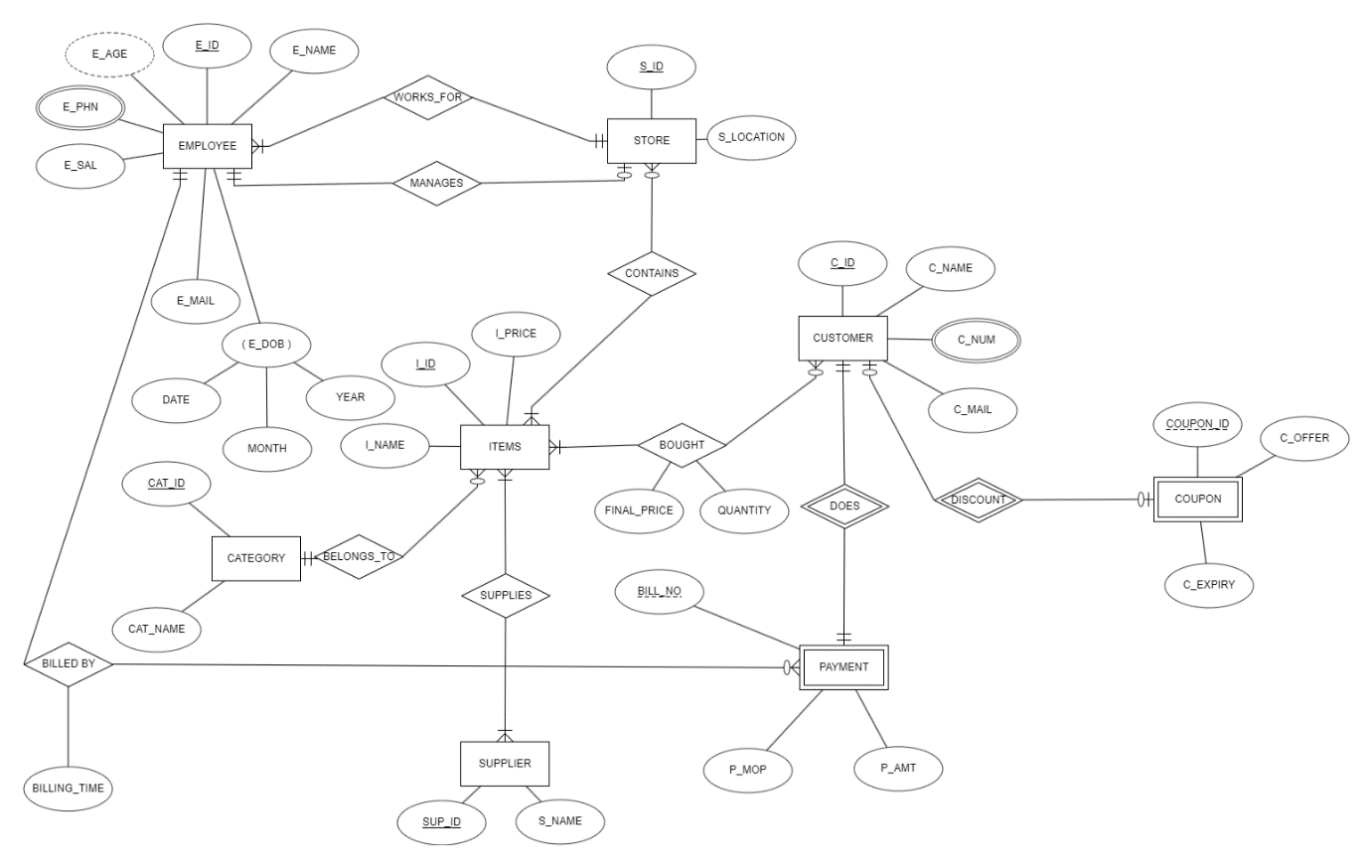
# PROBLEM STATEMENT:

The motive of this project is to design a data base for a super market chain that has its branches in many cities it sells all the essentials, groceries etc. The data base maintains the record for the employees, suppliers, each store and the billing and payment details of the customers and the discounts offered to them based on the purchasing quantity.

