SSN COLLEGE OF ENGINEERING

KALAVAKKAM – 603110

UCS1404 - DATABASE MANAGEMENT SYSTEM PROJECT

TITLE : MEESHO - ONLINE TEXTILE INDUSTRY

PROJECT MEMBERS :

B.E CSE,2nd Year,

1. 205001050 - JENISHDA PUSHPA MARY S
2. 205001301 - AAFREEN M
3. 205001304 - DIVYA M

SUBMITTED TO :

Dr. P.MIRUNALINI

ASSOCIATE PROFESSOR

CSE DEPARTMENT

PROBLEM STATEMENT :

Now a days Customers are hard time for shopping their clothes in a single platform and there is a lack of availability of all popular brands in a stores.

AIM :

This project aims at to monitor all process in the single platform, like order receiving, delivery etc. As these works are done manually at the stores at present it takes a lot of time to complete the work. So, the basic aim of the project is to develop a system, which is very simple, user friendly, easy retrieval and simple access.

PROJECT OVERVIEW :

The main objective of the Online Textile Industry is to reduce the manual works, easy to access wide range of products and various brands, increase the processing speed and ensure reliability of data. And also provides attractive offers like discounts, gifts etc. Moreover , International products can also be distributed around the globe easily through the Online Textile Industry.

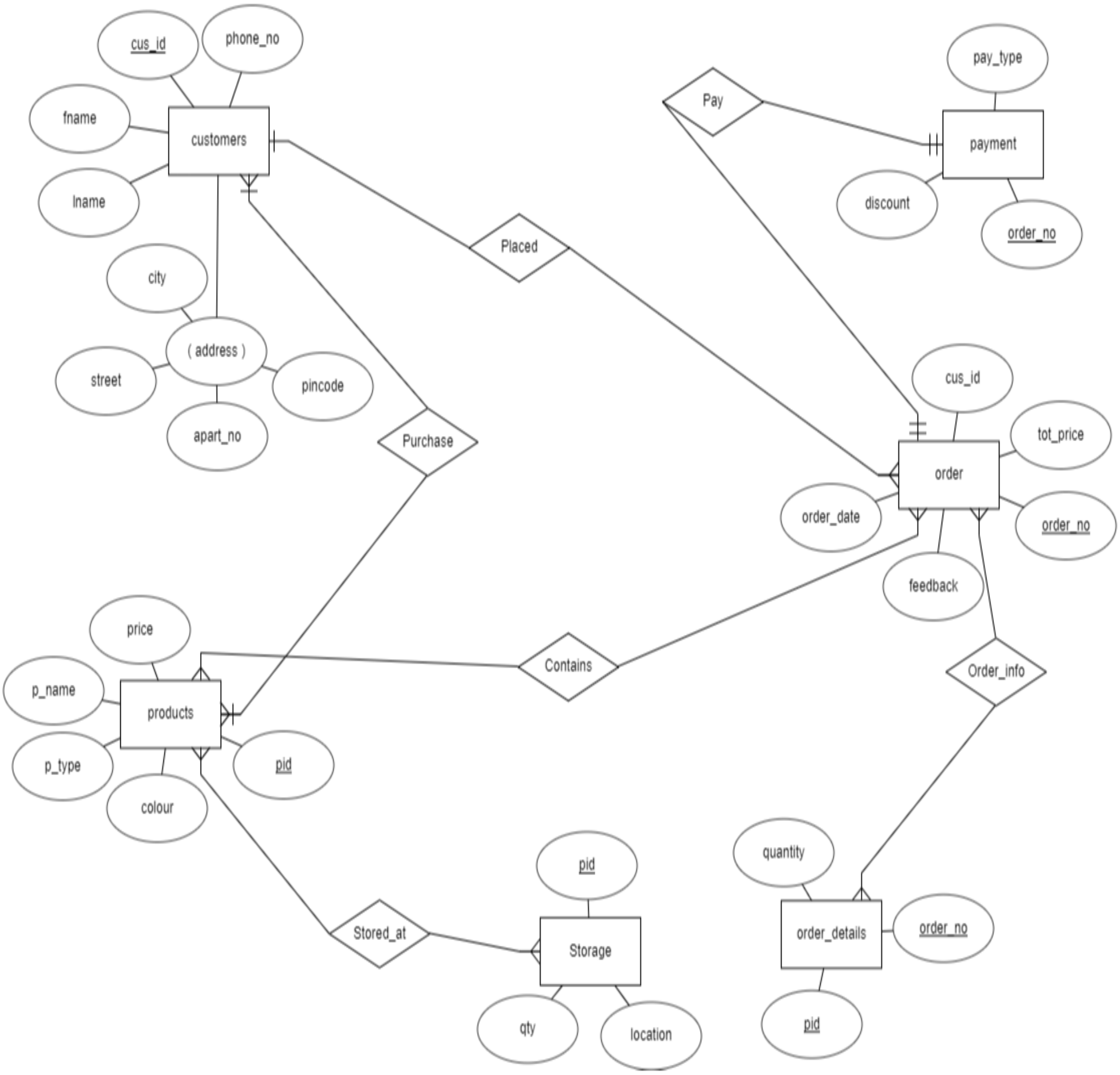
This project consists of the following entities :

