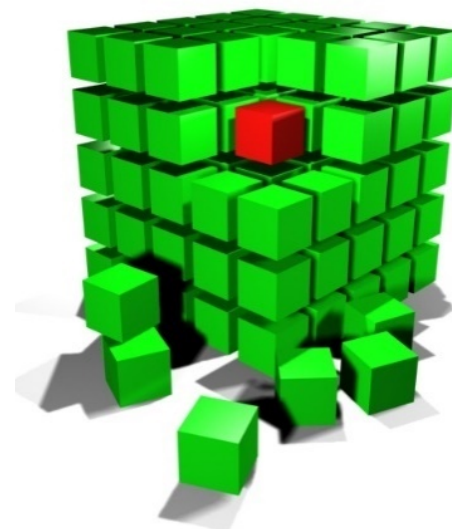




- **Aula 05**
  - Álgebra relacional
  - Exemplos de consultas





## ■ Introdução

- Os dados das tabelas relacionais possuem um valor limitado, a menos que possam ser manipulados para gerar informações úteis.
- Como o modelo relacional se baseia na teoria dos conjuntos, nada mais natural do que ele possuir uma linguagem formal que permita trabalhar com as relações e seus elementos.
- A **definição teórica** para a manipulação do conteúdo de tabelas relacionais é dado pela **álgebra relacional**.





## ■ Álgebra Relacional

- A álgebra relacional é uma linguagem formal, composta por um conjunto de operadores classificados de acordo com sua funcionalidade, que permite expressar qualquer operação de manipulação de dados em um banco de dados relacional.
- Esses operadores permitem construir expressões capazes de retornar uma relação contendo tuplas, que satisfazem determinadas condições.





## ■ Álgebra Relacional

- Inicialmente, Ted Codd definiu oito tipos de operações que podem ser realizadas para manipular os dados das tabelas relacionais: **Seleção**, **Projeção**, **Produto Cartesiano**, **União**, **Diferença de Conjuntos**, **Junção**, **Intersecção** e **Divisão**.
- Essas operações podem ser classificadas em:
  - **Unárias**: Seleção e Projeção;
  - **Binárias**: Produto Cartesiano, União, Diferença de Conjuntos, Junção, Intersecção e Divisão.





## ■ Álgebra Relacional

- As operações binárias de união e diferença exigem o que é denominado de **tabelas compatíveis para a união**.
- Isso significa que, nessas operações, somente podem ser utilizadas duas tabelas que possuam a mesma estrutura de dados, ou seja, o mesmo conjunto de colunas, dispostas na mesma ordem.





## ■ Exemplo de união (UNION)

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

**UNION**

P_CODE	P_DESCRIPT	PRICE
345678	Microwave	160.00
345679	Dishwasher	500.00

**Resulta em**



P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99
345678	Microwave	160
345679	Dishwasher	500





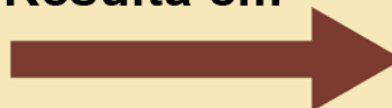
- Exemplo de intersecção (INTERSECT)

F_NAME
George
Jane
Elaine
Wilfred
Jorge

**INTERSECT**

F_NAME
Jane
William
Jorge
Dennis

**Resulta em**



F_NAME
Jane
Jorge





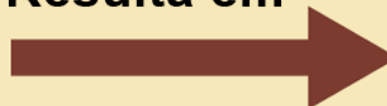
- Exemplo de diferença de conjuntos (DIFFERENCE)

F_NAME
George
Jane
Elaine
Wilfred
Jorge

**DIFFERENCE**

F_NAME
Jane
William
Jorge
Dennis

**Resulta em**



F_NAME
George
Elaine
Wilfred







## ■ Operadores da Álgebra Relacional

- **Projeção**, ou **PROJECT**: retorna uma nova relação, envolvendo todas as linhas de uma tabela, exibindo os dados somente dos atributos que forem especificados.
- **Seleção**, ou **SELECT**: retorna uma nova relação, envolvendo todos os atributos de uma tabela, filtrando as linhas de acordo com uma determinada condição.
- **Produto Cartesiano**, ou **PRODUCT**: retorna uma nova relação, envolvendo as informações de duas tabelas, juntando todos os seus atributos e multiplicando a quantidade de suas linhas.
- **Junção**, ou **JOIN**: retorna uma nova relação, envolvendo informações de duas tabelas, filtrando as linhas de acordo com os valores de suas chaves primária e estrangeira.





## ■ Exemplo de projeção (PROJECT)

**Tabela Original**

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

**PROJECT PRICE resulta em**

**Nova Tabela ou Lista**

PRICE
5.26
25.15
10.99
1.92
1.47
34.99

**PROJECT P\_DESCRIPT e PRICE resulta em**

P_DESCRIPT	PRICE
Flashlight	5.26
Lamp	25.15
Box Fan	10.99
9v battery	1.92
100W bulb	1.47
Powerdrill	34.99

**PROJECT P\_CODE e PRICE resulta em**

P_CODE	PRICE
123456	5.26
123457	25.15
123458	10.99
213345	1.92
254467	1.47
311452	34.99





## ■ Exemplo de seleção (SELECT)

**Tabela Original**

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

**SELECT ALL** resulta em

**Nova Tabela ou Lista**

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

**SELECT** apenas atributo PRICE inferior a US\$ 2.00 resulta em

P_CODE	P_DESCRIPT	PRICE
213345	9v battery	1.92
254467	100W bulb	1.47

**SELECT** apenas atributo P\_CODE = 311452 resulta em

P_CODE	P_DESCRIPT	PRICE
311452	Powerdrill	34.99





## ■ Exemplo de produto cartesiano (PRODUCT)

P_CODE	P_DESCRIPT	PRICE
123456	Flashlight	5.26
123457	Lamp	25.15
123458	Box Fan	10.99
213345	9v battery	1.92
254467	100W bulb	1.47
311452	Powerdrill	34.99

**PRODUCT**

STORE	aisle	shelf
23	vV	5
24	K	9
25	Z	6

Resulta em



P_CODE	P_DESCRIPT	PRICE	STORE	aisle	shelf
123456	Flashlight	5.26	23	vV	5
123456	Flashlight	5.26	24	K	9
123456	Flashlight	5.26	25	Z	6
123457	Lamp	25.15	23	vV	5
123457	Lamp	25.15	24	K	9
123457	Lamp	25.15	25	Z	6
123458	Box Fan	10.99	23	vV	5
123458	Box Fan	10.99	24	K	9
123458	Box Fan	10.99	25	Z	6
213345	9v battery	1.92	23	vV	5
213345	9v battery	1.92	24	K	9
213345	9v battery	1.92	25	Z	6
311452	Powerdrill	34.99	23	vV	5
311452	Powerdrill	34.99	24	K	9
311452	Powerdrill	34.99	25	Z	6
254467	100W bulb	1.47	23	vV	5
254467	100W bulb	1.47	24	K	9
254467	100W bulb	1.47	25	Z	6





## ■ Operadores de junção (JOINS)

- **Junção natural**, ou **NATURAL JOIN**: liga duas tabelas, selecionando automaticamente apenas as linhas com valores comuns em seu atributo comum (geralmente, chave primária e chave estrangeira).
- **Junção interna**, ou **INNER JOIN**: liga duas tabelas, selecionando as linhas de acordo com uma condição de igualdade, utilizada para comparar as colunas especificadas de cada tabela (geralmente, chave primária e chave estrangeira).
- **Junção teta**, ou **THETA JOIN**: liga duas tabelas, selecionando as linhas de acordo com uma condição de comparação, com exceção do operador de igualdade, utilizada para comparar as colunas especificadas de cada tabela (geralmente, chave primária e chave estrangeira).
- **Junção externa**, ou **OUTER JOIN**: liga duas tabelas, selecionando as linhas de acordo com uma condição de igualdade, utilizada para comparar as colunas especificadas de cada tabela (geralmente, chave primária e chave estrangeira). Retorna apenas os valores dos pares com correspondência. Valores sem correspondência na outra tabela são deixados nulos.





## ■ Operadores de junção (JOINS)

- A nova relação, obtida por meio de uma operação de junção, representa o resultado de um processo que envolve três estágios.
  1. Aplicamos o produto cartesiano (**PRODUCT**), para as duas tabelas envolvidas na operação de junção;
  2. Sobre o resultado obtido, executamos uma operação de seleção (**SELECT**), retornando apenas as linhas onde o valor do campo comum entre as duas tabelas seja idêntico;
  3. Para finalizar, executamos uma operação de projeção (**PROJECT**), eliminando os atributos duplicados.





