

# Introdução a Bancos de Dados

# Álgebra Relacional

## Parte 2

Mirella M. Moro

Departamento de Ciência da Computação

Universidade Federal de Minas Gerais

[mirella@dcc.ufmg.br](mailto:mirella@dcc.ufmg.br)

# Programa

- Introdução
  - Conceitos básicos, características da abordagem de banco de dados, modelos de dados, esquemas e instâncias, arquitetura de um sistema de banco de dados, componentes de um sistema de gerência de banco de dados.
- Modelos de dados e linguagens
  - Modelo entidade-relacionamento (ER), modelo relacional, [álgebra relacional](#), SQL.
- Projeto de bancos de dados
  - Fases do projeto de bancos de dados, projeto lógico de bancos de dados relacionais, normalização.
- Novas Tecnologias e Aplicações de Banco de Dados

# REVISÃO operadores básicos da álgebra relacional

# Álgebra Relacional: Resumo

- Dadas duas relações R1 e R2
- Operações Básicas
  - *Selection* ( $\sigma$ ) Seleciona um subconjunto de tuplas da relação
  - *Projection* ( $\pi$ ) Seleciona colunas da relação
  - *Rename* ( $\rho$ ) Altera o nome da relação ou dos atributos
  - *Cross-product* ( $\times$ ) Permite combinar R1 e R2 [produto cartesiano]
  - *Set-difference* ( $-$ ) Seleciona tuplas em R1, mas não em R2
  - *Union* ( $\cup$ ) Seleciona tuplas em R1 ou em R2
- Operações Adicionais: Interseção, *junção*, divisão

Já que cada operação retorna uma relação, operações podem ser *compostas* !

# Álgebra Relacional: Seleção

Relation Sells:

bar	beer	price
Joe's	Bud	2.50
Joe's	Miller	2.75
Sue's	Bud	2.50
Sue's	Miller	3.00

JoeMenu  $\leftarrow \sigma_{\text{bar}=\text{"Joe's"}}(\text{Sells}):$

bar	beer	price
Joe's	Bud	2.50
Joe's	Miller	2.75

$\sigma$  mantém  
as colunas

$\sigma$  seleciona  
linhas de  
acordo  
com condição

# Álgebra Relacional: Projeção

Relation Sells:

bar	beer	price
Joe's	Bud	2.50
Joe's	Miller	2.75
Sue's	Bud	2.50
Sue's	Miller	3.00

**MUITO  
IMPORTANTE  
ÁLGEBRA  
ELIMINA  
DUPLICATAS**

Prices  $\leftarrow \pi_{\text{beer,price}}(\text{Sells})$ :

beer	price
Bud	2.50
Miller	2.75
Miller	3.00

$\pi$  mantém  
colunas  
conforme  
definido

# Álgebra Relacional: Prod. Cartesiano

$$R3 \leftarrow R1 \times R2$$

R1(

A,	B )
1	2
3	4

R2(

B,	C )
5	6
7	8
9	10

R3(

A,	R1.B,	R2.B,	C )
1	2	5	6
1	2	7	8
1	2	9	10
3	4	5	6
3	4	7	8
3	4	9	10

PRODUTO CARTESIANO: funciona exatamente igual ao produto cartesiano de conjuntos da matemática PORÉM cada elemento é uma LINHA INTEIRA

Número de colunas = #colunas em R1 + #colunas em R2 → sempre

Número de linhas = #linhas em R1 x #linhas em R2

**Se houver duas colunas com nome igual (B) o resultado fica Tabela.Coluna**

# Álgebra Relacional: Junção

Funciona como um produto cartesiano seguido de uma seleção cuja condição é especificada junto ao operador

Sells(	bar,	beer,	price	)	Bars(	name,	addr	)
	Joe's	Bud	2.50			Joe's	Maple St.	
	Joe's	Miller	2.75			Sue's	River Rd.	
	Sue's	Bud	2.50					
	Sue's	Coors	3.00					

Condição de junção

BarInfo  $\leftarrow$  Sells  $\bowtie_{\text{Sells.bar} = \text{Bars.name}}$  Bars

BarInfo(	bar,	beer,	price,	name,	addr	)
	Joe's	Bud	2.50	Joe's	Maple St.	
	Joe's	Miller	2.75	Joe's	Maple St.	
	Sue's	Bud	2.50	Sue's	River Rd.	
	Sue's	Coors	3.00	Sue's	River Rd.	



