



Universidade Federal
de Campina Grande



Banco de Dados I

Unidade 3: Modelo Relacional: Conceitos e Álgebra Relacional

Prof. Cláudio de Souza Baptista, Ph.D.
Laboratório de Sistemas de Informação – LSI
UFCG

Histórico

- Foi introduzido por Codd (1970);
- Tornou-se um padrão de fato para aplicações comerciais, devido a sua simplicidade e performance.
- É um modelo formal, baseado na teoria matemática das relações.
- É padronizado ISO: SQL standard
- Um dos SGBD's precursores que implementaram este modelo foi o System R (IBM). Baseado em seus conceitos surgiram: DB2 (IBM), SQL Server, Oracle, Informix, Ingres, Sybase, SQL Server, Postgresql, MySQL, entre outros.

Conceitos do Modelo Relacional

- O modelo relacional representa os dados num BD como uma coleção de tabelas (relações);
- Cada tabela terá um nome, que será único, e um conjunto de atributos com seus respectivos nomes e domínios;
- Todos os valores de uma coluna são do mesmo tipo de dados.

Exemplo

- Exemplo de uma tabela de empregados em uma empresa:

Ex. Empregado

Matr	Nome	Endereço	Função	Salário	Depart
100	Ana	R. Pedro I, 12, A. Branco	Secretária	500,00	D1
250	Pedro	R. J. Silva, 24, Liberdade	Engenheiro	1500,00	D1
108	André	R. Italia, 33, B. Nações	Técnico	950,00	D2
210	Paulo	R. Pará, 98, B. Estados	Engenheiro	1810,00	D2
105	Sônia	R. Oliveira, 76, A. Branco	Engenheiro	2500,00	D1

Conceitos do Modelo Relacional

■ Terminologia:

- Na terminologia de BD Relacional, uma linha é chamada **tupla**, um nome de coluna é chamado de **atributo** e cada tabela é chamada de **relação**.

■ Definições:

- **Domínio:** um domínio D é um conjunto de valores atômicos.

Ex. Fone: conjunto de 10 dígitos

CPF: conjunto de 7 dígitos

data_aniversario: DATE

Exemplo

Diagram illustrating a database table structure and its data:

- Nome da relação:** ALUNO
- Atributos:** Nome, SSN, FoneResidencia, Endereco, FoneEscritorio, Idade, MPG

ALUNO	Nome	SSN	FoneResidencia	Endereco	FoneEscritorio	Idade	MPG
	Benjamin Bayer	305-61-3456	375-1816	2918 Bluebonnet Lane	null	18	3.21
	Katherine Ashby	381-62-1345	375-4409	125 Kirby Road	null	18	2.89
	Dick Davidson	422-11-2320	null	3452 Elgin Road	749-1253	25	3.53
	Charles Cooper	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
	Barbara Benson	533-69-1236	839-8461	7384 Fontana Lane	null	19	3.25

Tuplas: (Indicated by arrows pointing to the data rows)

Conceitos do Modelo Relacional

- A cada domínio está associado um tipo de dados ou formato.

Ex. Fone: (ddd) dddd dddd ; onde $d = \{0,1,2,3,4,5,6,7,8,9\}$

IdadeEmpregado: inteiro entre 16 e 70.

- Um esquema de relação R , denotado por $R(A_1, A_2, \dots, A_n)$, é um conjunto de atributos $R = \{A_1, A_2, \dots, A_n\}$. Cada atributo A_i é o nome de um papel realizado por algum domínio D na relação R .
- O grau de uma relação é o número de atributos que seu esquema contém.

