

Álgebra Relacional

Carina F. Dorneles
dorneles@inf.ufsc.br

Parte I

Linguagens de consulta a Banco de dados

- Permitem **manipulação** e **recuperação** de dados de um BD
- No modelo relacional
 - Forte **fundamentação teórica** baseada em lógica
 - Permite **otimizações**

Linguagens de consulta	<u>VS</u>	Linguagens de programação
Não objetiva construção de cálculos		Suporta cálculos
Acesso a dados		Implementação de algoritmos
Não há laços e condições		<i>If-then-else; while; if; case...</i>



Linguagens de consulta – modelo relacional

- Base

- Álgebra relacional

- Predominantemente **operacional**, útil para representar planos de execução

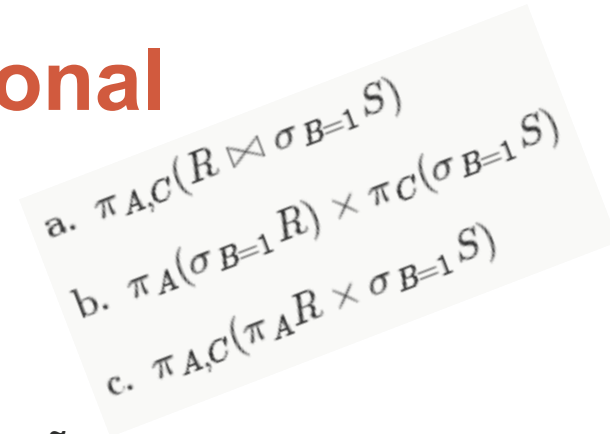
- Cálculo Relacional

- Usuários descreverem **o que** querem, e **não como** querem (**declarativa**)

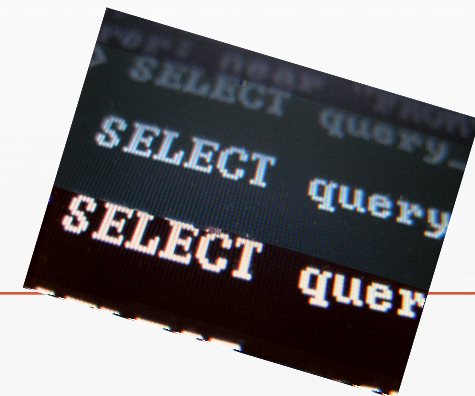
VS

- Prática

- SQL (**declarativa**)



a. $\pi_{A,C}(R \bowtie \sigma_{B=1} S)$
b. $\pi_A(\sigma_{B=1} R) \times \pi_C(\sigma_{B=1} S)$
c. $\pi_{A,C}(\pi_A R \times \sigma_{B=1} S)$



SELECT query
SELECT query
SELECT query

Álgebra relacional

- Linguagem de consulta **formal**
 - Baseada na **lógica de primeira ordem**
 - Coleção de **operações** de alto nível sobre **relações** ou conjuntos
 - Entrada: relação
 - Saída: relação
 - Consultas **SQL** são transformadas em **árvores algébricas** antes de serem executadas no BD
-

Operações

Projeção

Seleção

União

Intersecção

Diferença

Produto Cartesiano

A	Y	Y
A	Y	Z
B	Y	Y
B	Y	Z
C	Z	Y
C	Z	Z

Junção

A	Y	Y
B	Y	Y
C	Z	Z

Divisão

A	Y
A	Z
B	Z
C	Y

Operadores

- Seleção - σ
 - Projeção - π
 - Produto Cartesiano - \times
 - Renomeação - ρ
 - Junção - \Join
 - Diferença - $-$
 - União - \cup
 - Intersecção \cap
 - Divisão \div
-

Banco de dados Exemplo



Livro

<u>Codigo</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700

Pessoa

<u>Codigo</u>	Nome	Idade	fone	CodEsposa	Sexo
PE02	Aninha	23	9999.9999	NULL	F
PE10	Paulinho	20	8888.8888	NULL	M
PE87	Juca	34	7777.7777	PE02	M
PE23	Luana	30	6666.6666	NULL	F
PE54	Beto	28	5555.5555	PE23	M