# ER MODEL:

The following ER Diagram shows the way that the entities and the attributes being matched and represented with the constraints involved.



# RELATION SCHEMA:

The following Relational Schema shows the relational schema of the ER model of the Supermarket database. They are represented in tables with the attributes mentioned and indicating the primary foreign key for each relation in the schema.

# 

Now the schema is to be normalised.

# NORMALISATION:

The relations are:

Employee

* E\_ID
* E\_NAME
* E\_SAL
* E\_MAIL
* E\_DOB
* S\_ID

Functional dependencies are:

E\_ID -> (E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID)

Relation is in BCNF implying it is also in 1NF,2NF,3NF.

Emp\_phn

* E\_ID
* E\_PHN

There are no functional dependencies implying it’s in BCNF.

Store

* S\_ID
* S\_LOCATION
* M\_ID

Functional dependencies are:

S\_ID -> (S\_location,M\_ID)

Relation is in BCNF implying it is also in 1NF,2NF,3NF.

Customer

* C\_ID
* C\_NAME
* C\_MAIL

Functional dependencies are:

C\_ID -> (C\_NAME,C\_MAIL)

Relation is in BCNF implying it is also in 1NF,2NF,3NF.

C\_PHN

* C\_ID
* C\_NUM

There are no functional dependencies implying it’s in BCNF.

CATEGORY

* CAT\_ID
* CAT\_NAME

Functional dependencies are:

CAT\_ID -> (CAT\_NAME)

Relation is in BCNF implying it is also in 1NF,2NF,3NF.

SUPPLIERS

* SUP\_ID
* SUP\_NAME

Functional dependencies are:

SUP\_ID -> (SUP\_NAME)

Relation is in BCNF implying it is also in 1NF,2NF,3NF.

ITEMS

* I\_ID
* I\_NAME
* I\_PRICE
* CAT\_ID

Functional dependencies are:

I\_ID -> (I\_NAME,I\_PRICE,CAT\_ID)

Relation is in BCNF implying it is also in 1NF,2NF,3NF.

SUPPLIES

* I\_ID
* SUP\_ID

There are no functional dependencies implying it’s in BCNF.

COUPON

* COUPON\_NUM
* C\_ID
* C\_OFFER
* C\_EXPIRY

Functional dependencies are:

C\_ID,COUPON\_NUM -> (C\_OFFER,C\_EXPIRY)

Relation is in BCNF implying it is also in 1NF,2NF,3NF.

Bought

* C\_ID
* I\_ID
* QUANTITY
* FINAL\_PRICE

Functional dependencies are:

C\_ID,I\_ID -> (FINAL\_PRICE,QUANTITY)

Relation is in BCNF implying it is also in 1NF,2NF,3NF.

CONTAINS

* S\_ID
* I\_id

There are no functional dependencies implying it’s in BCNF.

Payment

* Bill\_No
* C\_ID
* P\_MOP
* P\_AMT
* BILLING\_TIME
* E\_ID

Functional dependencies are:

C\_ID,BILL\_NO -> (P\_MOP,P\_AMT,BILLING\_TIMR,E\_ID)

Relation is in BCNF implying it is also in 1NF,2NF,3NF.

# RELATIONAL TABLE CREATION:

EMPLOYEE:

CREATE TABLE EMPLOYEE

(

E\_ID INT PRIMARY KEY,

E\_NAME VARCHAR(20),

E\_SAL INT,

E\_MAIL VARCHAR(20),

E\_DOB DATE,

S\_ID INT

);

# 

EMP\_PHN

CREATE TABLE EMP\_PHN

(

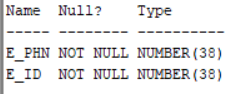
E\_PHN INT,

E\_ID INT,

PRIMARY KEY(E\_PHN,E\_ID),

FOREIGN KEY (E\_ID) REFERENCES EMPLOYEE(E\_ID)