Ex. Seja o esquema:

Estudante(matricula, nome, fone, idade, curso) \Rightarrow grau = 5

Conceitos do Modelo Relacional

- Um instante (snapshot ou estado) de uma relação r , do esquema $R(A_1, A_2, \dots, A_n)$, denotado por $r(R)$, é o conjunto de n -tuplas $r = \{t_1, t_2, \dots, t_n\}$. Cada tupla t é uma lista ordenada de valores $t = \langle v_1, v_2, \dots, v_n \rangle$
- Os termos **intenção** para o esquema R e **extensão** para instante de relação são também empregados.
- Uma instância $r(R)$ é um subconjunto do produto cartesiano dos domínios de R .
- $r(R) \subseteq (\text{dom}(A_1) \times \text{dom}(A_2) \times \dots \times \text{dom}(A_n))$

Conceitos do Modelo Relacional

■ Características de Relações:

- A ordem das tuplas não tem importância;
- Todo atributo possui valor **atômico**
 - Não existe atributo composto,
 - Não existe atributo multi-valorado
- Cada atributo numa relação tem um **nome** que é **único** dentro de uma relação
- NULL para valores desconhecidos ou não aplicáveis.
Por default, toda coluna exceto a chave aceita NULL.

Conceitos do Modelo Relacional

■ Chaves de uma relação:

- Uma relação pode ter várias chaves, chamadas chaves candidatas (Ex. CPF, RG, Matricula).
- Apenas uma das chaves candidatas é escolhida para ser Chave Primária (Primary Key – PK). As demais, são chamadas chaves alternativas ou secundárias/
- Convenciona-se sublinhar os atributos que compõem a chave primária.

Ex.: Empregado(Matrícula, Nome, Endereço, Função, Salário)

- A chave primária pode ser **Simples** ou **Composta**
- Ex. Chave primária simples

Estudante(matr, nome, rua, num, bairro, cidade, estado, cep)

- Ex. Chave primária composta

Matricula (matr, coddisc, sem, turma, nota)

Conceitos do Modelo Relacional

■ Restrições de Integridade Básicas:

- **Integridade de Domínio:** toda tem um domínio associado. Um tipo primitivo como `int`, `float`, `date`, `boolean`, `char`, etc.
- **Integridade NOT NULL:** indica que o valor de uma coluna `não pode` ser NULL

Conceitos do Modelo Relacional

■ Restrições de Integridade básicas:

- **Integridade de Chave:** toda tupla tem um conjunto de atributos que a identifica de maneira única na relação;
- **Integridade de Entidade:** nenhum valor de chave primária poderá ser NULO.
- **Integridade Referencial:** uma relação pode ter um conjunto de atributos que contêm valores com mesmo domínio de um conjunto de atributos que forma a chave primária de uma outra relação. Este conjunto é chamado de **chave estrangeira**.

Conceitos do Modelo Relacional

Exemplo de Integridade Referencial

Departamento

CodDep	Nome	MatrGerent
D2	Produção	210
D1	Custos	105
D5	Pessoal	NULL

↓
Empregado

Matr	Nome	Endereço	Função	Salário	Depart
100	Ana	R. Pedro I, 12, A. Branco	Secretária	500,00	D1
250	Pedro	R. J. Silva, 24, Liberdade	Engenheiro	1500,00	D1
108	André	R. Italia, 33, B. Nações	Técnico	950,00	D2
210	Paulo	R. Pará, 98, B. Estados	Engenheiro	1810,00	D2
105	Sônia	R. Oliveira, 76, A. Branco	Engenheiro	2500,00	D1

Conceitos do Modelo Relacional

- Um conjunto de atributos de uma relação **R1** é uma chave estrangeira se satisfaz às seguintes regras:
 - Os atributos da chave estrangeira têm o mesmo domínio dos atributos da chave primária de outra relação **R2**;
 - Um valor da chave estrangeira em uma tupla **t1** de **R1** possui o mesmo valor da chave primária para alguma tupla **t2** em **R2** ou é NULO.
- A integridade referencial estabelece que todo valor de chave estrangeira numa relação deve corresponder a um valor de chave primária de uma segunda relação ou deve ser nulo.

Conceitos do Modelo Relacional

- Uma chave estrangeira pode referenciar-se à sua própria relação (auto-relacionamento).

Ex.: Empregado(Mat, Nome, Salário, Matr_supervisor)

- As restrições de integridade devem ser implementadas pelo SGBD. Muitos SGBD's implementam integridade de chave e de entidade, mas não implementam integridade referencial.

