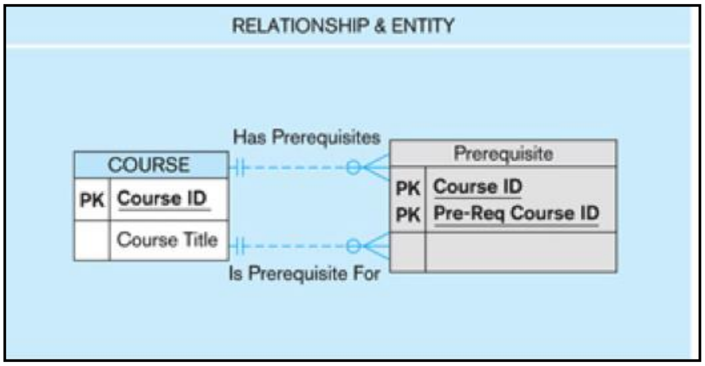


Copyright ©2013 Pearson Education, publishing as Prentice Hall

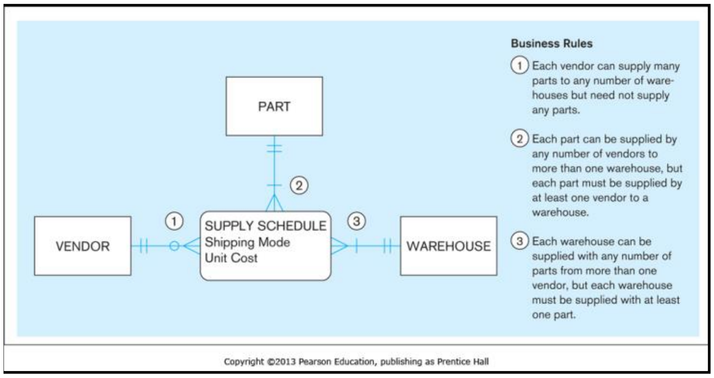
2. Certificate is a relationship entity (associative entity, intersection relation, same thing) Be sure that you show the correct primary key. Note that a course can have many topics. You may want to assume that a topic may be covered in different courses as well.



3. This is a unary relationship. A course can have many prerequisites and can be a prerequisite for many other courses. It is an m:n unary relationship.



4. This is a ternary relationship.



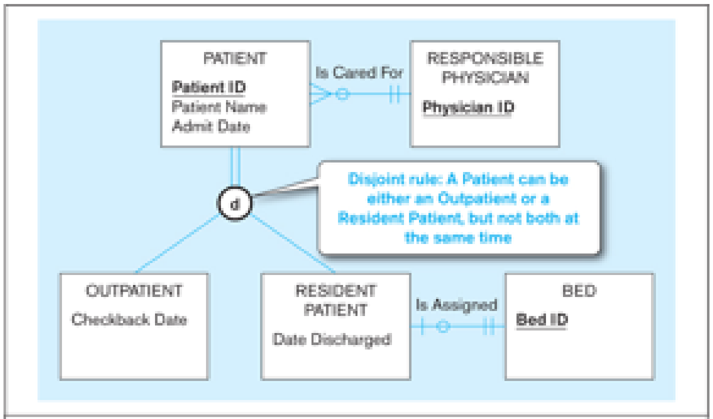
Business Rules

1. Each vendor can supply many parts to any number of warehouses but need not supply any parts.

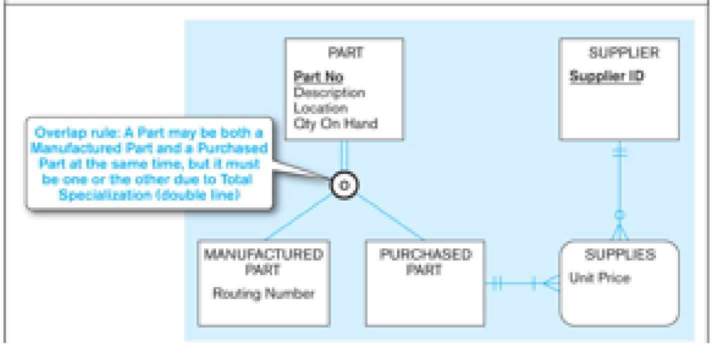
2. Each part can be supplied by any number of vendors to more than one warehouse, but each part must be supplied by at least one vendor to a warehouse.

3. Each warehouse can be supplied with any number of parts from more than one vendor, but each warehouse must be supplied with at least one part.

