

1. Find latest order by customer name.

Query-

```
SELECT c.Name, MAX(o.orderdate) AS latest_order
FROM customers c
INNER JOIN orders o
ON c.customerid = o.customerid
GROUP BY c.customerid;
```

Output-

	Name	latest_order
▶	Aarav Sharma	2025-04-28
	Isha Gupta	2025-06-25
	Rohan Mehta	2025-08-11
	Priya Nair	2025-03-11
	Kunal Kapoor	2021-02-15
	Ananya Singh	2025-08-16
	Vikram Joshi	2024-11-14

2. Find oldest order by customer name.

Query-

```
SELECT c.Name, min(o.orderdate) AS oldest_order
FROM customers c
INNER JOIN orders o
ON c.customerid = o.customerid
GROUP BY c.customerid
order by oldest_order;
```

Output-

	Name	oldest_order
▶	Kunal Kapoor	2021-02-15
	Meera Kulkarni	2021-03-10
	Ravi Shetty	2021-05-21
	Sneha Pillai	2021-07-04
	Kiran Dev	2021-08-12
	Payal Jadhav	2021-09-30
	Krishnan Nair	2021-11-18

3. Show highest price categorywise.

Query-

```
select ct.categoryname, max(p.price) as High_Value  
from categories ct  
inner join Products p  
on ct.categoryid = p.categoryid  
group by ct.categoryname;
```

Output-

	categoryname	High_Value
▶	Electronics	45999.00
	Clothing	3499.00
	Home & Kitchen	7499.00
	Books	599.00
	Beauty	2499.00
	Sports	3499.00

4. Show highest price productwise.

Query-

```
select p.name, max(P1.price) as high_value  
from products p  
inner join products p1  
on p.productid = p1.productid  
group by p.name  
order by high_value desc;
```

Output-

	name	high_value
▶	Laptop Pro 14	45999.00
	4K Smart TV 43"	24999.00
	Digital Camera 24MP	19999.00
	Smartphone X1	15499.00
	LED Monitor 24"	9999.00
	Microwave Oven 20L	7499.00
	Water Purifier	7499.00

5. Show category wise product wise highest selling price product.

Query-

```
select ct.categoryname as cat, p.name, p.price as value  
from categories ct  
left join products p
```

```
on ct.categoryid = p.categoryid  
where p.price = (select max(p1.price) from products p1)  
order by cat;
```

output-

	cat	name	value
▶	Electronics	Laptop Pro 14	45999.00

6. Show category wise product wise lowest selling price product.

Query-

```
select ct.categoryname as cat, p.name, p.price as value  
from categories ct  
left join products p  
on ct.categoryid = p.categoryid  
where p.price = (select min(p1.price) from products p1)  
order by cat;
```

Output-

	cat	name	value
▶	Beauty	Lip Balm	99.00

7. Show category wise product wise out of stock products along with their prices.

Query-

```
select ct.categoryname as cat, p.name, p.price, p.stockquantity as Stock_Status  
from categories ct  
left join products p  
on ct.categoryid = p.categoryid  
where p.stockquantity = 0  
order by p.price;
```

output-

	cat	name	price	Stock_Status
▶	Beauty	Sunscreen SPF 50	399.00	0
	Clothing	Winter Sweater Wool	1599.00	0
	Clothing	Denim Jacket	1999.00	0
	Clothing	Winter Jacket	2499.00	0

8. Show category wise product wise stock status along with their prices.

Query-

```
select ct.categoryname, p.name, p.price, p.stockquantity as Stock_Status
from categories ct
left join products p
on ct.categoryid = p.categoryid
order by Stock_Status desc;
```

output-

	categoryname	name	price	Stock_Status
▶	Electronics	Wireless Headphones Z	2999.00	200
	Beauty	Hair Oil 200ml	199.00	200
	Beauty	Herbal Shampoo 500ml	249.00	200
	Beauty	Lip Balm	99.00	200
	Beauty	Hand Cream 100ml	149.00	190
	Beauty	Face Wash Neem	149.00	190
	Beauty	Conditioner 500ml	249.00	190

