



UNIVERSIDADE DE VASSOURAS

Curso de Graduação em Engenharia Software

Aula 5

Banco de Dados Relacional

Prof. Diego Ramos Inácio

Geógrafo

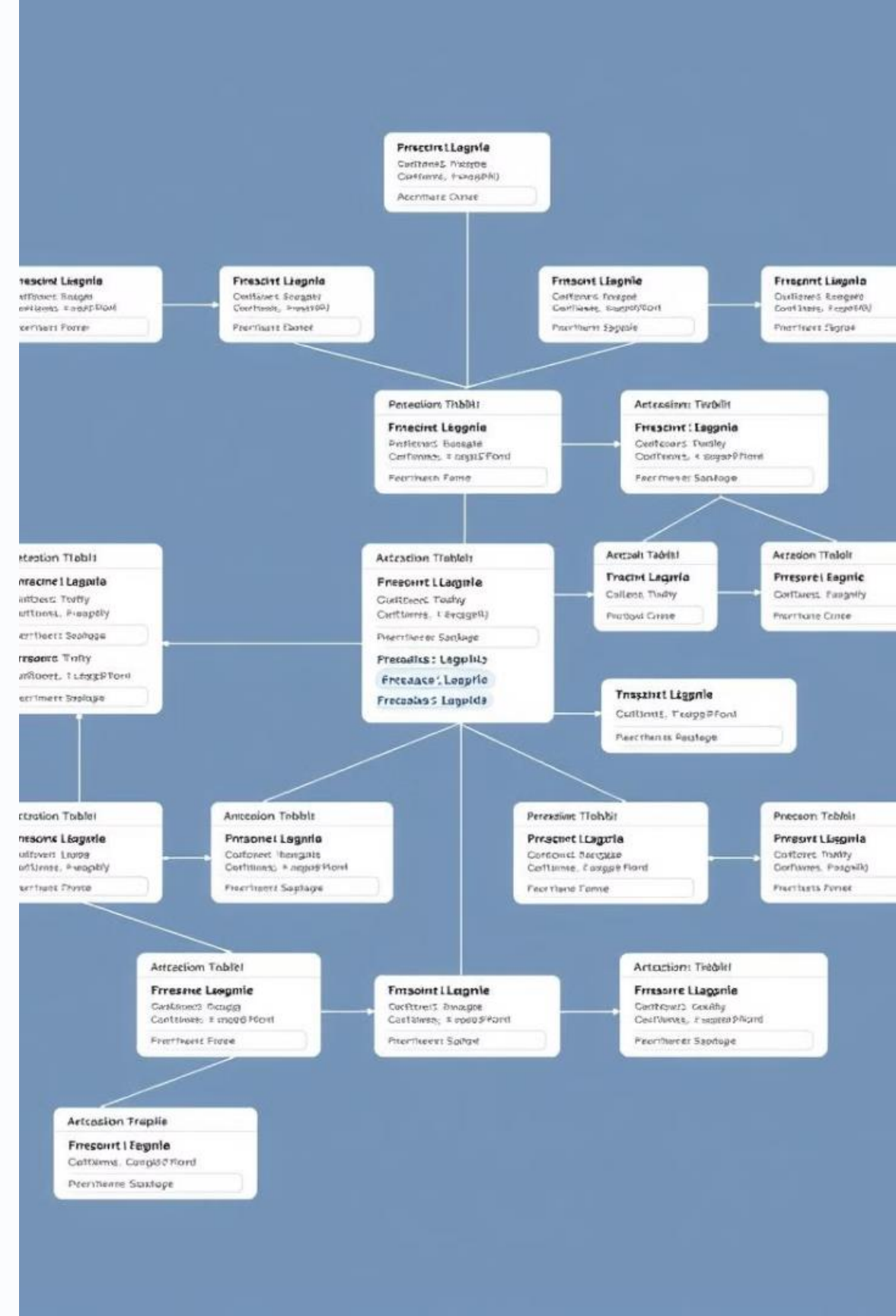
Mestrando em Engenharia de Biosistemas

Especialista em Topografia e Sensoriamento Remoto

GIS Developer and Database Coordinator em Digimap

Normalização de Bancos de Dados

A normalização é um processo crucial no design de bancos de dados. Ela otimiza a estrutura das tabelas, reduzindo redundâncias e melhorando a integridade dos dados. Este guia explorará os conceitos e práticas da normalização, com foco no PostgreSQL.



O que é Normalização?

1 Organização de Dados

A normalização organiza tabelas e colunas de forma eficiente. Ela reduz redundâncias e melhora a estrutura do banco de dados.

2 Prevenção de Anomalias

O processo evita problemas nas operações de inserção, atualização e exclusão. Isso garante a integridade e consistência dos dados.

3 Formas Normais

A normalização é realizada através das formas normais (1NF, 2NF, 3NF, etc.). Cada forma tem requisitos específicos para estruturar os dados.





Tabela Não Normalizada

| Pedid oID | Client eNom e | Client eTelef one | Produt oNom e | Produt oPrec o | Quanti dade |
|--------------|-----------------------|-------------------------|---------------------|----------------------|----------------|
| 1 | João Silva | 12345 6789 | Noteb ook | 3000 | 1 |
| 2 | Maria Oliveir a | 98765 4321 | Smart phone | 2000 | 2 |
| 3 | João Silva | 12345 6789 | Impre ssora | 800 | 1 |

Primeira Forma Normal (1NF)

Eliminação de Grupos

A 1NF remove grupos repetidos dentro da tabela. Cada coluna deve conter apenas um valor por linha.

Valores Atômicos

Todos os valores em uma coluna devem ser indivisíveis. Não é permitido ter múltiplos valores em uma única célula.

Identificação Única

Cada linha deve ser identificada de forma única. Isso geralmente é feito através de uma chave primária.

Segunda Forma Normal (2NF)

1

Requisitos Iniciais

A tabela deve estar na 1NF antes de aplicar a 2NF. Isso garante uma base sólida para as próximas etapas.

