Team 10
Ponnnurangam Kumaraguru
18 November 2022

Homework 4

Normalisation and Functional Dependencies

Aaryan Sharma	Prisha Arora	Vanshika Dhingra
2022121001	2021101075	2021101092

1 Part One

Consider the attribute set R = ABCDEGH and the FD set F = {AB → C, AC → B, AD → E, B → D, BC → A, E → G}.

1. For each of the following attribute sets, do the following: (i) Compute the set of dependencies that hold over the set and write down a minimal cover. (ii) Name the strongest normal form not violated by the relation containing these attributes. (iii) Decompose it into a collection of BCNF relations if it is not in BCNF.

(a) ABC (b) ABCD (c) ABCEG (d) DCEGH (e) ACEH

2. Which of the following decompositions of R = ABCDEG, with the same set of dependencies F, is

(a) dependency-preserving? (b) lossless-join?

(a) {AB, BC, ABDE, EG }

(b) {ABC, ACDE, ADG }

1. (a) i. $R_1 = ABC$: The FDs are $AB \rightarrow C$, $AC \rightarrow B$, $BC \rightarrow A$

   ii. This is already a minimal cover.

   iii. This is in BCNF since AB, AC and BC are candidate keys for R_1. (In

fact, these are all the candidate keys for R_1).

(b) i. $R_2 = ABCD$: The FDs are $AB \rightarrow C$, $AC \rightarrow B$, $BC \rightarrow A$, $B \rightarrow D$

ii. This is already a minimal cover.

iii. The keys are: AB, AC, BC. R_2 is not in BCNF or even 2NF because of the FD $B \rightarrow D$ (B is a proper subset of a key!) However, it is in 1NF. Decompose as in: ABC, BD. This is a BCNF decomposition.

(c) i. $R_3 = ABCEG$; The FDs are $AB \rightarrow C$, $AC \rightarrow B$, $BC \rightarrow A$, $E \rightarrow G$.

ii. This is already a minimal cover.

iii. The keys are: ABE, ACE, BCE. It is not even in 2NF since E is a proper subset of the keys, and there is an FD $E \rightarrow G$. It is in1NF. Decompose as in: ABE, ABC, EG. This is a BCNF decomposition.

(d) i. $R_4 = DCEGH$; The FD is $E \rightarrow G$.

ii. This is already a minimal cover.

iii. The keys are: $DCEH$. It is not in BCNF since, in the FD $E \rightarrow G$, E is a subset of the key and is not in 2NF either. It is in 1 NF Decompose as in: $DCEH$, EG

(e) i. $R_5 = ACEH$; No FDs exist.

ii. This is already a minimal cover.

iii. Key is $ACEH$ itself. It is in BCNF.

2. (a) The decomposition. $\{AB, BC, ABDE, EG\}$ is not lossless. To prove this consider the following instance of R:

$$\{(a_1, b, c_1, d_1, e_1, g_1), (a_2, b, c_2, d_2, e_2, g_2)\}$$

Because of the functional dependencies $BC \rightarrow A$ and $AB \rightarrow C$, $a_1 \neq a_2$ if and only if $c_1 \neq c_2$. It is easy to that the join $AB \bowtie BC$ contains 4 tuples:

$$\{(a_1, b, c_1), (a_1, b, c_2), (a_2, b, c_1), (a_2, b, c_2)\}$$

So the join of AB, BC, $ABDE$ and EG will contain at least 4 tuples, (actually it contains 8 tuples) so we have a lossy decomposition here.

This decomposition does not preserve the FD, $AB \rightarrow C$ (or $AC \rightarrow B$)

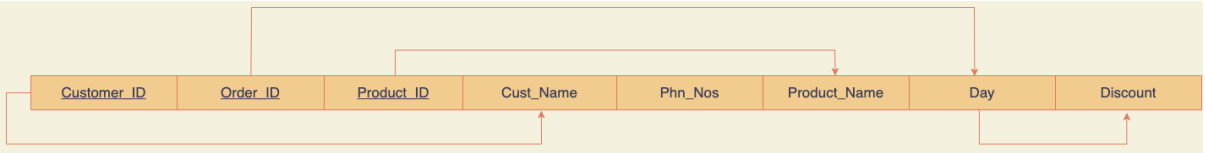
(b) The decomposition $\{ABC, ACDE, ADG\}$ is lossless. Intuitively, this is because the join of ABC, $ADCE$ and ADG can be constructed in two steps; first construct the join of ABC and $ACDE$: this is lossless because their (attribute) intersection is AC which is a key for

$ABCDE$ (in fact $ABCDEG$) so this is lossless. Now join this intermediate join with ADG. This is also lossless because the attribute intersection is AD and $AD \rightarrow ADG$. So by the test mentioned in the text, this step is also a lossless decomposition.