5. Here are two different EER diagrams. Let's call the first Patient, and the second, Part.
For one of them, it is possible to map the diagram to relations in two different ways.
Map both of these EER diagrams, and specify two different mappings (if you can think of them) for one of the diagrams.

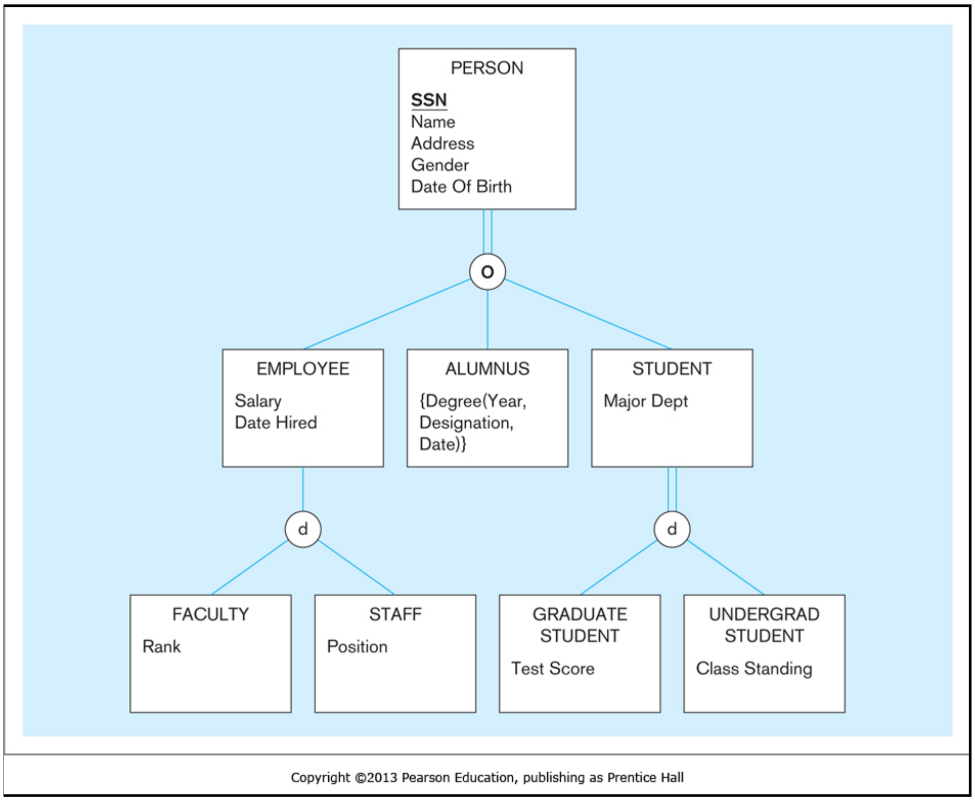


Disjoint rule: A Patient can be either an Outpatient or a Resident Patient, but not both at the same time



Overlap rule: A Part may be both a Manufactured Part and a Purchased Part at the same time, but it must be one or the other due to Total Specialization (double line)

6. This one has several levels of supertypes/subtypes, with partial, complete, overlap and disjoint. Have fun with this one!!



Part II. Normalization Example

This example uses a slightly different approach to normalization. We begin with all of the attributes in one big
We then have to decompose this relation into several normalized relations. You will achieve this step-by-step,
converting the relation into a set of 1NF relations, then 2NF relations, then 3NF relations. After each normal fo
submit your answers and retrieve an answer key for that normal form. You will then work off of the answer ke
deternine the next normal form relations. This way, you won't propagate any errors that you may make.

Consider the following case:

LikeNew Cars! buys used and damaged cars and trucks, refurbishes them, and then resells them. LikeNew Ca these vehicles from repair shops, auctions, used car dealers, or individuals who are sellin their vehicles. They l of *finders* whose job it is to search and locate vehicles. When the finder finds an *owner* of a vehicle for sale, t calls the owner and asks him or her questions about the vehicle manufacturer, model and the year the vehicle w Sometimes, an owner has an entire fleet of trucks or cars for sale. Two individuals, called *Inspectors* , are hire the condition of the car or truck, appraise its value, and reports the *bid* to the finder. The finder decides what t owner for the vehicle, and if the owner agrees, the sale is completed.

LikeNew Cars! is currently storing all of this information in an Excel spreadsheet, with about 2000 records. A 1500 of the records involve 1 or 2 cars or trucks. The remaining 500 records have between 3 and 20 vehicles.

