

# BANCO DE DADOS 2

## AULA 7

MSC SANDIR  
RODRIGUES CAMPOS

3”



# JUNÇÕES CRUZADAS

Exercício:

Criar uma consulta com junção cruzada onde os pedidos só estejam relacionadas com seu respectivo vendedor.

SINTAXE:

```
SELECT * FROM tabela1 CROSS JOIN tabela2 WHERE  
(condição que relacione apenas a cidade com seu respectivo estado)
```



# TIPOS DE JOIN

INNER

LEFT

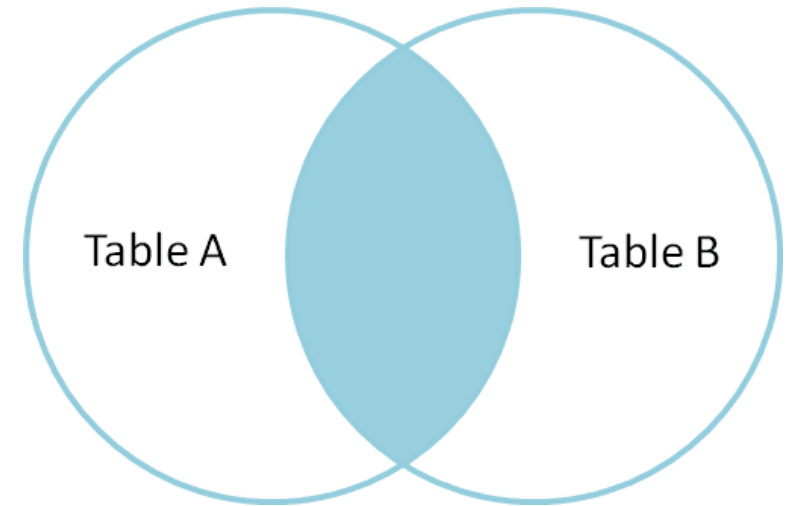
RIGHT

FULL

CROSS

# INNER JOIN

- O **INNER JOIN** é utilizado quando queremos retornar os registros que tenham correspondência nas duas tabelas presentes na junção.



# INNER JOIN

Exemplo: em um banco de dados de uma empresa, existem duas tabelas que se relacionam entre si

⌘ COD_FUNCIONARIO	⌘ NOME	⌘ COD_CARGO
1	JOSE	3
2	DANIELLA	1
3	ANA	2
4	CARLOS	(null)

# INNER JOIN

E a segunda possui os campos **código do cargo** e **descrição dos cargos**:

⚙️ COD_CARGO	⚙️ DESCRIÇÃO
1	VENDEDOR
2	CAIXA
3	GERENTE
4	ENTREGADOR

# INNER JOIN NO SQL

	cod_cargo integer	dedricao character varying(20)
1	1	vendedor
2	2	caixa
3	3	gerente
4	4	entregador

cargo

	cod_funcionario integer	nome character varying(20)	cod_cargo integer
1	1	jose	3
2	2	daniella	1
3	3	ana	2
4	4	carlos	

funcionario

- Exemplo: em um banco de dados de uma empresa, existem duas tabelas que se relacionam entre si

# SINTAXE INNER JOIN

---

Alias - Apelido

---

```
SELECT <select_list> FROM Tabela A INNER JOIN Tabela B  
ON A.Key = B.Key
```

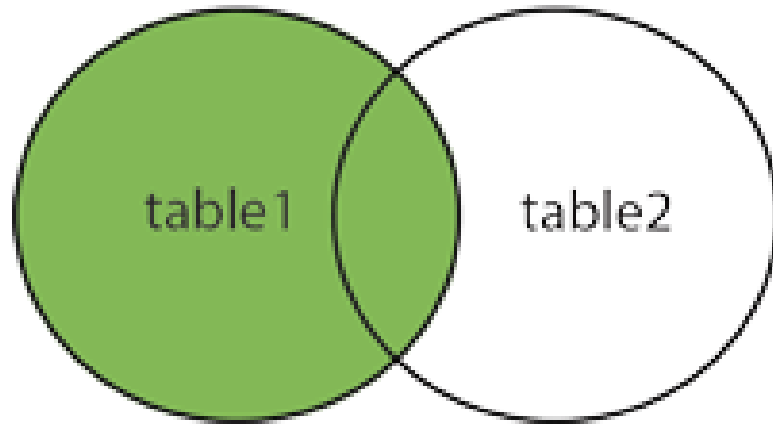
---

```
SELECT f.nome, f.cod_cargo, c.descricao FROM funcionario f  
INNER JOIN cargo c ON f.cod_cargo = c.cod_cargo;
```

# INNER JOIN RESULTADO

<code>nome</code> <code>character varying(20)</code>	<code>cod_cargo</code> <code>integer</code>	<code>descricao</code> <code>character varying(20)</code>
jose	3	gerente
daniella	1	vendedor
ana	2	caixa

## LEFT JOIN



# LEFT JOIN

- O LEFT JOIN é utilizado quando queremos retornar apenas os registros da tabela da esquerda (tabela que está antes da cláusula LEFT JOIN) e os registros que tenham correspondência na tabela da direita.

