DBMS Project
Supermarket

Team E

Arpit Singh (111601031)
Pankaj Kumar (111601014)
Sparsh Jain (111601026)

Contribution:

1. Arpit Singh: Database Design including ERD designing, Remove constraints to maintain atomicity while keeping the integrity of the database constant, decide roles, create views for ease of access, design triggers, fine tuning the front end for specific roles

2. Sparsh Jain: State basic requirements of the database to start the design, Schema Development, grant privileges required to for each role (including dependencies due to triggers), write functions/procedures for ease of use, testing of constraint fulfillment by designing tricky test cases

3. Pankaj Kumar: Proof checking of requirements and ERD, Insert Values to visualise the effects of database, BCNF checking, major front end development involving creation of various pages in visually appealing format

Requirements:

Who are the users of this system?

1. Admin
2. Managers
3. Cashiers
4. Non-Login Users (Customers, Other Employees)

Roles:

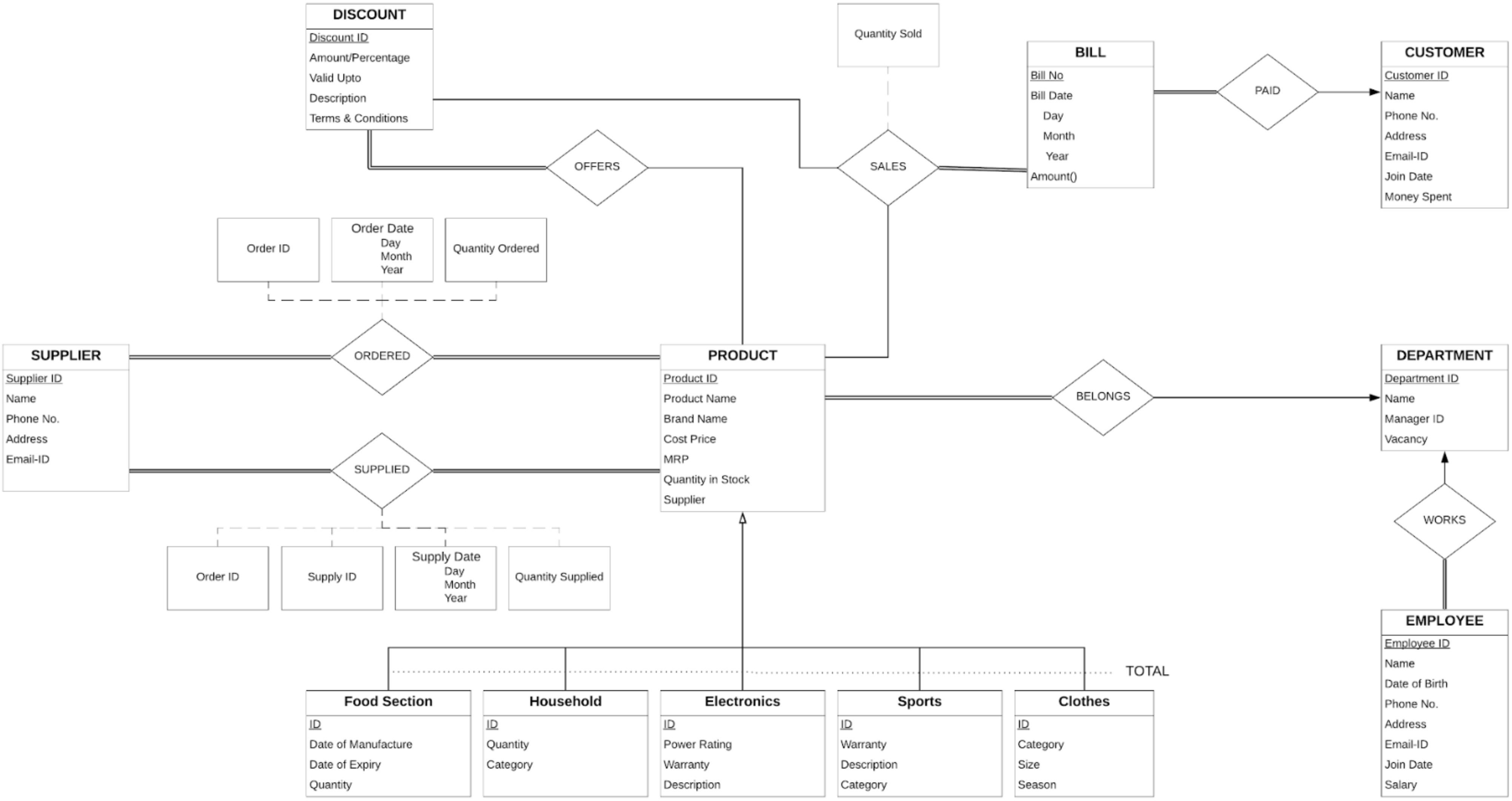
1. Admin: Can Access and Modify all entries of the database.
2. Managers: Can Hire/Fire Employees of his own department, Update vacancies in his department, Update Employee details of his department, Can add/remove/modify/order a product from its suppliers, and update the supply table when product is supplied. Can create a new discount or modify an existing one. Has read access to all the tables and Modify access to only tables required to do the above tasks for his own department only.
3. Cashiers: Can create a new bill, add/remove items from a bill, may apply discount if applicable, has read access to products, discounts, and customer details. Can also add/modify the details of a customer. On customer request, he can also produce details of a bill.
4. Non-Login Users can check details of a product and applicable discounts.

