

Projeto de Banco de Dados

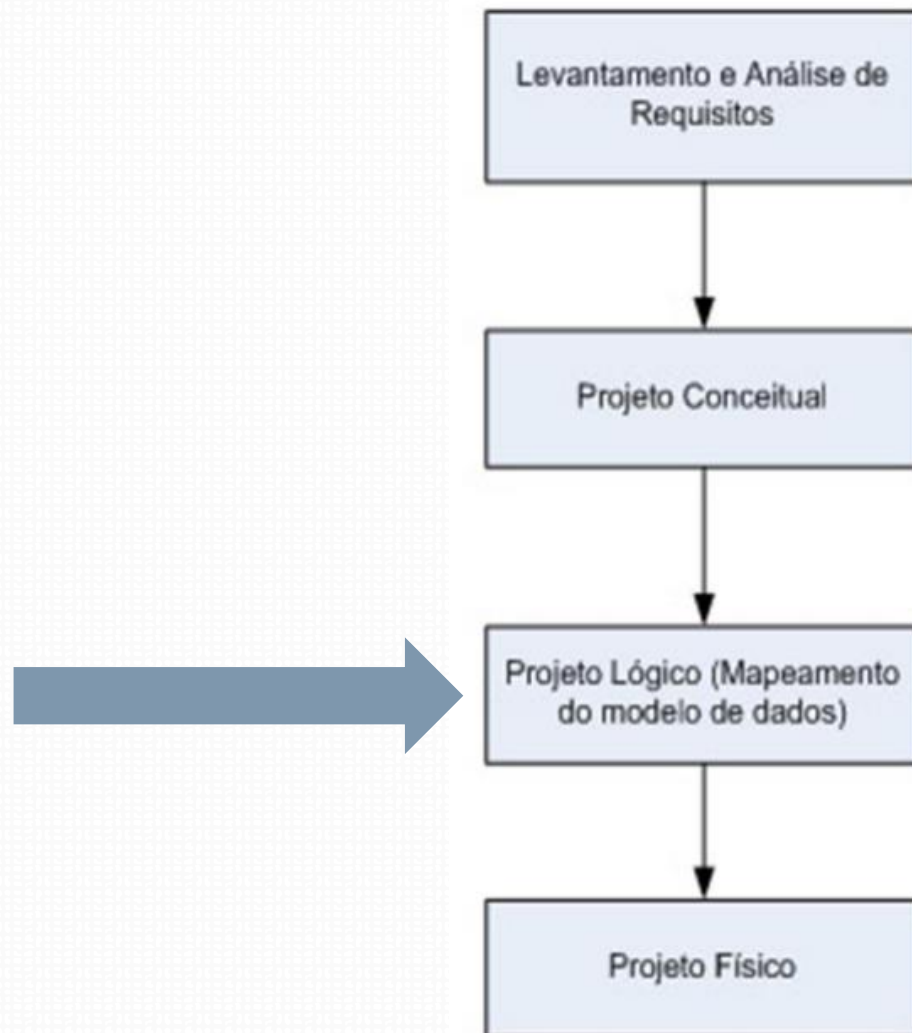
Prof. Alex Luciano Roesler Rese, MSc.

Adaptado: Prof. Lucas Debatin, MSc.