- Customers
- Products
- Storage
- Order
- Order_details
- Payment

MAIN FEATURES :

- Easy to store and manage details of all the products in a stock.
- Handles the delivery details very accurately.
- Maintaining record of Customers.
- Reduce the errors caused by manual processing of data.
- Reduce the chance of stealing the information.
- Stores feedback of the customers and other information.
- Monitor the details keep track of dates and address throughout the project.

ER DIAGRAM :


cus_id
phone_no
fname
customers
lname
city
(address)
street
pincode
apart_no
Placed
Purchase
Pay
pay_type
payment
discount
order_no
cus_id
tot_price
order
order_date
order_no
feedback
Order_info
price
p_name
products
p_type
pid
colour
Contains
Stored_at
pid
Storage
qty
location
quantity
order_details
order_no
pid

ER TO RELATIONAL MAPPING :

Simple attributes : ER attributes map directly onto the relation

So, The relation should be like this :

customer

cus_id	fname	lname	phone_no	address

products

pid	p_name	p_type	colour	price

storage

pid	qty	location

order

order_no	cus_id	tot_price	ordrer_date	feedback

order_details

order_no	pid	quantity

payment

order_no	pay_type	discount

1:1 Relationship :

For each binary 1:1 relationship type R in the ER schema, identify the relations S and T that correspond to the entity types participating in R.

1) **Foreign Key approach :** Choose one of the relations say S and include a foreign key in S the primary key of T. It is better to choose an entity type with total participation in R in the role of S.

2) 1:1 relation PAY is mapped by choosing the participating entity type ORDER to serve in the role of S.

1:N Relationship :

1)For each regular binary 1:N relationship type R, identify the relation S that represent the participating entity type at the N-side of the relationship type.

2)Include as foreign key in S(N side) the primary key of the relation (1 side) T that represents the other entity type participating in R.

3)Include any simple attributes of the 1:N relation type as attributes of S.

1:N relationship types PLACED in the figure. For CUSTOMER we include the primary key CUS_ID of the ORDER relation as foreign key in the CUSTOMER_PLACED relation and call it CUS_ID.

M:N Realtionship :

1) For each regular binary M:N relationship type R, create a new relation S to represent R.

2) Include as foreign key attributes in S the primary keys of the relations that represent the participating entity types; their combination will form the primary key of S.

3)Also include any simple attributes of the M:N relationship type (or simple components of composite attributes) as attributes of S.

