



SQL

ORDER OF EXECUTION





In SQL - the "**order of execution**" is the order in which clauses within a query are evaluated.

This is often **different** to the order that the query is written in.

Understanding this order can be useful when looking to **debug** & **optimise** queries





FROM

Points to the table where the
required data is stored





JOIN + ON

Obtains the matching records based upon shared column(s) when the required data lives in multiple tables





WHERE

Filters the requested source data at a row level. Does not allow column aliases or aggregate functions to be applied





GROUP BY

Groups rows with the same values together. Often used with aggregate functions. Column aliases are allowed





HAVING

Applies conditions and/or filters
after aggregation takes place.
Does not allow column aliases





SELECT

+ WINDOW FUNCTIONS

Select specifies the columns to be returned. Window functions look at values from a set of rows (the window) and based on this, return a value for each row





ORDER BY

Sorts the resulting rows after filtering & aggregation. Column aliases are allowed





LIMIT / FETCH

Limits the number of rows that
are returned by the query





▶ **FROM**



▶ **JOIN + ON**



▶ **WHERE**



▶ **GROUP BY**



▶ **HAVING**



▶ **SELECT + WINDOW FUNCTIONS**



▶ **ORDER BY**



▶ **LIMIT / FETCH**