# SINTAXE LEFT JOIN

---

```
SELECT <select_list> FROM Tabela A LEFT JOIN  
Tabela B ON A.Key = B.Key
```

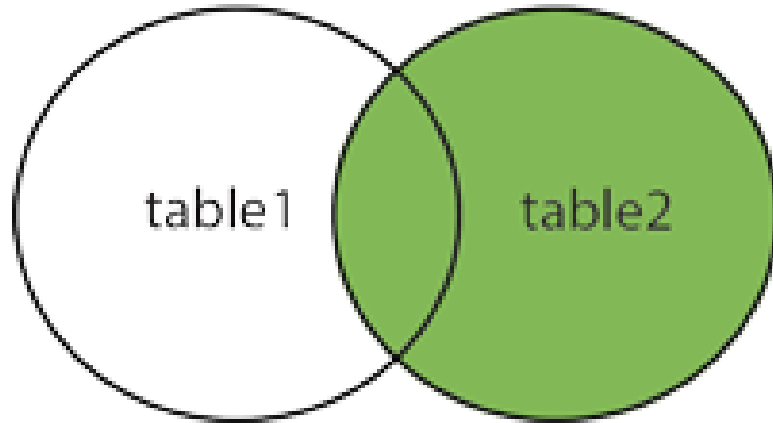
---

```
SELECT f.nome, f.cod_cargo, c.descricao FROM  
funcionario f LEFT JOIN cargo c ON f.cod_cargo =  
c.cod_cargo;
```

# LEFT JOIN RESULTADO

<code>nome</code> <code>character varying(20)</code>	<code>cod_cargo</code> <code>integer</code>	<code>descricao</code> <code>character varying(20)</code>
jose	3	gerente
daniella	1	vendedor
ana	2	caixa
carlos		

## RIGHT JOIN



# RIGHT JOIN

- O RIGHT JOIN é utilizado quando queremos retornar apenas os registros da tabela da direita (tabela que está após a cláusula RIGHT JOIN) e os registros que tenham correspondência na tabela da esquerda.

# SINTAXE RIGHT JOIN

```
SELECT <select_list> FROM Tabela A RIGHT JOIN Tabela B ON A.Key = B.Key
```

```
SELECT f.nome,f.cod_cargo,c.descricao FROM funcionario f RIGHT JOIN cargo c ON  
f.cod_cargo = c.cod_cargo;
```

