

Banco de Dados – IMD0401

Aula 12 – Álgebra Relacional

João Carlos Xavier Júnior

jcxavier@imd.ufrn.br

Álgebra Relacional

□ Conceitos:

- ❖ É uma linguagem de consulta procedural.
- ❖ Os operadores de álgebra relacional recebem **uma** ou **duas** relações como operandos e produzem uma **nova relação** como resultado.
- ❖ Classificação dos **operadores**:
 - Fundamentais: seleção, projeção, produto cartesiano, união e diferença.
 - Adicionais: interseção, junção, divisão e junção externa.

Álgebra Relacional

- ❑ Operadores relacionais:

- ❖ $=, \neq, <, \leq, >, \geq$

- ❑ Operadores lógicos:

- ❖ \wedge (e), \vee (ou)

- ❑ Operações (tipos):

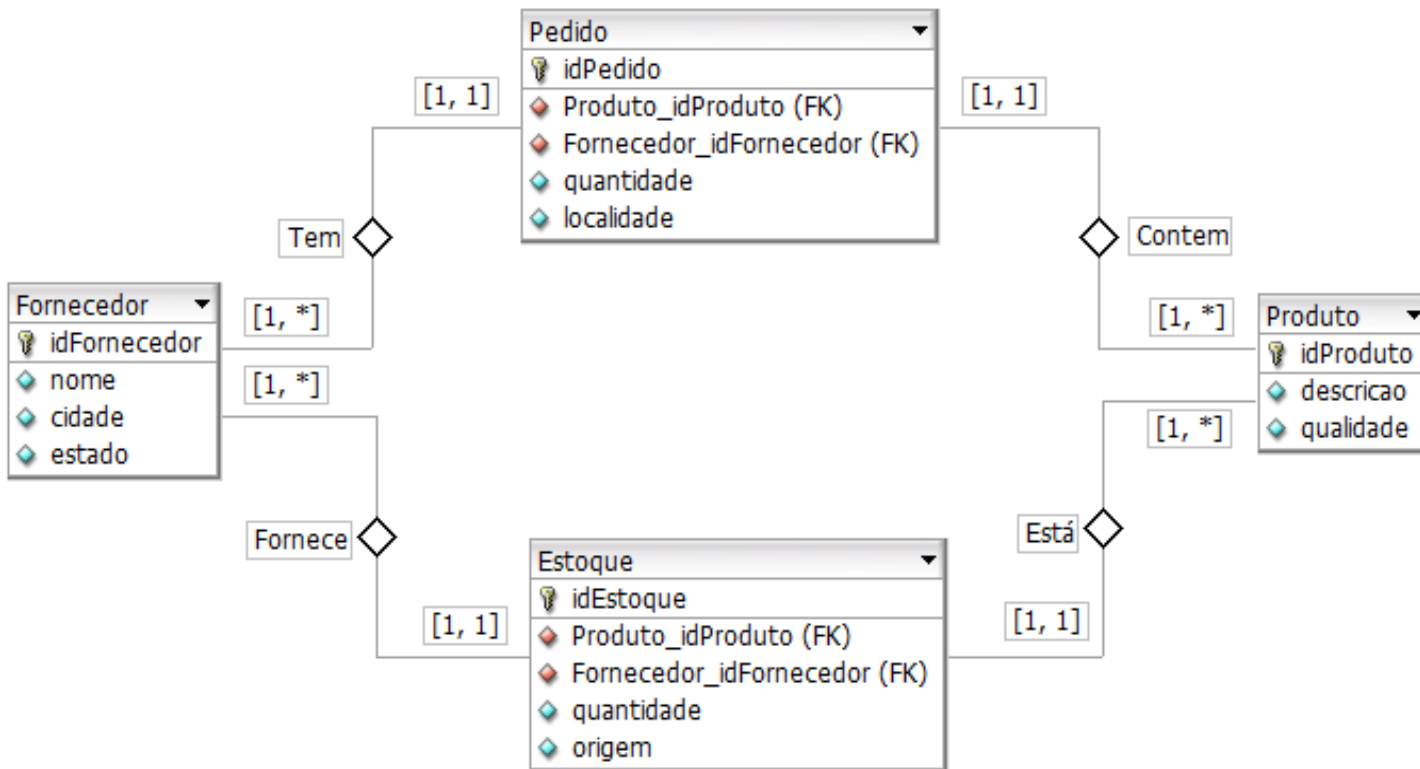
- ❖ Unárias: envolvem apenas uma tabela.

- ❖ Binárias: envolvem duas tabelas.

