**-- E-commerce website data in the database**

1. ,2) create table ,

CREATE TABLE supplier (

SUPP\_ID INT PRIMARY KEY,

SUPP\_NAME VARCHAR(50) NOT NULL,

SUPP\_CITY VARCHAR(50) NOT NULL,

SUPP\_PHONE VARCHAR(50) NOT NULL

);

CREATE TABLE customer (

CUS\_ID INT PRIMARY KEY,

CUS\_NAME VARCHAR(20) NOT NULL,

CUS\_PHONE VARCHAR(10) NOT NULL,

CUS\_CITY VARCHAR(30) NOT NULL,

CUS\_GENDER CHAR

);

CREATE TABLE category (

CAT\_ID INT PRIMARY KEY,

CAT\_NAME VARCHAR(20) NOT NULL

);

CREATE TABLE product (

PRO\_ID INT PRIMARY KEY,

PRO\_NAME VARCHAR(20) NOT NULL DEFAULT 'Dummy',

PRO\_DESC VARCHAR(60),

CAT\_ID INT,

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

);

CREATE TABLE supplier\_pricing (

PRICING\_ID INT PRIMARY KEY,

PRO\_ID INT,

SUPP\_ID INT,

SUPP\_PRICE INT DEFAULT 0,

FOREIGN KEY (PRO\_ID) REFERENCES product(PRO\_ID),

FOREIGN KEY (SUPP\_ID) REFERENCES supplier(SUPP\_ID)

);

CREATE TABLE orders (

ORD\_ID INT PRIMARY KEY,

ORD\_AMOUNT INT NOT NULL,

ORD\_DATE DATE NOT NULL,

CUS\_ID INT,

PRICING\_ID INT,

FOREIGN KEY (CUS\_ID) REFERENCES customer(CUS\_ID),

FOREIGN KEY (PRICING\_ID) REFERENCES supplier\_pricing(PRICING\_ID)

);

CREATE TABLE rating (

RAT\_ID INT PRIMARY KEY,

ORD\_ID INT,

RAT\_RATSTARS INT NOT NULL,

FOREIGN KEY (ORD\_ID) REFERENCES "order"(ORD\_ID)

);

3).insert the following data in table ..

-- Insert data into the supplier table

INSERT INTO supplier (SUPP\_ID, SUPP\_NAME, SUPP\_CITY, SUPP\_PHONE)

VALUES

(1, 'Rajesh Retails', 'Delhi', '1234567890'),

(2, 'Appario Ltd.', 'Mumbai', '2589631470'),

(3, 'Knome products', 'Banglore', '9785462315'),

(4, 'Bansal Retails', 'Kochi', '8975463285'),

(5, 'Mittal Ltd.', 'Lucknow', '7898456532');

-- Insert data into the customer table

INSERT INTO customer (CUS\_ID, CUS\_NAME, CUS\_PHONE, CUS\_CITY, CUS\_GENDER)

VALUES

(1, 'AAKASH', '9999999999', 'DELHI', 'M'),

(2, 'AMAN', '9785463215', 'NOIDA', 'M'),

(3, 'NEHA', '9999999999', 'MUMBAI', 'F'),

(4, 'MEGHA', '9994562399', 'KOLKATA', 'F'),

(5, 'PULKIT', '7895999999', 'LUCKNOW', 'M');

-- Insert data into the category table

INSERT INTO category (CAT\_ID, CAT\_NAME)

VALUES

(1, 'BOOKS'),

(2, 'GAMES'),

(3, 'GROCERIES'),

(4, 'ELECTRONICS'),

(5, 'CLOTHES');

-- Insert data into the product table

INSERT INTO product (PRO\_ID, PRO\_NAME, PRO\_DESC, CAT\_ID)

VALUES

(1, 'GTA V', 'Windows 7 and above with i5 processor and 8GB RAM', 2),

(2, 'TSHIRT', 'SIZE-L with Black, Blue and White variations', 5),

(3, 'ROG LAPTOP', 'Windows 10 with 15inch screen, i7 processor, 1TB SSD', 4),

(4, 'OATS', 'Highly Nutritious from Nestle', 3),

(5, 'HARRY POTTER', 'Best Collection of all time by J.K Rowling', 1),

(6, 'MILK', '1L Toned Milk', 3),

(7, 'Boat Earphones', '1.5Meter long Dolby Atmos', 4),

(8, 'Jeans', 'Stretchable Denim Jeans with various sizes and color', 5),

(9, 'Project IGI', 'compatible with Windows 7 and above', 2),

(10, 'Hoodie', 'Black GUCCI for 13 yrs and above', 5),

(11, 'Rich Dad Poor Dad', 'Written by Robert Kiyosaki', 1),

(12, 'Train Your Brain', 'By Shireen Stephen', 1);

-- Insert data into the supplier\_pricing table

INSERT INTO supplier\_pricing (PRICING\_ID, PRO\_ID, SUPP\_ID, SUPP\_PRICE)

VALUES

(1, 1, 2, 1500),

(2, 3, 5, 30000),

(3, 5, 1, 3000),

(4, 2, 3, 2500),

(5, 4, 1, 1000),

(6, 12, 2, 780),

(7, 12, 4, 789),

(8, 3, 1, 31000),

(9, 1, 5, 1450),

(10, 4, 2, 999),

(11, 7, 3, 549),

(12, 7, 4, 529),

(13, 6, 2, 105),

(14, 6, 1, 99),

(15, 2, 5, 2999),

(16, 5, 2, 2999);

