INVENTORY
DATABASE MANAGEMENT SYSTEM

Team Members:

1. C B Ananya
2. B Ayshwarya
3. Harini Mohan

Inventory management system:

The Inventory Management System refers to the system and processes to manage the stock of organization with the involvement of Technology system. This system can be used to store the details of the inventory, stock maintenance, update the inventory based on the sales details, generate sales and inventory report over a period of time. Inventory Management System is important to ensure quality control in businesses that handle transactions resolving around consumer goods. Without proper inventory control, a large retail store may runout of stock on an important item. Thus, it allows organizations to store, manage and track their goods and services and also analyze the past transactions and deals. It also enables them to manage the records of customers and their orders for future references and further details.

Features:

a. **User Roles**: The system will have two main user roles: admin and staff. The admin will have full access to all features and functionalities, while the staff will have limited access based on their assigned permissions like stock management and billing.

b. **Inventory Management**: The system will allow users to manage inventory, including adding new products, updating product information (e.g., name, description, price), and deleting products.

c. **Stock Tracking:** The system will track the quantity of each product in stock.

d. **Purchase Management**: Users will be able to create purchase orders for restocking inventory. The system will record and track purchase details such as supplier information & order date.

e. **Billing and Invoicing:** The system will generate invoices for customer purchases and track billing information. It will calculate the total amount due, apply discounts if applicable.

f. **User Authentication and Security:** The system will require users to authenticate themselves before accessing the application.

g.**Generate bill:** The invoice that has been generated for the particular order can be generated as a hard copy by connecting to system printer and also can be obtained as a pdf.

The relational schema consists of set of relational tables and associated items that are related to one another.

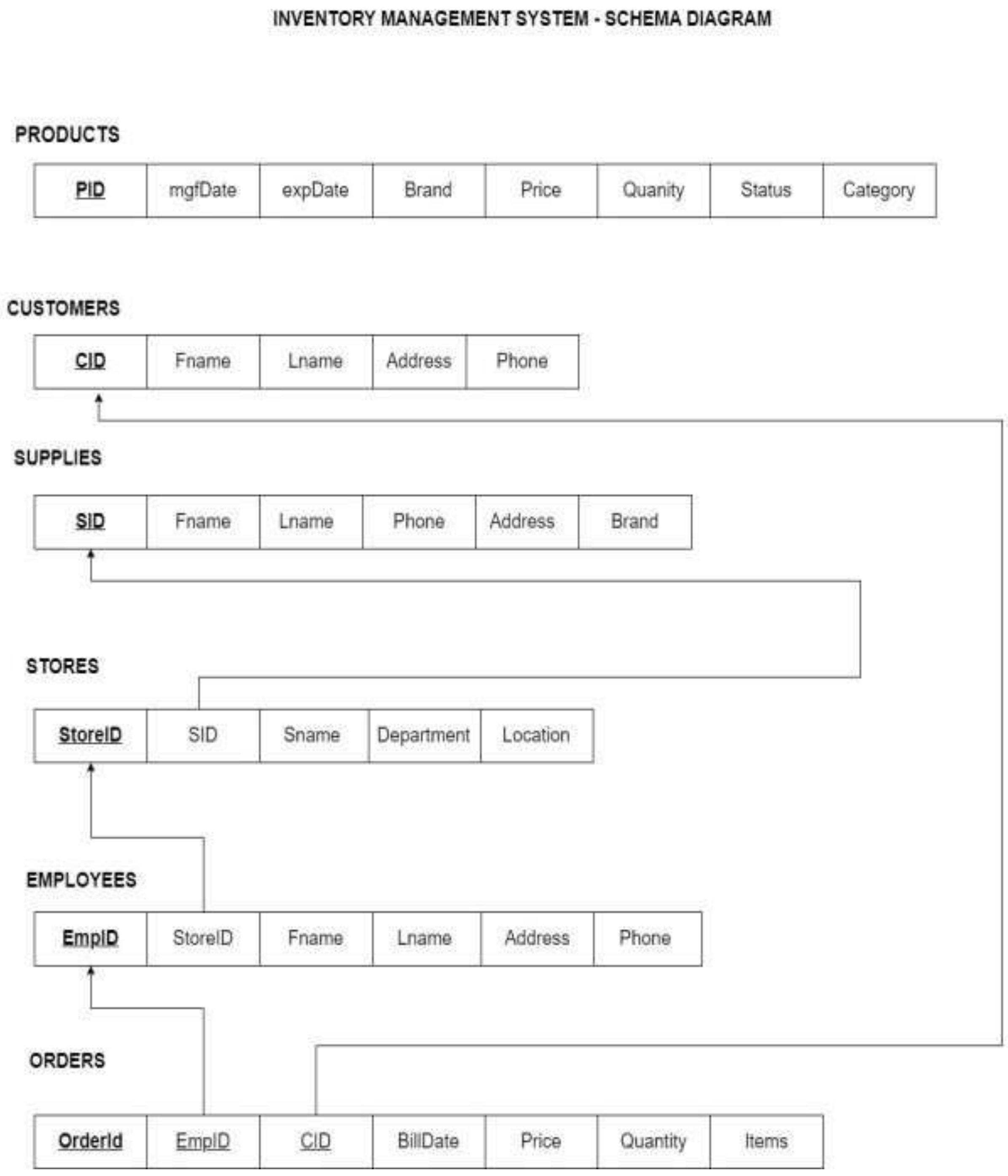
The relational schema for the inventory management system are as follows:

I. Normalised table for base relation

1. **Orders** (Order_ID, CID, EID, PID, bill_date, price, quantity)
2. Products (PID, mfdate, expdate, brand, price, quantity, pname, category)
3. Customers (CID, fname, lname, phno, doorno, street, area, city, state)
4. Suppliers (SID, sname, phno, doorno, street_no,area, city, state)
5. Employees (EID, storeid, fname, lname, phno, doorno, street, area, city, state)
6. Stores (Storeid, phno, sname, doorno, street, area, city, state)

7. **Supply**(SID, PID, SID, Storeid, supdate)

SCHEMA Diagram:



II. Fds, Normalisation & Candidate keys

EMPLOYEE RELATION

Employee(EmpID,fname,lname,address,phone,StoreID)

Functional dependencies:

EmpID->fname,lname,phone,address

EmpID->StoreID

Normalisation:

1NF

Address is a composite attribute in the relation Employee.

For Employee to be in 1NF, we need to decompose the composite attribute.

Therefore, address is decomposed into area, city & state.

The Employee relation is now in 1NF.

Employee(EmpID,fname,lname,area,city,state,phone,StoreID)

Irreducible set of FDs:

EmpID=A, fname=B, lname=C, area = D, city=E, state=F, phone=G, storeID=H

Resulting FD:

A->BCDEFGH

Irreducible set of FDs:

A->BCDEFGH

A->B

A->C

A->D

A->E

A->F

A->G

A->H

$\{A,B,C,D,E,F,G, H\}^+= \{A,B,C,D,E,F,G,H\}$

A->B:

$\{A,C,D,E,F, G,H\}^+=\{A,B,C,D,E,F,G,H\}$

A->C:

$\{A,B,D,E,F,G,H\}^+=\{A,B,C,D,E,F,G,H\}$

A->D:

$\{A,B,C,E,F,G,H\}^+=\{A,B,C,D,E,F,G,H\}$

A->E:

$\{A,B,C,D,F,G,H\}^+=\{A,B,C,D,E,F,G,H\}$

A->F:

$\{A,B,C,D,E,G,H\}^+=\{A,B,C,D,E,F,G,H\}$

A->G:

$\{A,B,C,D,E,F,H\}^+=\{A,B,C,D,E,F,G,H\}$

A->H

$\{A,B,C,D,E,F,G\}^+=\{A,B,C,D,E,F,G,H\}$

{A} is the super key.

{A} is the candidate key.

Since A is not present in the RHS of any FD, it is the only candidate key of the relation Employee.

Candidate key : {A} i.e. EmpID

Proper subset of Candidate key: { }

Proper subset of candidate key -> Non prime attributes? NO

Therefore, the relation Employee is fully functionally independent.

No partial dependency.

The relation is in 2NF.

Prime attribute: {A}

Non-prime attributes:{B,C,D,E,F,G,H}

3NF:

The LHS of all FDs is a super key and there is no transitive dependency. Hence the Employee relation is in 3NF.

BCNF:

The LHS of all FDs is the candidate key itself. Hence the relation is in BCNF.

