

Introdução

Floriano Ferreira dos Reis Filho

Agenda

- Álgebra Relacional
- Seleção
- Projeção
- Seleção e Projeção
- União
- Intersecção
- Condições para União e Intersecção
- Diferença
- Produto Cartesiano
- Junção
- Operadores da Álgebra Relacional

Álgebra Relacional

Relação

- Representa uma tabela de duas dimensões (linhas e colunas)

Tupla

- Corresponde a uma linha da relação

Atributo

- Corresponde a cada coluna da relação

Álgebra Relacional

The diagram illustrates a relational database table. Above the table, five blue arrows point down to the column headers, each labeled 'Atributo' in blue. To the left of the table, five orange arrows point right to the rows, with the label 'Tuplas' in orange. To the right of the table, a blue arrow points left to the column headers, labeled 'Atributos' in blue. The table itself has five columns and six rows. The first row contains the column headers: 'Mat.', 'Nome', 'Sexo', 'Data Adm.', and 'Cód. Depto.', all in red text. The subsequent rows contain data: (1, Zé Mané, M, 01/02/1986, 20), (2, Caboclinha, F, 13/08/1995, 30), (3, Badé, M, 16/03/1997, 20), (4, Serafim, M, 03/03/1993, 10), a row with ellipses (...), and (150, Sam Miojo, F, 26/11/2001, 40).

Mat.	Nome	Sexo	Data Adm.	Cód. Depto.
1	Zé Mané	M	01/02/1986	20
2	Caboclinha	F	13/08/1995	30
3	Badé	M	16/03/1997	20
4	Serafim	M	03/03/1993	10
...
150	Sam Miojo	F	26/11/2001	40

Relação ALUNO

Álgebra Relacional

Álgebra Relacional

É um conjunto básico de operações sobre modelos relacionais de dados.

O resultado de uma operação sobre uma ou mais relações gera uma outra relação.

A álgebra relacional auxilia o entendimento e o aprendizado da linguagem SQL.

Álgebra Relacional

Operações primárias

- Seleção
- Projeção

Operações binárias

- União
- Intersecção
- Diferença
- Produto Cartesiano

Operações adicionais

- Junção
- Divisão

Álgebra Relacional

Funcionário

Cód.	Nome	Sexo	Data Admis.	Salário	Depto.
10	Visconde	M	01/02/1983	R\$ 1230,00	3
20	Pedrinho	M	29/07/1990	R\$ 867,00	3
30	Dona Benta	F	30/11/1992	R\$ 2560,00	1
40	Emília	F	22/02/1998	R\$ 1170,00	2
50	Rabicó	M	08/09/2000	R\$ 2300,00	1

Departamento

CódDepto.	Nome Depto.
1	Administração
2	Contabilidade
3	Informática

Seleção

Seleção

- ❑ É a operação usada para selecionar um subconjunto de tuplas de uma relação R que satisfaça uma condição de seleção.

Seleção

É representada como:

$\sigma_{\text{<condição>}}(R)$

<condição> permite o uso dos operadores:

- ❑ De comparação: =, <>, <, >, <=, >=
- ❑ Lógicos: and (\wedge), or (\vee), not (\neg)

Seleção

- ❑ Representação gráfica
 - ❑ Obtenção de um subconjunto horizontal de uma relação R

Tuplas que satisfazem
uma condição



Seleção

- ❑ Exemplo 1:
 - ❑ Selecionar todos os funcionários do sexo feminino

$\sigma_{\text{Sexo} = \text{'F'}} (\text{Funcionário})$

Cód.	Nome	Sexo	Data Admis.	Salário	Depto.
30	Dona Benta	F	30/11/1992	R\$ 2560,00	1
40	Emília	F	22/02/1998	R\$ 1170,00	2

Seleção

- ❑ Exemplo 2:
 - ❑ Selecionar todos os funcionários do sexo masculino que ganham mais de R\$ 1000,00

**σ Sexo = 'M' \wedge Salário > 1000
(Funcionário)**

Cód.	Nome	Sexo	Data Admis.	Salário	Depto.
10	Visconde	M	01/02/1983	R\$ 1230,00	3
50	Rabico	M	08/09/2000	R\$ 2300,00	1

Projeção

Projeção

- ❑ É a operação usada para selecionar determinadas colunas de uma relação R .

