

Junções - revisão

JOIN

PRODUTO CARTESIANO

CLIENTE

ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA

ID	DATA	CLIID
1	10/01/2013	1
2	07/02/2013	3
3	23/03/2013	3

SELECT * FROM CLIENTE, VENDA

CLIENTE		VENDA		
ID	NOME	CPF	ID	DATA

1º Resulta em uma nova tabela

2º Possui colunas das duas tabelas

PRODUTO CARTESIANO

CLIENTE

ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA

ID	DATA	CLIID
1	10/01/2013	1
2	07/02/2013	3
3	23/03/2013	3

SELECT * FROM CLIENTE , VENDA

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
1	JOÃO SILVA	258.159.752-85	2	07/02/2013	3
1	JOAO SILVA	258.159.752-85	3	23/03/2013	3
2	MARCOS PAULO	658.456.552-65	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65	2	07/02/2013	3
2	MARCOS PAULO	658.456.552-65	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

3º É a combinação de todos os registros
independente de estar correto

PRODUTO CARTESIANO

CLIENTE

ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA

ID	DATA	CLIID
1	10/01/2013	1
2	07/02/2013	3
3	23/03/2013	3

SELECT * FROM CLIENTE , VENDA

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
1	JOÃO SILVA	258.159.752-85	2	07/02/2013	3
1	JOAO SILVA	258.159.752-85	3	23/03/2013	3
2	MARCOS PAULO	658.456.552-65	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65	2	07/02/2013	3
2	MARCOS PAULO	658.456.552-65	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

4º É possível que o resultado possua registros inconsistentes...

JOIN
só faz sentido combinado com a
condição de junção
Comparar a PK e a sua referência

SELECCIONAR COM PRODUTO CARTESIANO

CLIENTE

ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA

ID	DATA	CLIID
1	10/01/2013	1
2	07/02/2013	3
3	23/03/2013	3

`SELECT * FROM cliente, venda WHERE cliente.id = venda.cliid`

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
1	JOÃO SILVA	258.159.752-85	2	07/02/2013	3
1	JOÃO SILVA	258.159.752-85	3	23/03/2013	3
2	MARCOS PAULO	658.456.552-65	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65	2	07/02/2013	3
2	MARCOS PAULO	658.456.552-65	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

5º Combinado com WHERE (selecionar), os registros da última tabela ficam corretas!!!

SELEÇÃO COM PRODUTO CARTESIANO

CLIENTE

ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA

ID	DATA	CLIID
1	10/01/2013	1
2	07/02/2013	3
3	23/03/2013	3

`SELECT * FROM cliente, venda WHERE cliente.id = venda.cliid`

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
1	JOÃO SILVA	258.159.752-85	2	07/02/2013	3
1	JOÃO SILVA	258.159.752-85	3	23/03/2013	3
2	MARCOS PAULO	658.456.552-65	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65	2	07/02/2013	3
2	MARCOS PAULO	658.456.552-65	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

6º O resultado do SQL acima é uma tabela composta por registros que possuem vínculo nas DUAS TABELAS

JOIN

- É uma operação binária;
- Realizada somente a cada 2 tabelas;
- Quando há mais que 2 tabelas
 - É gerado um resultado parcial;
 - E deste resultado é realizado uma nova junção.

INNER JOIN

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65	2	07/02/2013	3
3	MARIA ANTONIA	458.259.952-09	3	23/03/2013	3

```
SELECT * FROM cliente INNER JOIN venda ON cliente.id = venda.cliid
```

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

O INNER JOIN produz o mesmo resultado da operação produto cartesiano com selecionar

Junção

- Comece pelo FROM;
 - No enunciado terá todas as tabelas necessárias
- Verifique se as tabelas tem relação;
 - Caso não tenha, acrescente as tabelas necessárias;
 - Aqui é necessário entender sobre associações;
- Defina as condições de junção;
 - Aqui é preciso saber sobre PK e FK;
 - Saber a respeito do identificador e sua referência;
- Defina as condições do enunciado (seleção);
- Agora sim, defina as colunas (projeção).

Limites de JOIN

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65	2	07/02/2013	3
3	MARIA ANTONIA	458.259.952-09	3	23/03/2013	3

```
SELECT * FROM cliente INNER JOIN venda ON cliente.id = venda.cliid
```

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

E se precisarmos apresentar TODOS OS CLIENTES e as vendas desses cliente SE POSSUIR?

