

# **Bacharelado em Sistemas de Informação**

## **Banco de Dados**



# Álgebra Relacional

## Aula 2



# Álgebra Relacional

## Operações da álgebra relacional

As operações da álgebra relacional são normalmente divididas em dois grupos:

- \* conjunto de operações da teoria de conjuntos: UNION, INTERSECTION, CARTESIAN PRODUCT e DIFFERENCE

- \* as operações desenvolvidas especificamente para bases de dados relacionais, tais como: SELECT, PROJECT, JOIN e DIVISÃO.



## Operador: SELECT

SELECT: é usado para selecionar um subconjunto de tuplas de uma relação as quais devem satisfazer uma **condição de seleção**.

$$\sigma \text{ <condição de seleção> ( <nome da relação> )}$$

Onde:

- o símbolo  $\sigma$  : é usado para denotar o operador SELECT



## Operador: SELECT

$\sigma$  <condição de seleção> (<nome da relação>)

Onde:

**<condição de seleção>** é uma expressão booleana composta da seguinte forma:

**<nome de atributo>**

nome de um atributo da relação.

**<operador de comparação>**

Um dos operadores relacionais

Pode ser combinado com AND, OR e NOT

**<valor constante>**

Valor constante



# Álgebra Relacional

## Operador: SELECT

$\sigma$  <condição de seleção> (<nome da relação>)

Exemplo: Selecione na relação de nome R\_a, apenas as tuplas cujo o valor do atributo “A” seja igual “1”

R\_a

A	B	C
1	c	d
2	d	f
1	f	l
4	g	m

$\sigma_{(A=1)}(R_a)$

A	B	C
1	c	d
1	f	l



# Álgebra Relacional

## Operador: SELECT

Um exemplo mais prático

$\sigma$  (curso='Informática') (Alunos)

```
Alunos = {Matr, Nome, Curso}
{
<001, José, Engenharia>,
<012, João, Informática>,
<021, Pedro, Direito>,
<034, José, Informática>,
<045, Carol, Direito>
}
```

```
Resultado = {Matr, Nome, Curso}
{
<012, João, Informática>,
<034, José, Informática>
}
```



# Álgebra Relacional

## Operador: SELECT

A <condição de seleção> pode ser compostas. Neste caso, utilizamos as cláusulas em conjunto com os operadores lógicos {AND, OR NOT}

Exemplo: Selecione na relação de nome R\_a, as tuplas cujo o valor do atributo “A” e “C” sejam iguais a “1”

R\_a

A	B	C
1	c	d
2	d	f
1	f	1
4	g	m

$\sigma_{(A=1 \text{ AND } C=1)}(R_a)$

A	B	C
1	f	1





# Álgebra Relacional

## Operador: SELECT

Um exemplo mais prático

$\sigma$  (curso='Informática' OR nome='João') (Alunos)

```
Alunos = {Matr, Nome, Curso}
{
<001, José, Engenharia>,
<012, João, Informática>,
<021, Pedro, Direito>,
<034, José, Informática>,
<045, Carol, Direito>
}
```

```
Resultado = {Matr, Nome, Curso}
{
<012, João, Informática>,
<034, José, Informática>
}
```



## Operador: SELECT

Atenção:

- O operador SELECT é **UNÁRIO**, isto é, ele é aplicado somente a uma relação. Assim, não pode ser usado para selecionar tuplas de mais de uma relação.
- O operador SELECT é **COMUTATIVO**, isto é:  
$$\sigma \langle \text{cond1} \rangle ( \sigma \langle \text{cond2} \rangle (R) ) = \sigma \langle \text{cond2} \rangle ( \sigma \langle \text{cond1} \rangle (R) )$$
- A operação de seleção é aplicada individualmente para cada tupla. O grau da relação resultante é a mesma que a relação original.
- O número de tuplas da relação resultante é sempre menor ou igual ao número de tuplas da relação original.



## Operador: PROJECT

PROJECT deve ser utilizado se existir o interesse em projetar certos atributos da relação, pois ele simplesmente apresenta as tuplas da relação com as colunas especificadas, descartando os dados das outras colunas não solicitadas.