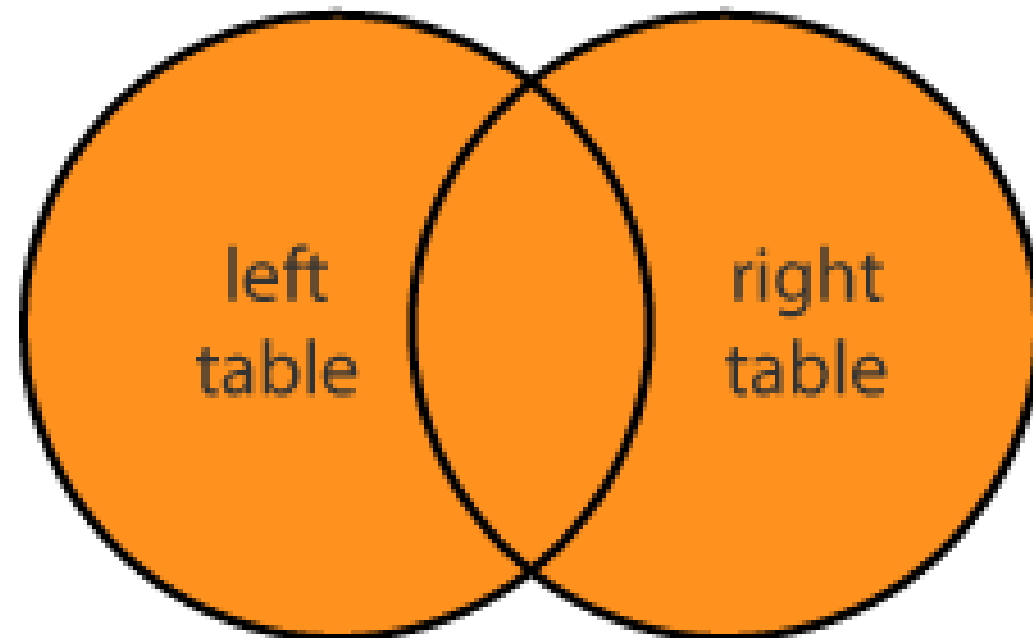
# RIGHT JOIN RESULTADO

<code>nome</code> <code>character varying(20)</code>	<code>cod_cargo</code> <code>integer</code>	<code>descricao</code> <code>character varying(20)</code>
jose	3	gerente
daniella	1	vendedor
ana	2	caixa
		entregador

# FULL JOIN

O FULL JOIN é utilizado quando queremos retornar registros que tenham correspondência em qualquer uma das tabelas presentes na junção.

## FULL JOIN



# SINTAXE FULL JOIN

---

```
SELECT <select_list> FROM Tabela A FULL JOIN  
Tabela B
```

---

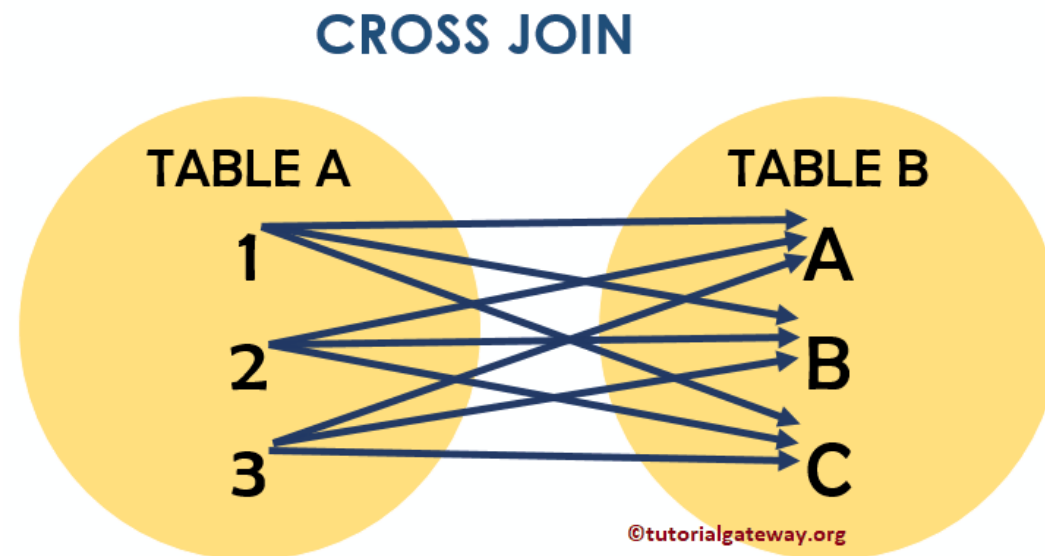
```
SELECT A.nome, A.cod_cargo, B.descricao,  
B.cod_cargo FROM funcionario A FULL JOIN cargo  
B;
```

# FULL JOIN RESULTADO

<code>nome</code> <code>character varying(20)</code>	<code>cod_cargo</code> <code>integer</code>	<code>descricao</code> <code>character varying(20)</code>
jose	3	gerente
daniella	1	vendedor
ana	2	caixa
carlos		
		entregador

# CROSS JOIN

- O CROSS JOIN é utilizado quando queremos retornar os registros realizando um cruzamento entre os dados das tabelas presentes na junção.



# SINTAXE CROSS JOIN

---

```
SELECT <select_list> FROM Tabela A CROSS JOIN  
Tabela B
```

---

```
SELECT f.nome, f.cod_cargo, c.descrição,  
c.cod_cargo FROM funcionario f CROSS JOIN  
cargo c;
```

# CROSS JOIN RESULTADO

nome character varying(20)	cod_cargo integer	descricao character varying(20)	cod_cargo integer
jose	3	vendedor	1
daniella	1	vendedor	1
ana	2	vendedor	1
carlos		vendedor	1
jose	3	caixa	2
daniella	1	caixa	2
ana	2	caixa	2
carlos		caixa	2
jose	3	gerente	3
daniella	1	gerente	3
ana	2	gerente	3
carlos		gerente	3
jose	3	entregador	4
daniella	1	entregador	4
ana	2	entregador	4
carlos		entregador	4

