

Banco de Dados

Jordana S. Salamon

jssalamon@inf.ufes.br

jordanasalamon@gmail.com

DEPARTAMENTO DE INFORMÁTICA
CENTRO TECNOLÓGICO
UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO

Introdução

- ▶ **Modelo de dados:** conceitos para a definição das restrições e estrutura do banco de dados + conjunto de operações para manipular banco de dados
- ▶ **Conjunto básico de operações:** **Álgebra relacional**
 - ▶ Permitem aos usuários especificar solicitações ao esquema de banco de dados, cujo resultado será uma nova relação
 - ▶ Uma sequencia de operações de álgebra relacional forma uma expressão de álgebra relacional

Introdução

- ▶ Álgebra relacional e cálculo relacional: Duas linguagens de consulta formais do modelo relacional
- ▶ Estas linguagens de Consulta são concisas e formais
- ▶ Entender álgebra e cálculo relacional é **fundamental** para compreender SQL

Álgebra Relacional

- ▶ Proposta por Codd juntamente com o Modelo Relacional.
- ▶ As relações são consideradas conjuntos sobre os quais são executadas as operações.
- ▶ Todas as operações aplicáveis sobre conjuntos são aplicáveis sobre relações.
- ▶ Codd definiu originalmente 8 operadores:
 - ▶ *Operações de Conjuntos Tradicionais:*
 - ▶ União, Interseção, Diferença e Produto Cartesiano.
 - ▶ *Operações Relacionais Especiais:*
 - ▶ Seleção, Projeção, Junção e Divisão.

Importância da Álgebra Relacional

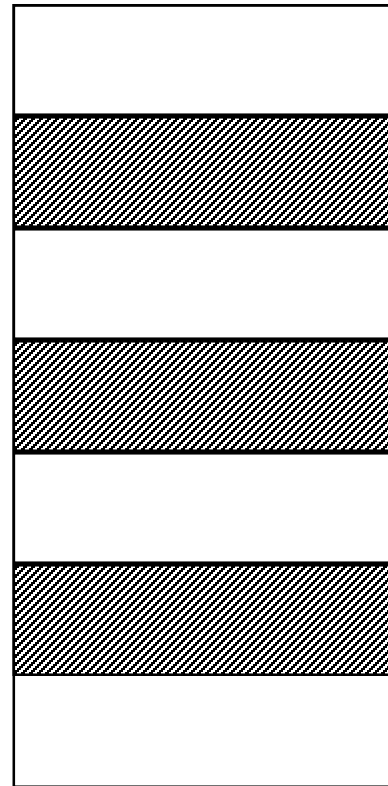
- ▶ Fornece um fundamento formal para operações do modelo relacional
- ▶ É usada como base para implementar consultas em SGBDs relacionais
- ▶ Algumas consultas são incorporadas pela linguagem SQL

Álgebra Relacional

- Linguagem procedural
- Operadores básicos
 - Seleção (select): σ
 - Projeção (project): π
 - União (union): \cup
 - Interserção: \cap
 - Diferença de conjunto (set difference): $-$
 - Produto cartesiano (Cartesian product) \times
 - Junção (Join) \bowtie
 - Renomear (rename): ρ
- Utiliza uma ou duas relações como entrada e produz uma nova relação como resultado.

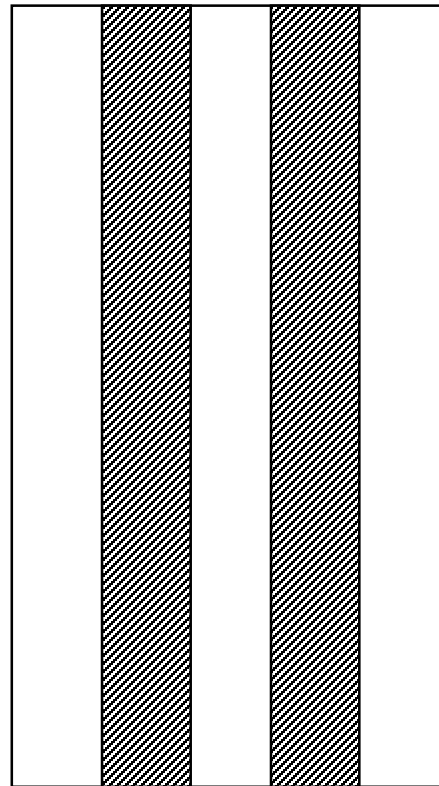
Álgebra Relacional

- ▶ **Seleção:** Seleciona tuplas de uma relação que atendem a uma certa condição de pesquisa.