# Álgebra Relacional: Junção Natural

Sells(	<b>bar,</b>	beer,	price	)	Bars(	<b>bar,</b>	addr	)
	Joe's	Bud	2.50			Joe's	Maple St.	
	Joe's	Miller	2.75			Sue's	River Rd.	
	Sue's	Bud	2.50					
	Sue's	Coors	3.00					

Condição de junção **implícita** = conjunção da igualdade de todos os pares de colunas de mesmo nome

BarInfo := Sells ⋈ Bars

**Note:** Bars.name agora é Bars.bar para fazer junção natural

BarInfo(	<b>bar,</b>	beer,	price,	addr	)
	Joe's	Bud	2.50	Maple St.	
	Joe's	Miller	2.75	Maple St.	
	Sue's	Bud	2.50	River Rd.	
	Sue's	Coors	3.00	River Rd.	

No resultado: para os pares de colunas de mesmo nome, apenas uma cópia é considerada

# RENAME e sequências de operações

- As relações geradas por expressões da álgebra não têm nome
- Pode ser interessante atribuir um nome a uma relação resultante de uma expressão algébrica, para uso futuro ou como resultado intermediário de uma operação mais complexa
- Ex.: listar nome e salário dos empregados do depto. 5
  - $\pi_{\text{NOME, SOBRENOME, SALARIO}}(\sigma_{\text{NUD}=5}(\text{EMPREGADO}))$
- Equivale a:
  - $\text{DEP5\_EMP} \leftarrow \sigma_{\text{NUD}=5}(\text{EMPREGADO})$
  - $\text{RESULTADO} \leftarrow \pi_{\text{NOME, SOBRENOME, SALARIO}}(\text{DEP5\_EMP})$

# RENAME e seqüências de operações

- É possível renomear atributos:

$TEMP \leftarrow \sigma_{NUD=5}(EMPREGADO))$

$R(N, SN, SAL) \leftarrow \pi_{NOME, SOBRENOME, SALARIO}(DEP5\_EMP)$

- Notação

- $\rho_{S(B1, B2, \dots, Bn)}(R)$  : renomeia atributos e a relação
- $\rho_S(R)$  : renomeia apenas a relação
- $\rho_{(B1, B2, \dots, Bn)}(R)$  : renomeia apenas os atributos
- $B_i$  são os novos nomes dos atributos, na ordem em que existem em R

Figura 6.2 Resultados de uma seqüência de operações.

(a)  $\pi_{\text{PNOME, UNOME, SALARIO}}(\text{EMPREGADO})$

(b)  $\text{TEMP} \leftarrow \sigma_{\text{DNO}=5}(\text{EMPREGADO})$

$\text{R}(\text{PRIMEIRONOME, ULTIMONOME, SALARIO}) \leftarrow \pi_{\text{NOME, SOBRENOME, SALARIO}}(\text{DEP5\_EMP})$

(a)

PNOME	UNOME	SALARIO
John	Smith	30000
Franklin	Wong	40000
Ramesh	Narayan	38000
Joyce	English	25000

(b)

TEMP	PNOME	MINICIAL	UNOME	SSN	DATANASC	ENDereco	SEXO	SALARIO	SUPERSSN	DNO
	John	B	Smith	123456789	1965-01-09	731 Fondren,Houston,TX	M	30000	333445555	5
	Franklin	T	Wong	333445555	1955-12-08	638 Voss,Houston,TX	M	40000	888665555	5
	Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak,Humble,TX	M	38000	333445555	5
	Joyce	A	English	453453453	1972-07-31	5631 Rice,Houston,TX	F	25000	333445555	5

R	PRIMEIRONOME	ULTIMONOME	SALARIO
	John	Smith	30000
	Franklin	Wong	40000
	Ramesh	Narayan	38000
	Joyce	English	25000

# Mais operações e detalhes

1. Divisão
2. Regras de Precedência
3. Junção: Natural, EquiJoin, OuterJoin, SelfJoin
4. Esquemas resultantes

# Divisão

## Divisão de duas relações R e S

- Os valores de um atributo (ou mais) de R que se relacionam com **todos os valores** de um atributo (ou mais) de S
- Utilizada para consultas que incluam o termo **para todos** ou **em todos**
- Requer que R tenha mais atributos que S e pelo menos um atributo em comum

# Divisão: Exemplos

- Quais pessoas possuem contas bancárias em TODOS os bancos estatais do país?
  - Não se sabe quantos bancos são ou quais são, então não é possível fazer uma condição de seleção primeiro
- Quais pessoas possuem cartão de fidelidade para TODAS as companhias aéreas brasileiras?
- Quais estudantes se matricularam em TODAS as disciplinas ofertadas pela profa Mirella?
- Quais estudantes se matricularam em TODAS as disciplinas de primeiro semestre oferecidas em 20172?
- Quais meninas se matricularam em TODAS as disciplinas nas quais o príncipe William se matriculou?
- Quais genomas possuem TODAS as características presentes na proteína X?