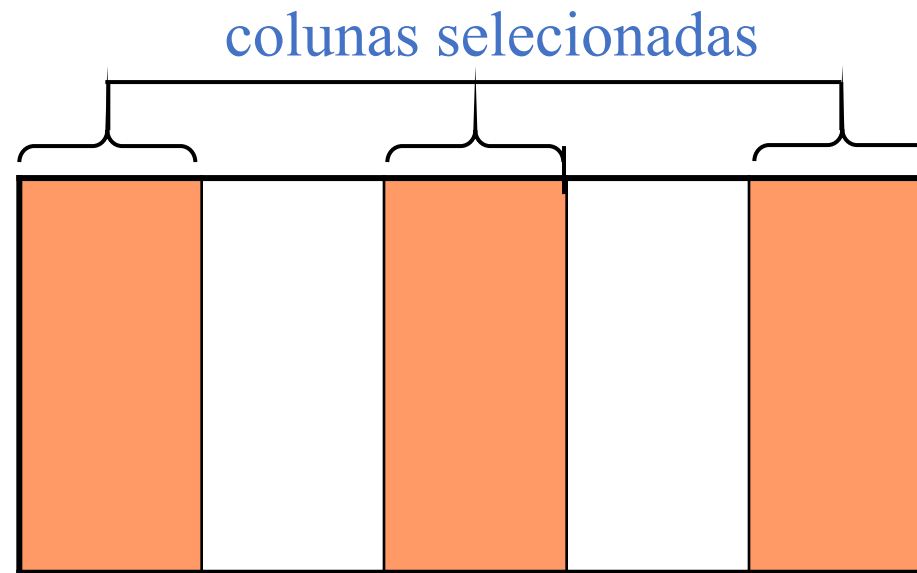
Projeção

- ❑ Já que a relação é um conjunto, quaisquer linhas em duplicidade são eliminadas.
- ❑ É representada como:

$\pi_{\langle \text{lista de atributos} \rangle} (R)$

Projeção

- ❑ Representação gráfica.
 - ❑ Obtenção de um subconjunto vertical de uma relação R



Projeção

- ❑ Exemplo 3:
 - ❑ Listar o nome, o sexo e o salário de todos os funcionários

π Nome, Sexo, Salário (Funcionário)

Nome	Sexo	Salário
Visconde	M	R\$ 1230,00
Pedrinho	M	R\$ 867,00
Dona Benta	F	R\$ 2560,00
Emília	F	R\$ 1170,00
Rabicó	M	R\$ 2300,00

Seleção e Projeção

- ❑ As operações podem ser utilizadas em conjunto.
- ❑ Exemplo 4:
 - ❑ Selecionar o nome e o salário de todos os funcionários que ganham mais de R\$ 2000,00

$\pi_{\text{Nome, Salário}} (\sigma_{\text{Salário} > 2000} (\text{Funcionário}))$

Nome	Salário
Dona Benta	R\$ 2560,00
Rabicó	R\$ 2300,00

Funcionário

Cód.	Nome	Sexo	Data Admis.	Salário	Depto.
10	Visconde	M	01/02/1983	R\$ 1230,00	3
20	Pedrinho	M	29/07/1990	R\$ 867,00	3
30	Dona Benta	F	30/11/1992	R\$ 2560,00	1
40	Emília	F	22/02/1998	R\$ 1170,00	2
50	Rabico	M	08/09/2000	R\$ 2300,00	1

Aluno

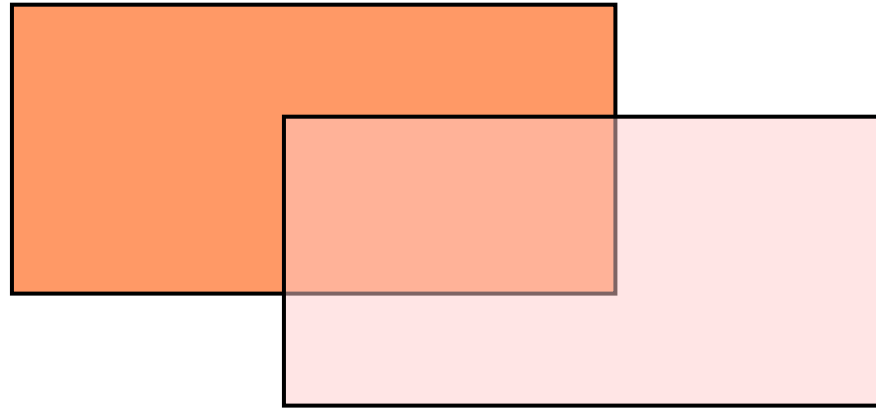
Cód.	Nome	Sexo
1	Visconde	M
2	Dona Benta	F
3	Rabico	M
4	Cuca	F

União

❑ $R \cup S$

❑ A união de duas relações R e S é o conjunto de todas as tuplas pertencentes à relação R ou à relação S ou a ambas.

❑ Representação gráfica:



União

- ❑ Exemplo 5:
 - ❑ Obter os nomes e o sexo de todos os funcionários e alunos

$\Pi_{\text{Nome, Sexo}}$ (Funcionário)
 \cup
 $\Pi_{\text{Nome, Sexo}}$ (Aluno)

