

Banco de Datos I

Álgebra Relacional

Modelo Relacional - Manipulação

- Duas categorias de linguagens
 - **Formais** : álgebra relacional e cálculo relacional
 - **Alto Nível** (Comerciais) - baseadas nas linguagens formais - SQL
- Linguagens Formais – Características
 - Orientadas a Conjuntos
 - Linguagens de base: linguagens relacionais devem ter no mínimo um poder de expressão equivalente ao de uma linguagem formal
 - Fechamento: resultados de consultas são relações

Álgebra Relacional

- Álgebra desenvolvida para **descrever operações sobre uma base de dados relacional**
- O **conjunto de objetos** são as **relações**
- Possui operadores para consulta e alteração de relações
- Linguagem procedural
 - uma expressão na álgebra define uma execução sequencial de operadores
 - a execução de cada operador produz uma relação
- Os **operadores** da álgebra relacional **recebem uma ou mais relações de entrada** e geram uma **nova relação de saída**

Álgebra Relacional

- É uma **linguagem formal** para acesso ao banco de dados;
- **Base para a SQL (*Structured Query Language*)**;
- Segundo (CONNOLLY, 2010), é uma **linguagem procedural de alto nível**;
- Usada para **construir uma nova relação a partir de relações existentes**;
- Uma **expressão realiza operações sobre relações** (tabelas) e o **resultado é uma nova relação** (tabela).
 - Isso **permite que as operações sejam aninhadas**, ou seja, o resultado de uma operação pode ser aplicado como entrada para outra e assim sucessivamente, da mesma forma como nas expressões aritméticas.
- As operações são:
 - **Seleção, Projeção, Produto Cartesiano, União, Intersecção, Diferença e Junção Natural.**

