

PRACTICAL EXERCISE 2

Using snowflake

Q1 List all orders along with the customer name and product name.

The screenshot shows a cloud-based database interface with a dark theme. At the top, there are three time points: "Load sample data with Pyt..." (2025-04-23 12:20am), "2025-04-24 10:20pm", and "2025-04-24 10:25pm". The current session user is "ACCOUNTADMIN" with a "COMPUTE_WH (X-Small)" quota. A "Share" button and a "Code Versions" link are also visible. The code editor contains the following SQL query:

```
10 SELECT OrderId, orderdate, customername, productname, quantity
11 FROM PRACTICAL2.PUBLIC."ORDER" AS O
12 INNER JOIN PRACTICAL2.PUBLIC.CUSTOMER AS C
13 ON O.CUSTOMERID = C.CUSTOMERID
14 INNER JOIN PRACTICAL2.PUBLIC.PRODUCTS AS P
15 ON O.PRODUCTID = P.PRODUCTID;
```

The results table has columns: ORDERID, ORDERDATE, CUSTOMERNAME, PRODUCTNAME, and QUANTITY. The data is as follows:

ORDERID	ORDERDATE	CUSTOMERNAME	PRODUCTNAME	QUANTITY
1	2023-06-10	Customer_1251	Product_2014	10
2	2023-12-07	Customer_1236	Product_2004	5
3	2024-10-26	Customer_1170	Product_2171	9
4	2023-02-17	Customer_1344	Product_2007	2
5	2024-11-06	Customer_1319	Product_2061	2
6	2024-11-23	Customer_1185	Product_2190	3
7	2023-07-29	Customer_1011	Product_2099	8

On the right side, there is a "Query Details" panel showing a query duration of 100ms, 8K rows, and a query ID of 01bbf093-0000-f5e5-0... . A search bar and a dropdown menu are also present.

Q2 Which customers have placed at least one order?

The screenshot shows a cloud-based database interface with a dark theme, similar to the first one. At the top, there are three time points: "Load sample data with Pyt..." (2025-04-23 12:20am), "2025-04-24 10:20pm", and "2025-04-24 10:25pm". The current session user is "ACCOUNTADMIN" with a "COMPUTE_WH (X-Small)" quota. A "Share" button and a "Code Versions" link are also visible. The code editor contains the following SQL query:

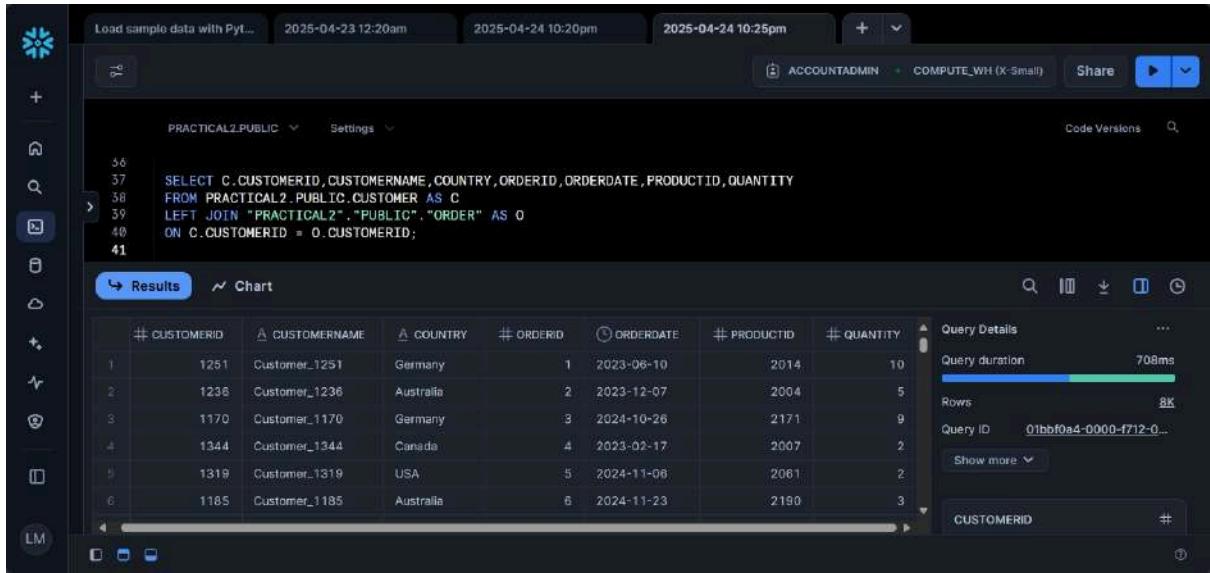
```
29
30
31 SELECT C.CUSTOMERID, CUSTOMERNAME, COUNTRY, ORDERID, ORDERDATE
32 FROM "PRACTICAL2"."PUBLIC"."CUSTOMER" AS C
33 INNER JOIN "PRACTICAL2"."PUBLIC"."ORDER" AS O
34 ON C.CUSTOMERID = O.CUSTOMERID;
```

