|  |  |
| --- | --- |
| **SQL query solution patterns**  -what can be done with LEFT/RIGHT OUTER JOIN can also be done by a correlated scalar subquery in the select list that can then be renamed with an alias. But in-efficient. You repeat the scalar subquery for every value to be fetched.  -CROSS JOIN used for generating sequence of values (either using ROW\_NUMBER or table of 0 to 9 values) with TOP for cutoff, mimicking a loop and transposing/pivoting/unpivoting. For some recursive problems, recursive CTE can be used in place of Loop.  -when performing JOINs, think carefully before using AGGREGATE functions if the relationship is not one-to-one. Row repetition! Tables that form the ‘many’ part of the relationship have to be aggregated before joining.  -FULL OUTER JOIN is the UNION of results of LET and RIGHT OUTER JOINs.  -use EXISTS and NOT EXISTS where ever you IN and NOT IN being used. |  |