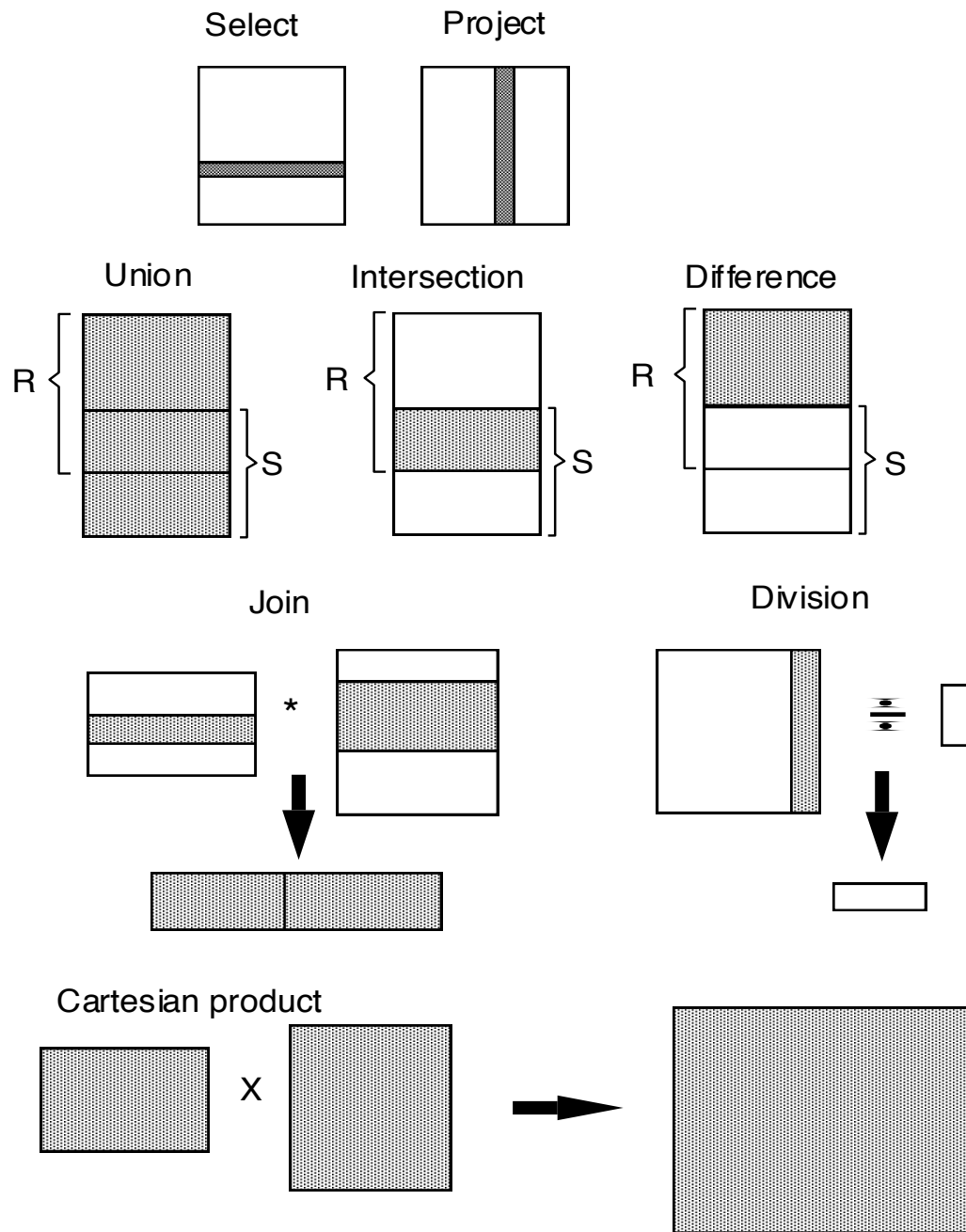
Álgebra Relacional

- Porque aprender:
 - Compreender álgebra relacional é **base** para SQL
 - Não há SGBD que implementa álgebra diretamente como DML (*Data Manipulation Language*), mas **SQL incorpora** conceitos de **Álgebra Relacional**
 - Algoritmos de **otimização de consulta** são definidos sobre Álgebra Relacional
 - Usados internamente no SGBD

Álgebra Relacional

- Operadores sobre **conjuntos** (uma tabela é um conjunto de linhas):
 - União
 - Interseção
 - Diferença
 - Produto Cartesiano
- Operadores específicos da **Álgebra Relacional**:
 - Seleção
 - Projeção
 - Junção
 - Divisão

Operações



Esquema Relacional: Exemplo

Funcionario

id_func	nome_func	id_dep	id_categ
200	Pedro	1	1
201	Paulo	1	2
202	Maria	1	1
203	Ana	2	2

Categoria

id_categ	salario
1	3.000
2	3.500

Projeto

id_proj	nome	id_dep
A	Projeto A	1
B	Projeto B	2

ProjetoFuncionario

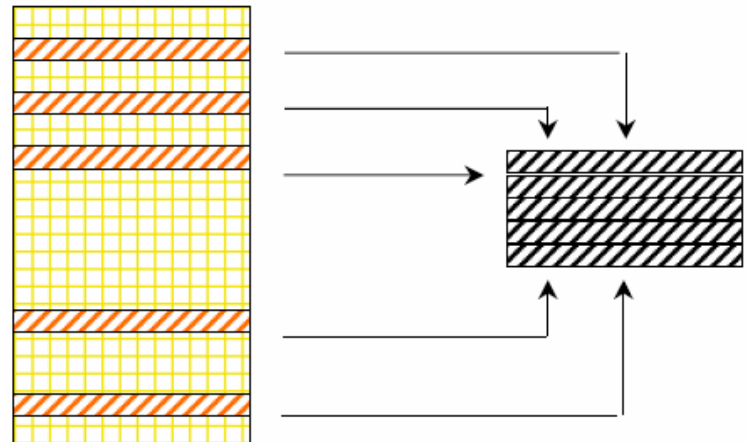
id_proj	id_func	data_ini	data_fim
A	200	01/01/2011	atual
A	201	01/01/2011	atual
A	202	01/02/2010	18/02/2015
B	203	15/02/2012	15/02/2015

Departamento

id_dep	nome
1	Contábil
2	Vendas

Seleção (σ)

- Retorna **tuplas** que satisfazem uma **condição**
- Age como um **filtro** que mantém somente as tuplas que satisfazem a condição
- O resultado:
 - É uma relação que contém as tuplas que satisfazem a condição
 - Possui os mesmos atributos da relação de entrada



Seleção (σ)

- Sintaxe: σ <condição> (<R>)
- **Sigma(σ)**: é o símbolo que representa a seleção
- <condição> é uma expressão booleana que envolve literais e valores de atributos da relação
 - CLAUSULAS:

<nome do atributo> <operador de comparação> <valor constante> OU
<nome do atributo> <operador de comparação> <nome do atributo>

 - Nome do atributo: é um atributo de R;
 - Operador de comparação: =, <, <=, >, >=, <>
 - Valor constante: é um valor do domínio do atributo
 - Podem ser ligadas pelos operadores **OR** (\vee), **AND** (\wedge) e **NOT** (\neg)
- <R> é o nome de uma relação (tabela) ou uma *expressão da álgebra relacional* de onde as tuplas serão buscadas

Seleção (σ) - Exemplo

- Selecionar os funcionários cuja categoria salarial tenha valor igual a 1.

$\sigma_{\text{categoria} = 1}$ (Funcionario)

Funcionario

id_func	nome_func	id_dep	id_categ
200	Pedro	1	1
201	Paulo	1	2
202	Maria	1	1
203	Ana	2	2

Resultado

id_func	nome_func	id_dep	id_categ
200	Pedro	1	1
202	Maria	1	1

Seleção (σ) - Exemplo

- Selecionar os funcionários cuja categoria salarial seja igual a 2 e que sejam do Departamento de Vendas

$\sigma_{id_categ = 2 \text{ AND } id_dep = 2}$ (Funcionario)

Funcionario

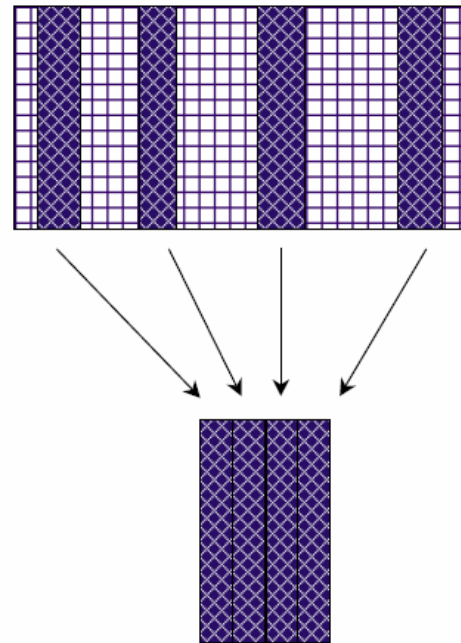
id_func	nome_func	id_dep	id_categ
200	Pedro	1	1
201	Paulo	1	2
202	Maria	1	1
203	Ana	2	2

Resultado

id_func	nome_func	id_dep	id_categ
203	Ana	2	2

Projeção (π)

- Recupera determinados atributos de uma relação (tabela), produzindo subconjunto vertical da relação.
- Retorna um ou mais **atributos** de interesse
- O resultado é uma relação que **contém apenas as colunas selecionadas**.



Projeção (π)

- Sintaxe:

π <lista de atributos> (<R>)

onde:

- π (Pi) é o símbolo do operador de projeção.
- **<lista de atributos>** é uma lista que contém nomes de colunas de uma ou mais relações.
- **<R>** é o nome da relação ou uma expressão da álgebra relacional de onde a lista de atributos será buscada

Projeção (π) – Exemplo

- Selecionar o código e o nome de todos os funcionários

π id_func, nome (Funcionario)

Funcionario

id_func	nome_func	id_dep	id_categ
200	Pedro	1	1
201	Paulo	1	2
202	Maria	1	1
203	Ana	2	2

Resultado

id_func	nome_func
200	Pedro
201	Paulo
202	Maria
203	Ana

Projeção e Seleção

- Operadores diferentes podem ser aninhados
 - Exemplo: Selecionar o código e nome dos funcionários do departamento contábil

$\pi_{id_func, nome_func} (\sigma_{id_dep = 1} (Funcionario))$

Funcionario

id_func	nome_func	id_dep	id_categ
200	Pedro	1	1
201	Paulo	1	2
202	Maria	1	1
203	Ana	2	2

