

1. Álgebra Relacional e Operações Relacionais

A álgebra relacional é uma linguagem de consulta procedural. Ela consiste em um conjunto de operações que tomam uma ou duas tabelas como entrada e produzem uma nova tabela como resultado. Essas operações baseiam-se na teoria dos conjuntos (as tabelas correspondem a conjuntos).

- Linguagem procedural: linguagem que requer sempre a existência de definição quanto à ordem em que as operações serão realizadas

1.1 Operadores de Álgebra Relacional

São definidas nove operações ou operadores para se trabalhar com álgebra relacional, eles podem ser classificados da seguinte maneira:

Quanto à sua origem:

- Fundamentais - através dela qualquer expressão de consulta de dados é permitida:

1. *Projeção*
2. *Seleção*
3. *Produto Cartesiano*
4. *União*

5. *Diferença, Subtração*

- Derivados – derivam dos operadores fundamentais, são definidos para facilitar a especificação de certos procedimentos:

6. *Intersecção*
7. *Junção (normal e natural)*
8. *Divisão*

- Especiais – operadores que não se enquadram nos itens anteriores:

9. *Renomeação e alteração*

Quanto ao número de relações (tabelas) operandas:

Unários - operam em uma única tabela. São eles: seleção, projeção, renomeação e alteração;

Binários – operam em duas tabelas. São eles: união, intersecção, diferença, produto cartesiano, junção e divisão.

Quanto à origem da área da matemática:

Teoria dos Conjuntos – operadores usuais da teoria de conjuntos da matemática. São eles: união, intersecção, diferença e produto cartesiano;

Especiais – operadores adicionais, definidos pela álgebra relacional para manipulação de dados. São eles: seleção, projeção, junção, divisão, renomeação e alteração.

Além desses operadores, é definido também o operador de *atribuição* que permite atribuir o resultado de uma expressão de álgebra a uma tabela.

Quadro Resumo de Operações/Operadores em Álgebra Relacional:

OPERAÇÃO	SÍMBOLO	SINTAXE
Projeção	π ("pi")	π <lista de campos> (Tabela)
Seleção/ Restrição	σ ("sigma")	σ <condição de seleção> (Tabela)
União	\cup	(Tabela 1) \cup (Tabela 2)
Intersecção	\cap	(Tabela 1) \cap (Tabela 2)
Diferença	$-$	(Tabela 1) $-$ (Tabela 2)
Produto Cartesiano	\times	(Tabela 1) \times (Tabela 2)
Junção	$ X $	(Tabela 1) $ X $ <condição de junção> (Tabela 2)
Divisão	\div	(Tabela 1) \div (Tabela 2)
Renomeação	ρ ("rho")	ρ Nome(Tabela)
Atribuição	\leftarrow	Variável \leftarrow Tabela

Para nosso trabalho com os operadores relacionais assumiremos nesta apostila o banco de dados definido pelo esquema abaixo:

Tabela Cargo

CdCargo	DescCargo	VlrSalario
C1	Aux.Vendas	350,00
C2	Vigia	400,00
C3	Vendedor	800,00
C4	Aux.Cobrança	250,00
C5	Gerente	1000,00
C6	Diretor	2500,00

Tabela Depto

CdDepto	DescDepto	RamalTel
D1	Assist.Técnica	2246
D2	Estoque	2589
D3	Administração	2772
D4	Segurança	1810
D5	Vendas	2599
D6	Cobrança	2688

Tabela Funcionário

NumReg	NomeFunc	DtAdmissão	Sexo	CdCargo	CdDepto
101	Luis Sampaio	10/08/2003	M	C3	D5
104	Carlos Pereira	02/03/2004	M	C4	D6
134	Jose Alves	23/05/2002	M	C5	D1
121	Luis Paulo Souza	10/12/2001	M	C3	D5
195	Marta Silveira	05/01/2002	F	C1	D5
139	Ana Luiza	12/01/2003	F	C4	D6
123	Pedro Sergio	29/06/2003	M	C7	D3
148	Larissa Silva	01/06/2002	F	C4	D6
115	Roberto Fernandes	15/10/2003	M	C3	D5
22	Sergio Nogueira	10/02/2000	M	C2	D4

Projeção (π)

Pode ser entendida como uma operação que filtra as colunas de uma tabela de nosso banco de dados ou uma tabela resultante de uma outra operação relacional executada. Por operar em apenas um conjunto de entrada, a projeção é classificada como uma operação unária.