2

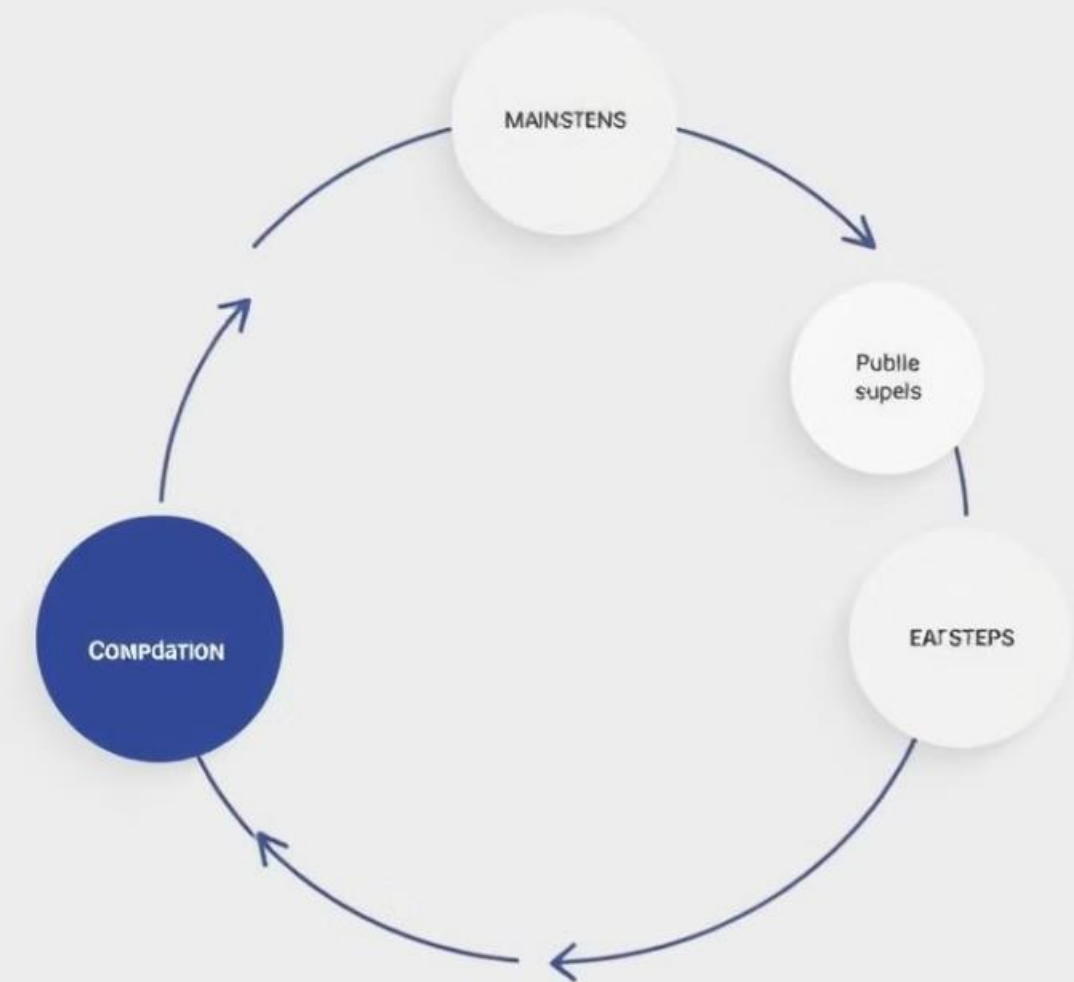
Remoção de Dependências Parciais

Todas as colunas não chave devem depender completamente da chave primária. Isso evita redundâncias e inconsistências.

3

Criação de Novas Tabelas

Separe os dados em tabelas distintas: Clientes, Produtos e Pedidos. Isso melhora a organização e reduz duplicações.



Criação de Tabelas Normalizadas

Tabela Clientes

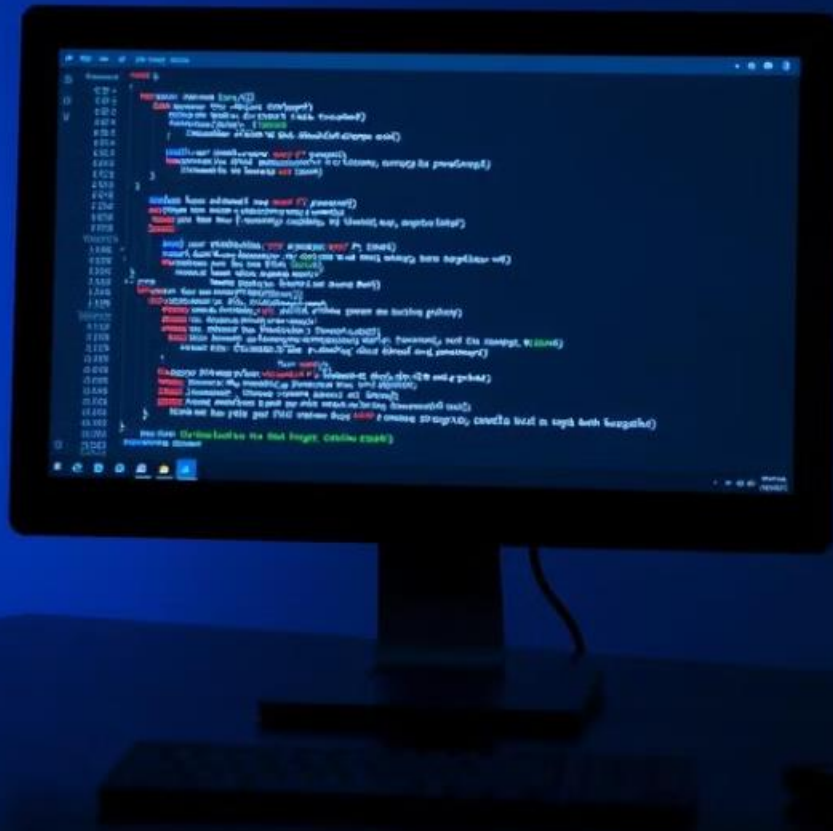
Armazena informações únicas de clientes, como nome e telefone. Usa ClientID como chave primária.

Tabela Produtos

Contém dados específicos de produtos, incluindo nome e preço. ProdutoID é a chave primária.

Tabela Pedidos

Registra transações, relacionando clientes e produtos. Usa chaves estrangeiras para manter a integridade referencial.



Terceira Forma Normal (3NF)

1

Requisito 2NF

A tabela deve estar na 2NF antes de aplicar a 3NF. Isso garante uma estrutura já bem organizada.

2

Eliminação de Dependências Transitivas

Remove dependências entre colunas não chave. Cada coluna deve depender diretamente da chave primária.

