Subqueries, JOINS, EXISTS, and NOT EXISTS.

\*\*Subqueries:\*\*

1. Retrieve the names of customers who have placed orders for products in the "Mobile" category.

SELECT DISTINCT CustomerName

FROM Customer

WHERE CustomerID IN (

SELECT CustomerID

FROM Orders

WHERE ProductID IN (

SELECT ProductID

FROM Product

WHERE ProductCategory = 'Mobile'

)

);

2. Find the product names that have been ordered by customers belonging to group "A."

SELECT DISTINCT ProductName

FROM Product

WHERE ProductID IN (

SELECT ProductID

FROM Orders

WHERE CustomerID IN (

SELECT CustomerID

FROM Customer

WHERE Group = 'A'

)

);

3. Get the total number of orders placed by customers with email addresses.

SELECT COUNT(\*)

FROM Orders

WHERE CustomerID IN (

SELECT CustomerID

FROM Customer

WHERE Email IS NOT NULL

);

4. List customers who have placed more than two orders.

SELECT CustomerName

FROM Customer

WHERE CustomerID IN (

SELECT CustomerID

FROM Orders

GROUP BY CustomerID

HAVING COUNT(\*) > 2

);

5. Find the product categories that have the most orders.

SELECT ProductCategory

FROM Product

WHERE ProductID IN (

SELECT ProductID

FROM Orders

GROUP BY ProductID

HAVING COUNT(\*) = (

SELECT MAX(OrderCount)

FROM (

SELECT ProductID, COUNT(\*) AS OrderCount

FROM Orders

GROUP BY ProductID

) AS Subquery

)

);

6. Retrieve the customer names who have placed orders with "Express" shipping.

SELECT DISTINCT CustomerName

FROM Customer

WHERE CustomerID IN (

SELECT CustomerID

FROM Orders

WHERE ShippingSpeed = 'Express'

);

7. Get the order IDs for orders with the highest shipping speed.

SELECT OrderID

FROM Orders

WHERE ShippingSpeed = (

SELECT MAX(ShippingSpeed)

FROM Orders

);

8. List the customers who have never placed an order.

SELECT CustomerName

FROM Customer

WHERE CustomerID NOT IN (

SELECT DISTINCT CustomerID

FROM Orders

);

9. Find the product IDs for products with no orders.

SELECT ProductID

FROM Product

WHERE ProductID NOT IN (

SELECT DISTINCT ProductID

FROM Orders

);

10. Retrieve the customer names who have placed orders for products from the "Laptop" category.

SELECT DISTINCT CustomerName

FROM Customer

WHERE CustomerID IN (

SELECT CustomerID

FROM Orders

WHERE ProductID IN (

SELECT ProductID

FROM Product

WHERE ProductCategory = 'Laptop'

)

);

\*\*JOINS:\*\*

1. Retrieve a list of all orders with the customer names who placed them.

SELECT Orders.OrderID, Customer.CustomerName

FROM Orders

INNER JOIN Customer ON Orders.CustomerID = Customer.CustomerID;

2. Get a list of products and their corresponding categories.

SELECT Product.ProductName, Product.ProductCategory

FROM Product;

3. List all orders with product details (name and brand) and customer names.

SELECT Orders.OrderID, Product.ProductName, Product.ProductBrand, Customer.CustomerName

FROM Orders

INNER JOIN Product ON Orders.ProductID = Product.ProductID

INNER JOIN Customer ON Orders.CustomerID = Customer.CustomerID;

4. Retrieve a list of customers and their associated orders, including shipping speed.

SELECT Customer.CustomerName, Orders.OrderID, Orders.ShippingSpeed

FROM Customer

LEFT JOIN Orders ON Customer.CustomerID = Orders.CustomerID;

5. Find the total number of orders placed by each customer along with their names.

SELECT Customer.CustomerName, COUNT(Orders.OrderID) AS TotalOrders

FROM Customer

LEFT JOIN Orders ON Customer.CustomerID = Orders.CustomerID

GROUP BY Customer.CustomerName;

6. List the products ordered by customers in group "B."

SELECT Product.ProductName

FROM Product

INNER JOIN Orders ON Product.ProductID = Orders.ProductID

INNER JOIN Customer ON Orders.CustomerID = Customer.CustomerID

WHERE Customer.Group = 'B';

7. Get the order IDs, customer names, and product names for each order.

SELECT Orders.OrderID, Customer.CustomerName, Product.ProductName

FROM Orders

INNER JOIN Customer ON Orders.CustomerID = Customer.CustomerID

INNER JOIN Product ON Orders.ProductID = Product.ProductID;

8. Find the customers who have ordered products from the "TV" category.

SELECT DISTINCT Customer.CustomerName

FROM Customer

INNER JOIN Orders ON Customer.CustomerID = Orders.CustomerID

INNER JOIN Product ON Orders.ProductID = Product.ProductID

WHERE Product.ProductCategory = 'TV';

9. Retrieve a list of products that have never been ordered.

SELECT Product.ProductName

FROM Product

LEFT JOIN Orders ON Product.ProductID = Orders.ProductID

WHERE Orders.OrderID IS NULL;

