NORMALIZAÇÃO DE DADOS

AULA 08

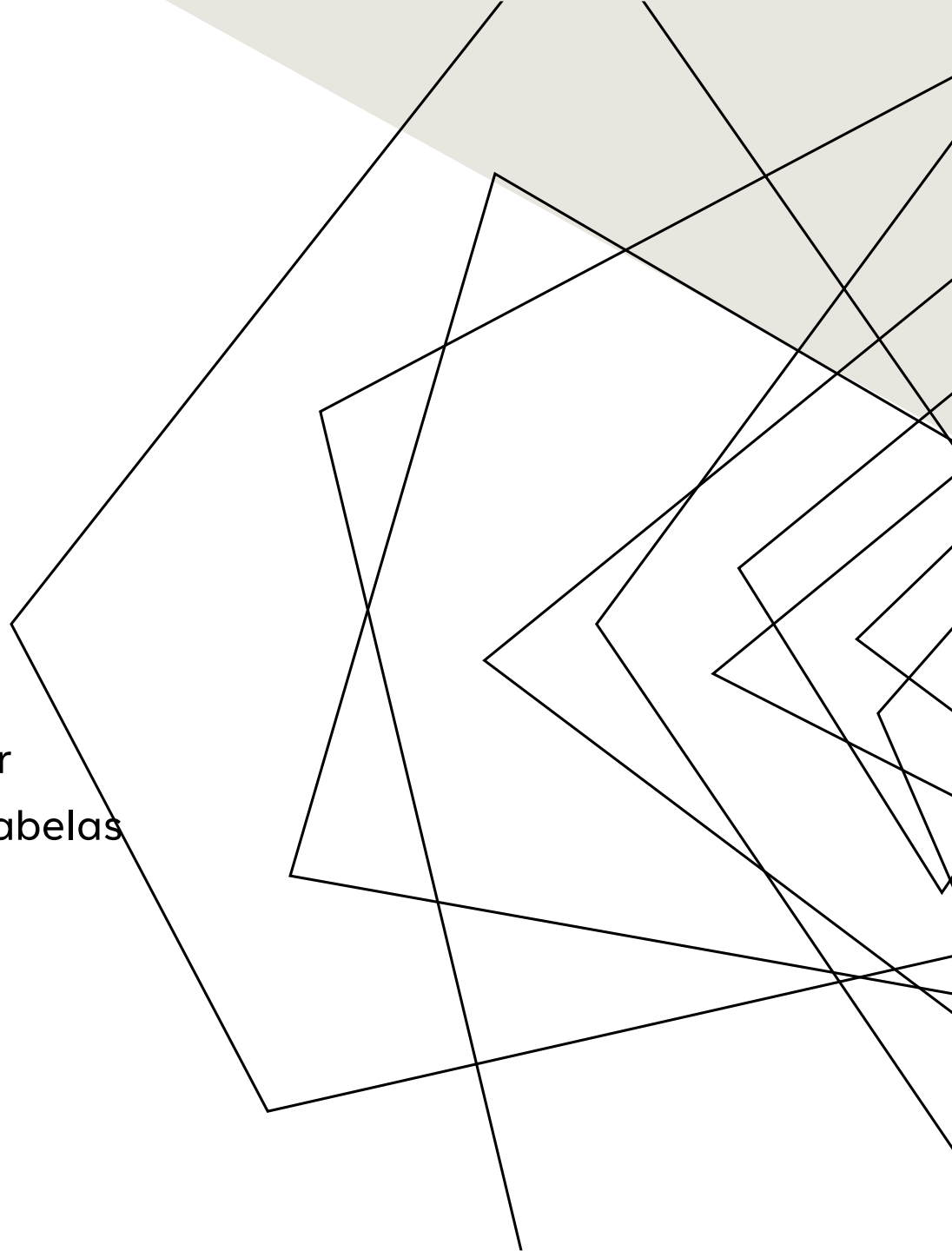
PROFESSOR: EDUARDO KAZENSKI

CONTEÚDO

- Modelagem de dados.
    - Normalização de Dados.

