

Feature / Join type	CROSS JOIN	INNER JOIN	LEFT JOIN	RIGHT JOIN	FULL OUTER JOIN	UNION
Definition	Cartesian product of both tables	Matches rows with matching keys	All matches + unmatched from left table	All matches + unmatched from right table	All matches + unmatched from both sides	Combines rows from 2 queries
Join condition	Not required	Required	Required	Required	Required	Not applicable
Null handling	None; all rows paired	Rows with no match are excluded	Fills NULLs for unmatched right rows	Fills NULLs for unmatched left rows	Fills NULLs on both sides where no match	Duplicate rows removed unless UNION ALL
Result size	All combinations (large result sets)	Only matched rows	Matched + unmatched left	Matched + unmatched right	Matched + unmatched on both sides	Rows stacked vertically
Use cases	Generate all combinations	Standard matching (e.g., FK-PK)	Keep all left-side data	Keep all right-side data	Keep everything	Merge results from two SELECTs
Typical example	Product variants (e.g., sizes × colors)	Orders and customers	Customers and all their orders (even none)	Orders and their customers (even unknown)	Employees and departments (including unassigned)	Combining SELECT results from different tables
SQL syntax	FROM A CROSS JOIN B	FROM A INNER JOIN B ON ...	FROM A LEFT JOIN B ON ...	FROM A RIGHT JOIN B ON ...	FROM A FULL OUTER JOIN B ON ...	SELECT ... UNION SELECT ...
Performance caution	Risky on large tables	Depends on index and join condition	Can slow down with NULLs	Similar to LEFT JOIN	Often slower than INNER JOIN	Watch for duplicate rows
Filtering with WHERE	Often required to reduce size	Common for filtering matched values	Can refine unmatched results	Can refine unmatched results	Often used for final shaping	WHERE can be used in each SELECT