Sintaxe: π coluna1, coluna2,..., colunaN (Tabela)

Consulta: Qual o nome e data de admissão dos funcionários?

Comando: π Nomefunc, DtAdmissao (Funcionario)

Resultado:

NomeFunc	DtAdmissão
Luis Sampaio	10/08/2003
Carlos Pereira	02/03/2004
Jose Alves	23/05/2002
Luis Paulo Souza	10/12/2001
Marta Silveira	05/01/2002
Ana Luiza	12/01/2003
Pedro Sergio	29/06/2003
Larissa Silva	01/06/2002
Roberto Fernandes	15/10/2003
Sergio Nogueira	10/02/2000

Seleção/ Restrição (σ)

Pode ser entendida como a operação que filtra, seleciona as linhas de uma tabela, realizando também uma projeção, e opera em um conjunto de dados, sendo portando uma operação unária.

Sintaxe: σ <condição de seleção ou predicado> (Tabela)

Consulta: Quais os funcionários de sexo masculino?

Comando: σ Sexo='M' (Funcionario)

Resultado:

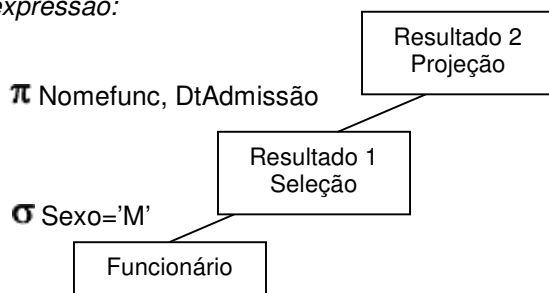
NumReg	NomeFunc	DtAdmissão	Sexo	CdCargo	CdDepto
101	Luis Sampaio	10/08/2003	M	C3	D5
104	Carlos Pereira	02/03/2004	M	C4	D6
134	Jose Alves	23/05/2002	M	C5	D1
121	Luis Paulo Souza	10/12/2001	M	C3	D5
123	Pedro Sergio	29/06/2003	M	C6	D3
115	Roberto Fernandes	15/10/2003	M	C3	D5
22	Sergio Nogueira	10/02/2000	M	C2	D4

Agora se quisermos saber ainda neste exemplo, somente o nome e a data de admissão dos funcionários então começamos a utilizar operações combinadas pois deveremos fazer uma projeção desse resultado.

Consulta: Quais os nomes e data de admissão dos funcionários de sexo masculino?

Comando: π Nomefunc, DtAdmissão (σ Sexo='M' (Funcionário))

Facilitando o entendimento, graficamente teríamos a *árvore de expressão*:



Observe operação abaixo fazendo primeiro a projeção e depois a seleção:

σ Sexo='M' (π Nomefunc, DtAdmissão(Funcionário))

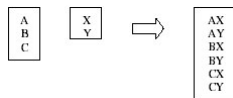
inicialmente parece estar correta, mas isso não é verdade pois ao fazermos a projeção o campo Sexo, necessário à seleção não está compreendido no resultado da projeção causando erro na operação, portanto essa operação **não é válida**.

Resultado:

NomeFunc	DtAdmissão
Luis Sampaio	10/08/2003
Carlos Pereira	02/03/2004
Jose Alves	23/05/2002
Luis Paulo Souza	10/12/2001
Pedro Sergio	29/06/2003
Roberto Fernandes	15/10/2003
Sergio Nogueira	10/02/2000

- Nas consultas complexas para facilitar o entendimento construa a consulta graficamente antes de elaborar o comando em álgebra relacional.

Produto Cartesiano (X)



Utiliza a mesma notação de operação matemática de dois conjuntos, tendo como resultado do produto cartesiano de duas tabelas uma terceira tabela contendo as combinações possíveis entre os elementos das tabelas originais. Essa tabela resultante possui um número de colunas que é igual à soma do número de colunas das tabelas iniciais e um número de linhas igual ao produto do número de linhas das duas tabelas.