Álgebra Relacional

Esquema Relacional:

❖ Controle de estoque



Esquema Relacional

□ Tabela Fornecedor:

File Edit View Tools Help				
100 rows				
	idfornecedor [PK] integer	nome character varying(45)	cidade character varying(20)	estado character varying(2)
1	1	Pedro Henrique	Natal	RN
2	2	Eliana Silva	Parnamirim	RN
3	3	Antônio dos Santos	Recife	PE
4	4	Maria José	Campina Grande	PB
5	5	Ana das Virgens	Natal	RN

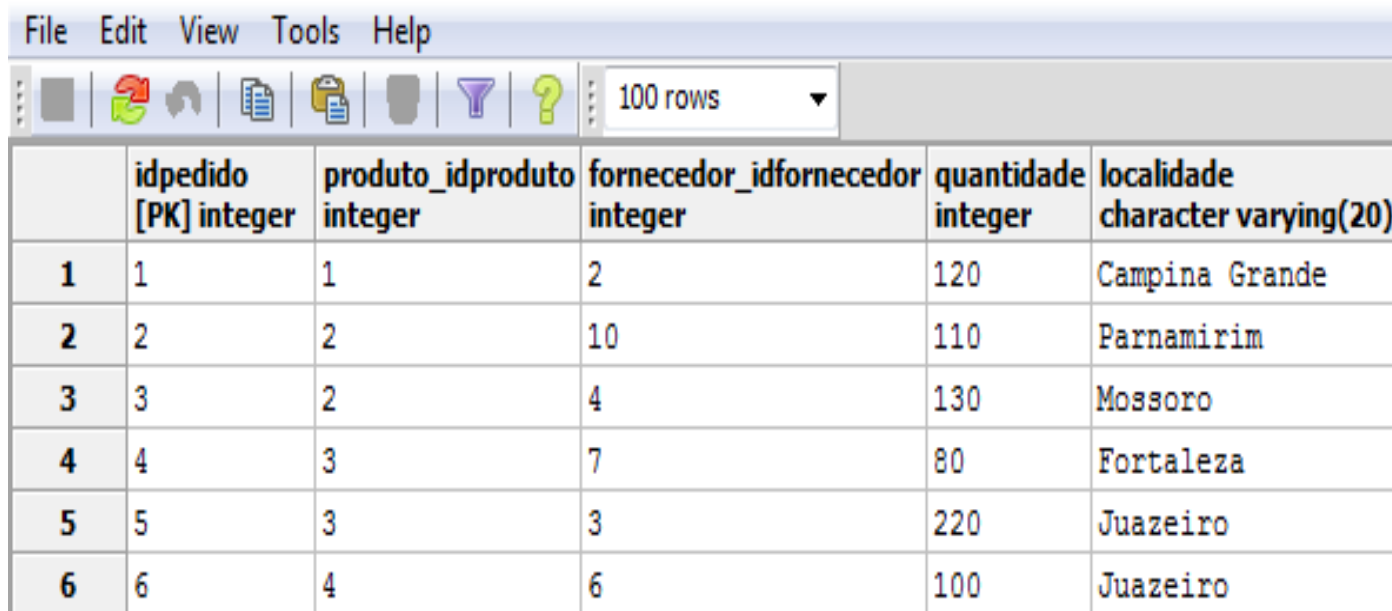
Esquema Relacional

□ Tabela Produto:

	idproduto [PK] integer	descricao character varying(45)	qualidade character varying(2)
1	1	Soja	1a
2	2	Laranja	1a
3	3	Laranja	2a
4	4	Arroz	1a
5	5	Arroz	2a
6	6	Uva	1a
7	7	Cacau	1a
8	8	Trigo	1a
9	9	Feijão	1a
10	10	Feijão	2a