- Estudaremos as integridades semânticas mais a frente.

Ex.: “Nenhum Empregado pode ganhar mais que seu gerente”

Operações de atualização em Relações

■ Inserção:

- Inserir <'102', 'André', NULL, 'Engenheiro', '1.980', 'D2'>
é aceito sem problemas.
- Inserir <'100', 'Maria', NULL, 'Técnica', '950', 'D1'>
viola a restrição de chave.
- Inserir <NULL, 'Cecília', NULL, 'Engenheiro', '1.950', 'D1'>
viola a restrição de integridade de entidade
- Inserir <'308', 'Mauro', 'Rua 4', 'Técnico', '980', 'B6'>
viola a restrição de integridade referencial

Operações de atualização em Relações

■ Remoção:

- Remover da tabela *Empregado* a tupla com matrícula = '100'.

remoção aceita sem problemas.

- Remover da tabela Departamento a tupla com CodDep = 'D1'.

viola a regra de integridade referencial, pois existem empregados que estão alocados neste departamento.

Operações de atualização em Relações

■ Modificação:

- Modificar o salário do empregado com matrícula = '250'

operação aceita sem problemas.

- Modificar o número do departamento da tupla de empregado com matrícula '210' para 'D1'

operação aceita sem problemas.

- Modificar o número do departamento de empregado '108' para 'D9'

viola a integridade referencial

Ex. Esquema de BD Relacional

Empregado(matr, nomeE, end, sexo, salário, superv, depto)

depto REFERENCIA Departamento,

superv REFERENCIA Empregado

Departamento(codDepto, nomeD, matrGerente),

matrGerente REFERENCIA Empregado

DepLocalizações(codDepto, localização)

coddepto REFERENCIA Departamento

Alocação(matrEmp, codProj, numHoras)

matrEmp REFERENCIA Empregado

codProj REFERENCIA Projeto

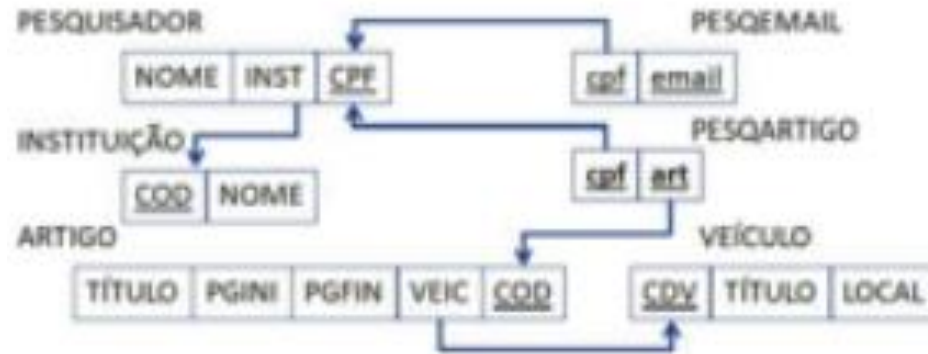
Projetos(codProj, nome, localização, deptoControla)

deptoControla REFERENCIA Departamento

Dependentes(matrEmp, nomeDep, sexo, dataNasc)

matrEmp REFERENCIA Empregado

Ex.2 de ESQUEMA relacional



Pesquisador (CPF, nome, inst)

inst referencia Instituição

PesqEmail (CPF, email)

CPF referencia Pesquisador

Instituição (cod, nome)

Artigo (cod, título, pgini, pgfin, veic)

veic referencia Veículo

Veículo (cdv, título, local)

PesqArtigo (cpf, art)

cpf referencia Pesquisador

art referencia Artigo

Álgebra Relacional

- É uma linguagem de Banco de Dados procedural e formal.

