



JOINS IN SQL

SQL Series Part 9

-Mayuri Dandekar

JOINS

JOIN is a method of **combining information** from two tables.

INNER JOIN -- Returns records that have matching values in both tables

LEFT JOIN -- Returns all records from the left table, and the matched records from the right table

RIGHT JOIN -- Returns all records from the right table, and the matched records from the left table

FULL JOIN -- Returns all records when there is a match in either left or right table

Sample dataset used

	emp_id	emp_name	department_id
▶	1	john	101
	2	alice	102
	3	bob	101
	4	mary	103

	department_id	department_name
▶	101	it
	102	hr
	103	marketing
	104	sales

INNER JOIN

An INNER JOIN retrieves rows from both tables where **there is a match in the specified columns**.

```
3 • SELECT employee.emp_id, employee.emp_name, department.department_name
4 FROM employee
5 INNER JOIN department ON employee.department_id = department.department_id;
```

<





Result Grid |   Filter Rows: | Export:  | Wrap Cell Content: 

	emp_id	emp_name	department_name
▶	3	bob	it
	1	john	it
	2	alice	hr
	4	mary	marketing

LEFT JOIN

A LEFT JOIN returns all rows from the left table and matching rows from the right table, with NULL values where there is no match in the right table.





```
7 • SELECT employee.emp_id, employee.emp_name, department.department_name
8 FROM employee
9 LEFT JOIN department ON employee.department_id = department.department_id;
```

Result Grid			Filter Rows: <input type="text"/>	Export: 	Wrap Cell Content: 
	emp_id	emp_name	department_name		
▶	1	john	it		
	2	alice	hr		
	3	bob	it		
	4	mary	marketing		

RIGHT JOIN

A RIGHT JOIN returns all rows from the right table and matching rows from the left table, with NULL values where there is no match in the left table

```
11 • SELECT employee.emp_id, employee.emp_name, department.department_name
12 FROM employee
13 RIGHT JOIN department ON employee.department_id = department.department_id;
```

<			
Result Grid   Filter Rows: <input type="text"/> Export:  Wrap Cell Content: 			
	emp_id	emp_name	department_name
▶	3	bob	it
	1	john	it
	2	alice	hr
	4	mary	marketing
	NULL	NULL	sales

FULL JOIN

A FULL JOIN returns all rows from both tables and NULL values where there is no match.

```
15 • SELECT employee.emp_id, employee.emp_name, department.department_name
16 FROM employee
17 LEFT JOIN department ON employee.department_id = department.department_id
18 UNION ALL
19 SELECT employee.emp_id, employee.emp_name, department.department_name
20 FROM employee
21 RIGHT JOIN department ON employee.department_id = department.department_id
22 WHERE employee.emp_id = NULL;
```

emp_id	emp_name	department_name
1	john	it
2	alice	hr
3	bob	it
4	mary	marketing

Note—

My database does not support full join directly, so I tried it by combining LEFT JOIN, RIGHT JOIN & UNION ALL

CROSS JOIN

A CROSS JOIN returns the Cartesian product of the two tables, meaning it combines every row from the first table with every row from the second table.

```
19 • SELECT employee.emp_id, employee.emp_name, department.department_name
20 FROM employee
21 CROSS JOIN department;
22
```

	emp_id	emp_name	department_name
▶	4	mary	it
	3	bob	it
	2	alice	it
	1	john	it
	4	mary	hr
	3	bob	hr
	2	alice	hr
	1	john	hr
	4	mary	marketing
	3	bob	marketing
	2	alice	marketing
	1	john	marketing
	4	mary	sales
	3	bob	sales
	2	alice	sales
	1	john	sales




SELF JOIN

A self join is a special type of join where a table is joined with itself. This is useful when you have hierarchical data or need to compare rows within the same table.

	emp_id	emp_name	manager_id
▶	1	john	0
	2	alice	1
	3	bob	1
	4	mary	2

```
38 • SELECT c.emp_name AS employee_name, m.emp_name AS manager_name
39 FROM company c
40 LEFT JOIN company m ON c.manager_id = m.emp_id;
```

<

Result Grid |  Filter Rows: | Export:  | Wrap Cell Content: 




	employee_name	manager_name
▶	john	NULL
	alice	john
	bob	john
	mary	alice

UNION

UNION is used to combine the results of two or more SELECT statements into a single result set. It removes duplicate rows by default

```
49 • SELECT emp_id, emp_name FROM employee1
50 UNION
51 SELECT emp_id, emp_name FROM employee2;
52
```

<




Result Grid   Filter Rows: Export:  Wrap

	emp_id	emp_name
▶	1	john
	2	alice
	3	bob
	4	mary

UNION ALL

UNION ALL is similar to UNION, but it retains duplicate rows from the combined result sets.

```
53 • SELECT emp_id, emp_name FROM employee1  
54 UNION ALL  
55 SELECT emp_id, emp_name FROM employee2;  
56
```

<	Result Grid			Filter Rows: <input type="text"/>	Export: 	Wrap Cell
	emp_id	emp_name				
▶	1	john				
	2	alice				
	1	john				
	2	alice				
	3	bob				
	4	mary				



THANK YOU!!!

SQL Series Part 9

-Mayuri Dandekar