Update the stock of the product automatically when an order is supplied.

The order can be fulfilled through multiple supplies, and multiple orders can be fulfilled through a single supply.

Update the stock of the product automatically when it is added/removed from a bill.

An interface for the admin and managers (for his own department) to perform few important combination of queries such as profit/loss in a given time period, etc.


DISCOUNT
Discount ID
Amount/Percentage
Valid Upto
Description
Terms & Conditions
Quantity Sold
BILL
Bill No
Bill Date
Day
Month
Year
Amount()
PAID
CUSTOMER
Customer ID
Name
Phone No.
Address
Email-ID
Join Date
Money Spent
OFFERS
SALES
Order ID
Order Date
Day
Month
Year
Quantity Ordered
SUPPLIER
Supplier ID
Name
Phone No.
Address
Email-ID
ORDERED
PRODUCT
Product ID
Product Name
Brand Name
Cost Price
MRP
Quantity in Stock
Supplier
BELONGS
DEPARTMENT
Department ID
Name
Manager ID
Vacancy
SUPPLIED
WORKS
Order ID
Supply ID
Supply Date
Day
Month
Year
Quantity Supplied
EMPLOYEE
Employee ID
Name
Date of Birth
Phone No.
Address
Email-ID
Join Date
Salary
TOTAL
Food Section
ID
Date of Manufacture
Date of Expiry
Quantity
Household
ID
Quantity
Category
Electronics
ID
Power Rating
Warranty
Description
Sports
ID
Warranty
Description
Category
Clothes
ID
Category
Size
Season

View details

1. Create views for employees of different department to see the details of products of their respective departments (5 x {department}Details) = 5 views
2. Create view for customer to see relevant details of products of all departments (5 x view{department} + 1 viewProduct) = 6 views
3. Create a view viewDiscount to see all valid discounts applicable across products
4. Create views for manager to update product/employee details in their own departments (5 x [product/employee]{department} + employeeCashier) = 11 views
   Total - 23 views

Roles

1. Admin role (all privileges)
2. Individual managers of 5 product department + cashier
3. Customer
4. Cashier
5. Employees of each department

Privileges

1. All privileges to admin with grant option

2. Select on bill to all managers
3. Select on customer to all managers
4. All privileges on discount to all managers
5. All privileges on offers to all managers
6. Select on product to all managers
7. All privileges on ordered to all managers
8. Select on department to all managers
9. Update only vacancy in department to all managers
10. Select on sales to all managers
11. All privileges on supplier to all managers
12. All privileges on supplied to all managers