## ■ Tabelas utilizadas para demonstrar os JOINS

Nome da Tabela: CUSTOMER

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE
1132445	Walker	32145	231
1217782	Adares	32145	125
1312243	Rakowski	34129	167
1321242	Rodriguez	37134	125
1542311	Smithson	37134	421
1657399	Vanloo	32145	231



Chave Estrangeira

Nome da Tabela: AGENT

AGENT_CODE	AGENT_PHONE
125	6152439887
167	6153426778
231	6152431124
333	9041234445



Chave Primária





## ■ Etapa 1: produto cartesiano, ou PRODUCT

CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1132445	Walker	32145	231	125	6152439887
1132445	Walker	32145	231	167	6153426778
1132445	Walker	32145	231	231	6152431124
1132445	Walker	32145	231	333	9041234445
1217782	Adares	32145	125	125	6152439887
1217782	Adares	32145	125	167	6153426778
1217782	Adares	32145	125	231	6152431124
1217782	Adares	32145	125	333	9041234445
1312243	Rakowski	34129	167	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1312243	Rakowski	34129	167	231	6152431124
1312243	Rakowski	34129	167	333	9041234445
1321242	Rodriguez	37134	125	125	6152439887
1321242	Rodriguez	37134	125	167	6153426778
1321242	Rodriguez	37134	125	231	6152431124
1321242	Rodriguez	37134	125	333	9041234445
1542311	Smithson	37134	421	125	6152439887
1542311	Smithson	37134	421	167	6153426778
1542311	Smithson	37134	421	231	6152431124
1542311	Smithson	37134	421	333	9041234445
1657399	Vanloo	32145	231	125	6152439887
1657399	Vanloo	32145	231	167	6153426778
1657399	Vanloo	32145	231	231	6152431124
1657399	Vanloo	32145	231	333	9041234445







- Etapa 1: produto cartesiano, ou PRODUCT

CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1132445	Walker	32145	231	125	6152439887
1132445	Walker	32145	231	167	6153426778
1132445	Walker	32145	231	231	6152431124
1132445	Walker	32145	231	333	9041234445
1217782	Adares	32145	125	125	6152439887
1217782	Adares	32145	125	167	6153426778
1217782	Adares	32145	125	231	6152431124
1217782	Adares	32145	125	333	9041234445
1312243	Rakowski	34129	167	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1312243	Rakowski	34129	167	231	6152431124
1312243	Rakowski	34129	167	333	9041234445
1321242	Rodriguez	37134	125	125	6152439887
1321242	Rodriguez	37134	125	167	6153426778
1321242	Rodriguez	37134	125	231	6152431124
1321242	Rodriguez	37134	125	333	9041234445
1542311	Smithson	37134	421	125	6152439887
1542311	Smithson	37134	421	167	6153426778
1542311	Smithson	37134	421	231	6152431124
1542311	Smithson	37134	421	333	9041234445
1657399	Vanloo	32145	231	125	6152439887
1657399	Vanloo	32145	231	167	6153426778
1657399	Vanloo	32145	231	231	6152431124
1657399	Vanloo	32145	231	333	9041234445





- Etapa 1: produto cartesiano, ou PRODUCT

CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1132445	Walker	32145	231	125	6152439887
1132445	Walker	32145	231	167	6153426778
1132445	Walker	32145	231	231	6152431124
1132445	Walker	32145	231	333	9041234445
1217782	Adares	32145	125	125	6152439887
1217782	Adares	32145	125	167	6153426778
1217782	Adares	32145	125	231	6152431124
1217782	Adares	32145	125	333	9041234445
1312243	Rakowski	34129	167	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1312243	Rakowski	34129	167	231	6152431124
1312243	Rakowski	34129	167	333	9041234445
1321242	Rodriguez	37134	125	125	6152439887
1321242	Rodriguez	37134	125	167	6153426778
1321242	Rodriguez	37134	125	231	6152431124
1321242	Rodriguez	37134	125	333	9041234445
1542311	Smithson	37134	421	125	6152439887
1542311	Smithson	37134	421	167	6153426778
1542311	Smithson	37134	421	231	6152431124
1542311	Smithson	37134	421	333	9041234445
1657399	Vanloo	32145	231	125	6152439887
1657399	Vanloo	32145	231	167	6153426778
1657399	Vanloo	32145	231	231	6152431124
1657399	Vanloo	32145	231	333	9041234445





- Etapa 2: operação de seleção, ou SELECT

CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1132445	Walker	32145	231	125	6152439887
1132445	Walker	32145	231	167	6153426778
1132445	Walker	32145	231	231	6152431124
1132445	Walker	32145	231	333	9041234445
1217782	Adares	32145	125	125	6152439887
1217782	Adares	32145	125	167	6153426778
1217782	Adares	32145	125	231	6152431124
1217782	Adares	32145	125	333	9041234445
1312243	Rakowski	34129	167	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1312243	Rakowski	34129	167	231	6152431124
1312243	Rakowski	34129	167	333	9041234445
1321242	Rodriguez	37134	125	125	6152439887
1321242	Rodriguez	37134	125	167	6153426778
1321242	Rodriguez	37134	125	231	6152431124
1321242	Rodriguez	37134	125	333	9041234445
1542311	Smithson	37134	421	125	6152439887
1542311	Smithson	37134	421	167	6153426778
1542311	Smithson	37134	421	231	6152431124
1542311	Smithson	37134	421	333	9041234445
1657399	Vanloo	32145	231	125	6152439887
1657399	Vanloo	32145	231	167	6153426778
1657399	Vanloo	32145	231	231	6152431124
1657399	Vanloo	32145	231	333	9041234445





## ■ Etapa 2: operação de seleção, ou SELECT

CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1132445	Walker	32145	231	125	6152439887
1132445	Walker	32145	231	167	6153426778
1132445	Walker	32145	231	231	6152431124
1132445	Walker	32145	231	333	9041234445
1217782	Adares	32145	125	125	6152439887
1217782	Adares	32145	125	167	6153426778
1217782	Adares	32145	125	231	6152431124
1217782	Adares	32145	125	333	9041234445
1312243	Rakowski	34129	167	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1312243	Rakowski	34129	167	231	6152431124
1312243	Rakowski	34129	167	333	9041234445
1321242	Rodriguez	37134	125	125	6152439887
1321242	Rodriguez	37134	125	167	6153426778
1321242	Rodriguez	37134	125	231	6152431124
1321242	Rodriguez	37134	125	333	9041234445
1542311	Smithson	37134	421	125	6152439887
1542311	Smithson	37134	421	167	6153426778
1542311	Smithson	37134	421	231	6152431124
1542311	Smithson	37134	421	333	9041234445
1657399	Vanloo	32145	231	125	6152439887
1657399	Vanloo	32145	231	167	6153426778
1657399	Vanloo	32145	231	231	6152431124
1657399	Vanloo	32145	231	333	9041234445





- Etapa 2: operação de seleção, ou SELECT

CUS_CODE	CUS_LNAME	CUS_ZIP	CUSTOMER.AGENT_CODE	AGENT.AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	125	6152439887
1321242	Rodriguez	37134	125	125	6152439887
1312243	Rakowski	34129	167	167	6153426778
1132445	Walker	32145	231	231	6152431124
1657399	Vanloo	32145	231	231	6152431124



Os valores de AGENT\_CODE têm  
de ser o mesmo nas duas tabelas

AGENT\_CODE é o campo em comum entre as tabelas





- Etapa 3: operação de projeção, ou PROJECT

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	6152439887
1321242	Rodriguez	37134	125	6152439887
1312243	Rakowski	34129	167	6153426778
1132445	Walker	32145	231	6152431124
1657399	Vanloo	32145	231	6152431124



Eliminamos as colunas duplicadas, deixando somente uma cópia de cada atributo







## ■ Junção por igualdade, ou EQUIJOIN

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	6152439887
1321242	Rodriguez	37134	125	6152439887
1312243	Rakowski	34129	167	6153426778
1132445	Walker	32145	231	6152431124
1657399	Vanloo	32145	231	6152431124



Retorna somente as linhas onde o valor do campo em comum exista nas duas tabelas

As linhas onde AGENT\_CODE = 333 ou AGENT\_CODE = 421 não são exibidas





- **Junção por igualdade, ou EQUIJOIN**

**Nome da Tabela: CUSTOMER**

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE
1132445	Walker	32145	231
1217782	Adares	32145	125
1312243	Rakowski	34129	167
1321242	Rodriguez	37134	125
1542311	Smithson	37134	421
1657399	Vanloo	32145	231

**Nome da Tabela: AGENT**

AGENT_CODE	AGENT_PHONE
125	6152439887
167	6153426778
231	6152431124
333	9041234445

Essas linhas não são retornadas







- **Junção externa à esquerda, ou LEFT JOIN**

**Nome da Tabela: CUSTOMER**

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE
1132445	Walker	32145	231
1217782	Adares	32145	125
1312243	Rakowski	34129	167
1321242	Rodriguez	37134	125
1542311	Smithson	37134	421
1657399	Vanloo	32145	231