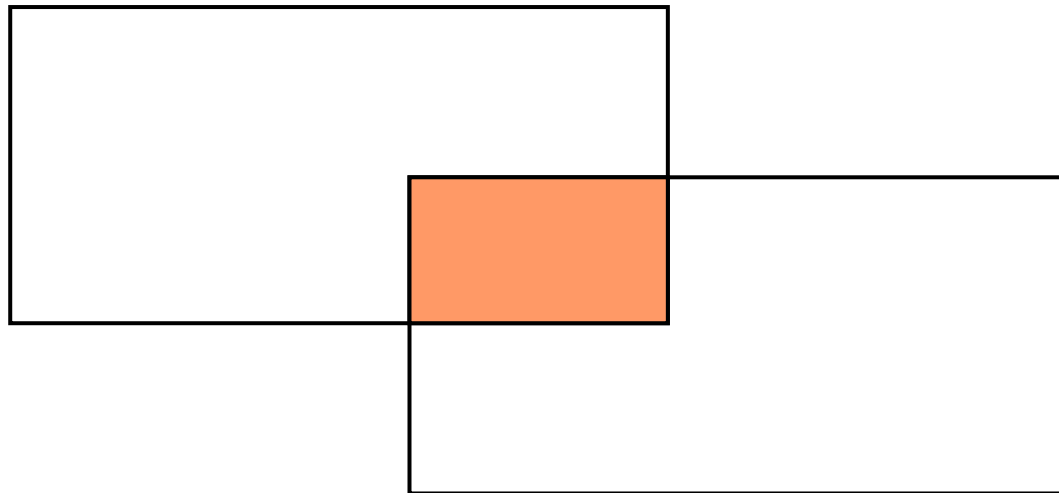
Nome	Sexo
Visconde	M
Pedrinho	M
Dona Benta	F
Emília	F
Rabicó	M
Cuca	F

Intersecção

□ $R \cap S$

□ A intersecção de duas relações R e S é o conjunto de todas as tuplas pertencentes à relação R e à relação S .

□ Representação gráfica:



Intersecção

- Exemplo 6:
 - Obter o nome e o sexo de todos os funcionários que são alunos.

$\pi_{\text{Nome, Sexo}}(\text{Funcionário})$
 \cap
 $\pi_{\text{Nome, Sexo}}(\text{Aluno})$

Nome	Sexo
Visconde	M
Dona Benta	F
Rabicó	M

Condições para União e Intersecção

Condições

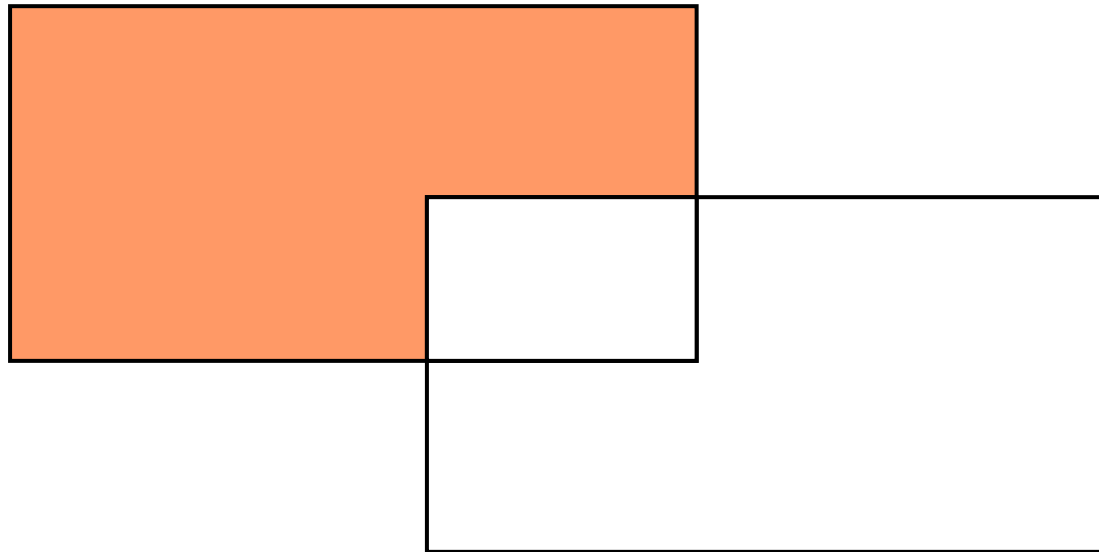
- ☐ Uma operação de União / Intersecção só é válida se:
- ☐ As relações R e S possuírem o mesmo número de atributos;

Diferença

- ❑ $R - S$

- ❑ A diferença de duas relações R e S é o conjunto de todas as tuplas pertencentes à relação R e não pertencentes à relação S .

- ❑ Representação gráfica:



Diferença

- ❑ Exemplo 7:
 - ❑ Obter os nomes e o sexo de todos os funcionários que não são alunos.