Álgebra Relacional

- ▶ **Projeção:** Extrai atributos específicos de uma relação.



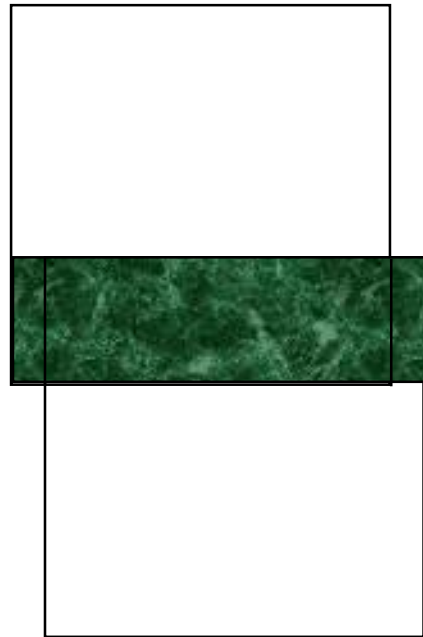
Álgebra Relacional

- ▶ **União:** Constrói uma relação consistindo em todas as tuplas que aparecem em pelo menos uma das duas relações.



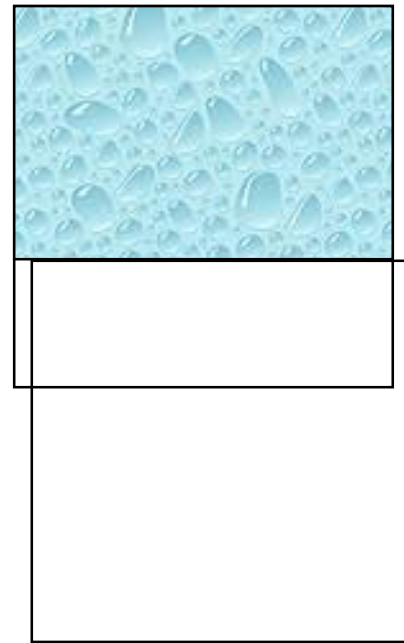
Álgebra Relacional

- ▶ **Interseção:** Constrói uma relação consistindo em todas as tuplas que aparecem simultaneamente nas duas relações.