A forma geral do operador PROJECT é:

$$\pi \langle \text{lista de atributos} \rangle (\langle \text{nome da relação} \rangle)$$

Onde:

- o símbolo  $\pi$  : é usado para denotar o operador PROJECT



# Álgebra Relacional

## Operador: PROJECT

A operação PROJECT remove implicitamente quaisquer tuplas duplicadas, tal que o resultado da operação PROJECT seja um conjunto de tuplas e assim, uma relação válida.

Exemplo: Projeção das colunas A e C da relação R\_a

R\_a

A	B	C
1	e	d
2	d	f
1	f	l
4	g	m

$\pi_{A,C}(R_a)$

A	C
1	d
2	f
1	l
4	m



## Operador: PROJECT

Atenção:

- O número de tuplas na relação resultante sempre será igual ou menor que a quantidade de tuplas na relação original.
- A comutatividade **NÃO** é válida para PROJECT.
- Note-se que:

$$\pi \langle \text{lista1} \rangle \left( \pi \langle \text{lista2} \rangle (R) \right) = \pi \langle \text{lista1} \rangle (R)$$

somente se,  $\langle \text{lista2} \rangle$  contenha os atributos de  $\langle \text{lista1} \rangle$ ;



## Operador: PROJECT

Um exemplo mais prático

$\pi$  nome,curso (Alunos)

```
Alunos = {Matr, Nome, Curso}
{
<001, José, Engenharia>,
<012, João, Informática>,
<021, Pedro, Direito>,
<034, José, Informática>,
<045, Carol, Direito>
}
```

```
Alunos = {Nome, Curso}
{
<José, Engenharia>,
<João, Informática>,
<Pedro, Direito>,
<José, Informática>,
<Carol, Direito>
}
```



## Operador: JOIN

JOIN: é usado para combinar tuplas relacionadas de relações em uma única tupla, ou seja, identifica as tuplas relacionadas (relacionamentos) entre as Relações (tabelas).

O operador Join é representado pelo símbolo:  $\bowtie$

A forma geral da operação JOIN sobre duas relações Rel\_A e Rel\_B é:

$$\text{Rel\_A} \bowtie \langle \text{condição da junção} \rangle \text{Rel\_B}.$$



# Álgebra Relacional

## Operador: JOIN

Atenção:

O Join realiza uma única operação e é correspondente à sequência:

$$\text{Rel\_A} \square \langle \text{condição da junção} \rangle \text{Rel\_B}$$

CARTESIAN PRODUCT seguido de SELECT.

$$R\_C \leftarrow \text{Rel\_A} \chi \text{Rel\_B}$$

$$R \leftarrow \sigma \langle \text{condição da junção} \rangle (\text{Rel\_C})$$





# Álgebra Relacional

## Operador: JOIN

[INNER] JOIN: Esse é um formato comum de join, que retorna dados apenas quando as duas relações possuem atributos correspondentes. Ou seja, aparecem no resultado somente as tuplas de Rel\_A que têm tuplas correspondentes em Rel\_B (e vice-versa)

Exemplo:

$$\text{Rel\_A} \bowtie_{A=B} \text{Rel\_B}$$


# Álgebra Relacional

## Operador: JOIN

Um exemplo prático [INNER] JOIN:

### Alunos

```
Alunos = {Matr, Nome, CodCurso}
{
  <001, José, 1>,
  <012, João, 3>,
  <021, Pedro, 2>,
  <034, José, 3>,
  <045, Carol, 2>
}
```

### Cursos

```
Cursos = {Cod, Nome, Departamento}
{
  <1, Engenharia, DepIII>,
  <2, Direito, DepIV>,
  <3, Informática, DepII>,
  <4, Humanas, DepI>,
  <5, Exatas, DepV>
}
```



# Álgebra Relacional

## Operador: JOIN

Um exemplo prático [INNER] JOIN:

Alunos  $\square$  CodCurso=Cod Cursos

```
Resultado = {Matr, Nome, CodCurso, Cod, Nome, Departamento }
{
  <001,  José ,  1,  1,  Engenharia, DepIII >,
  <012,  João,  3,  3,  Informática, DepII  >,
  <021,  Pedro, 2,  2,  Direito,     DepIV  >,
  <034,  José,  3,  3,  Informática, DepII  >,
  <045,  Carol, 2,  2,  Direito,     DepIV  >
}
```



# Álgebra Relacional

## Operador: JOIN

EQUIJOIN JOIN: condição join que contém apenas a igualdade.