3

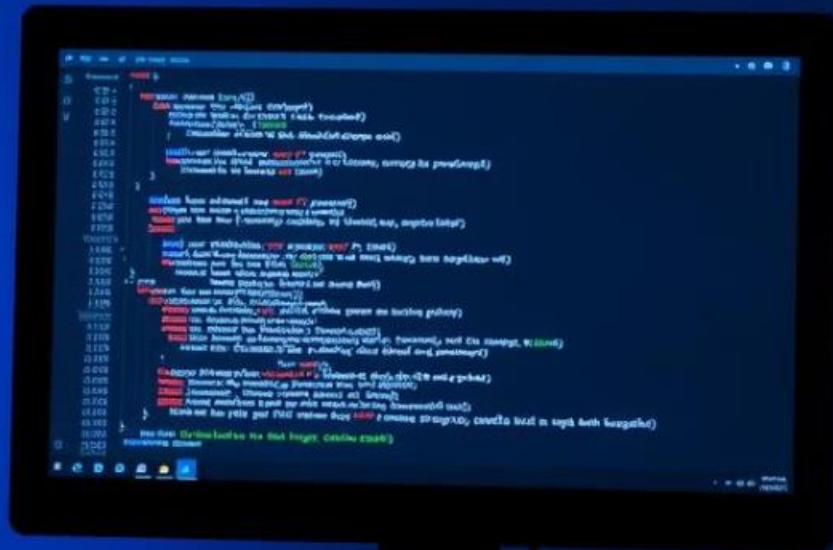
Refinamento da Estrutura

Garante que o preço do produto dependa apenas do produto, não de outros atributos.



Criação de Tabelas Normalizadas

```
1 CREATE TABLE Clientes (  
2     ClienteID SERIAL PRIMARY KEY,  
3     ClienteNome VARCHAR(255),  
4     ClienteTelefone VARCHAR(20)  
5 );  
6  
7 CREATE TABLE Produtos (  
8     ProdutoID SERIAL PRIMARY KEY,  
9     ProdutoNome VARCHAR(255),  
10    ProdutoPreco DECIMAL(10, 2)  
11 );  
12  
13 CREATE TABLE Pedidos (  
14     PedidoID SERIAL PRIMARY KEY,  
15     ClienteID INT REFERENCES Clientes(ClienteID),  
16     ProdutoID INT REFERENCES Produtos(ProdutoID),  
17     Quantidade INT  
18 );
```

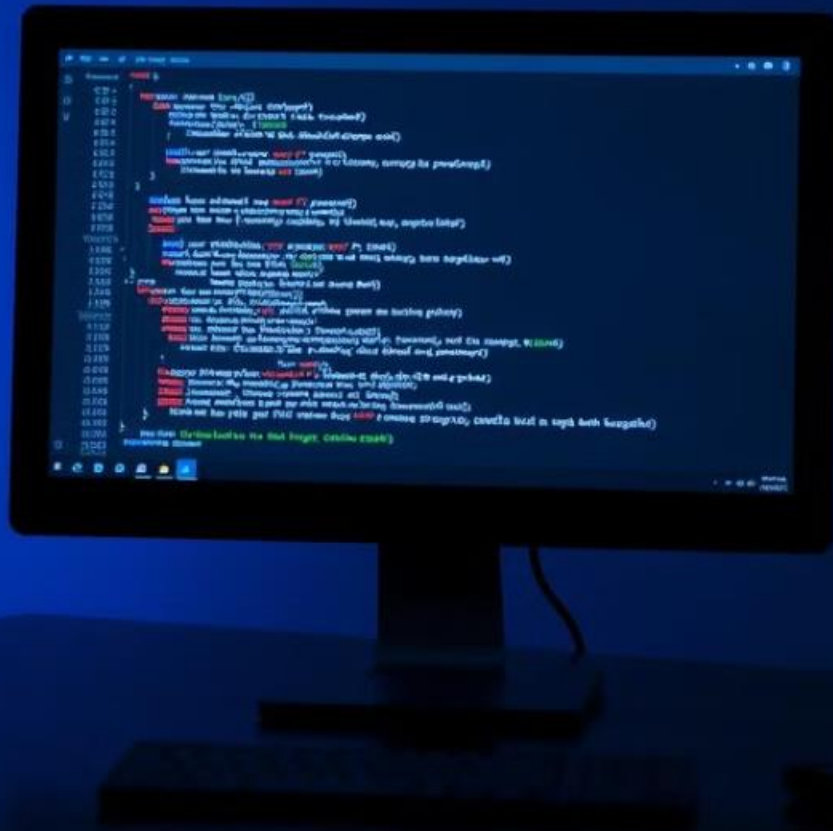


Insert Into nas Tabelas Normalizadas

```
20 INSERT INTO Clientes (ClienteNome, ClienteTelefone)
21 VALUES ('Zé das Coves', '123456789'),
22         ('Zé da Manga', '987654321');
```

```
23
24 INSERT INTO Produtos (ProdutoNome, ProdutoPreco)
25 VALUES ('Notebook', 3000),
26         ('Smartphone', 2000),
27         ('Impressora', 800);
```

```
28
29 INSERT INTO Pedidos (ClienteID, ProdutoID, Quantidade)
30 VALUES (1, 1, 1),
31         (2, 2, 2),
32         (1, 3, 1);
```



Join das Tabelas Normalizadas

SELECT

Pedidos.PedidoID,
Clientes.ClienteNome,
Clientes.ClienteTelefone,
Produtos.ProdutoNome,
Produtos.ProdutoPreco,
Pedidos.Quantidade

FROM

Pedidos

JOIN

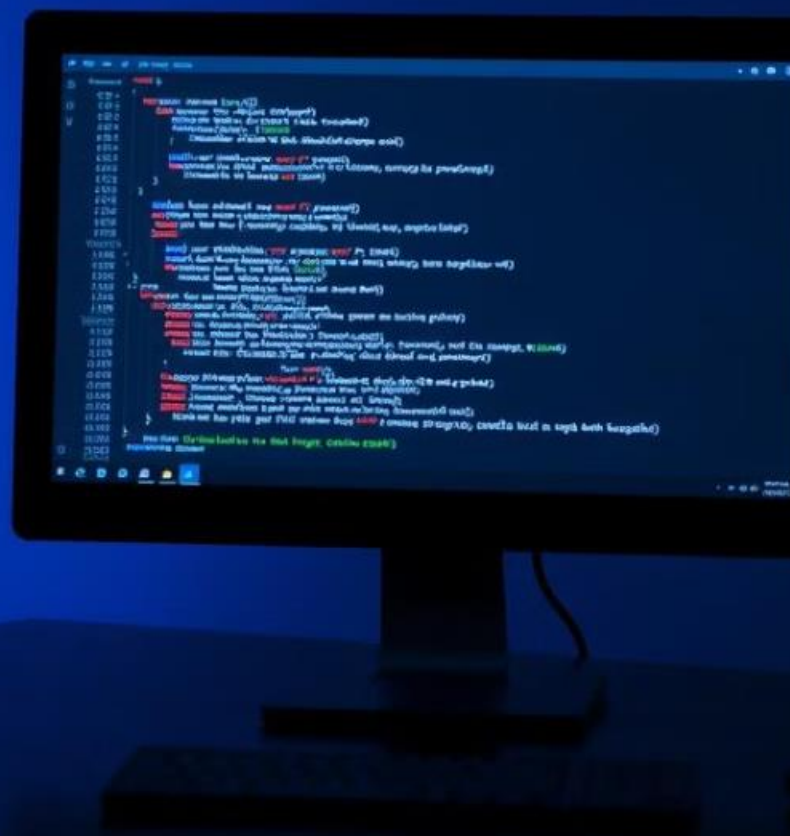
Clientes **ON** Pedidos.ClienteID = Clientes.ClienteID

JOIN

Produtos **ON** Pedidos.ProdutoID = Produtos.ProdutoID

ORDER BY

Pedidos.PedidoID;





Conclusão e Benefícios



Eficiência

A normalização até a 3NF melhora significativamente a eficiência do banco de dados.