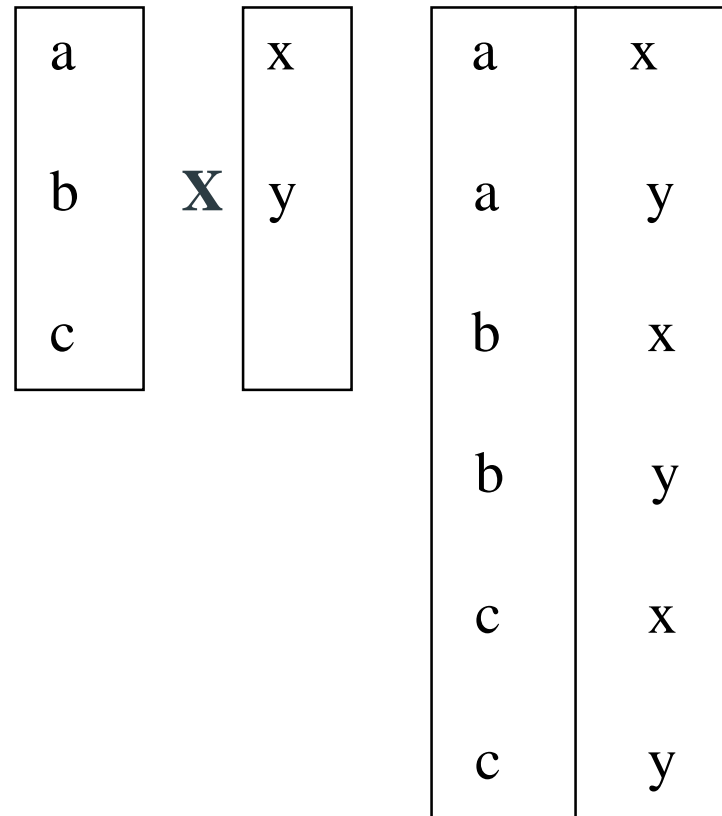
Álgebra Relacional

- **Diferença:** Constrói uma relação consistindo em todas as tuplas que aparecem somente na primeira relação (mas não na segunda).



Álgebra Relacional

- ▶ **Produto Cartesiano:** Concatena cada uma das tuplas de uma relação com todas as outras tuplas da outra relação.



Álgebra Relacional

- **Junção:** Constrói uma nova relação a partir de duas outras, concatenando tuplas das duas relações, com base na igualdade de valores de um atributo em comum.

R1	R2	Junção de R1 e R2																					
<table border="1"><tr><td>a 1</td><td>b1</td></tr><tr><td>a 2</td><td>b1</td></tr><tr><td>a 3</td><td>b2</td></tr></table>	a 1	b1	a 2	b1	a 3	b2	<table border="1"><tr><td>b1</td><td>c1</td></tr><tr><td>b2</td><td>c2</td></tr><tr><td>b3</td><td>c3</td></tr></table>	b1	c1	b2	c2	b3	c3	<table border="1"><tr><td>a1</td><td>b1</td><td>c1</td></tr><tr><td>a2</td><td>b1</td><td>c1</td></tr><tr><td>a3</td><td>b2</td><td>c2</td></tr></table>	a1	b1	c1	a2	b1	c1	a3	b2	c2
a 1	b1																						
a 2	b1																						
a 3	b2																						
b1	c1																						
b2	c2																						
b3	c3																						
a1	b1	c1																					
a2	b1	c1																					
a3	b2	c2																					

Álgebra Relacional

- **Divisão:** Constrói uma nova relação a partir de duas outras, considerando apenas uma tupla de um conjunto de tuplas da primeira relação, tal que, neste conjunto, as tuplas contenham TODOS os valores do atributo de divisão da segunda relação.

a	x	x	a
a	y	z	
a	z		
b	x		
c	y		

Banco de dados: Exemplo

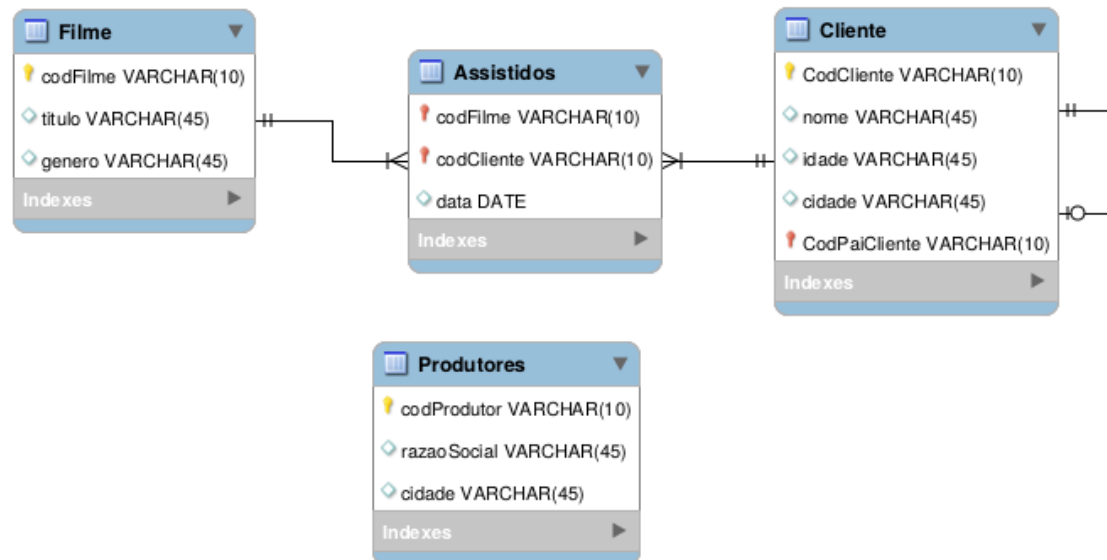
- Um sistema para distribuição online de filmes