- The M:N relationship type PURCHASE from the ER diagram is mapped by creating a relation PURCHASE in the relational database schema.The primary keys of the CUSTOMER and PRODUCTS relations are included as foreign keys in PURCHASE and renamed CNO and PNO, respectively.

Purchase

cno	pn

- The M:N relationship type CONTAINS from the ER diagram is mapped by creating a relation CONTAINS in the relational database schema.The primary keys of the PRODUCTS and ORDER relations are included as foreign keys in CONTAINS and renamed PROD_ID and ONO, respectively.

contains

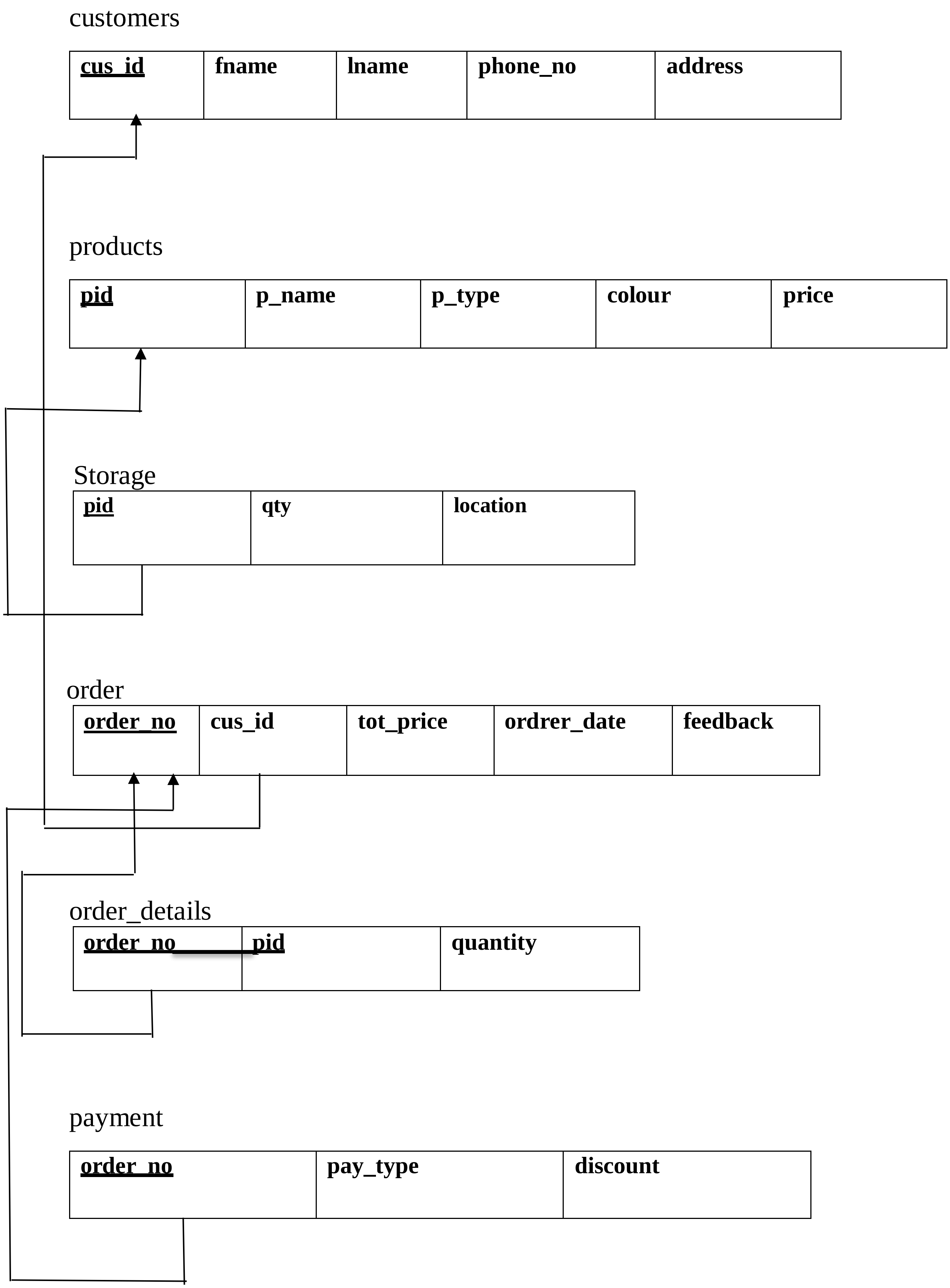
prod_id	ono

- The M:N relationship type ORDER_INFO from the ER diagram is mapped by creating a relation ORDER_INFO in the relational database schema.The primary keys of the ORDER_DETAILS and ORDER relations are included as foreign keys in ORDER_INFO and renamed CUSTOMER_ID and O_NO, respectively.

order_info

customer_id	o_no

ER TO RELATIONAL SCHEMA :



customers

cus_id	**fname**	**lname**	**phone_no**	**address**

products

pid	**p_name**	**p_type**	**colour**	**price**

Storage

pid	**qty**	**location**

order

order_no	**cus_id**	**tot_price**	**ordrer_date**	**feedback**

order_details

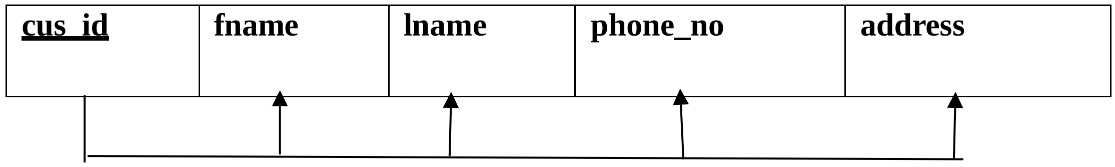
order_no	**pid**	**quantity**

payment

order_no	**pay_type**	**discount**

FUNCTIONAL DEPENDENCY AND NORMALIZATION :

CUSTOMER TABLE :



cus_id	fname	lname	phone_no	address

cus_id -> fname, lname, phone_no, address

customer