-- Insert data into the order table

INSERT INTO orders (ORD\_ID, ORD\_AMOUNT, ORD\_DATE, CUS\_ID, PRICING\_ID)

VALUES

(101, 1500, '2021-10-06', 2, 1),

(102, 1000, '2021-10-12', 3, 5),

(103, 30000, '2021-09-16', 5, 2),

(104, 1500, '2021-10-05', 1, 1),

(105, 3000, '2021-08-16', 4, 3),

(106, 1450, '2021-08-18', 1, 9),

(107, 789, '2021-09-01', 3, 7),

(108, 780, '2021-09-07', 5, 6),

(109, 3000, '2021-10-10', 5, 3),

(110, 2500, '2021-09-10', 2, 4),

(111, 1000, '2021-09-15', 4, 5),

(112, 789, '2021-09-16', 4, 7),

(113, 31000, '2021-09-16', 1, 8),

(114, 1000, '2021-09-16', 3, 5),

(115, 3000, '2021-09-16', 5, 3),

(116, 99, '2021-09-17', 2, 14);

-- Insert data into the rating table

INSERT INTO rating (RAT\_ID, ORD\_ID, RAT\_RATSTARS)

VALUES

(1, 101, 4),

(2, 102, 3),

(3, 103, 1),

(4, 104, 2),

(5, 105, 4),

(6, 106, 3),

(7, 107, 4),

(8, 108, 4),

(9, 109, 3),

(10, 110, 5),

(11, 111, 3),

(12, 112, 4),

(13, 113, 2),

(14, 114, 1),

(15, 115, 1),

(16, 116, 0);

-- 4 Display the total number of customers based on gender who have placed individual orders of worth at least Rs.3000.

SELECT orders.ORD\_ID, product.PRO\_NAME

FROM orders

JOIN supplier\_pricing ON orders.PRICING\_ID = supplier\_pricing.PRICING\_ID

JOIN product ON supplier\_pricing.PRO\_ID = product.PRO\_ID

WHERE orders.ORD\_DATE > '2021-10-05';

-- 5 Display all the orders along with product name ordered by a customer having Customer\_Id=2.--

SELECT orders.ORD\_ID, orders.ORD\_AMOUNT, orders.ORD\_DATE, customer.CUS\_NAME, product.PRO\_NAME

FROM orders

JOIN customer ON orders.CUS\_ID = customer.CUS\_ID

JOIN supplier\_pricing ON orders.PRICING\_ID = supplier\_pricing.PRICING\_ID

JOIN product ON supplier\_pricing.PRO\_ID = product.PRO\_ID

WHERE customer.CUS\_ID = 2;

-- 6 Display the Supplier details who can supply more than one product.

SELECT supplier.\*

FROM supplier

JOIN supplier\_pricing ON supplier.SUPP\_ID = supplier\_pricing.SUPP\_ID

GROUP BY supplier.SUPP\_ID

HAVING COUNT(DISTINCT supplier\_pricing.PRO\_ID) > 1;

-- 7 Find the least expensive product from each category and print the table with category id, name, product name, and price of the product.

SELECT category.CAT\_ID, category.CAT\_NAME, product.PRO\_NAME, MIN(supplier\_pricing.SUPP\_PRICE) AS Min\_Price

FROM category

JOIN product ON category.CAT\_ID = product.CAT\_ID

JOIN supplier\_pricing ON product.PRO\_ID = supplier\_pricing.PRO\_ID

GROUP BY category.CAT\_ID, product.PRO\_NAME;

-- 8 Display the Id and Name of the Product ordered after “2021-10-05”.

SELECT orders.ORD\_ID, product.PRO\_NAME

FROM orders

JOIN supplier\_pricing ON orders.PRICING\_ID = supplier\_pricing.PRICING\_ID

JOIN product ON supplier\_pricing.PRO\_ID = product.PRO\_ID

WHERE orders.ORD\_DATE > '2021-10-05';

-- 9 Display customer name and gender whose names start or end with character 'A'.

SELECT CUS\_NAME, CUS\_GENDER

FROM customer

WHERE CUS\_NAME LIKE 'A%' OR CUS\_NAME LIKE '%A';

-- 10 Create a stored procedure to display supplier id, name, Rating (Average rating of all the products sold by every customer), and Type\_of\_Service.

DELIMITER //

CREATE PROCEDURE GetSupplierRating()

BEGIN

SELECT

supplier.SUPP\_ID,

supplier.SUPP\_NAME,

AVG(rating.RAT\_RATSTARS) AS Avg\_Rating,

CASE

WHEN AVG(rating.RAT\_RATSTARS) = 5 THEN 'Excellent Service'

WHEN AVG(rating.RAT\_RATSTARS) > 4 THEN 'Good Service'

WHEN AVG(rating.RAT\_RATSTARS) > 2 THEN 'Average Service'

ELSE 'Poor Service'

END AS Type\_of\_Service

FROM supplier

LEFT JOIN supplier\_pricing ON supplier.SUPP\_ID = supplier\_pricing.SUPP\_ID

LEFT JOIN orders ON supplier\_pricing.PRICING\_ID = orders.PRICING\_ID

LEFT JOIN rating ON orders.ORD\_ID = rating.ORD\_ID

GROUP BY supplier.SUPP\_ID;

END //

DELIMITER ;

CALL GetSupplierRating();