OBJETIVOS

- Entender o processo de organizar os dados para reduzir redundâncias e melhorar a integridade, decompondo tabelas complexas em tabelas menores e mais simples



Pense no seu guarda-roupa. Se você simplesmente jogar todas as suas roupas lá dentro (calças, camisetas, meias, casacos), na hora de encontrar uma peça específica, será uma bagunça, certo? Você perde tempo e pode até pegar a roupa errada.

A normalização é como **organizar esse guarda-roupa**. Em vez de um monte de roupas jogadas, você cria gavetas e seções: uma para camisetas, outra para calças, outra só para meias. Tudo fica mais fácil de achar, de guardar e de manter limpo.

NORMALIZAÇÃO DE DADOS – FUNDAMENTOS

- Processo de análise que visa assegurar que uma relação está bem formada.
- Ideia: **decompor relações** (tabelas) com anomalias para produzir relações menores e mais bem estruturadas.
- As tabelas normalizadas podem passar por: Inserção, remoção, atualização isentas de **anomalias**.

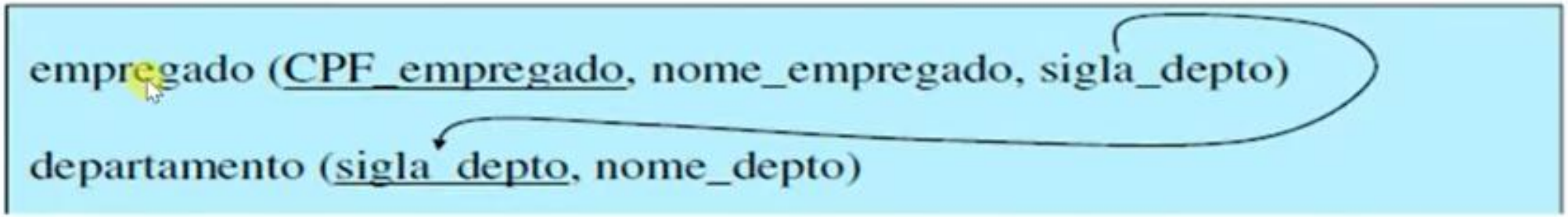
- A normalização é um conjunto de regras que usamos para organizar as tabelas do nosso banco de dados para evitar esses problemas. As regras mais famosas são as **Formas Normais (FN)**. Vamos focar nas três primeiras, que são as mais usadas no dia a dia.

NORMALIZAÇÃO DE DADOS – ANOMALIAS

- São problemas que ocorrem em Bancos de dados
- Decorrem de mau planejamento (na hora de modelar nossos dados), podendo:
    - não modelar de forma muito adequada no modelo de Entidade-Relacionamentos;
    - fazer uma modelagem boa, mas implementamos errado;
- Há 3 tipos de anomalias:
    - Anomalia de Inserção;
    - Anomalia de Remoção;
    - Anomalia de Atualização.

ANOMALIA DE INSERÇÃO

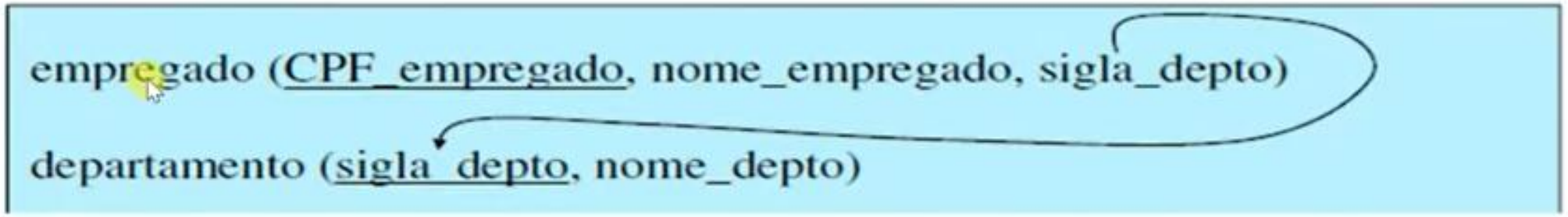
- Há uma ordem para inclusão em relações. Veja o exemplo:



  - Não é possível inserir uma tupla em empregado sem que antes tenhamos inserido uma tupla pro departamento a ser referenciado.
  - Tentativa de inserção fora de ordem gera anomalia de inclusão.