Resultado

id_func	nome_func
200	Pedro
201	Paulo
202	Maria

Operações

Operações - Teoria dos Conjuntos

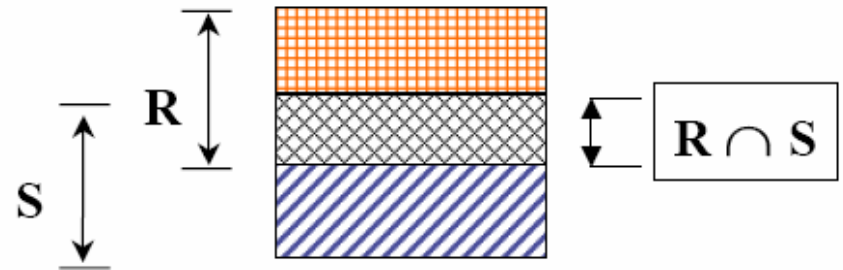
- A álgebra relacional utiliza **4 operadores** da teoria dos conjuntos:
 - União, Intersecção, Diferença e Produto Cartesiano
- Todos os operadores utilizam ao menos **DUAS** relações
- As relações devem ser compatíveis:
 - possuir o mesmo número de atributos
 - o **domínio** da i-ésima coluna de uma relação deve ser idêntico ao domínio da i-ésima coluna da outra relação
- Quando os nomes dos atributos forem diferentes, adota-se a convenção de usar os nomes dos atributos da primeira relação

Intersecção (\cap)

- É um **operador binário** (envolve duas relações como operandos de entrada) e **cria uma relação com apenas as tuplas que estão presentes nas duas relações envolvidas**.
- **Tuplas duplicadas são removidas do resultado**.
- As duas **relações devem ser compatíveis**, ou seja, devem ter o mesmo número de atributos, com aqueles de mesmo domínio dispostos nas mesmas posições.
- Notação: **<relação 1> \cap <relação 2>**
 - Tanto relação1 quanto relação2 podem ser tabelas originais do banco de dados ou expressões da Álgebra Relacional.

Intersecção (\cap)

- Retorna uma relação com as tuplas comuns a R e S
- Notação: $R \cap S$



R

x	y	z
1	1	1
1	2	2
2	2	3
3	1	1

S

x	y	z
1	1	1
1	2	1
3	1	1


$R \cap S$

x	y	z
1	1	1
3	1	1

Intersecção (\cap) - Exemplo

- Recuperar a lista de descrições de projetos que são também descrições de departamentos.

$\pi_{\text{Descrição}}(\text{Projeto}) \cap \pi_{\text{Descrição}}(\text{Departamento})$



A avaliação desta expressão ocorre da seguinte forma: primeiro é executada a projeção sobre Projeto: $\pi_{\text{Descrição}}(\text{Projeto})$; em seguida, a Projeção sobre Departamento: $\pi_{\text{Descrição}}(\text{Departamento})$; e, por fim, é realizada a Intersecção.

- Para realizar esta consulta pode-se usar o operador de Intersecção, entretanto, antes de aplicar a operação, é preciso tornar os dois operandos compatíveis.
- Por isso, **as projeções são realizadas em cada tabela**, Projeto e Departamento.
- A seguir um exemplo passo a passo.

Intersecção (\cap) – Exemplo passo a passo

<ul style="list-style-type: none">Resultado de $\pi_{\text{Descrição}}$ (Projeto)	<table><tr><th>Descrição</th></tr><tr><td>Projeto de BD</td></tr><tr><td>Implantação de ERP</td></tr><tr><td>Contabilidade empresarial</td></tr></table>	Descrição	Projeto de BD	Implantação de ERP	Contabilidade empresarial
Descrição					
Projeto de BD					
Implantação de ERP					
Contabilidade empresarial					
<ul style="list-style-type: none">Resultado de $\pi_{\text{Descrição}}$ (Departamento)	<table><tr><th>Descrição</th></tr><tr><td>Contabilidade empresarial</td></tr><tr><td>Marketing e vendas</td></tr><tr><td>Capital Humano e treinamento</td></tr></table>	Descrição	Contabilidade empresarial	Marketing e vendas	Capital Humano e treinamento
Descrição					
Contabilidade empresarial					
Marketing e vendas					
Capital Humano e treinamento					
<ul style="list-style-type: none">Resultado de $\pi_{\text{Descrição}}$ (Projeto) \cap $\pi_{\text{Descrição}}$ (Departamento)	<table><tr><th>Descrição</th></tr><tr><td>Contabilidade empresarial</td></tr></table>	Descrição	Contabilidade empresarial		
Descrição					
Contabilidade empresarial					