Therefore, the relation is in 1NF, 2NF, 3NF and BCNF

Employee(EmpID,fname,lname,area,city,state,phone,StoreID)

STORE RELATION

STORE(StoreID,sname,department,address,phone)

Functional Dependencies:

StoreID-> sname,department,address,phone

Normalisation:

1NF

Address is a composite attribute in the relation Store.

For Store to be in 1NF, we need to decompose the composite attribute.

Therefore, address is decomposed into area, city & state.

The Store relation is now in 1NF.

Store(StoreID,sname,department,area,city,state,phone)

StoreID=A, sname=B, department=C, area = D, city=E, state=F, phone=G

Irreducible set of FDs:

Resulting FD

A->BCDEFG

Finding Irreducible sets of FDs:

A->B

A->C

A->D

A->E

A->F

A->G

$\{A,B,C,D,E,F,G\}^+ = \{A,B,C,D,E,F,G\}$

A->B:

$\{A,C,D,E,F,G\}^+=\{A,B,C,D,E,F,G\}$

A->C:

$\{A,B,D,E,F,G\}^+=\{A,B,C,D,E,F,G\}$

A->D:

$\{A,B,C,E,F,G\}^+=\{A,B,C,D,E,F,G\}$

A->E:

$\{A,B,C,D,F,G\}^+=\{A,B,C,D,E,F,G\}$

A->F:

$\{A,B,C,D,E,G\}^+=\{A,B,C,D,E,F,G\}$

A->G:

$\{A,B,C,D,E,F\}^+=\{A,B,C,D,E,F,G\}$

{A} is the super key.

{A} is the candidate key.

Since A is not present in the RHS of any FD, it is the only candidate key of the relation Store.

Candidate key : {A} i.e. StoreID

Proper subset of Candidate key: { }

Proper subset of candidate key -> Non prime attributes? NO

Therefore, the relation Store is fully functionally independent.

No partial dependency.

The relation is in 2NF.

Prime attribute: {A}

Non-prime attributes:{B,C,D,E,F,G}

3NF:

The LHS of all FDs is a super key and there is no transitive dependency. Hence the Store relation is in 3NF.

BCNF:

The LHS of all FDs is the candidate key itself. Hence the relation is in BCNF.

Therefore, the relation is in 1NF, 2NF, 3NF and BCNF

Store(StoreID,sname,department,area,city,state,phone)

CUSTOMER RELATION:

Customer(CID, fname, lname, phone, address)

Functional Dependencies:

CID=A, fname=B, lname=C, phone= D, address=E

{A->BCDE}

Normalisation:

1NF:

Address is a composite attribute in the relation Customer.

For Customer to be in 1NF, we need to decompose the composite attribute.

Therefore, address is decomposed into area, city & state.

The Customer relation is now in 1NF.

Customer(CID, fname, lname, phone, area, city, state)

CID=A, fname=B, lname=C, phone= D, area=E, city=F, state=G

Resulting FD

A->BCDEFG

Irreducible sets of FDs:

A->B

A->C

A->D

A->E

A->F

A->G

$\{A,B,C,D,E,F,G\}^{+}= \{A,B,C,D,E,F,G\}$

A->B:

$\{A,C,D,E,F,G\}^{+}=\{A,B,C,D,E,F,G\}$

A->C:

$\{A,B,D,E,F,G\}^{+}=\{A,B,C,D,E,F,G\}$

A->D:

$\{A,B,C,E,F,G\}^{+}=\{A,B,C,D,E,F,G\}$

A->E:

$\{A,B,C,D,F,G\}^{+}=\{A,B,C,D,E,F,G\}$

A->F:

$\{A,B,C,D,E,G\}^{+}=\{A,B,C,D,E,F,G\}$

A->G:

$\{A,B,C,D,E,F\}^{+}=\{A,B,C,D,E,F,G\}$

{A} is the super key.

{A} is the candidate key.

Since A is not present in the RHS of any FD, it is the only candidate key of the relation Customer.

Candidate key : {A} i.e. CID

Proper subset of Candidate key: { }

Proper subset of candidate key -> Non prime attributes? NO

Therefore, the relation customer is fully functionally independent.

No partial dependency.

The relation is in 2NF.

Prime attribute: {A}

Non-prime attributes:{B,C,D,E,F,G}

3NF:

The LHS of all FDs is a super key and there is no transitive dependency. Hence the Customer relation is in 3NF.

BCNF:

The LHS of all FDs is the candidate key itself. Hence the relation is in BCNF.

