

# ***Structured Query Language (SQL)*** **Parte 6**

## **Consultas envolvendo Junções**

Kelly Rosa Braghetto  
[kellyrb@ime.usp.br](mailto:kellyrb@ime.usp.br)

**Departamento de Ciência da Computação**

# Consultas mais avançadas



Este material é uma adaptação dos *slides*  
do prof. Jeffrey Ullman,  
da *Stanford University*

<http://infolab.stanford.edu/~ullman/dscb/gslides.html>

# Exemplo para a aula



- As consultas SQL serão baseadas no seguinte esquema relacional de BD:

**Refrigerante(nome, fabricante)**

**Lanchonete(nome, endereco, cnpj)**

**Cliente(nome, endereco, telefone)**

**Apreciador(nome cliente, nome refri)**

**Vendedor(nome lanch, nome refri, preco)**

**Frequentador(nome cliente, nome lanch)**

# Expressões de Junção (JOIN)



- SQL possui várias versões de junções.
- Os operadores de junção (JOIN) só foram introduzidos no padrão SQL-92.
- Padrões anteriores da SQL (SQL-89 e SQL-86) não possuíam operadores explícitos para a junção.
- Mas é sempre possível obter o mesmo efeito deles por meio de uma consulta do tipo SELECT-FROM-WHERE.
- As expressões JOIN podem ser consultas “por si só” ou podem ser usadas no lugar de relações em uma cláusula FROM.

# Produto Cartesiano



- É o tipo de junção mais simples:

## **R CROSS JOIN S;**

- Combina todas as tuplas de R com todas as tuplas de S.
- As relações envolvidas no produto cartesiano também podem ser subconsultas parentizadas.
  - Isso vale para todos os tipos de JOIN.
- O produto cartesiano sozinho raramente é útil.

# Junção Teta (INNER JOIN)



- Forma da junção teta:

**R INNER JOIN S ON <condição de junção>**

- Equivale a um produto cartesiano de R por S, seguido de uma seleção de tuplas
- A **<condição de junção>** é usada para a seleção das tuplas
  - Todo par de tuplas de R X S que satisfaz a condição de junção é selecionado
- A palavra-chave “INNER” não precisa ser escrita
  - Ele é o tipo padrão de JOIN

# Exemplo: Junção Teta



- Usando

Cliente(nome, endereço) e

Frequentador(nome\_cliente, nome\_lanch)

queremos encontrar todas triplas **(c, e, l)** tais que o cliente **c** mora no endereço **e** em São Paulo e frequenta a lanchonete **l**.

```
SELECT nome, endereço e nome_lanch  
FROM Cliente JOIN Frequentador  
ON nome = nome_cliente  
WHERE endereço like '%São Paulo%';
```

# Junção Natural (NATURAL JOIN)



- Forma da junção natural:

## **R NATURAL JOIN S**

- Equivale a uma junção teta, mas tem uma condição de junção pré-definida.
- A condição de junção é sempre a igualdade sobre os pares de atributos das duas relações que possuem o mesmo nome.
- Elimina da resposta as duplicações de colunas.



# Exemplo: Junção Natural



- Usando

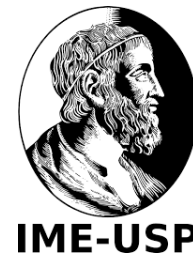
Vendedor(nome\_lanch, nome\_refri) e  
Frequentador(nome\_cliente, nome\_lanch),

Queremos encontrar todas as triplas  $(l, r, c)$  tais que a lanchonete  $l$  vende o refrigerante  $r$  e é frequentada pelo cliente  $c$ .

```
SELECT *  
FROM Vendedor NATURAL JOIN Frequentador;
```

Obs.: A coluna nome\_lanch aparece uma só vez na resposta.

# Exemplo: Junção Natural



- Usando

Vendedor(**nome\_lanch**, nome\_refri) e  
Frequentador(nome\_cliente, **nome\_lanch**),

Queremos encontrar os nomes dos frequentadores das lanchonetes que vendem 'Fanfa'.

```
SELECT nome_cliente  
FROM Frequentador NATURAL JOIN  
    (SELECT * FROM Vendedor  
     WHERE nome_refri = 'Fanfa') as V;
```

# Junção Externa (OUTER JOIN)

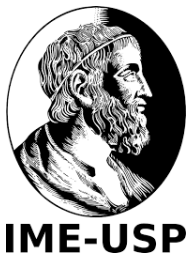


- Forma da junção externa:

**R OUTER JOIN S ON <condição de junção>**

- Se difere dos tipos anteriores de junção por incluir na resposta tuplas soltas.
- Uma tupla solta é uma tupla que não “casa” com nenhuma outra da outra relação da junção.