INNER JOIN

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65	2	07/02/2013	3
3	MARIA ANTONIA	458.259.952-09	3	23/03/2013	3

`SELECT * FROM cliente INNER JOIN venda ON cliente.id = venda.cliid`



CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65	null		
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

Com JOIN ou INNER JOIN não é possível efetuar esta consulta.

OUTER JOIN

Para que serve?

OUTER JOIN

- Qual a diferença?
 - OUTER JOIN é uma junção que traz também os dados que não satisfazem a condição de junção;
 - Linhas que não tem “par” – a referência do identificador.
- Quando utilizar?
 - Quando você precisa de uma lista COMPLETA de uma das tabelas e SE POSSUIR, os registros da outra tabela

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65	null		
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

OUTER JOIN

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	2	07/02/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

JOIN OU INNER JOIN

OUTER JOIN

- A operação do JOIN ou INNER JOIN não resulta em colunas NULAS, pois são compostas pelos registros que possuem vínculo nas duas tabelas (PK e a sua FK)
- O OUTER JOIN pode haver valores nulos
 - Junção a esquerda – LEFT JOIN;
 - Junção a direita – RIGHT JOIN;
 - Junção completa – FULL JOIN;
 - MySQL não tem – mas há como fazer de outras formas.

LEFT JOIN

- Quando dizemos que a tabela será LEFT, estamos solicitando *TODOS REGISTROS* da tabela da *ESQUERDA*;
- O banco de dados irá trazer todos registros da tabela da esquerda e verificar se possui o registro que atende a condição de junção da tabela da direita;
- O LEFT JOIN pode haver valores nulos nas colunas da tabela da direita;

SINTAXE

```
SELECT coluna1, coluna2, . . . , coluna_n  
FROM tabela1  
LEFT JOIN tabela2 ON condição_junção  
[WHERE outras condições];
```

LEFT JOIN

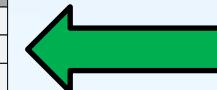
NÃO POSSUI FK, porque o registro antigo efetuado antes de implementar o sistema

CLIENTE		
ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA		
ID	DATA	CLIID
1	10/01/1985	
2	07/02/2013	1
3	23/03/2013	3
4	23/03/2013	3

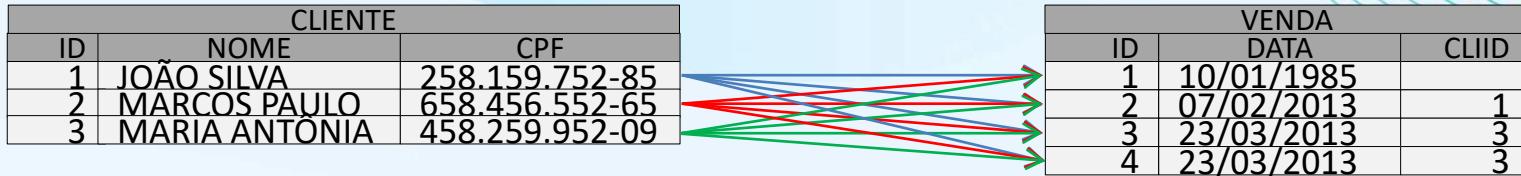
`SELECT * FROM cliente LEFT JOIN venda ON cliente.id = venda.cliid`

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65			null
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	4	23/03/2013	3



Não possui vínculo...
Ele vai para o resultado?

LEFT JOIN



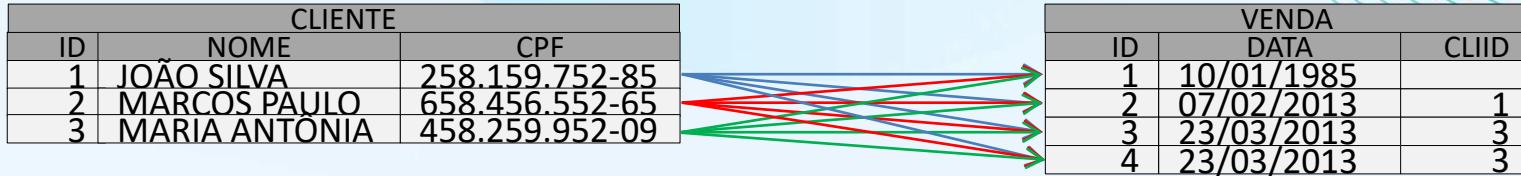
```
SELECT * FROM cliente LEFT JOIN venda ON cliente.id = venda.cliid
```

The result table shows the joined data from both tables. The 'CLIENTE' table rows are listed, and for each row, there is a corresponding row in the 'VENDA' table if it exists. A green arrow points from the result table back to the original 'CLIENTE' and 'VENDA' tables.