Esquema Relacional

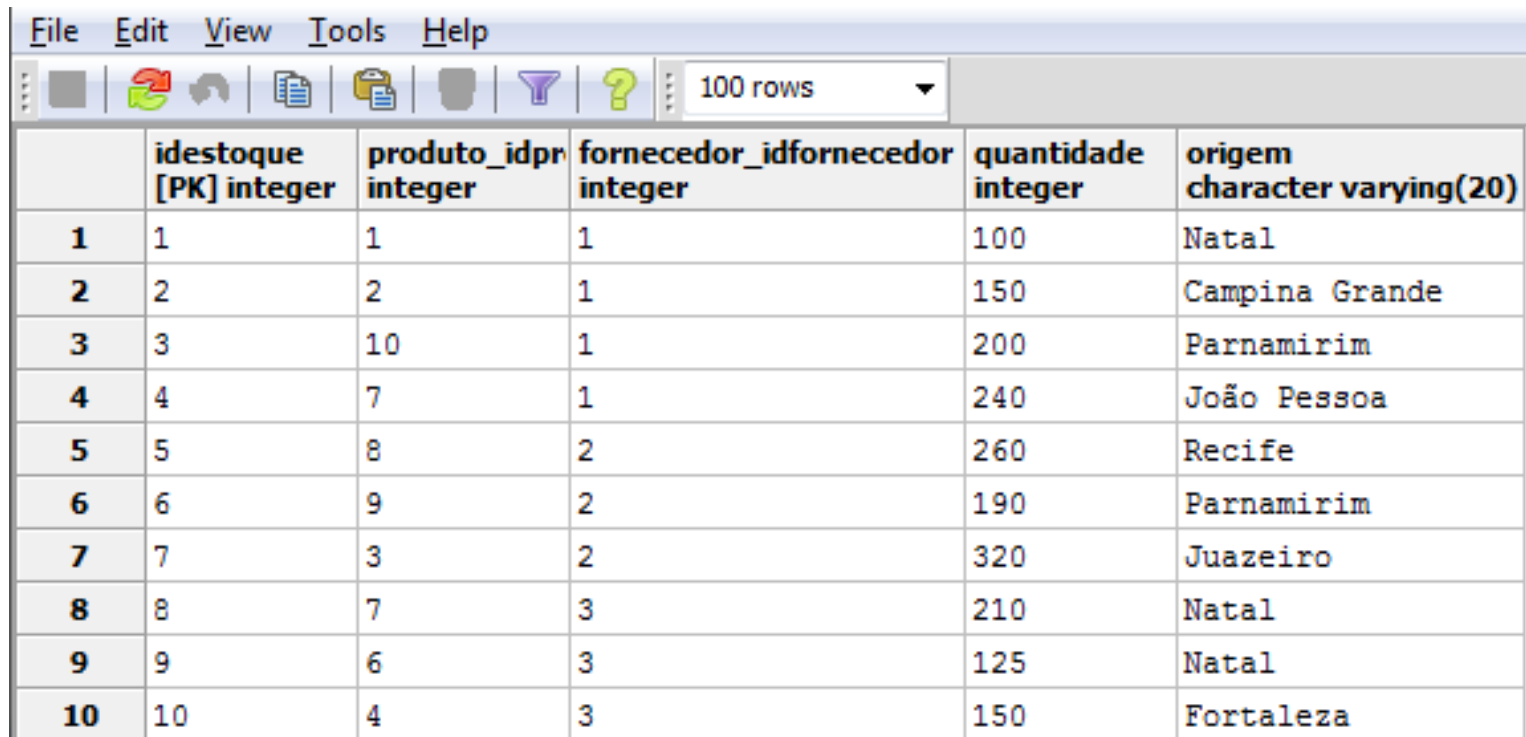
□ Tabela Pedido:



	idpedido [PK] integer	produto_idproduto integer	fornecedor_idfornecedor integer	quantidade integer	localidade character varying(20)
1	1	1	2	120	Campina Grande
2	2	2	10	110	Parnamirim
3	3	2	4	130	Mossoro
4	4	3	7	80	Fortaleza
5	5	3	3	220	Juazeiro
6	6	4	6	100	Juazeiro

Esquema Relacional

□ Tabela Estoque:



	idestoque [PK] integer	produto_id integer	fornecedor_id integer	quantidade integer	origem character varying(20)
1	1	1	1	100	Natal
2	2	2	1	150	Campina Grande
3	3	10	1	200	Parnamirim
4	4	7	1	240	João Pessoa
5	5	8	2	260	Recife
6	6	9	2	190	Parnamirim
7	7	3	2	320	Juazeiro
8	8	7	3	210	Natal
9	9	6	3	125	Natal
10	10	4	3	150	Fortaleza

Álgebra Relacional

❑ Seleção (σ):

- ❖ Seleciona tuplas (linhas) que satisfazem um dado **predicado** (condição lógica) nos valores dos **atributos**.

❖ Exemplo:

$\sigma_{\text{Cidade} = \text{"Parnamirim"}} (\text{Fornecedor})$

Atributo *Condição* *Valor*

Entidade



❖ Resultado:

	idfornecedor integer	nome character varying(45)	cidade character varying(20)	estado character varying(2)
1	2	Eliana Silva	Parnamirim	RN

Álgebra Relacional

❑ Seleção (σ):

σ Quantidade ≥ 200 (Estoque)

	idestoque integer	produto_idproduto integer	fornecedor_idfornecedor integer	quantidade integer	origem character varying(20)
1	3	1	10	200	Parnamirim
2	4	1	7	240	João Pessoa
3	5	2	8	260	Recife
4	7	2	3	320	Juazeiro
5	8	3	7	210	Natal