);



STORE:

CREATE TABLE "STORE"

(

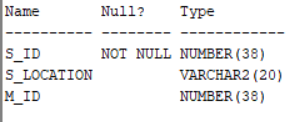
S\_ID INT PRIMARY KEY,

S\_LOCATION VARCHAR(20),

M\_ID INT,

FOREIGN KEY (M\_ID) REFERENCES EMPLOYEE(E\_ID)

);



ALTER TABLE EMPLOYEE

ADD FOREIGN KEY (S\_ID) REFERENCES "STORE"(S\_ID);

CATEGORY:

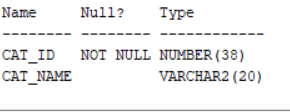
CREATE TABLE CATEGORY

(

CAT\_ID INT PRIMARY KEY,

CAT\_NAME VARCHAR(20)

);



SUPPLIERS:

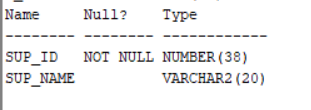
CREATE TABLE SUPPLIERS

(

SUP\_ID INT PRIMARY KEY,

SUP\_NAME VARCHAR(20)

);



ITEMS:

CREATE TABLE ITEMS

(

I\_ID INT PRIMARY KEY,

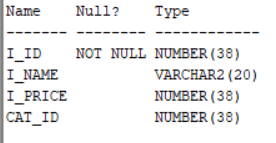
I\_NAME VARCHAR(20),

I\_PRICE INT,

CAT\_ID INT,

FOREIGN KEY (CAT\_ID) REFERENCES CATEGORY(CAT\_ID)

);



SUPPLIES:

CREATE TABLE SUPPLIES

(

SUP\_ID INT ,

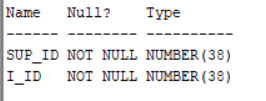
I\_ID INT,

PRIMARY KEY(SUP\_ID,I\_ID),

FOREIGN KEY(SUP\_ID) REFERENCES SUPPLIERS(SUP\_ID),

FOREIGN KEY(I\_ID) REFERENCES ITEMS(I\_ID)

);



CONTAINS:

CREATE TABLE CONTAINS

(

S\_ID INT ,

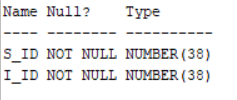
I\_ID INT,

PRIMARY KEY(S\_ID,I\_ID),

FOREIGN KEY(S\_ID) REFERENCES "STORE"(S\_ID),

FOREIGN KEY(I\_ID) REFERENCES ITEMS(I\_ID)

);



CUSTOMER:

CREATE TABLE CUSTOMER

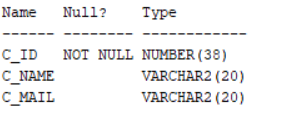
(

C\_ID INT PRIMARY KEY,

C\_NAME VARCHAR(20),

C\_MAIL VARCHAR(20)

);



C\_PHN:

CREATE TABLE C\_PHN

(

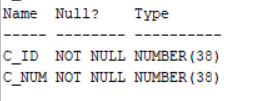
C\_ID INT ,

C\_NUM INT,

PRIMARY KEY(C\_ID,C\_NUM),

FOREIGN KEY (C\_ID) REFERENCES CUSTOMER(C\_ID)

);



COUPON:

CREATE TABLE COUPON

(

COUPON\_NUM INT,

C\_ID INT,

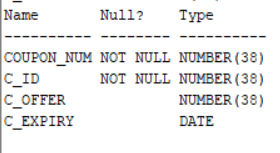
C\_OFFER INT,

C\_EXPIRY DATE,

PRIMARY KEY(COUPON\_NUM,C\_ID),

FOREIGN KEY (C\_ID) REFERENCES CUSTOMER(C\_ID)

);



BOUGHT:

CREATE TABLE BOUGHT

(

C\_ID INT,

I\_ID INT,

FINAL\_PRICE INT,

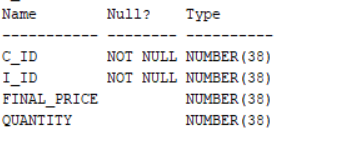
QUANTITY INT,

PRIMARY KEY(C\_ID,I\_ID),

FOREIGN KEY (C\_ID) REFERENCES CUSTOMER(C\_ID),

FOREIGN KEY (I\_ID) REFERENCES ITEMS(I\_ID)

);



PAYMENT:

CREATE TABLE PAYMENT

(

BILL\_NO INT,

C\_ID INT,

P\_MOP VARCHAR(20),

P\_AMT INT,

BILLING\_TIME TIMESTAMP,

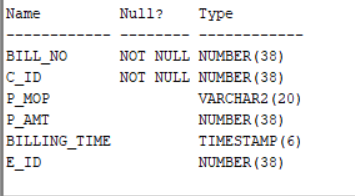
E\_ID INT,

PRIMARY KEY(BILL\_NO,C\_ID),

FOREIGN KEY(C\_ID) REFERENCES CUSTOMER(C\_ID),

FOREIGN KEY(E\_ID) REFERENCES EMPLOYEE(E\_ID)

);



# INSERTION OF VALUES INTO THE RELATIONS:

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (1, 'John', 50000, 'john@email.com', '01-01-1990', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (2, 'Sarah', 55000, 'sarah@email.com', '02-02-1991', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (3, 'Mike', 60000, 'mike@email.com', '03-03-1992', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (4, 'Emily', 65000, 'emily@email.com', '04-04-1993', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (5, 'David', 70000, 'david@email.com', '05-05-1994', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (6, 'Jessica', 75000, 'jessica@email.com', '06-06-1995', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (7, 'Mark', 80000, 'mark@email.com', '07-07-1996', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (8, 'Linda', 85000, 'linda@email.com', '08-08-1997', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (9, 'Brian', 90000, 'brian@email.com', '09-09-1997', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (10, 'Emma', 95000, 'emma@email.com', '10-10-1999', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (11, 'Kevin', 100000, 'kevin@email.com', '11-11-2000', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (12, 'Olivia', 105000, 'olivia@email.com', '12-12-2001', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (13, 'Jacob', 110000, 'jacob@email.com', '01-01-2002', NULL);

INSERT INTO employee (E\_ID, E\_NAME, E\_SAL, E\_MAIL, E\_DOB, S\_ID) VALUES (14, 'Avery', 115000, 'avery@email.com', '02-02-2003', NULL);

DELETE FROM EMPLOYEE;

SELECT \* FROM EMPLOYEE;

INSERT INTO store (S\_ID, S\_LOCATION, M\_ID) VALUES (1, 'New York', 1);

INSERT INTO store (S\_ID, S\_LOCATION, M\_ID) VALUES (2, 'Los Angeles', 2);

INSERT INTO store (S\_ID, S\_LOCATION, M\_ID) VALUES (3, 'Chicago', 3);

SELECT \* FROM STORE;

UPDATE EMPLOYEE SET S\_ID=1 WHERE E\_ID IN (1,4,7);

UPDATE EMPLOYEE SET S\_ID=2 WHERE E\_ID IN (2,5,8);

UPDATE EMPLOYEE SET S\_ID=3 WHERE E\_ID IN (3,6,9,10);

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (1, '123-456-7890');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (1, '987-654-3210');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (2, '555-555-5555');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (2, '111-111-1111');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (2, '222-222-2222');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (3, '333-333-3333');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (3, '444-444-4444');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (4, '555-555-5555');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (5, '666-666-6666');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (5, '777-777-7777');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (6, '888-888-8888');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (7, '999-999-9999');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (7, '111-222-3333');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (7, '444-555-6666');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (8, '777-777-7777');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (8, '888-888-8888');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (8, '999-999-9999');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (9, '111-111-1111');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (9, '222-222-2222');