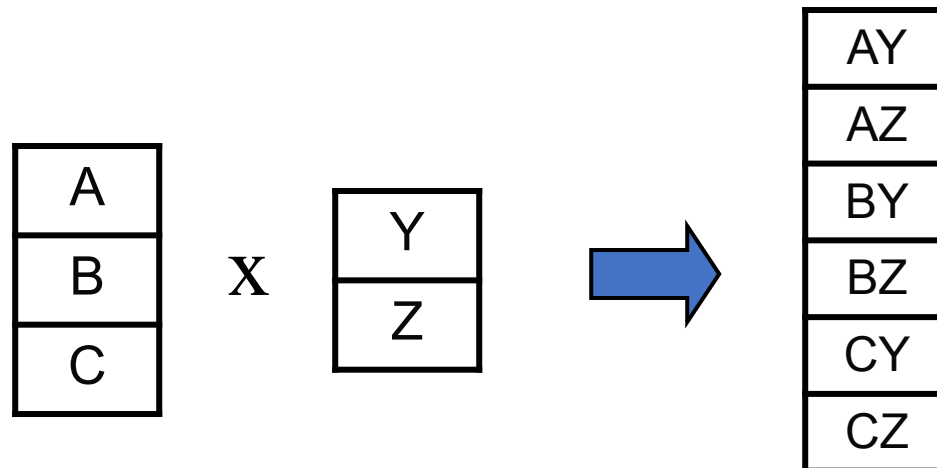
$\pi_{\text{Nome, Sexo}}(\text{Funcionário}) - \pi_{\text{Nome, Sexo}}(\text{Aluno})$

Nome	Sexo
Pedrinho	M
Emília	F

Produto Cartesiano

□ $R \times S$

- O produto cartesiano de duas relações R e S é o uma relação T cujas tuplas são a combinação das tuplas das relações R e S , tomando-se uma tupla de R e concatenando-a com cada tupla de S .
- Representação gráfica:



Produto Cartesiano

- ❑ Exemplo 8:
 - ❑ Obter, para cada funcionário, uma lista de todos os departamentos.

Nome Func.	Sexo	Data Adm.	Nome Depto.
Vinconde	M	01/02/1983	Administração
Visconde	M	01/02/1983	Contabilidade
Visconde	M	01/02/1983	Informática
(continua...)			

Produto Cartesiano

Nome Func.	Sexo	Data Adm.	Nome Depto.
Pedrinho	M	29/07/1990	Administração
Pedrinho	M	29/07/1990	Contabilidade
Pedrinho	M	29/07/1990	Informática
Dona Benta	F	30/11/1992	Administração
Dona Benta	F	30/11/1992	Contabilidade
Dona Benta	F	30/11/1992	Informática
Emília	F	22/02/1998	Administração
Emília	F	22/02/1998	Contabilidade
Emília	F	22/02/1998	Informática
Rabicó	M	08/09/2000	Administração
Rabicó	M	08/09/2000	Contabilidade
Rabicó	M	08/09/2000	Informática

Junção

- ❑ É utilizada para combinar *tuplas relacionadas* de duas relações em uma única tupla
- ❑ É representada como:

$$R \bowtie_{\langle \text{condição de junção} \rangle} S$$


Junção

- ❑ Exemplo 9:
 - ❑ Listar todos os funcionários e o nome do departamento em que cada um deles trabalha

Funcionário ⋈ **Departamento**
Dept. = CódDept.

Junção

Junção



Nome	Sexo	Data Admis.	Depto.	CódDepto.	Nome Depto.
Visconde	M	01/02/1983	3	3	Informática
Pedrinho	M	29/07/1990	3	3	Informática
Dona Benta	F	30/11/1992	1	1	Administração
Emília	F	22/02/1998	2	2	Contabilidade
Rabicó	M	08/09/2000	1	1	Administração

Operadores da Álgebra Relacional

Operadores da
Álgebra Relacional

Símbolo	Operação	Sintaxe	Tipo
π	Projeção (project)	$\pi_{atributos}(Relacao)$	Primitiva - Unária
σ	Seleção (select)	$\sigma_{condicao}(Relacao)$	Primitiva - Unária
\times	Produto Cartesiano (cartesian product)	$Relacao1 \times Relacao2$	Primitiva - Binária
\cup	União (union)	$Relacao1 \cup Relacao2$	Primitiva - Binária
$-$	Diferença entre Conjuntos (set difference)	$Relacao1 - Relacao2$	Primitiva - Binária
\cap	Conjunto Intersecção (set intersection)	$Relacao1 \cap Relacao2$	Derivada - Binária
\bowtie	Junção Natural (natural join)	$Relacao1 \bowtie Relacao2$	Derivada - Binária
\div	Divisão (division)	$Relacao1 \div Relacao2$	Derivada - Binária
ρ	Renomear Relação ou Atributo	$\rho_{Nome}(Relacao)$	Especial - Unária
\leftarrow	Designação, ou Atribuição (assignment)	$variavel \leftarrow Relacao$	Especial - Unária

Exercício

CódigoCliente	Nome	TipoRelação	Sexo	DataNasc
0001	Maria	Esposa	F	01/01/1970
0001	Vítor	Filho	M	02/02/2002
0001	Ana	Filha	F	03/03/2003
1000	João	Filho	M	02/02/2002
1000	Vítor	Filho	M	02/02/2002
1000	Vítor	Marido	M	02/02/1971
9876	Sônia	Esposa	F	01/01/1970

O conjunto de atributos de uma relação é chamado de **relação esquema**

Cada atributo possui um **domínio**.

dom = domínio

O **grau** de uma relação é o número de atributos da relação.