Sintaxe: (Tabela 1) X (Tabela 2)

Consulta: Trazer as informações dos funcionários e de seus cargos.

Comando: Funcionarios X Cargos

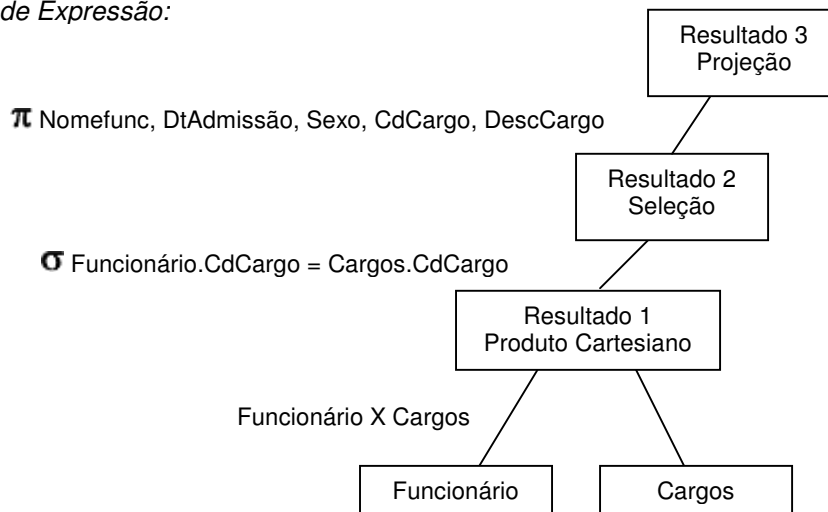
Resultado:

NumReg	NomeFunc	DtAdmissão	Sexo	CdCargo	CdDepto	CdCargo	DescCargo	VlrSalario
101	Luis	10/08/2003	M	C3	D5	C1	Aux.Vendas	350,00
101	Luis	10/08/2003	M	C3	D5	C3	Vendedor	800,00
101	Luis	10/08/2003	M	C3	D5	C6	Diretor	2500,00
101	Luis	10/08/2003	M	C3	D5	C2	Vigia	400,00
101	Luis	10/08/2003	M	C3	D5	C5	Gerente	1000,00
101	Luis	10/08/2003	M	C3	D5	C4	Aux.Cobrança	250,00
104	Carlos	02/03/2004	M	C4	D6	C1	Aux.Vendas	350,00
104	Carlos	02/03/2004	M	C4	D6	C3	Vendedor	800,00
104	Carlos	02/03/2004	M	C4	D6	C6	Diretor	2500,00
104	Carlos	02/03/2004	M	C4	D6	C2	Vigia	400,00
104	Carlos	02/03/2004	M	C4	D6	C5	Gerente	1000,00
104	Carlos	02/03/2004	M	C4	D6	C4	Aux.Cobrança	250,00
.....
.....
22	Sergio	10/02/2000	M	C2	D4	C1	Aux.Vendas	350,00
22	Sergio	10/02/2000	M	C2	D4	C3	Vendedor	800,00
22	Sergio	10/02/2000	M	C2	D4	C6	Diretor	2500,00
22	Sergio	10/02/2000	M	C2	D4	C2	Vigia	400,00
22	Sergio	10/02/2000	M	C2	D4	C5	Gerente	1000,00
22	Sergio	10/02/2000	M	C2	D4	C4	Aux.Cobrança	250,00

Notamos que como nossa operação é um produto cartesiano temos como resultado o produto das linhas das duas tabelas, isso em um projeto lógico para nós está incorreto, pois para nossa consulta as linhas que realmente nos interessam seriam as que os campos CdCargo sejam iguais das duas tabela ou seja $\text{Funcionário.CdCargo} = \text{Cargos.CdCargo}$, aplicamos então uma seleção no resultado:

Consulta: Trazer o nome, data de admissão código do cargo e descrição do cargo dos funcionários.