# Junção Externa (OUTER JOIN)



- ◆ **R OUTER JOIN S** é o núcleo de uma expressão de junção externa. Ela pode conter também as seguintes cláusulas:

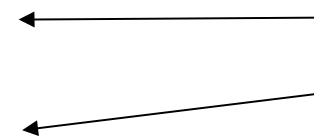
## 1) **LEFT**, **RIGHT**, ou **FULL** antes de **OUTER**

- **LEFT** = inclui apenas as tuplas soltas de R
- **RIGHT** = inclui apenas as tuplas soltas de S
- **FULL** = inclui as tuplas soltas de ambas; esta é a opção padrão

## 2) **NATURAL** antes de tudo **ou**

**ON** <condição> depois de **JOIN**

- O palavra-chave “**OUTER**” é opcional na expressão.



Uma (e apenas uma) entre essas duas opções precisa aparecer no comando

# Exemplo: Junção Externa



- Usando `Cliente(nome, endereço)`  
`Frequentador(nome_cliente, nome_lanch)`

```
SELECT  C.*, F.nome_lanch  
FROM Cliente as C LEFT OUTER JOIN Frequentador as F  
ON nome = nome_cliente;
```

Essa consulta nos dá uma lista de tuplas, onde cada tupla associa os dados de um cliente ao nome de uma lanchonete que ele frequenta.

Se um cliente não frequenta nenhuma lanchonete, seus dados aparecerão na lista associados ao valor NULL (para o nome de lanchonete).

## Exemplo:

select \* from R left outer join S  
on (B=D and C=E);



IME-USP

A	B	C
1	2	3
4	5	6
7	8	9

Relação *R*

D	E	F
2	3	10
2	3	11
6	7	12

Relação *S*

A	B	C	D	E	F
1	2	3	2	3	10
1	2	3	2	3	11
4	5	6	NULL	NULL	NULL
7	8	9	NULL	NULL	NULL

Resultado de  $R \bowtie_{B=D, C=E} S$

## Exemplo:

select \* from R full outer join S  
on (B=D and C=E);



IME-USP

A	B	C
1	2	3
4	5	6
7	8	9

Relação *R*

D	E	F
2	3	10
2	3	11
6	7	12

Relação *S*

A	B	C	D	E	F
1	2	3	2	3	10
1	2	3	2	3	11
4	5	6	NULL	NULL	NULL
7	8	9	NULL	NULL	NULL
NULL	NULL	NULL	6	7	12

Resultado de  $R \bowtie_{B=D, C=E} S$

## Exemplo:

select \* from R right outer join S  
on (B=D and C=E);



IME-USP

A	B	C
1	2	3
4	5	6
7	8	9

Relação *R*

D	E	F
2	3	10
2	3	11
6	7	12

Relação *S*

A	B	C	D	E	F
1	2	3	2	3	10
1	2	3	2	3	11
NULL	NULL	NULL	6	7	12

Resultado de  $R \bowtie_{B=D, C=E} S$



## Exemplo:

select \* from R full outer join S  
on (B=D and C=E);



IME-USP

A	B	C
1	2	3
4	5	6
7	8	9

Relação *R*

D	E	F
2	3	10
2	3	11
6	7	12

Relação *S*

A	B	C	D	E	F
1	2	3	2	3	10
1	2	3	2	3	11
4	5	6	NULL	NULL	NULL
7	8	9	NULL	NULL	NULL
NULL	NULL	NULL	6	7	12

Resultado de  $R \bowtie_{B=D, C=E} S$

## Exemplo:

select \* from R natural  
left outer join S;



A	B	C
1	2	3
4	5	6
7	8	9

Relação *R*

B	C	D
2	3	10
2	3	11
6	7	12

Relação *S*

A	B	C	D
1	2	3	10
1	2	3	11
4	5	6	NULL
7	8	9	NULL

Resultado da junção natural externa à esquerda

## Exemplo:

`select * from R natural  
right outer join S;`



IME-USP

A	B	C
1	2	3
4	5	6
7	8	9

Relação *R*

B	C	D
2	3	10
2	3	11
6	7	12

Relação *S*

A	B	C	D
1	2	3	10
1	2	3	11
NULL	6	7	12

Resultado da junção natural externa à direita

## Exemplo:

select \* from R natural full  
outer join S;



IME-USP

A	B	C
1	2	3
4	5	6
7	8	9

Relação *R*

B	C	D
2	3	10
2	3	11
6	7	12

Relação *S*

A	B	C	D
1	2	3	10
1	2	3	11
4	5	6	NULL
7	8	9	NULL
NULL	6	7	12

Resultado da junção natural externa completa

# Referências bibliográficas



- *Database Systems – The Complete Book*, Garcia-Molina, Ullman e Widom. 2002.  
Capítulo 6
- *Sistemas de Bancos de Dados* (6ª edição), Elmasri e Navathe. 2010.  
Capítulo 5