**Nome da Tabela: AGENT**

AGENT_CODE	AGENT_PHONE
125	6152439887
167	6153426778
231	6152431124
333	9041234445



Tabela do lado esquerdo



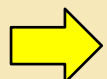


- **Junção externa à esquerda, ou LEFT JOIN**

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	6152439887
1321242	Rodriguez	37134	125	6152439887
1312243	Rakowski	34129	167	6153426778
1132445	Walker	32145	231	6152431124
1657399	Vanloo	32145	231	6152431124
1542311	Smithson	37134	421	



Retorna todas as linhas da tabela do lado esquerdo



**SELECT CUSTOMER LEFT OUTER JOIN AGENT**



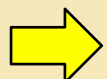


- Junção externa à esquerda, ou **LEFT JOIN**

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	6152439887
1321242	Rodriguez	37134	125	6152439887
1312243	Rakowski	34129	167	6153426778
1132445	Walker	32145	231	6152431124
1657399	Vanloo	32145	231	6152431124
1542311	Smithson	37134	421	



Valor Nulo



**SELECT CUSTOMER LEFT OUTER JOIN AGENT**





- **Junção externa à direita, ou RIGHT JOIN**

**Nome da Tabela: CUSTOMER**

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE
1132445	Walker	32145	231
1217782	Adares	32145	125
1312243	Rakowski	34129	167
1321242	Rodriguez	37134	125
1542311	Smithson	37134	421
1657399	Vanloo	32145	231

**Nome da Tabela: AGENT**

AGENT_CODE	AGENT_PHONE
125	6152439887
167	6153426778
231	6152431124
333	9041234445



**Tabela do lado direito**



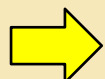


- **Junção externa à direita, ou RIGHT JOIN**

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	6152439887
1321242	Rodriguez	37134	125	6152439887
1312243	Rakowski	34129	167	6153426778
1132445	Walker	32145	231	6152431124
1657399	Vanloo	32145	231	6152431124
			333	9041234445



Retorna todas as linhas da tabela do lado direito



**SELECT CUSTOMER RIGHT OUTER JOIN AGENT**



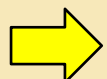


- **Junção externa à direita, ou RIGHT JOIN**

CUS_CODE	CUS_LNAME	CUS_ZIP	AGENT_CODE	AGENT_PHONE
1217782	Adares	32145	125	6152439887
1321242	Rodriguez	37134	125	6152439887
1312243	Rakowski	34129	167	6153426778
1132445	Walker	32145	231	6152431124
1657399	Vanloo	32145	231	6152431124
			333	9041234445



Valores Nulos



**SELECT CUSTOMER RIGHT OUTER JOIN AGENT**





## ■ Exemplo de divisão (DIVIDE)

Colunas Originais

Dividendo

CODE	LOC
A	5
A	9
A	4
B	5
B	3
C	6
D	7
D	8
E	8

**DIVIDE**

Divisor

CODE
A
B

Resulta em

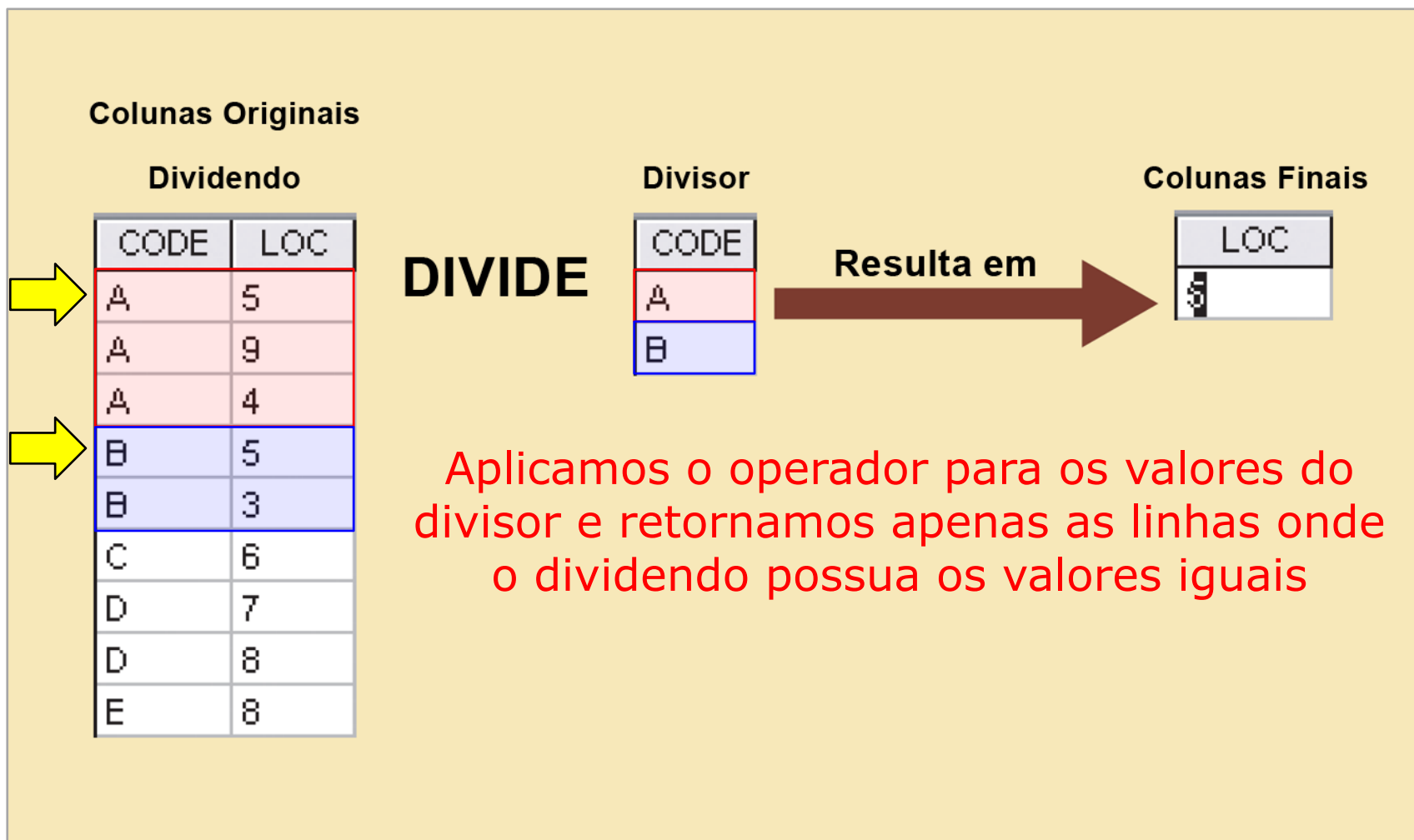
Colunas Finais

LOC
5





## ■ Exemplo de divisão (DIVIDE)







## ■ Consultas utilizando Álgebra Relacional

Álgebra Relacional	SQL
(ALUNOS)	SELECT * FROM ALUNOS;
$\sigma_{\text{Sexo} = 'M'}(\text{ALUNOS})$	SELECT * FROM ALUNOS WHERE SEXO = 'M';
$\pi_{\text{Nome}, \text{Sexo}}(\text{ALUNOS})$	SELECT Nome, Sexo FROM ALUNOS;
$\pi_{\text{Nome}, \text{Sexo}}(\sigma_{\text{Sexo} = 'M'}(\text{ALUNOS}))$	SELECT Nome, Sexo FROM ALUNOS WHERE SEXO = 'M';





## ■ Tabela ALUNOS

RA	Aluno	Sexo	Telefone
1	Ana	F	99660-9090
2	João	M	99661-9080
3	Pedro	M	99760-6070
4	Carlos	M	99761-7070
5	Maria	F	99797-8080
6	Renata	F	99776-9090





## ■ Tabela TURMAS

CodTurma	Turma
T1	Turma 1
T2	Turma 2
T3	Turma 3
T4	Turma 4





## ■ Tabela MATRICULAS

CodMatricula	DataMatricula	RA	CodTurma
1	02/02/2022	5	T1
2	02/02/2022	3	T1
3	03/02/2022	4	T3
4	05/02/2022	6	T3
5	10/02/2022	2	T1
6	15/02/2022	1	T2





## ■ Tabela DISCIPLINAS

CodDisciplina	Disciplina	RA
BDD	Banco de Dados	1
EDD	Estrutura de Dados	2
LOP	Lógica de Programação	3
RDC	Redes de Computadores	4
ANS	Análise de Sistemas	5
LIP	Linguagem de Programação	3