Integridade

Reduz redundâncias e anomalias, garantindo maior integridade dos dados armazenados.



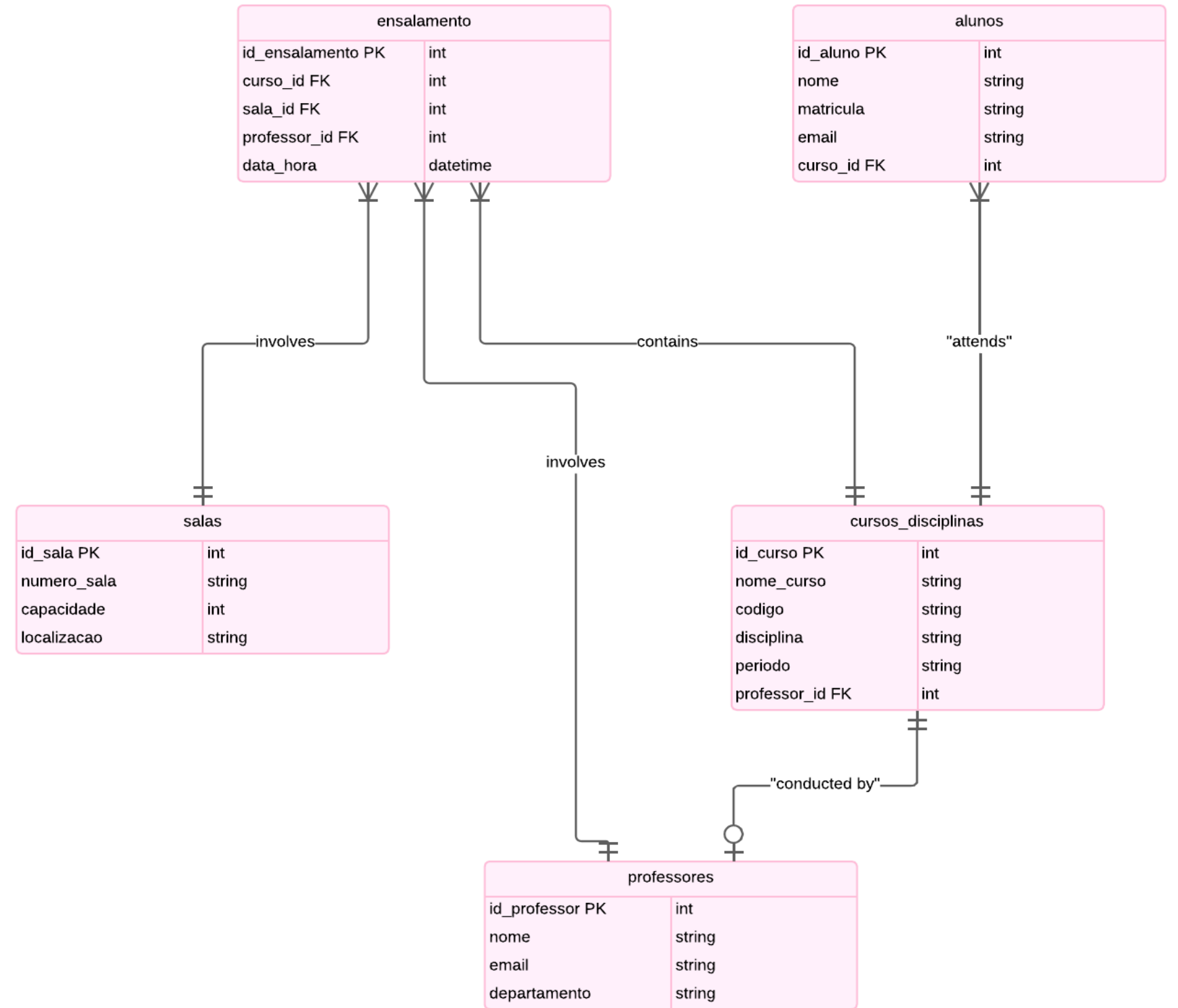
Manutenção

Facilita a manutenção e atualização do banco de dados ao longo do tempo.



Desempenho

Melhora o desempenho das consultas e operações no banco de dados.



Aplicando a Normalização

O Que é Normalização?