ANOMALIA DE EXCLUSÃO

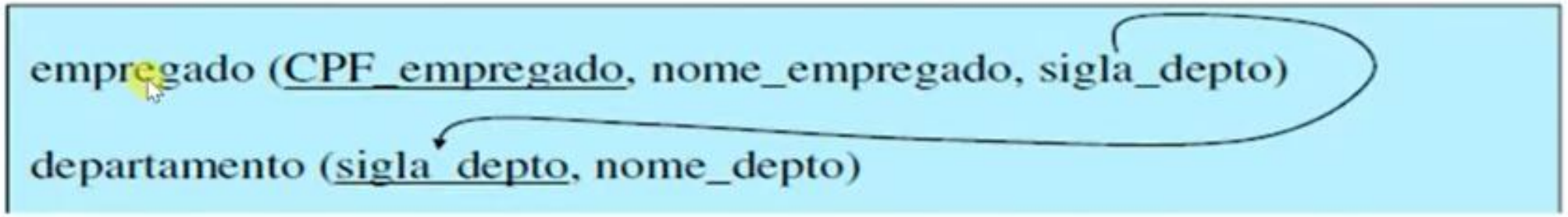
- Ao excluir um registro em uma relação, os dados referentes ao registro em outras relações devem ser excluídos também.



  - A remoção de um departamento deve remover tuplas referentes a ele em empregado. Caso contrário, temos uma anomalia de exclusão.
  - Ou colocar NULL na sigla de departamento.

ANOMALIA DE ATUALIZAÇÃO

- Ao alterar um registro em uma relação, os dados referentes aos registros em outras relações devem ser alterados também.



  - A atualização da sigla de um departamento deve atualizar as tuplas referentes a ele em empregado. Caso contrário, temos uma anomalia de atualização.

NORMALIZAÇÃO DE DADOS

- Conceito introduzido em 1972 por Codd
- Normalização = conjunto de testes para certificar que uma relação satisfaz as **formas normais (FN)**
- **Codd propôs que boas relações (sem anomalias) atendem ao menos 3 formas normais:**
  - **1FN**
  - **2FN**
  - **3FN**
- Existem outras formas, mas seguiremos apenas com essas 3.

PRIMEIRA FORMA NORMAL 1FN

1FN

- Reprova atributos multivalorados, compostos (que podem ser divididos em atributos menores) e suas combinações.
- Cada coluna de uma tabela deve ter APENAS um valor (ser atômica)
- Não pode haver grupos de colunas repetidas.

Problema: A coluna Telefones tem dois valores na mesma célula para o João. Isso dificulta a busca por um telefone específico. Como você procuraria apenas o número (48) 8888-7777? Complicado!

AlunoID	Nome	Telefones
1	João Silva	(48) 9999-8888, (48) 8888-7777
2	Maria Costa	(49) 7777-6666

PRIMEIRA FORMA NORMAL 1FN

1FN

- A solução mais assertiva para o problema seria criar uma segunda tabela que detalha todos possíveis registros de Telefones para todos ID da tabela principal:

AlunoID	Nome	Telefones
1	João Silva	(48) 9999-8888, (48) 8888-7777
2	Maria Costa	(49) 7777-6666

Agora sim! Cada célula tem um único valor. Mas note que o nome "João Silva" está repetido. Isso nos leva à próxima forma normal.