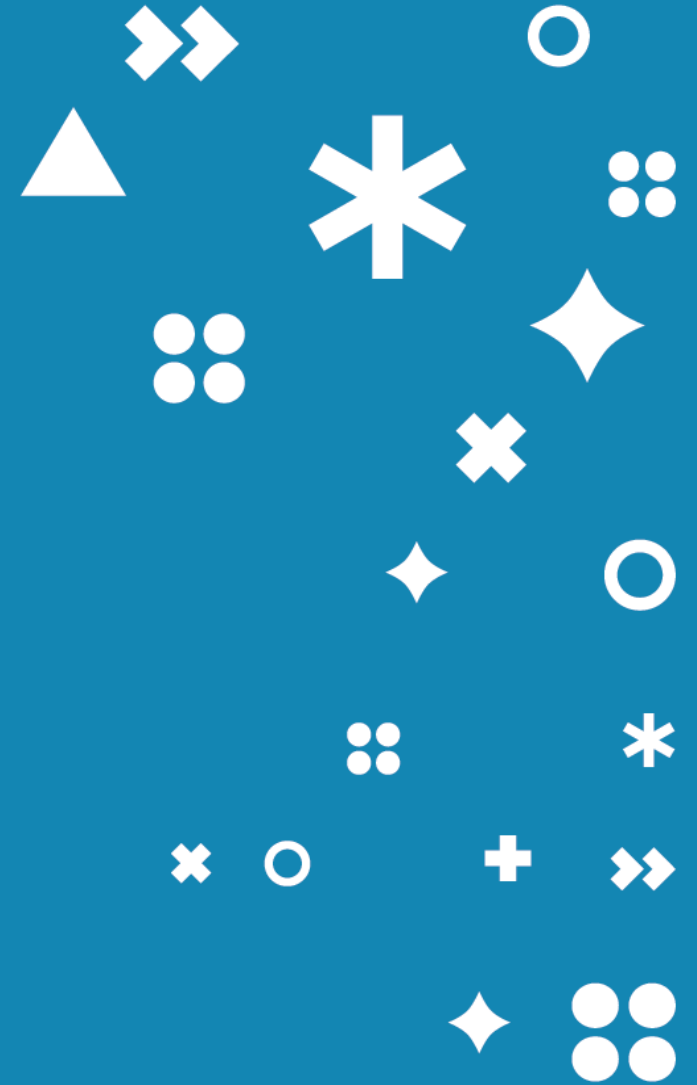
Conteúdo

- Normalização

Conteúdo



Normalização



Normalização

- Conjunto de regras aplicadas numa tabela com o objetivo de corrigir possíveis erros de projeto.
- Existem diversas regras de normalização.
- São aplicadas no Projeto Conceitual ou no Projeto Lógico!

