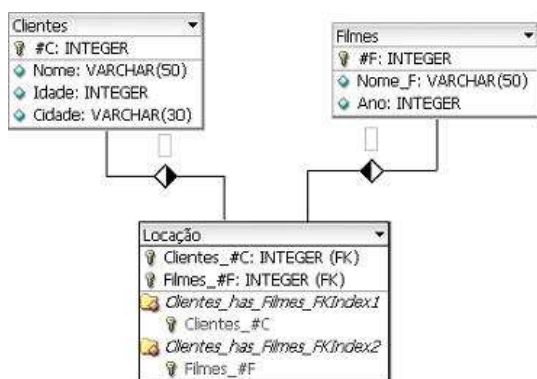


## Álgebra Relacional

A Álgebra Relacional é uma linguagem de consulta procedural (o usuário descreve os passos a serem executados) e formal a qual a técnica utilizada é fundamental para a extração de dados de um banco de dados, além de ser um conjunto de operações, os quais utilizam como recurso de entrada uma ou mais relações, produzindo então, uma nova relação. Neste contexto uma seqüência de operações forma uma expressão em álgebra relacional cujo resultado também produzirá uma relação (uma consulta).

Na Álgebra Relacional existem os operadores tradicionais e os operadores relacionais, onde cada um apresenta a sua função e uma representação na construção de consultas. Para estudarmos as operações de Álgebra relacional deste artigo, apresenta-se abaixo um modelo de banco de dados elaborado a partir da ferramenta DBDesigner):



**Figura 1** – Modelo E-R elaborado no DBDesigner.

A seguir visualizaremos as respectivas tabelas “populadas”, as quais servirão de modelo para o aprendizado de álgebra relacional.

<u>#F</u>	NOME F	ANO
F1	Titanic	1997
F2	Conan	1984
F3	Harry Potter I	2001
F4	Duro de Matar I	2000
F5	Uma mente Brilhante	2001
F6	Independence Day	1996

<u>#C</u>	#F
C1	F1
C1	F2
C1	F3
C1	F4
C1	F5
C1	F6
C2	F3
C2	F6

<u>#C</u>	NOME	IDADE	CIDADE
C1	Artur	22	São Paulo
C2	Catarina	18	Rio de Janeiro
C3	Daniel	36	Petrópolis
C4	Gabriela	44	Teresópolis
C5	Sophia	56	Petrópolis
C6	Maria Eduarda	19	Rio de Janeiro

C3	F1
C3	F3
C3	F6
C4	F2
C4	F4
C5	F2
C5	F2
C6	F4
C6	F5
C6	F6

Obs.: Os campos sublinhados são as chaves primárias e os em itálico as chaves estrangeiras.

## Operadores

### União (Union)

É uma operação binária que cria uma relação partindo de duas outras relações, estabelecendo assim, a união de todas as tuplas dessa nova relação. Seu símbolo é: **U**.

Exemplo: R = conjunto de tuplas de clientes de Petrópolis

S = conjunto de tuplas dos clientes que alugaram o filme “F2”

C = R **U** S

#C	NOME	IDADE	CIDADE
C1	Artur	22	São Paulo
C3	Daniel	36	Petrópolis
C4	Gabriela	44	Teresópolis
C5	Sophia	56	Petrópolis

### Intersecção (Intersection)

É uma operação binária que cria uma relação partindo de duas outras relações, estabelecendo assim a intersecção das duas, gerando uma nova relação as quais estarão presentes apenas os dados pertencentes às duas condições pré-estabelecidas. Seu símbolo é:  $\cap$ .

$$C = R \cap S$$

#C	NOME	IDADE	CIDADE
C5	Sophia	56	Petrópolis

### Diferença (Difference)

É uma operação que cria uma relação partindo de duas outras relações, o qual resultará em uma nova relação que contenha todas as tuplas que pertençam à primeira relação e não pertença à segunda.

Seu símbolo é:  $-$ .

$$C = R - S$$

#C	NOME	IDADE	CIDADE
C1	Artur	22	São Paulo
C3	Daniel	36	Petrópolis
C4	Gabriela	44	Teresópolis

### Produto Cartesiano (Product)

Operação binária que permite a combinação de informações de duas relações quaisquer, assim a relação resultante conterá todas as tuplas da primeira relação relacionadas às da segunda relação. Seu símbolo é:  $\times$ .