The results table has columns: CUSTOMERID, CUSTOMERNAME, COUNTRY, ORDERID, and ORDERDATE. The data is as follows:

CUSTOMERID	CUSTOMERNAME	COUNTRY	ORDERID	ORDERDATE
1251	Customer_1251	Germany	1	2023-06-10
1236	Customer_1236	Australia	2	2023-12-07
1170	Customer_1170	Germany	3	2024-10-26
1344	Customer_1344	Canada	4	2023-02-17
1319	Customer_1319	USA	5	2024-11-06
1185	Customer_1185	Australia	6	2024-11-23
1011	Customer_1011	Germany	7	2023-07-29

On the right side, there is a "Query Details" panel showing a query duration of 908ms, 8K rows, and a query ID of 01bbf09a-0000-f5e5-0... . A search bar and a dropdown menu are also present.

Q3 List all customers and any orders they might have placed. Include customers who have not placed any orders.

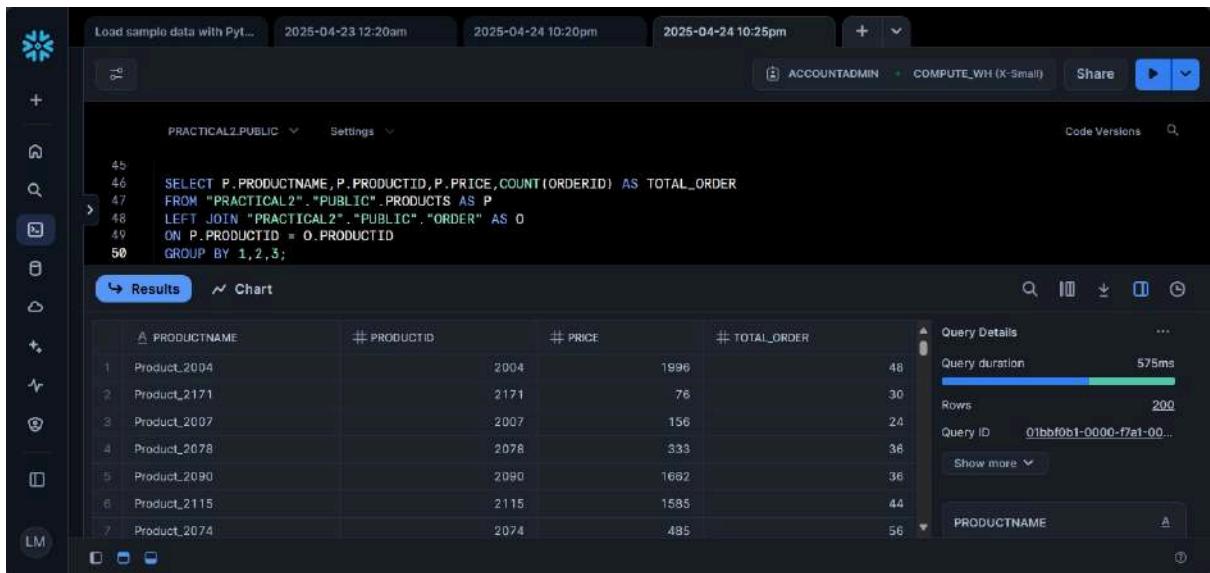


The screenshot shows a database query interface with the following details:

- Time Range:** 2025-04-24 10:20pm - 2025-04-24 10:25pm
- User:** ACCOUNTADMIN
- Compute Type:** COMPUTE_WH (X-Small)
- Code Version:** Code Versions
- Query:** SELECT C.CUSTOMERID, CUSTOMERNAME, COUNTRY, ORDERID, ORDERDATE, PRODUCTID, QUANTITY FROM PRACTICAL2.PUBLIC.CUSTOMER AS C LEFT JOIN "PRACTICAL2"."PUBLIC"."ORDER" AS O ON C.CUSTOMERID = O.CUSTOMERID;
- Results:** A table with 6 rows showing customer information and their orders.
- Query Details:** Query duration: 708ms, Rows: 8K, Query ID: 01bbf0a4-0000-f712-0...