## ■ Tabela PROFESSORES

RA	Professor
1	Paulo Giovani
2	Marques Sousa
3	Augusto Manzano
4	João Evangelista
5	Elton Ferreira





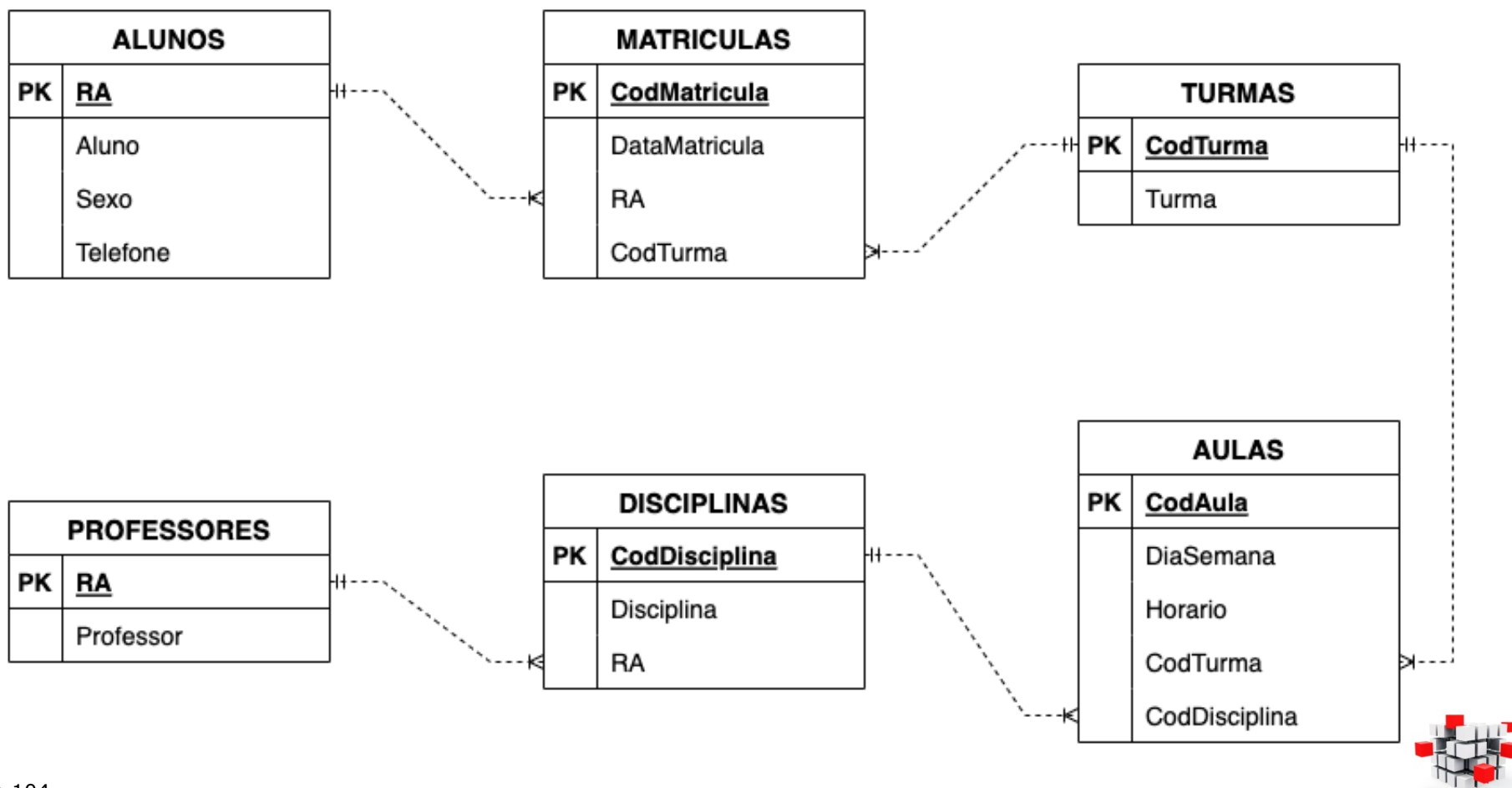
## ■ Tabela AULAS

CodAula	DiaSemana	Horario	CodTurma	CodDisciplina
1	Segunda-feira	19:00	T3	BDD
2	Sexta-feira	19:00	T3	BDD
3	Segunda-feira	21:00	T3	EDD
4	Terça-feira	19:00	T1	LOP
5	Quarta-feira	21:00	T2	ANS
6	Sexta-feira	19:00	T2	LIP
7	Quinta-feira	19:00	T4	RDC





## ■ DER (Pé-de-Galinha)







## ■ Operador de projeção (PROJECT)

- Indicada pela letra grega Pi,  $\pi$ , retorna uma nova relação contendo todas as linhas do conjunto de entrada. Os atributos dessa nova relação são especificados nos argumentos da operação.

$$\pi_{\text{nome coluna}_1, \text{nome coluna}_2, \dots, \text{nome coluna}_n} (\text{Nome da Tabela})$$

Diagram illustrating the syntax of the Project operator ( $\pi$ ):

- The operator  $\pi$  is followed by a list of column names (arguments) enclosed in a brace labeled "Argumentos da Operação".
- The list of column names is followed by the name of the input relation (set) enclosed in a brace labeled "Conjunto de Entrada".





## ■ Tabela ALUNOS

RA	Aluno	Sexo	Telefone
1	Ana	F	99660-9090
2	João	M	99661-9080
3	Pedro	M	99760-6070
4	Carlos	M	99761-7070
5	Maria	F	99797-8080
6	Renata	F	99776-9090





## ■ Operador de projeção (PROJECT)

$\pi$  *Aluno* (ALUNOS)

Observe que a operação de projeção é utilizada para retornar todas as linhas dentro do conjunto de entrada

Aluno
Ana
João
Pedro
Carlos
Maria
Renata





## ■ Tabela ALUNOS

RA	Aluno	Sexo	Telefone
1	Ana	F	99660-9090
2	João	M	99661-9080
3	Pedro	M	99760-6070
4	Carlos	M	99761-7070
5	Maria	F	99797-8080
6	Renata	F	99776-9090





- Operador de projeção (PROJECT)

$\pi$  RA, Aluno (ALUNOS)

RA	Aluno
1	Ana
2	João
3	Pedro
4	Carlos
5	Maria
6	Renata





## ■ Tabela AULAS

CodAula	DiaSemana	Horario	CodTurma	CodDisciplina
1	Segunda-feira	19:00	T3	BDD
2	Sexta-feira	19:00	T3	BDD
3	Segunda-feira	21:00	T3	EDD
4	Terça-feira	19:00	T1	LOP
5	Quarta-feira	21:00	T2	ANS
6	Sexta-feira	19:00	T2	LIP
7	Quinta-feira	19:00	T4	RDC





## ■ Operador de projeção (PROJECT)

$\pi$  CodAula,  
CodTurma,  
CodDisciplina  
(AULAS)

CodAula	CodTurma	CodDisciplina
1	T3	BDD
2	T3	BDD
3	T3	EDD
4	T1	LOP
5	T2	ANS
6	T2	LIP
7	T4	RDC





## ■ Operador de seleção (SELECT)

- Indicada pela letra grega Sigma,  $\sigma$ , a operação de seleção permite que, dado um conjunto inicial de entrada, seja produzido uma nova relação contendo todos os atributos, porém, apenas com as linhas que atendam a uma determinada condição.

$\sigma$  Condição (Nome da Tabela)







## ■ Tabela ALUNOS

RA	Aluno	Sexo	Telefone
1	Ana	F	99660-9090
2	João	M	99661-9080
3	Pedro	M	99760-6070
4	Carlos	M	99761-7070
5	Maria	F	99797-8080
6	Renata	F	99776-9090





## ▪ Operador de seleção (SELECT)

$\sigma_{\text{Sexo} = 'M'} (\text{ALUNOS})$

RA	Aluno	Sexo	Telefone
2	João	M	99661-9080
3	Pedro	M	99760-6070
4	Carlos	M	99761-7070



Observe que a operação de seleção é utilizada para filtrar as linhas dentro do conjunto de entrada





## ■ Tabela MATRICULAS

CodMatricula	DataMatricula	RA	CodTurma
1	02/02/2022	5	T1
2	02/02/2022	3	T1
3	03/02/2022	4	T3
4	05/02/2022	6	T3
5	10/02/2022	2	T1
6	15/02/2022	1	T2





- Operador de seleção (SELECT)

$\sigma_{\text{CodMatricula} > 4} (\text{MATRICULAS})$

CodMatricula	DataMatricula	RA	CodTurma
5	10/02/2022	2	T1
6	15/02/2022	1	T2





## ■ Combinando as operações

- Para obter apenas alguns atributos do conjunto de entrada, selecionando as linhas de acordo com determinada condição, devemos combinar as operações de projeção e seleção em uma única expressão, considerando a ordem de precedência das operações.
- A **operação de projeção** exibe todas as linhas para as colunas que forem especificadas. A **operação de seleção** exibe todas as colunas, aplicando um filtro para poder selecionar as linhas de acordo com algum critério.
- Dessa forma, devemos primeiro **selecionar** aquilo que deverá ser **projetado**.