cus_id	fname	lname	phone_no	address

CHECKING FOR 1NF :

Conditions :

- No multivalued or composite attributes allowed.
- No nested relational allowed.
- If not decompose the table.

Here address is a composite attribute. Therefore its decomposed table as below:

cus_id	fname	lname	phone_no	apart_no	street	pincode	city

Table is in 1NF.

CHECKING FOR 2NF :

Conditions :

- The first condition for the table to be in 2NF is that the table has to be in1NF.
- The table should not possess partial dependency.
- The partial dependency here means the proper subset of the candidate key should give a non-prime attribute.

Every non-prime attribute A should be fully functionally dependent on the primary key of customer relation.

There are no partial dependencies in the customer table.

Table is in 2NF.

CHECKING FOR 3NF :

Condition :

- The first condition for the table to be in 3NF is that the table should be in the 2NF.
- The second condition is that there should be no transitive dependency for non-prime attributes, which indicates that non-prime attributes should not depend on other non-prime attributes in a table.

No non-prime attribute A should be transitively dependent on the primary key of customer relation.

Table is in 3NF.

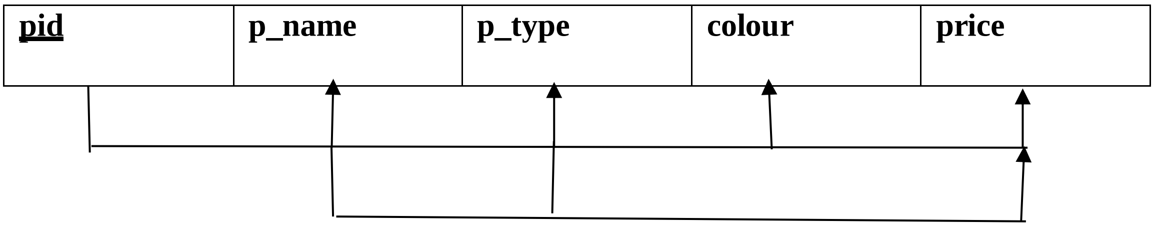
CHECKING FOR BCNF :

Condition :

- The first condition for the table to be in BCNF is that the table should be in the3NF.
- If an FD X → Y holds in relation R , then X must be a super key of R.