União (\cup)

- É um **operador binário** (envolve duas relações como operandos de entrada) e **cria uma relação que contém a união de todas as tuplas das duas relações** envolvidas;
- **Tuplas duplicadas são removidas do resultado;**
- As duas **relações devem ser compatíveis**, ou seja, devem ter o mesmo número de atributos, com aqueles de mesmo domínio dispostos nas mesmas posições;
- Notação: **<relação 1> \cup <relação 2>**
 - Tanto relação1 quanto relação2 podem ser tabelas originais do banco de dados ou expressões da 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.
- Retorna a **união das tuplas** de duas relações R e S
- **Eliminação automática de duplicatas**
- Notação: $R \cup S$

R

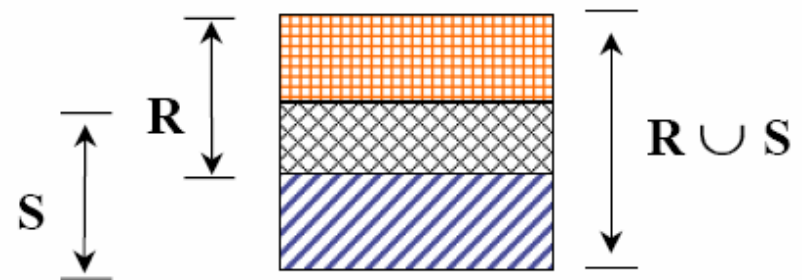
x	y	z
1	1	1
1	2	2
2	2	3
3	1	1

S

x	y	z
1	1	1
1	2	1
1	2	3

$R \cup S$

x	y	z
1	1	1
1	2	1
1	2	2
1	2	3
2	2	3
3	1	1



União (\cup)

- Exemplo:
 - Buscar o nome e o CPF dos médicos e dos pacientes cadastrados no hospital

$$\pi_{\text{nome, CPF (Medico)}} \cup \pi_{\text{nome, CPF (Paciente)}}$$

Exemplos:

- Recuperar a lista de descrições de projetos e de departamentos empregados pela empresa.

$\pi_{\text{Descrição}}(\text{Projeto}) \cup \pi_{\text{Descrição}}(\text{Departamento})$

- Para realizar esta consulta pode-se usar o operador de União, entretanto, antes de aplicar a operação, é preciso tornar os dois operandos compatíveis.
- Por isso, as projeções são realizadas em cada tabela, Projeto e Departamento.
- A seguir um exemplo passo a passo.

<ul style="list-style-type: none">Resultado de $\pi_{\text{Descrição}}$ (Projeto)	<table><tr><th>Descrição</th></tr><tr><td>Projeto de BD</td></tr><tr><td>Implantação de ERP</td></tr></table>	Descrição	Projeto de BD	Implantação de ERP			
Descrição							
Projeto de BD							
Implantação de ERP							
<ul style="list-style-type: none">Resultado de $\pi_{\text{Descrição}}$ (Departamento)	<table><tr><th>Descrição</th></tr><tr><td>Contabilidade empresarial</td></tr><tr><td>Marketing e vendas</td></tr><tr><td>Capital Humano e treinamento</td></tr></table>	Descrição	Contabilidade empresarial	Marketing e vendas	Capital Humano e treinamento		
Descrição							
Contabilidade empresarial							
Marketing e vendas							
Capital Humano e treinamento							
<ul style="list-style-type: none">Resultado de $\pi_{\text{Descrição}}$ (Projeto) U $\pi_{\text{Descrição}}$ (Departamento)	<table><tr><th>Descrição</th></tr><tr><td>Projeto de BD</td></tr><tr><td>Implantação de ERP</td></tr><tr><td>Contabilidade empresarial</td></tr><tr><td>Marketing e vendas</td></tr><tr><td>Capital Humano e treinamento</td></tr></table>	Descrição	Projeto de BD	Implantação de ERP	Contabilidade empresarial	Marketing e vendas	Capital Humano e treinamento
Descrição							
Projeto de BD							
Implantação de ERP							
Contabilidade empresarial							
Marketing e vendas							
Capital Humano e treinamento							

Diferença (-)

- O operador de **Diferença** é um **operador binário** (envolve duas relações como operandos de entrada) e **cria uma relação com as tuplas que fazem parte da relação1, mas não fazem parte da relação2.**
- **Tuplas duplicadas são removidas do resultado.**
- As duas **relações devem ser compatíveis**, ou seja, devem ter o mesmo número de atributos, com aqueles de mesmo domínio dispostos nas mesmas posições.
- Notação: **<relação 1> – <relação 2>**
 - Tanto relação1 quanto relação2 podem ser tabelas originais do banco de dados ou expressões da 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**.

