

Consultas SQL

- SELECT -

PARTE 2

Prof. Dr. Anderson Chaves Carniel

`accarniel@ufscar.br`



Base de dados (esquema relacional) considerada

Empregado = {PrimeiroNome, InicialMeio, UltimoNome, NumEmpregado, DataNascimento, Endereco, Sexo, Salario, NumSupervisor, NumDepto}

Departamento = {NomeDepto, NumDepto, NumGerente, DataInicioGerencia}

Localizacao_Depto = {NumDepto, Localizacao}

Projeto = {NomeProj, NumProj, Localizacao, NumDepto}

Trabalha = {NumEmpregado, NumProj, Horas}

Dependente = {NumEmpregado, NomeDependente, Sexo, DataAniversario, Parentesco}

O esquema possui os dados disponibilizados no Moodle
Carregá-los usando o pgAdmin

Álgebra relacional VERSUS SQL: UNIÃO

- **União:**
 - Resulta em uma tabela que contém todas as tuplas das tabelas envolvidas (colunas devem ser compatíveis de domínio)
- Exemplo: **retornar todos os nomes de empregados e dependentes**

COMO fica (em álgebra relacional e em SQL)?

Álgebra relacional VERSUS SQL: UNIÃO

- **União:**

- Resulta em uma tabela que contém todas as tuplas das tabelas envolvidas (colunas devem ser compatíveis de domínio)

- Exemplo: retornar todos os nomes de empregados e dependentes

- Álgebra relacional:

$\Pi_{\{nomedependente\}} dependente \cup \Pi_{\{primeironome\}} empregado$

- SQL:

SELECT nomedependente as nomes FROM dependente

UNION

SELECT primeironome FROM empregado

	nomes character varying
1	Romeu
2	Vicente
3	Nicoli
4	Raissa
5	Carlos
6	Fran
7	André
8	Karla
9	Ana
10	Antonio
11	Carla
12	Geraldo
13	Julietta
14	Pita
15	Roberto
16	Lucia
17	Nicolai
18	Rodrigo
19	Roberta

Álgebra relacional VERSUS SQL: INTERSECÇÃO

- **Intersecção:**
 - Resulta em uma tabela que contém somente as tuplas em comum entre as tabelas envolvidas (colunas devem ser compatíveis de domínio)
 - Exemplo: **retornar todos os nomes comuns de empregados e dependentes**

COMO fica (em álgebra relacional e em SQL)?

Álgebra relacional VERSUS SQL: INTERSECÇÃO

- **Intersecção:**

- Resulta em uma tabela que contém somente as tuplas em comum entre as tabelas envolvidas (colunas devem ser compatíveis de domínio)
- Exemplo: retornar todos os nomes comuns de empregados e dependentes
- Álgebra relacional:

$\Pi_{\{nomedependente\}} dependente \cap \Pi_{\{primeironome\}} empregado$

- SQL:

SELECT nomedependente as nomes FROM dependente

INTERSECT

SELECT primeironome FROM empregado

resultado vazio para o consulta, considerando o conjunto de dados atual

Exercício: inserir 2 tuplas para retornar um resultado para esta consulta

Álgebra relacional VERSUS SQL: DIFERENÇA

- **Diferença:**
 - Resulta em uma tabela que contém somente as tuplas da primeira relação mas que não estão na segunda relação (colunas devem ser compatíveis de domínio)
- Exemplo: **retornar todos os nomes de empregados que não são usados como nomes de dependentes**

COMO fica (em álgebra relacional e em SQL)?

Álgebra relacional VERSUS SQL: DIFERENÇA

- **Diferença:**

- Resulta em uma tabela que contém somente as tuplas da primeira relação mas que não estão na segunda relação (colunas devem ser compatíveis de domínio)

- Exemplo: retornar todos os nomes de empregados que não são usados como nomes de dependentes

- Álgebra relacional:

$$\Pi_{\{\text{primeironome}\}} \text{ empregado} - \Pi_{\{\text{nomedependente}\}} \text{ dependente}$$

- SQL:

SELECT primeironome FROM empregado

EXCEPT

SELECT nomedependente FROM dependente



	primeironome character varying
1	Julieta
2	Romeu
3	Lucia
4	Carlos
5	Fran
6	André
7	Karla
8	Ana
9	Antonio
10	Carla
11	Geraldo

Álgebra relacional versus SQL: operações de conjunto

- PostgreSQL tem variações nesses operadores, verificar o manual online!
 - O padrão para essas operações é eliminar tuplas repetidas (**DISTINCT**)
 - Para manter tuplas repetidas no resultado, inclusive no **UNION**, deve-se usar o **ALL** depois da palavra do operador de conjunto

Álgebra relacional	SQL
\cup	UNION
\cap	INTERSECT
$-$	EXCEPT