Filmes(codFilme, titulo, gênero)

Clientes(codCli, nome, idade, cidade, codPaiCli)

Assistidos(codFilme, codCli, data)

Produtores(codProdutor, razaoSocial, cidade)



Banco de Dados Utilizado nos Exemplos

Relação Filmes

codFilme	título	gênero
f1	Coração Valente	Aventura
f2	Mont Python	Comédia
f3	A lagoa azul	Romance
f4	Se beber, não case	Comédia

Relação Clientes

codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2
c2	José	28	Vitória	c4
c3	Luís	25	Belo Horizonte	c2
c4	Sílvio	43	Vitória	

Banco de Dados Utilizado nos Exemplos

Tabela de Assistidos

codFilme	codCli	data
f1	c1	01/01/2018
f2	c3	05/01/2018
f3	c1	01/02/2018
f2	c2	03/01/2018

Tabela de Produtores

codProdutor	razaoSocial	Cidade
r1	Sol&Mar Ltda.	Rio de Janeiro
r2	Mulptel	São Paulo
r3	Oficina das Máquinas	Vitória
r4	Brinquedos & Cia. Ltda.	Belo Horizonte

Operação de Seleção

- **Operador de seleção (σ):** seleciona linhas de uma tabela. A seleção é feita através da avaliação de uma condição de seleção.
- **Sintaxe:** σ <critério de seleção>(<relação>)
- **Exemplos:**
 σ (Filmes)

Resultado:

codFilme	título	gênero
f1	Coração Valente	Aventura
f2	Mont Python	Comédia
f3	A lagoa azul	Romance
f4	Se beber, não case	Comédia

Operação de Seleção

- **Operador de seleção (σ):** seleciona linhas de uma tabela. A seleção é feita através da avaliação de uma condição de seleção.
- **Sintaxe:** σ <critério de seleção>(<relação>)
- Pode-se usar comparações: =, \neq , <, \leq , >, \geq
- Pode-se concatenar critérios usando os conectivos *e* (\wedge) e *ou* (\vee)
- **Exemplos:**
 σ gênero='Comédia' (Filmes)

Resultado:

codFilme	título	gênero
f1	Coração Valente	Aventura
f2	Mont Python	Comédia
f3	A lagoa azul	Romance
f4	Se beber, não case	Comédia



Relação resultante

codFilme	título	gênero
f2	Mont Python	Comédia
f4	Se beber, não case	Comédia

Operação de Seleção

- **Exemplos:**

Selecione todos os clientes que moram em Vitória com mais de 30 anos

codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2
c2	José	28	Vitória	c4
c3	Luís	25	Belo Horizonte	c2
c4	Sílvio	43	Vitória	

Operação de Seleção

- **Exemplos:**

Selecione todos os clientes que moram em Vitória com mais de 30 anos

σ (idade > 30 \wedge cidade = 'Vitória') (Clientes)

Resultado:

codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2
c4	Sílvio	43	Vitória	

Operação de Projeção

- Operador de projeção (π): projeta colunas no resultado.
- Sintaxe: π <lista de colunas>(<tabela>)
- Exemplos:
 π gênero (Filmes)

Resultado:

gênero
Aventura
Comédia
Romance

Obs.: é importante ressaltar que na projeção, as linhas duplicadas são eliminadas, ou seja, aparecem apenas uma vez no resultado. Isso ocorre porque uma Tabela é um *conjunto* de linhas e conjuntos não podem conter elementos repetidos.



