

**INSTITUTO FEDERAL**

Goiano

Campus Morrinhos

# Banco de Dados

## O modelo

## Relacional

Professor Marcel Melo

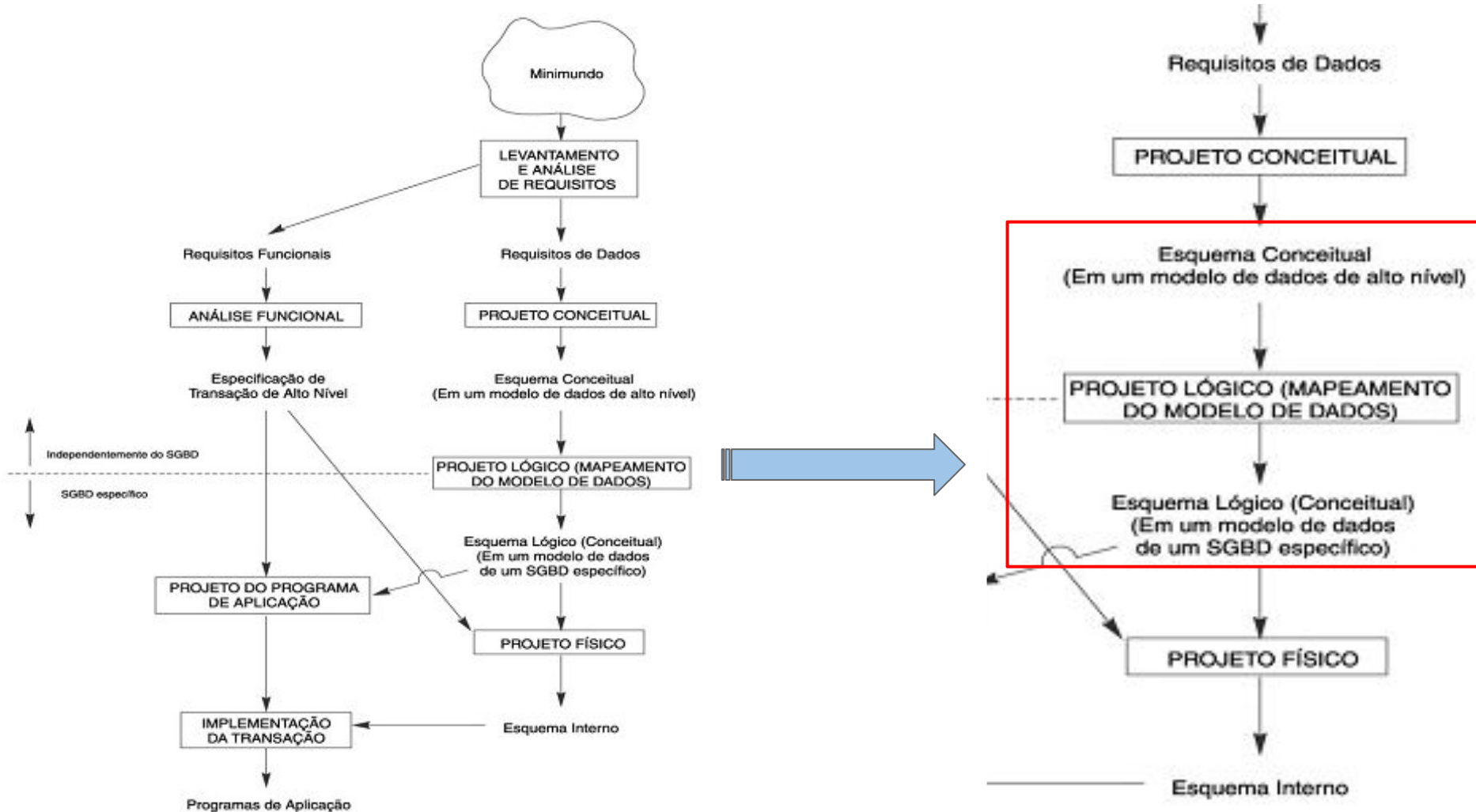
[marcel.melo@ifgoiano.edu.br](mailto:marcel.melo@ifgoiano.edu.br)

## **O modelo conceitual é dado por modelos de alto-nível que não incluem detalhes de implementação**

- Fáceis de entender e podem ser usados para comunicação com usuários não-técnicos.
- Registra os dados que irão aparecer no BD, mas não registra como estes dados serão armazenados a nível de SGBD

## **Projeto Lógico (Mapeamento do modelo de dados)**

- Transforma o esquema conceitual em um modelo de dados alto nível de acordo com um modelo de dados de implementação
- Contém detalhes do modelo de dados de implementação (Modelo Relacional)



# O modelo Relacional

# Modelo Relacional

**O modelo relacional foi introduzido por Ted Codd, da IBM Research, em 1970.**

- Chamou atenção por sua simplicidade e base matemática.

