



CENTRO UNIVERSITÁRIO
SANTO AGOSTINHO

Tipos de JOINS

BANCO DE DADOS

Aluno: Marcus Mlkael Rodrigues Vieira

Inner JOIN

Retorna apenas os registros com correspondência nas duas tabelas.

Ex:

```
SELECT c.nome, p.data_pedido  
FROM clientes c  
INNER JOIN pedidos p ON c.id_cliente = p.id_cliente;
```

Resultado:

Tabela Normal

Id_cliente [PK] integer	nome character varying (100)
1	João Silva
2	Maria Oliveira
3	Carlos Santos
4	Ana Souza
5	Pedro Rocha
6	Marcus Mikael
7	Anderson Soares

Id_pedido [PK] integer	Id_cliente integer	data_pedido date
1	1	2024-03-01
3	2	2024-03-10
4	3	2024-03-15
5	4	2024-03-20
6	5	2024-03-25
10	5	2024-04-15
11	8	2024-08-20
12	10	2024-10-20

Inner JOIN

nome character varying (100)	data_pedido date
João Silva	2024-03-01
Maria Oliveira	2024-03-10
Carlos Santos	2024-03-15
Ana Souza	2024-03-20
Pedro Rocha	2024-03-25
Pedro Rocha	2024-04-15

Left JOIN

Retorna todos os registros da tabela da esquerda e os correspondentes da direita

Ex:

```
SELECT c.nome, p.data_pedido  
FROM clientes c  
LEFT JOIN pedidos p ON c.id_cliente = p.id_cliente;
```

Resultado:

Tabela Normal

Id_cliente [PK] integer	nome character varying (100)
1	João Silva
2	Maria Oliveira
3	Carlos Santos
4	Ana Souza
5	Pedro Rocha
6	Marcus Mikael
7	Anderson Soares

Id_pedido [PK] integer	Id_cliente integer	data_pedido date
1	1	2024-03-01
3	2	2024-03-10
4	3	2024-03-15
5	4	2024-03-20
6	5	2024-03-25
10	5	2024-04-15
11	8	2024-08-20
12	10	2024-10-20

Left JOIN

nome character varying (100)	data_pedido date
João Silva	2024-03-01
Maria Oliveira	2024-03-10
Carlos Santos	2024-03-15
Ana Souza	2024-03-20
Pedro Rocha	2024-03-25
Pedro Rocha	2024-04-15
Marcus Mikael	[null]
Anderson Soares	[null]

Right JOIN

Retorna todos os registros da tabela da direita e os correspondentes da esquerda.

Ex:

```
SELECT c.nome, p.data_pedido  
FROM clientes c  
RIGHT JOIN pedidos p ON c.id_cliente = p.id_cliente;
```


Resultado:

Tabela Normal

Id_cliente [PK] integer	nome character varying (100)
1	João Silva
2	Maria Oliveira
3	Carlos Santos
4	Ana Souza
5	Pedro Rocha
6	Marcus Mikael
7	Anderson Soares

Id_pedido [PK] integer	Id_cliente integer	data_pedido date
1	1	2024-03-01
3	2	2024-03-10
4	3	2024-03-15
5	4	2024-03-20
6	5	2024-03-25
10	5	2024-04-15
11	8	2024-08-20
12	10	2024-10-20

Right JOIN

nome character varying (100)	data_pedido date
João Silva	2024-03-01
Maria Oliveira	2024-03-10
Carlos Santos	2024-03-15
Ana Souza	2024-03-20
Pedro Rocha	2024-03-25
Pedro Rocha	2024-04-15
[null]	2024-08-20
[null]	2024-10-20

Full JOIN

Retorna todos os registros das duas tabelas, combinando onde possível.