Therefore, the relation is in 1NF, 2NF, 3NF and BCNF

Customer(CID, fname, lname, phone, area, city, state)

SUPPLIER RELATION

Suppliers (SID, fname, lname, phno, address, brand)

Functional Dependencies:

SID-> fname, lname, phno, address, brand

1NF:

Address is a composite attribute in the relation Supplier.

For Supplier to be in 1NF, we need to decompose the composite attribute.

Therefore, address is decomposed into area, city & state.

The Supplier relation is now in 1NF.

SID=A, fname=B, lname=C, phno = D, area= E, city=F, state=G, brand=H

Resulting FD:

A->BCDEFGH

Irreducible set of FDs:

A->BCDEFGH

A->B

A->C

A->D

A->E

A->F

A->G

A->H

$\{A,B,C,D,E,F,G, H\}^{+}= \{A,B,C,D,E,F,G,H\}$

A->B:

$\{A,C,D,E,F, G,H\}^{+}=\{A,B,C,D,E,F,G,H\}$

A->C:

$\{A,B,D,E,F,G,H\}^{+}=\{A,B,C,D,E,F,G,H\}$

A->D:

$\{A,B,C,E,F,G,H\}^{+}=\{A,B,C,D,E,F,G,H\}$

A->E:

$\{A,B,C,D,F,G,H\}^{+}=\{A,B,C,D,E,F,G,H\}$

A->F:

$\{A,B,C,D,E,G,H\}^{+}=\{A,B,C,D,E,F,G,H\}$

A->G:

$\{A,B,C,D,E,F,H\}^{+}=\{A,B,C,D,E,F,G,H\}$

A->H

$\{A,B,C,D,E,F,G\}^{+}=\{A,B,C,D,E,F,G,H\}$

{A} is the super key.

{A} is the candidate key.

Since A is not present in the RHS of any FD, it is the only candidate key of the relation Supplier.

Candidate key : {A}

Proper subset of Candidate key: { }

Proper subset of candidate key -> Non prime attributes? NO

Therefore, the relation supplier is fully functionally independent.

No partial dependency.

The relation is in 2NF.

Prime attribute: {A}

Non-prime attributes:{B,C,D,E,F,G,H}

3NF:

The LHS of all FDs is a super key and there is no transitive dependency. Hence the Supplier relation is in 3NF.

BCNF:

The LHS of all FDs is the candidate key itself. Hence the relation is in BCNF.

Therefore, the relation is in 1NF, 2NF, 3NF and BCNF

Supplier(SID, fname, lname, phone, area, city, state, address)

ORDER RELATION:

Order(OrderID, CID, EmpID, billdate, price, quantity, items, billamt)

Functional Depedencies:

1. OrderID -> CID, EmpID, billdate, price, quantity, items, billamt
2. OrderID, item -> price, quantity
3. Item, quantity -> billamt
4. CID, OrderID -> EmpID

Let A=OrderID, B=CID, C=EmpID, D=billdate, E=price, F=quantity, G=items, H=billamt

Normalisation:

1. Finding Candidate key for FDs:

$\{A,B,C,D,E,F,G,H\}^+$ -> {A,B,C,D,E,F,G,H}

A -> B,C,D,E,F,G,H

$\{A\}^+$ -> {A,B,C,D,E,F,G,H}

{A} is the super key.

{A} is the candidate key.

Since A is not present in the RHS of any FD, it is the only candidate key of the relation Product.

The LHS of the **FD1, FD2, FD4** contains super key A. Hence, the FD exists in BCNF.

i.e, It satisfies 1NF, 2NF, 3NF and BCNF.

Prime attribute : A

Non Prime attribute : B,C,D,E,F,G,H

The 3rd FD exists in 2NF, as there is no partial dependency

The 3rd FD doesn't exist in 3NF,since there exist a transitive dependency (i.e, Non prime attribute determining other Non prime attribute).

G,F -> H (transitive dependency)

Decomposing the FDs:

R1 (A,B,C,D,E)

$\{A\}^+$ -> {A,B,C,D,E}

A is the candidate key.

Only FD is A -> B,C,D,E

There is no transitive dependency. Hence, FD exists in 3NF.

Super key A exists on LHS of the FD, hence the FD is in BCNF.