AlunoID	Nome	Telefone
1	João Silva	(48) 9999-8888
1	João Silva	(48) 8888-7777
2	Maria Costa	(49) 7777-6666

PRIMEIRA FORMA NORMAL 2FN

2FN

- A tabela já deve estar na 1FN.
- Todas as colunas que não são chave primária devem depender da chave primária inteira, e não apenas de parte dela. (Essa regra faz mais sentido quando temos uma chave primária composta por mais de uma coluna).
- Vamos criar uma tabela para controlar os itens de um pedido. A chave primária aqui é a combinação de **PedidoID** e **ProdutoID**.

PedidoID	ProdutoID	NomeProduto	PrecoProduto	Quantidade	DataPedido
101	1	Teclado Gamer	R$ 150,00	1	28/06/2025
101	2	Mouse Vertical	R$ 90,00	1	28/06/2025
102	1	Teclado Gamer	R$ 150,00	2	29/06/2025

PRIMEIRA FORMA NORMAL 2FN

2FN

- **NomeProduto** e **PrecoProduto** dependem apenas do **ProdutoID**. Não importa qual o pedido, um Teclado Gamer sempre será um Teclado Gamer.
- **DataPedido** depende apenas do **PedidoID**.
- A única coluna que depende da chave inteira (**PedidoID** + **ProdutoID**) é a Quantidade.
- Redundância: O nome e o preço do "Teclado Gamer" se repetem. A data do pedido 101 também. Imagine se o preço do teclado mudar? Teríamos que alterar em todas as linhas onde ele aparece!

PedidoID	ProdutoID	NomeProduto	PrecoProduto	Quantidade	DataPedido
101	1	Teclado Gamer	R$ 150,00	1	28/06/2025
101	2	Mouse Vertical	R$ 90,00	1	28/06/2025
102	1	Teclado Gamer	R$ 150,00	2	29/06/2025

PRIMEIRA FORMA NORMAL 2FN

2FN

- A SOLUÇÃO é dividirmos a tabela em três! Cada uma com sua responsabilidade.

Tabela: Pedidos

PedidoID (PK)	DataPedido
101	28/06/2025
102	29/06/2025

Tabela: Produtos

ProdutoID (PK)	NomeProduto	PrecoProduto
1	Teclado Gamer	R$ 150,00
2	Mouse Vertical	R$ 90,00

Tabela: Item_pedido

PedidoID (FK)	ProdutoID (FK)	Quantidade
101	1	1
101	2	1
102	1	2

Agora, se o preço do teclado mudar, alteramos em um único lugar: na tabela Produtos. Sem repetição, sem bagunça!

PRIMEIRA FORMA NORMAL 3FN

3FN

- A tabela já deve estar na 2FN.
- Nenhuma coluna que não é chave pode depender de outra coluna que também não é chave. É a chamada dependência transitiva.
- Imagine uma tabela de funcionários que mostra o departamento onde eles trabalham.

Tabela: Funcionarios

FuncionarioID (PK)	NomeFuncionario	DepartamentoID	NomeDepartamento
1	Ana	10	Vendas
2	Carlos	20	TI
3	Bia	10	Vendas

PRIMEIRA FORMA NORMAL 3FN

3FN

- Problema:
  - A coluna **NomeDepartamento** depende do **DepartamentoID**. E o **DepartamentoID** depende do **FuncionarioID**.
  - Isso é uma dependência transitiva: **FuncionarioID** -> **DepartamentoID** -> **NomeDepartamento**.
  - A **NomeDepartamento** não depende diretamente da chave primária (**FuncionarioID**), mas sim de outra coluna (**DepartamentoID**).
  - Redundância: O nome "Vendas" está repetido. Se a empresa decidir renomear o departamento de "Vendas" para "Comercial", teremos que atualizar a linha da Ana e da Bia.

PRIMEIRA FORMA NORMAL 3FN

3FN

- A SOLUÇÃO é dividirmos a tabela em três! Cada uma com sua responsabilidade.