Seja o esquema relacional de uma empresa hipotética a seguir:

Empregado(matr, nomeE, endereço, sexo, salário, supervisor, depto)

Departamento(codDepto, nomeD, matrGerente)

DepLocalizações(codDepto, localização)

Alocação(matrEmp, codProj, numHoras)

Projetos(codProj, nome, localização, deptoControla)

Dependentes(matrEmp, nomeDep, sexo, dataNasc, parentesco)

Álgebra Relacional

- Usaremos o esquema anterior para exemplificar os diversos operadores da álgebra relacional.
- **Operação de Seleção (σ)**
 - Seleciona um subconjunto de tuplas de uma relação, de acordo com uma condição
 - Sintaxe: $\sigma_{\langle \text{predicado} \rangle} (\langle \text{Relação} \rangle)$
 - Onde:
 - σ = operador de seleção
 - **Predicado:** $\langle \text{atributo} \rangle \langle \text{op} \rangle \langle \text{constante} \rangle$
ou $\langle \text{atributo} \rangle \langle \text{op} \rangle \langle \text{atributo} \rangle$
 $\langle \text{op} \rangle = \{=, >, <, \leq, \geq, \neq\}$
 - No predicado podemos ter as cláusulas conectadas pelos conectivos booleanos AND, OR e NOT,

Álgebra Relacional

Exemplo 1: Selecione os empregados que trabalham no departamento 4.

$$\sigma_{\text{DEPTO} = 4} (\text{Empregado})$$

Exemplo 2: Selecione os empregados que ganham mais de R\$ 3.000,00

$$\sigma_{\text{SALARIO} > 3000} (\text{Empregado})$$

Exemplo 3: Selecione os empregados que ganham mais de R\$ 2.000,00 e trabalham no departamento 4, ou ganham menos de R\$ 500,00 e trabalham no departamento 5.

$$\sigma_{(\text{SALARIO} > 3000 \text{ AND DEPTO} = 4) \text{ OR } (\text{SALARIO} < 500 \text{ AND DEPTO} = 5)} (\text{Empregado})$$

Álgebra Relacional

- O operador de seleção é unário (aplicado a uma única relação);
- O grau da relação resultante é o mesmo da relação original;
- O número de tuplas da relação resultante é menor ou igual ao número de tuplas da relação original;
- A seleção é comutativa:

$$\sigma_{\langle \text{COND1} \rangle} (\sigma_{\langle \text{COND2} \rangle} (R)) = \sigma_{\langle \text{COND2} \rangle} (\sigma_{\langle \text{COND1} \rangle} (R))$$

$$\sigma_{\langle \text{COND1} \rangle} (\sigma_{\langle \text{COND2} \rangle} (\dots (\sigma_{\langle \text{CONDn} \rangle} (R)) \dots))$$

Álgebra Relacional

■ Operação de Projeção (π)

- Seleciona um subconjunto de atributos de uma dada relação.
- Sintaxe: π <LISTA DE ATRIBUTOS> (<Relação>)
- Obs.: se a lista de atributos inclui apenas atributos não-chave, tuplas duplicadas poderão aparecer no resultado, porém, a operação de projeção elimina esta duplicação.

Ex.:

π <FUNÇÃO, SALÁRIO> (Empregado)

Álgebra Relacional

■ Operação de Projeção (π)

- O número de tuplas resultante será menor ou igual ao da relação original.

$$\pi_{\langle \text{LISTA-1} \rangle} (\pi_{\langle \text{LISTA-2} \rangle} (\text{Relação})) = \\ (\pi_{\langle \text{LISTA-1} \rangle} (\text{Relação})) \Leftrightarrow \langle \text{LISTA-2} \rangle \supset \langle \text{LISTA-1} \rangle$$

Exemplo:

$$\pi_{\text{NOME, SALÁRIO}} (\pi_{\text{NOME, FUNÇÃO, SEXO, SALÁRIO}} (\text{Empregado})) = \\ \pi_{\text{NOME, SALÁRIO}} (\text{Empregado})$$

Álgebra Relacional

■ Combinando Seleção e Projeção:

Exemplo: Obtenha o nome e o salário dos empregados do departamento 5.

$$\pi_{\text{NOME, SALÁRIO}} (\sigma_{\text{DEPTO} = 5} (\text{Empregado}))$$

- Alternativamente podemos usar uma notação que usa uma sequência dos resultados dando nome às relações intermediárias:

$$\text{EmpDepto5} \leftarrow \sigma_{\text{DEPTO} = 5} (\text{Empregado})$$

$$\text{Resultado} \leftarrow \pi_{\text{NOME, SALÁRIO}} (\text{EmpDepto5})$$

Álgebra Relacional

■ União (\cup)

- A união de duas relações, $R \cup S$, é o conjunto de tuplas que está em R ou S ou em ambas. Duplicatas são eliminadas.

