1. (a): Transform dependencies to canonical forms and drop extraneous attributes:

B->E ~~AB->E~~

B->D ~~ABE->ED:~~ ~~AB->D~~

C->D

C->E

C->F

~~DC->A:~~ D->A

~~DF->A:~~ F->A

E->D

Get the minimum canonical cover

B->E, E->D implies B->D, thus B->D is redundant

C->E, E->D, implies C->D, thus C->D is redundant

Thus, the minimum canonical cover is:

**B->E**

**C->F**

**C->E**

**E->D**

**D->A**

**F->A**

Since B and C do not appear on the right-hand side, B and C are the primary keys.

Group some dependencies with the same determinant: C->EF

Construct relation for each group**:**

**R1: (B,E), R2: (C,E,F), R3: (E,D), R4:(D,A), R5:(F,A), R6(B,C)** (B,C is the superkey)

All of these relations are in BCNF (also 3NF).

(b): Construct the table

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E | F |
| R1: (B,E) | U | K | U | U | K | U |
| R2: (C,A,E,F) | K | U | K | U | K | K |
| R3: (E,D) | U | U | U | K | K | U |
| R4:(D,A) | K | U | U | K | U | U |
| R5:(F,A) | K | U | U | U | U | K |
| R6:(B,C) | U | K | K | U | U | U |

Enforce FDs, we get

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | A | B | C | D | E | F |
| R1: (B,E) | K | K | U | K | K | U |
| R2: (C,A,E,F) | K | U | K | K | K | K |
| R3: (E,D) | K | U | U | K | K | U |
| R4:(D,A) | K | U | U | K | U | U |
| R4:(F,A) | K | U | U | U | U | K |
| R6:(B,C) | K | K | K | K | K | K |

Since all columns at R6 are known, the relation decomposition is lossless.

1. The set of relations we have at the beginning is simply the universal relation itself:

R={R1= (MedicineID, Ingredients, Uses, Warnings, Directions, OrderID, OrderDate, PatientID, TotalPrice, Address, City, State, ZipCode, PhoneNumber, MedicineQuantity)}

*FD1: MedicineID→Ingredients, Uses, Warnings, Directions*

*FD2: OrderID→OrderDate, PatientID, TotalPrice*

*FD3: PatientID→Address, City, State, ZipCode, PhoneNumber*

*FD4: MedicineID, OrderID→MedicineQuantity*

**Decomposition:**

R1 is not BCNF due to FD1

R2=(MedicineID, OrderID, OrderDate, PatientID, TotalPrice, Address, City, State, ZipCode, PhoneNumber, MedicineQuantity)

R3=(MedicineID, Ingredients, Uses, Warnings, Directions)

R= (R-R1) ∪ R2 ∪ R3 -> R={R2,R3}

R2 is not BCNF due to FD3

R4=(MedicineID, OrderID, OrderDate, PatientID, TotalPrice, MedicineQuantity)

R5=(PatientID, Address, City, State, ZipCode, PhoneNumber)

R= (R-R2) ∪ R4 ∪ R5 -> R={R3,R4,R5}

R4 is not BCNF due to FD2

R6=(MedicineID, OrderID, MedicineQuantity)

R7=(OrderID, OrderDate, PatientID, TotalPrice)

R=(R-R4) ∪ R6 ∪ R7 -> R={R3,R5,R6,R7}

**Final result:**

R3=(MedicineID, Ingredients, Uses, Warnings, Directions)

R5=(PatientID, Address, City, State, ZipCode, PhoneNumber)

R6=(MedicineID, OrderID, MedicineQuantity)

R7=(OrderID, OrderDate, PatientID, TotalPrice)

1. For each registered user, record his/her name, address, email, a unique login-name and pass-word.

•For each product put for auction, record its name, an (optional) description, one or several categories that it belongs to (e.g., ‘books-and-records’, ‘software’, ‘automobiles’, ‘appliances’,etc). Each product should have a unique auction-id.

•Keep track of information about a product for auction such as who is selling it, the minimum acceptable price, auction starting date and its status (i.e., ‘under auction’, ‘sold’, ‘withdrawn’).

•Keep track of every bid made by registered users, such as the bidder’s name, the date when the bid was made, and the amount of the bid, etc

•If a product was sold successfully, we want to know who bought the product with what bidding price, and when it was sold.

•For each product category, record its (unique) name. We want to organize the categories into a hierarchical structure such that one category can contain 0 or more subcategories.

(a):

Product

User

qz

Category

Sub-categories

Sub-categories

d

Bid