**O conceito usa o conceito de uma relação matemática como seu bloco de construção básica e tem sua base teórica na teoria de conjuntos e na lógica de predicados de primeira ordem.**

**Atualmente, é o modelo de dados mais utilizado em aplicações comerciais de processamento de dados**

# Modelo Relacional

**O modelo relacional representado banco de dados como uma coleção de relações.**

- Informalmente, cada relação se parece com uma tabela de valores

**Quando uma relação é pensada como uma tabela de valores, cada linha representa uma coleção de valores de dados relacionais.**

- Cada linha na tabela representa um fato que corresponde a uma entidade do mundo real

**O nome da tabela e nomes das colunas são usados na interpretação do significado de cada linha**

# Modelo Relacional - Terminologia

<b>Termos informais</b>	<b>Termos formais</b>
Tabela	Relação
Coluna	Atributo/Domínio
Linha	Tupla
Valores em uma coluna	Domínio
Definição da tabela	Esquema da relação
Tabela populada	Extensão

# Modelo Relacional

Uma **relação** (tabela) é composta por **atributos** (colunas) que são associados a domínios (**tipo de dados**) e por **tuplas** (linhas) que são identificadas por uma **chave** (Identificador de valor único).

**Cada atributo declarado na tabela (relação) possui um domínio de valores, ou o tipo do dado. O tipo de dado que descreve os tipos de valores que podem aparecer em cada coluna.**

- Cada domínio (tipo de dados) é um conjunto de valores atômicos
- Um Valor atômico é um valor indivisível no domínio
  - Inteiro: Não é possível dividir um inteiro;
  - Booleano: Não é possível dividir um valor booleano;



# Exemplo de uma relação - modelo relacional

Nome da relação

Atributos

ALUNO	Nome	SSN	FoneResidencia	Endereco	FoneEscritorio	Idade	MPG
	Benjamin Bayer	305-61-2435	373-1616	2918 Bluebonnet Lane	<i>null</i>	19	3.21
	Katherine Ashly	381-62-1245	375-4409	125 Kirby Road	<i>null</i>	18	2.89
	Dick Davidson	422-11-2320	<i>null</i>	3452 Elgin Road	749-1253	25	3.53
	Charles Cooper	489-22-1100	376-9821	265 Lark Lane	749-6492	28	3.93
	Barbara Benson	533-69-1238	839-8461	7384 Fontana Lane	<i>null</i>	19	3.25

Tuplas

# Representação Textual de uma tabela

# Relação ou tabela de um banco de dados

Alunos				
<u>nroMatr</u>	Nome	Telefone	Celular	Idade
123456	Marcel	(34) 3456-9999	(34) 9999-9999	20
345678	José	(64) 3234-8888	(64) 8888-8888	22
456789	Maria	(62) 3243-3333	(62) 9355-3333	24
987654	Marcos	(11) 3454-4544	(11) 9345-3476	26

# DEFINIÇÃO FORMAL - eSTRUTURA DA RELAÇÃO

Aluno (nroMatr, nome, telefone, celular, idade, cod\_endereço#)

## Definição lógica

- **Número de matrícula:** conjunto de dígitos válidos para matrícula de aluno
- **Número de telefone:** conjunto de números de telefone válido no Brasil
- **Nome de aluno:** conjunto de todos nomes possíveis para uma pessoa
- **Idade:** conjunto de idades possíveis para alunos
- Cod\_endereco

## Definição de tipo de dados

- **Números de matrícula:** inteiro de 8 dígitos
- **Números de telefone:** inteiro de 10 dígitos
- **Nomes de alunos:** string de 60 caracteres
- **Idade:** Inteiro entre 0 e 127

# rEPRESENTAÇÃO teXTUAL das tuplas de uma tabela

**Aluno (nroMatr, nome, telefone, celular, idade)**

**Representação textual dos dados de uma relação:**

**Aluno = {**  
    <123456, Marcel, (34) 3456-9999, (34) 9999-9999, 20> ,  
    <345678, José, (64) 3234-8888, (64) 8888-8888, 22> ,  
    <456789, Maria, (62) 3243-3333, (62) 9355-3333, 24> ,  
    <987654, Marcos, (11) 3454-4544, (11) 9345-3476, 26>  
**}**

# Exemplo de um banco de dados

STUDENT	Name	StudentNumber	Class	Major
	Smith	17	1	CS
	Brown	8	2	CS

COURSE	CourseName	CourseNumber	CreditHours	Department
	Intro to Computer Science	CS1310	4	CS
	Data Structures	CS3320	4	CS
	Discrete Mathematics	MATH2410	3	MATH
	Database	CS3380	3	CS

SECTION	SectionIdentifier	CourseNumber	Semester	Year	Instructor
	85	MATH2410	Fall	98	King
	92	CS1310	Fall	98	Anderson
	102	CS3320	Spring	99	Knuth
	112	MATH2410	Fall	99	Chang
	119	CS1310	Fall	99	Anderson
	135	CS3380	Fall	99	Stone

GRADE_REPORT	StudentNumber	SectionIdentifier	Grade
	17	112	B
	17	119	C
	8	85	A
	8	92	A
	8	102	B
	8	135	A