Empréstimo

<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	PE02	10/10/2000	8:00
LI670	PE02	10/10/2000	8:00
LI340	PE23	01/11/2000	11:50
LI003	PE54	20/11/2000	10:00
LI005	PE10	11/11/2000	14:00
LI670	PE87	23/05/2001	16:15

Seleção - σ

- **Seleciona linhas** de uma relação – operação **unária**
- Sintaxe:

σ <condicao de selecao> <(relação)>

- Onde:
 - <relação>: **nome da relação** que se deseja recuperar dados
 - <condicao de selecao>: **expressão booleana** que envolve literais e valores de atributos da tabela
 - O parâmetro <relação> **pode ser outra expressão algébrica**, pois uma expressão algébrica retorna uma relação
-

Seleção - exemplo

- Q1: Recuperar livros de 2000.

Livro

<u>Codigo</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700

Seleção - exemplo

- Q1: Recuperar livros de 2000.

Livro

<u>Codigo</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700

$\sigma_{ano=2000} (livro)$

Livro

<u>Codigo</u>	Título	Ano	NrPaginas
LI670	Introdução a Banco de Dados	2000	500
LI003	Algoritmos e Lógica	2000	700

Seleção - exemplos

- Q2: Pessoas do sexo feminino e com mais de 25 anos

$\sigma_{\text{sexo}=\text{'F'} \wedge \text{idade} > 25} (\text{pessoa})$

- Q3: Pessoas do sexo feminino ou que tenham mais de 25 anos

$\sigma_{\text{sexo}=\text{'F'} \vee \text{idade} > 25} (\text{pessoa})$

Projeção - π

- Projeta coluna de uma relação – operação **unária**
- Sintaxe:

π `<atributo1,..., atributon >` `<(relação)>`

- Onde:
 - `<relação>`: **nome da relação** que se deseja projetar os atributos
 - `<atributo1,..., atributon >`: **atributos** que se deseja recuperar
 - O parâmetro `<relação>` **pode ser outra expressão algébrica**, pois uma expressão algébrica retorna uma relação
-

Projeção - exemplo

- Q4: Recuperar título e NrPaginas dos livros.

Livro

<u>Codigo</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700



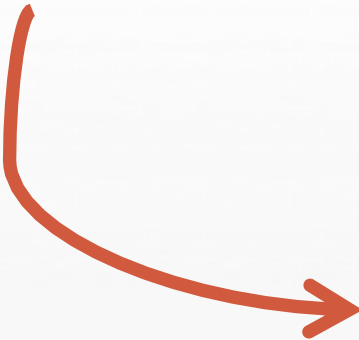
Projeção - exemplo

- Q4: Recuperar título e NrPaginas dos livros.

Livro

<u>Codigo</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700

π titulo, NrPaginas (livro)



Livro	
Título	NrPaginas
Web e Banco de dados	330
Introdução a Banco de Dados	500
Programação C	250
Algoritmos e Lógica	700

Projeção - exemplos

- Q5: Selecionar nome e idade das pessoas

$$\pi_{\text{nome, idade}}(\text{pessoa})$$

- Q6: Recuperar data dos empréstimos

$$\pi_{\text{data}}(\text{empréstimo})$$

Seleção e Projeção juntos

- Q7: Recuperar o título e ano dos livros com mais de 350 páginas

Livro

<u>Codigo</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700



Seleção e Projeção juntos

- Q7: Recuperar o título e ano dos livros com mais de 350 páginas

Livro

<u>Codigo</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700



π titulo, ano (σ nrPaginas > 350 (Livro))

Execução no Banco de dados



A árvore de consulta é executada de baixo para cima

- primeiro **acessa a tabela**
- depois executa **seleção**
- em seguida a **projeção**