13. All privileges of Cashier to Cash Manager with admin option
14. All privileges on employeeCashier to Cash Manager

15. All privileges of foodEmployee to foodManager with admin option
16. All privileges on foodSection to foodManager
17. All privileges on productFood to foodManager
18. All privileges on employeeFood to foodManager

19. All privileges of householdEmployee to householdManager with admin option
20. All privileges on householdSection to householdManager
21. All privileges on productHousehold to householdManager
22. All privileges on employeeHousehold to householdManager

23. All privileges of sportsEmployee to sportsManager with admin option
24. All privileges on sportsSection to sportsManager
25. All privileges on productSports to sportsManager
26. All privileges on employeeSports to sportsManager

27. All privileges of electronicsEmployee to electronicsManager with admin option
28. All privileges on electronicsSection to electronicsManager
29. All privileges on productElectronics to electronicsManager
30. All privileges on employeeElectronics to electronicsManager

31. All privileges of clothesEmployee to clothesManager with admin option
32. All privileges on clothesSection to clothesManager
33. All privileges on productClothes to clothesManager
34. All privileges on employeeClothes to clothesManager

35. All privileges of customer to all employees
36. Select on discount to all employees
37. Select on offers to all employees

38. Select on foodDetails to foodEmployee
39. Select on householdDetails to householdEmployee
40. Select on sportsDetails to sportsEmployee
41. Select on electronicsDetails to electronicsEmployee
42. Select on clothesDetails to clothesEmployee

43. All privileges of customer to cashier with admin option
44. Select on product to cashier
45. Update only quantity in stock on product to cashier
46. Select on discount to cashier
47. Select on offers to cashier
48. Select on viewDiscounts to cashier
49. All privileges on sales to cashier
50. Select on bill to cashier
51. Insert on bill to cashier
52. Update only amount on bill to cashier
53. Select on customer to cashier
54. Insert on customer to cashier
55. Update on customer to cashier

56. Select on viewFood to customer
57. Select on viewHousehold to customer
58. Select on viewSports to customer
59. Select on viewElectronics to customer
60. Select on viewClothes to customer
61. Select on viewProducts to customer
62. Select on viewDiscounts to customer