R

A	B	C
a	b	c
d	a	f
c	b	d

S

D	E	F
b	g	a
d	a	f

$R \cup S$

a	b	C
d	a	f
c	b	d
b	g	a

Álgebra Relacional

Exemplo: Obtenha a matrícula dos empregados que trabalham no departamento 5 ou supervisionam empregados que trabalham no departamento 5.

$\text{EmpDepto5} \leftarrow \sigma_{\text{DEPTO} = 5} (\text{Empregado})$

$\text{Temp1} \leftarrow \pi_{\text{MATRÍCULA}} (\text{EmpDepto5})$

$\text{Temp2} \leftarrow \pi_{\text{SUPERVISOR}} (\text{EmpDepto5})$

$\text{Resultado} \leftarrow \text{Temp1} \cup \text{Temp2}$

Álgebra Relacional

■ Interseção (\cap)

- A interseção de duas relações, $R \cap S$, é uma relação que inclui todas as tuplas que estão em R e em S.

R		
A	B	C
a	b	c
d	a	f
c	b	d

S		
D	E	F
b	g	a
d	a	f

$R \cap S$		
d	a	f

Álgebra Relacional

■ Diferença (-)

- A diferença entre duas relações, $R - S$, é o conjunto de tuplas que estão em R mas não estão em S .

R

A	B	C
a	b	c
d	a	f
c	b	d

S

D	E	F
b	g	a
d	a	f

R - S

a	b	c
c	b	d

Álgebra Relacional

■ Observações:

□ As operações de conjunto (União, Interseção, Diferença) devem ser **compatíveis de união**. Duas relações $R(A_1, A_2, \dots, A_n)$ e $S(B_1, B_2, \dots, B_n)$ são compatíveis de união se têm o mesmo grau e $\text{domínio}(A_i) = \text{domínio}(B_i)$, para $1 \leq i \leq n$.

$$\square R \cup S = S \cup R$$

$$\square R \cap S = S \cap R$$

$$\square R - S \neq S - R$$

$$\square R \cup (S \cap T) = (R \cup S) \cap T$$

$$\square R \cap (S \cup T) = (R \cap S) \cup T$$

Álgebra Relacional

■ Produto Cartesiano (X)

- O produto cartesiano de duas relações $R \times S$ combina cada tupla de R com cada tupla de S .
- O resultado de $R(A_1, A_2, \dots, A_n) \times S(B_1, B_2, \dots, B_m)$ é uma relação Q com $n + m$ atributos $Q(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$.
- Se R tem x tuplas e S tem y tuplas $\rightarrow R \times S$ terá $x*y$ tuplas.

Álgebra Relacional

Exemplo: Obtenha para cada empregado do sexo feminino, uma lista dos nomes de seus dependentes.

Mulher $\leftarrow \sigma_{\text{sexo} = 'F'} (\text{Empregado})$

NomesMulheres $\leftarrow \pi_{\text{matricula}, \text{nome}} (\text{Mulher})$

DependentesMulher1 $\leftarrow \text{NomesMulheres} \times \text{Dependentes}$

DependentesMulher2 $\leftarrow \sigma_{\text{matr} = \text{matrEmp}} (\text{DependentesMulher1})$

Resultado $\leftarrow \pi_{\text{nomeE}, \text{nomeDep}} (\text{DependentesMulher2})$

Álgebra Relacional

■ Junção ($|X|$)