Tuplas são as linhas da tabela.

t = tupla

Cada **relação** pode ser entendida como uma **tabela**.

A figura apresenta a Relação DEPENDENTE

t = <0001, Ana, Filha, F, 03/03/2003> é uma tupla

t[CódigoCliente] = 0001

t[Nome, Sexo] = <Ana, F>.

DEPENDENTE(CódigoCliente, Nome, TipoRelação, Sexo, DataNasc)

- É a **relação esquema**.
- **DEPENDENTE** é o nome da relação.
- O **Grau da Relação** é 5.
- Os **Domínios** dos Atributos são:
 - dom(CódigoCliente) = 4 dígitos que representam o Código do Cliente.
 - dom(Nome) = Caracteres que representam nomes dos dependentes.
 - dom(TipoRelação) = Tipo da Relação (filho, esposa, pai, mãe e outras) do dependente em relação do seu cliente.
 - dom(Sexo) = Caractere: (M: Masculino, F: Feminino) do dependente.
 - dom(DataNasc) = Datas de Nascimento do dependente.

Exercício:

Analise as afirmações e indique na resposta **V** para Verdadeira e **F** para Falsa.

Afirmação	Resposta
O conjunto de atributos de uma relação é chamado de relação esquema.	V
Cada atributo possui um domínio.	v
O grau de uma relação é o número de atributos da relação.	∇
Tuplas são as linhas da tabela.	V
Cada relação pode ser entendida como uma tabela.	V



Cada atributo possui um **domínio**.

dom = domínio

O grau de uma relação é o número de atributos da relação.

Tuplas são as linhas da tabela.

t = tupla

Cada relação pode ser entendida como uma tabela.

ENGENHEIRO (codigo, nome)

dom(codigo): código numérico de 4 posições

dom(nome): cadeia de caracteres de até 50 posições

PROJETO (codigo, nome)

dom(codigo): código numérico de 3 posições

dom(titulo): cadeia de caracteres de até 30 posições

ATUACAO (codproj, codeng, funcao)

codproj Referencia PROJETO

codeng Referencia ENGENHEIRO

dom(codproj): código numérico de 3 posições

dom(codeng): código numérico de 4 posições

dom(funcao): cadeia de caracteres de até 15 posições

ENGENHEIRO (Código, Nome)

- (Código, Nome) é a **relação esquema**.
- **ENGENHEIRO** é o nome da relação.
- O **Grau da Relação** é 2.
- Os **Domínios** dos Atributos são:
dom(codigo): código numérico de 4 posições
dom(nome): cadeia de caracteres de até 50 posições

Exercício:

Analise as afirmações e indique na resposta **V** para Verdadeira e **F** para Falsa.

Afirmação	Resposta
Os registros (ou tuplas) contidos em cada relação são apresentados nas Tabelas.	V
O símbolo σ (sigma) é usado para indicar o operador SELEÇÃO e a condição de seleção é uma expressão booleana (condição) especificada nos atributos da relação R.	V
É a operação usada para selecionar um subconjunto de tuplas de uma relação R que satisfaça uma condição de projeção.	F
A união de duas relações R e S é o conjunto de todas as tuplas pertencentes à relação R ou à relação S ou a ambas.	V
A junção é representada pelo símbolo Theta, essa junção também é conhecida como Junção Condicional e baseia-se em uma combinação dos operadores produto cartesiano e seleção.	F

Exercício

QUESTÃO 23

Considere o esquema de banco de dados relacional apresentado a seguir, formado por 4 relações, que representa o conjunto de estudantes de uma universidade que podem, ou não, morar em repúblicas (moradias compartilhadas por estudantes). A relação *Estudante* foi modelada como um subconjunto da relação *Pessoa*. Considere que os atributos grifados correspondam à chave primária da respectiva relação e os atributos que são seguidos da palavra *referencia* sejam chaves estrangeiras.

```
Pessoa (IdPessoa:integer,      Nome:varchar(40),  
Endereco:varchar(40))  
FonePessoa (IdPessoa:integer referencia Pessoa,  
DDD:varchar(3), Prefixo:char(4), Nro:char(4))  
Republica (IdRep:integer,      Nome:varchar(30),  
Endereco:varchar(40))  
Estudante (RA:integer,      Email:varchar(30),  
IdPessoa:integer referencia Pessoa,  
IdRep:integer referencia Republica)
```

Suponha que existam as seguintes tuplas no banco de dados:

```
Pessoa(1, 'José Silva', 'Rua 1, 20');  
Republica(20, 'Várzea', 'Rua Chaves, 2001')
```

Qual opção apresenta apenas tuplas válidas para esse esquema de banco de dados relacional?