## ■ Tabela ALUNOS

RA	Aluno	Sexo	Telefone
1	Ana	F	99660-9090
2	João	M	99661-9080
3	Pedro	M	99760-6070
4	Carlos	M	99761-7070
5	Maria	F	99797-8080
6	Renata	F	99776-9090





## ■ Combinando as operações

- Imagine a seguinte consulta: *Selecionar o RA, nome e telefone dos alunos. Exibir o resultado somente para os alunos que forem do sexo feminino.*

$\pi_{RA, Aluno, Telefone} \left( \underbrace{\sigma_{Sexo = 'F'} (ALUNOS)} \right)$

Retorna um conjunto, filtrando as linhas de acordo com o sexo dos alunos





## ■ Combinando as operações

- Imagine a seguinte consulta: *Selecionar o RA, nome e telefone dos alunos. Exibir o resultado somente para os alunos que forem do sexo feminino.*

$$\underbrace{\pi_{RA, Aluno, Telefone}}_{\text{Realiza a projeção dos atributos desejados}} (\sigma_{Sexo = 'F'} (ALUNOS))$$

Realiza a projeção dos atributos desejados







## ■ Combinando as operações

- Imagine a seguinte consulta: *Selecionar o RA, nome e telefone dos alunos. Exibir o resultado somente para os alunos que forem do sexo feminino.*

RA	Aluno	Sexo	Telefone
1	Ana	F	99660-9090
5	Maria	F	99797-8080
6	Renata	F	99776-9090





## ■ Combinando as operações

- Imagine a seguinte consulta: *Selecionar o RA, nome e telefone dos alunos. Exibir o resultado somente para os alunos que forem do sexo feminino.*

RA	Aluno	Telefone
1	Ana	99660-9090
5	Maria	99797-8080
6	Renata	99776-9090



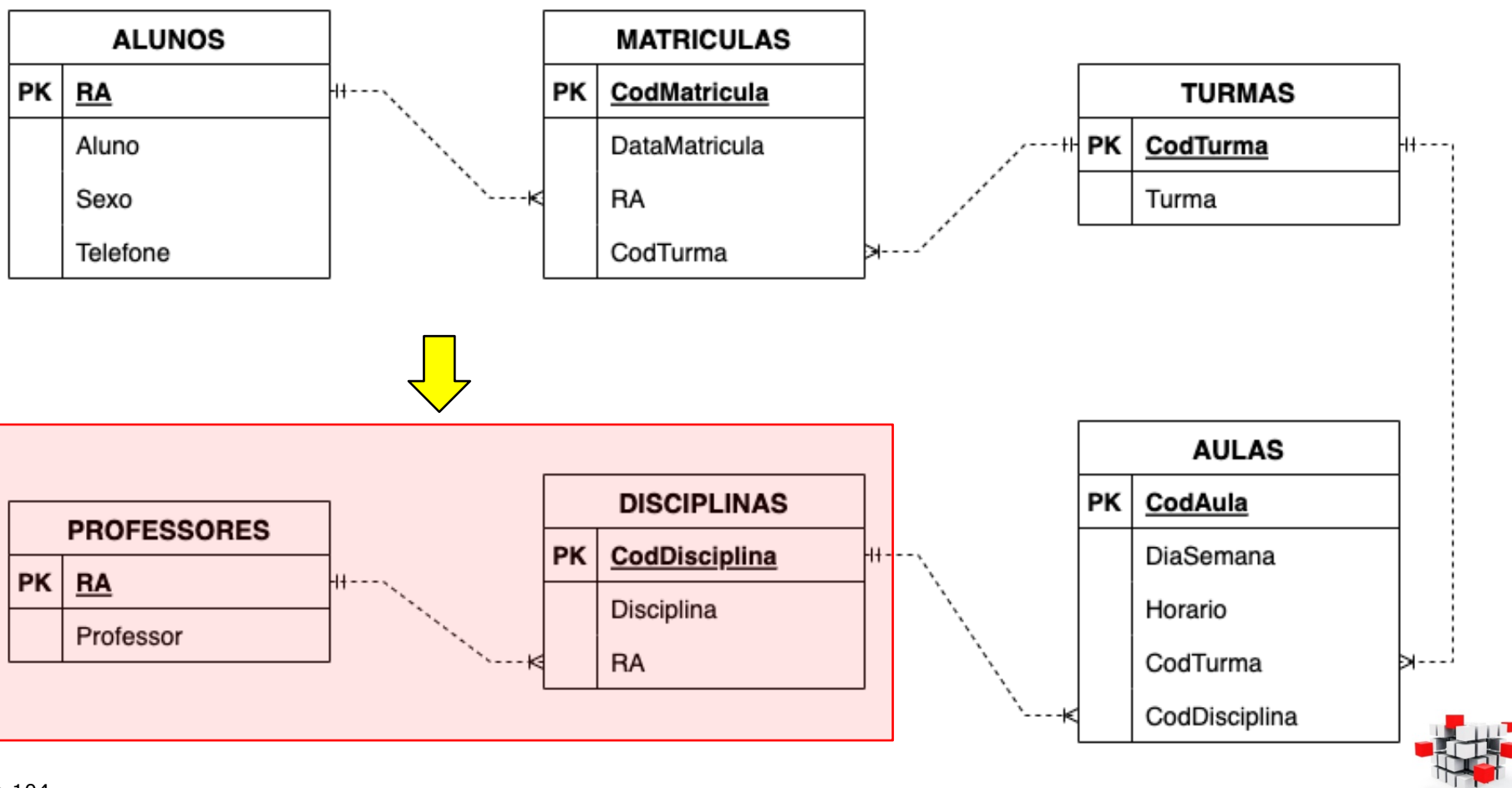


- **Consultas envolvendo mais tabelas**
    - Imagine a seguinte consulta: *Selecionar o RA, nome do professor e o nome da disciplina que ele ministra.*
- ➡ **Problema:** essa consulta envolve mais de uma tabela!





## ■ Consultas envolvendo mais tabelas





## ■ Produto cartesiano

- Para combinar os dados de mais de uma tabela, utilizamos uma operação chamada **produto cartesiano**, representada pela letra grega Khi,  $\times$ .
- O produto cartesiano representa uma operação binária, envolvendo dois conjuntos de entrada.
- A notação adotada na Matemática para representar o produto cartesiano entre dois conjuntos indica tipicamente a natureza dessa operação (**Conjunto 1**  $\times$  **Conjunto 2**).





## ■ Produto cartesiano

- O resultado do produto cartesiano de duas tabelas é uma **terceira tabela**, com as combinações possíveis entre os elementos das tabelas originais.
- A tabela resultante possui um **número de colunas** igual à soma do número de colunas das duas tabelas originais. O **número de linhas** da tabela resultante será igual ao produto do número de linhas das duas tabelas originais.





## ■ Produto cartesiano

- Por exemplo:
  - 4 colunas + 3 colunas = 7 colunas
  - 10 linhas \* 7 linhas = 70 linhas
- O produto cartesiano não é muito utilizado para se descobrir todas as combinações possíveis. Entretanto, esta é a forma primitiva que existe para unir informações de duas tabelas diferentes.





## ■ Produto cartesiano

- Para obtermos apenas algumas das combinações resultantes de um produto cartesiano, será necessário executar uma **operação de seleção** do seu resultado, de forma a descartar as combinações que não são de interesse entre as linhas das tabelas originais.







- **Consultas envolvendo mais tabelas**
  - Imagine a seguinte consulta: *Selecionar o RA, nome do professor e o nome da disciplina que ele ministra.*
- ➔ **Solução:** realizar uma operação de produto cartesiano envolvendo as tabelas DISCIPLINAS e PROFESSORES!