Operação de Projeção

- **Exemplos:**

π codFilme, título (σ gênero='Comédia' (Filmes))

Resultado:

A expressão mais interna é a primeira a ser efetuada

codFilme	título
f2	Mont Python
f4	Se beber, não case

Operação de Renomeação

- **Operador de renomeação (ρ):** permite dar um novo nome a uma tabela.
- **Sintaxe:** ρ <novo nome>(<tabela>)
- **Exemplos:**
 ρ FilmesRomance (π codFilme, título (σ gênero='Romance' (Filmes)))

Resultado:

FilmesRomance

codFilme	título
f3	A lagoa azul

Operação de União

- **Operador de união (\cup):** a união de duas tabelas produz como resultado uma terceira tabela que contém todas as tuplas da primeira e da segunda.
- **Sintaxe:** $\langle \text{tabela1} \rangle \cup \langle \text{tabela2} \rangle$
- **Exemplos:**
 $(\pi \text{ cidade (Produtores)}) \cup (\pi \text{ cidade (Clientes)})$

Resultado:

cidade
Rio de Janeiro
São Paulo
Vitória
Belo Horizonte

Operação de Produto Cartesiano

- **Operador de produto cartesiano (x):**

o produto cartesiano de duas tabelas produz como resultado uma tabela que contém todos os campos da primeira tabela, seguidos de todos os campos da segunda tabela. Combina as linhas da primeira tabela com cada linha da segunda.

- A tabela resultado possui:
- *(número de colunas da primeira tabela + número de colunas da segunda) colunas; e (número de linhas da primeira tabela x número de linhas da segunda) linhas.*
- **Sintaxe: <tabela1> x <tabela2>**

Operação de Produto Cartesiano

- **Exemplo:** Filmes x Assistidos

codFilme	título	gênero	codFilme	codCli	data
f1	Coração Valente	Aventura	f1	c1	01/01/2018
f1	Coração Valente	Aventura	f2	c3	05/01/2018
f1	Coração Valente	Aventura	f3	c1	01/02/2018
f1	Coração Valente	Aventura	f2	c2	03/01/2018
f2	Mont Python	Comédia	f1	c1	01/01/2018
f2	Mont Python	Comédia	f2	c3	05/01/2018
f2	Mont Python	Comédia	f3	c1	01/02/2018
f2	Mont Python	Comédia	f2	c2	03/01/2018
...					

Operação de Produto Cartesiano

- Exemplos:

Tabela que contém os nomes de clientes e de seus pais.

π Clientes.nome, PaiClientes.nome (σ Clientes.codPaiCli = PaiClientes.codCli (Clientes \times ρ PaiClientes (Clientes)))

codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2
c2	José	28	Vitória	c4
c3	Luís	25	Belo Horizont e	c2
c4	Sílvio	43	Vitória	

→ PaiClientes

codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2
c2	José	28	Vitória	c4
c3	Luís	25	Belo Horizont e	c2
c4	Sílvio	43	Vitória	

Operação de Produto Cartesiano

- **Exemplos:**

Tabela que contém os nomes de clientes e de seus pais.

π Clientes.nome, PaiClientes.nome (σ Clientes.codPaiCli = PaiClientes.codCli
(Clientes x ρ PaiClientes (Clientes)))

codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2
c2	José	28	Vitória	c4
c3	Luís	25	Belo Horizont e	c2
c4	Sílvio	43	Vitória	

PaiClientes

codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2
c2	José	28	Vitória	c4
c3	Luís	25	Belo Horizont e	c2
c4	Sílvio	43	Vitória	

Operação de Produto Cartesiano

- Exemplos:

Tabela que contém os nomes de clientes e de seus pais.

π Clientes.nome, PaiClientes.nome (σ Clientes.codPaiCli = PaiClientes.codCli (Clientes x ρ PaiClientes (Clientes)))

PaiClientes

codCli	nome	idade	cidade	codPaiCli	codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2	c2	José	28	Vitória	c4
c2	José	28	Vitória	c4	c4	Sílvio	43	Vitória	
c3	Luís	25	Belo Horizonte	c2	c2	José	28	Vitória	c4

Operação de Produto Cartesiano

- Exemplos:**

Tabela que contém os nomes de clientes e de seus pais.