#	CUSTOMERID	CUSTOMERNAME	COUNTRY	#	ORDERID	ORDERDATE	#	PRODUCTID	#	QUANTITY
1	1251	Customer_1251	Germany	1		2023-06-10	2014		10	
2	1236	Customer_1236	Australia	2		2023-12-07	2004		5	
3	1170	Customer_1170	Germany	3		2024-10-26	2171		9	
4	1344	Customer_1344	Canada	4		2023-02-17	2007		2	
5	1319	Customer_1319	USA	5		2024-11-08	2061		2	
6	1185	Customer_1185	Australia	6		2024-11-23	2190		3	

Q4 List all products and how many times each was ordered (if any).



The screenshot shows a database query interface with the following details:

- Time Range:** 2025-04-24 10:20pm - 2025-04-24 10:25pm
- User:** ACCOUNTADMIN
- Compute Type:** COMPUTE_WH (X-Small)
- Code Version:** Code Versions
- Query:** SELECT P.PRODUCTNAME, P.PRODUCTID, P.PRICE, COUNT(ORDERID) AS TOTAL_ORDER FROM "PRACTICAL2"."PUBLIC".PRODUCTS AS P LEFT JOIN "PRACTICAL2"."PUBLIC"."ORDER" AS O ON P.PRODUCTID = O.PRODUCTID GROUP BY 1,2,3;
- Results:** A table with 7 rows showing product names, IDs, prices, and total order counts.
- Query Details:** Query duration: 575ms, Rows: 200, Query ID: 01bbf0b1-0000-f7a1-0...

#	PRODUCTNAME	PRODUCTID	#	PRICE	#	TOTAL_ORDER		
1	Product_2004		2004		1996		48	
2	Product_2171		2171		78		30	
3	Product_2007		2007		156		24	
4	Product_2078		2078		333		36	
5	Product_2090		2090		1682		36	
6	Product_2115		2115		1585		44	
7	Product_2074		2074		485		56	

Q5 Find all orders along with product details, including any products that might not have been ordered

The screenshot shows a database query interface with the following details:

- Database:** PRACTICAL2.PUBLIC
- Time Range:** 2025-04-23 12:20am to 2025-04-24 10:25pm
- User:** ACCOUNTADMIN
- Compute Type:** COMPUTE_WH (X-Small)
- Code:**

```
54
55
56 SELECT ORDERID,ORDERDATE,O.PRODUCTID,PRODUCTNAME,PRICE,QUANTITY
57 FROM "PRACTICAL2"."PUBLIC"."ORDER" AS O
58 RIGHT JOIN "PRACTICAL2"."PUBLIC"."PRODUCTS" AS P
59 ON O.PRODUCTID = P.PRODUCTID;
```
- Results:** A table with columns: ORDERID, ORDERDATE, PRODUCTID, PRODUCTNAME, PRICE, QUANTITY. The data is as follows:

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
5	2024-11-06	2061	Product_2061	1585	2
6	2024-11-23	2190	Product_2190	1755	3
7	2023-07-29	2099	Product_2099	1874	8

- Query Details:** Query duration: 576ms, Rows: 8K, Query ID: 01bbf0b9-0000-f711-00...

Q6 Which customers have made orders, and include customers even if they have never placed an order.

The screenshot shows a database query interface with the following details:

- Database:** PRACTICAL2.PUBLIC
- Time Range:** 2025-04-23 12:20am to 2025-04-24 10:25pm
- User:** ACCOUNTADMIN
- Compute Type:** COMPUTE_WH (X-Small)
- Code:**

```
61
62
63 SELECT C.CUSTOMERID,CUSTOMERNAME,COUNTRY,ORDERID,ORDERDATE,PRODUCTID,QUANTITY
64 FROM "PRACTICAL2"."PUBLIC"."CUSTOMER" AS C
65 RIGHT JOIN "PRACTICAL2"."PUBLIC"."ORDER" AS O
66 ON C.CUSTOMERID = O.CUSTOMERID;
```
- Results:** A table with columns: CUSTOMERID, CUSTOMERNAME, COUNTRY, ORDERID, ORDERDATE, PRODUCTID, QUANTITY. The data is as follows:

CUSTOMERID	CUSTOMERNAME	COUNTRY	ORDERID	ORDERDATE	PRODUCTID	QUANTITY
1251	Customer_1251	Germany	1	2023-06-10	2014	10
1236	Customer_1236	Australia	2	2023-12-07	2004	5
1170	Customer_1170	Germany	3	2024-10-26	2171	9
1344	Customer_1344	Canada	4	2023-02-17	2007	2
1319	Customer_1319	USA	5	2024-11-06	2061	2
1185	Customer_1185	Australia	6	2024-11-23	2190	3

- Query Details:** Query duration: 421ms, Rows: 8K, Query ID: 01bbf0c1-0000-f684-0...

Q7 List all customers and orders, showing NULLs where customers have not ordered or where orders have no customer info.

The screenshot shows a database interface with a dark theme. The top navigation bar includes tabs for 'Load sample data with Py...', '2025-04-23 12:20am', '2025-04-24 10:20pm', and '2025-04-24 10:25pm'. The current user is 'ACCOUNTADMIN' with a 'COMPUTE_WH (X-Small)' quota. The database selected is 'PRACTICAL2.PUBLIC'. The code editor contains the following SQL query:

```
67  
68  
69 | SELECT C.CUSTOMERID, CUSTOMERNAME, COUNTRY, ORDERID, ORDERDATE, PRODUCTID, QUANTITY  
70 | FROM "PRACTICAL2"."PUBLIC"."CUSTOMER" AS C  
71 | FULL OUTER JOIN "PRACTICAL2"."PUBLIC"."ORDER" AS O  
72 | ON C.CUSTOMERID = O.CUSTOMERID;
```

The results table has columns: CUSTOMERID, CUSTOMERNAME, COUNTRY, ORDERID, ORDERDATE, PRODUCTID, and QUANTITY. The data is as follows:

CUSTOMERID	CUSTOMERNAME	COUNTRY	ORDERID	ORDERDATE	PRODUCTID	QUANTITY
1	Customer_1251	Germany	1	2023-06-10	2014	10
2	Customer_1236	Australia	2	2023-12-07	2004	5
3	Customer_1170	Germany	3	2024-10-26	2171	9
4	Customer_1344	Canada	4	2023-02-17	2007	2
5	Customer_1319	USA	5	2024-11-08	2061	2
6	Customer_1185	Australia	6	2024-11-23	2190	3

On the right side, there is a 'Query Details' panel showing a duration of 707ms and 8K rows. A search bar at the bottom is set to 'CUSTOMERID'.

Q8 List all products and orders, showing NULLs where products were never ordered or orders are missing product info.

The screenshot shows a database interface with a dark theme. The top navigation bar includes tabs for 'Load sample data with Py...', '2025-04-23 12:20am', '2025-04-24 10:20pm', and '2025-04-24 10:25pm'. The current user is 'ACCOUNTADMIN' with a 'COMPUTE_WH (X-Small)' quota. The database selected is 'PRACTICAL2.PUBLIC'. The code editor contains the following SQL query:

```
73  
74  
75 | SELECT P.PRODUCTID, PRODUCTNAME, PRICE, ORDERID, ORDERDATE, CUSTOMERID, QUANTITY  
76 | FROM "PRACTICAL2"."PUBLIC"."PRODUCTS" AS P  
77 | FULL OUTER JOIN "PRACTICAL2"."PUBLIC"."ORDER" AS O  
78 | ON P.PRODUCTID = O.PRODUCTID;
```

The results table has columns: PRODUCTID, PRODUCTNAME, PRICE, ORDERID, ORDERDATE, CUSTOMERID, and QUANTITY. The data is as follows:

PRODUCTID	PRODUCTNAME	PRICE	ORDERID	ORDERDATE	CUSTOMERID	QUANTITY
1	Product_2014	522	1	2023-06-10	1251	10
2	Product_2004	1998	2	2023-12-07	1236	5
3	Product_2171	78	3	2024-10-26	1170	9
4	Product_2007	158	4	2023-02-17	1344	2
5	Product_2061	1595	5	2024-11-08	1319	2
6	Product_2190	1755	6	2024-11-23	1185	3
7	Product_2099	1674	7	2023-07-29	1011	8

On the right side, there is a 'Query Details' panel showing a duration of 272ms and 8K rows. A search bar at the bottom is set to 'PRODUCTID'.