



WWW.UNICARIOCA.EDU.BR

Banco de Dados II

Tema 8

Profª Giselle Batalha



MELHOR CENTRO UNIVERSITÁRIO DO RIO DE JANEIRO!

União

Produz como resultado **uma Relação que contém todas as linhas da primeira Relação seguidas de todas as linhas da segunda tabela**. A Relação resultante possui a mesma quantidade de colunas que as relações originais, e tem um número de linhas que é no máximo igual à soma das linhas das relações fornecidas como operandos, já que as linhas que são comuns a ambas as relações aparecem uma única vez no resultado.

Retorna a união das tuplas de duas relações R1 e R2 com eliminação automática de duplicatas

União

Notação: Relação1 \cup Relação2 ($R1 \cup R2$)

Obs: As relações devem possuir o mesmo número de atributos.

Alunos

id	nome	idade	curso
10	Macoratti	45	Quimica
20	Miriam	43	Artes
30	Bianca	21	Fisica

Professores

id	nome	idade	setor
100	Pedro	50	Quimica
200	Maria	45	Fisica
300	Bianca	21	Artes

Funcionarios

id	nome	setor	idade
10	Margarida	Quimica	46
20	Jamil	Fisica	32

Domínio:

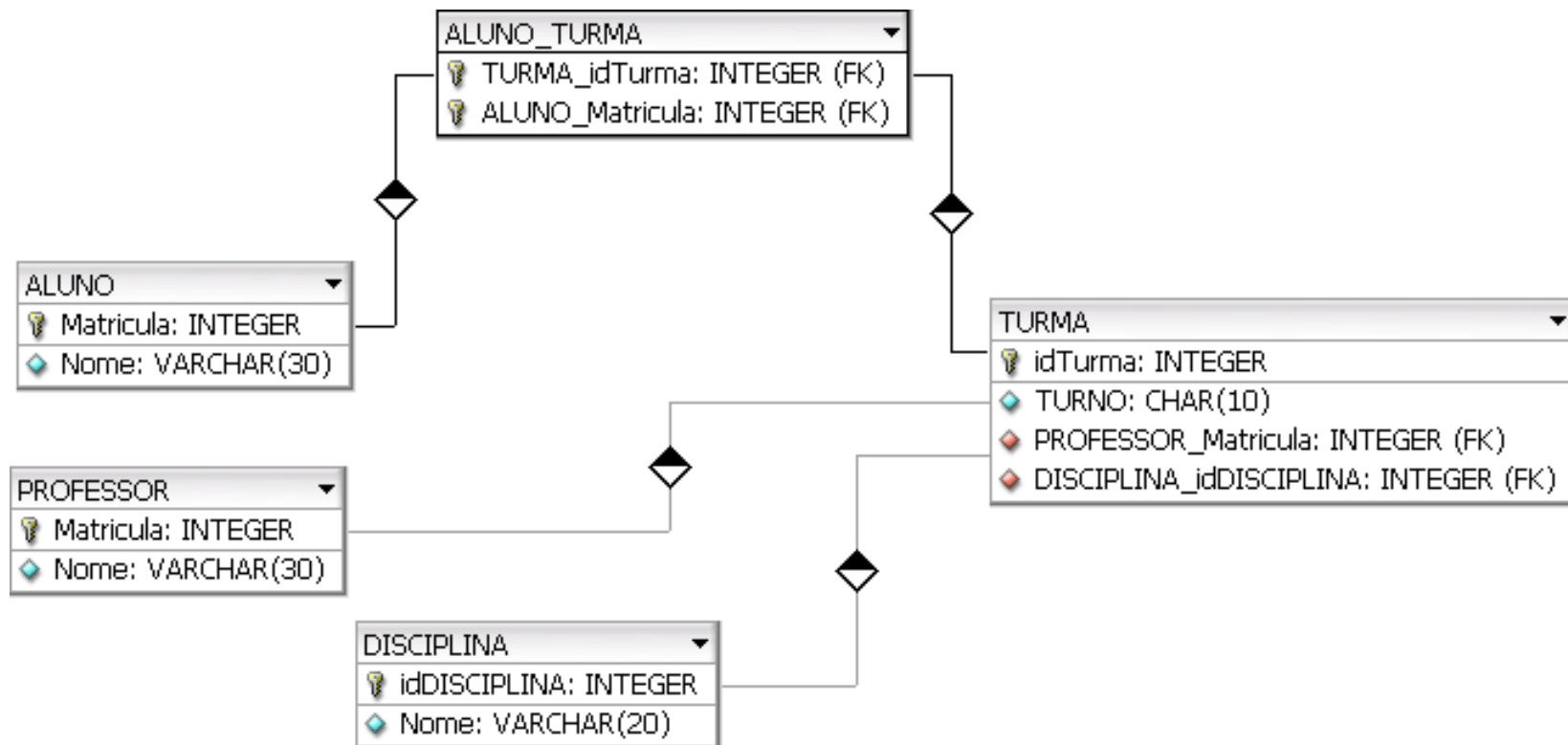
id = int
nome = varchar(30)
idade = int
curso = varchar(30)
setor = varchar(30)

A relação **Alunos** é compatível com **Professores** mas não é compatível com **Funcionarios**.

Ex1: Encontre uma relação com todos os alunos e com todos os professores:

resultado: **Alunos \cup Professores**

id	nome	idade	curso
10	Macoratti	45	Quimica
20	Miriam	43	Artes
30	Bianca	21	Fisica
100	Pedro	50	Quimica
200	Maria	45	Fisica
300	Bianca	21	Artes



BANCO DE DADOS BASE PARA ESSA AULA

```
CREATE DATABASE AULA10;  
USE AULA10;
```

```
CREATE TABLE ALUNO (  
  Matricula INTEGER PRIMARY KEY,  
  Nome VARCHAR(30) NOT NULL);
```

```
CREATE TABLE PROFESSOR (  
  Matricula INTEGER PRIMARY KEY,  
  Nome VARCHAR(30) NOT NULL);
```

```
CREATE TABLE DISCIPLINA (  
  idDISCIPLINA INTEGER PRIMARY KEY,  
  Nome VARCHAR(20) NULL);
```

```
CREATE TABLE TURMA (  
  idTurma INTEGER PRIMARY KEY,  
  turno CHAR(10) NOT NULL,  
  Matricula INTEGER NOT NULL,  
  idDISCIPLINA INTEGER NOT NULL,  
  FOREIGN KEY (Matricula) REFERENCES PROFESSOR(Matricula),  
  FOREIGN KEY (idDISCIPLINA) REFERENCES DISCIPLINA(idDISCIPLINA));
```

```
CREATE TABLE ALUNO_TURMA (  
  idTurma INTEGER NOT NULL,  
  Matricula INTEGER NOT NULL,  
  PRIMARY KEY(idTurma, Matricula),  
  FOREIGN KEY (idTurma) REFERENCES TURMA(idTurma),  
  FOREIGN KEY (Matricula) REFERENCES ALUNO(Matricula));
```

Listar as pessoas (professor e aluno) da base de dados.

```
SELECT * from PROFESSOR
```

UNION

```
SELECT * from ALUNO;
```

Result Grid			Filter Rows:
	Matricula	Nome	
▶	1001	MONTEIRO	
	1002	BATALHA	
	1000	MARIA	
	2000	MARIANA	
	3000	MARIO	
	4000	MARIANGELA	
	5000	MARIAJUPIRA	