Árvore de Expressão:



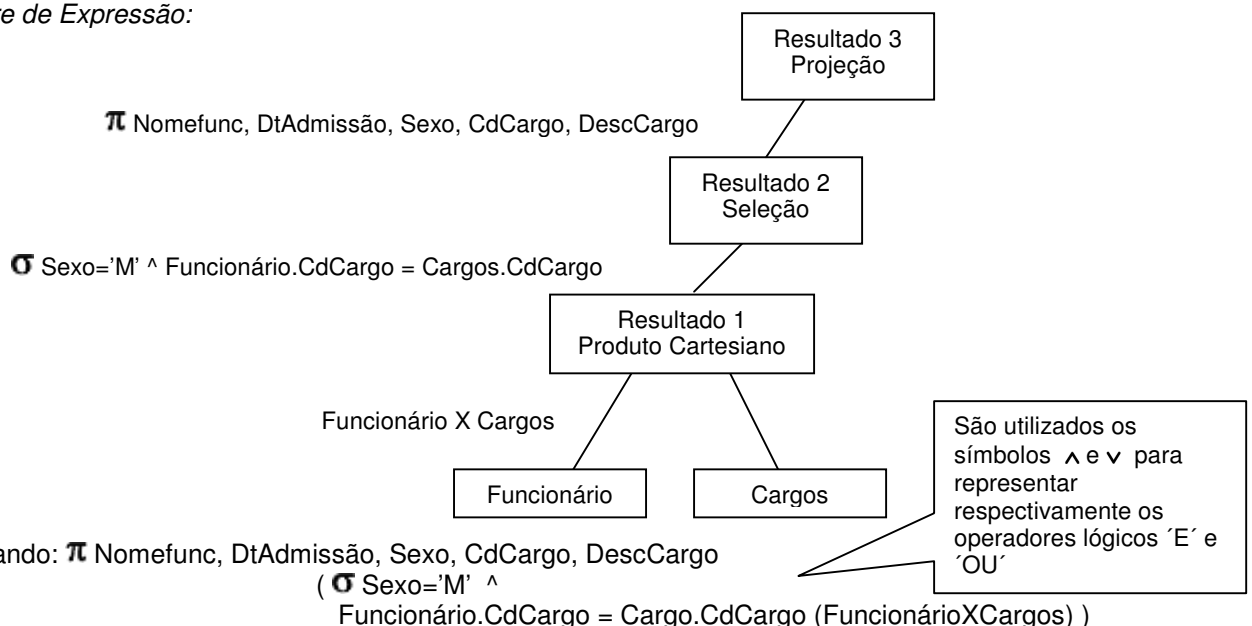
Comando: π Nomefunc, DtAdmissão, Sexo, CdCargo, DescCargo
 (σ Funcionário.CdCargo = Cargos.CdCargo (FuncionárioXCargos))

Resultado:

NomeFunc	DtAdmissão	Sexo	CdCargo	DescCargo
Luis Sampaio	10/08/2003	M	C3	Vendedor
Carlos Pereira	02/03/2004	M	C4	Aux.Cobrança
Jose Alves	23/05/2002	M	C5	Gerente
Luis Paulo Souza	10/12/2001	M	C3	Vendedor
Marta Silveira	05/01/2002	F	C1	Aux.Vendas
Ana Luiza	12/01/2003	F	C4	Aux.Cobrança
Pedro Sergio	29/06/2003	M	C6	Diretor
Larissa Silva	01/06/2002	F	C4	Aux.Cobrança
Roberto Fernandes	15/10/2003	M	C3	Vendedor
Sergio Nogueira	10/02/2000	M	C2	Vigia

Consulta: Trazer o nome, data admissão, cód. do cargo e descrição, dos funcionários do sexo masculino.

Árvore de Expressão:



Comando: π Nomefunc, DtAdmissão, Sexo, CdCargo, DescCargo
 (σ Sexo='M' \wedge Funcionário.CdCargo = Cargo.CdCargo (FuncionárioXCargos))

Resultado:

NomeFunc	DtAdmissão	Sexo	CdCargo	DescCargo
Luis Sampaio	10/08/2003	M	C3	Vendedor
Carlos Pereira	02/03/2004	M	C4	Aux.Cobrança
Jose Alves	23/05/2002	M	C5	Gerente
Luis Paulo Souza	10/12/2001	M	C3	Vendedor
Pedro Sergio	29/06/2003	M	C7	Diretor
Roberto Fernandes	15/10/2003	M	C3	Vendedor
Sergio Nogueira	10/02/2000	M	C2	Vigia

União (\cup)

É uma operação binária, ou seja cria uma tabela a partir de duas outras tabelas união compatíveis levando as linhas comuns e não comuns a ambas. As informações duplicadas aparecerão somente uma vez no resultado.