σ Quantidade $> 100 \wedge$ Localidade = 'Mossoro'
(Pedido)

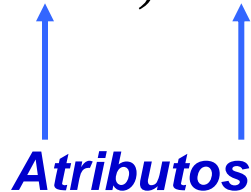
	idpedido integer	produto_idproduto integer	fornecedor_idfornecedor integer	quantidade integer	localidade character varying(20)
1	3	2	4	130	Mossoro

Álgebra Relacional

❑ Projeção (π):

- ❖ Copia a relação dada como argumento, deixando alguns atributos (*colunas*) de lado.
- ❖ Exemplo:

π **Nome, Cidade** (σ **Estado = “RN”** (Fornecedor))


Atributos

- ❖ Resultado:

	nome character varying(45)	cidade character varying(20)
1	Pedro Henrique	Natal
2	Eliana Silva	Parnamirim
3	Ana das Virgens	Natal

Álgebra Relacional

□ Projeção (π):

π IdFornecedor, IdProduto, Quantidade (σ Origem = 'Parnamirim'
(Estoque))

	fornecedor_idfornecedor integer	produto_idproduto integer	quantidade integer
1	10	1	200
2	9	2	190

π IdFornecedor, IdProduto, Localidade (σ Quantidade \geq 100
(Pedido))

	fornecedor_idfornecedor integer	produto_idproduto integer	localidade character varying(20)
1	2	1	Campina Grande
2	10	2	Parnamirim
3	4	2	Mossoro
4	3	3	Juazeiro

Álgebra Relacional

□ Produto Cartesiano (\times):

- ❖ Permite combinar informações de duas relações.
- ❖ O esquema resultante é a concatenação dos esquemas das duas relações fornecidas como argumento.
- ❖ As linhas são obtidas combinando-se cada linha da primeira tabela com todas as linhas da segunda tabela.

❖ Exemplo:

***Produto**(IdProduto, Descricao, Qualidade)*
 \times ***Estoque**(IdProduto, Quantidade)*

Álgebra Relacional

❑ Produto Cartesiano (\times):

❖ Explicando

π Descricao, Qualidade, Quantidade

$(\sigma_{\text{Produto.IdProduto} = \text{Estoque.IdProduto}}(\text{Produto} \times \text{Estoque}))$

	descricao character varying(45)	qualidade character varying(2)	quantidade integer
1	Soja	1a	100
2	Laranja	1a	150
3	Feijão	2a	200
4	Cacau	1a	240
5	Trigo	1a	260
6	Feijão	1a	190
7	Laranja	2a	320
8	Cacau	1a	210
9	Uva	1a	125
10	Arroz	1a	150

Álgebra Relacional

□ Produto Cartesiano (\times):

$\pi_{\text{Descricao}, \text{Qualidade}, \text{Quantidade}}$
 $(\sigma_{\text{Produto.IdProduto} = \text{Estoque.IdProduto}}$
 $\wedge \text{Estoque.Quantidade} \geq 200 (\text{Produto} \times \text{Estoque}))$

	descricao character varying(45)	qualidade character varying(2)	quantidade integer
1	Feijão	2a	200
2	Cacau	1a	240
3	Trigo	1a	260
4	Laranja	2a	320
5	Cacau	1a	210

Álgebra Relacional

□ União (\cup):

- ❖ Requer que as **duas relações** fornecidas como argumento tenham o **mesmo esquema**.
- ❖ Resulta em uma nova relação, com o mesmo esquema, cujo conjunto de linhas é a união dos conjuntos de linhas das relações dadas como argumento.
- ❖ Exemplo:
$$\pi_{\text{IdFornecedor}, \text{Quantidade}} (\sigma_{\text{IdProduto} = 2} (\text{Estoque})) \cup$$
$$\pi_{\text{IdFornecedor}, \text{Quantidade}} (\sigma_{\text{IdProduto} = 2} (\text{Pedido}))$$

Álgebra Relacional

□ União (\cup):

❖ Resolvendo a **união**

Estoque

	fornecedor_idfornecedor integer	quantidade integer
1	1	150

Pedido

	fornecedor_idfornecedor integer	quantidade integer
1	10	110
2	4	130



	fornecedor_idfornecedor integer	quantidade integer
1	1	150
2	10	110
3	4	130

Álgebra Relacional

□ Diferença (-):

- ❖ Requer que as duas relações fornecidas como argumento tenham o **mesmo esquema**.
- ❖ Resulta em uma nova relação, com o mesmo esquema, cujo conjunto de linhas é o conjunto de linhas da primeira relação menos as linhas existentes na segunda.
- ❖ Exemplo:
 $\pi_{\text{IdFornecedor}}(\text{Estoque}) - \pi_{\text{IdFornecedor}}(\text{Pedido})$