π Clientes.nome, PaiClientes.nome (σ Clientes.codPaiCli = PaiClientes.codCli (Clientes x ρ PaiClientes (Clientes)))

PaiClientes

codCli	nome	idade	cidade	codPaiCli	codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2	c2	José	28	Vitória	c4
c2	José	28	Vitória	c4	c4	Sílvio	43	Vitória	
c3	Luís	25	Belo Horizont e	c2	c2	José	28	Vitória	c4

Operação de Produto Cartesiano

- Exemplos:**

Tabela que contém os nomes de clientes e de seus pais.

π Clientes.nome, PaiClientes.nome (σ Clientes.codPaiCli = PaiClientes.codCli (Clientes x ρ PaiClientes (Clientes)))

PaiClientes

codCli	nome	idade	cidade	codPaiCli	codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2	c2	José	28	Vitória	c4
c2	José	28	Vitória	c4	c4	Sílvio	43	Vitória	
c3	Luís	25	Belo Horizont e	c2	c2	José	28	Vitória	c4



nome	nome
João	José
José	Sílvio
Luís	José

Operação de Junção

- ▶ **Operador de junção (\bowtie):** uma junção é uma operação de seleção aplicada a uma operação de produto cartesiano.

Como a junção é uma operação muito comum, foi criado um operador para facilitar a sequência de operações necessárias para a realização de uma consulta.

- ▶ **Sintaxe:** <tabela 1> \bowtie (<critério de seleção>) <tabela2>

Observação: Quando o critério de seleção é apenas a comparação entre a chave primária da tabela 1 com a chave estrangeira da tabela 2 a mesma pode ser omitida

Operação de Junção

- Exemplos:**

Tabela que contém os nomes de clientes e de seus pais.

Utilizando produto cartesiano

$(\sigma \text{ Clientes.codPaiCli} = \text{PaiClientes.codCli} (\text{Clientes} \times (\text{Clientes})))$

$\text{Clientes} \bowtie (\text{Clientes.codCli} = \text{Clientes.codPaiCli}) \text{ Clientes}$

codCli	nome	idade	cidade	codPaiCli	codCli	nome	idade	cidade	codPaiCli
c1	João	31	Vitória	c2	c2	José	28	Vitória	c4
c2	José	28	Vitória	c4	c4	Sílvio	43	Vitória	
c3	Luís	25	Belo Horizont e	c2	c2	José	28	Vitória	c4

Operação de Junção

► Exemplos:

Assistidos ⋈ (Assistidos.codCli = Clientes.codCli) Clientes

Resultado: Tabela com os filmes assistidos pelos respectivos clientes.

codFilme	codCli	data	codCli	nome	idade	cidade	codPaiCli
f1	c1	01/01/2000	c1	João	31	Porto Alegre	c2
f2	c3	05/01/2000	c3	Luís	25	Novo Hamburgo	c2
f3	c1	01/02/2000	c1	João	31	Porto Alegre	c2
f2	c2	03/01/2000	c2	José	28	Porto Alegre	c4

Operação de Junção

► Exercício:

Obter o nome dos filmes assistidos pelos clientes

Para facilitar considere o exemplo anterior

$\rho(\text{filmesassistidos})_{\text{Assistidos}} \bowtie (\text{Assistidos.codCli} = \text{Clientes.codCli}) \text{ Clientes}$

codFilme	codCli	data	codCli	nome	idade	cidade	codPaiCli
f1	c1	01/01/2000	c1	João	31	Porto Alegre	c2
f2	c3	05/01/2000	c3	Luís	25	Novo Hamburgo	c2
f3	c1	01/02/2000	c1	João	31	Porto Alegre	c2
f2	c2	03/01/2000	c2	José	28	Porto Alegre	c4

Operação de Junção

- ▶ A condição de junção pode utilizar outros operadores, como $<$, $>$, \neq etc.
- ▶ A operação de junção genérica (que envolve qualquer critério de seleção) é chamada de Junção Theta.
- ▶ Quando o critério de seleção envolve apenas uma igualdade de atributos, a operação de junção é chamada de *Equijunção*.

Operação de Junção

- ▶ Quando os campos na *Equijunção* tiverem os mesmos nomes (como no exemplo anterior), os nomes das colunas podem ser omitidos.
- ▶ Neste caso, a junção é chamada de *Junção ou Ligação natural*.

