CSCI 5408

DATA MANAGEMENT AND WAREHOUSING

LAB - 2

Banner ID: B00984406

Table of Contents

1: Normalize INVOICE table to 1NF

Current Invoice Table: -

Table 1: Provided Invoice Table

InvoiceID	Date	Branch	Branch Location	City	Customer Type	Customer Name	Gender	Product	UnitPrice	Quantity
1101	2/5/2024	1,004, 008	Quinpool, Windsor, Dartmouth	Halifax	Member, Normal	Alex, Rohan, Mark	Female, Male	Phone, Battery, Cover	200,50 ,25	1,2,5
2104	3/3/2024	3,007, 008	Dowry Street, Abbey Lane, GlenView	Toronto	Member, Normal	Suzan, Carla, Andreq	Female, Male	Code, Power bank	10,35	20,45

Assumption:

- In this I assume that the branch code is unique only within a specific city, which means that it is possible for the same branch code to be assigned to multiple stores located in different cities. For instance, branch code 008 is used for both Dartmouth in Halifax and GlenView in Toronto.
- Furthermore, I have made the decision to assign the female gender to Alex, while Rohan and Mark are identified as males. Similarly, Suzan is also female, whereas Carla and Andreq are males.
- Alex and Suzan are considered members, while Rohan, Mark, Carla, and Andrew are classified as regular customers.

Explanation:

- For normalizing in 1st normal form, there should not be any group values. Here we can see that many columns have multiple values so, we divide that column.
- The combination of InvoiceId and product serves as a unique identifier for each row in the 1NF.

After the 1st Normal Form:

Table 2: Invoice Table 1st Normal Form

InvoiceID	Date	Branch	Branch Location	City	Customer Type	Customer Name	Gender	Product	UnitPrice	Quantity
1101	2/5/2024	1	Quinpool	Halifax	Member	Alex	Female	Phone	200	1
1101	2/5/2024	004	Windsor	Halifax	Normal	Rohan	Male	Battery	50	2
1101	2/5/2024	008	Dartmouth	Halifax	Normal	Mark	Male	Cover	25	5
2104	3/3/2024	3	Dowry Street	Toronto	Member	Suzan	Female	Code	10	20

2104	3/3/2024	007	Abbey Lane	Toronto	Normal	Carla	Male	Power bank	35	45
2104	3/3/2024	008	GlenView	Toronto	Normal	Andreq	Male	Power bank	35	45

Dependency Diagram:

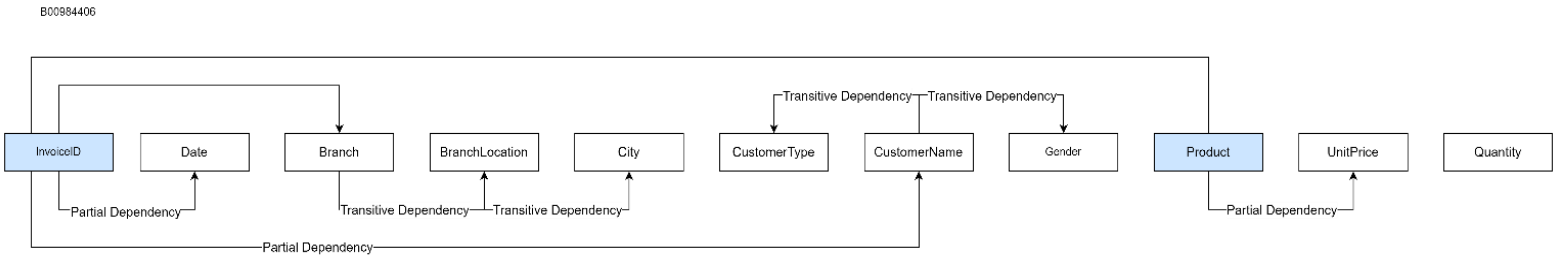


Figure 1: Dependency diagram after the 1st Normal Form

2: Normalize INVOICE table to 1NF

Explanation:

- A table is said to be in second normal form (2NF) when there are no partial dependencies present within the table.
- Figure 1 demonstrates that Branch, a non-prime attribute, is connected to a subset of the primary key, InvoiceID. Similarly, CustomerName is associated with InvoiceID, and UnitPrice is dependent on Product.
- So, in order to get the 2nd Normal form table we need to remove that partial dependency by dividing the table and we get the 4 tables as a result.

