**Queries →**

Problem 3: Display the total number of customers based on gender who have placed orders of worth at least Rs.3000.

Solution: select count(\*), c.CUS\_GENDER from customer as c inner join `order` as o on c.CUS\_ID = o.CUS\_ID where o.ORD\_AMOUNT>=3000 group by c.CUS\_GENDER;

Problem 4: Display all the orders along with product name ordered by a customer having Customer\_Id=2

Solution: select Ord.ORD\_ID as Order\_ID, Ord.ORD\_AMOUNT as Oredr\_Amount, Ord.ORD\_DATE as Order\_Date, Product.PRO\_NAME as Product\_Name, ord.CUS\_ID as Customer\_ID from `Order` as Ord inner join Supplier\_pricing as Sp on Ord.Pricing\_ID = Sp.Pricing\_ID inner join Product on Product.Pro\_ID = Sp.Pro\_Id where Ord.CUS\_ID=2;

Problem 5: Display the Supplier details who can supply more than one product.

Solution: select supplier.\* from supplier where supplier.supp\_id in (select supp\_id from supplier\_pricing group by supp\_id having count(supp\_id) > 1) group by supplier.supp\_id;

Problem 6: Find the least expensive product from each category and print the table with category id, name, product(Category) name and price of the product

Solution: select catg.Cat\_Id as Category\_Id, catg.cat\_Name as Category\_Name, min(sp.Supp\_Price) as Price\_Of\_Product from category catg inner join product prod on catg.Cat\_Id = prod.Cat\_Id inner join supplier\_pricing sp on sp.Pro\_Id = prod.Pro\_Id group by catg.Cat\_Id;

Problem 7: Display the Id and Name of the Product ordered after “2021-10-05”.

Solution: select prod.Pro\_Id, prod.Pro\_Name, o.Ord\_Date from product prod inner join supplier\_pricing sp on sp.Pro\_Id = prod.Pro\_Id inner join `order-directory`.order o on o.Pricing\_Id = sp.Pricing\_Id where o.Ord\_Date > '2021-10-05';

Problem 8: Display customer name and gender whose names start or end with character 'A'

Solution: select Cus\_Name, Cus\_gender from customer where Cus\_Name like 'A%' or Cus\_Name like '%A';

Problem 9: Create a stored procedure to display supplier id, name, rating 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”.

Solution: select supp.Supp\_id, supp.Supp\_Name, rat.Rat\_Ratstars, case

when rat.Rat\_Ratstars = 5 then 'Excellent Service'

when rat.Rat\_Ratstars = 4 then 'Excellent Service'

when rat.Rat\_Ratstars = 2 then 'Excellent Service'

else 'poor Service'

end as Type\_pf\_Service

from supplier supp

inner join supplier\_pricing sp on sp.Supp\_Id = supp.Supp\_Id

inner join `order-directory`.order o on sp.pricing\_Id = o.Pricing\_Id

inner join rating rat on rat.Ord\_Id = o.Ord\_Id;

CREATION OF A STORED PROCEDURE

DELIMITER //

CREATE PROCEDURE sp\_rating ()

BEGIN

SELECT

supp.Supp\_id,

supp.Supp\_Name,

rat.Rat\_Ratstars,

CASE

WHEN rat.Rat\_Ratstars = 5 THEN 'Excellent Service'

WHEN rat.Rat\_Ratstars = 4 THEN 'Excellent Service'

WHEN rat.Rat\_Ratstars = 2 THEN 'Excellent Service'

ELSE 'poor Service'

END AS Type\_pf\_Service

FROM

supplier supp

INNER JOIN

supplier\_pricing sp ON sp.Supp\_Id = supp.Supp\_Id

INNER JOIN

`order-directory`.order o ON sp.pricing\_Id = o.Pricing\_Id

INNER JOIN

rating rat ON rat.Ord\_Id = o.Ord\_Id

END //

DELIMITER;