- ☐ A Estudante(10, 'jsilva@ig.com.br', null, 20);
FonePessoa(10, '019', '3761', '1370')
- ☒ B Estudante(10, 'jsilva@ig.com.br', 1, null);
FonePessoa(10, '019', '3761', '1370')
- ☐ C Estudante(10, 'jsilva@ig.com.br', 1, 20);
FonePessoa(1, null, '3761', '1370')
- ☐ D Estudante(10, 'jsilva@ig.com.br', 1, 50);
FonePessoa(1, '019', '3761', '1370')
- ☐ E Estudante(10, 'jsilva@ig.com.br', 1, null);
FonePessoa(1, '019', '3761', '1370')

Exercício:

Analise as afirmações e indique na resposta **V** para Verdadeira e **F** para Falsa.

Afirmação	Resposta
Para uma operação de União e Intersecção ser válida é preciso que as relações R e S possuam o mesmo número de atributos.	V
Seleção é a operação usada para selecionar determinadas colunas de uma relação R.	F
Junção é utilizada para combinar tuplas relacionadas de duas relações em uma única tupla.	V
O produto cartesiano de duas relações R e S é o uma relação T cujas tuplas são a combinação das tuplas das relações R e S, tomando-se uma tupla de R e concatenando-a com cada tupla de S.	V
A diferença de duas relações R e S é o conjunto de todas as tuplas pertencentes à relação R e não pertencentes à relação S.	V

Exercício

QUESTÃO 40 – DISCURSIVA

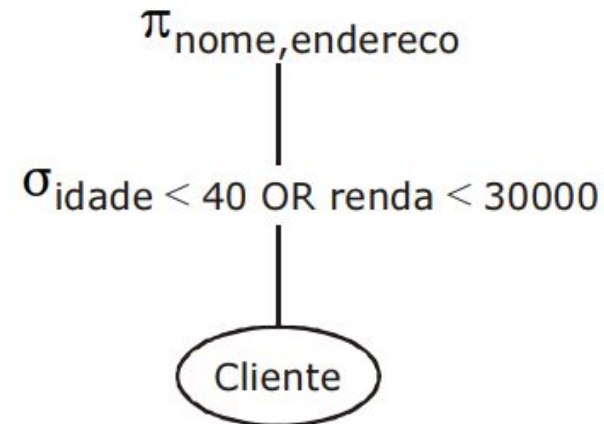
O banco de dados de um sistema de controle bancário implementado por meio de um SGBD relacional possui a relação `Cliente`, com as informações apresentadas a seguir, em que a chave primária da relação é grifada.

```
Cliente(nroCliente, nome, endereco,  
       data_nascimento, renda, idade)
```

Para essa relação, foram criados dois índices secundários: `IndiceIdade`, para o atributo `idade`, e `IndiceRenda`, para o atributo `renda`. Existe um tipo de serviço nesse banco cujo alvo são tanto os clientes que possuem menos de 40 anos de idade quanto aqueles que possuem renda mensal superior a 30.000 reais. Para recuperar esses clientes, a seguinte expressão de consulta em SQL foi utilizada:

```
SELECT nome, endereco  
FROM Cliente  
WHERE idade < 40 OR renda > 30000;
```

Com o aumento do número de clientes desse banco, essa consulta passou a apresentar problemas de desempenho. Verificou-se, então, que o otimizador de consultas não considerava os índices existentes para `idade` e `renda`, e a consulta era realizada mediante varredura seqüencial na relação `Cliente`, tornando essa consulta onerosa. O plano de execução da consulta, usado pelo otimizador, é apresentado na árvore de consulta abaixo, na qual π e σ representam as operações de projeção e de seleção, respectivamente.



Exercício

Para que o otimizador de consultas passasse a utilizar os índices, a solução encontrada foi elaborar a consulta em dois blocos separados — um que recupera os clientes com idade inferior a 40 anos, e outro que recupera os clientes com renda mensal superior a 30.000 reais — para, então, juntar as tuplas das duas relações geradas.

Considerando a situação apresentada, faça o que se pede a seguir.

A Escreva o código de uma consulta em SQL que corresponda à solução proposta.

(valor: 5,0 pontos)

B Desenhe a árvore de consulta para essa solução.

(valor: 5,0 pontos)

Exercício

QUESTÃO 24

Considere as seguintes tabelas de um banco de dados:

1. Fornecedor (cod_fornec, nome_fornec, telefone, cidade, UF)
2. Estado (UF, nome_estado)

A expressão SQL que obtém os nomes dos estados para os quais não há fornecedores cadastrados é

A

```
SELECT E.UF
FROM Estado AS E
WHERE E.nome_estado NOT IN (
    SELECT F.UF
    FROM Fornecedor AS F);
```

B

```
SELECT E.nome_estado
FROM Estado AS E, FROM
Fornecedor AS F
WHERE E.UF = F.UF;
```

C

```
SELECT E.nome_estado
FROM Estado AS E
WHERE E.UF NOT IN (
    SELECT F.UF
    FROM Fornecedor AS F);
```

