

ADMINISTRAÇÃO DE BANCO DE DADOS

Linguagem SQL – Junções

Prof. Ms. Helton Souza Lima

Junções

- Quando a operação de seleção é utilizada envolvendo mais de uma tabela
- **Tipos de junção**
- Junção de produto cartesiano
- Junção interna
- Junção externa

Produto cartesiano

- Quando utilizamos mais de uma tabela na cláusula FROM, o SGBD realiza um produto cartesiano das tabelas envolvidas
- Por exemplo, se X é conjunto dos 13 elementos do baralho inglês:
- $\{A, K, Q, J, 10, 9, 8, 7, 6, 5, 4, 3, 2\}$
- e Y é o conjunto dos quatro naipes:
- $Y = \{\spadesuit, \heartsuit, \diamondsuit, \clubsuit\}$
- então o produto cartesiano desses dois conjuntos será o conjunto com as 52 cartas do baralho:
- $X \times Y = \{(A, \spadesuit), (K, \spadesuit), \dots, (2, \spadesuit), (A, \heartsuit), \dots, (3, \clubsuit), (2, \clubsuit)\}.$

Produto cartesiano

- Todos os registros de uma tabela serão associados a cada um dos registros da outra tabela
- Operação exponencialmente custosa com o aumento da quantidade de registros

Produto cartesiano

- Considerando a utilização de mais de uma tabela
- Vamos fazer um teste
- **SELECT * FROM empregados, departamentos**
 - Produto cartesiano não é recomendado com muitos dados

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

DEPARTAMENTOS	
<u>Código</u>	Nome
D1	Compras
D2	Médico
D3	Educação

Produto cartesiano

- Considerando a utilização de mais de uma tabela
- Selecione o nome e o endereço de todos os empregados do departamento de Compras
- `SELECT e.nome, e.endereco FROM empregados e, departamentos d WHERE d.nome = 'Compras' and d.codigo = e.departamento`

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

DEPARTAMENTOS	
<u>Código</u>	Nome
D1	Compras
D2	Médico
D3	Educação

Comando SELECT

- **Autorelacionamento**
- Todo empregado tem um supervisor
- ALTER TABLE empregados add constraint fk_sup foreign key (supervisor) references empregados(matricula)
- UPDATE empregados SET supervisor=100 WHERE matricula <> 100

EMPREGADOS						
<u>Matrícula</u>	Nome	Endereço	Salário	Função	Departamento	Supervisor
100	Ana	R. Pedro I, 12, A. Branco	500,00	Secretária	D1	
250	Pedro	R. J. Silva, 24, Liberdade	1500,00	Engenheiro	D1	100
108	André	R. Itália, 33, B. Nações	950,00	Técnico	D2	100
210	Paulo	R. Pará, 98, B. Estados	1810,00	Engenheiro	D2	100
105	Sônia	R. Oliveira, 76, A. Branco	2500,00	Engenheiro	D1	108