10. List all orders with the product name and customer group.

SELECT Orders.OrderID, Product.ProductName, Customer.Group

FROM Orders

INNER JOIN Product ON Orders.ProductID = Product.ProductID

INNER JOIN Customer ON Orders.CustomerID = Customer.CustomerID;

\*\*EXISTS:\*\*

1. Find customers who have placed at least one order.

SELECT Customer.CustomerName

FROM Customer

WHERE EXISTS (

SELECT 1

FROM Orders

WHERE Orders.CustomerID = Customer.CustomerID

);

2. List orders with "Express" shipping that exist in the Orders table.

SELECT \*

FROM Orders

WHERE ShippingSpeed = 'Express'

AND EXISTS (

SELECT 1

FROM Orders AS Subquery

WHERE Orders.OrderID = Subquery.OrderID

);

3. Retrieve product categories that have at least one order.

SELECT DISTINCT ProductCategory

FROM Product

WHERE EXISTS (

SELECT 1

FROM Orders

WHERE Product.ProductID = Orders.ProductID

);

4. Get a list of customers who do not have an associated email address.

SELECT Customer.CustomerName

FROM Customer

WHERE NOT EXISTS (

SELECT 1

FROM Customer AS Subquery

WHERE Customer.CustomerID = Subquery.CustomerID

AND Subquery.Email IS NOT NULL

);

5. Find orders placed by customers who do not belong to groups "A" or "B."

SELECT \*

FROM Orders

WHERE NOT EXISTS (

SELECT 1

FROM Customer AS Subquery

WHERE Orders.CustomerID = Subquery.CustomerID

AND Subquery.Group IN ('A', 'B')

);

6. List all products that have been ordered at least once.

SELECT \*

FROM Product

WHERE EXISTS (

SELECT 1

FROM Orders

WHERE Product.ProductID = Orders.ProductID

);

7. Get the order IDs for orders placed by customers without email addresses.

SELECT OrderID

FROM Orders

WHERE NOT EXISTS (

SELECT 1

FROM Customer AS Subquery

WHERE Orders.CustomerID = Subquery.CustomerID

AND Subquery.Email IS NOT NULL

);

8. Find customers who have never placed orders and belong to group "D."

SELECT Customer.CustomerName

FROM Customer

WHERE Group = 'D'

AND NOT EXISTS (

SELECT 1

FROM Orders

WHERE Customer.CustomerID = Orders.CustomerID

);

9. Retrieve orders with shipping speed other than "Regular" if they exist.

SELECT \*

FROM Orders

WHERE ShippingSpeed != 'Regular'

AND EXISTS (

SELECT 1

FROM Orders AS Subquery

WHERE Orders.OrderID = Subquery.OrderID

);

10. List customers who have never placed orders for products from the "TV" category.

SELECT Customer.CustomerName

FROM Customer

WHERE NOT EXISTS (

SELECT 1

FROM Orders

INNER JOIN Product ON Orders.ProductID = Product.ProductID

WHERE Customer.CustomerID = Orders.CustomerID

AND Product.ProductCategory = 'TV'

);

\*\*NOT EXISTS:\*\*

1. Find customers who have never placed an order.

SELECT CustomerName

FROM Customer

WHERE CustomerID NOT IN (

SELECT DISTINCT CustomerID

FROM Orders

);

2. Retrieve orders with "Air" shipping that do not exist in the Orders table.

SELECT \*

FROM SomeOtherTable

WHERE ShippingType = 'Air'

AND NOT EXISTS (

SELECT 1

FROM Orders

WHERE SomeOtherTable.OrderID = Orders.OrderID

);

3. List product categories that have never received orders.

SELECT ProductCategory

FROM Product

WHERE ProductID NOT IN (

SELECT DISTINCT ProductID

FROM Orders

);

4. Get a list of customers who do not have an associated email address.

SELECT CustomerName

FROM Customer

WHERE Email IS NULL;

5. Find orders placed by customers who do not belong to groups "A" or "B."

SELECT \*

FROM Orders

WHERE CustomerID NOT IN (

SELECT CustomerID

FROM Customer

WHERE Group IN ('A', 'B')

);

6. List all products that have never been ordered.

SELECT ProductName

FROM Product

WHERE ProductID NOT IN (

SELECT DISTINCT ProductID

FROM Orders

);

7. Get the order IDs for orders placed by customers without email addresses.

SELECT OrderID

FROM Orders

WHERE CustomerID NOT IN (

SELECT CustomerID

FROM Customer

WHERE Email IS NOT NULL

);

8. Find customers who have never placed orders and belong to group "D."

SELECT CustomerName

FROM Customer

WHERE Group = 'D'

AND CustomerID NOT IN (

SELECT DISTINCT CustomerID

FROM Orders

);

9. Retrieve orders with shipping speed other than "Express" if they do not exist.

SELECT \*

FROM SomeOtherTable

WHERE ShippingType != 'Express'

AND NOT EXISTS (

SELECT 1

FROM Orders

WHERE SomeOtherTable.OrderID = Orders.OrderID

);

10. List customers who have never placed orders for products from the "TV" category.

SELECT CustomerName

FROM Customer

WHERE CustomerID NOT IN (

SELECT DISTINCT CustomerID

FROM Orders

WHERE ProductID IN (

SELECT ProductID

FROM Product

WHERE ProductCategory = 'TV'

)

);