PREREQUISITE	CourseNumber	PrerequisiteNumber
	CS3380	CS3320
	CS3380	MATH2410
	CS3320	CS1310

Instância do banco de dados de exemplo

# Características de um banco de dados relacional

# Características principais

- Toda informação em um banco de dados relacional deve ser apresentada em nível lógico por valores em tabelas;
- Todo dado em um banco de dados relacional tem garantia de ser logicamente acessível.
- Uma tabela é acessível por qualquer campo (atributo), independente de ele ser declarado como chave ou não.
- O SGBD relacional deve ter uma linguagem para definição, detalhamento e manipulação de dados (SQL);



# Características principais

## Ordenação das tuplas em uma relação (nível abstrato)

- Matematicamente não há ordem entre os elementos de um conjunto.
- Na implementação de um SGBD existe uma ordem física de armazenamento das tuplas na memória externa. Essa ordem é apresentada na recuperação das informações, mas **não pode-se dizer que os dados estão ordenados.**

## Ordenação de tuplas em uma relação (nível lógico)

- Em nível lógico, muitas ordens podem ser definidas para uma relação:
  - Relação Aluno pode ser ordenada por NOME, Data de Nascimento, etc.

# Características principais

## **Uma tupla é uma lista ordenada de $n$ valores**

- Existe uma ordem dos atributos de uma tupla que segue a disposição dos atributos na definição do esquema da relação.

## **Os valores presentes nas tuplas deve estar de acordo com seu domínio (tipo de dados) e ser atômicos.**

- **São atômicos (monovalorados)**
  - Valores multivalorados e compostos não são permitidos
- **O valor null é usado quando o valor de um atributo é desconhecido ou não se aplica a uma determinada tupla**

# COnceito de chaves no modelo Relacional

# Chaves no modelo relacional

As chaves são atributos utilizados para especificar restrições de integridade na tabela. As chaves podem ser:

- **Chave primária:** é um conjunto mínimo de atributos, normalmente um único atributo, pertencente a tabela, que identifica unicamente um registro;(tupla/linha).
  - Uma tabela **deve ter uma única chave primária que não é permitido ser preenchida com valores vazios** (NOT NULL);
  - O conteúdo de uma chave-primária deve ser único, exclusivo e imutável para cada linha dessa entidade.
  - Normalmente utiliza-se valores numéricos por questões de performance.

# Chaves no modelo relacional

- **Chave candidata:** As chaves candidatas possuem características semelhantes as chaves primárias (identificam valores de forma única, não podem ter valores vazios (NOT NULL)). Porém não se tornam chaves primárias pois em uma tabela somente um atributo, ou conjunto de atributos, pode assumir o papel de chave-primária.
  - Esta chave é utilizada como meio de classificação e pesquisa de entidades
- **Chave estrangeira:** Atributo ou um conjunto de atributos que permitem o relacionamento entre tabelas. Seus valores (domínio) devem aparecer necessariamente como chave primária de uma outra tabela.

# Exemplo

**Aluno (nroMatr, nome, telefone, celular, idade, CPF)**

- Número de matrícula (**nroMatr**) é a **chave-primária**;
- **CPF é uma chave candidata** (também identifica unicamente cada aluno da relação Aluno)

Outros atributos podem, ou não, ser chaves candidatas. Tal decisão depende de regras definidas no escopo do problema.

Idade não pode ser chave primária ou chave candidata.

# Restrições Sobre uma tabela

# Restrições sobre uma relação (Tabela)

**Restrição de Domínio:** Especificam que, dentro de cada tupla, o valor de cada atributo deve ser um valor atômico do domínio (tipo de dados) especificado.

**Unicidade de chave:** Conjunto mínimo de atributos que identifica de forma única cada tupla da relação. (Chave-primária)

**Integridade de entidade:** nenhum valor de chave primária pode ser nulo.

- Permitir valor nulo em chave primária implica que não pode-se identificar algumas tuplas, não podendo identificá-las ou referenciá-las.



# Restrições sobre uma relação

**Restrição de valor Null:** Especificam se o valor **null** é ou não permitido para um determinado atributo. Por padrão o valor null é permitido, sendo necessário especificar quando não é permitido (**NOT NULL**)

**Por definição, todos os elementos de um conjunto são distintos, por isso, todos as tuplas da relação devem ser distintas**

- Duas linhas nunca terão a mesma combinação de valores para todos seus atributos (no mínimo a chave-primária deve ser diferente);

# Restrição de chave estrangeira

**Chave estrangeira:** Atributo ou um conjunto de atributos que permitem o relacionamento entre tabelas. Seus valores (domínio) devem aparecer necessariamente como chave primária de uma outra tabela.

**Uma chave estrangeira (Foreign Key - FK) de uma relação R1 é um atributo que faz referência à uma outra relação R2.**

- Os atributos Chave-Estrangeira da tabela R1 têm o(s) mesmo(s) domínio(s) (tipo de dados) que os atributos da chave primária da tabela R2.
- O valor da Chave estrangeira pode ser NULL quando o relacionamento é opcional e não acontece naquela tupla (linha).