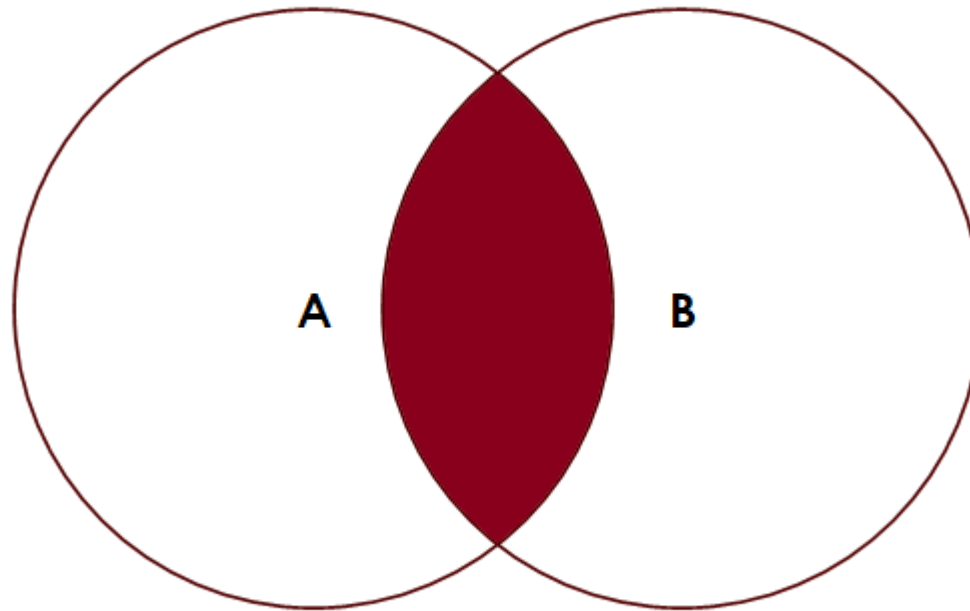
Comando SELECT

- **Autorelacionamento**
- Para cada empregado, recupere seu nome e o nome do seu supervisor
- SELECT e.nome as empregado, s.nome as supervisor FROM empregados e, empregados s WHERE e.matricula = s.supervisor

EMPREGADOS						
<u>Matrícula</u>	Nome	Endereço	Salário	Função	Departamento	Supervisor
100	Ana	R. Pedro I, 12, A. Branco	500,00	Secretária	D1	
250	Pedro	R. J. Silva, 24, Liberdade	1500,00	Engenheiro	D1	100
108	André	R. Itália, 33, B. Nações	950,00	Técnico	D2	100
210	Paulo	R. Pará, 98, B. Estados	1810,00	Engenheiro	D2	100
105	Sônia	R. Oliveira, 76, A. Branco	2500,00	Engenheiro	D1	100

