
1. CREATE DATABASE

```
CREATE DATABASE retail_db;  
USE retail_db;
```

2. CREATE TABLES

A. Customers Table

(Some customers have no orders → used for RIGHT JOIN and OUTER JOIN.)

```
CREATE TABLE customers (  
    customer_id INT PRIMARY KEY,  
    customer_name VARCHAR(50),  
    email VARCHAR(80),  
    city VARCHAR(50)  
) ;
```

B. Orders Table

(Some orders have NULL product_id or NULL customer_id to demonstrate LEFT JOIN.)

```
CREATE TABLE orders (  
    order_id INT PRIMARY KEY,  
    order_date DATE,  
    customer_id INT NULL,  
    total_amount DECIMAL(10,2),  
    product_id INT NULL  
) ;
```

C. Products Table

(Some products have never been ordered → used for RIGHT JOIN.)

```
CREATE TABLE products (
    product_id INT PRIMARY KEY,
    product_name VARCHAR(100),
    category VARCHAR(50),
    price DECIMAL(10, 2)
);
```

3. INSERT DATA

Customers

```
INSERT INTO customers VALUES
(1, 'Aisha Khan', 'aisha@xyz.com', 'Mumbai'),
(2, 'Rahul Sharma', 'rahul@xyz.com', 'Delhi'),
(3, 'John Daniel', 'john@xyz.com', 'Bangalore'),
(4, 'Meera Iyer', 'meera@xyz.com', 'Chennai'),
(5, 'Sanjay Patel', 'sanjay@xyz.com', 'Hyderabad');
```

Customer 5 will have no orders.

Products

```
INSERT INTO products VALUES
(101, 'Laptop HP 15', 'Electronics', 52000),
(102, 'Samsung Phone A54', 'Electronics', 28000),
(103, 'Jeans Blue Fit', 'Fashion', 1500),
(104, 'T-Shirt Classic', 'Fashion', 700),
(105, 'Wireless Mouse', 'Accessories', 900),
(106, 'Rice 5KG Bag', 'Groceries', 320),
(107, 'Olive Oil 1L', 'Groceries', 540),
(108, 'Printer Canon G2012', 'Electronics', 12500);
```

Products 107 and 108 have zero orders.

Orders

```
INSERT INTO orders VALUES
(1001, '2024-01-05', 1, 52000, 101),
(1002, '2024-01-06', 2, 28000, 102),
(1003, '2024-01-07', 3, 1500, 103),
(1004, '2024-01-07', 1, 700, 104),
(1005, '2024-01-08', 2, 900, 105),
(1006, '2024-01-08', NULL, 320, 106),      -- customer unknown
(1007, '2024-01-09', 1, 540, NULL),        -- product unknown
(1008, '2024-01-10', 3, 12500, 108),
(1009, '2024-01-10', 4, 320, 106),
(1010, '2024-01-11', NULL, 700, 104),      -- customer null
(1011, '2024-01-12', 2, 540, 107);        -- product exists but never ordered
```

SECTION A: INNER JOIN (10)

1. List all orders with customer names and email.
 2. Show product name, category, price for every ordered product.
 3. List all orders with customer name and product name.
 4. Show customer name and total_amount for all valid customer orders.
 5. List all Electronics products that have been ordered.
 6. Find customers who ordered Fashion products.
 7. Show all orders above 1000 with customer and product details.
 8. Show customers from Mumbai who placed at least one order.
 9. Show number of orders per customer (using INNER JOIN + GROUP BY).
 10. List all customers and the total amount they have spent.
-

SECTION B: LEFT JOIN (Orders → Customers) (10)

11. Show all orders, even those without customer info.
12. Find all orders where customer_id is NULL.
13. Display orders with customer city (NULL when customer is missing).
14. Show all customer names and the number of orders they placed (include zero).

15. Show customers who have not placed ANY order.
 16. Show all orders and label missing customers as "Guest Checkout".
 17. Show orders that have missing product details.
 18. Show orders placed by customers from Delhi or missing customer info.
 19. Count total orders including ones without customer linkage.
 20. Show the percentage of orders with missing customers.
-

SECTION C: RIGHT JOIN (Products → Orders)

(10)

21. Show all products and the orders associated with them (NULL for unused products).
 22. List products that have never been ordered.
 23. Show Electronics products and the number of times they were ordered.
 24. Show Groceries products and their first order date.
 25. List all products with their total sales, including those with zero sales.
 26. For each category, show number of ordered products (include zero).
 27. Show products that were ordered by customers from Bangalore.
 28. Show products missing from order table (never sold).
 29. Show the count of orders grouped by product name (including zero).
 30. Show products that appeared in at least one NULL customer order.
-