Table is in BCNF.

PRODUCT TABLE :



pid	**p_name**	**p_type**	**colour**	**price**

pid -> p_name, p_type, colour, price

p_name, p_type -> price

CHECKING FOR 1NF :

Conditions :

- No multivalued or composite attributes allowed.
- No nested relational allowed.
- If not decompose the table.

products

pid	p_name	p_type	colour	price

Here no multivalued or composite attributes and nested relational are there.

Table is in 1NF.

CHECKING FOR 2NF :

Conditions :

- The first condition for the table to be in 2NF is that the table has to be in1NF.
- The table should not possess partial dependency.
- The partial dependency here means the proper subset of the candidate key should give a non-prime attribute.

Every non-prime attribute A should be fully functionally dependent on the primary key of products relation.

There are no partial dependencies in the products table.

Table is in 2NF.

CHECKING FOR 3NF :

Condition :

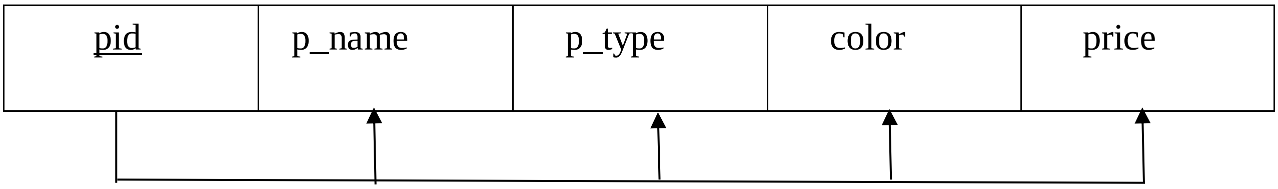
- The first condition for the table to be in 3NF is that the table should be in the 2NF.
- The second condition is that there should be no transitive dependency for non-prime attributes, which indicates that non-prime attributes should not depend on other non-prime attributes in a table.

No non-prime attribute A should be transitively dependent on the primary key of products relation.

Here, consider the FD’s pid -> p_name, p_type and p_name, p_type -> price.There should be a transitive dependency.

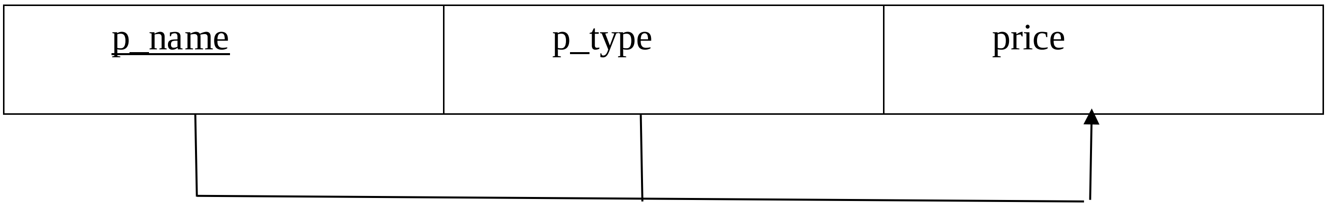
So, table should be decomposed as follows:

R1 :



pid	p_name	p_type	color	price

R2 :



p_name	p_type	price

Table is in 3NF.