  

	categoryname	name	price	Stock_Status
	Home & Kitchen	Water Purifier	7499.00	60
	Home & Kitchen	Microwave Oven 20L	7499.00	60
	Electronics	Laptop Pro 14	45999.00	50
	Clothing	Winter Sweater Wool	1599.00	0
	Clothing	Denim Jacket	1999.00	0
	Clothing	Winter Jacket	2499.00	0
	Beauty	Sunscreen SPF 50	399.00	0

9. Show customers placed more than one order.

Query-

```
select cs.customerid, cs.name, count(o.orderid) as No_Of_Orders
from customers cs
left join orders o
on cs.customerid = o.customerid
group by cs.customerid
having No_Of_Orders > 1;
```

output-

	customerid	name	No_Of_Orders
▶	12	Meera Kulkarni	4
	25	Gaurav Sinha	4
	39	Arjun Ghosh	5
	75	Chirag Shah	4

10. Show customers placed single order.

Query-

```
select cs.customerid, cs.name  
from customers cs  
left join orders o  
on cs.customerid = o.customerid  
group by cs.customerid  
having count(o.orderid) = 1;
```

output-

	customerid	name
▶	1	Aarav Sharma
	2	Isha Gupta
	3	Rohan Mehta
	4	Priya Nair
	6	Ananya Singh
	9	Siddharth Menon
	10	Ritika Jain

11. Show customers who never placed a single order.

Query-

```
select cs.customerid, cs.name, o.orderid  
from customers cs  
left join orders o  
on cs.customerid = o.customerid  
where o.orderid is null;
```

output-

Result Grid | Filter Rows:  Export: Wrap Cell Co

	customerid	name	orderid
	5	Kunal Kapoor	NULL
	7	Vikram Joshi	NULL
▶	8	Neha Verma	NULL
	11	Aditya Rao	NULL
	18	Pooja Reddy	NULL
	22	Shruti Desai	NULL
	29	Rajeev Raina	NULL

12. Show customers who ordered products more than 1 qty.

Query-

```
select c.name, o.orderid, count(od.quantity) as OrdQTY
from customers c
inner join orders o on c.customerid = o.customerid
left join orderdetails od on o.orderid = od.order_id
group by o.orderid
having ordqty > 1;
```

output-

Result Grid | Filter Rows:  Export: Wrap Cell Co

	name	orderid	OrdQTY
▶	Meera Kulkarni	2	2
	Tanvi Shah	59	2
	Arjun Ghosh	68	2
	Anurag Chopra	19	2
	Krishnan Nair	7	2
	Omprakash Tripathi	14	2

13. Show products ordered more than 1 qty.

Query-

```
select p.name, count(od.quantity) as OrdQTY
from products p
inner join orderdetails od on p.productid = od.product_id
group by p.name
having ordqty > 1;
```

output-

	name	OrdQTY
▶	Cricket Bat Pro	2
	Romantic Novel Vol. 2	2
	Microwave Oven 20L	2
	Yoga Mat Comfort	2
	Historical Novel	2

14. Show top 5 highest selling products.

Query-

```
select p.productid, p.name, sum(od.quantity) as OrdQTY
from products p
inner join orderdetails od on p.productid = od.product_id
group by p.productid
order by ordqty desc limit 5;
```

output-

	productid	name	OrdQTY
▶	18	Yoga Mat Comfort	9
	11	Romantic Novel Vol. 2	6
	28	Historical Novel	6
	42	Raincoat Waterproof	5
	27	Dinner Set 24pcs	5

15. Show top 5 highest revenue products per order.

Query-

```
select p.productid, p.name, sum(od.quantity*p.price) as Top_Revenue
from products p
inner join orderdetails od on p.productid = od.product_id
group by p.productid, p.name
order by Top_Revenue desc limit 5;
```

output-

	productid	name	Top_Revenue
▶	2	Laptop Pro 14	137997.00
	37	4K Smart TV 43"	74997.00
	1	Smartphone X1	61996.00
	57	Digital Camera 24MP	39998.00
	8	Microwave Oven 20L	29996.00