Seleção sempre é executada **antes** da **projeção**

Produto Cartesiano - X

- Operação **binária**
- A tabela resultado possui:
 - (número de colunas da primeira tabela + número de colunas da segunda tabela) **colunas**
 - (número de linhas da primeira tabela x número de linhas da segunda tabela) **linhas**
- Sintaxe:

<relação> X <relacao>

<relação> pode ser outra expressão algébrica



- Produz como resultado uma tabela que contém todos os campos da primeira tabela, seguidos de todos os campos da segunda tabela.
-

Produto Cartesiano - exemplo

- Q8: Recuperar dados dos livros emprestados

Livro X emprestimo

Livro

<u>Codigo</u>	<u>Título</u>	<u>Ano</u>	<u>NrPaginas</u>
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700
LI999	Introdução à Computação	2010	200

Empréstimo

<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	PE02	10/10/2000	8:00
LI670	PE02	10/10/2000	8:00
LI340	PE23	01/11/2000	11:50
LI003	PE54	20/11/2000	10:00
LI005	PE10	11/11/2000	14:00
LI670	PE87	23/05/2001	16:15

Livro X empréstimo

<u>Codigo</u>	<u>Título</u>	<u>Ano</u>	<u>NrPaginas</u>	<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	Web e Banco de dados	2013	330	LI005	PE02	10/10/2000	8:00
LI005	Web e Banco de dados	2013	330	LI670	PE02	10/10/2000	8:00
LI005	Web e Banco de dados	2013	330	LI340	PE23	01/11/2000	11:50
LI005	Web e Banco de dados	2013	330	LI003	PE54	20/11/2000	10:00
LI005	Web e Banco de dados	2013	330	LI005	PE10	11/11/2000	14:00
LI005	Web e Banco de dados	2013	330	LI670	PE87	23/05/2001	16:15
LI670	Introdução a Banco de Dados	2000	500	LI005	PE02	10/10/2000	8:00
LI670	Introdução a Banco de Dados	2000	500	LI670	PE02	10/10/2000	8:00
LI670	Introdução a Banco de Dados	2000	500	LI340	PE23	01/11/2000	11:50
LI670	Introdução a Banco de Dados	2000	500	LI003	PE54	20/11/2000	10:00
...

Livro X empréstimo



<u>Codigo</u>	<u>Título</u>	<u>Ano</u>	<u>NrPaginas</u>	<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	Web e Banco de dados	2013	330	LI005	PE02	10/10/2000	8:00
LI005	Web e Banco de dados	2013	330	LI670	PE02	10/10/2000	8:00
LI005	Web e Banco de dados	2013	330	LI340	PE23	01/11/2000	11:50
LI005	Web e Banco de dados	2013	330	LI003	PE54	20/11/2000	10:00
LI005	Web e Banco de dados	2013	330	LI005	PE10	11/11/2000	14:00
LI005	Web e Banco de dados	2013	330	LI670	PE87	23/05/2001	16:15
LI670	Introdução a Banco de Dados	2000	500	LI005	PE02	10/10/2000	8:00
LI670	Introdução a Banco de Dados	2000	500	LI670	PE02	10/10/2000	8:00
LI670	Introdução a Banco de Dados	2000	500	LI340	PE23	01/11/2000	11:50
LI670	Introdução a Banco de Dados	2000	500	LI003	PE54	20/11/2000	10:00
...

Livro X empréstimo

<u>Codigo</u>	<u>Título</u>	<u>Ano</u>	<u>NrPaginas</u>	<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	Web e Banco de dados	2013	330	LI005	PE02	10/10/2000	8:00
LI005	Web e Banco de dados	2013	330	LI670	PE02	10/10/2000	8:00
LI005	Web e Banco de dados	2013	330	LI340	PE23	01/11/2000	11:50
LI005	Web e Banco de dados	2013	330	LI003	PE54	20/11/2000	10:00
LI005	Web e Banco de dados	2013	330	LI005	PE10	11/11/2000	14:00
LI005	Web e Banco de dados	2013	330	LI670	PE87	23/05/2001	16:15
LI670	Introdução a Banco de Dados	2000	500	LI005	PE02	10/10/2000	8:00
LI670	Introdução a Banco de Dados	2000	500	LI670	PE02	10/10/2000	8:00
LI670	Introdução a Banco de Dados	2000	500	LI340	PE23	01/11/2000	11:50
LI670	Introdução a Banco de Dados	2000	500	LI003	PE54	20/11/2000	10:00
...