Exemplo:

$$\text{Rel\_A} \bowtie_{A=B} \text{Rel\_B}$$

NATURAL JOIN: é um equijoin onde os atributos utilizados na comparação possuem o mesmo nome.

Exemplo:

$$\text{Rel\_A} \bowtie_{A=A} \text{Rel\_B} \quad \text{ou} \quad \text{Rel\_A} * \text{Rel\_B}$$



# Álgebra Relacional

## Operador: JOIN

Um exemplo prático [NATURAL] JOIN:

### Alunos

```
Alunos = {Matr, Nome, CodCurso}
{
  <001, José, 1>,
  <012, João, 3>,
  <021, Pedro, 2>,
  <034, José, 3>,
  <045, Carol, 2>
}
```

### Cursos

```
Cursos = {CodCurso, Nome, Departamento}
{
  <1, Engenharia, DepIII>,
  <2, Direito, DepIV>,
  <3, Informática, DepII>,
  <4, Humanas, DepI>,
  <5, Exatas, DepV>
}
```



# Álgebra Relacional

## Operador: JOIN

Um exemplo prático [NATURAL] JOIN:

Alunos \* Cursos

```
Resultado = {Matr, Nome, CodCurso, Nome, Departamento }  
{  
  <001,  José ,  1,      Engenharia, DepIII  >,  
  <012,  João,   3,      Informática, DepII   >,  
  <021,  Pedro,  2,      Direito,      DepIV   >,  
  <034,  José,   3,      Informática, DepII   >,  
  <045,  Carol,  2,      Direito,      DepIV   >  
}
```



## Operador: JOIN

### Junção Interna X Junção Externa

Nas operações de junção anteriores, vimos que todas as tuplas que não atendiam a condição de junção foram removidas da relação resultante. As operações deste tipo são designadas por operações de junção interna.

Um outro conjunto de operações permite manter parte ou a totalidade das tuplas das relações a combinar, mesmo que estas não satisfaçam a condição de junção. As operações deste tipo são denominadas por operações de junção externa.

Veremos os três tipos de operações de junção externa.



# Álgebra Relacional

## Operador: JOIN

LEFT [OUTER] JOIN

$Rel\_A \bowtie Rel\_B$

Junção externa à esquerda:

- mantém cada tupla de  $Rel\_A$  na tabela de junção
- preenche com valores nulos as tuplas de  $Rel\_B$  que não correspondem à coluna de junção em  $Rel\_A$





# Álgebra Relacional

## Operador: JOIN

Um exemplo prático LEFT [OUTER] JOIN:

### Alunos

```
Alunos = {Matr, Nome, CodCurso}
{
  <001,  José,  1>,
  <012,  João,  3>,
  <021,  Pedro, 6>,
  <034,  José,  3>,
  <045,  Carol, 4>
}
```

### Cursos

```
Cursos = {CodCurso, Nome, Departamento}
{
  <1, Engenharia, DepIII>,
  <2, Direito,    DepIV>,
  <3, Informática, DepII>,
  <4, Humanas,    DepI>,
  <5, Exatas,     DepV>
}
```



# Álgebra Relacional

## Operador: JOIN

Um exemplo prático LEFT [OUTER] JOIN:

### Alunos $\bowtie$ Cursos

Resultado = {Matr, Nome, CodCurso, CodCurso, Nome, Departamento }  
{  
    <001, José, 1, 1, Engenharia, DepIII >,  
    <012, João, 3, 3, Informática, DepII >,  
    <021, Pedro, 6, Null, Null, Null >,  
    <034, José, 3, 3, Informática, DepII >,  
    <045, Carol, 4, 4, Humanas, DepI >  
}



# Álgebra Relacional

## Operador: JOIN

RIGHT [OUTER] JOIN

$\text{Rel\_A} \bowtie \text{Rel\_B}$

Junção externa à direita:

- mantém cada tupla de Rel\_B na tabela de junção
- preenche com valores nulos as tuplas de Rel\_A que não correspondem à coluna de junção em Rel\_B