- Tabelas União Compatíveis: tabelas cuja quantidade, disposição e domínio dos atributos/ campos sejam os mesmos

Sintaxe: (Tabela 1) \cup (Tabela 2)

Comando: Imaginemos um caso onde temos a tabela de Contas e a de Empréstimos e queremos saber quais clientes possuem Conta ou Empréstimo, para isso realizamos as projeções respectivas para que tenhamos somente o campo NomeCliente de cada tabela e realizamos a união desse resultado.

	Projeção 1		Projeção 2		Resultado											
Exemplo:	<table><tr><th>NomeCliente</th></tr><tr><td>Luis Sampaio</td></tr><tr><td>Carlos Pereira</td></tr><tr><td>Jose Alves</td></tr></table>	NomeCliente	Luis Sampaio	Carlos Pereira	Jose Alves	∪	<table><tr><th>NomeCliente</th></tr><tr><td>Luis Sampaio</td></tr><tr><td>Luis Sampaio</td></tr></table>	NomeCliente	Luis Sampaio	Luis Sampaio	=	<table><tr><th>NomeCliente</th></tr><tr><td>Luis Sampaio</td></tr><tr><td>Carlos Pereira</td></tr><tr><td>Jose Alves</td></tr></table>	NomeCliente	Luis Sampaio	Carlos Pereira	Jose Alves
NomeCliente																
Luis Sampaio																
Carlos Pereira																
Jose Alves																
NomeCliente																
Luis Sampaio																
Luis Sampaio																
NomeCliente																
Luis Sampaio																
Carlos Pereira																
Jose Alves																

Interserção (\cap)

É uma operação binária, ou seja cria uma tabela a partir de duas outras tabelas levando sem repetição as linhas, que pertençam a ambas as tabelas presentes na operação.

Sintaxe: (Tabela 1) \cap (Tabela 2)

Comando: Imaginemos um caso onde temos a tabela de Contas e a de Empréstimos e queremos saber quais clientes possuem Conta e Empréstimo, para isso realizamos as projeções respectivas para que tenhamos somente o campo NomeCliente de cada tabela e realizamos a intersecção desse resultado.

	Projeção 1		Projeção 2		Resultado
Exemplo:	NomeCliente	∩	NomeCliente	=	NomeCliente
	Luis Sampaio		Luis Sampaio		Luis Sampaio
	Carlos Pereira		Luis Sampaio		
	Jose Alves				

Diferença ($-$)

Essa operação permite encontrarmos linhas que estão em uma tabela mas não estão em outra. A expressão Tabela 1 – Tabela 2 resulta em uma tabela que contém todas as linhas que estão na tabela 1 e não estão na Tabela 2. Observamos que Tabela 1 – Tabela 2 (Exemplo1) é diferente de Tabela 2 – Tabela 1 (Exemplo2).

Sintaxe: (Tabela 1) $-$ (Tabela 2)

Comando: Imaginemos um caso onde temos a tabela de Contas e a de Empréstimos e queremos saber quais clientes possuem uma Conta mas não tenham um Empréstimo, para isso realizamos as projeções respectivas para que tenhamos somente o campo NomeCliente de cada tabela e realizamos a diferença desse resultado.

	Projeção 1		Projeção 2		Resultado										
Exemplo1:	<table><tr><th>NomeCliente</th></tr><tr><td>Luis Sampaio</td></tr><tr><td>Carlos Pereira</td></tr><tr><td>Jose Alves</td></tr></table>	NomeCliente	Luis Sampaio	Carlos Pereira	Jose Alves	-	<table><tr><th>NomeCliente</th></tr><tr><td>Luis Sampaio</td></tr><tr><td>Luis Sampaio</td></tr></table>	NomeCliente	Luis Sampaio	Luis Sampaio	=	<table><tr><th>NomeCliente</th></tr><tr><td>Carlos Pereira</td></tr><tr><td>Jose Alves</td></tr></table>	NomeCliente	Carlos Pereira	Jose Alves
NomeCliente															
Luis Sampaio															
Carlos Pereira															
Jose Alves															
NomeCliente															
Luis Sampaio															
Luis Sampaio															
NomeCliente															
Carlos Pereira															
Jose Alves															