Exemplo: K = conjunto de todos os códigos dos moradores de "Petrópolis"

L = conjunto de todos os códigos de filmes de 2001

$$C = K \times L$$

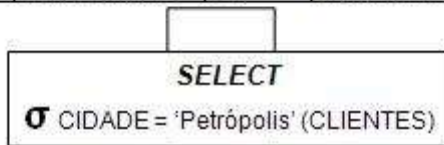
#C	#F
C3	F3
C3	F5
C5	F3
C5	F5

### Seleção (Select)

Operação unária, pois atua sobre uma única relação, produzindo um subconjunto apenas com os elementos especificados na condição. É conhecida também como uma operação que filtra linhas de uma tabela, através da construção de um subconjunto horizontal de uma relação. Seu símbolo é:  $\sigma$ .

Exemplo:  $\sigma$  CIDADE = 'Petrópolis' (CLIENTES)

#C	NOME	IDADE	CIDADE
C1	Artur	22	São Paulo
C2	Catarina	18	Rio de Janeiro
C3	Daniel	36	Petrópolis
C4	Gabriela	44	Teresópolis
C5	Sophia	56	Petrópolis
C6	Maria Eduarda	19	Rio de Janeiro



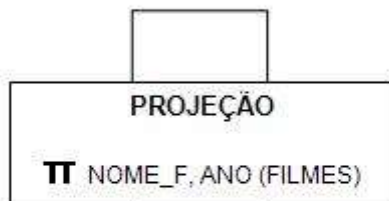
#C	NOME	IDADE	CIDADE
C3	Daniel	36	Petrópolis
C5	Sophia	56	Petrópolis

### Projeção (Projection)

Esta operação também é unária. Produz um subconjunto com apenas as colunas escolhidas na condição especificada, filtrando colunas as quais se coloca na operação o nome das colunas desejadas para a criação do novo conjunto. Este tipo de relação é vertical e o seu símbolo é:  $\pi$ .

Exemplo:  $\pi$  NOME\_F, ANO (FILMES)

#F	NOME_F	ANO
F1	Titanic	1997
F2	Conan	1984
F3	Harry Potter I	2001
F4	Duro de Matar 1	2000
F5	Uma mente Brilhante	2001
F6	Independence Day	1996

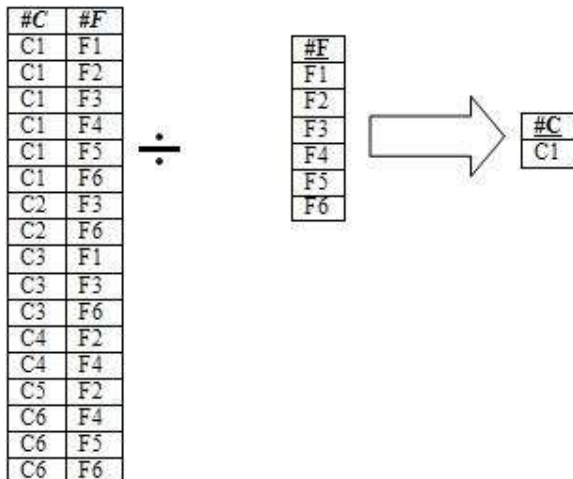


NOME_F	ANO
Titanic	1997
Conan	1984
Harry Potter I	2001
Duro de Matar 1	2000
Uma Mente Brilhante	2001
Independence Day	1996

### Divisão (Division)

Operação binária, a qual resultará em uma tabela que conterá apenas os elementos da primeira tabela que se relacionem com todos os elementos da segunda tabela. Seu símbolo é:  $\div$ .

Exemplo:  $\pi$  #C (LOCAÇÃO  $\div$  ( $\pi$  #F (FILMES)))



Neste exemplo utilizamos também a operação projeção com seu respectivo símbolo já explicado anteriormente.

### · Junção (Join)

Esta operação tem como objetivo unir duas tabelas, as quais possuem um atributo em comum. Este tipo de operação é muito utilizado quando tratamos de relacionamentos com chaves estrangeiras, em que é realizada uma seleção a partir dos atributos de um relacionamento sobre um produto cartesiano das tabelas. Seu símbolo é:  $\bowtie$ .