Álgebra Relacional

- ❑ Diferença (-):
 - ❖ Resolvendo a **diferença**

Estoque

	fornecedor_idfornecedor integer
1	1
2	1
3	1
4	1
5	2
6	2
7	2
8	3
9	3
10	3

Pedido

	fornecedor_idfornecedor integer
1	2
2	10
3	4
4	7
5	3
6	6



	fornecedor_idfornecedor integer
1	1

Álgebra Relacional

□ Interseção (\cap):

- ❖ Resulta em uma nova relação, com o mesmo esquema, cujo conjunto de linhas é comum nas relações.
- ❖ Requer que as **duas relações** fornecidas como argumento **tenham o mesmo esquema**.
- ❖ Exemplo:

$$\pi_{\text{IdFornecedor, IdProduto}} (\sigma_{\text{IdEstoque} > 0} (\text{Estoque}) \cap \pi_{\text{IdFornecedor, IdProduto}} (\sigma_{\text{IdPedido} > 0} (\text{Pedido}))$$

Álgebra Relacional

- ❑ Interseção (\cap):
 - ❖ Resolvendo a **interseção**

Estoque

	fornecedor_idfornecedor integer	produto_idproduto integer
1	1	1
2	1	2
3	1	10
4	1	7
5	2	8
6	2	9
7	2	3
8	3	7
9	3	6
10	3	4

Pedido

	fornecedor_idfornecedor integer	produto_idproduto integer
1	2	1
2	10	2
3	4	2
4	7	3
5	3	3
6	6	4
7	3	7



	fornecedor_idfornecedor integer	produto_idproduto integer
1	3	7

Questões...



Álgebra Relacional

Parte II

Álgebra Relacional

□ Junção Natural (\bowtie):

- ❖ Combina operações de **seleção** e **produto cartesiano**.
- ❖ Retorna uma relação que é um produto cartesiano de duas relações.
- ❖ Exemplo: considere Produto **P** e Estoque **E**

π **Descrição, Qualidade, Quantidade** ($\sigma_{P.IdProduto = E.IdProduto}(Produto \times Estoque)$)

- ❖ Pode ser escrito da seguinte forma:

π **Descrição, Qualidade, Quantidade**
(Produto \bowtie Estoque)

Álgebra Relacional

□ Junção Natural (\bowtie):

	descricao character varying(45)	qualidade character varying(2)	quantidade integer
1	Soja	1a	100
2	Laranja	1a	150
3	Feijão	2a	200
4	Cacau	1a	240
5	Trigo	1a	260
6	Feijão	1a	190
7	Laranja	2a	320
8	Cacau	1a	210
9	Uva	1a	125
10	Arroz	1a	150

Álgebra Relacional

❑ Junção Natural (\bowtie):

- ❖ As **entidades** que participam da junção **devem se relacionar** (possuir atributo em comum).
- ❖ Exemplo: considere Produto **P** e Fornecedor **F**