16. Show products with 5\* ratings.

Query-

```
select p.productid, p.name, max(rw.rating) as TopRated
from products p
inner join reviews rw on p.productid = rw.productid
group by p.productid
having toprated > 4;
```

output-

	productid	name	TopRated
▶	1	Smartphone X1	5
	2	Laptop Pro 14	5
	16	Professional Football	5
	17	Cricket Bat Pro	5
	21	Gaming Mouse RGB	5
	27	Dinner Set 24pcs	5
	34	Badminton Racket	5

17. Show products with comments along with ratings from top to bottom.

Query-

```
select p.productid, p.name, rw.comment, rw.rating
from products p
inner join reviews rw on p.productid = rw.productid
order by rw.rating desc;
```

output-

	productid	name	comment	rating
▶	65	Cookbook Recipes	Great collection of recipes!	5
	17	Cricket Bat Pro	Exceeded expectations.	5
	16	Professional Football	Very durable ball.	5
	34	Badminton Racket	Top quality! Highly recommend.	5
	1	Smartphone X1	Absolutely love this phone!	5
	55	Wireless Earbuds	Awesome sound quality.	5
	57	Digital Camera 24MP	Perfect for beginners.	5

	productid	name	comment	rating
	44	Induction Cooktop	Stopped working in a month.	2
	45	Water Purifier	Leaks often.	2
	11	Romantic Novel Vol. 2	Disappointed with print quality.	2
	18	Yoga Mat Comfort	Gets slippery during use.	2
	68	Nail Polish Set	Colors faded quickly.	2
	50	Hand Cream 100ml	Too greasy for hands.	2
	3	Wireless Headphones Z	Broke after a week.	1

18. Show products along with discounts.

Query-

```
select p.name, p.price, d.discountamount  
from products p  
left join discounts d  
on p.productid = d.productid;
```

output-

	name	price	discountamount
▶	Smartphone X1	15499.00	150.00
	Laptop Pro 14	45999.00	180.80
	Wireless Headphones Z	2999.00	45.67
	Cotton T-Shirt Classic	499.00	123.00
	Slim Fit Jeans	1499.00	75.25
	Winter Jacket	2499.00	130.50
	Mixer Grinder Pro	3499.00	145.60

19. Show top 10 products with highest discounts.

Query-

```
select p.name, p.price, max(d.discountamount) as maxDisc  
from products p  
left join discounts d  
on p.productid = d.productid  
group by p.productid  
order by maxdisc desc limit 10;
```

	name	price	maxDisc
	Badminton Racket	1199.00	222.22
	Raincoat Waterproof	1299.00	210.50
	Hair Serum	399.00	210.10
	Chinos Pants	1499.00	200.00
	Sunscreen SPF 50	399.00	199.99
	Adventure Novel	399.00	199.99
	Travel Guide	499.00	199.99

20. Show products without any discount.

Query-

```
select p.name, p.price, d.discountamount as disc  
from products p  
left join discounts d  
on p.productid = d.productid  
having Disc = 0;
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	name	price	disc
▶	Herbal Shampoo 500ml	249.00	0.00
	Hair Oil 200ml	199.00	0.00
	Face Wash Neem	149.00	0.00
	Conditioner 500ml	249.00	0.00
	Hand Cream 100ml	149.00	0.00
	Lip Balm	99.00	0.00
	Nail Polish Set	249.00	0.00

21. Show product count per category

Query-

```
Select c.categoryname, count(p.productID) as No_Of_Products
From categories c
Left join products p
On c.categoryid = p.categoryid
group by c.categoryid;
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	categoryname	No_Of_Products
▶	Electronics	15
	Clothing	15
	Home & Kitchen	14
	Books	12
	Beauty	12
	Sports	Beauty

22. Show customers placed orders in year 2024

Query-

```
SELECT c.Name, o.orderdate
FROM customers c
INNER JOIN orders o
ON c.customerid = o.customerid
where year(o.orderdate) = 2024;
```

output-

Result Grid | Filter Rows:  | Export: | Wrap Cell Co

	Name	orderdate
	Manish Patel	2024-05-27
	Pallavi Menon	2024-06-15
	Abhay Mishra	2024-07-23
	Kirti Malviya	2024-08-08
	Arvind Menon	2024-09-12
	Aditi Sood	2024-10-03
	Arjun Ghosh	2024-11-14
	Chirag Shah	2024-12-27