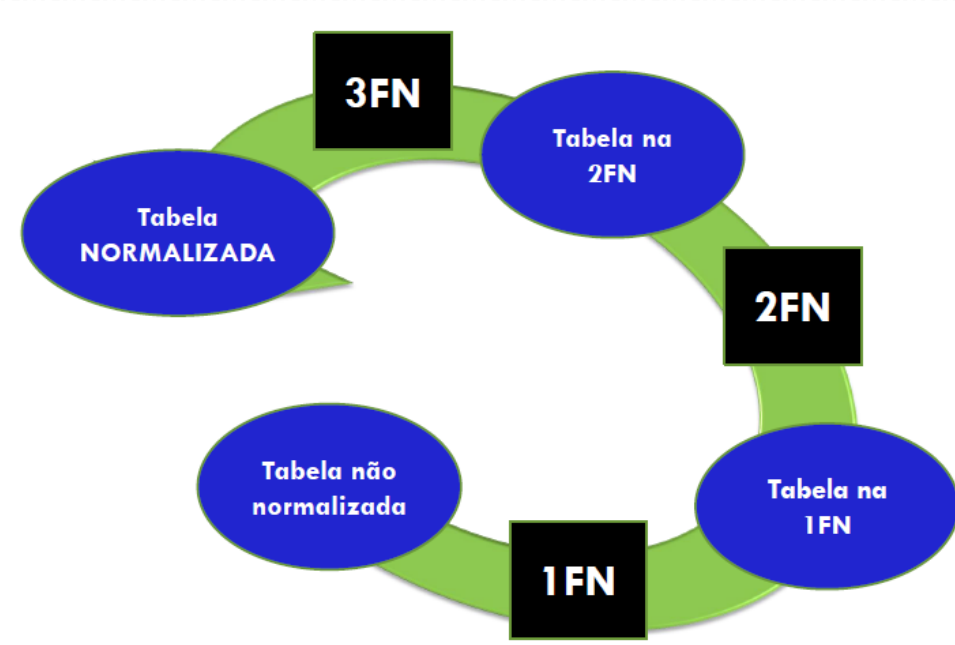
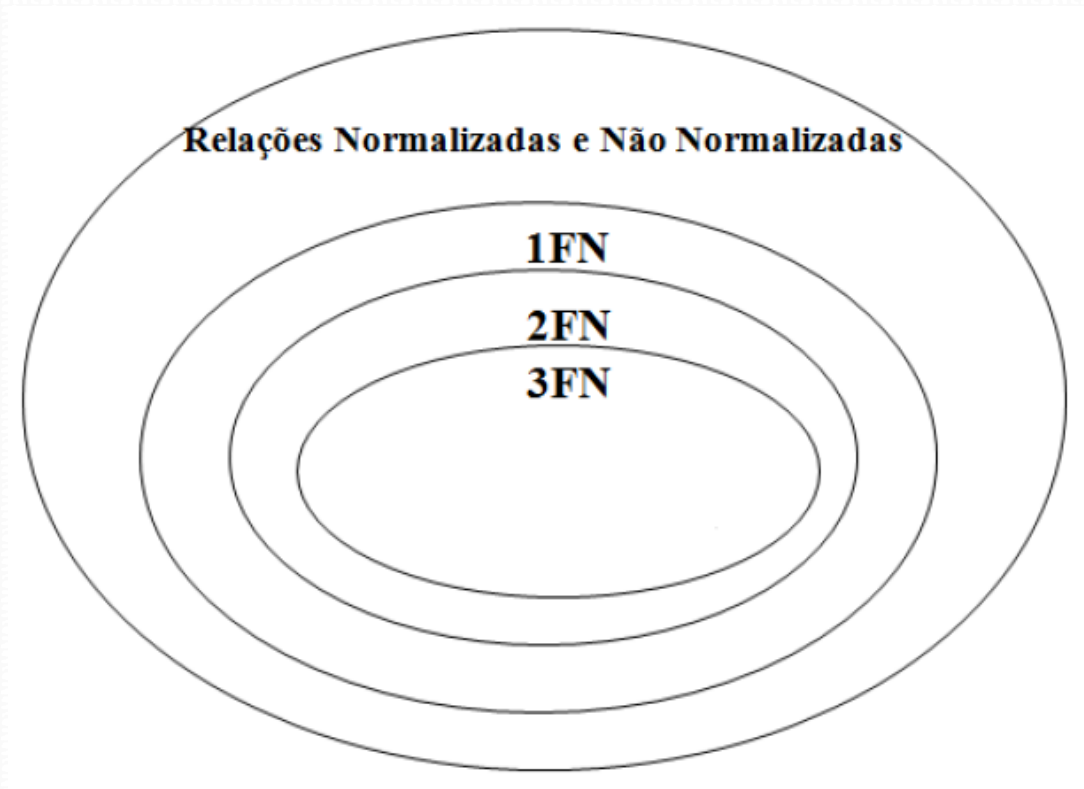
Objetivos

- Elimina redundância de dados;
- Facilita manutenção;
- Reduz o espaço de armazenamento.
- Faz com que a informação esteja sempre consistente.
 - Evita o risco de, ao gerar uma consulta, o dados virem desatualizados, pois a informação está armazenada em dois locais (um desatualizado e um atualizado).

Formas Normais

- Uma série de regras aplicadas em um BD para garantir que as tabelas possuam uma boa estrutura.
- As regras são chamadas Formas Normais:
 - 1ª Forma Normal (1 FN);
 - 2ª Forma Normal (2 FN);
 - 3ª Forma Normal (3 FN).

Formas Normais



1ª Forma Normal (1 FN)

- Dizemos que uma tabela está na primeira forma normal, quando ela não possui:
 - Atributos multivalorados;
 - Atributos compostos.
- Os atributos não podem ter valores repetidos e nem atributos possuindo mais de um valor.
- Cada atributo composto ou multivalorado deve ser transformado numa nova tabela.

