Apply the Functional Dependency and Normalize to INF Step 1: Identify Functional Dependencies (FDZ).

Consider the following relations and FD2:

orderTable (order-ID, Cost_ID, order-Date, order_Total,
Payment_Status)

• FDI: order_ID -> cost_ID, order_Date, order_Total, Payme.
-nt_Status.

Customer (cost_ID, cust_Name, cust_Contact, cust_Email

· EDQ: ander Cust_ID -> cust_Name, cust_Contact, cust_Email,

Menu - Item (Item_ID, Item_Name, Price_Category, Rest_ID)

• FD3: Item_ID -> Item. Name, Price, Category, Rest _ID.

Normalization to INF (First Normal Form)

- · Ensure that each column contains only atomic values .
- · Remove any repeating groups.
- · Evample:-

Order_ID Cost_ID Order_Date Order_Total Payment_Status

1 1 2 2025-01-20 800 Paid
2 2 2025-01-21 500 unpaid.

TASK-8 - Normalizing Database Osing Fonctional Dependent

-cies up to BCNF

Objective :-

To normalize the database created in TABK-2 using of functional dependencies (FDS) and apply normalization.

2. Normalize the Relations using FD+ and ex+

· Compute FD+ (closure of FDs) using Armstrong's Accioms.

· Identify minimal Keys and vemove vedondant FOs.

Closure for order Table:

· FD+: Sorder_ID -> COST_ID , Order_ Date, order_Total, Payme--nt_ Status?.

Closure for costomer:

· 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 Over-

Minimal Cover:

· FD1: Order_ID __ Kost_ID, order_Date, order_Total, Payment_sta

· FD2 : Cost_ID -> cost_Name, cost_Contact, Cost_Email, cost_Address

Canonical Cover:

· No redundancy detected.

4. Normalize to ENF:

- A velation is in ENF if it is in INF and has no Partial dependencies.

- Remove Partial dependencies by Creating Separate velations.

Normalization to ENF:-

- ·Order Table (order_ID, order_Date, order_Total, Payment_Status)
- · Customa (Cust_ID, cust_Name, Cust_Contact, cust_Email, Cost_Address)
- · Flenu_ Item (Item_ID, Item_Name, Price, Category, Rest_ID).

5. Normalize to BCNF +

- · Arelation is in BCNF if, for every functional dependency
- ((e)), it is a Super Key.

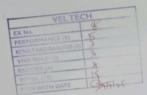
 Tidentify and remove transitive dependencies.

Normalization to BCNF:-

- · Ovder Table (order ID, cost ID, order Date, Order Table, Payment
- · Costomer (Cost_ID, cost_Name, cost_contact, cost_Email,
- · Menu_Item (Item_ID, Item_Name, Price, Category, Rest ID)

6. Normalize to BNF:

- · A relation is in SNF if it is in SNF and has no transitive dependencies.
- · Ensure non-Prime attributes depend only on Primary Kogs.
- · Restaurant (Rest ID, Rest-Name, Rest-location, Rest-Contact).



Result: Hence, Normalizing Database using functional dependences up to BONF done successfully.