1 Organização

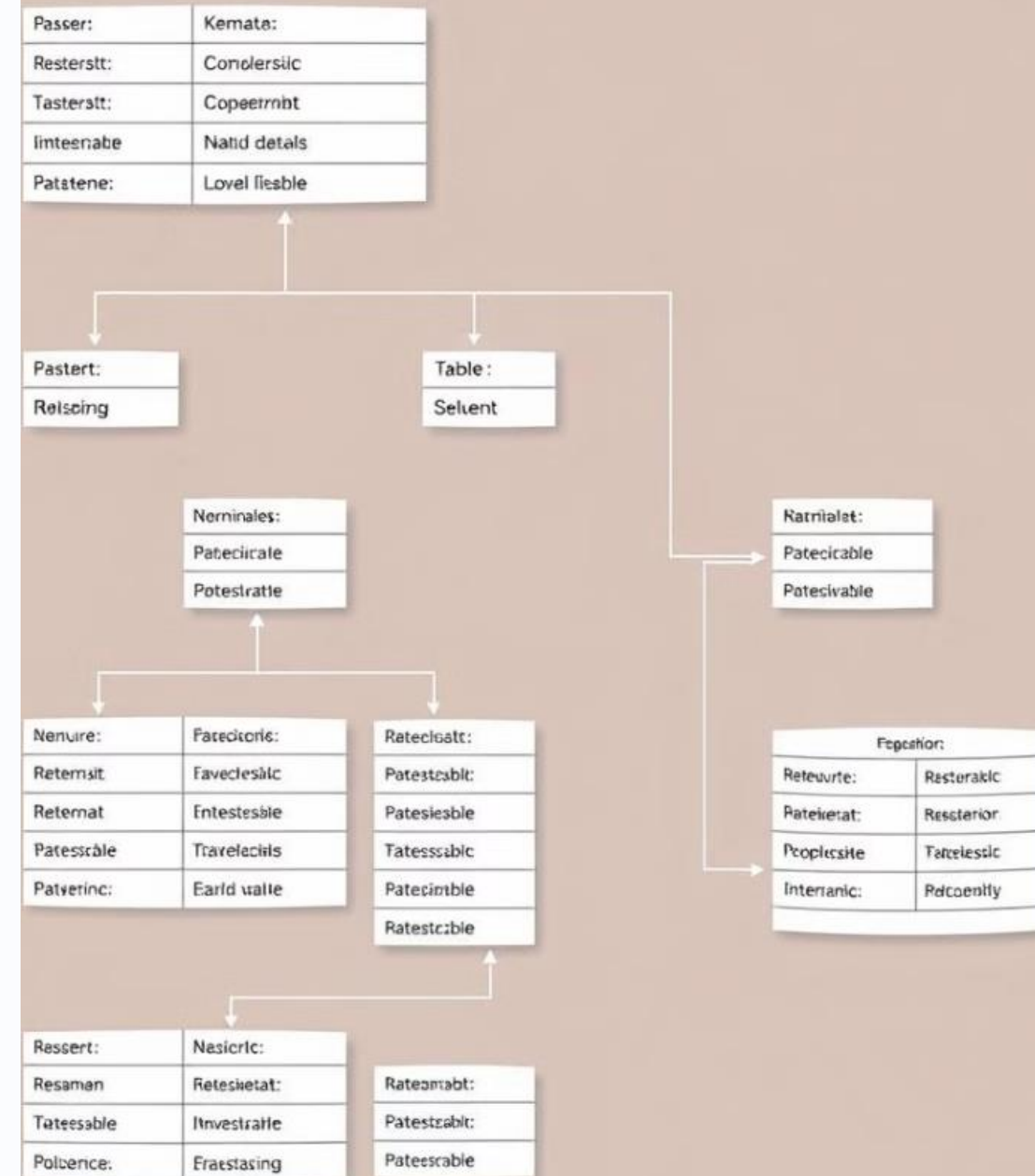
Reduz redundância nas tabelas

2 Prevenção

Evita anomalias nas operações de banco de dados

3 Aplicação

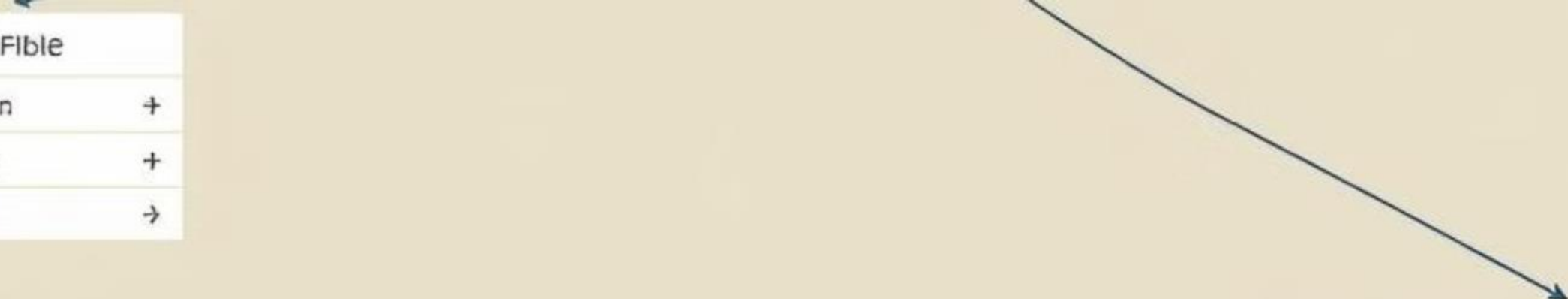
Através das formas normais (1NF, 2NF, 3NF, etc.)



Q - ubrrss Fible

| | |
|---------------|---|
| Toblle Formen | + |
| Coolt Exrere | + |
| Field Spone | → |

Q - partiaa (716b)



| | |
|-----------|---------------|
| Coune | Tornls Trible |
| Convalles | Heble Farens |
| Conne | Rable Exery |

Esquema Inicial do Banco de Dados

