**(1)**

R(A,B,C,D,E) with FD’s AB ->C, DE -> C and B -> E

We need to first determine the key(s). Notice that A, B and D do not appear on the right side of any FD. Therefore, they **have got to be** part of a key. Let us see if {A,B,D} by itself could be the key.

Let’s find out the closure of {A,B,D} = {A,B,C,D,E}.

|  |  |  |
| --- | --- | --- |
| **Iteration** | **result** | **using** |
| 1 | ABD |  |
| 2 | ABCD | AB-->C |
| 3 | ABCD | B-->E |
| 4 | ABCDE | CD-->E |
| 5 | ABCDE |  |

{A,B,D} + ={ A,B,C,D,E}, Thus, indeed the only key is {A,B,D}.

Compute the closures of all subsets of R

**Closure of {AB} =**

{AB}+ = {ABCE}

**Closure of {DE}=**

{DE}+ ={DEC}

**Closure of B=**

{B}+ ={BE}

The FDs that are in violation of BCNF are all of them, since they do not have a super key on the left. In other words, the violating FDs are

AB ->C, DE -> C and B -> E and DB -> C (we got this additional one)

Let us try to decompose it by the first FD: AB->C.

We get {A,B,C},{A,B,D,E}.

In {A,B,C}, the only FD that holds in AB->C. Since AB is the key, there is no violation of BCNF.

In {A,B,D,E}, the FD B->E is in violation of BCNF. We decompose it further to obtain {B,E},{A,B,D}. The final relations are

R1{A,B,C}

R2{B,E}

R3{A,B,D}}

---------------------------------------------------------------------------------------------------------------------------

**(2)**

R(A,B,C,D) with FD’s A ->B, B -> C

Key is {A,D}

Let’s find out the closure of {A,D} = {A,B,C,D}.

Decompose {ABCD}

Closure of A {A} += {A,B,C}

Closure of B {B} + = {B,C}

Closure of C {C} + = {C}

Decompose using A -> BC

{A,B, C} ,{A,D}

Decompose {A,B,C} using A -> B

{A,B},{A,C}

The final decomposition is {A,D},{A,B},{A,C}

Decompose {A,B,C,D} using A -> B

{A,B} ,{A,C,D}

Decompose {A,C,D} using A -> C

{A,C} {A,D}

The final decomposition is {A,B},{A,C},{A,D}

**--------------------------------------------------------------------------------------------------------------------------**

**(3)**

Order(Order#, Order\_date, Customer#, Total\_amount)  
Order\_Item(Order#, Item#, Quantity\_ordered, Price\_each, Total\_price, Discount%)

The schema of Order X Order\_Item as follows  
T1(Order#,Item#,Order\_date, Customer#, Total\_amount, Quantity\_ordered, Total\_price, Discount%)

Key for the above schema is Order#,Item# and has functional dependencies as following  
Order#,Item# --> Quantity\_ordered  
Order#,Item# --> Total\_price  
Item# --> Price\_each  
Item# --> Discount%  
Order# --> Order\_date  
Order# --> Customer#  
Order# --> Total\_amount

Following is in the BCNF

Order(Order#, Customer#, Order\_date, Total\_amount)  -- Order# is only determinant  
Item(Item#, Price\_each, Discount%)    -- Item# is only determinant  
Order\_Item(Order#, Item#, Quantity\_ordered, Total\_price) -- Order#, Item# is only determinant (candidate key)

-------------------------------------------------------------------------------

Rx(Doctor#, Patient#, Date, Diagnosis, Treatment\_code, Charge)

Key for the above schema is Doctor#, Patient#, Date and has functional dependencies as following

Doctor#, Patient#, Date --> Diagnosis  
Doctor#, Patient#, Date --> Treatment\_code  
Treatment\_code --> Charge

Because there are no partial dependencies, the given relation is in 2NF already.  This however is not 3NF because the   
Charge is a non-key attribute that is determined by another non-key attribute, Treatment\_code. We must decompose further:

Following is the final decomposition.  
   
 T1(Doctor#, Patient#, Date, Diagnosis, Treatment\_code) -- Doctor#, Patient#, Date is only determinant (candidate key)  
 T2(Treatment\_code, Charge)    -- Treatment\_code is only determinant