D

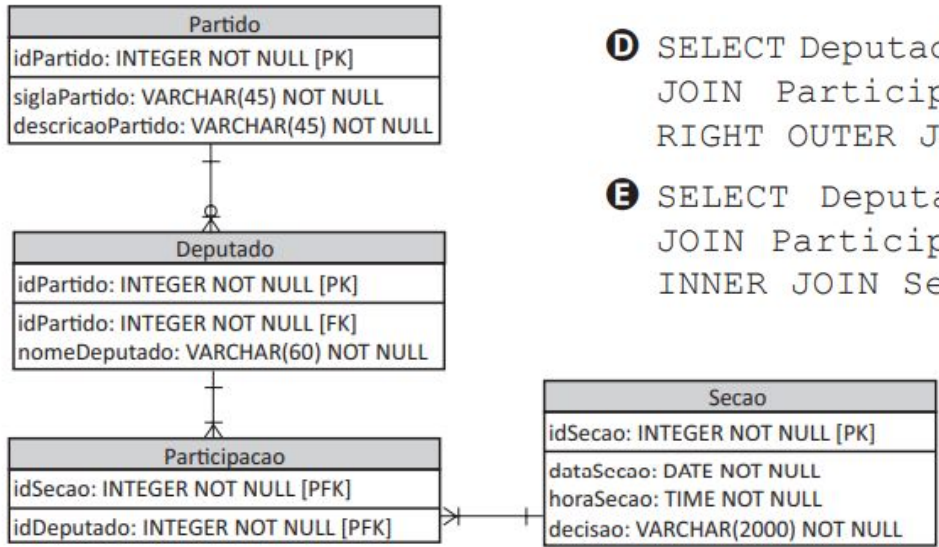
```
SELECT E.nome_estado
FROM Estado AS E, FROM
Fornecedor AS F
WHERE E.nome_estado = F.UF;
```

E

```
SELECT E.nome_estado
FROM Estado AS E
WHERE E.UF IN (
    SELECT F.UF
    FROM Fornecedor AS F);
```

QUESTÃO 19

Considere o diagrama Entidade-Relacionamento apresentado a seguir.



- D** `SELECT Deputado.nomeDeputado, Secao.dataSecao FROM Deputado RIGHT OUTER JOIN Participacao ON Deputado.idDeputado = Participacao.idDeputado RIGHT OUTER JOIN Secao ON Secao.idSecao = Participacao.idSecao;`
- E** `SELECT Deputado.nomeDeputado, Secao.dataSecao FROM Deputado INNER JOIN Participacao ON Deputado.idDeputado = Participacao.idDeputado INNER JOIN Secao ON Participacao.idSecao=Secao.idSecao;`

Qual código SQL exibe o nome de todos os deputados que compareceram a pelo menos uma seção e as datas de cada seção em que os deputados participaram?

- A** `SELECT Deputado.nomeDeputado, Secao.dataSecao FROM Deputado, Participacao, Secao WHERE Deputado.idDeputado=Participacao.idDeputado;`
- B** `SELECT Deputado.nomeDeputado, Secao.dataSecao FROM Deputado, Participacao, Secao WHERE Deputado.idDeputado = Participacao.idDeputado OR Secao.idSecao = Participacao.idSecao;`
- C** `SELECT Deputado.nomeDeputado, Secao.dataSecao FROM Deputado LEFT OUTER JOIN Participacao ON Deputado.idDeputado = Participacao.idDeputado LEFT OUTER JOIN Secao ON Secao.idSecao = Participacao.idSecao;`

Exercício:

QUESTÃO 30

```
CREATE TABLE JOGADOR (PSEUDONIMO VARCHAR(10) NOT NULL,  
NOME VARCHAR(25) NOT NULL,  
SENHA VARCHAR(6) NOT NULL,  
PRIMARY KEY (PSEUDONIMO));
```

```
CREATE TABLE NIVEL(NIVEL NUMERIC(3) NOT NULL,  
NOMEPSEUD VARCHAR(10) NOT NULL,  
PONTOS NUMERIC(5) NOT NULL,  
BONUS NUMERIC(5) NOT NULL,  
PRIMARY KEY(NIVEL, NOMEPSEUD),  
FOREIGN KEY (NOMEPSEUD) REFERENCES JOGADOR);
```

A partir do *script* SQL de criação de um banco de dados acima, assinale a opção que apresenta comando SQL que permite obter uma lista em ordem crescente de quantidade de bônus e que contenha somente o pseudônimo do jogador e seu bônus.

```
-- Alternativa A  
SELECT NOME, BONUS FROM JOGADOR, NIVEL ORDER BY NIVEL.BONUS DESC
```



```
-- Alternativa b  
SELECT * FROM JOGADOR, NIVEL WHERE  
JOGADOR.PSEUDONIMO = NIVEL.NOMEPSEUD ORDER BY NIVEL.BONUS DESC
```

```
-- Alternativa c  
SELECT NOME, BONUS FROM JOGADOR, NIVEL WHERE  
JOGADOR.PSEUDONIMO = NIVEL.NOMEPSEUD ORDER BY NIVEL.BONUS DESC
```

```
-- Alternativa d  
SELECT NOME, PONTOS FROM JOGADOR, NIVEL WHERE  
JOGADOR.PSEUDONIMO = NIVEL.NOMEPSEUD ORDER BY NIVEL.BONUS DESC
```

```
-- Alternativa e  
SELECT NOME, PONTOS FROM JOGADOR, NIVEL  
WHERE JOGADOR.PSEUDONIMO = NIVEL.NOMEPSEUD ORDER BY NIVEL.BONUS ASC
```

Exercício

Operação de Seleção

- 1. Implementar em SQL o Exemplo 1 e Exemplo 2 da operação de seleção.