After the 2nd Normal Form:

Table 3: Invoice Table 2nd Normal Form

InvoiceID	Date	Product	Quantity
1101	2/5/2024	Phone	1
1101	2/5/2024	Battery	2
1101	2/5/2024	Cover	5
2104	3/3/2024	Code	20
2104	3/3/2024	PowerBank	45

Table 4: Branch Table

InvoiceID	Branch	BranchLocation	City
1101	1	Quinpool	Halifax
1101	004	Windsor	Halifax
1101	008	Dartmouth	Halifax
2104	3	Dowry Street	Toronto
2104	007	Abbey Lane	Toronto
2104	008	GlenView	Toronto

Table 5: Customer Table

InvoiceID	CustomerName	CutomerType	Gender
1101	Alex	Member	Female
1101	Rohan	Normal	Male
1101	Mark	Normal	Male
2104	Suzan	Member	Female
2104	Carla	Normal	Male
2104	Andreq	Normal	Male

Table 6: Product Table

Product	Unit Price
Phone	200
Battery	50
Cover	25

Code	10
PowerBank	35

Dependency Diagram:

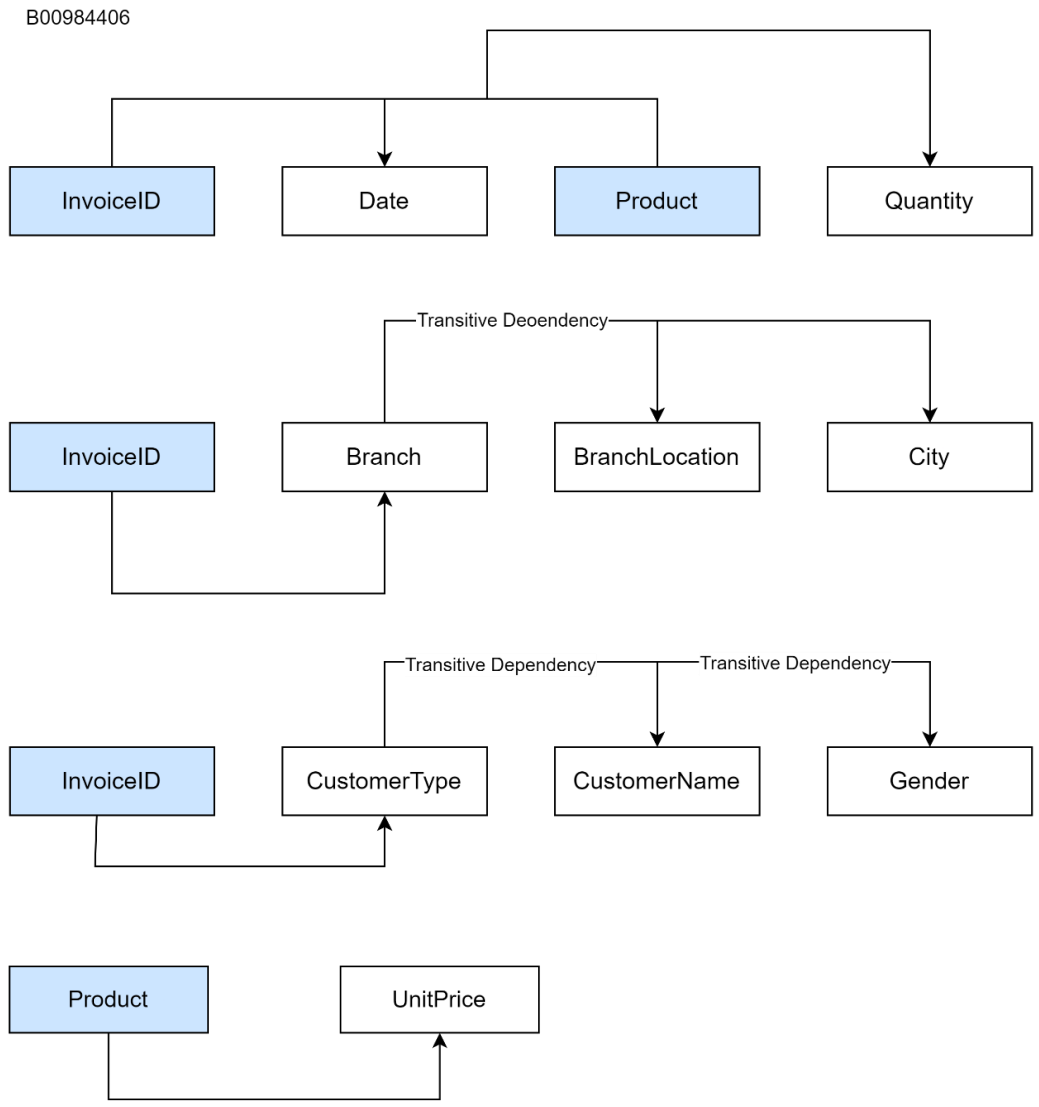


Figure 2: Dependency diagram after the 2nd Normal Form

3: Normalize INVOICE table to 3NF

Explanation:

- A table is said to be in third normal form (3NF) when there are no transitive dependencies present within the table.