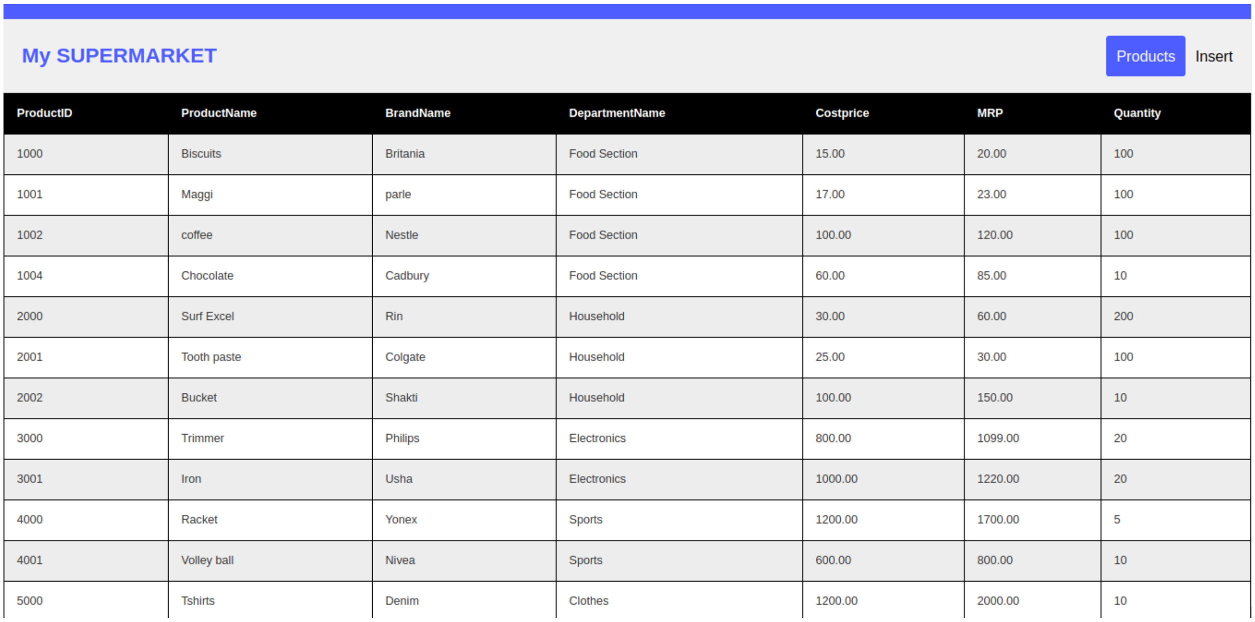
Triggers

1) Managers can ‘receive’ products for their own departments only
2) Managers can add new products for their own department only
3) Managers can offer a discount for their own department only
4) Managers can order products of their own department only
5) Managers can update vacancy of employees of their own department only
6) Managers can update supply of products of their own department only

7) Prevent fraudulent supply, ie do not allow supply of a product more than ordered
8) Automatically increment quantity in stock of products when a product is received
9) Prevent fraud transactions, ie do now allow customer to buy more products than in stock
10) Automatically decrement quanitity in stock of products when a product is sold
11) Detect whether the discount applied is available or expired
12) Automatically update the billing amount as the items are sold
13) Automatically update the total amount spent by the customer with billing amount