Exemplo2:

Projeção 2	–	Projeção 1	=	Resultado (vazio)
NomeCliente		NomeCliente		NomeCliente
Luis Sampaio		Luis Sampaio		
Luis Sampaio		Carlos Pereira		
		Jose Alves		

Junção (\bowtie)

Essa operação interage com o modelo relacional, ou seja trabalha com o modelo de relações entre tabelas realizando um produto cartesiano, combinando as linhas e somando as colunas de duas tabelas, só que partindo de campos comuns de ambas para realizar essa “seleção relacional”.

Essa operação possui uma condição onde se colocam os campos das tabelas que estão sendo usados para se efetivar a junção. Chamamos essa junção de junção com predicado. (Sintaxe 1)

Outro tipo de junção é a junção natural (Sintaxe 2). Nela não há especificação de condição sendo usado para isso todas as colunas comuns às duas tabelas. As colunas resultantes são a soma das colunas das duas tabelas sem a repetição das colunas idênticas (aparecerão uma vez somente). Não deve ser empregada quando se deseja associar duas tabelas apenas por um ou alguns dos seus atributos idênticos, caso isso seja feito os resultados são imprevisíveis.

Sintaxe1 (Junção com Predicado): (Tabela 1) \bowtie <condição de junção> (Tabela 2)

Sintaxe2 (Junção Natural): (Tabela 1) \bowtie (Tabela 2)

Ex: (Tabela 1) \bowtie <condição de junção> (Tabela 2)

Consulta: Trazer o nome, data de admissão e cargo dos funcionários do sexo masculino.

Comando: π Nomefunc, DtAdmissão, NumCargo

(σ Sexo='M' (Funcionário \bowtie Funcionário.CdCargo = Cargo.CdCargo Cargos))

Comando: π Nomefunc, DtAdmissão, NumCargo (σ Sexo='M' (Funcionário \bowtie Cargos))

Comando utilizando junção com predicado

Comando utilizando junção natural

Resultado:

NomeFunc	DtAdmissão	DescCargo
Luis Sampaio	10/08/2003	Vendedor
Carlos Pereira	02/03/2004	Aux.Cobrança
Jose Alves	23/05/2002	Gerente
Luis Paulo Souza	10/12/2001	Vendedor
Pedro Sergio	29/06/2003	Diretor
Roberto Fernandes	15/10/2003	Vendedor
Sergio Nogueira	10/02/2000	Vigia

Divisão (\div)

Essa operação produz como resultado a projeção de todos os elementos da primeira tabela que se relacionam com todos os elementos da segunda tabela. Essa operação também pode ser obtida através de outras operações de álgebra relacional.

Sintaxe: (Tabela 1) \div (Tabela 2)

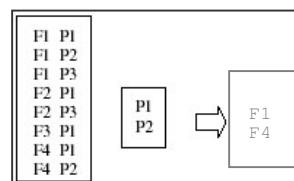
Utilizando as tabelas abaixo como exemplo:

Tabela Equipe

NumReg	CdProjeto
101	Projeto1
104	Projeto1
134	Projeto1
101	Projeto2
104	Projeto2
101	Projeto3

Tabela Projeto

CdProjeto	Descrição
Projeto1	Suporte
Projeto2	Manutenção
Projeto3	Desenvolvimento



Consulta: Quais os funcionários que trabalham em todos os projetos?

Comando: $(\pi$ NumReg, CdProjeto (EQUIPE)) \div $(\pi$ CdProjeto (PROJETOS))

Projeção 1	\div	Projeção 2	=	Resultado
NumReg		CdProjeto		NumReg
101		Projeto1		101
104		Projeto2		
134		Projeto3		
101				
104				
101				

O operando dividendo é a tabela resultante da Projeção 1, o operando divisor é a tabela resultante da Projeção 2, e a tabela de Resultado é o quociente da operação.

Renomeação (ρ)

É a operação que renomeia uma tabela.

Sintaxe: ρ Nome(Tabela) Exemplo: ρ Funcionario2(Funcionario)

Atribuição (\leftarrow)

É utilizada para simplificar comandos muito extensos definindo então passos de comando.

