

# MOST IMPORTANT SQL TOPICS AND PATTERNS FOR INTERVIEW

*Created By SOURAV BANERJEE*

Demystify.Tech

## Topics that come up regularly (be familiar with them)

DML/DDDL/DCL concepts
Handling NULLs creatively (e.g. with COALESCE)
Subqueries and the impact of subqueries on efficiency of the query
Temporary tables
Self joins
Window functions like PARTITION, LEAD, LAG, NTILE
UDFs (user defined functions)
Use of indexes in querying to make operations faster.
What is the difference between HAVING and WHERE?
When are COUNT (*) and COUNT DISTINCT equal?
Can left and inner join return the same results?
What is the difference between RANK and DENSE RANK?
Can UNION and UNION ALL return the same results?
What are the best scenarios to use a self-join?
How can you use SQL to dedupe rows?
In what scenarios is the LAG function useful?
Use a window function to calculate a running total.
Why would you use GROUP BY ROLLUP?
CASE WHEN. So many great use cases!
Self joins. Common in product user behaviour .
DISTINCT and GROUP BY
Left vs outer joins. Need I say more?
UNION. Rarely discussed but frequent.
SUM and COUNT. Nail the foundations!
Date-time manipulation. This will set you apart.
String formatting, substring.
Window functions like rank and row. Absolute gold!
Subqueries. Because they always show up...
LAG and LEAD. What do you use these for?
Understanding indexing. More intermediate.
Running totals. A fun use case to learn.
MAX and MIN. More common than anyone says!
Using SUM CASE WHEN to count

Strings and time conversions
How to debug a query.
How to speed up a query.
Rank and dense rank
Use a SQL window function to do a cumulative sum.
Try to get the same SQL result with/without a subquery
SELECT and WHERE for filtering and selection
COUNT, SUM, MAX, GROUP BY, HAVING for aggregating data"
DISTINCT, COUNT DISTINCT for producing useful distinct lists and distinct aggregates
OUTER (e.g. LEFT) and INNER JOIN when/where to use them
SQL Pivoting

### Important Question on SQL before Interview

1. Why use the WITH statement? What does it help you do?
2. Can you use COUNT DISTINCT on two columns? How?
3. When does a full outer join actually make sense to use?
4. If you want to allow ties in ranking, RANK or DENSE RANK?
5. Can UNION and UNION ALL return the same results?
6. When is it a bad idea to use a subquery?
7. What is the best approach to doing a running total?
8. In what scenarios is the LEAD function useful?
9. If an error is thrown using SQL, where do you start looking?
10. Inner joining tables A and B returns 50 rows. Left join B to A returns 125 rows. What's going on here?

*Important, but more rare topics (useful to know)*

1. Recursive CTEs
2. Dynamic SQL generation
3. Query optimization
4. Materialized views

Demystify.Tech