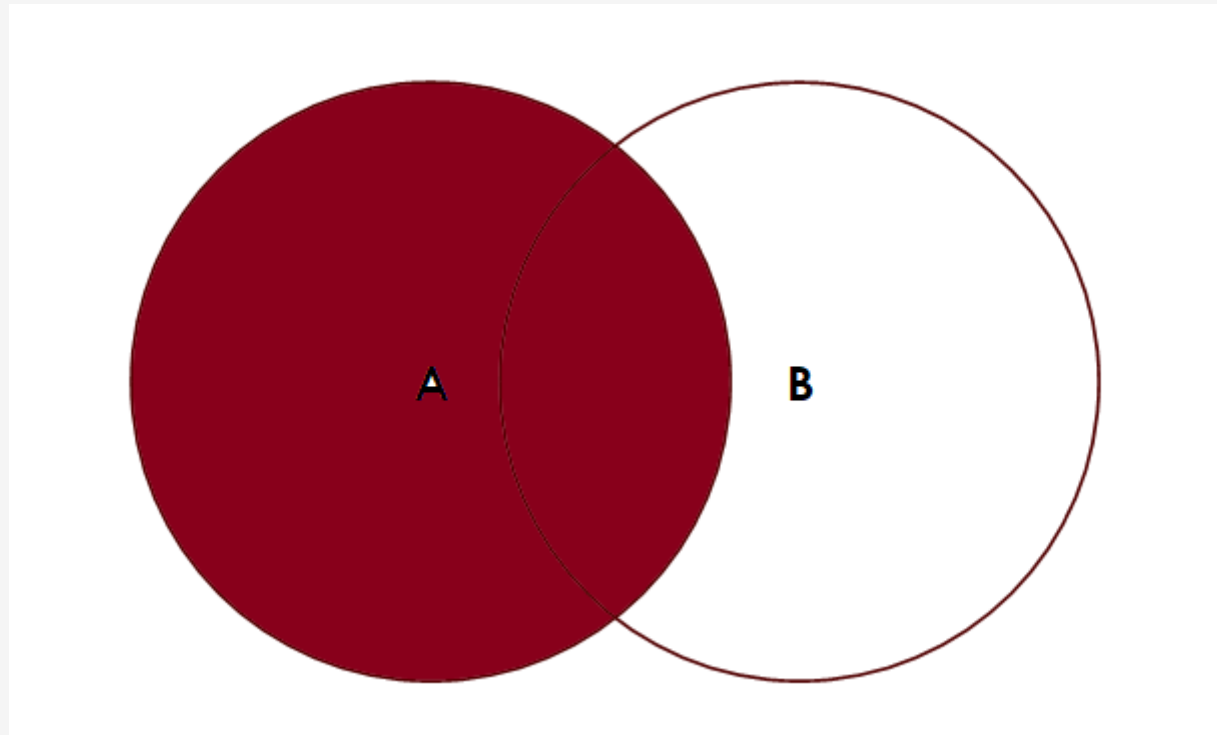
Junção interna

- [INNER] **JOIN**
- Apenas os registros coincidentes serão selecionados



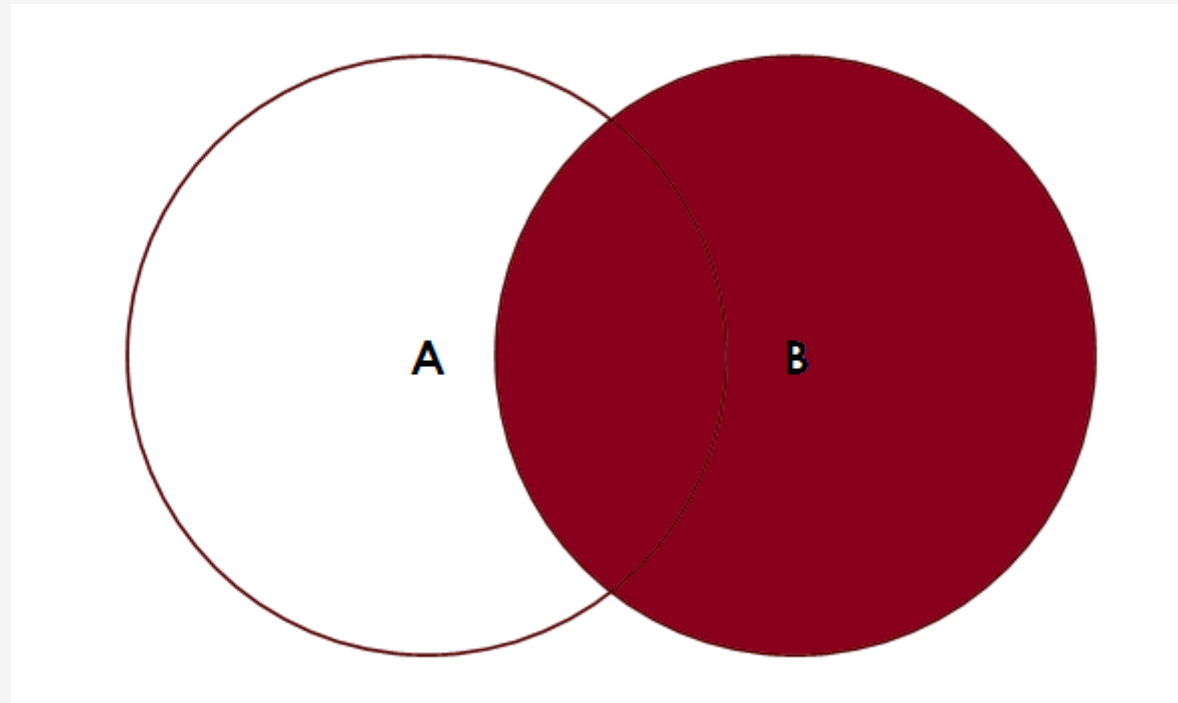
Junção externa

- **LEFT JOIN**
- Todos os registros de A junto a todos os registros coincidentes com B serão selecionados



Junção externa

- **RIGHT JOIN**
- Todos os registros de B junto a todos os registros coincidentes com A serão selecionados



INNER JOIN

- É uma junção entre duas tabelas que origina uma terceira tabela constituída por cada um dos elementos da primeira tabela combinados com cada elemento da segunda tabela
- Já vimos junção de produto cartesiano em consultas anteriores
- Selecione o nome e o endereço de todos os empregados do departamento de Compras
- `SELECT e.nome, e.endereco FROM empregados e, departamentos d WHERE d.nome = 'Compras' and d.codigo = e.departamento`