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65			null
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	4	23/03/2013	3

Em LEFT JOIN, quando a tupla da tabela da ESQUERDA não possui “UM PAR” vai para o resultado do mesmo jeito

LEFT JOIN



```
SELECT * FROM cliente LEFT JOIN venda ON cli.id = venda.cliid
```

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65			null
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	4	23/03/2013	3

TODOS OS CLIENTES
estão no resultado

A venda que não possui
vínculo não vai para o
resultado

LEFT JOIN

SELECT *

FROM cliente c

LEFT JOIN venda v **ON** v.cliid = c.id;

RIGHT JOIN

- Quando dizemos que a tabela será RIGHT, estamos solicitando todos os registros da tabela da *DIREITA*;
 - Os registros da tabela da esquerda irão para o resultado se atender a condição de junção.
- Nesta junção, poderá haver resultados NULOS nas colunas da tabela da esquerda.

SINTAXE

```
SELECT coluna1, coluna2, ..., coluna_n  
FROM tabela1  
RIGHT JOIN tabela2 ON condição_junção  
[WHERE outras condições];
```

RIGHT JOIN

CLIENTE		
ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA		
ID	DATA	CLIID
1	10/01/1985	
2	07/02/2013	1
3	23/03/2013	3
4	23/03/2013	3

```
SELECT * FROM cliente  
RIGHT JOIN venda ON cli.id = venda.cliid
```



CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
null			1	10/01/2013	
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	4	23/03/2013	3

Não possui vínculo...
Ele vai para o resultado?

RIGHT JOIN

CLIENTE		
ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA		
ID	DATA	CLIID
1	10/01/1985	
2	07/02/2013	1
3	23/03/2013	3
4	23/03/2013	3

```
SELECT * FROM cliente  
RIGHT JOIN venda ON cli.id = venda.cliid
```



CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
null			1	10/01/2013	
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	4	23/03/2013	3

Em RIGHT JOIN, quando as colunas da tabela da DIREITA não possui “UM PAR” vai para o resultado do mesmo jeito

RIGHT JOIN

CLIENTE		
ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA		
ID	DATA	CLIID
1	10/01/1985	
2	07/02/2013	1
3	23/03/2013	3
4	23/03/2013	3

```
SELECT * FROM cliente  
RIGHT JOIN venda ON cli.id = venda.cliid
```



CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
null			1	10/01/2013	
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	4	23/03/2013	3

SÓ VAI PARA O
RESULTADO OS CLIENTES
QUE POSSUEM UM PAR

TODAS AS VENDA IRÃO
PARA O RESULTADO

RIGHT JOIN

```
SELECT *  
FROM cliente c  
RIGHT JOIN venda v ON v.cliid = c.id;
```

OUTER JOIN

CLIENTE		
ID	NOME	CPF
1	JOÃO SILVA	258.159.752-85
2	MARCOS PAULO	658.456.552-65
3	MARIA ANTÔNIA	458.259.952-09

VENDA		
ID	DATA	CLIID
1	10/01/1985	
2	07/02/2013	1
3	23/03/2013	3
4	23/03/2013	3

SELECT * FROM CLIENTE C

RIGHT JOIN VENDA V ON C.ID = V.CLIENTE_ID;

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
null			1	10/01/2013	
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3

SELECT * FROM CLIENTE C

LEFT JOIN VENDA V ON C.ID = V.CLIENTE_ID;

CLIENTE			VENDA		
ID	NOME	CPF	ID	DATA	CLIID
1	JOÃO SILVA	258.159.752-85	1	10/01/2013	1
2	MARCOS PAULO	658.456.552-65			null
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3
3	MARIA ANTÔNIA	458.259.952-09	3	23/03/2013	3