R2(A,F,G,H)

$\{F,G\}^+$ -> {F,G,H}

F,G is the super key --> **FG is the candidate key** (No proper subset for the super key)

Prime attribute : F,G

Non prime attribute : H

Hence, the FD : FD -> H satisfies 3NF as there is no transitive dependency

The FD also satisfies BCNF as the LHS of the FD has a super key.

DECOMPOSED RELATIONS:

1. R1 A,B,C,D,E ()
2. R2(A,F,G,H)

That is:

- (OrderID, CID, EmpID, billdate, price)
- (items, quantity, billamt)

PRODUCT RELATION:

Product(PID, mgfdate, expdate, brand, price, quantity, soldstatus, category)

Functional Depedencies:

1. PID -> mgfdate, expdate, brand, price, quantity, soldstatus, category
2. Brand -> category
3. Brand, soldstatus, quantity -> price

Let A=PID, B=mgfdate, C=expdate, D=brand, E=price, F=quantity, G=soldstatus, H=category

Normalisation:

Finding Candidate key for FDs:

$\{A,B,C,D,E,F,G,H\}^+$ -> {A,B,C,D,E,F,G,H}

A -> B,C,D,E,F,G,H

$\{A\}^+$ -> {A,B,C,D,E,F,G,H}

{A} is the super key.

{A} is the candidate key.

Since A is not present in the RHS of any FD, it is the only candidate key of the relation Product.

The LHS of the **FD1** contains super key A. Hence, the FD exists in BCNF.

i.e, It satisfies 1NF, 2NF, 3NF and BCNF.

Prime attribute : A

Non Prime attribute : B,C,D,E,F,G,H

The 2nd and 3rd FDs exist in 2NF, as there is no partial dependency

The 2nd and 3rd FD doesn't exist in 3NF, since there exist a transitive dependency (i.e, Non prime attribute determining other non-prime attribute).

D -> H (transitive dependency)

D,F,G -> E (transitive dependency)

Decomposing the FDs:

R1 (D,H)

$\{D\}^+$ -> {D,H}

$\{H\}^+$ -> {H}

D is the super key --> **D is the candidate key** (No proper subset for the super key)

Prime attribute : D

Non prime attribute : H

Hence, the FD : D -> H satisfies 3NF as there is no transistive dependency

The FD also satisfies BCNF as the LHS of the FD has a super key.

R2(D,G,F,E)

$\{DGF\}^+$ -> {D,G,F,E}

DGF is the super key

DGF is the candidate key

FD is : {D,G,F} -> {E}

There is no transitive dependency. Hence, FD exists in 3NF.

Super key DGF exists on LHS of the FD, hence the FD is in BCNF.

R3(A,B,C,D)

$\{A\}^+$ -> {A,B,C,D,G,F}

A is the candidate key.

Only FD is A -> B,C,D,G,F

There is no transitive dependency. Hence, FD exists in 3NF.

Super key A exists on LHS of the FD, hence the FD is in BCNF.

DECOMPOSED RELATIONS:

1. R1 (D,H)
2. R2(D,G,F,E)
3. R3(A,B,C,D,G,F)

That is:

- (brand, category)
- (brand, soldstatus, quantity, price)
- (PID, mgfdate, expdate, brand, soldstatus, quantity)

Prove that the relations in higher normal form satisfy the lossless join property

Lossless-join decomposition is a process in which a relation is decomposed into two or more relations. This property guarantees that the extra or less tuple generation problem does not occur and no information is lost from the original relation during the decomposition.

Rules:

The union of the decomposed relations must consist of all the attributes of the orginal relation

The intersection of the decomposed relation must not be null.

In ***ORDER*** relation, the decomposed relations are:

1. R1 (A,F,G,H)
2. R2(A,B,C,D,E)

Union of R1 and R2

R1 U R2 = A,B,C,D,E,F,G,H

Intersection of R1 and R2

R1 $\cap$ R2 = A.

Therefore, it is a lossless decomposition.

In ***PRODUCT*** relation, the decomposed relations are:

1. R1 (D,H)
2. R2(D,G,F,E)
3. R3(A,B,C,D,G,F)

Union of R1 and R2

R1 U R2 = A,B,C,D,E,F,G,H

Intersection of R1 and R2

R1 $\cap$ R2 = D.

Therefore, it is a lossless decomposition.

Therefore, the higher order relations are lossless.

ER DIAGRAM (After normalisation):