- Uma junção de duas relações $R(A_1, A_2, \dots, A_n) \times S(B_1, B_2, \dots, B_m)$, denotada por $R \mid X \langle \text{condição de junção} \rangle S$, é usada para combinar tuplas de duas relações em uma única tupla.
- O resultado de uma junção é uma relação Q com $n + m$ atributos $Q(A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m)$.
- Q contém uma tupla para cada combinação de tuplas ($R \times S$) que satisfaz a condição de junção.
- Uma condição de junção tem a forma:

$\langle \text{cond} \rangle \text{ AND } \langle \text{cond} \rangle \text{ AND } \dots \text{ AND } \langle \text{cond} \rangle$

onde: $\langle \text{cond} \rangle = A_i \theta B_i \mid A_i \text{ é atributo de } R; B_i \text{ é atributo de } S;$
 $\text{dom}(A_i) = \text{dom}(B_i); \theta = \{=, <, >, \leq, \geq, \neq\}$

Álgebra Relacional

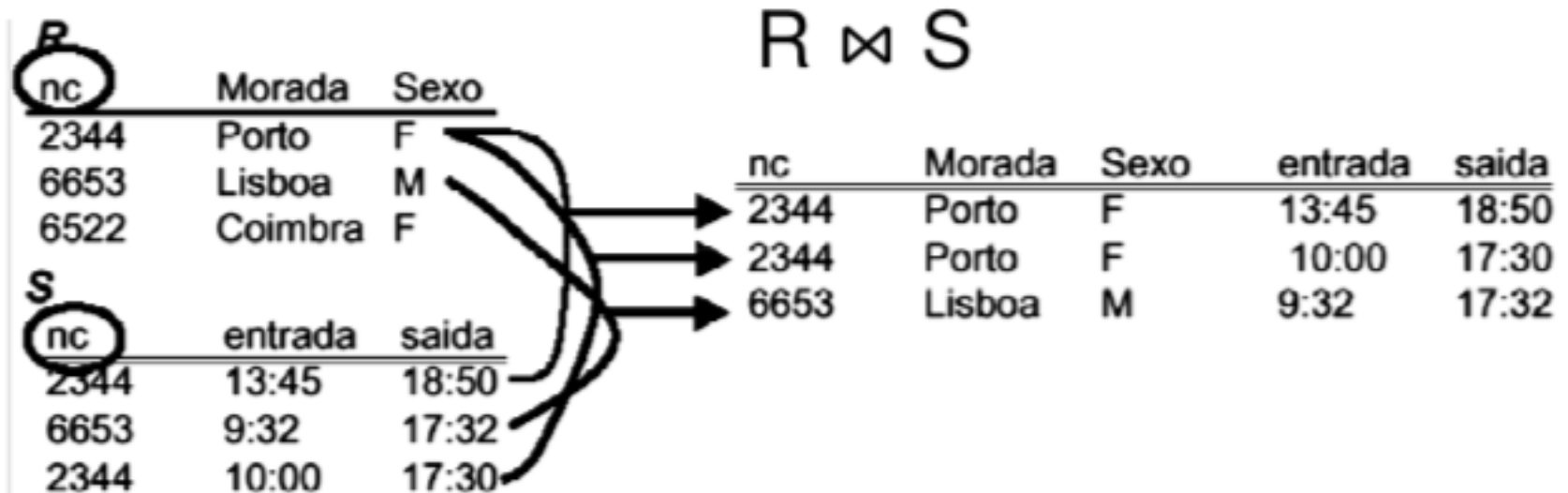
Exemplo: Obtenha o nome do gerente de cada departamento.

$\text{DeptoGer} \leftarrow \text{Departamento} \mid x \mid_{\text{matrGer} = \text{matr}} \text{Empregado}$

$\text{Resultado} \leftarrow \pi_{\text{nomeD}, \text{nomeE}} (\text{DeptoGer})$

- Quando a condição de junção é uma igualdade a junção é chamada de **equijoin**.
- **Junção Natural** é uma equijoin onde um dos atributos com mesmo nome são comparados, e um dos valores repetidos (condição de junção), é eliminado.
- Notação: $R \mid x \mid S$
- Veja que não tem condição de junção explícita, pois a mesma é implícita (conjunção de comparações de igualdade de atributos com o mesmo nome)!!!!