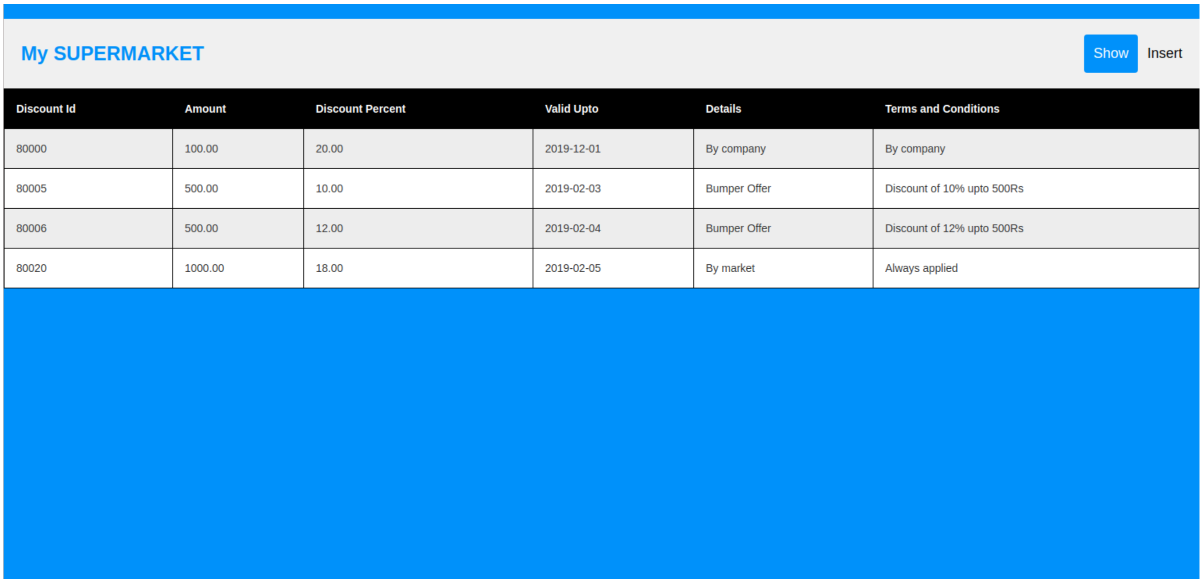
Graphical Interface

1. Show Product Table



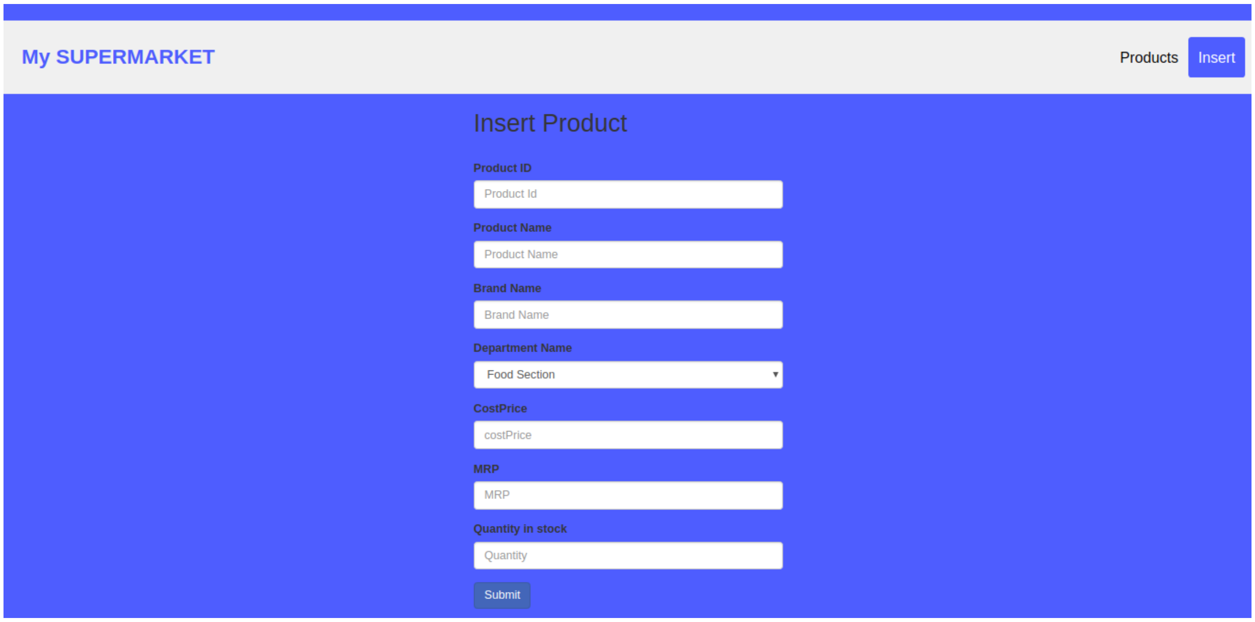
ProductID	ProductName	BrandName	DepartmentName	Costprice	MRP	Quantity
1000	Biscuits	Britania	Food Section	15.00	20.00	100
1001	Maggi	parle	Food Section	17.00	23.00	100
1002	coffee	Nestle	Food Section	100.00	120.00	100
1004	Chocolate	Cadbury	Food Section	60.00	85.00	10
2000	Surf Excel	Rin	Household	30.00	60.00	200
2001	Tooth paste	Colgate	Household	25.00	30.00	100
2002	Bucket	Shakti	Household	100.00	150.00	10
3000	Trimmer	Philips	Electronics	800.00	1099.00	20
3001	Iron	Usha	Electronics	1000.00	1220.00	20
4000	Racket	Yonex	Sports	1200.00	1700.00	5
4001	Volley ball	Nivea	Sports	600.00	800.00	10
5000	Tshirts	Denim	Clothes	1200.00	2000.00	10

2. Show Discount Table



Discount Id	Amount	Discount Percent	Valid Upto	Details	Terms and Conditions
80000	100.00	20.00	2019-12-01	By company	By company
80005	500.00	10.00	2019-02-03	Bumper Offer	Discount of 10% upto 500Rs
80006	500.00	12.00	2019-02-04	Bumper Offer	Discount of 12% upto 500Rs
80020	1000.00	18.00	2019-02-05	By market	Always applied

