**SQL JOINS**

SQL joins are used to combine data from two or more tables based on a common column or set of columns. There are several types of joins in SQL, including inner join, left join, right join, and full outer join.

**INNER JOIN**

An inner join returns only the rows from both tables that have matching values in the specified column or columns. In other words, it returns only the intersection of the two tables.

The syntax for an inner join is as follows:

SELECT column\_name(s)

FROM table1

INNER JOIN table2

ON table1.column\_name = table2.column\_name;

Here, table1 and table2 are the names of the tables being joined, and column\_name is the name of the column that the tables have in common. The ON keyword specifies the condition for the join.

**LEFT JOIN**

A left join returns all the rows from the left table (i.e., the first table mentioned in the join statement) and the matching rows from the right table. If there are no matching rows in the right table, the result will contain NULL values for the columns in the right table.

The syntax for a left join is as follows:

SELECT column\_name(s)

FROM table1

LEFT JOIN table2

ON table1.column\_name = table2.column\_name;

In this case, table1 is the left table, and table2 is the right table.

**RIGHT JOIN**

A right join is similar to a left join, but it returns all the rows from the right table and the matching rows from the left table. If there are no matching rows in the left table, the result will contain NULL values for the columns in the left table.

The syntax for a right join is as follows:

SELECT column\_name(s)

FROM table1

RIGHT JOIN table2

ON table1.column\_name = table2.column\_name;

In this case, table1 is the right table, and table2 is the left table.

**FULL OUTER JOIN**

A full outer join returns all the rows from both tables and includes NULL values for any columns that do not have matching values in the other table.

The syntax for a full outer join is as follows:

SELECT column\_name(s)

FROM table1

FULL OUTER JOIN table2

ON table1.column\_name = table2.column\_name;

In this case, both table1 and table2 are included in the result, and NULL values are included for any columns that do not have matching values in the other table.

It is worth noting that some databases do not support the full outer join syntax, in which case you may need to use a combination of left and right joins to achieve the same result.

**CROSS JOIN**

A cross join (also known as a Cartesian product) returns all possible combinations of rows from both tables. In other words, it returns the product of the two tables.

The syntax for a cross join is as follows:

SELECT column\_name(s)

FROM table1

CROSS JOIN table2;

**SELF JOIN**

A self-join is a join in which a table is joined with itself. This is useful when you want to compare or combine rows within the same table.

The syntax for a self-join is similar to that of other joins, but you use aliases to differentiate between the different instances of the same table:

SELECT column\_name(s)

FROM table1 t1

JOIN table1 t2

ON t1.column\_name = t2.column\_name;