Diferença (-)

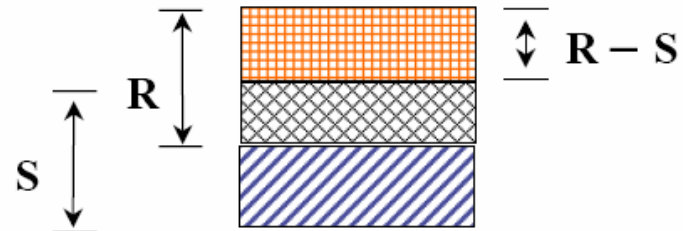
- Retorna as tuplas **presentes** em R e **ausentes** em S
- Notação:

$$R - S$$

R		
x	y	z
1	1	1
1	2	2
2	2	3
3	1	1

S		
x	y	z
1	1	1
1	2	1
3	1	1

R-S		
x	y	z
1	2	2
2	2	3



Diferença (-)

- Exemplo:
 - Buscar o número dos ambulatórios onde nenhum médico presta atendimento

$$\pi \text{ numeroA (Ambulatorio)} - \pi \text{ numeroA (Medico)}$$

Diferença (-) - Exemplo:

- Recuperar a lista de descrições de projetos que não são descrições de departamentos

$$\pi_{\text{Descrição}}(\text{Projeto}) - \pi_{\text{Descrição}}(\text{Departamento})$$

A avaliação desta expressão ocorre da seguinte forma:

primeiro é executada a projeção sobre Projeto:

$\pi_{\text{Descrição}}(\text{Projeto})$;

em seguida, a Projeção sobre Departamento:

$\pi_{\text{Descrição}}(\text{Departamento})$;

e, por fim, é realizada a Diferença.

- Para realizar esta consulta pode-se usar o operador de Intersecção, entretanto, antes de aplicar a operação, é preciso tornar os dois operandos compatíveis.
- Por isso, as projeções são realizadas em cada tabela, Projeto e Departamento.
- A seguir um exemplo passo a passo.

Diferença (-) - Exemplo:

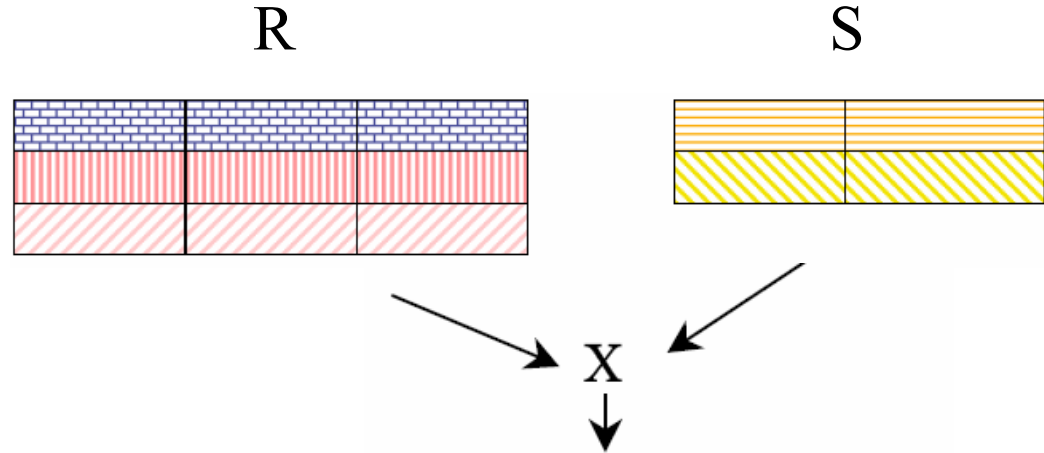
<ul style="list-style-type: none">Resultado de $\pi_{\text{Descrição}}$ (Projeto)	<table><tr><th>Descrição</th></tr><tr><td>Projeto de BD</td></tr><tr><td>Implantação de ERP</td></tr><tr><td>Contabilidade empresarial</td></tr></table>	Descrição	Projeto de BD	Implantação de ERP	Contabilidade empresarial
Descrição					
Projeto de BD					
Implantação de ERP					
Contabilidade empresarial					
<ul style="list-style-type: none">Resultado de $\pi_{\text{Descrição}}$ (Departamento)	<table><tr><th>Descrição</th></tr><tr><td>Contabilidade empresarial</td></tr><tr><td>Marketing e vendas</td></tr><tr><td>Capital Humano e treinamento</td></tr></table>	Descrição	Contabilidade empresarial	Marketing e vendas	Capital Humano e treinamento
Descrição					
Contabilidade empresarial					
Marketing e vendas					
Capital Humano e treinamento					
<ul style="list-style-type: none">Resultado de $\pi_{\text{Descrição}}$ (Projeto) - $\pi_{\text{Descrição}}$ (Departamento)	<table><tr><th>Descrição</th></tr><tr><td>Projeto de BD</td></tr><tr><td>Implantação de ERP</td></tr></table>	Descrição	Projeto de BD	Implantação de ERP	
Descrição					
Projeto de BD					
Implantação de ERP					

