

INNER JOINS

❖ INNER JOIN: Orders with Customer and Product Names

1.

The screenshot shows the Snowflake SQL Editor interface. On the left, the 'Databases' tab is active, showing a tree view of the database structure. The 'PRACTICAL2' database is selected, and the 'PUBLIC' schema is expanded, showing tables: CUSTOMERS_LARGE, ORDERS_LARGE, and PRODUCTS_LARGE. The main editor area contains the following SQL query:

```
1 SELECT OrderID,  
2      OrderDate,  
3      CustomerName,  
4      ProductName,  
5      Quantity  
6 FROM ORDERS_LARGE AS O  
7 INNER JOIN CUSTOMERS_LARGE AS C  
8 ON O.CustomerID= C.CustomerID  
9 INNER JOIN PRODUCTS_LARGE AS P  
10 ON P.ProductID= P.ProductID;
```

Below the query editor, the 'Results' tab is active, displaying a table with 4 rows and 6 columns. The columns are: #, ORDERID, ORDERDATE, CUSTOMERNAME, PRODUCTNAME, and # QUANTITY. The rows represent individual orders with their details.

#	ORDERID	ORDERDATE	CUSTOMERNAME	PRODUCTNAME	# QUANTITY
1	127	2024-06-30	Customer_1001	Product_2001	5
2	462	2024-06-14	Customer_1001	Product_2001	4
3	1106	2024-05-15	Customer_1001	Product_2001	10
4	1855	2024-05-04	Customer_1001	Product_2001	7

On the right side of the results, the 'Query Details' panel is visible, showing the query duration as 1.6s, the number of rows as 800K, and the query ID as 01bbdb9c-0000-f5e2-0...

❖ INNER JOIN: Customers Who Placed Orders

2.

2025-04-18 5:24pm Practical 2

ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PUBLIC

- Tables
 - CUSTOMERS_LARGE
 - ORDERS_LARGE
 - PRODUCTS_LARGE
- RETAIL_SALES_DATABASE
- SNOWFLAKE

ORDERS_LARGE 4K Rows

- ORDERID NUMBER(38,0)
- CUSTOMERID NUMBER(38,0)
- PRODUCTID NUMBER(38,0)
- QUANTITY NUMBER(38,0)
- ORDERDATE DATE

```

12 SELECT c.CustomerID,
13        c.CustomerName,
14        c.Country,
15        o.OrderID,
16        o.OrderDate
17 FROM CUSTOMERS_LARGE AS C
18 INNER JOIN ORDERS_LARGE AS O
19 ON C.CustomerID= O.CustomerID
20
21
22
23
24
25
26
27

```

Results Chart

	# CUSTOMERID	CUSTOMERNAME	COUNTRY	# ORDERID	ORDERDATE
1	1251	Customer_1251	Germany	1	2023-06-10
2	1236	Customer_1236	Australia	2	2023-12-07
3	1170	Customer_1170	Germany	3	2024-10-26
4	1344	Customer_1344	Canada	4	2023-02-17

Query Details

- Query duration 39ms
- Rows 4K
- Query ID 01bbdbec-0000-f5f6-0...

❖ LEFT JOIN: Product Order Count

3.

2025-04-18 5:24pm Practical 2

ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PUBLIC

- Tables
 - CUSTOMERS_LARGE
 - ORDERS_LARGE
 - PRODUCTS_LARGE
- RETAIL_SALES_DATABASE
- SNOWFLAKE

ORDERS_LARGE 4K Rows

- ORDERID NUMBER(38,0)
- CUSTOMERID NUMBER(38,0)
- PRODUCTID NUMBER(38,0)
- QUANTITY NUMBER(38,0)
- ORDERDATE DATE

```

26
27
28
29 SELECT c.CustomerID,
30        c.CustomerName,
31        c.Country,
32        o.OrderID,
33        o.OrderDate,
34        o.ProductID,
35        o.Quantity
36 FROM CUSTOMERS_LARGE AS C
37 LEFT JOIN ORDERS_LARGE AS O
38 ON C.CUSTOMERID= O.CUSTOMERID;
39
40
41

```

Results Chart

	# CUSTOMERID	CUSTOMERNAME	COUNTRY	# ORDERID	ORDERDATE	# PRODUCTID	# QUANT
1	1251	Customer_1251	Germany	1	2023-06-10	2014	
2	1236	Customer_1236	Australia	2	2023-12-07	2004	
3	1170	Customer_1170	Germany	3	2024-10-26	2171	
4	1344	Customer_1344	Canada	4	2023-02-17	2007	

Query Details

- Query duration 325ms
- Rows 4K
- Query ID 01bbdbef-0000-f5dd-0...