Livro X empréstimo

Linhas inconsistentes devem ser eliminadas

<u>Codigo</u>	<u>Título</u>	<u>Ano</u>	<u>NrPaginas</u>	<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	Web e Banco de dados	2013	330	LI005	PE02	10/10/2000	8:00
LI005	Web e Banco de dados	2013	330	LI670	PE02	10/10/2000	8:00
LI005	Web e Banco de dados	2013	330	LI340	PE20	04/11/2000	11:50
LI005	Web e Banco de dados	2013	330	LI003	PE54	20/11/2000	10:00
LI005	Web e Banco de dados	2013	330	LI005	PE10	11/11/2000	14:00
LI005	Web e Banco de dados	2013	330	LI670	PE87	23/05/2001	16:15
LI670	Introdução a Banco de Dados	2000	500	LI005	PE02	10/10/2000	8:00
LI670	Introdução a Banco de Dados	2000	500	LI670	PE02	10/10/2000	8:00
LI670	Introdução a Banco de Dados	2000	500	LI340	PE20	04/11/2000	11:50
LI670	Introdução a Banco de Dados	2000	500	LI003	PE54	20/11/2000	10:00
...

Produto Cartesiano - X

- Linhas **inconsistentes** devem ser **eliminadas**
- Fazer a **seleção** das linhas cujo valor da **PK de uma relação** seja igual ao valor de sua referência na **FK da outra relação**

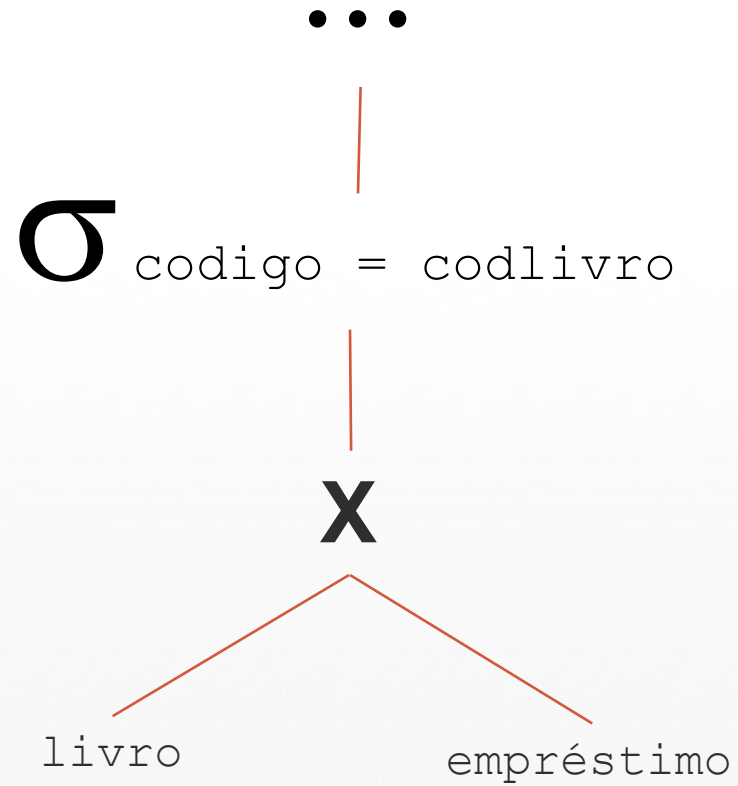
Q9: Recuperar dados dos livros emprestados