Produto Cartesiano (x)

- Multiplica duas relações para definir uma nova que consiste de todas as combinações das tuplas das duas relações.
- Retorna **todas as combinações de tuplas** de duas relações R e S
- O resultado é uma **relação cujas tuplas são a combinação das tuplas das relações** R e S, tomando-se uma tupla de R e concatenando-a com uma tupla de S
- Notação: **<Relação 1> X <Relação 2>**, ou seja,
 $R \times S$

Produto Cartesiano (x)

Total de atributos do produto cartesiano =
num. atributos de R +
num. atributos de S



Número de tuplas do produto cartesiano =
num. tuplas de R x
num. tuplas de S

Produto Cartesiano (x)

- Exemplo:

R			S	
x	y	z	w	y
1	1	1	1	1
2	2	2	2	2
3	3	3		

x	$R_1.y$	z	w	$R_2.y$
1	1	1	1	1
1	1	1	2	2
2	2	2	1	1
2	2	2	2	2
3	3	3	1	1
3	3	3	2	2

Produto Cartesiano – Exemplo 1

- **Exemplo 1:** Buscar o nome dos médicos que tem consulta marcada e as datas das suas consultas

π medico.nome, consulta.data (
 σ medico.CRM = consulta.CRM (Medico x Consulta))

Produto Cartesiano – Exemplo 2

- **Exemplo 2:** Buscar, para as consultas marcadas no período da manhã (7hs-12hs), o nome do médico, o nome do paciente e a data da consulta

π medico.nome, paciente.nome, consulta.data

(σ consulta.hora \geq 7 AND consulta.hora \leq 12 AND

medico.CRM=consulta.CRM AND

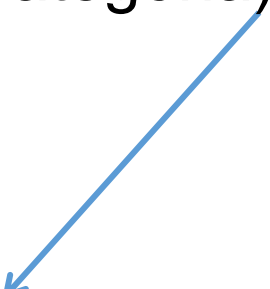
consulta.RG=paciente.RG

(Medico x Consulta x Paciente))

Produto Cartesiano – Exemplo 3:

- Recuperar os dados dos funcionários e o valor dos seus salários

σ Funcionario.id_categ=Categoria.id_categ (Funcionario X Categoria)



Para realizar esta consulta é preciso usar a operação de produto cartesiano para colocar os dados das relações Funcionário e Categoria na mesma tabela, já que os dados necessários para resolver o problema estão nessas duas tabelas.

- A Figura abaixo apresenta, inicialmente, o resultado do produto cartesiano.