## ■ Tabela DISCIPLINAS

CodDisciplina	Disciplina	RA
BDD	Banco de Dados	1
EDD	Estrutura de Dados	2
LOP	Lógica de Programação	3
RDC	Redes de Computadores	4
ANS	Análise de Sistemas	5
LIP	Linguagem de Programação	3





## ■ Tabela PROFESSORES

RA	Professor
1	Paulo Giovani
2	Marques Sousa
3	Augusto Manzano
4	João Evangelista
5	Elton Ferreira





- Produto cartesiano: (DISCIPLINAS X PROFESSORES)**

CodDisciplina	Disciplina	RA	RA	Professor
BDD	Banco de Dados	1	1	Paulo Giovani
BDD	Banco de Dados	1	2	Marques Sousa
BDD	Banco de Dados	1	3	Augusto Manzano
BDD	Banco de Dados	1	4	João Evangelista
BDD	Banco de Dados	1	5	Elton Ferreira
EDD	Estrutura de Dados	2	1	Paulo Giovani
...	....	...	...	...
LIP	Linguagem de Programação	3	5	Elton Ferreira





- **Produto cartesiano: (DISCIPLINAS X PROFESSORES)**

CodDisciplina	Disciplina	→		Professor
BDD	Banco de Dados	RA	RA	Paulo Giovani
BDD	Banco de Dados	1	2	Marques Sousa
BDD	Banco de Dados	1	3	Augusto Manzano
BDD	Banco de Dados	1	4	João Evangelista
BDD	Banco de Dados	1	5	Elton Ferreira
EDD	Estrutura de Dados	2	1	Paulo Giovani
...	....	...	...	...
LIP	Linguagem de Programação	3	5	Elton Ferreira





## ■ Consultas envolvendo mais tabelas

➡ *Selecionar o RA, nome do professor e o nome da disciplina que ele ministra.*

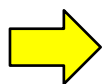
$\pi$  PROFESSORES.RA,  
Professor,  
Disciplina

$(\sigma \text{ DISCIPLINAS.RA} = \text{PROFESSORES.RA}$   
 $(\text{DISCIPLINAS X PROFESSORES}))$





## ■ Consultas envolvendo mais tabelas



*Selecionar o RA, nome do professor e o nome da disciplina que ele ministra.*

PROFESSORES.RA	Professor	Disciplina
1	Paulo Giovani	Banco de Dados
2	Marques Sousa	Estrutura de Dados
3	Augusto Manzano	Lógica de Programação
4	João Evangelista	Redes de Computadores
5	Elton Ferreira	Análise de Sistemas
3	Augusto Manzano	Linguagem de Programação





## ■ Ordenando o resultado

- Podemos ordenar o resultado obtido utilizando o operador de ordenação, representado pela letra grega Tau,  $\tau$ . A ordenação pode ser realizada de maneira crescente (ASC) ou decrescente (DESC).

$\tau$  Nome da Coluna 1 Tipo de Ordenação,  
Nome da Coluna 2 Tipo de Ordenação,  
...  
(Nome da Tabela)

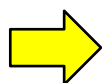
└──┘  
Cada coluna pode ser ASC ou DESC







## ■ Ordenando o resultado



*Selecionar o RA, nome do professor e o nome da disciplina que ele ministra.  
Ordenar o resultado pelo nome do professor em ordem alfabética, seguido  
pelo nome da disciplina em ordem decrescente.*

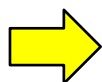
$\pi$  PROFESSORES.RA,  
Professor,  
Disciplina

( $\tau$  Professor, Disciplina DESC (  
( $\sigma$  DISCIPLINAS.RA = PROFESSORES.RA  
(DISCIPLINAS X PROFESSORES)))





## ■ Ordenando o resultado



*Selecionar o RA, nome do professor e o nome da disciplina que ele ministra. Ordenar o resultado pelo nome do professor em ordem alfabética, seguido pelo nome da disciplina em ordem decrescente.*

ASC

DESC

PROFESSORES.RA	Professor	Disciplina
3	Augusto Manzano	Lógica de Programação
3	Augusto Manzano	Linguagem de Programação
5	Elton Ferreira	Análise de Sistemas
4	João Evangelista	Redes de Computadores
2	Marques Sousa	Estrutura de Dados
1	Paulo Giovanni	Banco de Dados





## ■ Agregação e agrupamento

- Existem situações onde, além de recuperar as informações, também desejamos efetuar algum tipo de operação estatística básica, tal como a contagem de registros, a soma de valores, o cálculo da média, entre outros. Nesses casos, geralmente, precisamos agrupar os registros de acordo com os valores de determinada coluna.
- O agrupamento de registros é representado pela letra grega Gamma,  $\gamma$ .





## ▪ Agregação e agrupamento

- As funções de agregação disponíveis são: **COUNT**, **SUM**, **AVG**, **MIN** e **MAX**. No exemplo abaixo, obtemos os dados de uma tabela, agrupamos pelas colunas 1 e 2, e adicionamos uma terceira coluna, com o resultado da contagem das linhas para esse agrupamento.

$\pi$  Coluna 1,  
Coluna 2

Podemos ter mais colunas

→ Nova Coluna,  $\gamma$  Coluna 1, Coluna 2;

**COUNT** (Coluna Agregada) → Nova Coluna  
(Nome da Tabela)





## ▪ Agregação e agrupamento

➔ *Selecionar o RA, nome do professor e o total de disciplinas que ele ministra.*

```
 $\pi$  PROFESSORES.RA,  
Professor,  
Total_Disciplinas  $\gamma$  PROFESSORES.RA, Professor;  
COUNT (Disciplina)  $\rightarrow$  Total_Disciplinas  
( $\sigma$  DISCIPLINAS.RA = PROFESSORES.RA  
(DISCIPLINAS X PROFESSORES)  
)
```





## ■ Agregação e agrupamento

➡ *Selecionar o RA, nome do professor e o total de disciplinas que ele ministra.*

PROFESSORES.RA	Professor	Total_Disciplinas
1	Paulo Giovani	1
2	Marques Sousa	1
3	Augusto Manzano	2
4	João Evangelista	1
5	Elton Ferreira	1





## ■ Operador renomear

- Indicada pela letra grega Rho,  $\rho$ , o operador renomear é utilizado para alterar o nome de uma tabela, sempre que ela aparecer mais de uma vez em uma determinada consulta.

$\rho$  <Novo Nome> (Nome da Tabela)





## ▪ Operador renomear

- O operador  $\rho$  permite que se realize consultas como  $(\text{ALUNOS} \times \text{ALUNOS})$ .

$$\rho \underbrace{\text{ALUNOS2}}_{\text{ALUNOS}} (\text{ALUNOS})$$

Dessa forma, por exemplo, podemos realizar comparações entre ALUNOS.RA e ALUNOS2.RA







## ▪ Operador de atribuição

- O operador  $\leftarrow$  permite atribuir o resultado de uma consulta para uma determinada tabela.

```
RESULTADO  $\leftarrow$   $\pi$  PROFESSORES.RA,  
Professor,  
Disciplina
```

```
( $\sigma$  DISCIPLINAS.RA = PROFESSORES.RA  
(DISCIPLINAS X PROFESSORES)  
)
```





## ■ Operadores de conjunto

- Indicados pelos símbolos  $\cup$ ,  $\cap$  e  $-$ , esses operadores retornam uma nova relação, de acordo com as seguintes regras:
  - **UNIÃO**, ou  $\cup$ : retorna a união de todas as linhas dos dois conjuntos de entrada;
  - **INTERSECÇÃO**, ou  $\cap$ : retorna somente as linhas que são comuns dentro dos dois conjuntos de entrada;
  - **DIFERENÇA**, ou  $-$ : retorna as linhas que existem somente no primeiro conjunto de entrada.





## ■ Operador de união (UNION)

- Neste exemplo, separamos dois conjuntos de alunos, de acordo com seu gênero. Em seguida, realizamos uma operação de união.