The projection of the FDs of R onto ABC gives us: $AB \rightarrow C$, $AC \rightarrow B$ and $BC \rightarrow A$. The projection of the FDs of R onto $ADCE$ gives us: $AD \rightarrow E$ and The projection of the FDs of R onto ADG gives us: $AD \rightarrow G$ (by transitivity) The closure of this set of dependencies does not contain $E \rightarrow G$ nor does it contain $B \rightarrow D$. So this decomposition is not dependency-preserving.

2 Part Two

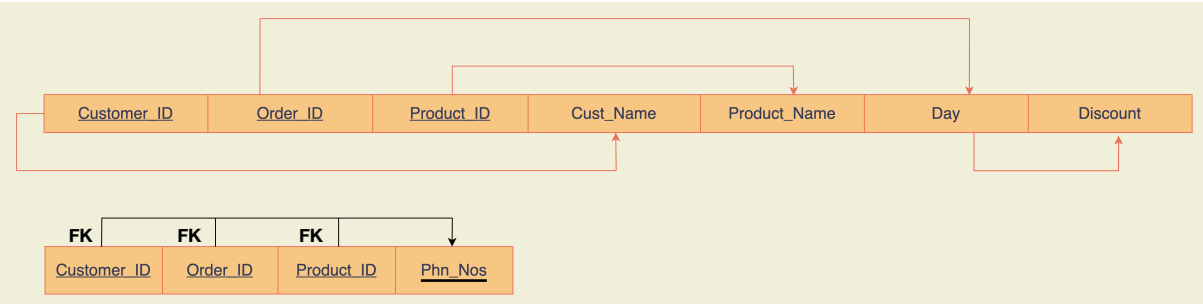
Given Schema:



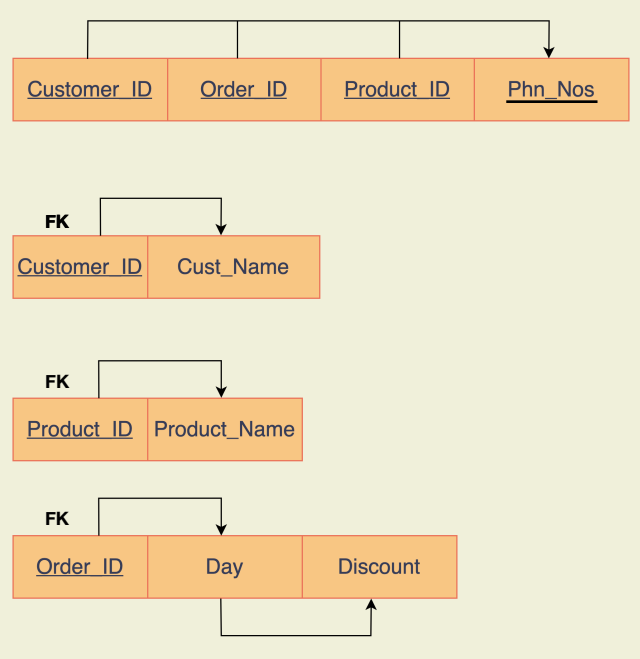
Functional Dependencies:

- Day → Discount
- Customer_ID →Cust_Name
- Product_ID → Product_Name
- Order_ID → Day

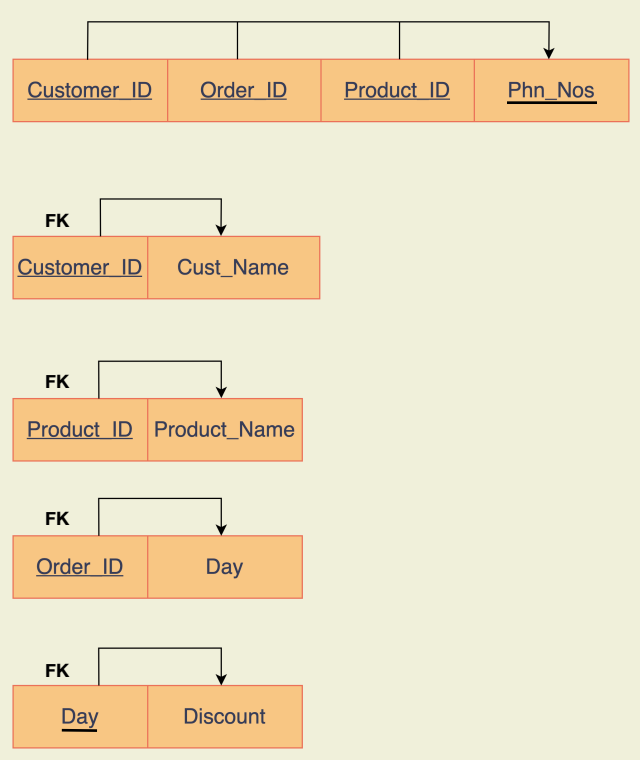
1NF : In this form, all the multivalued attributes are not stored in one table. Instead, a new table is created for multivalued attributes



2*NF* : In this form, all the partial dependencies are removed



3*NF* : In this form, all the transitive dependencies are removed



The name of the foreign key is the same as that of the primary key to which it refers. Thus not mentioned explicitly. Moreover, all the foreign keys together with the phone number make up the primary key of the first table.