Here is a list of the fields that are included in this spreadsheet:

Owner (Owner_ID, Owner_Name, Owner_Address, Owner_Bank, Owner_Phone#, Finder_Name, Finder_Pho Vehicle1_ID, Manufacturer1, Model_Name1, Year_Built1, VehicleAddress1, AskingPrice1, AmountPaid1, Inspector1_Vehicle1_Name, Inspector1_Vehicle1_Cell#, Inspector1_Vehicle1_Bid Inspector2_Vehicle1_Nam Inspector2_Vehicle1_Cell#, Inspector2_Vehicle1_Bid, ... , Vehicle20_ID, Manufacturer20, Model_Name20, Year_Built20, VehicleAddress20, AskingPrice20, AmountPaid20, Inspector1_Vehicle20_Name, Inspector1_Vehicle20_Cell#, Inspector1_Vehicle20_Bid Inspector2_Vehicle20_Name, Inspector2_Vehicle20_ Inspector2_Vehicle20_Bid)

1. Primary Key: Owner_ID

The attributes should be self explanatory, but just in case, here are some semantic definitions:

OWNER_ID	The unique identifier for each individual selling a vehicle
OWNER_NAME	The name of the individual selling a vehicle
OWNER_ADDRESS	The address of the individual selling a vehicle
OWNER_BANK	The name of the owner's bank
OWNER_PHONE#	The primary contact phone for the owner
FINDER_NAME	The name of the employee who searches for vehicles
FINDER_PHONE#	The phone number of the employee who searches for vehicl
VEHICLE1_ID	The vehicle identifier for the first vehicle
There can be up to 20 vehicles stored from each owner. So you will need the following fields up to 20 times!	
MANUFACTURER1	The manufacturer of the first vehicle
MODEL_NAME1	The model name for the first vehicle
YEAR_BUILT1	The year for first vehicle was built
VEHICLE_ADDRESS1	The address for the first vehicle
ASKING_PRICE1	The amount the owner wants for the first veicle
AMOUNT_PAID1	The amount paid for the first vehicle (if a sale is agreed upo
For each vehicle, there must be two inspectors.	
INSPECTOR1_VEHICLE1_NAME	The name of the first inspector who appraised the first vehic
INSPECTOR1_VEHICLE1_CELL#	The cell # of the first inspector who appraised the first vehic
INSPECTOR1_VEHICLE1_BID	The first inspector's bid (appraisal) of the first vehicle

INSPECTOR2_VEHICLE1_NAME	The name of the second inspector who appraised the first ve
INSPECTOR2_VEHICLE1_CELL#	The cell # of the second inspector who appraised the first ve
INSPECTOR2_VEHICLE1_BID	The second inspector's bid (appraisal) of the first vehicle

Clearly, this spreadsheet is problematic. Your job is to convert the Excel file into a set of 3NF relations.

1. If the primary key is the owner, then imagine up to 20 vehicles' information being stored in one record in this spreadsheet.
   - Identify one insertion anomaly. State what you would like to insert, and state why it can't be done with this structure.
   - Identify one update (modification) anomaly. State what you would like to update, and why it is a problem.
   - Identify one deletion anomaly. State what you would like to delete, and what unintended consequences would that cause?
2. Place the relation into a set of 1NF relations. Follow these steps:
   - Remove the attributes that violate 1NF, and put them into a separate relation.
   - Preserve the relationship: make sure that the new relation that you just created has a foreign key in it that relates it to the original relation.
   - Assign a key: the new relation should have a key that determines the non-key attributes in the relation. Recall that at this stage, you are only looking for 1NF compliance. Your key and your relation don't have to comply with anything beyond 1nf at this point.
3. Submit your 1NF relations on a separate sheet of paper. At the top of the sheet of paper, write "1NF", and then clearly print the names of your team.
4. Working off of the answer key for the 1NF relations:
   - identify any anolamies that still exist (give an example)
   - place the relations into a set of 2NF relations
   - submit your 2NF relations and collect an answer key.
5. Do the same for 3NF.
   - As a final step, explain why the anomalies that you identified earlier no longer exist.

; relation.
 first
orm, you will
:y to

rs! finds
ıave a team
he finder
ˈas built.
d to evaluate
o offer the

ıpproximately

ne#,

e,

Cell#,

es
n)
cle
cle

ehicle
ehicle