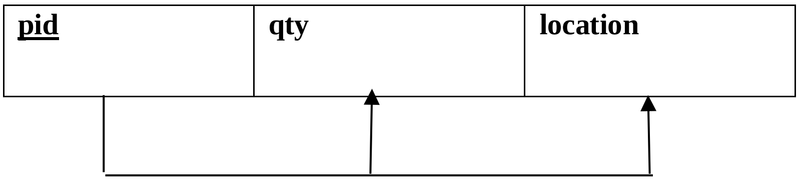
CHECKING FOR BCNF :

Condition :

- The first condition for the table to be in BCNF is that the table should be in the3NF.
- If an FD X → Y holds in relation R , then X must be a super key of R.

Table is in BCNF

STORAGE TABLE :



pid	**qty**	**location**

pid -> qty, location

CHECKING FOR 1NF :

Conditions :

- No multivalued or composite attributes allowed.
- No nested relational allowed.
- If not decompose the table.

storage

<u>**pid**</u>	**qty**	**location**

Here no multivalued or composite attributes and nested relational are there.

Table is in 1NF.

CHECKING FOR 2NF :

Conditions :

- The first condition for the table to be in 2NF is that the table has to be in1NF.
- The table should not possess partial dependency.
- The partial dependency here means the proper subset of the candidate key should give a non-prime attribute.

Every non-prime attribute A should be fully functionally dependent on the primary key of storage relation.

There are no partial dependencies in the storage table.

Table is in 2NF.

CHECKING FOR 3NF :

Condition :

- The first condition for the table to be in 3NF is that the table should be in the 2NF.
- The second condition is that there should be no transitive dependency for non-prime attributes, which indicates that non-prime attributes should not depend on other non-prime attributes in a table.

No non-prime attribute A should be transitively dependent on the primary key of storage relation.

Table is in 3NF.

CHECKING FOR BCNF :

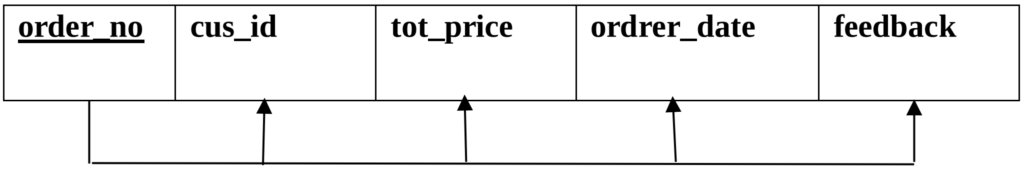
Condition :

- The first condition for the table to be in BCNF is that the table should be in the3NF.

- If an FD X → Y holds in relation R , then X must be a super key of R.

Table is in BCNF.

ORDER TABLE :



order_no	**cus_id**	**tot_price**	**ordrer_date**	**feedback**

order_no -> cus_id, tot_price, order_rate, feedback

CHECKING FOR 1NF :

Conditions :

- No multivalued or composite attributes allowed.
- No nested relational allowed.
- If not decompose the table.

order

order_no	**cus_id**	**tot_price**	**ordrer_date**	**feedback**

Here no multivalued or composite attributes and nested relational are there.

Table is in 1NF.

CHECKING FOR 2NF :

Conditions :

- The first condition for the table to be in 2NF is that the table has to be in1NF.

- The table should not possess partial dependency.

- The partial dependency here means the proper subset of the candidate key should give a non-prime attribute.

Every non-prime attribute A should be fully functionally dependent on the primary key of order relation.

There are no partial dependencies in the order table.

Table is in 2NF.

CHECKING FOR 3NF :

Condition :

- The first condition for the table to be in 3NF is that the table should be in the 2NF.

- The second condition is that there should be no transitive dependency for non-prime attributes, which indicates that non-prime attributes should not depend on other non-prime attributes in a table.

No non-prime attribute A should be transitively dependent on the primary key of order relation.

Table is in 3NF.