23. Show customers with order status.

Query-

```
SELECT c.name,
CASE
    WHEN o.customerid IS NOT NULL THEN 'Ordered'
    ELSE 'Not Ordered'
END AS Order_status
FROM customers c
LEFT JOIN orders o ON c.customerid = o.customerid;
```

Output-

Result Grid | Filter Rows:  | Export: | Wrap Cell Co

	name	Order_status
	Chirag Shah	Ordered
	Chirag Shah	Ordered
	Chirag Shah	Ordered
	Aditi Sood	Ordered
	Mahesh Pillai	Ordered
	Ipsita Patil	Not Ordered
	Keshav Bhat	Not Ordered
	Pallavi Rane	Ordered

24. Show customers who never ordered.

Query-

```
SELECT c.name,
CASE
    WHEN o.customerid IS NOT NULL THEN 'Ordered'
    ELSE 'Not Ordered'
END AS Order_status
FROM customers c
LEFT JOIN orders o ON c.customerid = o.customerid
Where o.customerid is null;
```

Output-

	name	Order_status
▶	Kunal Kapoor	Not Ordered
	Vikram Joshi	Not Ordered
	Neha Verma	Not Ordered
	Aditya Rao	Not Ordered
	Pooja Reddy	Not Ordered
	Shruti Desai	Not Ordered
	Rajeev Raina	Not Ordered
	Reema Fernandes	Not Ordered
	Kiran Dev	Not Ordered

25. Show products never ordered.

Query-

```
SELECT p.name,
CASE
    WHEN od.product_id IS NOT NULL THEN 'Ordered'
    ELSE 'Not Ordered'
END AS Order_status
FROM products p
LEFT JOIN orderdetails od ON p.productid = od.product_id
Where od.product_id is null;
```

Output-

	name	Order_status
▶	Luxury Perfume 50ml	Not Ordered
	Electric Kettle	Not Ordered
	Rice Cooker 1.8L	Not Ordered
	Ceiling Fan 1200mm	Not Ordered
	Chinos Pants	Not Ordered

26. Show customers whose orders never delivered.

Query-

```
select c.name, o.orderdate
from customers c
inner join orders o on c.customerid = o.customerid
left join shipping s on o.orderid = s.orderid
where deliverydate is Null;
```

output-

Result Grid | Filter Rows: [ ] | Export: [ ] | Wrap Cell Co

	name	orderdate
▶	Sneha Pillai	2021-07-04
	Gaurav Sinha	2022-02-28
	Vivek Anand	2023-05-19
	Arvind Menon	2024-09-12

27. Show customers details with their order\_status as low, mid, high.

Query-

```
SELECT c.name, p.name, (p.price*od.quantity) as amount,
CASE
    WHEN (p.price*od.quantity) < 10000 then 'LOW'
        When (p.price*od.quantity) between 10000 and 25000 then 'MID'
        Else 'HIGH'
END AS Order_Status
FROM customers c
inner JOIN orders o ON c.customerid = o.customerid
left join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid;
```

output-

Result Grid | Filter Rows: [ ] | Export: [ ] | Wrap Cell Co

	name	name	amount	Order_Status
▶	Aarav Sharma	Pressure Cooker 5L	6396.00	LOW
	Isha Gupta	Conditioner 500ml	747.00	LOW
	Rohan Mehta	Dinner Set 24pcs	14995.00	MID
	Priya Nair	Smartwatch Lite	3999.00	LOW
	Ananya Singh	Self-help Bestseller	998.00	LOW
	Siddharth Menon	Digital Camera 24MP	39998.00	HIGH
	Ritika Jain	Volleyball	2397.00	LOW
	Meera Kulkarni	External Hard Drive 1TB	11997.00	MID
	Meera Kulkarni	Romantic Novel Vol. 2	698.00	LOW

