Joins

CROSS JOINS

1. in SQL, a cross join(also known as Cartesian Product) is a type of join that returns the Cartesian Product of the two tables being joined , in other words it returns all the possible combinations of rows from the two tables.

INNER JOINS

1. In SQL ,it is a type of join operation that combines data from two or more tables based on a specified condition . The inner join returns the table from the two rows that satisfy the conditions.

LEFT JOINS

1. also known as left outer join , is a type of SQL operation that returns all the rows from the left table and matching rows from the right table. If there are matching rows in the right table the result will contain null values in the column that comes from the right table.

RIGHT JOINS

1. also known as right outer join is a type of SQL join operation that returns all the rows from the right table and matching rows from the left table

OUTER JOINS

1. a full outer join sometimes called a full join , is a type of join operation that returns all the matching rows from both left and right tables.

SELF JOINS

1. is a type of joins in which a table is joined with itself . This means that table is treated as two separate tables, with each row in the table being compared to every other row in the same table
2. are used when we want to compare the values of two different rows within the same table . For example , you might use a self join to compare the salaries of the two employees who might work in same department.

SETS

UNION

1. the union operator is used to combine the results of two or more SELECT statements into a single result set . The UNION operator **removes duplicate rows** between the various SELECT statements.

UNION ALL

1. the union all operator is similar to UNION operator but it does not remove duplicate rows from the result set.

INTERSECT

1. this operator returns only the rows that appear in both result of the two SELECT statements.

EXCEPT

1. the EXCEPT or MINUS operator returns only the distinct rows that appear in the first result set but not in the second result of the two SELECT statements.