



CSE 4/560
Databases and Query Languages
Homework 3
Total Marks 120

Your Name:

Your email ID:

Your UB Person ID:

1. [6] Discuss the strengths and weaknesses of the trigger mechanism. Contrast triggers with other integrity constraints supported by SQL.

2. [10] Consider a relation about people in the United States, including their name, Social Security number, street address, city, state, ZIP code, area code, and phone number (7 digits). What FD's would you expect to hold? What are the keys for the relation? To answer this question, you need to know something about the way these numbers are assigned. For instance, can an area code straddle two states? Can a ZIP code straddle two area codes? Can two people have the same Social Security number? Can they have the same address or phone number?

3.[6] Consider a relation representing the present position of molecules in a closed container. The attributes are an ID for the molecule, the x, y, and z coordinates of the molecule, and its velocity in the x, y, and z dimensions. What FD's would you expect to hold? What are the keys?

4. [6] Show that if a relation has no attribute that is functionally determined by all the other attributes, then the relation has no nontrivial FD's at all.

5. [4] Let X and Y be sets of attributes. Show that if $X \subset Y$, then $X^+ \subset Y^+$, where the closures are taken with respect to the same set of FD's.

6. [6] Suppose we have a relation schema $R(A, B, C)$ with FD $A \rightarrow B$. Suppose also that we decide to decompose this schema into $S(A, B)$ and $T(B, C)$. Give an example of an instance of relation R whose projection onto S and T and subsequent rejoining does not yield the same relation instance.

7. [12] Suppose we have relation $R(A, B, C, D, E)$, with some set of FD's, and we wish to project those FD's onto relation $S(A, B, C)$. Give the FD's that hold in S if the FD's for R are:
- a) $AB \rightarrow DE, C \rightarrow E, D \rightarrow C$, and $E \rightarrow A$.
 - b) $A \rightarrow D, BD \rightarrow E, AC \rightarrow E$, and $DE \rightarrow B$.
 - c) $AB \rightarrow D, AC \rightarrow E, BC \rightarrow D, D \rightarrow A$, and $E \rightarrow B$.
 - d) $A \rightarrow B, B \rightarrow C, C \rightarrow D, D \rightarrow E$, and $E \rightarrow A$.

In each case, it is sufficient to give a minimal basis for the full set of FD's of S

8. [12] Write the following constraints for attributes of the relation: Movies (title , year, length , genre, studioName, producerC)

- a) The year cannot be before 1915.
- b) The length cannot be less than 60 nor more than 250.
- c) The studio name can only be Disney, Fox, MGM, or Paramount.

9. [12] Write the following constraints on attributes from the below schema

Product(*maker*, *model*, *type*)

PC(*model*, *speed*, *ram*, *hd*, *price*)

Laptop(*model*, *speed*, *ram*, *hd*, *screen* , *price*)

Printer(*model*, *color*, *type*, *price*)

- a) The speed of a laptop must be at least 2.0.
- b) The only types of printers are laser, ink-jet, and bubble-jet.
- c) The only types of products are P C 's, laptops, and printers.
- d) A model of a product must also be the model of a PC, a laptop, or a printer.

10. [12] Write the following constraints as tuple-based CHECK constraints on one of the relations of our running movies example:

Movies (*title* , *year*, *length* , *genre*, *studioName*, *producerC*)

StarsIn (*movie Title* , *movieYear*, *starName*)

MovieStar(*name*, *address*, *gender*, *birth date*)

MovieExec(*name*, *ad d ress*, *c e rt* , *netWorth*)

Studio(*nam e*, *ad d ress*, *presC*)

If the constraint actually involves two relations, then you should put constraints in both relations so that whichever relation changes, the constraint will be checked on insertions and updates. Assume no deletions; it is not always possible to maintain tuple-based constraints in the face of deletions.

- a) A star may not appear in a movie made before they were born.
- b) No two studios may have the same address.
- c) A name that appears in *MovieStar* must not also appear in *MovieExec*.
- d) A studio name that appears in *S tudio* must also appear in at least one *Movies* tuple.

11. [12] Show how to alter your relation schemas for the movie example:

Movie (*title* , *year*, *length* , *genre*, *studioName*, *producerC*)

StarsIn (*movie Title* , *movieYear*, *starName*)

MovieStar(*name*, *ad d ress*, *gender*, *birth date*)

MovieExec(*name*, *address*, *cert* , *netWorth*)

Studio(*nam e*, *address*, *presC*)

in the following ways.

a) Make title and year the key for Movie.

b) Require the referential integrity constraint that the producer of every movie appear in MovieExec.

c) Require that no movie length be less than 60 nor greater than 250.

d) Require that no name appear as both a movie star and movie executive (this constraint need not be maintained in the face of deletions). e) Require that no two studios have the same address.

12. [12] Write the following assertions. The database schema is from the “PC ” example of Q9 above:

Product(maker, model, type)

PC(model, speed, ram, hd, price)

Laptop(model, speed, ram, hd, screen , price)

Printer(model, color, type, price)

- a) No manufacturer of PC ’s may also make laptops.
- b) A manufacturer of a PC must also make a laptop with at least as great a processor speed.
- c) If a laptop has a larger main memory than a PC, then the laptop must also have a higher price than the PC.
- d) If the relation Product mentions a model and its type, then this model must appear in the relation appropriate to that type.

13.[3+3+4] Write the following as triggers. In each case, disallow or undo the modification if it does not satisfy the stated constraint. The database schema is from the “PC” example of Q9 above:

Product(maker, model, type)

PC(model, speed, ram, hd, price)

Laptop(model, speed, ram, hd, screen , price)

Printer(model, color, type, price)

- a) When updating the price of a PC, check that there is no lower priced PC with the same speed.
- b) When inserting a new printer, check that the model number exists in Product.
- c) When making any modification to the Laptop relation, check that the average price of laptops for each manufacturer is at least \$1500.