- Figure 2 demonstrates that BranchLocation is determined by the combination of the Branch and City variables. Additionally, the Customer Type and Gender variables are found to be dependent on the CustomerName.
- So, to get the 3rd Normal Form table, we create the separate and remove the transitive dependency.

After the 3rd Normal Form:

Table 7: Invoice Table 3rd Normal Form

InvoiceID	Date	Product	Quantity
1101	2/5/2024	Phone	1
1101	2/5/2024	Battery	2
1101	2/5/2024	Cover	5
2104	3/3/2024	Code	20
2104	3/3/2024	PowerBank	45

Table 8: Branch Table

InvoiceID	Branch	City
1101	1	Halifax
1101	004	Halifax
1101	008	Halifax
2104	3	Toronto
2104	007	Toronto
2104	008	Toronto

Table 9: Invoice_Branch Table

Branch	BranchLocation	City
1	Quinpool	Halifax
004	Windsor	Halifax
008	Dartmouth	Halifax
3	Dowry Street	Toronto
007	Abbey Lane	Toronto
008	GlenView	Toronto

Table 10: Customer Table

InvoiceId	CustomerName
1101	Alex
1101	Rohan

1101	Mark
2104	Suzan
2104	Carla
2104	Andreq

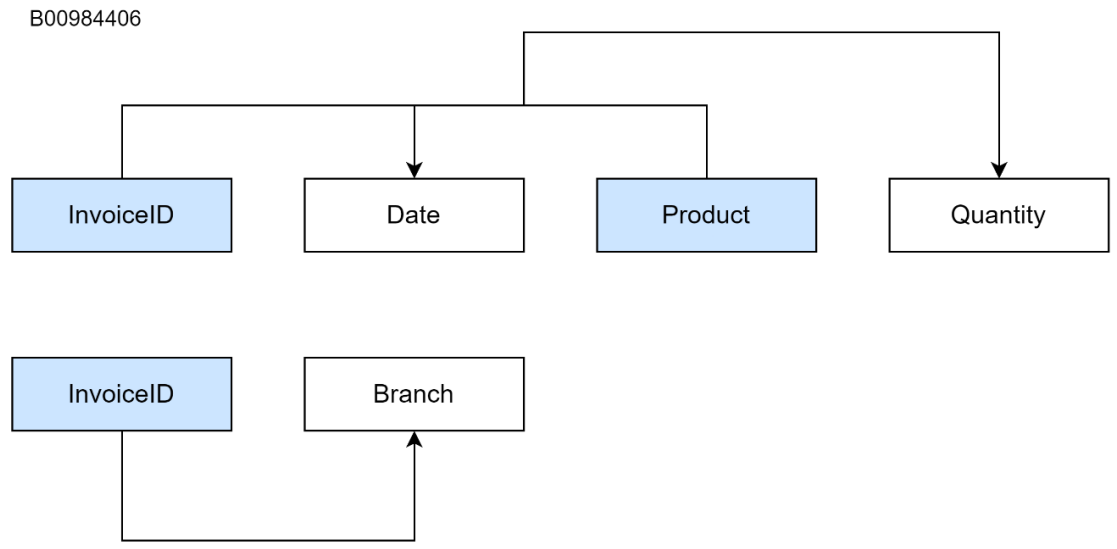
Table 11: Invoice_customer Table

CustomerName	CustomerType	Gender
Alex	Member	Female
Rohan	Normal	Male
Mark	Normal	Male
Suzan	Member	Female
Carla	Normal	Male
Andreq	Normal	Male