28. Show top 5 customers with highest paid order.

Query-

```
SELECT c.name, p.name, (p.price*od.quantity) as Amount
From customers c
inner JOIN orders o ON c.customerid = o.customerid
left join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid
order by Amount desc limit 5;
```

output-

Result Grid | Filter Rows:  | Export: | Wrap Cell Co

	name	name	Amount
	Chirag Shah	Laptop Pro 14	137997.00
	Arjun Ghosh	4K Smart TV 43"	74997.00
▶	Rahul Choudhary	Smartphone X1	61996.00
	Siddharth Menon	Digital Camera 24MP	39998.00
	Swati Chauhan	Ethnic Saree	17495.00

29. Show top 5 customers who availed maximum discount on order.

Query-

```
SELECT c.name, p.name, sum(d.discountamount) as Disc_Amt
From customers c
inner JOIN orders o ON c.customerid = o.customerid
left join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid
left join discounts d on p.productID = d.productid
group by c.name, p.name
order by Disc_Amt desc limit 5;
```

output-

Result Grid | Filter Rows:  | Export: | Wrap Cell Co

	name	name	Disc_Amt
▶	Parul Yadav	Professional Football	310.50
	Arjun Ghosh	Non-stick Cooking Pan	299.99
	Swati Chauhan	Ethnic Saree	230.10
	Simran Gill	Badminton Racket	222.22
	Krishnan Nair	Raincoat Waterproof	210.50

30. Show customers ordered from specific category.

Query-

```
Select c.name, cat.categoryname
From customers c
inner JOIN orders o ON c.customerid = o.customerid
left join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid
left join categories cat on p.categoryid = cat.categoryid
where cat.categoryname = "Electronics";
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	name	categoryname
▶	Rahul Choudhary	Electronics
	Chirag Shah	Electronics
	Akhil Prasad	Electronics
	Priya Nair	Electronics
	Poonam Dey	Electronics
	Anjali Mathur	Electronics
	Arjun Ghosh	Electronics
	Harsh Vardhan	Electronics
	Varun Sharma	Electronics

31. Show customers ordered from specific product.

Query-

```
Select c.name
From customers c
inner JOIN orders o ON c.customerid = o.customerid
left join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid
where p.name like '%jacket%';
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	name
▶	Meera Kulkarni
	Suresh Agarwal

32. Show details of customer when they placed their recent order, this year, last year, previous.

Query-

```
SELECT c.Name,
CASE
    WHEN year(max(o.orderdate)) > 2024 then 'This Year'
        When year(max(o.orderdate)) > 2023 then 'Last Year'
        Else 'Old Order'
END AS Order_Status
from customers c
INNER JOIN orders o
ON c.customerid = o.customerid
group by c.name;
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Cc

	Name	Order_Status
▶	Gaurav Sinha	Old Order
	Meera Kulkarni	This Year
	Ravi Shetty	Old Order
	Sneha Pillai	Old Order
	Payal Jadhav	Old Order
	Krishnan Nair	Old Order
	Arjun Ghosh	This Year
	Simran Gill	Old Order
	Chirag Shah	Last Year

33. Show customers with product availability status.

Query-

Select c.name, p.name,

Case

```
when p.stockquantity = 0 then 'Not Available'  
Else 'Available'
```

End as Avail\_Status

From customers c

inner JOIN orders o ON c.customerid = o.customerid

left join orderdetails od on o.orderid = od.order\_id

left join products p on od.product\_id = p.productid;

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Cc

	name	name	Avail_Status
▶	Kavya Iyer	Formal Trousers Slim	Available
	Manish Patel	Wireless Earbuds	Available
	Ravi Shetty	Sunscreen SPF 50	Not Available
	Tanvi Shah	Winter Sweater Wool	Not Available
	Tanvi Shah	Tennis Racket	Available
	Gaurav Sinha	Sports Shorts	Available
	Gaurav Sinha	Herbal Shampoo 500ml	Available
	Gaurav Sinha	Face Wash Neem	Available
	Gaurav Sinha	Mystery Novel Vol. 1	Available

Result 35 ×

34. Show customers placed orders for Out-of-Stock products.

Query-

Select c.name, p.name

From customers c

inner JOIN orders o ON c.customerid = o.customerid

left join orderdetails od on o.orderid = od.order\_id

left join products p on od.product\_id = p.productid

where p.stockquantity = 0;

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	name	name
▶	Meera Kulkarni	Winter Jacket
	Suresh Agarwal	Denim Jacket
	Tanvi Shah	Winter Sweater Wool
	Ravi Shetty	Sunscreen SPF 50

35. Show number of orders placed from specific mail domain, like gmail, rediff.

Query-