$\sigma_{\text{codigo} = \text{codLivro}} (\text{Livro} \mathbf{X} \text{Emprestimo})$

PK da relação livro

FK na relação emprestimo

Árvore algébrica



Renomeação - ρ

- Permite dar um **novo nome** à relação, ou a seus atributos – operação **unária**
- Sintaxe:

ρ NovoNomeRel (NovoAtrib1, ..., NovoAtribn) (<relação>)

- Onde:
 - <relação>: **nome da relação** que se deseja projetar os atributos
 - NovoNomeRel<atrib1, ..., atribn>: **novo nome** da relação e seus atributos (se for o caso deles serem renomeados)
-

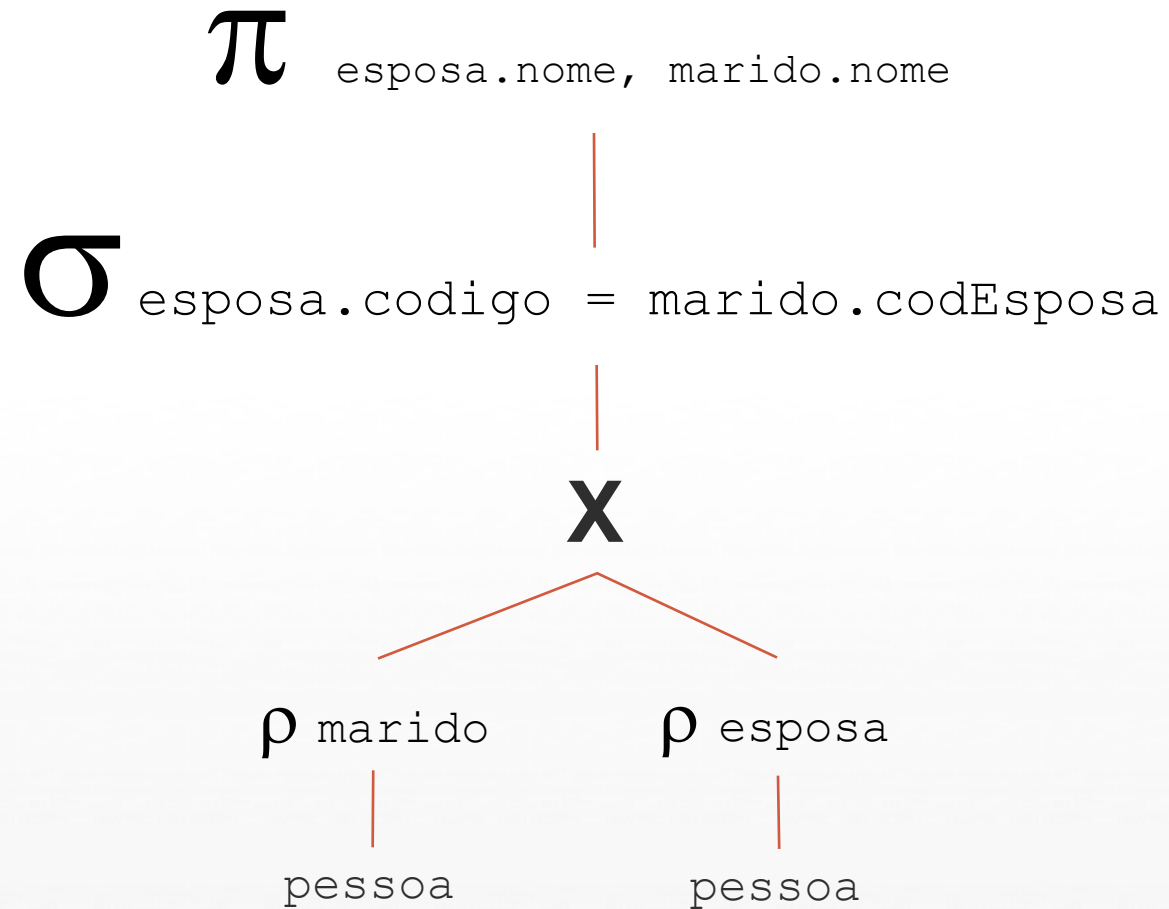
Quando usar a renomeação

- Quando a mesma tabela aparecer duas vezes na mesma consulta, mas com papéis diferentes
- Q10: Recuperar o nome da esposa e o nome do marido