Outro Exemplo: Liste os códigos dos clientes que já foram atendidos por todos os vendedores.

Esquema:

cliente (nro\_cli, nome\_cli, end\_cli, saldo)

atende (nro\_cli, cod\_vend)

vendedor (cod\_vend, nome\_vend)

R: atende

nro_cli	cod_vend
9	66
1	04
1	66
4	66
5	04
8	04

S:  $\pi_{\text{cod\_vend}}(\text{vendedor})$

cod_vend
66
04

$R \div S$

nro_cli
1

Se o cliente foi atendido apenas por um dos vendedores, não entra no resultado. É necessário ter sido atendido por todos os dois.



# Exemplos de $A \div B$ (ou $A/B$ )

sno	pno
s1	p1
s1	p2
s1	p3
s1	p4
s2	p1
s2	p2
s3	p2
s4	p2
s4	p4

*A*

pno
p2

*B1*

sno
s1
s2
s3
s4

*A/B1*

pno
p2
p4

*B2*

sno
s1
s4

*A/B2*

pno
p1
p2
p4

*B3*

sno
s1

*A/B3*

**AS**  $\leftarrow \sigma_{\text{sexo}=F}$  (ALUNOS)

Matr	Nome	Sexo	Cr
1	A	F	CC
4	D	F	MC
7	G	F	SI
8	H	F	SI

**M**  $\leftarrow \pi_{\text{Matr}, \text{Disc}, \text{Sem}}$   
(MATRICULAS  $\bowtie$  AS)

Matr	Disc	Sem
1	DCC011	20162
1	DCC851	20162
1	DCC834	20161
4	DCC011	20162
4	DCC851	20162
7	DCC011	20162
7	DCC851	20161
7	DCC834	20171
8	DCC011	20161
8	DCC254	20162
8	DCC234	20171
8	DCC851	20172

Alunas que se matricularam em todos os semestres

a.  $AS \div M$

NÃO FUNCIONA (sem denominador comum)

b.  $\pi_{\text{Matr}, \text{Nome}} AS \div \pi_{\text{Sem}} M$

NÃO FUNCIONA (sem denominador comum)

c.  $\pi_{\text{Matr}, \text{Sem}} M \div \pi_{\text{Sem}} M$

Retorna Matr de todos os semestres

d.  $\pi_{\text{Nome}} (AS \bowtie (\pi_{\text{Matr}, \text{Sem}} M \div \pi_{\text{Sem}} M))$

Retorna Nome matriculado em todos semestres

Alunas matriculadas nas mesmas disciplinas de "1"

a.  $\pi_{\text{Matr}, \text{Disc}} M \div \pi_{\text{Disc}} \sigma_{\text{Matr}=1} (M)$

Agora, no mesmo semestre

b.  $\pi_{\text{Matr}, \text{Disc}} M \div \pi_{\text{Disc}, \text{Sem}} \sigma_{\text{Matr}=1} (M)$

NÃO FUNCIONA (sem denominador comum)

c.  $\pi_{\text{Matr}, \text{Disc}, \text{Sem}} M \div \pi_{\text{Disc}, \text{Sem}} \sigma_{\text{Matr}=1} (M)$

# Divisão: Exemplos

$R(\text{matr}, \text{disc}) \div S(\text{disc}) \rightarrow$  os números de matrícula em R que se relacionam com todos os valores de disciplina em S.

O resultado é apenas a coluna **matr** (pense que na divisão tem disc no numerador e no denominador, então corta 😊).