INSERT INTO emp\_phn (E\_ID, E\_PHN) VALUES (10, '333-333-3333');

SELECT \* FROM EMP\_PHN;

INSERT INTO suppliers (sup\_id, sup\_name) VALUES (1, 'Supplier A');

INSERT INTO suppliers (sup\_id, sup\_name) VALUES (2, 'Supplier B');

INSERT INTO suppliers (sup\_id, sup\_name) VALUES (3, 'Supplier C');

INSERT INTO suppliers (sup\_id, sup\_name) VALUES (4, 'Supplier D');

INSERT INTO suppliers (sup\_id, sup\_name) VALUES (5, 'Supplier E');

INSERT INTO suppliers (sup\_id, sup\_name) VALUES (6, 'Supplier F');

SELECT \* FROM SUPPLIERS;

INSERT INTO category (cat\_id, cat\_name) VALUES (1, 'Category A');

INSERT INTO category (cat\_id, cat\_name) VALUES (2, 'Category B');

INSERT INTO category (cat\_id, cat\_name) VALUES (3, 'Category C');

INSERT INTO category (cat\_id, cat\_name) VALUES (4, 'Category D');

INSERT INTO category (cat\_id, cat\_name) VALUES (5, 'Category E');

SELECT \* FROM CATEGORY;

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (1, 'Item A', 10, 1);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (2, 'Item B', 20, 2);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (3, 'Item C', 5, 3);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (4, 'Item D', 15, 4);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (5, 'Item E', 8, 5);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (6, 'Item F', 12, 1);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (7, 'Item G', 7, 2);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (8, 'Item H', 25, 3);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (9, 'Item I', 18, 4);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (10, 'Item J', 14, 5);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (11, 'Item K', 22, 1);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (12, 'Item L', 30, 2);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (13, 'Item M', 16, 3);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (14, 'Item N', 11, 4);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (15, 'Item O', 20, 5);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (16, 'Item P', 15, 1);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (17, 'Item Q', 12, 2);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (18, 'Item R', 28, 3);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (19, 'Item S', 32, 4);

INSERT INTO items (i\_id, i\_name, i\_price, cat\_id) VALUES (20, 'Item T', 19, 5);

SELECT \* FROM ITEMS;

INSERT INTO supplies (i\_id, sup\_id) VALUES (1, 1);

INSERT INTO supplies (i\_id, sup\_id) VALUES (1, 3);

INSERT INTO supplies (i\_id, sup\_id) VALUES (2, 2);

INSERT INTO supplies (i\_id, sup\_id) VALUES (2, 3);

INSERT INTO supplies (i\_id, sup\_id) VALUES (3, 5);

INSERT INTO supplies (i\_id, sup\_id) VALUES (3, 6);

INSERT INTO supplies (i\_id, sup\_id) VALUES (4, 4);

INSERT INTO supplies (i\_id, sup\_id) VALUES (4, 2);

INSERT INTO supplies (i\_id, sup\_id) VALUES (5, 1);

INSERT INTO supplies (i\_id, sup\_id) VALUES (5, 4);

INSERT INTO supplies (i\_id, sup\_id) VALUES (6, 1);

INSERT INTO supplies (i\_id, sup\_id) VALUES (6, 2);

INSERT INTO supplies (i\_id, sup\_id) VALUES (7, 6);

INSERT INTO supplies (i\_id, sup\_id) VALUES (7, 3);

INSERT INTO supplies (i\_id, sup\_id) VALUES (8, 5);

INSERT INTO supplies (i\_id, sup\_id) VALUES (8, 6);

INSERT INTO supplies (i\_id, sup\_id) VALUES (9, 4);

INSERT INTO supplies (i\_id, sup\_id) VALUES (9, 2);

INSERT INTO supplies (i\_id, sup\_id) VALUES (10, 1);

INSERT INTO supplies (i\_id, sup\_id) VALUES (10, 6);