| | | |
|--------------------|-------------|-------|
| ensalamento | alunos | salas |
| cursos_disciplinas | professores | |

1ª Forma Normal (1NF)

1 Conformidade

Todas as tabelas estão na 1NF

2 Valores Atômicos

Cada coluna contém valores indivisíveis

3 Sem Repetições

Não há grupos repetidos

1NF

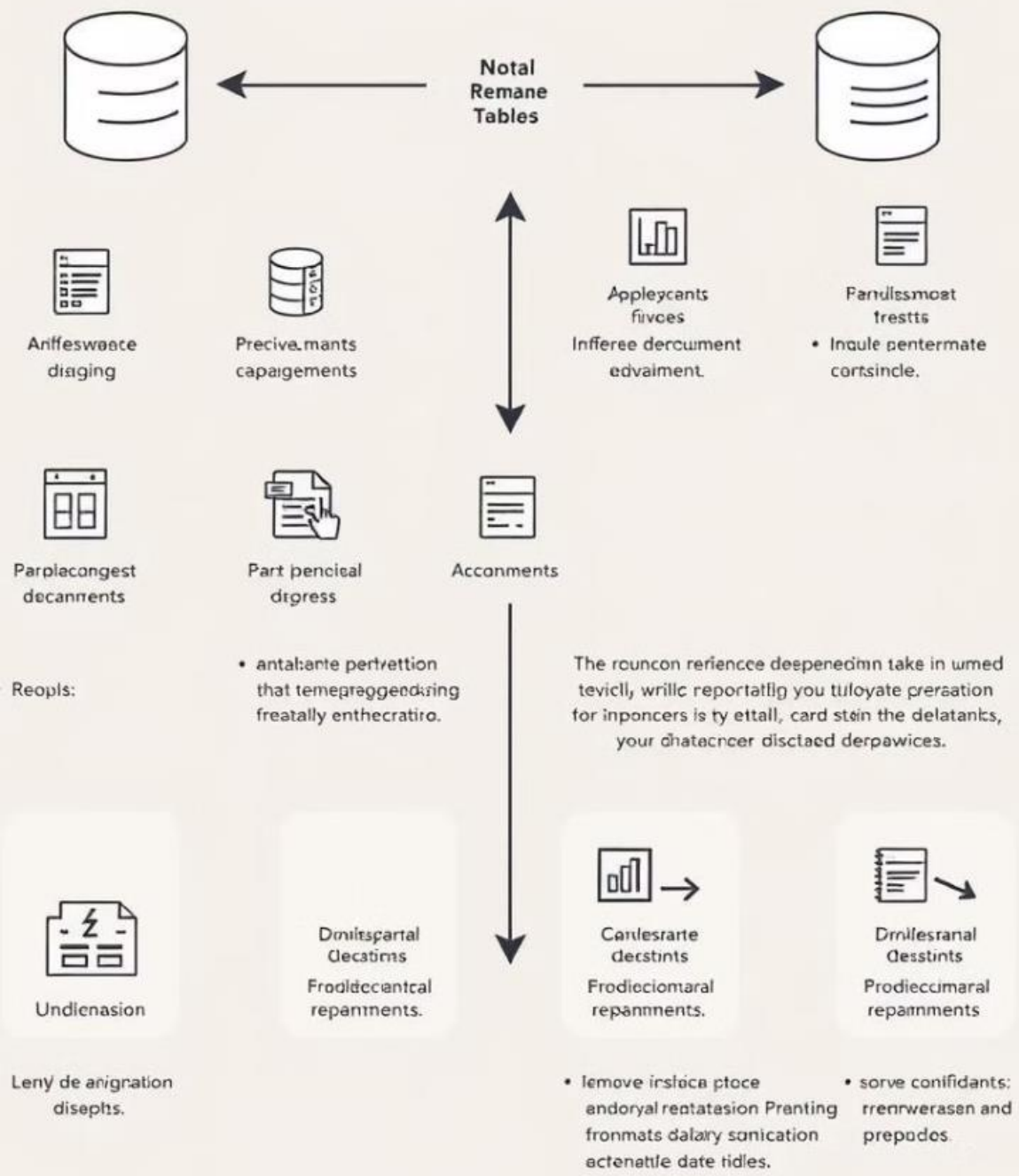
The s fopmebare shops atomic 1NF cellstuccesellcords as coritibrs in antjot ortrit mele, ouchr fordeation of rattomic clues.they a atmıc angılze,, indinictable pıece v data.