Sintaxe: Variável \leftarrow Tabela Exemplo: Nomes_Func $\leftarrow \pi$ Nomefunc(Funcionario)

1.2. Funções em Álgebra Relacional

Algumas operações muito comuns sobre base de dados não podem ser executadas usando apenas as operações de álgebra relacional e são implementadas na linguagem de consulta à base de dados (SQL) dos SGBD (Sistemas Gerenciadores de Banco de Dados).

Funções de Agregação: aplicadas em uma coleção de valores numéricos: SOMA, MÉDIA, MÁXIMO, MÍNIMO ou usada para contar linhas: CONTADOR.

BIBLIOGRAFIA

- MACHADO, Felipe Nery Rodrigues – Banco de Dados: Projeto e Implementação – Editora Érica;

EXERCÍCIOS

- 1- Dados o esquema abaixo, especificar as operações da álgebra relacional para as solicitações. Na resolução do exercício demonstrar a árvore de expressão:

FORNECEDOR (CD-FORN, NOME-FORN, CD-CONDICAO)
PRODUTO (CD-PROD, NOME-PROD, COR-PROD)
PEDIDO (CD-FORN, CD-PROD, QTD-PEDIDA)

Onde: PEDIDO.CD-FORN referencia FORNECEDOR.CD-FORN
 PEDIDO.CD-PROD referencia PRODUTO.CD-PROD

- Traga o código e nome dos fornecedores;
 - Traga o código e nome dos fornecedores que estão inativos (CD-CONDICAO='I');
 - Traga as informações dos fornecedores que forneceram o produto de código 'PD-23' e estão ativos (CD-CONDICAO='A'); (utilizar junção com predicado)
 - Qual o nome dos fornecedores que forneceram pelo menos um produto cuja a cor é 'VERDE'; (utilizar junção natural)
 - Qual o nome dos produtos de cor 'VERMELHA' com pedidos em quantidade superior a 5000;
 - Qual o código e nome dos fornecedores que forneceram algum produto. (trazer as linhas de resultado sem repetição)
- 2- Dados o esquema abaixo, especificar as operações da álgebra relacional para as solicitações. Na resolução do exercício demonstrar a árvore de expressão e o resultado final da solicitação:

ALUNO (CD-ALUNO, NOME-ALUNO, END-ALUNO, SEXO-ALUNO)
DISCIPLINA (CD-DISC, NOME-DISC, CH-DISC, CD-PROF)
HISTORICO (CD-ALUNO, CD-DISC, NOTA)
PROFESSOR (CD-PROF, NOME-PROFESSOR)

Onde: DISCIPLINA.CD-PROF referencia PROFESSOR.CD-PROF
 HISTORICO.CD-ALUNO referencia ALUNO.CD-ALUNO
 HISTORICO.CD-DISC referencia DISCIPLINA.CD-DISC

ALUNO

CD-ALUNO	Nome-Aluno	End-Aluno	Sexo-Aluno
1	Carlos	Rua A	M
2	Ana	Rua B	F
3	Pedro	Rua C	M

DISCIPLINA

CD-Disc	Nome-Disc	Ch-Disc	CD-Prof
CI	Cálculo-I	40	1
MD	Modelagem	80	2
LBD	Lab.BDs	80	2

HISTORICO

CD-Aluno	CD-Disc	Nota
1	CI	6,0
1	MD	7,0
2	CI	8,0

PROFESSOR

CD-Prof	Nome-Prof
1	Paulo
2	Antonio

- Traga o código, nome e sexo dos alunos;
- Traga o código, nome e sexo dos alunos do sexo masculino;
- Traga o código da disciplina, nome dela e nome dos professores que a lecionam;
- Traga o código da disciplina e nome dos professores que lecionam a disciplina de nome 'Cálculo-I';
- Traga as informações dos alunos que tiveram nota superior a 6 nas disciplinas de Modelagem e Cálculo-I?
- Qual o nome e código dos alunos que cursaram todas as disciplinas com carga horária (CH-DISC) maior que 50 horas, com nota superior a 7?
- Quais nomes de disciplinas onde todos os alunos que já a cursaram obtiveram nota superior a 5?