Table 12: Product Table

Product	Unit Proce
Phone	200
Battery	50
Cover	25
Code	10
PowerBank	35

Dependency Diagram:



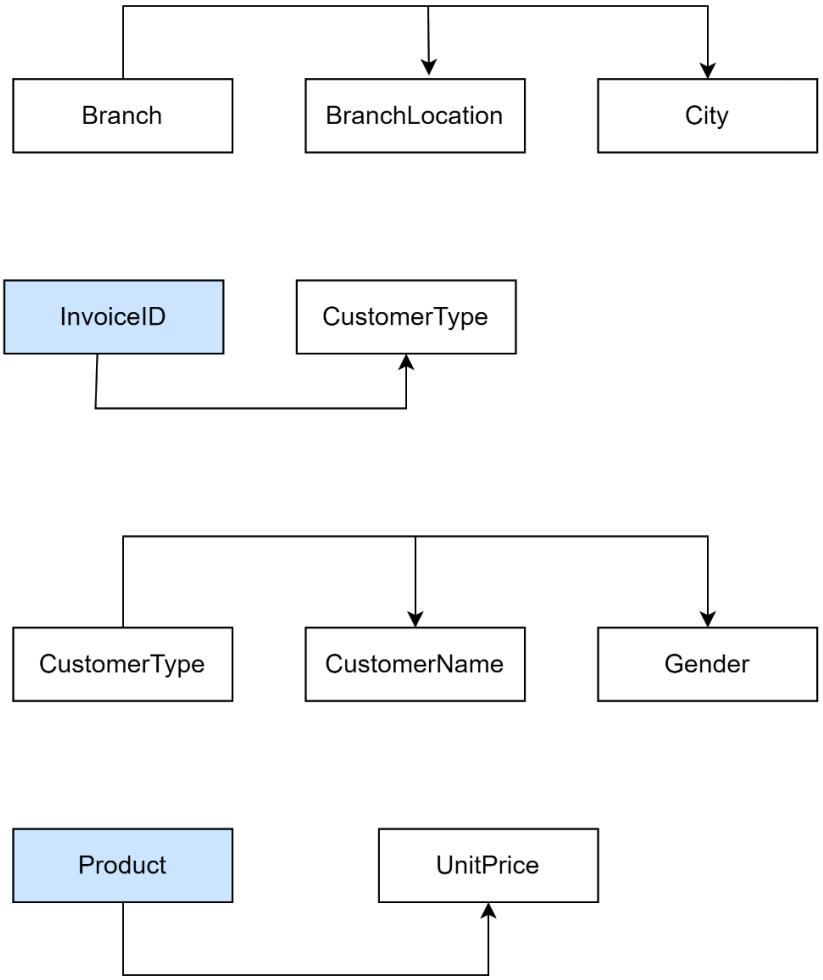


Figure 3: Dependency diagram after the 3rd Normal Form

4: Question

Do you think it will be a good design decision to de-normalize the normal forms generated for the Product table in the above steps? Why/Why not? Provide a brief explanation.

Ans:

- In my opinion, it relies on the need which kind of operations we are going to carry out on the generated tables. So, if we are in process of de normalization then for frequent read and writes queries, there would be a lot of joins for product invoice and all. Hence, degrading system's performance.
- On the contrary, normalized structures are more appropriate for cases where the main use is for frequent read/update/insert/delete instructions then tables must be in 3NF form for better performance.
- Therefore, in the above table, quantity of a transaction varies when multiple transactions happen since then it is preferable to normalize the table. Due to this, the stock of the table needs to be updated immediately after the transaction is completed.