```
SELECT
    SUBSTRING_INDEX(c.email, '@', -1) AS email_domain,
    COUNT(DISTINCT c.customerid) AS total_customers
FROM customers c
INNER JOIN orders o ON c.customerid = o.customerid
WHERE
    c.email LIKE '%@gmail%'
    OR c.email LIKE '%@yahoo%'
    OR c.email LIKE '%@rediff%'
GROUP BY email_domain;
```

Output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	email_domain	total_customers
▶	gmail.com	49
	rediffmail.com	8
	yahoo.com	10

36. Show customers whose orders delivered late e.g. after 4 days.

Query-

```
SELECT c.name, datediff(s.deliverydate, s.shipdate) as Delivered_In_Days
FROM customers c
INNER JOIN orders o ON c.customerid = o.customerid
LEFT JOIN orderdetails od ON o.orderid = od.order_id
LEFT JOIN shipping s ON od.order_id = s.orderid
where datediff(s.deliverydate, s.shipdate) > 4;
```

Output-

Result Grid | Filter Rows: [ ] | Export: [ ] | Wrap Cell Co

	name	Delivered_In_Days
▶	Meera Kulkarni	5
	Meera Kulkarni	5
	Simran Gill	5
	Chirag Shah	5
	Anurag Chopra	5
	Anurag Chopra	5
	Megha Banerjee	5
	Harsh Vardhan	5
	Sonia Fernandes	5

37. Show customers whose orders delivered late e.g. after 4 days.

Query-

```
SELECT c.name
FROM customers c
INNER JOIN orders o ON c.customerid = o.customerid
LEFT JOIN orderdetails od ON o.orderid = od.order_id
LEFT JOIN products p ON od.product_id = p.productid
LEFT JOIN discounts d ON p.productid = d.productid
WHERE d.discountamount IS NULL;
```

Output-

Result Grid | Filter Rows: [ ] | Export: [ ] | Wrap Cell Co

	name
▶	Amit Malhotra
	Nikhil Saxena
	Ramesh Bansal
	Kirti Malviya
	Ruchi Joshi
	Devendra Singh
	Pallavi Rane

38. Show total number of orders placed by unique customers.

Query-

```
SELECT c.name, count(o.orderID) as Orders
FROM customers c
INNER JOIN orders o ON c.customerid = o.customerid
group by C.name
order by Orders desc;
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	name	Orders
▶	Arjun Ghosh	5
	Meera Kulkarni	4
	Gaurav Sinha	4
	Chirag Shah	4
	Isha Gupta	1
	Rohan Mehta	1
	Priya Nair	1
	Ananya Singh	1
	Siddharth Menon	1

39. Show category-wise sale.

Query-

```
Select p.name, sum(p.price*od.quantity) as Total_Sales
From customers c
inner JOIN orders o ON c.customerid = o.customerid
left join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid
group by p.name;
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	categoryname	Total_Sales
▶	Home & Kitchen	117660.00
	Beauty	9516.00
	Electronics	373866.00
	Books	13866.00
	Sports	72755.00
	Clothing	70060.00
	NULL	NULL

40. Show product-wise ordered qty.

Query-

```
Select p.name, sum(od.quantity) as Total_Qty
From orders o
left join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid
group by p.name;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	name	Total_Qty
▶	Yoga Mat Comfort	9
	Romantic Novel Vol. 2	6
	Historical Novel	6
	Dinner Set 24pcs	5
	Formal Trousers Slim	5
	Sunscreen SPF 50	5
	Ethnic Saree	5
	Hiking Backpack	5
	Non-stick Cooking Pan	5