Junção Natural



SelfJoin

- Ex: Usando **Bares(bar, cerveja, preço)**, encontre os bares que vendem dois tipos diferentes de cerveja a mesmo preço
- Estratégia: usando renomeação, definir uma cópia de Bares: $S(\text{bar}, \text{cerveja1}, \text{preço})$:
 $S(\text{bar}, \text{cerveja1}, \text{preço}) \leftarrow \text{Bares}$
 $\text{Res} \leftarrow \sigma_{\text{cerveja} \neq \text{cerveja1}} (S \bowtie \text{Bares})$

A junção natural de Bares e S consiste de $(\text{bar}, \text{cerveja}, \text{cerveja1}, \text{preço})$, tal que o bar vende diferentes cervejas a mesmo preço, i.e $(\text{bar} = \text{bar} \text{ and } \text{preço} = \text{preço})$

Divisão

■ Divisão (\div)

- A divisão de duas relações, $R \div S$, onde $\text{atributos}(S) \subseteq \text{atributos}(R)$, resulta na relação T com $\text{atributos}(T) = \{ \text{atributos}(R) - \text{atributos}(S) \}$; onde, para cada tupla t que aparece no resultado, os valores de t devem aparecer em R combinado com cada tupla de S .

Álgebra Relacional - Divisão

Exemplo de divisão

R

A	B
a1	b1
a2	b1
a3	b1
a4	b1
a1	b2
a3	b2
a2	b3
a3	b3
a4	b3
a1	b4
a2	b4
a3	b4

S

A
a1
a2
a3

R ÷ S

B
b1
b4

Ex. Divisão

- Quais pessoas possuem contas bancárias em TODOS os bancos estatais do país?
- Quais pessoas possuem cartão de fidelidade para TODAS as companhias aéreas brasileiras?
- Quais estudantes se matricularam em TODAS as disciplinas ofertadas pela prof. João Silva?
- Quais estudantes se matricularam em TODAS as disciplinas de primeiro semestre oferecidas em 2018.2?
- Quais meninas se matricularam em TODAS as disciplinas nas quais Ângelo Jolie se matriculou?
- Quais genomas possuem TODAS as características presentes na proteína X?

Álgebra Relacional - Divisão

Obs.: Quase sempre, a divisão é usada quando temos nas consultas frases do tipo “para todos”.

Exemplo: Obtenha o nome dos empregados que trabalham em todos os projetos que Silva trabalha.

$Silva \leftarrow \sigma_{nome = 'silva'} (Empregado)$

$ProjSilva \leftarrow \pi_{codProj} (Alocação \mid x \mid_{matrEmp = matr} Silva)$

$ProjEmp \leftarrow \pi_{codProj, matrEmp} (Alocação)$

$TrabProjSilva \leftarrow (ProjEmp \div ProjSilva)$

$Result \leftarrow \pi_{nome} (TrabProjSilva \mid x \mid_{matrEmp = matr} Empregado)$

Álgebra Relacional

■ Operação Rename (ρ)

- Permite que renomeemos relações e/ou atributos para que se evite a ambiguidade na hora de compararmos atributos com mesmo nome de diferentes relações;
- Pode renomear uma relação ou os atributos da relação ou ambos;
- A operação Rename quando aplicada a uma relação **R** de grau **n** é denotada por:

$$\rho_{S(B1, B2, \dots, Bn)} (R)$$

ou $\rho_S (R)$

ou $\rho_{(B1, B2, \dots, Bn)} (R)$

onde:

ρ denota o operador Rename

S é o nome da nova relação

B1, ..., Bn são os nomes dos novos atributos

Consultas

Obtenha o nome e o endereço de todos os empregados do departamento de Pesquisa.