π `esposa.nome, marido.nome`

σ `esposa.codigo = marido.codesp`
`(p esposa (pessoa)) X (p marido (pessoa))`

Árvore algébrica



Explicitando a tabela

- Quando duas tabelas possuírem **atributos de mesmo nome**, que **precisam ser projetados**

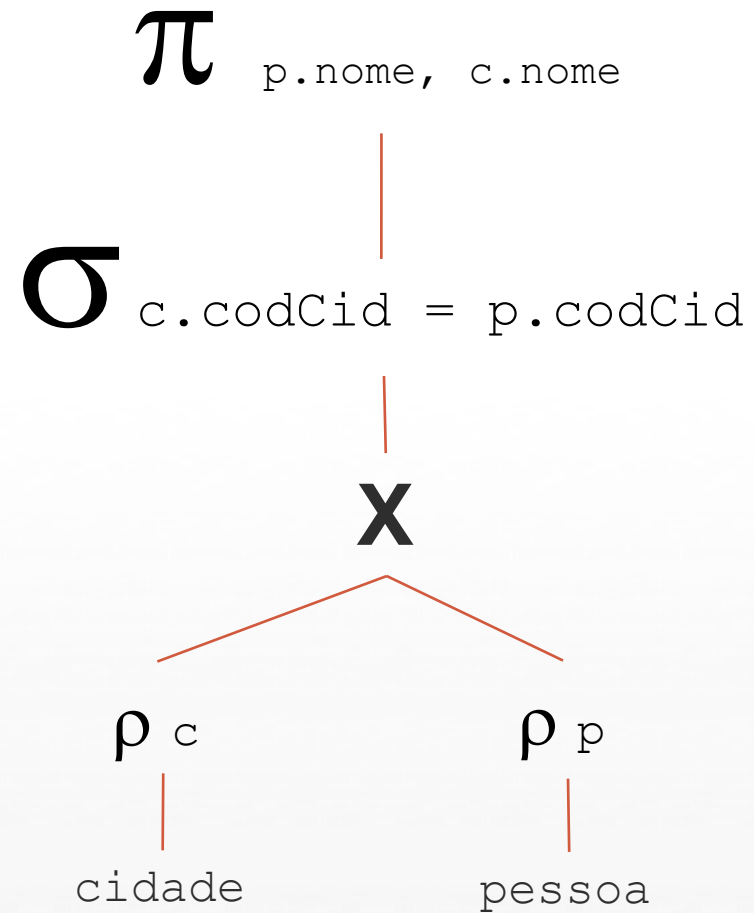
Cidade (codcid, nome, UF)

Pessoa (codpes, nome, codcid)

- Q11: Recuperar o nome das pessoas e o nome das cidades onde elas moram

$$\pi_{p.nome, c.nome}$$
$$(\sigma_{c.codcid = p.codcid} ((\rho_c(cidade)) \times (\rho_p(pessoa))))$$

Árvore algébrica



Atribuição

- Usado para **facilitar a escrita** de uma expressão algébrica
- **Atribui-se**, com o sinal `<-`, parte do resultado para uma variável
- Exemplo, a expressão:

$\pi_{\text{titulo, data}} (\sigma_{\text{codigo} = \text{codLivro}} (\text{Livro} \bowtie \text{Emprestimo}))$

- Pode ser escrita:

```
R1 <- Livro x Emprestimo
```

```
R2 <- ? codigo = codLivro (R1)
```

```
R3 <-  titulo, data (R2)
```