INSERT INTO supplies (i\_id, sup\_id) VALUES (11, 1);

INSERT INTO supplies (i\_id, sup\_id) VALUES (12, 4);

INSERT INTO supplies (i\_id, sup\_id) VALUES (13, 1);

INSERT INTO supplies (i\_id, sup\_id) VALUES (14, 2);

INSERT INTO supplies (i\_id, sup\_id) VALUES (14, 6);

INSERT INTO supplies (i\_id, sup\_id) VALUES (15, 3);

INSERT INTO supplies (i\_id, sup\_id) VALUES (15, 5);

INSERT INTO supplies (i\_id, sup\_id) VALUES (16, 6);

INSERT INTO supplies (i\_id, sup\_id) VALUES (16, 4);

INSERT INTO supplies (i\_id, sup\_id) VALUES (17, 2);

INSERT INTO supplies (i\_id, sup\_id) VALUES (17, 1);

INSERT INTO supplies (i\_id, sup\_id) VALUES (18, 6);

INSERT INTO supplies (i\_id, sup\_id) VALUES (19, 2);

INSERT INTO supplies (i\_id, sup\_id) VALUES (20, 1);

INSERT INTO supplies (i\_id, sup\_id) VALUES (20, 6);

SELECT \* FROM SUPPLIES;

INSERT INTO customer (c\_id, c\_name, c\_mail) VALUES (1, 'John Smith', 'john@gmail.com');

INSERT INTO customer (c\_id, c\_name, c\_mail) VALUES (2, 'Emily Brown', 'brown@gmail.com');

INSERT INTO customer (c\_id, c\_name, c\_mail) VALUES (3, 'Jacob Lee', 'lee@gmail.com');

INSERT INTO customer (c\_id, c\_name, c\_mail) VALUES (4, 'Sophie Davis', 'sop.gmail.com');

INSERT INTO customer (c\_id, c\_name, c\_mail) VALUES (5, 'William Johnson', 'johnson@gmail.com');

INSERT INTO customer (c\_id, c\_name, c\_mail) VALUES (6, 'Olivia Taylor', 'taylor@gmail.com');

INSERT INTO customer (c\_id, c\_name, c\_mail) VALUES (7, 'Ethan Wilson', 'wilson@gmail.com');

select \* from customer;

INSERT INTO coupon VALUES (1, 1,5, '31-12-2023');

INSERT INTO coupon VALUES (2, 2,6, '15-06-2024');

INSERT INTO coupon VALUES (3, 3,7, '30-09-2023');

INSERT INTO coupon VALUES (4, 4,8, '31-12-2024');

INSERT INTO coupon VALUES (5, 5,9, '31-03-2025');

INSERT INTO coupon VALUES (6, 6,10, '31-12-2023');

INSERT INTO coupon VALUES (7, 7,11, '30-06-2024');

select \* from coupon;

delete from coupon;

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (1, '1234567890');

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (1, '9876543210');

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (2, '5555555555');

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (3, '1111111111');

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (4, '9999999999');

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (5, '7777777777');

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (6, '8888888888');

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (7, '5553331234');

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (5, '5554444321');

INSERT INTO C\_PHN (c\_id, c\_num) VALUES (2, '5556665678');

SELECT \* FROM C\_PHN;

INSERT INTO contains (s\_id, i\_id) VALUES (1, 1);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 5);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 10);

INSERT INTO contains (s\_id, i\_id) VALUES (4, 15);

INSERT INTO contains (s\_id, i\_id) VALUES (5, 19);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 3);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 6);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 12);

INSERT INTO contains (s\_id, i\_id) VALUES (4, 18);

INSERT INTO contains (s\_id, i\_id) VALUES (5, 13);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 7);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 9);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 16);

INSERT INTO contains (s\_id, i\_id) VALUES (4, 4);

INSERT INTO contains (s\_id, i\_id) VALUES (5, 8);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 11);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 14);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 2);

INSERT INTO contains (s\_id, i\_id) VALUES (4, 17);