Operação de Projeção

- 2. Implementar em SQL o Exemplo 3 da operação de projeção.

Operação de Seleção e Projeção

- 3. Implementar em SQL o Exemplo 4 da operação de seleção e projeção.

Exercício

Operação de União

- 4. Implementar em SQL o Exemplo 5 da operação de união.

Operação de Intersecção

- 5. Implementar em SQL o Exemplo 6 da operação de intersecção.

Operação de Diferença

- 6. Implementar em SQL o Exemplo 7 da operação de diferença.

Exercício

Operação de Plano Cartesiano

- 7. Implementar em SQL o Exemplo 8 da operação de plano cartesiano.

Operação de Junção

- 8. Implementar em SQL o Exemplo 9 da operação de junção.

Referências

CHEN, Peter. Modelagem de dados: a abordagem entidade-relacionamento para projeto lógico. São Paulo: Makron Books, 1990.

DATE, C. J. Introdução a sistemas de banco de dados. Rio de Janeiro: Campus, 1991.

DEVMEDIA. SQL Magazine 02. Bancos de Dados Orientados a Objetos: Uma Introdução. Revista SQL Magazine, Edição 2, 2008. Disponível em: <<https://www.devmedia.com.br/artigo-sql-magazine-02-bancos-de-dados-orientados-a-objetos-uma-introducao/7623>> Acesso em: 08 fev. 2023.

ELMASRI, Ramez; NAVATHE, Shamkant B. Sistemas de banco de dados. 4. ed. São Paulo: Pearson Addison Wesley, 2005.

EMANUEL, Arthur. Java: Introdução à Orientação a Objetos. Instituto Federal de São Paulo, IFSP. Disponível em: <<https://pt.slideshare.net/ArthurEmanuel/java-introducao-orientao-a-objetos-66560721>> Acesso em: 09 fev. 2023.

IFSUL. Instituto Federal de Educação, Ciência e Tecnologia Sul-rio-grandense. Modelo Entidade Relacionamento. Disponível em: <<http://tics.ifsul.edu.br/matriz/conteudo/disciplinas/pbdr/ud/1/2.html>> Acesso em: 16 fev. 2023.

INEPa. Exame Nacional de Desempenho dos Estudantes. ENADE 2008. Disponível em: <https://download.inep.gov.br/download/Enade2008_RNP/TECNOLOGIA_DESENVOLVIMENTO_SISTEMAS.pdf> Acesso em: 15 fev. 2023.

INEPb. Exame Nacional de Desempenho dos Estudantes. ENADE 2014. Disponível em: <https://download.inep.gov.br/educacao_superior/enade/provas/2014/40_tecnologia_analise_desenv_sistemas.pdf> Acesso em: 15 fev. 2023.

INEPc. Exame Nacional de Desempenho dos Estudantes. ENADE 2014. Disponível em: <https://download.inep.gov.br/educacao_superior/enade/provas/2014/03_computacao_bacharelado.pdf> Acesso em: 15 fev. 2023.

Referências

HEUSER, Carlos Alberto. Projeto de banco de dados. Porto Alegre: Sagra Luzzatto, 2004.

MASLAKOWSKI, Mark. Aprenda em 21 dias MySQL / Mark Maslakowski, Tony Butcher; tradução de Edson Furmankiewicz, Joana Figueiredo. Rio de Janeiro: Campus, 2000

PRESSMAN, Roger S. Engenharia de software. São Paulo: Makron Books, 1995.

SILVA, Flávio de Oliveira. Projeto Banco de Dados. Faculdade de Computação. Universidade Federal de Uberlândia. Disponível em: <https://www.facom.ufu.br/~flavio/sbd/files/02-sbd-projeto-banco-dados.pdf>

SETZER, Valdemar W.; SILVA, Flávio Soares Corrêa da. Banco de dados: aprenda o que são, melhore seu conhecimento, construa os seus. São Paulo: Edgard Blücher, 2005.

SILBERSCHATZ, Abraham; KORTH, Henry F.; SUDARSHAN, S. Sistema de banco de dados. 3. ed. São Paulo: Makron Books, 1999.

WILL, Newton Carlos. Operações da Álgebra Relacional. Disponível em: https://coens.dv.utfpr.edu.br/will/wp-content/uploads/2022/03/Apostila_Algebra_Relacional.pdf Acesso em: 24 ago. 2023.

Obrigado.

Floriano Ferreira dos Reis Filho