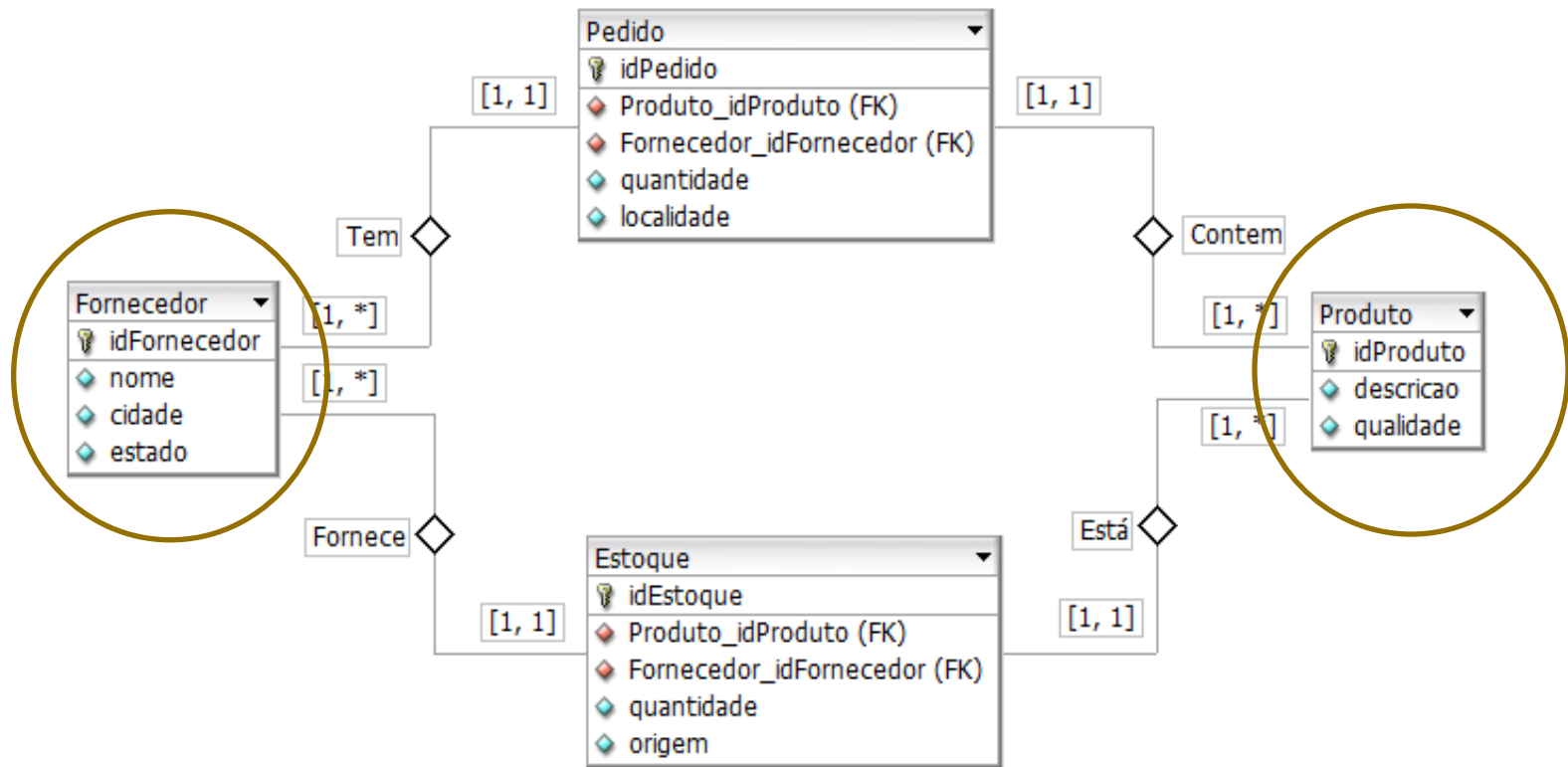


π **Nome, Cidade, UF** (Produto \bowtie Fornecedor)

Álgebra Relacional

Esquema Relacional:

❖ Controle de estoque



Álgebra Relacional

□ Junção Natural (\bowtie):

- ❖ É possível ter três entidades na mesma junção.
- ❖ Exemplo: considere Produto **P**, Estoque **E** e Pedido **Pe**

$\pi_{\text{Descrição, Qualidade, Origem, Localidade}}$
 $(\text{Produto} \bowtie \text{Estoque} \wedge \text{Produto} \bowtie \text{Pedido})$

Álgebra Relacional

□ Junção Natural (\bowtie):

$\pi_{\text{Descrição, Qualidade, Origem, Localidade}}$
(Produto \bowtie Estoque \wedge Produto \bowtie Pedido)

	descricao character varying(45)	qualidade character varying(2)	origem character varying(20)	localidade character varying(20)
1	Soja	1a	Natal	Campina Grande
2	Laranja	1a	Campina Grande	Parnamirim
3	Laranja	1a	Campina Grande	Mossoro
4	Laranja	2a	Juazeiro	Fortaleza
5	Laranja	2a	Juazeiro	Juazeiro
6	Arroz	1a	Fortaleza	Juazeiro
7	Cacau	1a	Natal	Macaiba
8	Cacau	1a	João Pessoa	Macaiba

Álgebra Relacional

□ Junção Externa :

❖ É uma seleção que não requer que os registros de uma tabela possuam **registros equivalentes** em outra. O **registro é mantido** na pseudo-tabela se não existe outro registro que lhe corresponda.

❖ Há três tipos:

- Junção externa à esquerda;
- Junção externa à direita;
- Junção externa completa.

Álgebra Relacional

□ Junção Externa à esquerda ($_ \mid \mathbf{X} \mid$):

- ❖ O resultado desta **seleção** sempre contém todos os registros da **tabela esquerda** (isto é, a primeira tabela mencionada na consulta), mesmo quando não exista registros **correspondentes na tabela direita**.
- ❖ Esta seleção retorna todos os **valores da tabela esquerda** com os valores da tabela direita correspondente, ou quando não há correspondência retorna um valor **NULL**.