INSERT INTO contains (s\_id, i\_id) VALUES (5, 1);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 1);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 2);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 3);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 4);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 5);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 6);

INSERT INTO contains (s\_id, i\_id) VALUES (1, 7);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 8);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 9);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 10);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 11);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 12);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 13);

INSERT INTO contains (s\_id, i\_id) VALUES (2, 14);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 15);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 16);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 17);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 18);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 19);

INSERT INTO contains (s\_id, i\_id) VALUES (3, 20);

SELECT \* FROM CONTAINS;

CREATE TRIGGER COST\_CAL

BEFORE INSERT

ON BOUGHT

FOR EACH ROW

DECLARE

IND\_PRICE INT;

BEGIN

SELECT ITEMS.I\_PRICE INTO IND\_PRICE FROM ITEMS WHERE ITEMS.I\_ID=:NEW.I\_ID;

:NEW.FINAL\_PRICE:=:NEW.QUANTITY\*IND\_PRICE;

END;

/

DELETE FROM BOUGHT;

INSERT INTO BOUGHT VALUES(1,1,20,2);

INSERT INTO BOUGHT VALUES(1,10,14,1);

INSERT INTO BOUGHT VALUES(1,20,57,3);

INSERT INTO BOUGHT VALUES(2,2,40,2);

INSERT INTO BOUGHT VALUES(2,11,22,1);

INSERT INTO BOUGHT VALUES(2,12,90,3);

INSERT INTO BOUGHT VALUES(3,3,10,2);

INSERT INTO BOUGHT VALUES(3,13,16,1);

INSERT INTO BOUGHT VALUES(3,7,21,3);

INSERT INTO BOUGHT VALUES(4,4,75,5);

INSERT INTO BOUGHT VALUES(4,5,16,2);

INSERT INTO BOUGHT VALUES(5,16,60,4);

INSERT INTO BOUGHT VALUES(6,15,50,2);

INSERT INTO BOUGHT VALUES(7,9,18,1);

SELECT \* FROM BOUGHT;

--SELECT BOUGHT.C\_ID,SUM(BOUGHT.FINAL\_PRICE) FROM BOUGHT GROUP BY BOUGHT.C\_ID ORDER BY C\_ID ASC;

CREATE TRIGGER PAMT

BEFORE INSERT

ON PAYMENT

FOR EACH ROW

DECLARE

DISCOUNT INT;

B\_DISCOUNT INT;

BEGIN

SELECT COUPON.C\_OFFER INTO DISCOUNT FROM COUPON WHERE :NEW.C\_ID=COUPON.C\_ID;

SELECT ABC INTO B\_DISCOUNT FROM

(SELECT BOUGHT.C\_ID as dup\_id,SUM(BOUGHT.FINAL\_PRICE) AS ABC

FROM BOUGHT

GROUP BY BOUGHT.C\_ID ORDER BY C\_ID ASC)DT

WHERE :NEW.C\_ID=DT.dup\_id;

:NEW.P\_AMT:=B\_DISCOUNT-DISCOUNT;

END;

/

insert into payment values (1,1,'cash',86,TO\_TIMESTAMP( '03-11-2001 03:22:10'),1);

insert into payment values (2,2,'card',146,TO\_TIMESTAMP( '04-12-2001 04:20:11'),2);

insert into payment values (3,3,'cash',40,TO\_TIMESTAMP( '05-01-2004 06:22:33'),3);

insert into payment values (4,4,'cash',73,TO\_TIMESTAMP( '06-11-2006 03:18:10'),4);

insert into payment values (5,5,'cash',51,TO\_TIMESTAMP( '05-04-2001 06:29:10'),5);

insert into payment values (6,6,'cash',40,TO\_TIMESTAMP( '03-11-2001 03:22:10'),6);

insert into payment values (7,7,'cash',7,TO\_TIMESTAMP( '03-06-2001 03:11:10'),7);

select \* from payment;

delete from payment;S