1ª Forma Normal (1 FN)

Código	Nome	Endereço	Telefone
1001	Diego Machado	Rua Tal 321 Porto	5312345678 5398765432
1002	Fulano de Tal	Avenida Tal 71 Centro	5187654321 5143215678

Código	Nome	Endereço	Bairro
1001	Diego Machado	Rua Tal 321	Porto
1002	Fulano de Tal	Avenida Tal 71	Centro

Código	Telefone
1001	5312345678
1001	5398765432
1002	5112345678
1002	5187654321

Dependência Funcional (DF)

- Para avançar para a 2ª FN e para a 3ª FN é necessário conhecer o conceito de dependência funcional.
- É a principal ferramenta de avaliação para identificar se o agrupamento de atributos de uma tabela é apropriado:
 - Evitando redundância de dados;
 - Inconsistências;
 - Perda de dados em operações de remoções ou alterações.

Dependência Funcional (DF)

- Existem 3 tipos:
 - Dependência Funcional Total (1 FN);
 - Dependência Funcional Parcial (2 FN);
 - Dependência Funcional Transitiva (3 FN).

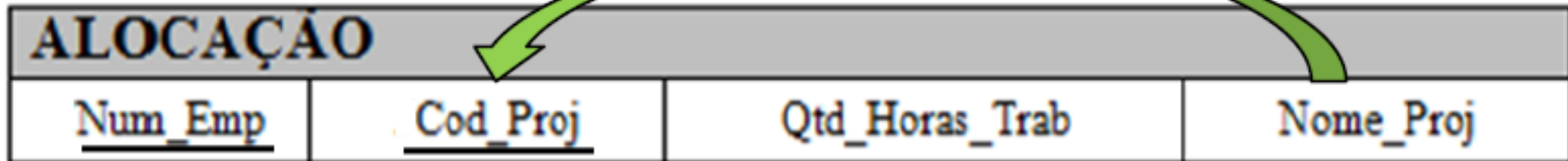
DF Total

- Os atributos não chave de uma tabela têm que depender totalmente da chave primária e somente dela.
 - Exemplo: uma determinada tabela possui sua chave primária composta pelos atributos A e B. Logo, C será dependente funcional total se e somente se C depender funcionalmente de A e B.



DF Parcial

- Os atributos não chave de uma tabela dependem de parte da chave primária.
 - Exemplo: uma determinada tabela possui sua chave primária composta pelos atributos A e B. Logo, C será dependente funcional parcial se e somente se C depender funcionalmente de A ou B.