Álgebra Relacional

□ Junção Externa à direita (\bowtie_X):

- ❖ Esta operação é inversa à anterior e retorna sempre todos os registros da **tabela à direita** (a segunda tabela mencionada na consulta), mesmo se não existir registro correspondente na **tabela à esquerda**.
- ❖ Nestes casos, o valor **NULL** é retornado quando não há correspondência.

Álgebra Relacional

□ Junção Externa total ($_ | \mathbf{X} | _$):

- ❖ Esta operação apresenta todos os dados das **tabelas à esquerda e à direita**, mesmo que não possuam correspondência em outra tabela.
- ❖ A tabela combinada possuirá assim todos **os registros de ambas as tabelas** e apresentará valores **nulos** para os registros sem correspondência.

Álgebra Relacional

□ Exemplificando ...

❖ Dadas as seguintes entidades/tabelas:

Empregado			
CPF	Nome	Rua	Cidade
874.444.234-30	Coyote	Toon	Hollywood
123.456.789-00	Rabbit	Tunnel	Carrotville
456.765.222-12	Williams	Seaview	Seattle
135.678.090-87	Smith	Resolver	Death Valley

Trabalhador_TI		
CPF	Agência	Salário
874.444.234-30	Mesa	1.500,00
123.456.789-00	Mesa	1.300,00
456.765.222-12	Redmond	1.500,00
001.987.346-34	Redmond	5.300,00

❖ Imagine a necessidade de gerar uma *junção natural* com todas as informações (CPF, Nome, Rua, Cidade, Agência e Salário).

$\pi_{\text{CPF, Nome, Rua, Cidade, Agencia, Salario}}$
(Empregado \bowtie Trabalhador_TI)

Álgebra Relacional

❏ Exemplificando ...

❖ Dadas as seguintes entidades/tabelas:

Empregado				Trabalhador_TI		
CPF	Nome	Rua	Cidade	CPF	Agência	Salário
→ 874.444.234-30	Coyote	Toon	Hollywood	→ 874.444.234-30	Mesa	1.500,00
→ 123.456.789-00	Rabbit	Tunnel	Carrotville	→ 123.456.789-00	Mesa	1.300,00
→ 456.765.222-12	Williams	Seaview	Seattle	→ 456.765.222-12	Redmond	1.500,00
135.678.090-87	Smith	Resolver	Death Valley	001.987.346-34	Redmond	5.300,00

$\pi_{CPF, Nome, Rua, Cidade, Agencia, Salario}$
(Empregado \bowtie Trabalhador_TI)

Álgebra Relacional

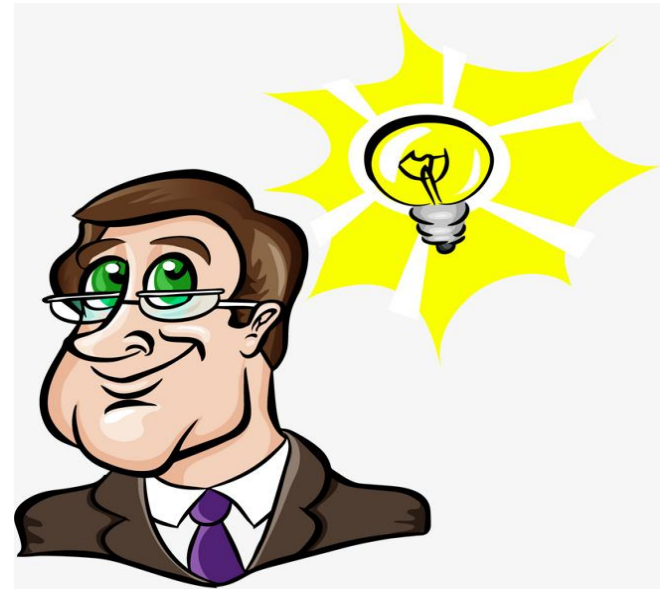
□ Exemplificando ...

❖ Resultado

	CPF	Nome	Rua	Cidade	Agência	Salário
→	874.444.234-30	Coyote	Toon	Hollywood	Mesa	1.500,00
→	123.456.789-00	Rabbit	Tunnel	Carrotville	Mesa	1.300,00
→	456.765.222-12	Williams	Seaview	Seattle	Redmond	1.500,00

Álgebra Relacional

- ❑ **Observação:** Note que os dados do empregado **Smith** referentes a Rua e Cidade foram perdidas.
- ❑ Por que?



Álgebra Relacional

- Podemos usar a operação de **junção externa** para evitar a perda de dados. Por exemplo usaremos a seguir a junção externa à esquerda.