MASCULINOS  $\leftarrow (\sigma_{\text{Sexo} = 'M'} (\text{ALUNOS}))$

FEMININOS  $\leftarrow (\sigma_{\text{Sexo} = 'F'} (\text{ALUNOS}))$

$(\text{MASCULINOS} \cup \text{FEMININOS})$

A estrutura dos dois conjuntos deve ser a mesma





## ▪ Operador de união (UNION)

(MASCULINOS  $\cup$  FEMININOS)

RA	Aluno	Sexo	Telefone
2	João	M	99661-9080
3	Pedro	M	99760-6070
4	Carlos	M	99761-7070
1	Ana	F	99660-9090
5	Maria	F	99797-8080
6	Renata	F	99776-9090





## ■ Consultas envolvendo mais tabelas

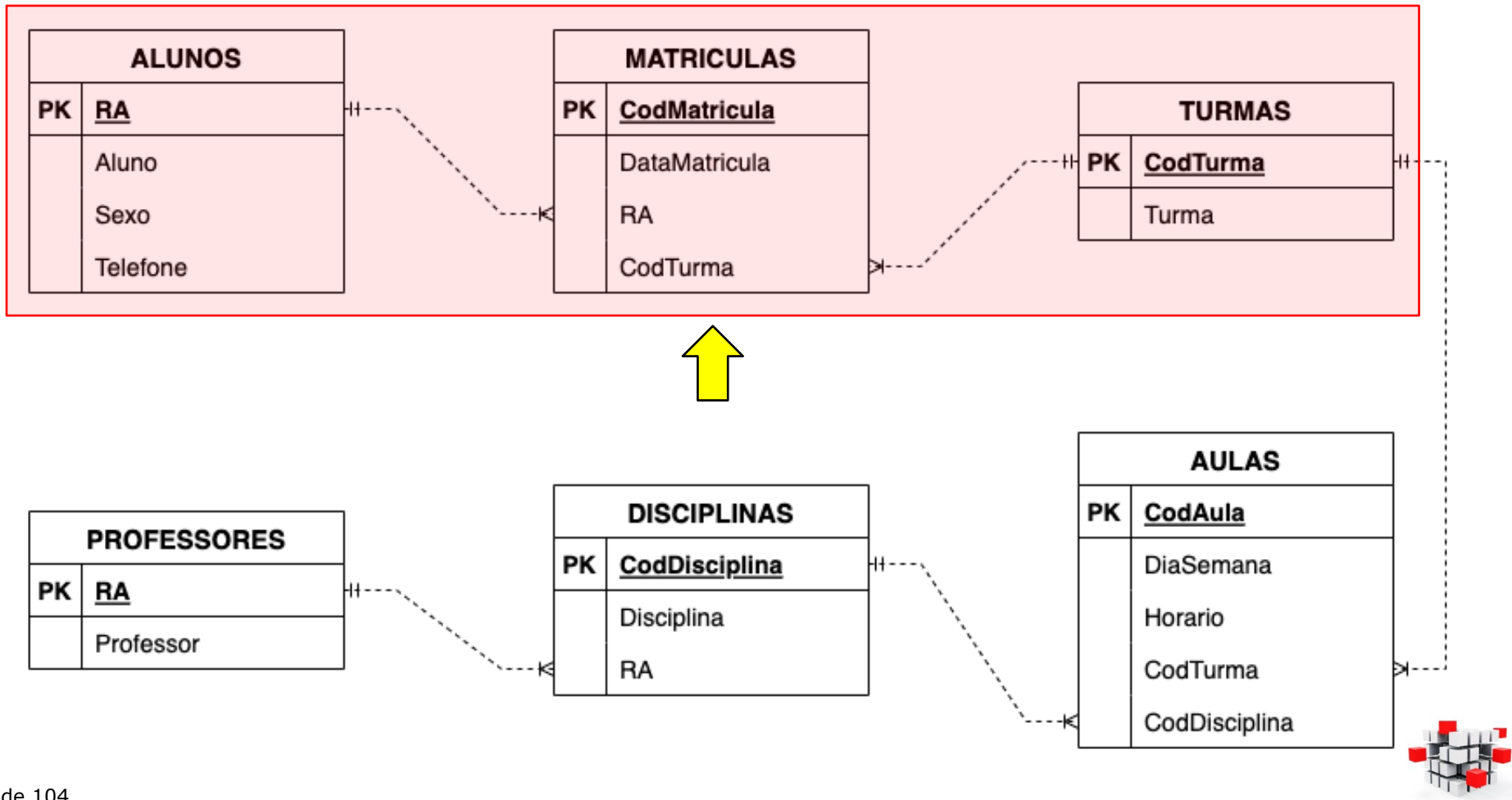
➡ Imagine a seguinte consulta: *Selecionar o código e data da matrícula, RA e nome do aluno, assim como o nome da sua turma. Exibir somente os dados de quem estiver na mesma turma da aluna Renata.*

- **Solução:** realizar uma operação de produto cartesiano envolvendo as tabelas ALUNOS, MATRICULAS e TURMAS!





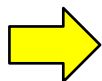
## ■ Consultas envolvendo mais tabelas





## ■ Tabela ALUNOS

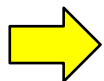
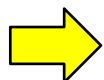
RA	Aluno	Sexo	Telefone
1	Ana	F	99660-9090
2	João	M	99661-9080
3	Pedro	M	99760-6070
4	Carlos	M	99761-7070
5	Maria	F	99797-8080
6	Renata	F	99776-9090





## ■ Tabela MATRICULAS

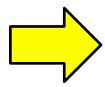
CodMatricula	DataMatricula	RA	CodTurma
1	02/02/2022	5	T1
2	02/02/2022	3	T1
3	03/02/2022	4	T3
4	05/02/2022	6	T3
5	10/02/2022	2	T1
6	15/02/2022	1	T2







## ■ Tabela TURMAS



CodTurma	Turma
T1	Turma 1
T2	Turma 2
T3	Turma 3
T4	Turma 4

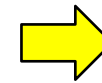




## ■ Consultas envolvendo mais tabelas

➡ *Recupera o código da turma da aluna Renata.*

TURMA\_RENATA ←



TURMAS.Codigo

$\rho$  Codigo ← TURMAS.CodTurma (

T3

$\pi$  TURMAS.CodTurma

( $\sigma$  MATRICULAS.RA = ALUNOS.RA  $\wedge$

ALUNOS.Aluno = 'Renata' (ALUNOS X (

$\sigma$  MATRICULAS.CodTurma = TURMAS.CodTurma

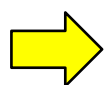
(TURMAS X MATRICULAS)

)))



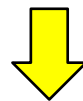


## ■ Consultas envolvendo mais tabelas



*Selecionar o código e data da matrícula, RA e nome do aluno, assim como o nome da sua turma. Exibir somente os dados de quem estiver na mesma turma da aluna Renata.*

$\pi$  MATRICULAS.CodMatricula,  
MATRICULAS.DataMatricula,  
ALUNOS.RA,  
ALUNOS.Aluno,  
TURMAS.Turma

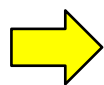


$(\sigma_{\text{TURMAS.CodTurma} = \text{TURMAS.Codigo}} (\text{TURMA\_RENATA} \times$   
 $(\sigma_{\text{MATRICULAS.RA} = \text{ALUNOS.RA}} (\text{ALUNOS} \times$   
 $(\sigma_{\text{MATRICULAS.CodTurma} = \text{TURMAS.CodTurma}}$   
 $(\text{TURMAS} \times \text{MATRICULAS))))))$





## ■ Consultas envolvendo mais tabelas



*Selecionar o código e data da matrícula, RA e nome do aluno, assim como o nome da sua turma. Exibir somente os dados de quem estiver na mesma turma da aluna Renata.*

MATRICULAS. CodMatricula	MATRICULAS. DataMatricula	ALUNOS.RA	ALUNOS.Aluno	TURMAS.Turma
3	03/02/2022	4	Carlos	Turma 3
4	05/02/2022	6	Renata	Turma 3