Clientes  $\bowtie$  Locação =  $\sigma$  Clientes.#C = Locação.#C (Clientes  $\times$  Locação)

Exemplo:  $\pi$  nome, idade ( $\sigma$  #F = F2 (Clientes  $\bowtie$  Locação))

NOME	IDADE
Artur	22
Gabriel	44
Sophia	56
Sophia	56

É apresentado os nomes e as idades dos clientes que alugaram o filme de código "F2". Observa-se a repetição da cliente Sophia, pois esta realiza duas locações do mesmo filme. Para resolvermos este problema, pode-se utilizar a operação de Junção Natural (Natural Join), na qual obtemos o mesmo resultado, porém sem repetições.

NOME	IDADE
Artur	22
Gabriel	44
Sophia	56

### Renomear (rename).

Esta operação redefine um nome para um relacionamento, muito utilizada em casos que contêm expressões relacionais grandes. Dessa forma podemos realizar a divisão da expressão em duas ou mais sub-expressões para se obter o resultado esperado de forma mais simples. Sua simbologia é:  $\rho$  ou  $\rho$ .

Exemplo:  $\rho$ Temp ( $\pi$  nome, idade ( $\sigma$  #F = F2 (Clientes  $\bowtie$  Locação)))

### Temp

NOME	IDADE
Artur	22
Gabriel	44
Sophia	56

### Atribuição (Assignment)

É uma operação utilizada para atribuir a uma variável um determinado relacionamento. Seu símbolo é: <---.

Exemplo:  $A \leftarrow (\pi \text{ nome, idade } (\sigma \#F = F2 (\text{Clientes } |x| \text{ Locação})))$

$B \leftarrow (\pi \text{ nome, idade } (\sigma \text{ idade} = 22 (\text{Clientes})))$

$A \cap B$

Com o uso dos operadores de álgebra relacional, é possível fazer uma simulação dos resultados a serem obtidos em uma consulta SQL. Logo, verifica-se a importância da álgebra relacional como um instrumento para o aprendizado de operações a serem realizadas em um banco de dados relacional, a partir de conceitos matemáticos da teoria dos conjuntos.

### Cálculo Relacional

O Cálculo Relacional é uma linguagem de consulta que descreve a resposta desejada em um banco de dados sem especificar como para obter que, ao contrário da álgebra relacional é tipo processual, cálculo relacional é declaratória; mas sempre os dois métodos alcançam os mesmos resultados.

Relação com lógica de primeira ordem

Podemos definir uma fórmula baseada em combinações de fórmulas atômicas.

Uma fórmula atômica é uma combinação de variáveis (tipo tupla ou tipo de domínio, conforme apropriado) e atributos ou constantes, graças ao uso de operadores como <, >, =, !=, <=, >=.

É também uma fórmula atômica variável  $\in$  Relação.

Combinações de fórmulas atômicas são geradas a partir do uso de operadores como NOT ( $\neg$ ), AND ( $\wedge$ ), OR ( $\vee$ ),  $\rightarrow$ .

Os quantificadores  $\exists$ ,  $\forall$  limitam uma variável.

Classes de cálculo relacional

Podemos distinguir, pelo menos, duas classes de cálculo relacional:

Cálculo relacional baseado em tuplas (CRT).

Uma consulta no CRT é da forma:

$\{T \mid \varphi(T)\}$

onde T é uma variável do tipo tupla e  $\varphi(T)$  é uma fórmula que descreve T. O resultado dessa consulta é o conjunto de todas as tuplas t para as quais a fórmula é verdadeira.

Variável do tipo de tupla

Uma variável de tipo de tupla T é uma variável capaz de assumir qualquer valor de tupla que pertença a uma relação (ou tabela).

Consulta de sintaxe no TRC

A sintaxe é definida a partir da lógica de primeira ordem. Onde a variável a ser usada é do tipo tupla. Uma variável está livre em uma fórmula (ou sub-fórmula) se a fórmula (sub) não contiver nenhuma ocorrência de quantificadores que a limitem. Em uma consulta CRT do formulário:  $\{T \mid \varphi(T)\}$ , T é a única variável livre.

Cálculo relacional baseado em domínios (DRC)

Ele é constituído com os mesmos operadores que o cálculo relacional de tuplas, mas não há tuplas, mas variáveis de domínio. As expressões do cálculo relacional de domínios são da forma  $\{(x, y, z, \dots) \mid P(x, y, z, \dots)\}$ , onde x, y, z representam as variáveis de domínio, P representa uma fórmula composta

