CIS 550: Database and Information Systems

Exercise 5: Relational Design Theory

Part 1: Functional Dependencies

Your friend likes to keep track of their computers and has created an Excel spreadsheet with the following columns (attributes), none of which are null:

Manufacturer, Model#, Serial#, Description, Store, PurchaseDate, Price

Each row in this spreadsheet corresponds to a computer that they own. Since your friend has a lot of computers to keep track of, and you are a fan of databases, you wish to create a relational database to replace the Excel spreadsheet. You initially design the database as a single relation consisting of all the attributes in the spreadsheet, but realize there are certain redundancies and want to improve the design based on what you are learning in this course.

1. Give a *minimal* set of functional dependencies F that hold for this schema assuming **only** that:

- The manufacturer, model number and serial number uniquely identifies the store, purchase date and price;
- The description of a computer is determined by the manufacturer and model number -- in particular, it does not vary depending on the store in which the computer was bought or the serial number of the computer.

ANSWER
F = {Manufacturer, Model#, Serial# → Store, Purchase Date, Price
Manufacturer, Model# → Description}

2. What is the key of your relation? Prove this by showing that the closure of the set of attributes in the key includes all attributes using the technique discussed in lecture.

ANSWER
The key is {Manufacturer, Model#, Serial#}.

{Manufacturer, Model#, Serial#}+= {Manufacturer, Model#, Serial#} U {Store, PurchaseDate, Price} U {Desc} = {Manufacturer, Model#, Serial#, Store, PurchaseDate, Price, Description}
The second set corresponds to the right-hand side of the 1st dependency, and the third to the right-hand side of the 2nd dependency.

Now suppose that you decompose into R1(Manufacturer, Model#, Serial#, Store, PurchaseDate, Price), R2(Manufacturer, Model#, Description).

3. Does this decomposition have a lossless join? Why or why not?

ANSWER
Yes, because R1∩R2= {Manufacturer, Model#), R2-R1= {Description} and Manufacturer, Model# → Description

4. What is the projection of F over R1? The projection of F over R2? Does the decomposition preserve dependencies (and say why)?

ANSWER
The projection of F over R1 is F1= {Manufacturer, Model#, Serial# → Store, Purchase Date, Price}. The projection of F over R2 is F2= {Manufacturer, Model# → Description}. The union of F1 and F2 is F, so it does preserve dependencies.

5. Given a schema AB and functional dependencies F = {A→ B}. What is the closure of F, F^+?

ANSWER
F^+= {A→ B, A→ A, B→ B, A→ AB, AB→ B, AB→ A, AB→ AB}

6. Given a schema ABCD and functional dependencies F= {A→B, B→D, C→A, A→D, C→D}, is the decomposition {ABC, BD} dependency preserving? Be careful to look at F+ in the projection of F over each schema.

ANSWER
F1= {A→ B, C→A}, F2={ B→ D}

The dependencies in F that are missing in the union of F1 and F2 are {A→ D, C→ D}, however, A→ D is a transitive dependency from A→ B and B→ D. Similarly, C→ D is a transitive dependency from C→A, A→ B and B→ D. So this is dependency preserving.

Part 2: Decompositions

Suppose we have a database for an investment firm, consisting of the following attributes: B (broker), O (office of a broker), I (investor), S (stock), Q (quantity of stock owned by an investor) and D (dividend paid by a stock).

Given R= BOISQD, the set of functional dependencies are:

F= {S→D,
I→B,
IS→Q,
B→O,
ISQ→D}

7. Find a key for R.

IS is a key for R, since IS+= {I, S, D, B, Q, O}

8. How many keys does R have? Justify your answer.

Only one. I and S must be in any key, since they never appear on the right-hand side of a dependency. Furthermore, IS is a key, so it must be the only key. Note that there are many other superkeys.

9. Suppose we decompose R into ISQ, IB, SD and ISO. Find the minimal covers for F projected onto each of these relation schemes.

F = {S→D, I→B, IS→Q, B→O}

F_{ISQ} = {IS→Q}

F_{IB} = {I→B}

F_{SD} = {S→D}

F_{ISO} = {I→O)

10. Does the decomposition (ISQ, IB, SD, ISO) preserve dependencies? If not, state the dependencies that are lost.

No, B→O is lost. ISQ→D is preserved since it is implied by augmentation from S→D.

11. Find a decomposition of R into 3NF which has a lossless join and preserves dependencies, using the technique discussed in class.

First we must find an equivalent, minimal set of dependencies. Note that ISQ→D is redundant (implied by augmentation from S→D). So the minimal equivalent set is:

F' = {S⟶D, I⟶B, IS⟶Q, B⟶O}

We can now output a relation for every dependency.
SD, IB, ISQ, BO

The relation ISQ contains a key (IS), so we are done.

Part 3: Normal Forms

12. Given R= ABCD. For each of the following sets of FDs, say whether it is in 3NF, BCNF or neither. Justify your answers.

a. C⟶D, C⟶A, B⟶C

   The key is B. The relation is in neither BCNF nor 3NF since C⟶D, C is not a superset of a key (superkey) and D is not part of a key (prime).

b. B⟶C, D⟶A

   The key is BD. It is in neither BCNF nor 3NF since B⟶C, B is not a superkey, and C is not prime.

c. ABC⟶D, D⟶A

   ABC and BCD are both candidate keys. It is in 3NF but not BCNF since in D⟶A, D is not a superkey but A is prime.