Traduzindo para expressões algébricas:

Esquema:  $MATRICULAS(\underline{\text{matr}}, \text{disc}, \text{sem})$

$ALUNOS(\underline{\text{matr}}, \text{nome}, \text{curso})$

1.  $(\pi_{\text{matr}, \text{disc}} MATRICULAS) \div (\pi_{\text{disc}} \sigma_{\text{sem}=20172} MATRICULAS) \rightarrow$  número de matrícula dos alunos matriculados em todas as disciplinas oferecidas no semestre de 20172

2.  $((\pi_{\text{matr}, \text{disc}} (MATRICULAS \bowtie (\sigma_{\text{curso}=\text{CC}} ALUNOS))) \div (\pi_{\text{disc}} \sigma_{\text{sem}=20172} MATRICULAS)) \rightarrow$  número de matrícula dos alunos da Computação matriculados em todas as disciplinas oferecidas no semestre 20172

# Regras de Precedência

- Precedência dos operadores relacionais:
  1.  $[\sigma, \pi, \rho]$  (mais alta)
  2.  $[x, \bowtie]$
  3.  $\cap$
  4.  $[\cup, -]$

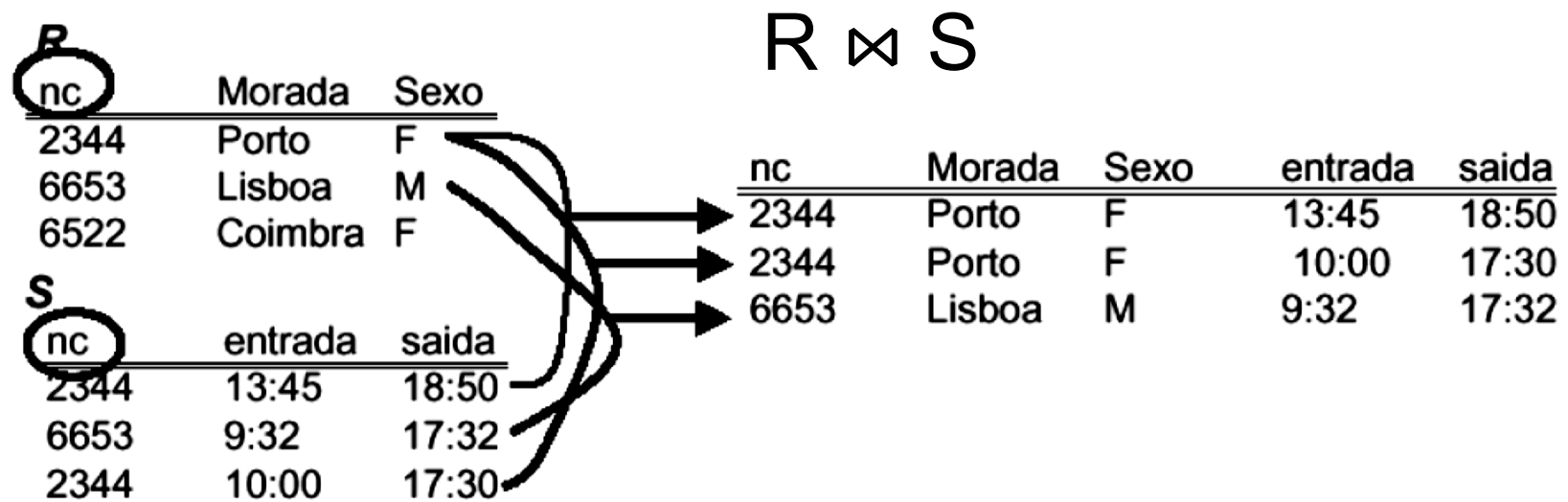
## Junção: Natural, EquiJoin, OuterJoin, SelfJoin

- Produz uma tabela composta de duas outras que se relacionam
- A operação de junção deriva da combinação das operações de produto cartesiano e seleção, sendo executadas conjuntamente.
- Também chamada de “Join”

# Junção Natural

- Conecta duas relações R e S:
  - Conjunção da igualdade de pares de atributos de mesmo nome
  - Projeta uma cópia de cada par dos atributos igualados
- Notação:  $R \bowtie S$

# Junção Natural



# EquiJoin / EquiJunção

- Junção em que a condição da junção tem a forma

$$R.\text{nome1} = S.\text{nome2}$$

- Teste de igualdade em dois campos com nomes diferentes (um campo de cada relação)
- Notação:  $R \bowtie_{R.\text{nome1}=S.\text{nome2}} S$
- Resultado: nome1 e nome2 estarão no esquema final



# Outer Join / Junção Externa

- Para junções naturais, apenas as tuplas de R que também estão em S e vice-versa, são selecionadas.
- *Outer Join*: Permite que tuplas **fora** da junção sejam selecionadas com valores nulos para seus atributos
  1. Realiza a junção normal entre as duas relações
  2. Adiciona as linhas que ficaram de fora (do resultado da junção natural) preenchendo os valores de fora da junção com NULL
- Notação:  $R \bowtie_{R.nome1=S.nome2} S \rightarrow \text{OuterJoin à esquerda}$   
 $R \bowtie S \rightarrow \text{OuterJoin natural à direita (condição implícita tal como junção natural)}$