3. Insert in Product Table



Insert Product

Product ID

Product Id

Product Name

Product Name

Brand Name

Brand Name

Department Name

Food Section

CostPrice

costPrice

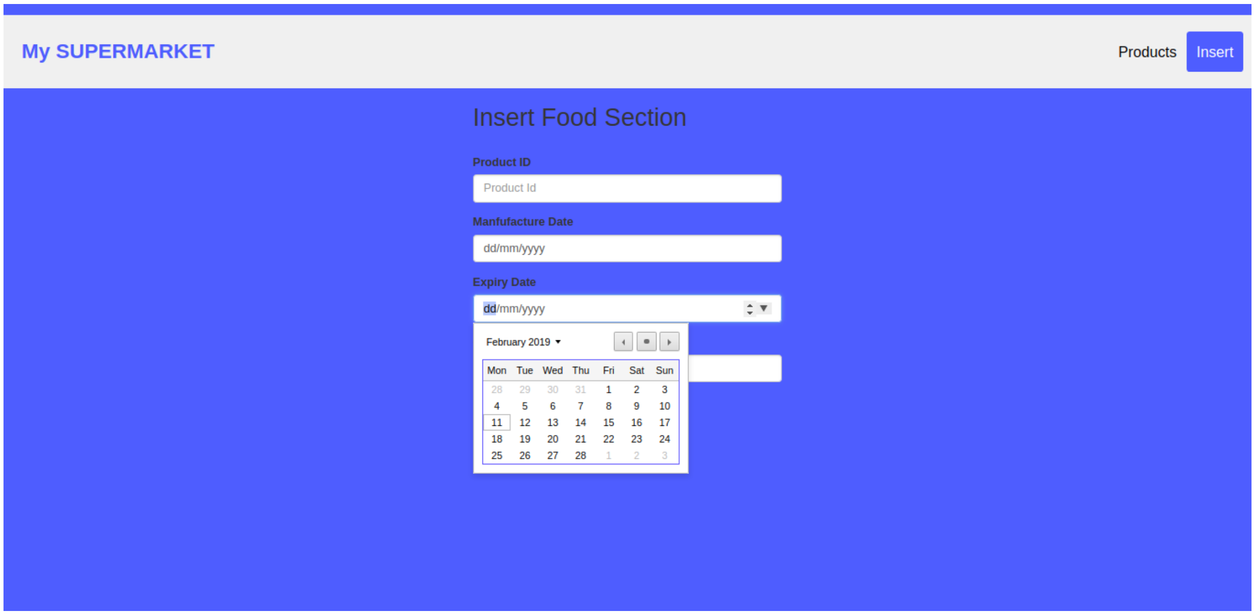
MRP

MRP

Quantity in stock

Quantity

4. Insert in Food Section Table



Insert Food Section

Product ID

Product Id

Manfufacture Date

dd/mm/yyyy

Expiry Date

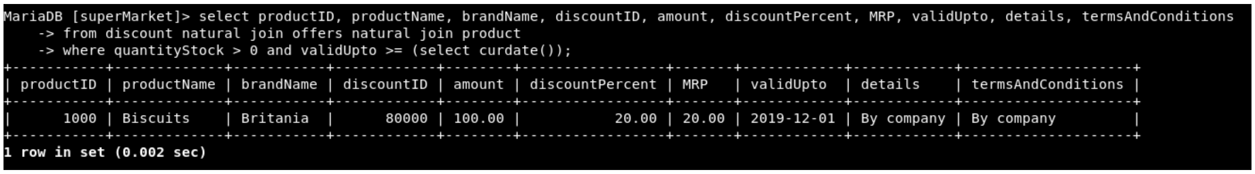
dd/mm/yyyy

February 2019

Mon	Tue	Wed	Thu	Fri	Sat	Sun
28	29	30	31	1	2	3
4	5	6	7	8	9	10
11	12	13	14	15	16	17
18	19	20	21	22	23	24
25	26	27	28	1	2	3

