**🔹 1. Total number of records in the Dataset**

Counts all entries in the Dataset table.

sql

CopyEdit

SELECT COUNT(\*) AS total\_records FROM Dataset;

**🔹 2. Average value of each category**

Groups data by category and shows the average value.

sql

CopyEdit

SELECT category, AVG(value) AS average\_value

FROM Dataset

GROUP BY category;

**🔹 3. Records with value greater than 8000**

Finds high-value transactions.

sql

CopyEdit

SELECT \* FROM Dataset

WHERE value > 8000;

**🔹 4. List of users with role 'Analyst'**

Simple filtering on the Users table.

sql

CopyEdit

SELECT \* FROM Users

WHERE role = 'Analyst';

**🔹 5. Find all operations performed by a specific user**

Filters logs by user\_id.

sql

CopyEdit

SELECT \* FROM analysis\_logs

WHERE user\_id = 5;

**🔹 6. Daily average value trend**

Aggregates average value per day.

sql

CopyEdit

SELECT date\_trunc('day', timestamp) AS day, AVG(value) AS avg\_value

FROM Dataset

GROUP BY day

ORDER BY day;

**🔹 7. Find categories with more than 30 records**

Uses HAVING to filter grouped data.

sql

CopyEdit

SELECT category, COUNT(\*) AS record\_count

FROM Dataset

GROUP BY category

HAVING COUNT(\*) > 30;

**🔹 8. Top 3 most frequent labels**

Uses aggregation and sorting to find common labels.

sql

CopyEdit

SELECT label, COUNT(\*) AS freq

FROM Dataset

GROUP BY label

ORDER BY freq DESC

LIMIT 3;

**🔹 9. Top 5 values by category (window function)**

Uses RANK() to get top 5 records within each category.

sql

CopyEdit

SELECT \*

FROM (

SELECT \*, RANK() OVER (PARTITION BY category ORDER BY value DESC) AS rnk

FROM Dataset

) ranked

WHERE rnk <= 5;

**🔹 10. Compare mean value of each label**

Shows which label group has the highest average value.

sql

CopyEdit

SELECT label, AVG(value) AS mean\_value

FROM Dataset

GROUP BY label

ORDER BY mean\_value DESC;

**🔹 11. Find earliest and latest timestamp in Dataset**

Min and max functions for time range.

sql

CopyEdit

SELECT MIN(timestamp) AS earliest, MAX(timestamp) AS latest

FROM Dataset;

**🔹 12. Operations log with user names (join)**

Shows which users performed what operations.

sql

CopyEdit

SELECT a.log\_id, u.name, u.role, a.operation, a.log\_time

FROM analysis\_logs a

JOIN Users u ON a.user\_id = u.user\_id

ORDER BY a.log\_time DESC;

**🔹 13. Number of operations done per user**

Aggregates logs per user.

sql

CopyEdit

SELECT u.name, COUNT(\*) AS operations\_count

FROM analysis\_logs a

JOIN Users u ON a.user\_id = u.user\_id

GROUP BY u.name

ORDER BY operations\_count DESC;

**🔹 14. Labels per category (distribution summary)**

Helps understand label usage by category.

sql

CopyEdit

SELECT category, label, COUNT(\*) AS count

FROM Dataset

GROUP BY category, label

ORDER BY category, count DESC;

**🔹 15. Users who performed more than 5 operations**

Identifies active users.

sql

CopyEdit

SELECT u.user\_id, u.name, COUNT(\*) AS operation\_count

FROM analysis\_logs a

JOIN Users u ON a.user\_id = u.user\_id

GROUP BY u.user\_id, u.name

HAVING COUNT(\*) > 5;

**🔹 16. Most recent 10 dataset entries**

Ordered by timestamp.

sql

CopyEdit

SELECT \* FROM Dataset

ORDER BY timestamp DESC

LIMIT 10;

**🔹 17. Find outliers using 2 standard deviations**

Identifies unusually high or low values.

sql

CopyEdit

SELECT \*

FROM Dataset

WHERE value > (SELECT AVG(value) + 2 \* STDDEV(value) FROM Dataset)

OR value < (SELECT AVG(value) - 2 \* STDDEV(value) FROM Dataset);

**🔹 18. Most frequent label in each category**

Uses RANK() to find top label per category.

sql

CopyEdit

SELECT category, label, count

FROM (

SELECT category, label, COUNT(\*) AS count,

RANK() OVER (PARTITION BY category ORDER BY COUNT(\*) DESC) AS rank

FROM Dataset

GROUP BY category, label

) ranked

WHERE rank = 1;

**🔹 19. User activity on a specific day**

Filter logs by date.

sql

CopyEdit

SELECT u.name, a.operation, a.log\_time

FROM analysis\_logs a

JOIN Users u ON a.user\_id = u.user\_id

WHERE DATE(a.log\_time) = '2025-07-25';

**🔹 20. Average value of each category by day**

Grouped on two levels: day and category.

sql

CopyEdit

SELECT date\_trunc('day', timestamp) AS day, category, AVG(value) AS avg\_value

FROM Dataset

GROUP BY day, category

ORDER BY day, category;