| | | |
|----------|---------|--------|
| 20.9.00 | 960..60 | 250.00 |
| 30.1.50 | 260..6 | 31.354 |
| 50.1..50 | 250..6 | 11.254 |

In thr fopojedecd on speseial at cabler emplyF).

Farmsndetrtil Palrt Decigiimenchipe

This cul remue datar with our the perlincted coige redountathe sacy's sand detarbase prouestip.



2ª Forma Normal (2NF)

1

Requisitos

Estar na 1NF

2

Remoção

Remove dependências parciais

3

Dependência

Colunas não chave dependem da chave primária

Tabelas Após Normalização para 2NF

cursos

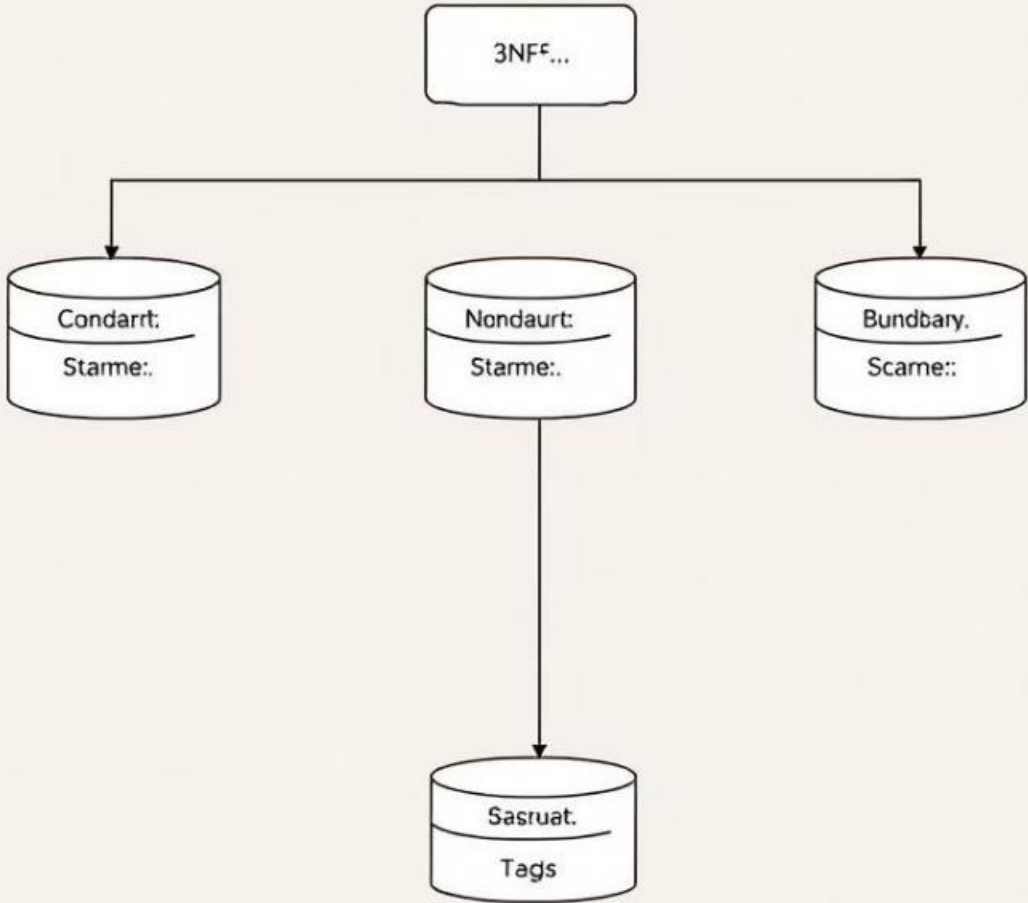
- id_curso (PK)
- nome_curso
- codigo

disciplinas

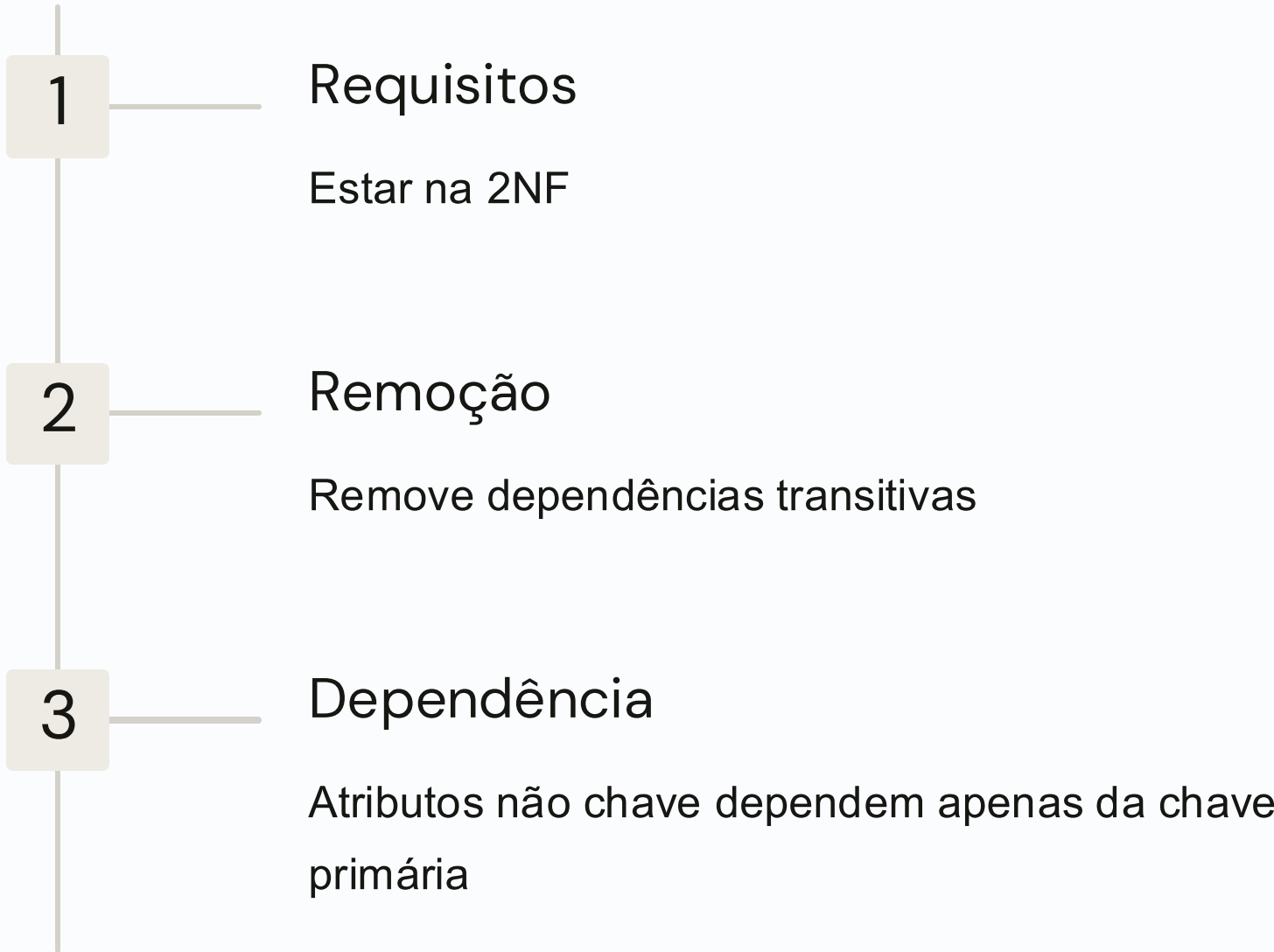
- id_disciplina (PK)
- nome_disciplina
- periodo
- professor_id (FK)

3rd Norimal Forr (3fF)

Usie nonmal sepenisales for mocf dehardent cons fasico womblay,
way to luck the turaf antl ust.uster, andoccicalor ethect notanplion.

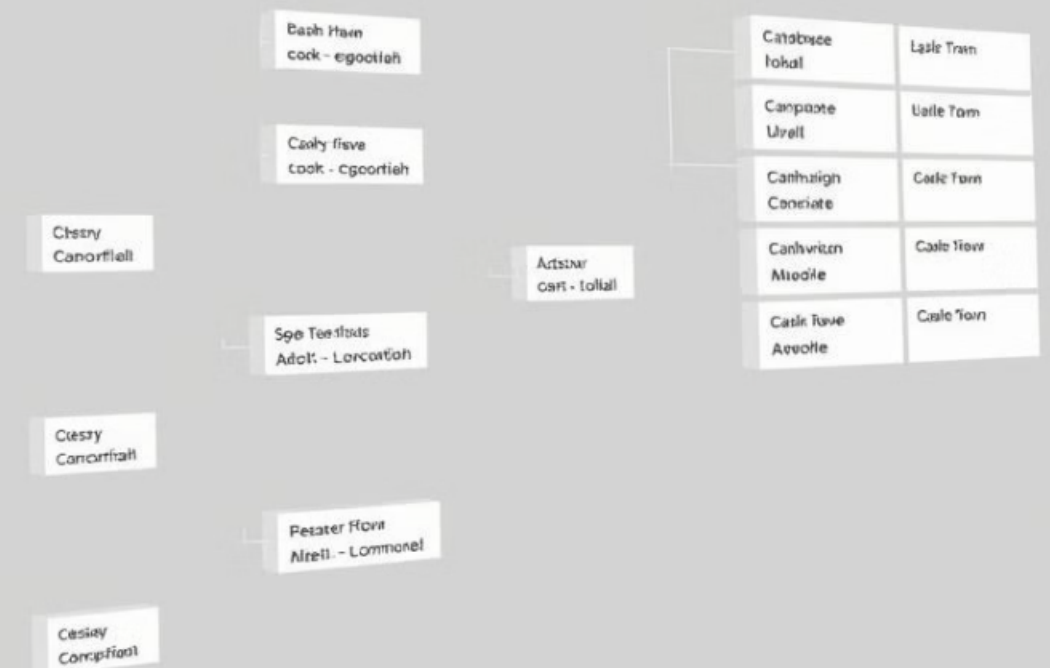


3ª Forma Normal (3NF)