41. Show order count placed per product.

Query-

```
Select p.name, count(o.orderid) as Total_Qty
From orders o
inner join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid
group by p.name
order by Total_Qty desc;
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	name	Total_Qty
	Romantic Novel Vol. 2	2
	Yoga Mat Comfort	2
	Microwave Oven 20L	2
	Historical Novel	2
	Raincoat Waterproof	1
	Wireless Headphones Z	1
	Track Pants	1
	Self-help Bestseller	1
	Slim Fit Jeans	1

42. Show yearly sales.

Query-

```
Select year(o.orderdate) as year, sum(p.price*od.quantity) as Total_Sales
From customers c
inner JOIN orders o ON c.customerid = o.customerid
left join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid
group by year;
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	year	Total_Sales
▶	2021	124168.00
	2022	181962.00
	2023	74567.00
	2024	77159.00
	2025	199867.00

43. Show products not sold in specific year.

Query-

```
select distinct p.name
from products p
inner join orderdetails od on od.product_id = p.productid
inner join orders o on od.order_id = o.orderid
where year(o.orderdate) not in (2024,2025);
```

output-

Result Grid | Filter Rows: | Export: | Wrap Cell Co

	name
	Encyclopedia Kids
	External Hard Drive 1TB
	Face Wash Neem
	Formal Trousers Slim
	Hair Oil 200ml
	Hair Serum
	Hand Cream 100ml
	Herbal Shampoo 500ml
	Induction Cooktop

44. Show products sold in specific year.

Query-

```
select distinct p.name
from products p
inner join orderdetails od on od.product_id = p.productid
inner join orders o on od.order_id = o.orderid
where year(o.orderdate) in (2022);
```

output-

	name
▶	Track Pants
	Badminton Racket
	Romantic Novel Vol. 2
	Induction Cooktop
	Herbal Shampoo 500ml
	Face Wash Neem
	Portable Power Bank
	Hand Cream 100ml
	Laptop Pro 14
	Casual Shirt Cotton

45. Show customers not placed orders in specific year.

Query-

```
select c.name
from customers c
inner join orders o on o.customerid = c.customerid
where year(o.orderdate) not in (2025);
```

output-

	name
	Sneha Pillai
	Meera Kulkarni
	Payal Jadhav
	Krishnan Na
	Payal Jadhav
	Arjun Ghosh
	Simran Gill
	Gaurav Sinha
	Chirag Shah
	Sahil Oberoi
	Mona Chatter...

46. Show customers placed orders in specific year.

Query-

```
select c.name
from customers c
inner join orders o on o.customerid = c.customerid
where year(o.orderdate) in (2025);
```

output-

Result Grid				Filter Rows:	<input type="text"/>	Export:		Wrap Cell Co
	name							
▶	Meera Kulkarni							
	Nisha Ahuja							
	Ashok Rawat							
	Sakshi Raina							
	Raghavendra Reddy							
	Priya Nair							
	Amit Malhotra							
	Anjali Mathur							
	Mohit Kumar							
	Juhi Pathak							

47. Show customers with order details.

Query-

```
select c.customerid, c.name, o.orderid, o.orderdate
from customers c
inner join orders o on o.customerid = c.customerid;
```

output-

Result Grid				Filter Rows:	<input type="text"/>	Export:		Wrap Cell Co
	customerid	name	orderid	orderdate				
▶	25	Gaurav Sinha	1	2021-02-15				
	12	Meera Kulkarni	2	2021-03-10				
	23	Ravi Shetty	3	2021-05-21				
	34	Sneha Pillai	4	2021-07-04				
	12	Meera Kulkarni	5	2021-08-12				
	56	Payal Jadhav	6	2021-09-30				
	67	Krishnan Nair	7	2021-11-18				
	39	Arjun Ghosh	8	2021-12-05				
	14	Simran Gill	9	2022-01-12				
	25	Gaurav Sinha	10	2022-02-28				
Result Grid				Filter Rows:	<input type="text"/>	Export:		Wrap Cell Co
	customerid	name	orderid	orderdate				
	72	Poonam Dey	71	2025-08-07				
	3	Rohan Mehta	72	2025-08-11				
	17	Deepak Khanna	73	2025-08-13				
	27	Nikhil Saxena	74	2025-08-14				
	38	Lavanya Dutta	75	2025-08-15				
	49	Ramesh Bansal	76	2025-08-15				
	60	Ruchi Joshi	77	2025-08-16				
	71	Devendra Singh	78	2025-08-16				
	80	Pallavi Rane	79	2025-08-16				
	6	Ananya Singh	80	2025-08-16				