▶ **Junção natural - Sintaxe:**

<tabela1> ⋈ <tabela2>

Exemplo: **Assistidos** ⋈ **Clientes** - mesmo resultado do exemplo anterior.

Operação de Junção

► Considerações:

- A operação de junção concatena duas linhas das tabelas que estão sendo juntadas com base no critério de seleção (normalmente por igualdade de valores de atributos).
- Uma linha que não possua nenhuma linha correspondente na outra tabela associada pelo critério de junção não aparece na tabela resultado.
- Há situações em que é necessário garantir que todas as linhas de uma das tabelas (ou de ambas) apareçam no resultado.

Operação de Junção Externa (Outer Join)

- ▶ **Operador de junção externa (\bowtie):** junção natural na qual as tuplas de uma ou ambas as relações que não são combinadas são mesmo assim preservadas no resultado.
- ▶ Podem ser:
- ▶ **Junção externa à esquerda (left outer join)**
- ▶ $\langle \text{tabela1} \rangle \bowtie \langle \text{critério} \rangle \langle \text{tabela2} \rangle$

Tuplas da relação à esquerda são preservadas.

- ▶ **Junção externa à direita (right outer join)**
- ▶ $\langle \text{tabela1} \rangle \bowtie \langle \text{critério} \rangle \langle \text{tabela2} \rangle$

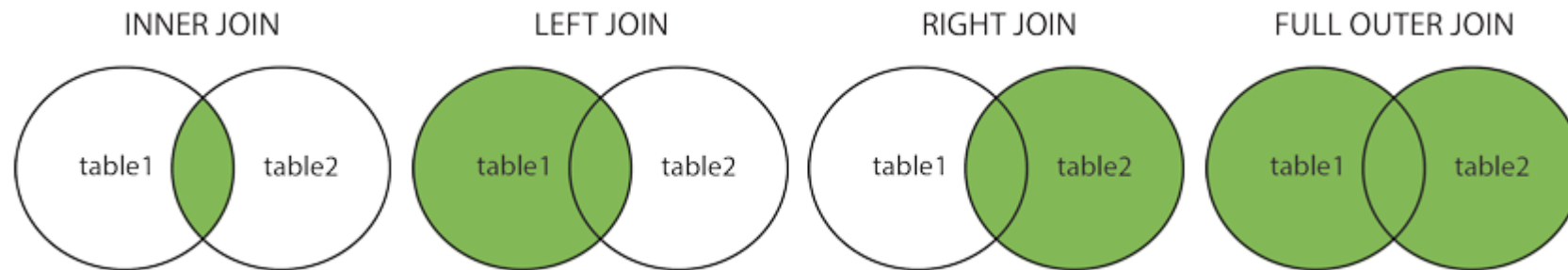
Tuplas da relação à direita são preservadas.

Operação de Junção Externa (Outer Join)

Junção externa completa (full outer join)

► $\langle \text{tabela1} \rangle \bowtie (\langle \text{critério} \rangle) \langle \text{tabela2} \rangle$

Tuplas de ambas as relações são preservadas.



Operação de Junção

▶ Exemplos:

Clientes \bowtie (Clientes.codCli = Assistidos.codCli) Assistidos

codCli	nome	idade	cidade	codPai Cli	codFilme	codCli	data
c1	João	31	Porto Alegre	c2	f1	c1	01/01/2000
c1	João	31	Porto Alegre	c2	f3	c1	01/02/2000
c2	José	28	Porto Alegre	c4	f2	c2	03/01/2000
c3	Luís	25	Novo Hamburgo	c2	f2	c3	05/01/2000
c4	Sílvio	43	Porto Alegre				

Operação de Interseção

- **Operador de interseção (\cap):** a interseção de duas tabelas produz como resultado uma tabela que contém as tuplas presentes na primeira e na segunda tabela ao mesmo tempo.
- **Sintaxe:** $\langle \text{tabela1} \rangle \cap \langle \text{tabela2} \rangle$
- **Exemplos:**
 $(\pi \text{ cidade (Produtores)}) \cap (\pi \text{ cidade (Clientes)})$

Resultado:

cidade
Vitória

Operação de Interseção

- **Exemplos:**

$(\pi \text{ codFilme (Filmes)}) \cap (\pi \text{ codFilme (Assistidos)})$

Resultado:

Filmes que já foram assistidos.

codFilme
f1
f2
f3

Operação de Diferença

- **Operador de diferença (-):** a diferença de duas tabelas produz como resultado uma tabela que contém as tuplas presentes na primeira que não constam da segunda.