Funcionario

id_func	nome_func	id_dep	id_categ
200	Pedro	1	1
201	Paulo	1	2
202	Maria	1	1
203	Ana	2	2

Categoria

id_categ	salario
1	3.000
2	3.500

Resultado

Funcionario. id_func	Funcionario. nome_func	Funcionario. id_dep	Funcionario. id_categ	Categoria. id_categ	Categoria. salario
200	Pedro	1	1	1	3000
200	Pedro	1	1	2	3500
201	Paulo	1	2	1	3000
201	Paulo	1	2	2	3500
202	Maria	1	1	1	3000
202	Maria	1	1	2	3500
203	Ana	2	2	1	3000
203	Ana	2	2	2	3500

Dados das duas tabelas estão agrupados na mesma tabela, mas com todas combinações possíveis, inclusive aquelas que não tem resultado significativo; Tabela resultante tem atributos nomeados com prefixo, que é o nome da tabela de origem;

Atributos **Funcionario.id_categ** e **Categoria.id_categ** estão em outra cor porque estabelecem a relação entre as duas tabelas originais, pois são a **chave estrangeira** de Funcionário e a **chave primária** de Categoria, respectivamente;

Veja que as únicas tuplas que tem resultado significativo são aquelas onde os valores dos atributos que estabelecem a relação entre as duas tabelas são os mesmos, ou seja, as tuplas onde o valor de **Funcionario.id_categ** é igual a **Categoria.id_categ**. Essas tuplas estão também pintadas de outra cor.

Produto Cartesiano – Exemplo 3:

A figura abaixo apresenta a tabela final, resultado de:

$\sigma_{\text{Funcionario.Categoria} = \text{Categoria.Categ}}$ (Funcionário X Categoria)

Resultado

Funcionario. id_func	Funcionario. nome_func	Funcionario. id_dep	Funcionario. id_categ	Categoria. id_categ	Categoria. salario
200	Pedro	1	1	1	3000
201	Paulo	1	2	2	3500
202	Maria	1	1	1	3000
203	Ana	2	2	2	3500

Junção Natural

- É um **operador binário** (envolve duas relações como operandos de entrada) e **cria uma relação com as tuplas que têm os mesmos valores nos atributos de ligação entre as duas relações.**
- **Todos os atributos** das duas relações envolvidas na junção **são levados para a relação resultante**, entretanto, **apenas uma cópia de cada atributo de ligação aparece no resultado.**
- Notação: **<relação1> |X| <relação2>**
 - Tanto relação1 quanto relação2 podem ser tabelas originais do banco de dados ou expressões da Álgebra Relacional.

Junção Natural – Exemplo:

- Recuperar os dados dos funcionários e o valor dos seus salários.

Funcionario |X| Categoria

- Este mesmo problema pode ser resolvido com **Produto Cartesiano** e **Seleção**.
- Entretanto, é possível usar a operação de Junção Natural, **que tem a mesma função do Produto Cartesiano mais a Seleção**, já que recupera os dados que tem relação entre as duas tabelas, ou seja, busca as tuplas que têm os mesmos valores nos atributos de ligação.

Junção Natural – Exemplo:

O exemplo a seguir apresenta o resultado da Junção Natural entre **Funcionario** e **Categoria**

Note que o conjunto de atributos do resultado não contém os atributos repetidos (atributos de ligação) e eles também não são nomeados tendo como prefixo o nome da tabela de origem.

Funcionario

id_func	nome_func	id_dep	id_categ
200	Pedro	1	1
201	Paulo	1	2
202	Maria	1	1
203	Ana	2	2

Categoria

id_categ	salario
1	3.000
2	3.500

Resultado

id_func	nome_func	id_dep	id_categ	salario
200	Pedro	1	1	3000
201	Paulo	1	2	3500
202	Maria	1	1	3000
203	Ana	2	2	3500

Referências

- Bogorny, Vania. Instituto de Informática e Estatística. Universidade Federal de Santa Catarina. Modelo Entidade-Relacionamento, Notas de Aula.
- Muller, Gilberto. Universidade do Vale do Rio dos Sinos. Modelo ER, 2009. Notas de Aula.
- SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S. **Sistemas de Banco de Dados**. 3.a Ed. São Paulo: Pearson, 2010.
- HEUSER, Carlos Alberto. **Projeto de Banco de Dados**. 6. ed. Bookman Companhia Ed, 2009.
- ELMASRI, R.; NAVATHE S. B. **Sistemas de Banco de Dados**. 4 ed. Editora Addison-Wesley. 2005.