48. Show order details with product name and amount.

Query-

```
Select c.customerid, c.name, o.orderid, p.name, sum(p.price*od.quantity) as Total_Sales
```

From customers c

```
inner JOIN orders o ON c.customerid = o.customerid  
left join orderdetails od on o.orderid = od.order_id  
left join products p on od.product_id = p.productid  
group by c.name, o.orderid, p.name;
```

output-

	customerid	name	orderid	name	Total_Sales
▶	1	Aarav Sharma	57	Pressure Cooker 5L	6396.00
	2	Isha Gupta	65	Conditioner 500ml	747.00
	3	Rohan Mehta	72	Dinner Set 24pcs	14995.00
	4	Priya Nair	50	Smartwatch Lite	3999.00
	6	Ananya Singh	80	Self-help Bestseller	998.00
	9	Siddharth Menon	29	Digital Camera 24MP	39998.00
	10	Ritika Jain	36	Volleyball	2397.00
	12	Meera Kulkarni	2	External Hard Drive 1TB	11997.00
	12	Meera Kulkarni	2	Romantic Novel Vol. 2	698.00
	12	Meera Kulkarni	5	Yoga Mat Comfort	3196.00

	customerid	name	orderid	name	Total_Sales
	72	Poonam Dey	71	Bluetooth Speaker Mini	1499.00
	73	Raghavendra R...	49	Formal Blazer	14995.00
	74	Sonia Fernandes	28	Children's Story Book	1495.00
	75	Chirag Shah	11	Laptop Pro 14	137997.00
	75	Chirag Shah	22	Skipping Rope	598.00
	75	Chirag Shah	33	Slim Fit Jeans	4497.00
	75	Chirag Shah	44	Cotton Kurta	2998.00
	76	Aditi Sood	42	Historical Novel	1347.00
	77	Mahesh Pillai	21	Basketball Pro	1699.00
	80	Pallavi Rane	79	NULL	NULL

49. Show order count placed per product.

Query-

```
select c.name, p.name, cat.categoryname  
from customers c  
inner join orders o on o.customerid = c.customerid  
left join orderdetails od on o.orderid = od.order_id  
left join products p on od.product_id = p.productid  
left join categories cat on cat.categoryid = p.categoryid  
where  
cat.categoryname in (select c1.categoryname from categories c1  
inner join products p1 on p1.categoryid = c1.categoryid where p1.name like '%book%');
```

output-

Result Grid | Filter Rows:  Export: Wrap Cell Co

	name	name	categoryname
▶	Gaurav Sinha	Mystery Novel Vol. 1	Books
	Omprakash Tripathi	Romantic Novel Vol. 2	Books
	Meera Kulkarni	Romantic Novel Vol. 2	Books
	Ananya Singh	Self-help Bestseller	Books
	Aditi Sood	Historical Novel	Books
	Abhay Mishra	Historical Novel	Books
	Sanya Bhatia	Science Fiction Novel	Books
	Sonia Fernandes	Children's Story Book	Books
	Shalini Mukherjee	Biography Book	Books
	Ajay Thakur	Poetry Collection	Books

50. Show customers who ordered products including specific product name.

Query-

```
select c.name, p.name, od.quantity
from customers c
inner join orders o on o.customerid = c.customerid
left join orderdetails od on o.orderid = od.order_id
left join products p on od.product_id = p.productid
where p.name in (select p1.name from products p1 where p1.name like '%jacket%');
```

output-

Result Grid | Filter Rows:  Export: Wrap Cell Co

	name	name	quantity
▶	Meera Kulkarni	Winter Jacket	1
	Suresh Agarwal	Denim Jacket	2