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

SELECT CUS\_GENDER, COUNT(\*) AS TotalCustomers

FROM customer

WHERE CUS\_ID IN (

SELECT DISTINCT CUS\_ID

FROM "order"

WHERE ORD\_AMOUNT >= 3000

)

GROUP BY CUS\_GENDER;

Displaying all the orders along with the product name ordered by a customer with Customer\_Id = 2.

SELECT O.ORD\_ID, P.PRO\_NAME

FROM "order" O

JOIN product P ON O.PRICING\_ID = P.PRO\_ID

WHERE O.CUS\_ID = 2;

Displaying the Supplier details who can supply more than one product.

SELECT S.\*

FROM supplier S

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

GROUP BY S.SUPP\_ID, S.SUPP\_NAME, S.SUPP\_CITY, S.SUPP\_PHONE

HAVING COUNT(SP.PRO\_ID) > 1;

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

SELECT C.CAT\_ID, C.CAT\_NAME, P.PRO\_NAME, P.PRO\_DESC, MIN(SP.SUPP\_PRICE) AS LowestPrice

FROM category C

JOIN product P ON C.CAT\_ID = P.CAT\_ID

JOIN supplier\_pricing SP ON P.PRO\_ID = SP.PRO\_ID

GROUP BY C.CAT\_ID, C.CAT\_NAME, P.PRO\_NAME, P.PRO\_DESC;

Displaying the Id and Name of the Product ordered after "2021-10-05".

SELECT P.PRO\_ID, P.PRO\_NAME

FROM "order" O

JOIN product P ON O.PRICING\_ID = P.PRO\_ID

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

Displaying customer name and gender whose names start or end with the character 'A'.

SELECT CUS\_NAME, CUS\_GENDER

FROM customer

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

Creating a stored procedure to display supplier id, name, Rating (average rating of all the products sold by every customer), and Type\_of\_Service. For Type\_of\_Service, if rating = 5, print "Excellent Service"; if rating > 4, print "Good Service"; if rating > 2, print "Average Service"; else print "Poor Service". Note that there should be one rating per supplier.

DELIMITER //

CREATE PROCEDURE CalculateSupplierServiceType()

BEGIN

SELECT

SP.SUPP\_ID,

S.SUPP\_NAME,

IFNULL(AVG(R.RAT\_RATSTARS), 0) AS AverageRating,

CASE

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

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

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

ELSE 'Poor Service'

END AS Type\_of\_Service

FROM supplier S

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

LEFT JOIN "order" O ON SP.PRICING\_ID = O.PRICING\_ID

LEFT JOIN rating R ON O.ORD\_ID = R.ORD\_ID

GROUP BY S.SUPP\_ID, S.SUPP\_NAME;

END //

DELIMITER ;

You can call the stored procedure using:

CALL CalculateSupplierServiceType();