Ex:

```
SELECT c.nome, p.data_pedido  
FROM clientes c  
FULL JOIN pedidos p ON c.id_cliente = p.id_cliente;
```


Resultado:

Tabela Normal

Id_cliente [PK] integer	nome character varying (100)
1	João Silva
2	Maria Oliveira
3	Carlos Santos
4	Ana Souza
5	Pedro Rocha
6	Marcus Mikael
7	Anderson Soares

Id_pedido [PK] integer	Id_cliente integer	data_pedido date
1	1	2024-03-01
3	2	2024-03-10
4	3	2024-03-15
5	4	2024-03-20
6	5	2024-03-25
10	5	2024-04-15
11	8	2024-08-20
12	10	2024-10-20

Full JOIN

nome character varying (100)	data_pedido date
João Silva	2024-03-01
Maria Oliveira	2024-03-10
Carlos Santos	2024-03-15
Ana Souza	2024-03-20
Pedro Rocha	2024-03-25
Pedro Rocha	2024-04-15
[null]	2024-08-20
[null]	2024-10-20
Marcus Mikael	[null]
Anderson Soares	[null]

Cross JOIN

Combina cada linha da primeira tabela com todas da segunda.

Ex:

```
SELECT c.nome, p.data_pedido  
FROM clientes c  
CROSS JOIN pedidos p ORDER BY c.nome;
```

Resultado:

Tabela Normal

Id_cliente [PK] integer	nome character varying (100)
1	João Silva
2	Maria Oliveira
3	Carlos Santos
4	Ana Souza
5	Pedro Rocha
6	Marcus Mikael
7	Anderson Soares

Id_pedido [PK] integer	Id_cliente integer	data_pedido date
1	1	2024-03-01
3	2	2024-03-10
4	3	2024-03-15
5	4	2024-03-20
6	5	2024-03-25
10	5	2024-04-15
11	8	2024-08-20
12	10	2024-10-20

Cross JOIN

nome character varying (100)	data_pedido date
Ana Souza	2024-04-15
Ana Souza	2024-03-01
Ana Souza	2024-03-10
Ana Souza	2024-08-20
Ana Souza	2024-10-20
Ana Souza	2024-03-15
Ana Souza	2024-03-25
Ana Souza	2024-03-20
Anderson Soares	2024-04-15
Anderson Soares	2024-10-20
Anderson Soares	2024-03-15

Self JOIN

É um tipo de JOIN onde uma tabela se relaciona com ela mesma.

Usado quando há relacionamentos entre linhas da mesma tabela.

Ex:

```
SELECT c1.nome AS cliente, c2.nome AS indicado_por  
FROM clientes c1  
LEFT JOIN clientes c2 ON c1.id_indicado = c2.id_cliente  
ORDER BY c1.nome;
```

Processo:

Indicação de um amigo

```
ALTER TABLE clientes ADD COLUMN id_indicado INT;
```

```
UPDATE clientes SET id_indicado = 1 WHERE id_cliente = 2; -- Maria foi indicada por João
```

```
UPDATE clientes SET id_indicado = 1 WHERE id_cliente = 3; -- Carlos também
```

```
UPDATE clientes SET id_indicado = 2 WHERE id_cliente = 4; -- Ana por Maria
```

UPDATE clientes: Atualizar dados da tabela clientes

SET id_indicado = 1: Colocando o valor 1 no campo id_indicado, ou seja, indicando que João (id 1) foi quem indicou essa pessoa

WHERE id_cliente = 2: Essa atualização só deve acontecer para o cliente com ID 2 → Maria

Resultado:

Tabela Normal

Id_cliente [PK] integer	nome character varying (100)
1	João Silva
2	Maria Oliveira
3	Carlos Santos
4	Ana Souza
5	Pedro Rocha
6	Marcus Mikael
7	Anderson Soares

Id_pedido [PK] integer	Id_cliente integer	data_pedido date
1	1	2024-03-01
3	2	2024-03-10
4	3	2024-03-15
5	4	2024-03-20
6	5	2024-03-25
10	5	2024-04-15
11	8	2024-08-20
12	10	2024-10-20

Self JOIN

cliente character varying (100)	Indicado_por character varying (100)
Ana Souza	Maria Oliveira
Anderson Soares	[null]
Carlos Santos	João Silva
João Silva	[null]
Marcus Mikael	[null]
Maria Oliveira	João Silva
Pedro Rocha	[null]

Obrigado!