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

DEPARTAMENTOS	
<u>Código</u>	Nome
D1	Compras
D2	Médico
D3	Educação

INNER JOIN

- SELECT e.nome, e.endereco FROM empregados e, departamentos d WHERE d.nome = 'Compras' and **d.codigo = e.departamento**
- SELECT e.nome, e.endereco FROM empregados e [INNER] JOIN departamentos d ON (d.codigo = e.departamento) WHERE d.nome = 'Compras'

NOME	ENDereco
Ana	R. Pedro I, 12, A. Branco
Pedro	R. J. Silva, 24, Liberdade
Sônia	R. Oliveira, 76, A. Branco

Junção externa

- Vamos acrescentar um empregado que não está alocado a nenhum departamento
- E acrescentar um departamento que não tem nenhum empregado alocado
 - **INSERT INTO EMPREGADOS**(matricula, nome, endereco, funcao, salario) **values** (300, 'Miguel','R. Dom Bosco, 345, Catolé','Professor', 1000)
 - **INSERT INTO DEPARTAMENTOS**(codigo, nome) **values** ('E1', 'Segurança')

EMPREGADOS					
<u>Matrícula</u>	Nome	Endereço	Salário	Função	Departamento
100	Ana	R. Pedro I, 12, A. Branco	500,00	Secretária	D1
250	Pedro	R. J. Silva, 24, Liberdade	1500,00	Engenheiro	D1
108	André	R. Itália, 33, B. Nações	950,00	Técnico	D2
210	Paulo	R. Pará, 98, B. Estados	1810,00	Engenheiro	D2
105	Sônia	R. Oliveira, 76, A. Branco	2500,00	Engenheiro	D1
300	Miguel	R. Dom Bosco, 345	1000,00	Professor	

DEPARTAMENTOS	
<u>Código</u>	Nome
D1	Compras
D2	Médico
D3	Educação
E1	Segurança

LEFT [OUTER] JOIN

- Verificando a diferença usando JOIN ou LEFT JOIN

SELECT e.nome, e.endereco, d.nome **FROM** empregados **JOIN** departamentos d **ON** (e.departamento = d.codigo)

NOME	ENDEREÇO	NOME
Ana	R. Pedro I, 12, A. Branco	Compras
Pedro	R. J. Silva, 24, Liberdade	Compras
André	R. Itália, 33, B. Nações	Médico
Paulo	R. Pará, 98, B. Estados	Médico
Sônia	R. Oliveira, 76, A. Branco	Compras

SELECT e.nome, e.endereco, d.nome **FROM** empregados **LEFT JOIN** departamentos d **ON** (e.departamento = d.codigo)

NOME	ENDEREÇO	NOME
Ana	R. Pedro I, 12, A. Branco	Compras
Pedro	R. J. Silva, 24, Liberdade	Compras
André	R. Itália, 33, B. Nações	Médico
Paulo	R. Pará, 98, B. Estados	Médico
Sônia	R. Oliveira, 76, A. Branco	Compras
Miguel	R. Dom Bosco, 345, Catolé	

FULL JOIN

- Verificando a diferença usando JOIN ou FULL JOIN

SELECT e.nome, e.endereco, d.nome **FROM** empregados **JOIN** departamentos d **ON** (e.departamento = d.codigo)

NOME	ENDERECO	NOME
Ana	R. Pedro I, 12, A. Branco	Compras
Pedro	R. J. Silva, 24, Liberdade	Compras
André	R. Itália, 33, B. Nações	Médico
Paulo	R. Pará, 98, B. Estados	Médico
Sônia	R. Oliveira, 76, A. Branco	Compras

SELECT e.nome, e.endereco, d.nome **FROM** empregados **FULL JOIN** departamentos d **ON** (e.departamento = d.codigo)

NOME	ENDERECO	NOME
Ana	R. Pedro I, 12, A. Branco	Compras
Pedro	R. J. Silva, 24, Liberdade	Compras
André	R. Itália, 33, B. Nações	Médico
Paulo	R. Pará, 98, B. Estados	Médico
Sônia	R. Oliveira, 76, A. Branco	Compras
		Educação
		Segurança
Miguel	R. Dom Bosco, 345, Catolé	