CHECKING FOR BCNF :

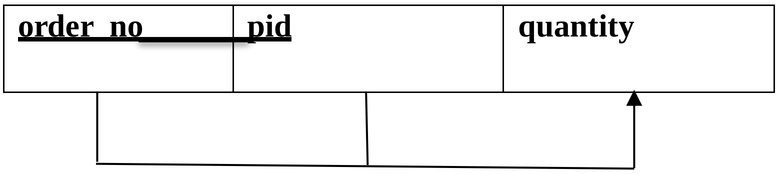
Condition :

- The first condition for the table to be in BCNF is that the table should be in the3NF.

- If an FD X → Y holds in relation R , then X must be a super key of R.

Table is in BCNF

ORDER_DETAILS TABLE :



order_no	**pid**	**quantity**

order_no, pid -> quantity

CHECKING FOR 1NF :

Conditions :

- No multivalued or composite attributes allowed.
- No nested relational allowed.
- If not decompose the table.

order_details

order_no	pid	quantity

Here no multivalued or composite attributes and nested relational are there.

Table is in 1NF.

CHECKING FOR 2NF :

Conditions :

- The first condition for the table to be in 2NF is that the table has to be in1NF.
- The table should not possess partial dependency.
- The partial dependency here means the proper subset of the candidate key should give a non-prime attribute.

Every non-prime attribute A should be fully functionally dependent on the primary key of order_details relation.

Here , order_no, pid-> quantity, tot_price is a Full FD since neither order_no -> quantity nor pid -> quantity holds.

There are no partial dependencies in the order_details table.

Table is in 2NF.

CHECKING FOR 3NF :

Condition :

- ➢ The first condition for the table to be in 3NF is that the table should be in the 2NF.
- ➢ The second condition is that there should be no transitive dependency for non-prime attributes, which indicates that non-prime attributes should not depend on other non-prime attributes in a table.

No non-prime attribute A should be transitively dependent on the primary key of order_details relation.

Table is in 3NF.

CHECKING FOR BCNF :

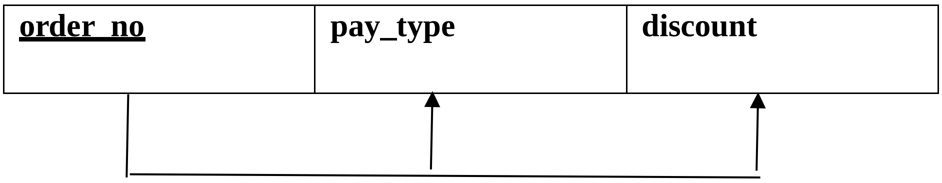
Condition :

- ➢ The first condition for the table to be in BCNF is that the table should be in the3NF.

- ➢ If an FD X → Y holds in relation R , then X must be a super key of R.

Table is in BCNF

PAYMENT TABLE :



order_no	**pay_type**	**discount**

order_no -> pay_type, discount

CHECKING FOR 1NF :

Conditions :

- ➢ No multivalued or composite attributes allowed.
- ➢ No nested relational allowed.
- ➢ If not decompose the table.

payment

order_no	**pay_type**	**discount**

Here no multivalued or composite attributes and nested relational are there.

Table is in 1NF.

CHECKING FOR 2NF :

Conditions :

- The first condition for the table to be in 2NF is that the table has to be in1NF.
- The table should not possess partial dependency.
- The partial dependency here means the proper subset of the candidate key should give a non-prime attribute.

Every non-prime attribute A should be fully functionally dependent on the primary key of payment relation.

There are no partial dependencies in the payment table.

Table is in 2NF.

CHECKING FOR 3NF :

Condition :

- The first condition for the table to be in 3NF is that the table should be in the 2NF.
- The second condition is that there should be no transitive dependency for non-prime attributes, which indicates that non-prime attributes should not depend on other non-prime attributes in a table.

No non-prime attribute A should be transitively dependent on the primary key of payment relation.

Table is in 3NF.

CHECKING FOR BCNF :

Condition :

- The first condition for the table to be in BCNF is that the table should be in the3NF.

- Table is in BCNF.