$\pi_{CPF, Nome, Rua, Cidade, Agencia, Salario} (Empregado _|\mathbf{x}| Trabalhador_TI)$

Álgebra Relacional

□ Resultado

CPF	Nome	Rua	Cidade	Agência	Salário
874.444.234-30	Coyote	Toon	Hollywood	Mesa	1.500,00
123.456.789-00	Rabbit	Tunnel	Carrotville	Mesa	1.300,00
456.765.222-12	Williams	Seaview	Seattle	Redmond	1.500,00
135.678.090-87	Smith	Resolver	Death Valley	null	null

Álgebra Relacional

- ❑ **Outro caso:** e se quisermos manter os dados referentes à tabela da direita. O que fazer?

Álgebra Relacional

□ Resultado

$\pi_{CPF, Nome, Rua, Cidade, Agencia, Salario}$ (Empregado **|x|**_Trabalhador_TI)

CPF	Nome	Rua	Cidade	Agência	Salário
874.444.234-30	Coyote	Toon	Hollywood	Mesa	1.500,00
123.456.789-00	Rabbit	Tunnel	Carrotville	Mesa	1.300,00
456.765.222-12	Williams	Seaview	Seattle	Redmond	1.500,00
001.987.346-34	null	null	null	Redmond	5.300,00

Álgebra Relacional

❑ Situação pouco comum:

- ❖ Caso seja necessário mostrar todas as tuplas da relação da esquerda, mesmo que não tenham relação na direita, assim como mostrar todas as tuplas da relação da direita que não tenham relação na esquerda.
- ❖ O que fazer?

Álgebra Relacional

❑ Solução ...

π CPF, Nome, Rua, Cidade, Agencia,
Salario (Empregado _|**x**|_ Trabalhador_TI)

CPF	Nome	Rua	Cidade	Agência	Salário
874.444.234-30	Coyote	Toon	Hollywood	Mesa	1.500,00
123.456.789-00	Rabbit	Tunnel	Carrotville	Mesa	1.300,00
456.765.222-12	Williams	Seaview	Seattle	Redmond	1.500,00
135.678.090-87	Smith	Resolver	Death Valley	null	null
001.987.346-34	null	null	null	Redmond	5.300,00

Álgebra Relacional

❑ Divisão (\div):

- ❖ É usada nas consultas onde se emprega a frase “**para todos**”.
- ❖ Exemplo: encontrar os produtos que estão em todos os pedidos do fornecedor “Antônio”.

$\pi_{\text{IdProduto}, \text{Descrição}} (\text{Produto} \bowtie \text{Pedido})$

\div

$\pi_{\text{IdFornecedor}, \text{IdProduto}, \text{quantidade}} (\text{Pedido} \bowtie \text{Fornecedor} \wedge \sigma_{\text{Nome}=\text{'Antônio dos Santos'}})$

$R1 \div R2$

Álgebra Relacional

$R1 = \pi_{IdProduto, Descrição} (Produto \bowtie Pedido)$

	idproduto integer	descricao character varying(45)
1	1	Soja
2	2	Laranja
3	2	Laranja
4	3	Laranja
5	3	Laranja
6	4	Arroz
7	7	Cacau

$R2 = \pi_{IdFornecedor, IdProduto, Quantidade} (Pedido \bowtie Fornecedor \wedge \sigma_{Nome='Antônio dos Santos'})$

	fornecedor_idfornecedor integer	produto_idproduto integer	quantidade integer
1	3	3	220
2	3	7	125

Álgebra Relacional

Ou seja $R1 \div R2$:

	idproduto integer	descricao character varying(45)	quantidade integer	fornecedor_idfornecedor integer
1	3	Laranja	220	3
2	7	Cacau	125	3

Álgebra Relacional

❑ Funções Agregadas:

- ❖ São aquelas que, quando aplicadas, tomam uma coleção de valores e retornam um **valor simples** como resultado.
- ❖ Exemplos:
 - Sum
 - Avg
 - Max
 - Min
 - Count

Álgebra Relacional

❑ Funções Agregadas:

Trabalhador_TI		
CPF	Agência	Salário
874.444.234-30	Mesa	1.500,00
123.456.789-00	Mesa	1.300,00
456.765.222-12	Redmond	1.500,00
001.987.346-34	Redmond	5.300,00

Álgebra Relacional

□ Sum:

- ❖ Essa função retorna o somatório dos valores de uma seleção.
- ❖ Exemplo:

```
sum (salario ( $\sigma$ agencia = 'Mesa' (Trabalhador_TI)))
```

Agência	Salário
Mesa	1500
Mesa	1300

SUM
2800

Álgebra Relacional

□ Max:

- ❖ Essa função retorna o valor máximo de um atributo.
- ❖ Exemplo:

```
max (salario (Trabalhador_TI))
```

Salário
5.300,00

Questões...



Obrigado!!!

