HOMEWORK - 4

TEAM NUMBER: 12

TEAM MEMBERS:

1) Sriteja Reddy Pashya (2021111019)
2) Keval Jain (2021111030)
3) Romica Raisinghani (2021101053)

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 that is not violated by the relation containing these attributes.

(iii) Decompose it into a collection of BCNF relations if it is not in BCNF.

Answer: The following concepts are used in answering each of the questions above:

—> A **minimal cover** of a set of functional dependencies (FD) E is a minimal set of dependencies F that is equivalent to E. It must satisfy the below conditions:

- Every dependency in F has a single attribute in its right-hand side.
- We cannot replace any dependency X->A in F with a dependency Y->A, where Y is a proper subset of X, and still have a set of dependencies that is equivalent to F.
- We cannot remove any dependency from F and still have a set of dependencies that are equivalent to F.

—> In **1st Normal Form (1NF)**, all relations must be in atomic form. This means that an attribute of a table cannot hold multiple values.

—> In **2nd Normal Form (2NF)**, the relation must be in 1NF. Moreover, there should be no partial dependency i.e no proper subset of a candidate key should determine a non-prime attribute.

—> In **3rd Normal Form (3NF)**, the relation must be in 2NF. Moreover, there should be no transitive dependency for non-prime attributes i.e all non-prime attributes must depend only on prime attributes.

—> In **Boyce - Codd Normal Form (BCNF)**, the relation must be in 3NF. Moreover, for every functional dependency A->B, A has to be the super key of that particular relation.

a. ABC

i. The FD's are AB → C, BC → A, AC → B. This satisfies all the requirements of a minimal cover.

ii. Since all attributes are dependent only on superkeys, hence the strongest normal form that is not violated is Boyce - Codd Normal Form (BCNF).

iii. No changes are required as the given relations are already in BCNF form.

b. ABCD

i. The FD's are AB → C, BC → A, B → D, AC → B. This satisfies all the requirements of a minimal cover.
ii. Candidate keys are: AB, AC and BC. The FD B → D violates 2nd Normal Form since B is proper subset of candidate keys AB and BC. Hence the strongest normal form that is not violated is 1st Normal Form(1NF).
iii. We decompose ABCD to ABC and BD. This results in Boyce - Codd Normal Form (BCNF), as required.

c. ABCEG

i. The FD's are AB → C, AC → B, BC → A, E → G. This satisfies all the requirements of a minimal cover.
ii. Candidate keys are: ABE, ACE, and BCE. The FD E → G violates 2NF as E is proper subset of candidate keys. Hence the strongest normal form that is not violated is 1st Normal Form(1NF).

iii. We decompose ABCEG to ABE, ABC and EG. This decomposition gives us Boyce - Codd Normal Form (BCNF).

d. DCEGH

i. The FD is E → G. This is in minimal cover already.
ii. The key in this case is DCEH. The FD E → G violates 2NF as E is proper subset of candidate key. Hence the strongest normal form that is not violated is 1st Normal Form(1NF).
iii. Decompose DCEGH to DCEH, EG to make it into Boyce - Codd Normal Form (BCNF).

e. ACEH

i.No FDs exist. So no minimal cover.
ii. Since ACEH is the primary key and all functional dependencies depend on it, hence the strongest normal form that is not violated is Boyce - Codd Normal Form (BCNF).
iii. No changes are required as the given relations are already in BCNF form.

---

2. **Which of the following decompositions of R = ABCDEG, with the same set of dependencies F , is**
   (a) dependency-preserving?

   (b) lossless-join?

Answer: The decompositions R1, R2, R2…Rn for a relation schema R are said to be Lossless if their natural join results in the original relation R. Otherwise, if their natural join results into addition of extraneous tuples with the original relation R, then they are lossy. If the intersections of the decomposition forms a super key of the relation then the join is a lossless join, else it is a lossy join.

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

a) AB → C , AC → B and BC → A do not get preserved since ABC does appear together in a single decomposed relation. Hence it is **not dependency preserving**.

b) Since the intersections of the decompositions do not form a super key of the relations, hence it is **not a lossless-join** and is indeed a lossy join .

(b) {ABC, ACDE, ADG}

a) Since BD and EG do not appear together in any decomposed relation, hence $E \rightarrow G$ ans $B \rightarrow D$ are not preserved. Hence the decomposition is **not dependency preserving**.

b) If we first join ABC and ACDE, their intersection forms AC which is a key of ABC. If we now join this with ADG, the intersection is AD and AD is a key since $AD \rightarrow E$ and $E \rightarrow G$, hence $ADG \rightarrow G$. Since the intersections form the keys of the relation, hence the join **is a lossless join**.

PART TWO

Answers:

a) 1NF

- Because “Phn_Nos” is a multivalued attribute, we need to create a new table for it
- the primary key for the new table must include “Phn_Nos”

Purchases:

Customer_ID	Order_ID	Product_ID	Cust_Name	Product_Name	Day	Discount

Phone_Nos:

Customer_ID	Order_ID	Product_ID	Phn_Nos

---

b) 2NF

- The functional dependencies present are
  - Customer_ID —> Cust_Name
  - Order_ID —> Day, Discount
  - Product_ID —> Product_Name
- Therefore we must make new tables for each of them, with the respective attributes as primary keys

Customer:

Customer_ID	Cust_Name

Orders:

Order_ID	Day	Discount

Products:

Product_ID	Product_Name

Phone_Nos:

Customer_ID	Order_ID	Product_ID	Phn_Nos

Purchases:

Customer_ID	Order_ID	Product_ID

---

c) 3NF

- Order_ID —> Day and Day —> Discount is a transitive dependency,
- Therefore we must create a new table with “Day” as primary key

Customer:

Customer_ID	Cust_Name

Orders:

Order_ID	Day

Discounts:

Day	Discount

Products:

Product_ID	Product_Name

Phone_Nos:

Customer_ID	Order_ID	Product_ID	Phn_Nos

Purchases:

Customer_ID	Order_ID	Product_ID