JOINS

Joins are used to combine data from two or more tables based on a related column. Joins help retrieve meaningful information by linking tables using a common key.

There are 6 types of joins

1. INNER JOIN

   An INNER JOIN is used to return only the rows that have matching values in both tables. If there is no match, the row is excluded from the result.

   SYNTAX: SELECT columns FROM table1 INNER JOIN table2 ON table1.common_column = table2.common_column;

2. LEFT JOIN

   Returns all rows from the left table and matching rows from the right table. If there is no match, NULL values are returned.

   SYNTAX: SELECT columns FROM table1 LEFT JOIN table2 ON table1.common_column = table2.common_column;

3. RIGHT JOIN

   Returns all rows from the right table and matching rows from the left table. If there is no match, NULL values are returned.

   SYNTAX: SELECT columns FROM table1 RIGHT JOIN table2 ON table1.common_column = table2.common_column;

4. CROSS JOIN

   Returns all possible combinations of rows from both tables (Cartesian Product). No need for an ON condition.

   SYNTAX: SELECT columns FROM table1 CROSS JOIN table2;

5. SELF JOIN

   A table joins itself to compare rows within the same table.

   SYNTAX: SELECT A.column_name, B.column_name FROM table_name A INNER JOIN table_name B ON A.common_column = B.common_column;

6. FULL JOIN

   Returns all rows from both tables, filling NULLs where there is no match. MySQL does not support FULL JOIN directly, but you can simulate it using UNION.

   SYNTAX: SELECT * FROM table1 LEFT JOIN table2 ON table1.common_column = table2.common_column

   UNION

   SELECT * FROM table1 RIGHT JOIN table2 ON table1.common_column = table2.common_column;