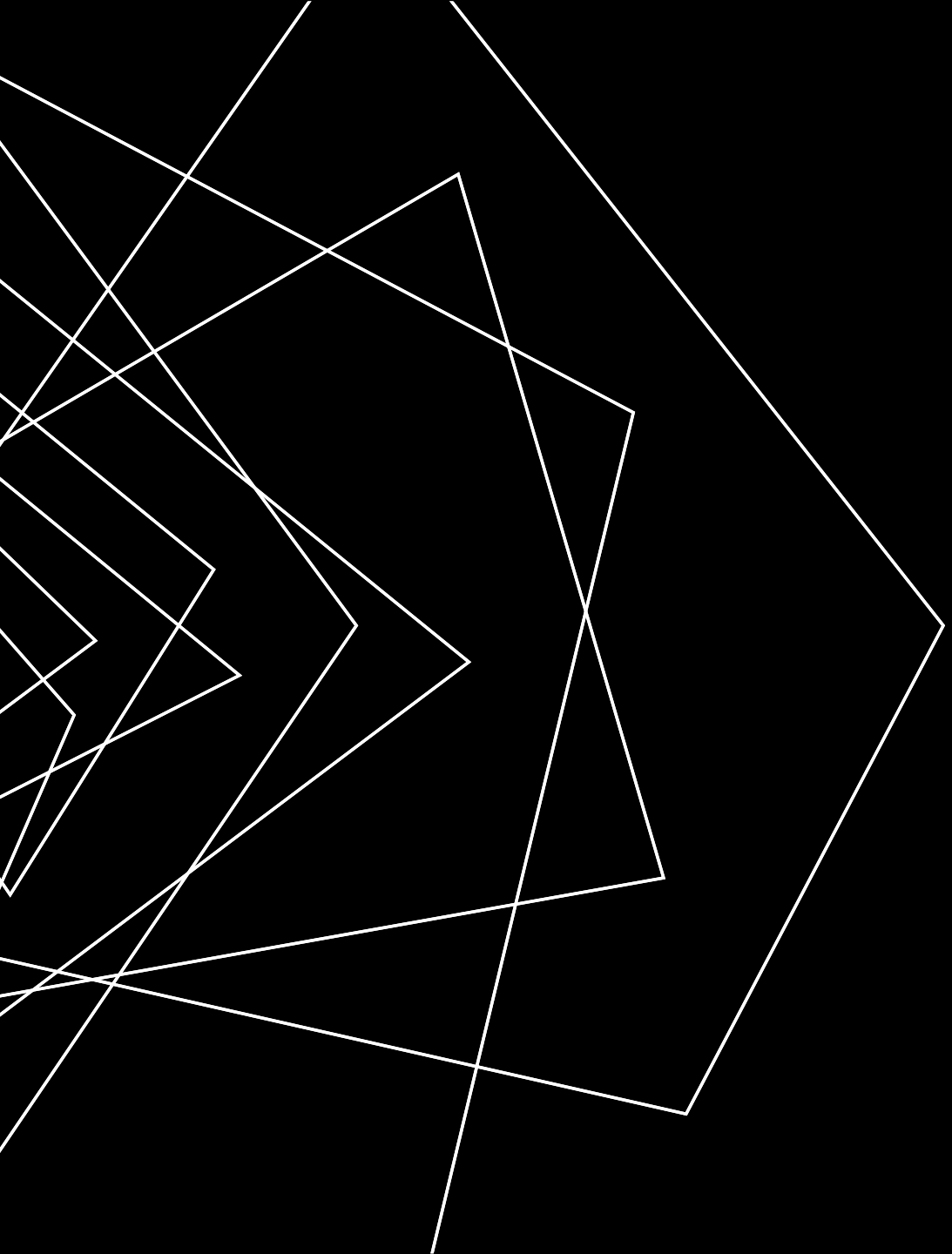
Tabela: Funcionarios

FuncionarioID (PK)	NomeFuncionario	DepartamentoID (FK)
1	Ana	10
2	Carlos	20
3	Bia	10

Tabela: Departamentos

DepartamentoID (PK)	NomeDepartamento
10	Vendas
20	TI

Perfeito! Agora, para mudar o nome do departamento, só precisamos alterar um registro na tabela Departamentos.



FIM DE AULA