Join vs Subqueries

What do you mean by Joins?

JOIN is an operation that exists in SQL that helps us to combine rows from two or more tables based on a related column between them. When many tables exist in the FROM clause of a query, a join is done. Any column from any of these tables can be selected from the query's choose list. A cartesian product is created if the join condition is missing or faulty.

Types of JOINS:

1. **INNER JOIN**: This returns a resulting table that has matching values from both the table and all the tables.
2. **LEFT JOIN**: This returns a resulting table that all the data from the left table and the matched data from the right table.
3. **RIGHT JOIN**: This returns a resulting table that all the data from the right table and the matched data from the left table.
4. **FULL JOIN**: This returns a resulting table that contains all data when there is a match on left or right table data.

What do you mean by Subquery?

Subqueries is a way of using queries within a particular query. It is accomplished using the parenthesis. The inner query is always executed first, followed by the execution of the outer query. Inner query results in a single value or set of values as an output. A subquery is a SELECT statement that is nested within another SQL statement's clause. They can be quite handy for selecting rows from a table based on data from the same or another table. A subquery is used to return data that will be utilized as a condition in the main query to further limit the data that may be retrieved. WHERE clause, SELECT clause, and FROM clause are the SQL clauses where the subquery can be used. Let us say we have a query Q2 and its data will be used by query Q1, then we can say that Q2 will be the Inner query of Main query Q1. Also, this query Q2 is called sub-query.

Join vs Subqueries

Joins	Subquery
Joins execute faster than subqueries	Subqueries execute slower than joins
In most cases, the retrieval time of query using joins is usually faster than subqueries.	There are very few scenarios in which retrieval time of query using subquery will be faster
Joins maximize the calculation burden on the database	Subquery keeps the responsibility of calculation on the user.
Joins are difficult to implement and understand	Subqueries are easier to understand and implement than joins
In Joins, we cannot use the result of other queries.	Subquery helps us to use the result of another query in the outer query.

Table 1 Join vs Subqueries