$\text{Pesquisa} \leftarrow \sigma_{\text{nomeD} = \text{'Pesquisa'}} (\text{Departamento})$

$\text{Resultado} \leftarrow \pi_{\text{nomeE}, \text{endereço}} ($
 $\text{Empregado } |x|_{\text{depto} = \text{codDepto Pesquisa}})$

Consultas

Para cada projeto localizado em 'Natal', liste o código do projeto, o código do departamento que controla o projeto e o nome, endereço e salário do gerente deste departamento.

$\text{ProjNatal} \leftarrow \sigma_{\text{localização} = \text{'Natal'}} (\text{Projetos})$

$\text{Result1} \leftarrow (\text{ProjNatal} \mid x \mid_{\text{depControla} = \text{codDepto}} \text{Departamento})$

$\text{Result2} \leftarrow (\text{Result1} \mid x \mid_{\text{matrGerente} = \text{matr}} \text{Empregado})$

$\text{Resultado} \leftarrow \pi_{\text{codProj}, \text{codDepto}, \text{nomeE}, \text{endereço}, \text{salario}} (\text{Result2})$

Consultas

Faça uma lista dos códigos dos projetos que envolvem um empregado cujo nome é 'Silva' como trabalhador ou como gerente do departamento que controla o projeto.

$Silva \leftarrow \pi_{matrEmp} (\sigma_{nomeE = 'silva'} (Empregado)$

$ProjSilvaPart \leftarrow \pi_{codProj} (Silva \mid x \mid_{matr = matrEmp} Alocação)$

$DepSilvaGer \leftarrow \pi_{codDepto} (Silva \mid x \mid_{matr = matrGerente}$
Departamento)

$ProjDepSilGer \leftarrow \pi_{codProj} (DepSilvaGer$
 $\mid x \mid_{codDepto = deptoControla} Projeto)$

$Resultado \leftarrow ProjSilvaPart \cup ProjDepSilGer$

Consultas

Liste os nomes dos empregados que não têm dependentes.

$\text{TodosEmpr} \leftarrow \pi_{\text{matr}} (\text{Empregado})$

$\text{EmpComDep} \leftarrow \pi_{\text{matrEmp}} (\text{Dependentes})$

$\text{EmpSemDep} \leftarrow (\text{TodosEmpr} - \text{EmpComDep})$

$\text{EmpSemDep2} \leftarrow \rho_{(\text{matricula})} (\text{EmpSemDep})$

$\text{Resultado} \leftarrow \pi_{\text{nomeE}} (\text{EmpSemDep2}$

$|x|_{\text{matricula} = \text{matr}} (\text{Empregado})$

Consultas

Liste os nomes dos gerentes que têm pelo menos 1 dependente.

$\text{Gerentes} \leftarrow \pi_{\text{matrGerente}} (\text{Departamento})$

$\text{EmpComDepend} \leftarrow \pi_{\text{matrEmp}} (\text{Dependentes})$

$\text{GerComDepend} \leftarrow \rho_{(\text{matrG})} (\text{Gerentes} \cap \text{EmpComDepend})$

$\text{Resultado} \leftarrow \pi_{\text{nomeE}} (\text{GerComDepend}$
 $\quad \quad \quad |x|_{\text{matrG} = \text{matr Empregado}})$

Consultas

Encontre os nomes dos empregados que trabalham em todos os projetos do departamento 6.

$$\text{ProjDep6} \leftarrow \pi_{\text{codProj}} (\sigma_{\text{depControla} = 6} (\text{Projetos}))$$

$$\text{Temp_1} \leftarrow \pi_{\text{matrEmp}, \text{codProj}} (\text{Alocação})$$

$$\text{Temp_2} \leftarrow (\text{Temp_1} \div \text{ProjDep6})$$

$$\text{Resultado} \leftarrow \pi_{\text{nomeE}} (\text{Temp_2} \mid x \mid_{\text{matrEmp} = \text{matr}} \text{Empregado})$$

Operadores Adicionais

- Mais recentemente, foram incorporados à Álgebra Relacional, novos operadores como:
 - Projeção Generalizada
 - Funções de Agregação e Agrupamento
 - Outer Joins
- Entretanto, estes operadores **não** serão cobertos neste curso!!!!