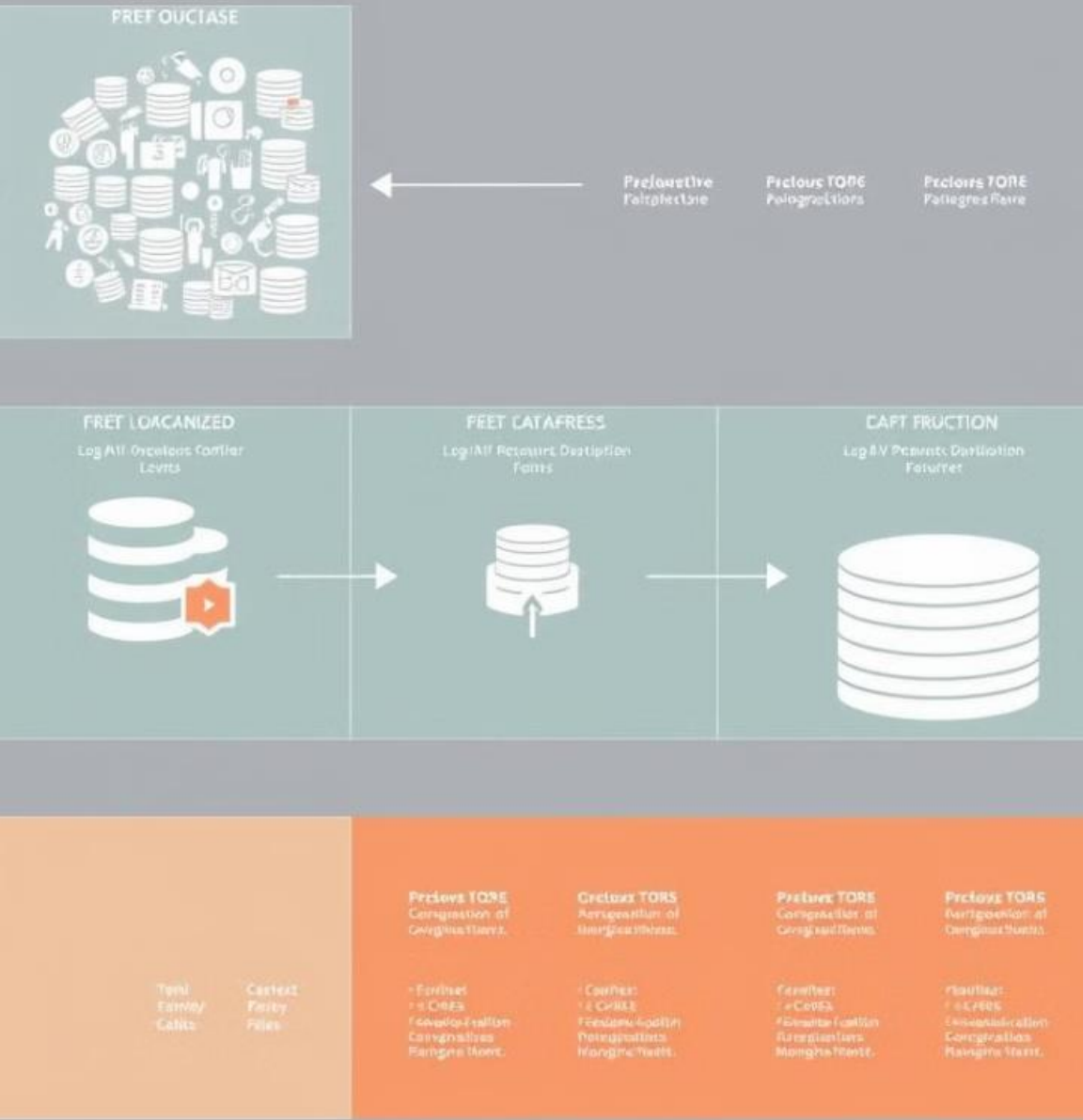
Esquema de Banco de Dados Normalizado

| | | |
|-------------|-------------|-------------|
| ensalamento | alunos | salas |
| cursos | disciplinas | professores |



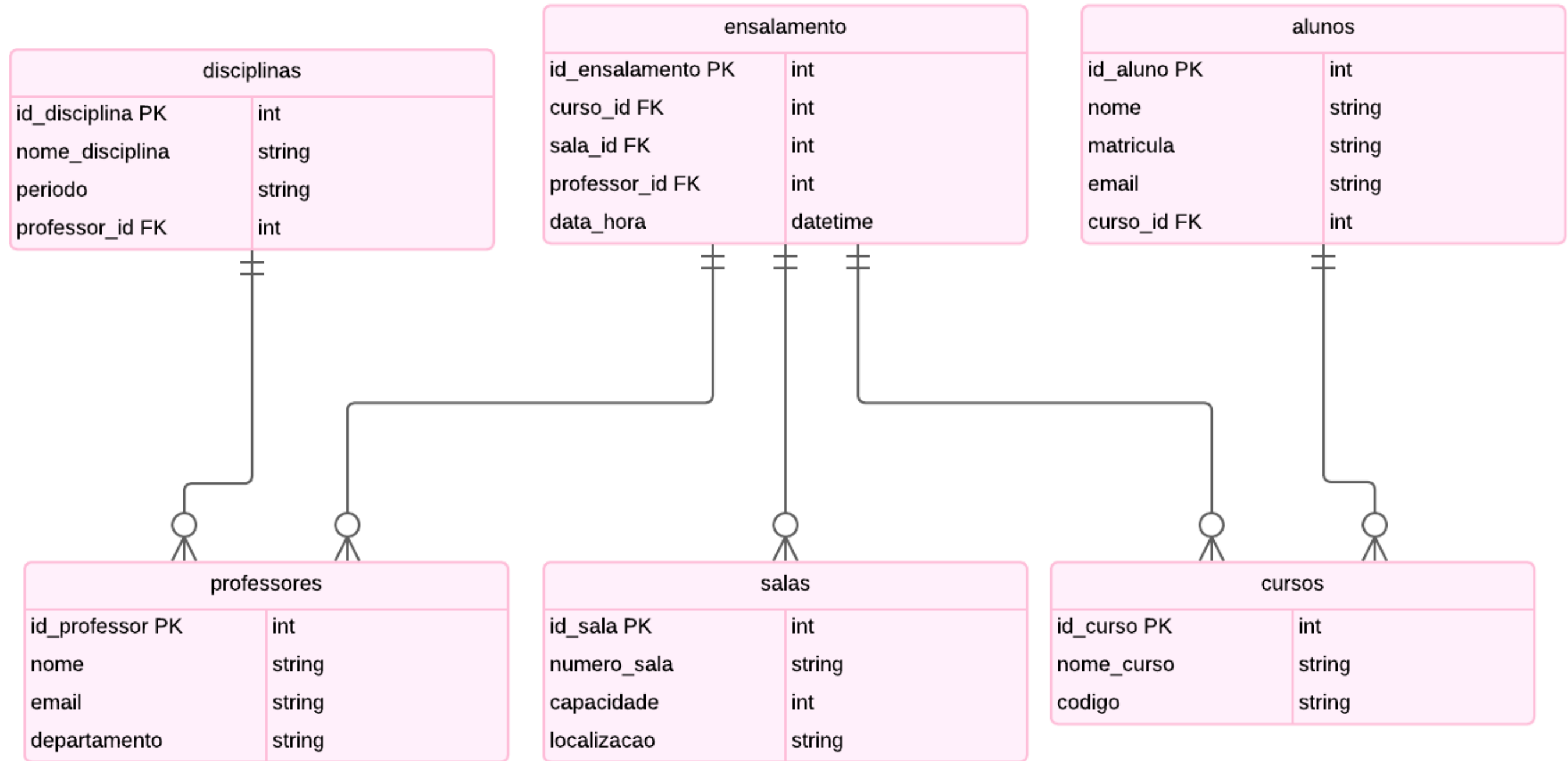
A: convattae of tiga from

Cher a the oplyit of ristey imcre manoy disb gatiés.



Conclusão

- 1 Normalização
Esquema normalizado até a 3ª Forma Normal
- 2 Benefícios
Redução de redundâncias e anomalias
- 3 Resultado
Banco de dados mais eficiente e organizado



Contato



Professor:
Diego Ramos Inácio

E-mail:
diego.inacio@univassouras.edu.br