Task 8

Lab Task 8: Normalizing Databases Using Functional Dependencies up to BCNF

Objective:

To normalize the database created in Task-2 using functional dependencies (FDs) and apply normalization techniques up to **BCNF (Boyce-Codd Normal Form)**.

1. Apply the Functional Dependency and Normalize to 1NF

Step 1: Identify Functional Dependencies (FDs)

Consider the following relations and FDs:

OrderTable (Order_ID, Cust_ID, Order_Date, Order_Total, Payment_Status)

- FD1: Order_ID → Cust_ID, Order_Date, Order_Total, Payment_Status

Customer (Cust_ID, Cust_Name, Cust_Contact, Cust_Email, Cust_Address)

- FD2: Cust_ID → Cust_Name, Cust_Contact, Cust_Email, Cust_Address

Menu_Item (Item_ID, Item_Name, Price, Category, Rest_ID)

- FD3: Item_ID → Item_Name, Price, Category, Rest_ID

Normalization to 1NF (First Normal Form)

- Ensure that each column contains only atomic (indivisible) values.
- Remove any repeating groups.
- Example:

Order_ID	Cust_ID	Order_Date	Order_Total	Payment_Status
1	1	2025-01-20	800	Paid
2	2	2025-01-21	500	Unpaid

2. Normalize the Relations Using FD+ and α+

- Compute **FD+ (Closure of FDs)** using Armstrong's Axioms.
- Identify minimal keys and remove redundant FDs.

Closure for OrderTable:

- FD+: { Order_ID → Cust_ID, Order_Date, Order_Total, Payment_Status }

Closure for Customer:

- FD+: { Cust_ID → Cust_Name, Cust_Contact, Cust_Email, Cust_Address }

Closure for Menu_Item:

- FD+: { Item_ID → Item_Name, Price, Category, Rest_ID }

3. Find the Minimal Cover and Canonical Cover

Minimal Cover:

- FD1: Order_ID → Cust_ID, Order_Date, Order_Total, Payment_Status
- FD2: Cust_ID → Cust_Name, Cust_Contact, Cust_Email, Cust_Address
- FD3: Item_ID → Item_Name, Price, Category, Rest_ID

Canonical Cover:

- No redundancy detected.

4. Normalize to 2NF

- A relation is in **2NF** if it is in 1NF and has no partial dependencies.
- Remove partial dependencies by creating separate relations.

Normalization to 2NF:

- **OrderTable (Order_ID, Order_Date, Order_Total, Payment_Status)**
- **Customer (Cust_ID, Cust_Name, Cust_Contact, Cust_Email, Cust_Address)**
- **Menu_Item (Item_ID, Item_Name, Price, Category, Rest_ID)**

5. Normalize to 3NF

- A relation is in **3NF** if it is in 2NF and has no transitive dependencies.
- Ensure non-prime attributes depend only on primary keys.

Normalization to 3NF:

- **Restaurant (Rest_ID, Rest_Name, Rest_Location, Rest_Contact)**
- **Menu_Item (Item_ID, Item_Name, Price, Category, Rest_ID)**

6. Normalize to BCNF

- A relation is in **BCNF** if, for every functional dependency (X → Y), X is a super key.
- Identify and remove transitive dependencies.

Normalization to BCNF:

- **OrderTable (Order_ID, Cust_ID, Order_Date, Order_Total, Payment_Status)**
- **Customer (Cust_ID, Cust_Name, Cust_Contact, Cust_Email, Cust_Address)**
- **Menu_Item (Item_ID, Item_Name, Price, Category, Rest_ID)**