- **Sintaxe:** <tabela1> - <tabela2>

- **Exemplos:**

$(\pi \text{ codFilme (Filmes)}) - (\pi \text{ codFilme (Assistidos)})$

Resultado:

Filmes que nunca foram assistidos.

codFilme
f4

Operações de União, Interseção e Diferença

- *Para as operações de União, Interseção e Diferença, os operadores (i.e. as tabelas) devem ser compatíveis.*
- *Para a União, o que significa:*
 - Devem conter o mesmo número de campos;
 - O domínio da i -ésima coluna da primeira tabela deve ser igual ao domínio da i -ésima coluna da segunda tabela;
 - Se os nomes das colunas forem diferentes, utiliza-se por convenção os nomes das colunas da primeira tabela.

Operação de Divisão

- **Operador de divisão (\div):** produz como resultado as tuplas da tabela 1 para as quais existe uma correspondência na tabela 2. Porém, as colunas resultantes são aquelas que aparecem na tabela 1, mas não aparecem na tabela 2.
- **Sintaxe:** $\langle \text{tabela1} \rangle \div \langle \text{tabela2} \rangle$

Operação de Divisão

- Exemplo: $(\pi \text{ codFilme, codCli (Assistidos)}) \div (\pi \text{ codFilme (Filmes)})$
Clientes que já assistiram todos os filmes.

codFilme	codCli
f1	c1
f2	c1
f3	c1
f4	c1
f2	c3
f3	c1
f2	c2

÷

codFilme
f1
f2
f3
f4

=

codCli
c1

Álgebra Relacional

- ▶ Para melhor combinação e eficiência no uso dos operadores da Álgebra Relacional, é importante:
 - ▶ Antes de juntar duas relações A e B, tentar diminuir as suas cardinalidades (número de tuplas) via seleção e grau (número de atributos) via projeção.
 - ▶ Quanto menor o número de tuplas e de menor tamanho, mais eficiente será a junção.



Exercícios

- ▶ Dadas as Relações abaixo, responda as perguntas em Álgebra Relacional :

Relações:

FUNCIONÁRIO (Matr, Nome)

TEM (MatrFunc, CodLivro)

LIVRO (Código, Título, Editora, Ano)

Exercícios

- a) Quais os nomes de dos funcionários que tem livros editados em 1987 pela editora Campus ?
- b) Quais os nomes dos funcionários que tem todos os livros da editora Campus ?
- c) Quais os Títulos dos livros da Funcionária “CAROLINA”?
- d) Quais os Anos dos Livros da editora Campus ?
- e) Quais os Funcionários que possuem livros do ano de 1995 ?

Exercícios:

- ▶ Dadas as Relações abaixo, responda as perguntas em Álgebra Relacional :

Relações:

Cliente (Cli, NomeCli, EstadoCli)

Fabricante (Fab, NomeFab, EstadoFab)

Peça (Pe, NomePe, Cor, Valor)

Compra (Cli, Pe, Data, Quantidade)

Fabrica (Fab, Pe)



Exercícios

1. Quais os nomes dos fabricantes que fornecem peças azuis ?
2. Quais os nomes dos fabricantes que fabricam peças verdes de mais de R\$ 500,00 compradas por clientes de São Paulo ?
3. De que Estados são os clientes que não compraram nenhuma peça ?
4. Quais são os nomes dos clientes que compram peças fabricadas por fabricantes de outro estado (que não o do cliente) ?
5. Em que Estado não são fabricadas peças ?
6. Quais os códigos dos fabricantes que fabricam todas as peças amarelas ?
7. Quais os nomes dos fabricantes que passaram a ter peças compradas em 1997 (peças destes fabricantes não eram fornecidas anteriormente) ?

That's all Folks!



nemo