**Q3: Display the total number of customers based on gender who have placed orders of worth at least Rs.3000.**

SELECT COUNT(\*) AS "TOTAL CUSTOMERS", cus\_gender from customer

where customer.cus\_id in(SELECT distinct(cus\_id) from `order` where ord\_amount >= 3000)

group by cus\_gender;

**Q4: Display all the orders along with product name ordered by a customer having Customer\_Id=2**

select product.pro\_name, `order`.\* from `order`, supplier\_pricing, product

where `order`.cus\_id=2 and `order`.pricing\_id=supplier\_pricing.pricing\_id and

supplier\_pricing.pro\_id=product.pro\_id;

**Q5: Display the Supplier details of who is supplying more than one product.**

SELECT supplier.\*

FROM supplier, supplier\_pricing

WHERE supplier.supp\_id = supplier\_pricing.supp\_id

GROUP BY supplier\_pricing.supp\_id

HAVING COUNT(supplier\_pricing.supp\_id ) > 2;

**Q6: Find the least expensive product from each category and print the table with category id, name, and price of the product**

select category.CAT\_ID,CAT\_NAME,min(supplier\_pricing.SUPP\_PRICE)

from supplier\_pricing

join product on product.PRO\_ID=supplier\_pricing.PRO\_ID

join category on category.CAT\_ID=product.CAT\_ID

group by product.CAT\_ID;

**Q7: Display the Id and Name of the Product ordered after “2021-10-05”.**

select product.pro\_id,product.pro\_name from `order` inner join supplier\_pricing on supplier\_pricing.pricing\_id=`order`.pricing\_id inner join product

on product.pro\_id=supplier\_pricing.pro\_id where `order`.ord\_date>"2021-10-05";

**Q8: Display customer name and gender whose names start or end with character 'A'.**

select customer.cus\_name,customer.cus\_gender from customer where customer.cus\_name like 'A%' or customer.cus\_name like '%A';

**Q9: Create a stored procedure to display supplier id, name, rating and Type\_of\_supplier. If rating >4 then “Genuine Supplier” if rating >2 “Average**

SELECT

supplier.SUPP\_ID,

supplier.SUPP\_NAME,

CASE

WHEN avg(rating.RAT\_RATSTARS) > 4 THEN 'Genuine Supplier'

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

END as rate

FROM

rating,

`order`,

supplier,

supplier\_pricing

WHERE

rating.ORD\_ID = `order`.ORD\_ID

AND supplier.SUPP\_ID = supplier\_pricing.SUPP\_ID

AND supplier\_pricing.PRICING\_ID = `order`.PRICING\_ID

GROUP BY supplier.SUPP\_ID;