❖ LEFT JOIN: Product Order Count

❖

4.

Query:

```

41
42
43
44 SELECT P.PRODUCTNAME,
45        COUNT (O.PRODUCTID) AS TOTALORDERS
46 FROM PRODUCTS_LARGE AS P
47 LEFT JOIN ORDERS_LARGE AS O
48 ON P.PRODUCTID= O.PRODUCTID
49 GROUP BY P.ProductName;
50
51
52
53
54
55
56
57

```

Results:

PRODUCTNAME	TOTALORDERS
Product_2014	22
Product_2004	24
Product_2171	15
Product_2007	12

Query Details:
 Query duration: 714ms
 Rows: 200
 Query ID: 01bbdc0a-0000-f5f3-0...

❖ RIGHT JOIN: Orders with Product Info (Include Products Not Ordered)

5.

Query:

```

51
52
53
54
55
56 SELECT o.OrderID,
57        o.OrderDate,
58        p.ProductID,
59        p.ProductName,
60        p.Price,
61        o.Quantity
62 FROM ORDERS_LARGE AS O
63 RIGHT JOIN PRODUCTS_LARGE AS P
64 ON O.PRODUCTID= P.PRODUCTID
65
66
67

```

Results:

ORDERID	ORDERDATE	PRODUCTID	PRODUCTNAME	PRICE	QUANTITY
1	2023-06-10	2014	Product_2014	522	10
2	2023-12-07	2004	Product_2004	1996	5
3	2024-10-26	2171	Product_2171	76	9
4	2023-02-17	2007	Product_2007	156	2

Query Details:
 Query duration: 582ms
 Rows: 4K
 Query ID: 01bbdc1b-0000-f5ff-00...

❖ RIGHT JOIN: Customer Info with Orders (Include All Customers)

6.

The screenshot shows a SQL IDE interface with a query editor and a results pane. The query is as follows:

```

67
68 SELECT c.CustomerID,
69        c.CustomerName,
70        c.Country,
71        o.OrderID,
72        o.OrderDate,
73        o.ProductID,
74        o.Quantity
75 FROM CUSTOMERS_LARGE AS C
76 RIGHT JOIN ORDERS_LARGE AS O
77 ON C.CUSTOMERID= O.CUSTOMERID
78
79
80
81
82
83

```

The results pane displays the following data:

	# CUSTOMERID	CUSTOMERNAME	COUNTRY	# ORDERID	ORDERDATE	# PRODUCTID	# QUANTITY
1	1251	Customer_1251	Germany	1	2023-06-10	2014	
2	1236	Customer_1236	Australia	2	2023-12-07	2004	
3	1170	Customer_1170	Germany	3	2024-10-26	2171	
4	1244	Customer_1244	Canada	4	2023-02-17	2007	

Query Details: Query duration 846ms, Rows 4K, Query ID 01bdc57-0000-f5e5-0...

7.

```

SELECT `OrderID`,
       `OrderDate`,
       `CustomerName`,
       `ProductName`,
       `Quantity`
FROM `Salestrends.Orders` AS O
INNER JOIN `Salestrends.Customers` AS C
ON O.CustomerID= C.CustomerID
INNER JOIN `Salestrends.Products` AS P
ON P.ProductID= P.ProductID;

```

```

SELECT C.`CustomerID`,
       `CustomerName`,
       `Country`,
       `OrderID`,
       `OrderDate`
FROM `Salestrends.Customers` AS C
INNER JOIN `Salestrends.Orders` AS O
ON C.CustomerID= O.CustomerID

```

```

SELECT C.`CustomerID`,
       `CustomerName`,
       `Country`,
       `OrderID`,
       `OrderDate`,
       `ProductID`,
       `Quantity`
FROM `Salestrends.Customers` AS C
LEFT JOIN `Salestrends.Orders` AS O
ON C.CustomerID= O.CustomerID

```

```

SELECT O.`ProductID`,
       `ProductName`,

```

```
        COUNT (`ProductID`) AS TOTALORDERS
FROM `Salestrends.Products` AS P
LEFT JOIN `Salestrends.Orders` AS O
ON P.ProductID= O.ProductID
GROUP BY 1,2
```