# JUNÇÕES INTERNAS

A utilização da cláusula INNER é o que caracteriza o comando para uma junção interna, porém, ele não é obrigatório. Pode parecer à primeira vista que as junções internas se equiparam com as junções cruzadas vistas anteriormente, até por que as duas consultas a seguir são equivalentes:

- **SELECT \* FROM tabela CROSS JOIN tabela2**
- **SELECT \* FROM tabela INNER JOIN tabela2 ON TRUE**



# JUNÇÕES INTERNAS

- Mas nas junções internas é sempre obrigatória a especificação de condição de junção, ou seja, quais linhas de uma tabela têm alguma ligação com a linha de outra tabela. Para isso podemos utilizar uma das cláusulas ON ou USING ou utilizar a palavra NATURAL no nosso comando.
- A cláusula ON é o mais comumente utilizado por se assemelhar com a cláusula WHERE, ou seja, um par de linhas de Tabela1 e Tabela2 são correspondentes, se a expressão da cláusula ON produz um resultado verdade (true) para este par de linhas.



# JUNÇÕES INTERNAS

- A cláusula **USING** traz alguma semelhança com o ON, por também retornar um valor verdadeiro ou falso para aquele conjunto de linhas, no entanto, ele é uma forma mais rápida e abreviada de criação da consulta.
- Passando um nome de coluna, a execução desta consulta irá procurar nas tabelas a coluna especificada e comparar as duas.
- Por exemplo, `t1 INNER JOIN t2 USING (a, b, c)` equivale a `t1 INNER JOIN t2 ON (t1.a = t2.a AND t1.b = t2.b AND t1.c = t2.c)`. Portanto, a consulta anterior equivale à consulta abaixo:
- **SELECT \* FROM tabela1 INNER JOIN tabela2 USING (idtabela1)**



Para facilitar mais, existe a utilização de **NATURAL**, que nada mais é abreviação de **USING**. Com **NATURAL**, a consulta encontrará todas as colunas que tem nomes iguais nas duas tabelas e fará a comparação de igualdade.

O exemplo de **USING** anterior equivale ao seguinte:

```
SELECT * FROM tabela1 NATURAL INNER JOIN tabela2
```

# JUNÇÕES INTERNAS



**Fornecedor**

CNPJ	Razao_Social	Endereco	Cidade
17624092000185	Alpha Soluções em Informática	Rua 9 de Julho, 738	Marília
47132482000176	MicroSystem Informática	Av. da Saudade, 98	Bauru
56021482000112	J.W. Sistemas	Rua Paes Leme, 172	Marília
27385372000136	ADABAS S/A	Rua 9 de Julho, 829	Marília

**Cliente**

Codigo	Nome	Endereco	Cidade
1	Francisco Santos	Rua XV de Novembro, 16	
2	Claudete Pereira		
3	José Souza	Rua das Bromélias, 871	
4	Sônia Santos Oliveira		Bauru

**Produto**

Codigo	Descricao	Estoque	Valor	CNPJ_Fornecedo r
100	Micro Pentium IV	14	850	56021482000112
200	Impressora EPSON	8	280	27385372000136
300	Scanner Genius	9	190	
400	Disquete 3M	30	10	17624092000185

**Nota**

Numero	Data	Tipo_Pagto	Codigo_Cliente
1001	18/05/2002	A Vista	2
1002	18/05/2002	A Vista	
1003	19/05/2002	A Prazo	4
1004	20/05/2002	A Vista	3

**Item**

Numero_Nota	Codigo_Produto	Qtde
1001	100	1
1001	400	5
1002	200	1
1003	300	1
1004	200	2
1004	300	1

1. Faça uma consulta que mostre a descrição e o estoque do produto e o nome do fornecedor do produto
2. Faça uma consulta que mostre o número, data, valor e o nome do cliente de todas as notas
3. Faça uma consulta que mostre o nome do cliente e a soma das notas desse cliente
4. Faça uma consulta que mostre os fornecedores que tenham produtos com estoque igual a zero
5. Faça uma consulta que mostre o valor total das notas por tipo de pagamento

# TRABALHO DE CONSULTAS

---

Qual é o produto que os clientes de Brasília mais compram? (caso não haja nenhum de BSB – popular pelo menos 5 tuplas)

---

Qual vendedor tem mais clientes de fora de Brasília ?

---

Quantos clientes tem o principal vendedor?

---

Qual é o produto mais vendido pelo maior vendedor?

---

Qual é o produto vendido pelo pior vendedor (que tenha pelo menos uma venda)?