# Álgebra Relacional

## Operador: JOIN

Um exemplo prático RIGHT [OUTER] JOIN:

### Alunos

```
Alunos = {Matr, Nome, CodCurso}
{
  <001,  José,   1>,
  <012,  João,   3>,
  <021,  Pedro,  6>,
  <034,  José,   3>,
  <045,  Carol,  4>
}
```

### Cursos

```
Cursos = {CodCurso, Nome, Departamento}
{
  <1, Engenharia, DepIII>,
  <2, Direito,    DepIV>,
  <3, Informática, DepII>,
  <4, Humanas,    DepI>,
  <5, Exatas,     DepV>
}
```



# Álgebra Relacional

## Operador: JOIN

Um exemplo prático RIGHT [OUTER] JOIN:

Alunos  $\bowtie$  Cursos

Resultado = {Matr, Nome, CodCurso, CodCurso, Nome, Departamento }  
{  
    <001, José, 1, 1, Engenharia, DepIII>,  
    <Null, Null, Null, 2, Direito, DepIV>,  
    <012, João, 3, 3, Informática, DepII>,  
    <034, José, 3, 3, Informática, DepII>,  
    <045, Carol, 4, 4, Humanas, DepI>,  
    <Null, Null, Null, 5, Exatas, DepV>  
}



# Álgebra Relacional

## Operador: JOIN

FULL [OUTER] JOIN

$Rel\_A \bowtie Rel\_B$

Junção externa completa:

- mantém cada tupla de  $Rel\_A$  e de  $Rel\_B$  na tabela de junção
- preenche com valores nulos as tuplas que não correspondem à coluna de junção



# Álgebra Relacional

## Operador: JOIN

Um exemplo prático FULL [OUTER] JOIN:

### Alunos

```
Alunos = {Matr, Nome, CodCurso}
{
  <001,  José,  1>,
  <012,  João,  3>,
  <021,  Pedro, 6>,
  <034,  José,  3>,
  <045,  Carol, 4>
}
```

### Cursos

```
Cursos = {CodCurso, Nome, Departamento}
{
  <1, Engenharia, DepIII>,
  <2, Direito,    DepIV>,
  <3, Informática, DepII>,
  <4, Humanas,    DepI>,
  <5, Exatas,     DepV>
}
```



# Álgebra Relacional

## Operador: JOIN

Um exemplo prático FULL [OUTER] JOIN:

### Alunos $\bowtie$ Cursos

Resultado = {Matr, Nome, CodCurso, CodCurso, Nome, Departamento }  
{  
    <001, José, 1, 1, Engenharia, DepIII >,  
    <Null, Null, Null, 2, Direito, DepIV >,  
    <012, João, 3, 3, Informática, DepII >,  
    <034, José, 3, 3, Informática, DepII >,  
    <045, Carol, 4, 4, Humanas, DepI >,  
    <Null, Null, Null, 5, Exatas, DepV >,  
    <021, Pedro, 6, Null, Null, Null >  
}





## Operador: DIVISION

DIVISION: utilizada quando desejamos selecionar os valores de um atributo de Rel\_A que fazem referência a todos os valores de um atributo de Rel\_B. É uma operação específica para consultas que contenham o termo “para todos” ou “em todos”

O operador Division é representado pelo símbolo:  $\div$

<nome da relação>  $\div$  <nome da relação>



# Álgebra Relacional

## Operador: DIVISION

Um exemplo prático:

### Clientes

```
Clientes = {CodC, Nome}
{
  <001, José >,
  <002, Carol >,
  <003, Maria >
}
```

### Rel\_Cli\_Ag

```
R = {CodC, CodA, Saldo}
{
  <001, E, 10.000>,
  <002, E, 24.000>,
  <003, A, 12.000>,
  <001, C, 300>,
  <002, A, 32.140>,
  <002, C, 150>,
  <003, E, 3.256>
}
```

### Agências

```
Agencias = {CodA, Nome, End}
{
  <E, Estilo, Rua 1 >,
  <C, Comum, Prça A >,
  <A, Agro, Av. B >
}
```



# Álgebra Relacional

## Operador: DIVISION

Um exemplo prático:

Clientes  $\div$  Agências

Resultado

```
{  
    <002, Carol >  
}
```