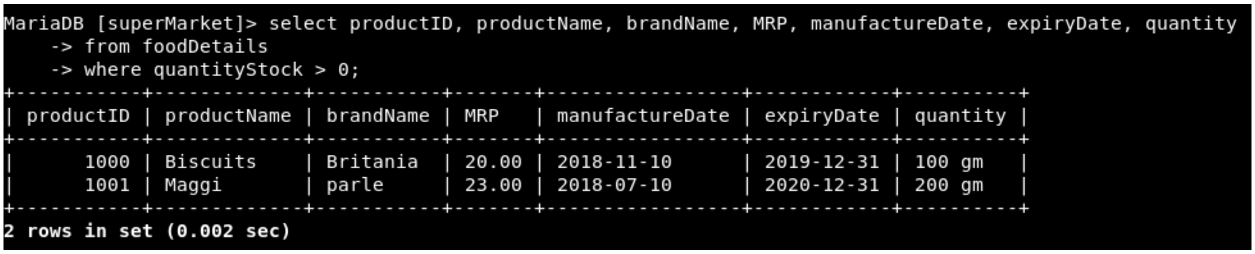
Use Cases

1. Show all Available discounts



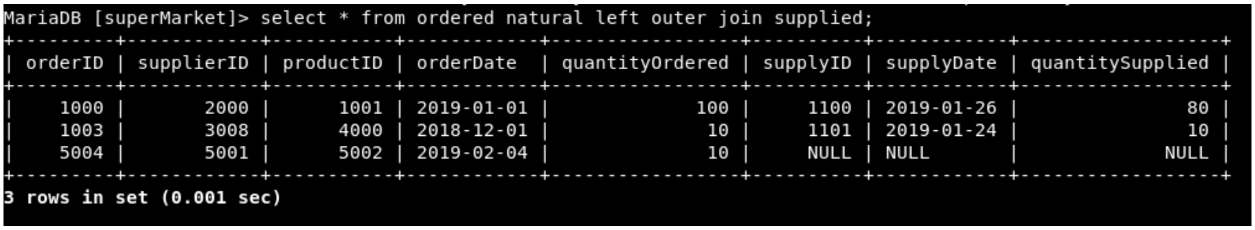
```
MariaDB [superMarket]> select productID, productName, brandName, discountID, amount, discountPercent, MRP, validUpto, details, termsAndConditions
    -> from discount natural join offers natural join product
    -> where quantityStock > 0 and validUpto >= (select curdate());
+-----------+-------------+-----------+------------+--------+-----------------+-------+------------+------------+--------------------+
| productID | productName | brandName | discountID | amount | discountPercent | MRP   | validUpto  | details    | termsAndConditions |
+-----------+-------------+-----------+------------+--------+-----------------+-------+------------+------------+--------------------+
|      1000 | Biscuits    | Britania  |      80000 | 100.00 |           20.00 | 20.00 | 2019-12-01 | By company | By company         |
+-----------+-------------+-----------+------------+--------+-----------------+-------+------------+------------+--------------------+
1 row in set (0.002 sec)
```

2. Show food product details to customer



```
MariaDB [superMarket]> select productID, productName, brandName, MRP, manufactureDate, expiryDate, quantity
    -> from foodDetails
    -> where quantityStock > 0;
+-----------+-------------+-----------+-------+-----------------+------------+----------+
| productID | productName | brandName | MRP   | manufactureDate | expiryDate | quantity |
+-----------+-------------+-----------+-------+-----------------+------------+----------+
|      1000 | Biscuits    | Britania  | 20.00 | 2018-11-10      | 2019-12-31 | 100 gm   |
|      1001 | Maggi       | parle     | 23.00 | 2018-07-10      | 2020-12-31 | 200 gm   |
+-----------+-------------+-----------+-------+-----------------+------------+----------+
2 rows in set (0.002 sec)
```