# Junção Natural, EquiJoin, Outer Join

**T1**

A	B
X	1
Y	2
Z	3

**T2**

B	C
1	D
5	E
6	F

**T3**

C	D
D	a
E	b
G	c

$T1 \bowtie T2$

$T1 \bowtie_{T1.b=T2.b} T2$

$T1 \bowtie T2$

$T1 \bowtie T2$

$T1 \bowtie T2$

$T1 \bowtie T2 \bowtie T3$

# Junção Natural, EquiJoin, OuterJoin

**T1**

A	B
X	1
Y	2
Z	3

**T2**

B	C
1	D
5	E
6	F

**T3**

C	D
D	a
E	b
G	c

$T1 \bowtie T2$

A	B	C
X	1	D

$T1 \bowtie_{T1.b=T2.b} T2$

A	B	B	C
X	1	1	D

$T1 \bowtie T2$

A	B	C
X	1	D
Y	2	
Z	3	

Todas as tuplas de T1 que satisfazem ou não a junção

$T1 \bowtie T2$

A	B	C
X	1	D
	5	E
	6	F

Todas as tuplas de T2 que satisfazem ou não a junção

$T1 \bowtie T2$

A	B	C
X	1	D
Y	2	
Z	3	
	5	E
	6	F

Ambos os lados também é possível

$T1 \bowtie T2 \bowtie T3$

A	B	C	D
X	1	D	a
Y	2		
Z	3		
	5	E	b
	6	F	
		G	c

# Self Join

- Usando  $\text{Sells}(\text{bar}, \text{beer}, \text{price})$ , encontre os bares que vendem dois tipos diferentes de cerveja a mesmo preço
- **Estratégia:** usando renomeação, definir uma cópia de Sells  $\text{S}(\text{bar}, \text{beer1}, \text{price})$

$\text{S}(\text{bar}, \text{beer1}, \text{price}) \leftarrow \text{Sells}$

$\sigma_{\text{beer} \neq \text{beer1}} (\text{S} \bowtie \text{Sells})$

A junção natural de Sell e S consiste de  $(\text{bar}, \text{beer}, \text{beer1}, \text{price})$ , tal que o bar vendo as mesmas cervejas a mesmo preço

# Agregação e Agrupamento

- Operador “F Script”

$$\langle \text{atributos de agrupamento} \rangle \mathcal{F} \langle \text{funções de agregação} \rangle (\text{rel})$$

- Agrupamento: formação de grupos de tuplas de acordo com um critério, como o valor de um atributo  
Default: toda a relação
- Agregação: funções que atuam sobre cada grupo, produzindo um resultado consolidado. Ex: MIN, MAX, AVG, STDEV, COUNT
- O resultado é uma nova relação, com uma tupla por grupo, e uma coluna para cada função de agregação, mais uma coluna com a indicação do grupo (se houver)

# Exemplos

$\text{DEPTO } \mathcal{I}_{\text{COUNT}}(\text{EMPREGADO})$  -> produz a quantidade de empregados por departamento

$\text{DEPTO } \mathcal{I}_{\text{MAX}(\text{SALÁRIO})}(\text{EMPREGADO})$  -> informa o maior salário em cada departamento

$\mathcal{I}_{\text{MAX}(\text{SALÁRIO})}(\text{EMPREGADO})$  -> informa o maior salário da empresa

$\mathcal{I}_{\text{AVG}(\text{SALÁRIO})}(\text{EMPREGADO})$  -> informa o valor médio de salário da empresa

# Esquemas Resultantes

- União, interseção e diferença

- Os esquemas dos dois operandos devem ser compatíveis (mesma sequência de atributos, mesmos domínios), e o resultante é o mesmo também

- Seleção

- O esquema resultante é o mesmo do esquema do operando

- Projeção

- A lista de atributos informa o esquema

# Esquemas Resultantes

- **Produto cartesiano**
  - O esquema é formado pelos atributos das duas relações
  - Usa prefixo R.A, etc., para distinguir atributos com nome A
- **Join**
  - Mesmo que produto
- **Natural join**
  - União dos atributos das duas relações
- **Renaming**
  - O operador define o esquema
- **Agregação e agrupamento**
  - Um atributo para o grupo (se houver), mais um atributo para cada função de agregação