## ■ Consultas envolvendo mais tabelas

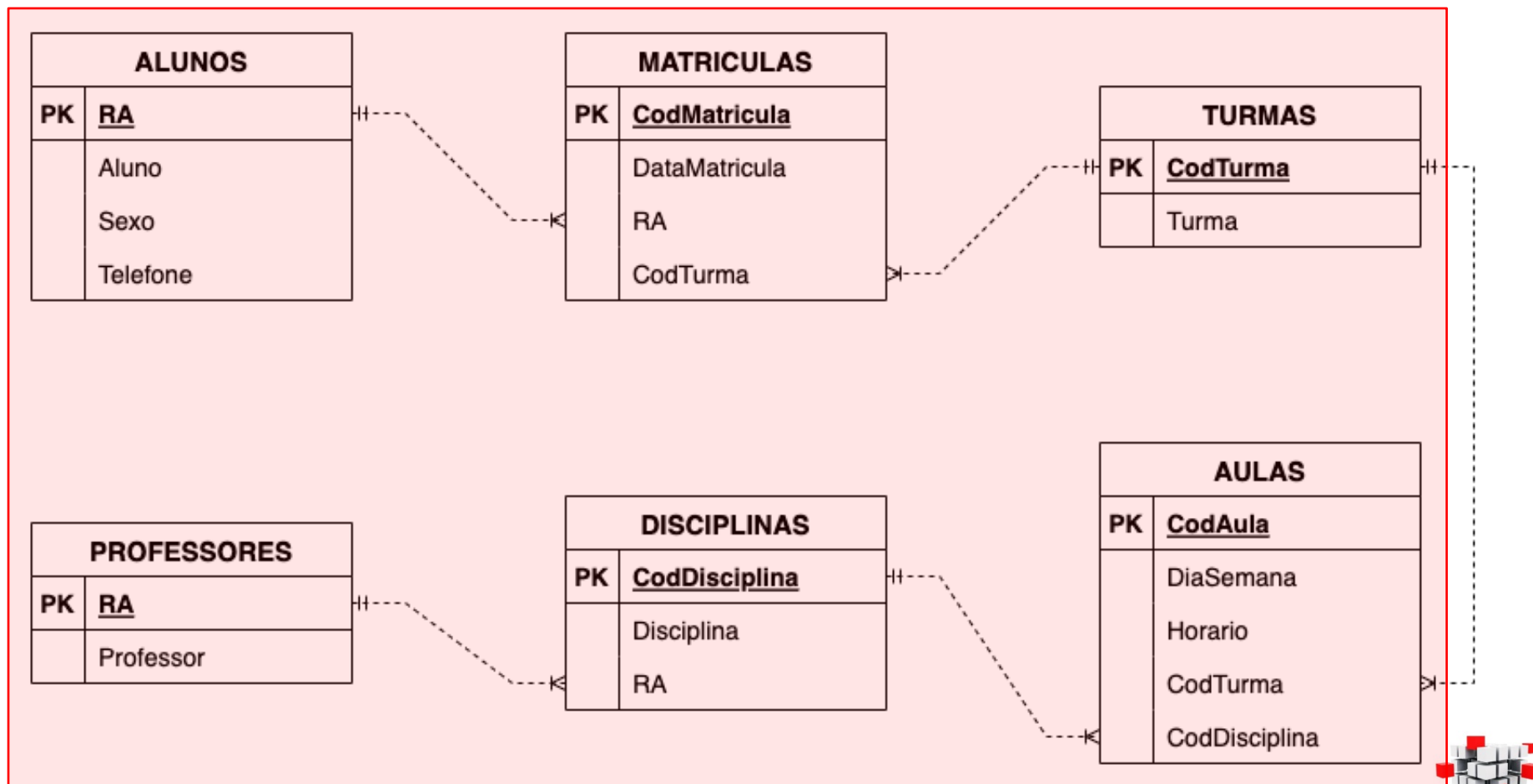
➡ Imagine a seguinte consulta: *Selecionar o RA e nome do aluno, o nome de sua turma, o nome das disciplinas que ele cursa, o nome do professor dessas disciplinas, o dia da semana e o horário em que essas disciplinas são ofertadas.*

- **Solução:** essa consulta envolve todas as tabelas!





## ■ Consultas envolvendo mais tabelas





## ■ Atributos que devem ser exibidos

$\pi$  ALUNOS.RA,  
ALUNOS.Aluno,  
TURMAS.Turma,  
DISCIPLINAS.Disciplina,  
PROFESSORES.Professor,  
AULAS.DiaSemana,  
AULAS.Horario

$(\sigma \dots)$  

Precisamos definir os critérios para seleção das linhas  
de cada relacionamento, entre todas as tabelas





## ■ Critérios para selecionar as linhas

$\pi$  ... 

```
( $\sigma$  ALUNOS.RA = MATRICULAS.RA (ALUNOS X  
  ( $\sigma$  MATRICULAS.CodTurma = TURMAS.CodTurma (MATRICULAS X  
    ( $\sigma$  TURMAS.CodTurma = AULAS.CodTurma (TURMAS X  
      ( $\sigma$  AULAS.CodDisciplina = DISCIPLINAS.CodDisciplina  
(AULAS X  
      ( $\sigma$  DISCIPLINAS.RA = PROFESSORES.RA (DISCIPLINAS X  
PROFESSORES))  
    ))  
  ))  
))  
))  
))
```







## ■ Resultado final

```
 $\pi$  ALUNOS.RA,  
ALUNOS.Aluno,  
TURMAS.Turma,  
DISCIPLINAS.Disciplina,  
PROFESSORES.Professor,  
AULAS.DiaSemana,  
AULAS.Horario  
( $\sigma$  ALUNOS.RA = MATRICULAS.RA (ALUNOS X  
  ( $\sigma$  MATRICULAS.CodTurma = TURMAS.CodTurma (MATRICULAS X  
    ( $\sigma$  TURMAS.CodTurma = AULAS.CodTurma (TURMAS X  
      ( $\sigma$  AULAS.CodDisciplina = DISCIPLINAS.CodDisciplina (AULAS X  
        ( $\sigma$  DISCIPLINAS.RA = PROFESSORES.RA (DISCIPLINAS X PROFESSORES))  
      ))  
    ))  
  ))  
))
```





## ■ Resultado final

ALUNOS.RA	ALUNO.Aluno	TURMA.Turma	DISCIPLINAS.Disciplina	PROFESSORES.Professor	AULAS.DiaSemana	AULAS.Horario
1	Ana	Turma 2	Análise de Sistemas	Elton Ferreira	Quarta-feira	21:00
1	Ana	Turma 2	Linguagem de Programação	Augusto Manzano	Sexta-feira	19:00
2	João	Turma 1	Lógica de Programação	Augusto Manzano	Terça-feira	19:00
3	Pedro	Turma 1	Lógica de Programação	Augusto Manzano	Terça-feira	19:00
4	Carlos	Turma 3	Banco de Dados	Paulo Giovani	Segunda-feira	19:00
4	Carlos	Turma 3	Banco de Dados	Paulo Giovani	Sexta-feira	19:00
4	Carlos	Turma 3	Estrutura de Dados	Marques Sousa	Segunda-feira	21:00
5	Maria	Turma 1	Lógica de Programação	Augusto Manzano	Terça-feira	19:00
6	Renata	Turma 3	Banco de Dados	Paulo Giovani	Segunda-feira	19:00
6	Renata	Turma 3	Banco de Dados	Paulo Giovani	Sexta-feira	19:00
6	Renata	Turma 3	Estrutura de Dados	Marques Sousa	Segunda-feira	21:00





## ■ Operadores de junção (JOINS)

- Existem situações onde as tabelas envolvidas em uma operação de produto cartesiano possuem o mesmo nome para seus atributos de chave primária e chave estrangeira. Nesses casos, podemos utilizar os operadores específicos de cada operação de junção:
  - **Junção cruzada**, ou **CROSS JOIN**:  $\times$
  - **Junção natural**, ou **NATURAL JOIN**:  $\bowtie$
  - **Junção interna**, ou **INNER JOIN**:  $\bowtie$
  - **Junção externa à esquerda**, ou **LEFT OUTER JOIN**:  $\bowtie\leftarrow$
  - **Junção externa à direita**, ou **RIGHT OUTER JOIN**:  $\rightarrow\bowtie$
  - **Junção externa completa**, ou **FULL OUTER JOIN**:  $\bowtie\rightleftarrows$





## ■ Exemplo de produto cartesiano (PRODUCT)

➔ *Selecionar o RA, nome do professor e o nome da disciplina que ele ministra.*

$\pi$  PROFESSORES.RA,  
Professor,  
Disciplina

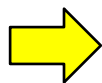
} Se o atributo for repetido, precisamos especificar o nome da tabela

$(\sigma$  DISCIPLINAS.RA = PROFESSORES.RA  
(DISCIPLINAS X PROFESSORES)  
)





## ■ Exemplo de produto cartesiano (PRODUCT)



*Selecionar o RA, nome do professor e o nome da disciplina que ele ministra.*

PROFESSORES.RA	Professor	Disciplina
1	Paulo Giovani	Banco de Dados
2	Marques Sousa	Estrutura de Dados
3	Augusto Manzano	Lógica de Programação
4	João Evangelista	Redes de Computadores
5	Elton Ferreira	Análise de Sistemas
3	Augusto Manzano	Linguagem de Programação





## ▪ Exemplo de junção interna (INNER JOIN)

➡ *Selecionar o RA, nome do professor e o nome da disciplina que ele ministra.*

$\pi$  RA,  
Professor,  
Disciplina

} Utilizando um operador de junção, não precisamos especificar o nome da tabela caso um atributo seja repetido

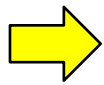
(DISCIPLINAS  $\bowtie$  PROFESSORES)

A ordem das tabelas pode influenciar a ordem do resultado





## ▪ Exemplo de junção interna (INNER JOIN)



*Selecionar o RA, nome do professor e o nome da disciplina que ele ministra.*

DISCIPLINAS.RA	Professor	Disciplina
1	Paulo Giovani	Banco de Dados
2	Marques Sousa	Estrutura de Dados
3	Augusto Manzano	Lógica de Programação
4	João Evangelista	Redes de Computadores
5	Elton Ferreira	Análise de Sistemas
3	Augusto Manzano	Linguagem de Programação



- **Na próxima aula veremos...**
  - Análise de Requisitos e Regras de Negócios.