3. Show orders and supply to check unfulfilled orders



```
MariaDB [superMarket]> select * from ordered natural left outer join supplied;
+---------+------------+-----------+------------+-----------------+----------+------------+------------------+
| orderID | supplierID | productID | orderDate  | quantityOrdered | supplyID | supplyDate | quantitySupplied |
+---------+------------+-----------+------------+-----------------+----------+------------+------------------+
|    1000 |       2000 |      1001 | 2019-01-01 |             100 |     1100 | 2019-01-26 |               80 |
|    1003 |       3008 |      4000 | 2018-12-01 |              10 |     1101 | 2019-01-24 |               10 |
|    5004 |       5001 |      5002 | 2019-02-04 |              10 |     NULL | NULL       |             NULL |
+---------+------------+-----------+------------+-----------------+----------+------------+------------------+
3 rows in set (0.001 sec)
```

BCNF Checking

- product (*productID*, productName, brandName, departmentName, costPrice, MRP, quantityStock)
  Dependencies:
  1. productID -> product (productID is primary key, hence superkey)
  2. (productName, brandName) -> departmentName ((productName, brandName) is a superkey)

- foodSection (*productID*, manufactureDate, expiryDate, quantity)
  Dependencies:
  1. productID -> foodSection (productID is primary key)

- household (*productID,* quantity, category)
  Dependencies:
  1. productID -> household (productID is primary key)

- electronics (*productID*, powerRating, warranty, details)
  Dependencies:
  1. productID -> electronics (productID is primary key)

- sports (*productID,* warranty, details, category)
  Dependencies:
  1. productID -> sports (productID is primary key)

- clothes (*productID*, category, size, season)
  Dependencies:
  1. productID -> clothes (productID is primary key)

- department (*departmentName*, managerID, vacancy)
  Dependencies:
  1. departmentName -> department (departmentName is primary key)

- employee (*employeeID,* employeeName, DOB, phoneNo, employeeAddress, emailID, joinDate, salary, departmentName)
  Dependencies:
  1. employeeID -> employee (employeeID is primary key)

- supplier (*supplierID,* supplierName, phoneNo, supplierAddress, emailID)
  Dependencies:
  1. supplierID -> supplier (supplierID is primary key)

- ordered (*orderID,* supplierID, productID, orderDate, quantityOrdered)
  Dependencies:
  1. orderID -> ordered (orderID is primary key)

- supplied (*supplyID,* orderID, supplyDate, quantitySupplied)
  Dependencies:
  1. supplyID -> supplied (supplyID is primary key)

- discount (*discountID*, amount, discountPercent, validUpto, details, termsAndConditions)
  Dependencies:
  1. discountID -> discount (discountID is primary key)

- offers (*discountID, productID)*
  Dependencies:
  1. (discountID, productID) -> offers (trivial)

- customer (*customerID*, customerName, phoneNo, customerAddress, emailID, joinDate, moneySpent)
  Dependencies:
  1. customerID -> customer (customerID is primary key)
  2. (customerName, phoneNo) -> customer ((customerName, phoneNo) is super key)

- bill (*billNo,* billDate, customerID, amount)
  Dependencies:
  1. billNo -> bill (billNo is primary key)

- sales (*billNo, productID,* discountID, quantitySold)
  Dependencies:
  1. (billNo, productID) -> sales ((billNo, productID) is primary key)

Hence All Tables are in Boyce-Codd Normal Form!

Functions and Procedures

1. Find current monthly profit per department: Takes input a department name and month & year, and gives corresponding profit of the department (profit is less due to various discounts applied on different products, some discounts are percent based, while others are value based)
2. Find critical products: Takes input a department name, and critical quantity, if quantity is stock is less than critical quantity, then order should be placed. Helps managers to decide which order should be placed
3. Customer of the Month: A customer each month will be given a gift voucher who spent the maximum in the market as an appreciation. Find details of that customer for any month easily with this procedure.
4. Best Customer: Customer who has all time high record of spenditure in the market is given yet another token of appreciation.
5. UnfulfilledOrders: Find which orders are still pending with the number of days passed to help managers differenciate between good and bad suppliers
6. GetDiscounts: Enter a product ID and get all the currently applicable discounts at the time of buying a product to get the best value of your money. A greate advertisement to showcase our market as better than other people’s ;)