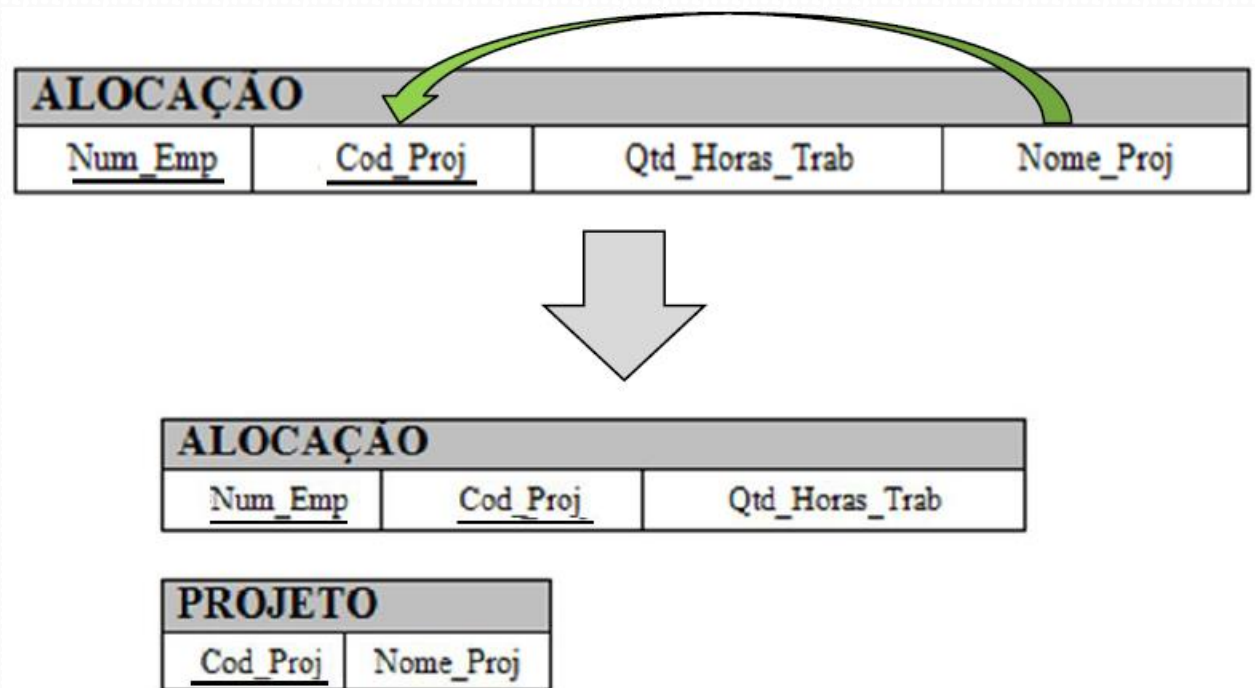


2ª Forma Normal (2 FN)

- Uma tabela está na 2 FN se ela já estiver na 1 FN.
- Todo atributo que não for chave primária deve ser dependente funcional total:
 - Não deve existir dependência funcional parcial.
 - O campo não chave deverá depender de todos os campos que compõem a chave primária.

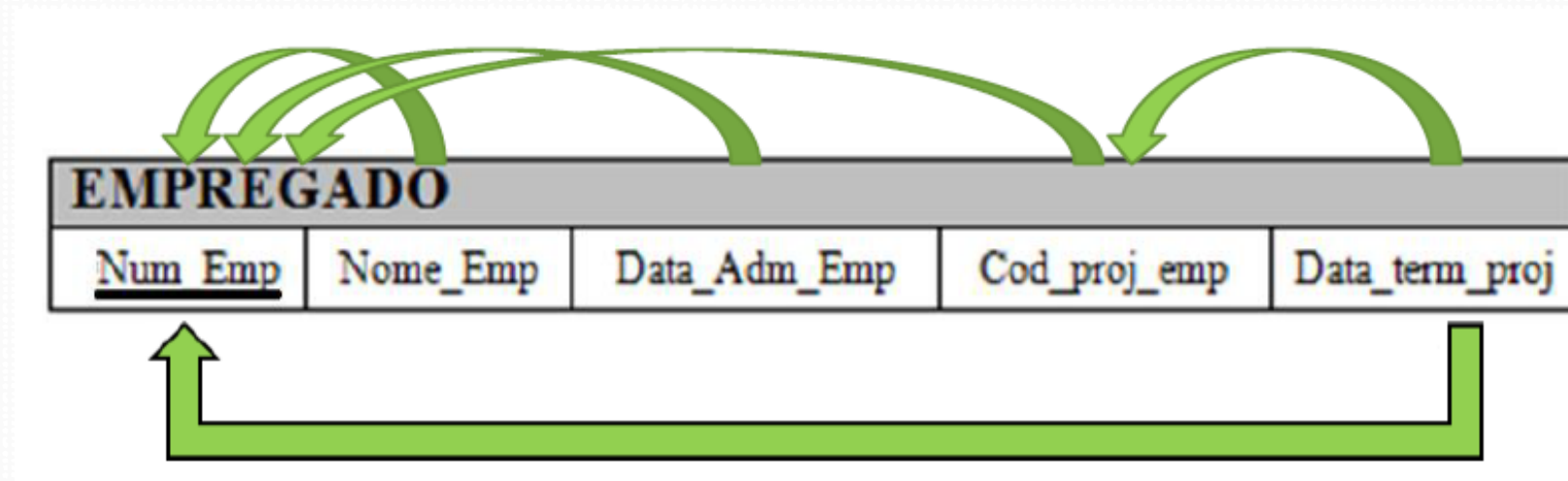
2ª Forma Normal (2 FN)

- Solução para dependência funcional parcial:



DF Transitiva

- O atributo C é dependente funcional transitivo de A se C é funcionalmente dependente de B e B funcionalmente dependente de A, na mesma tabela.

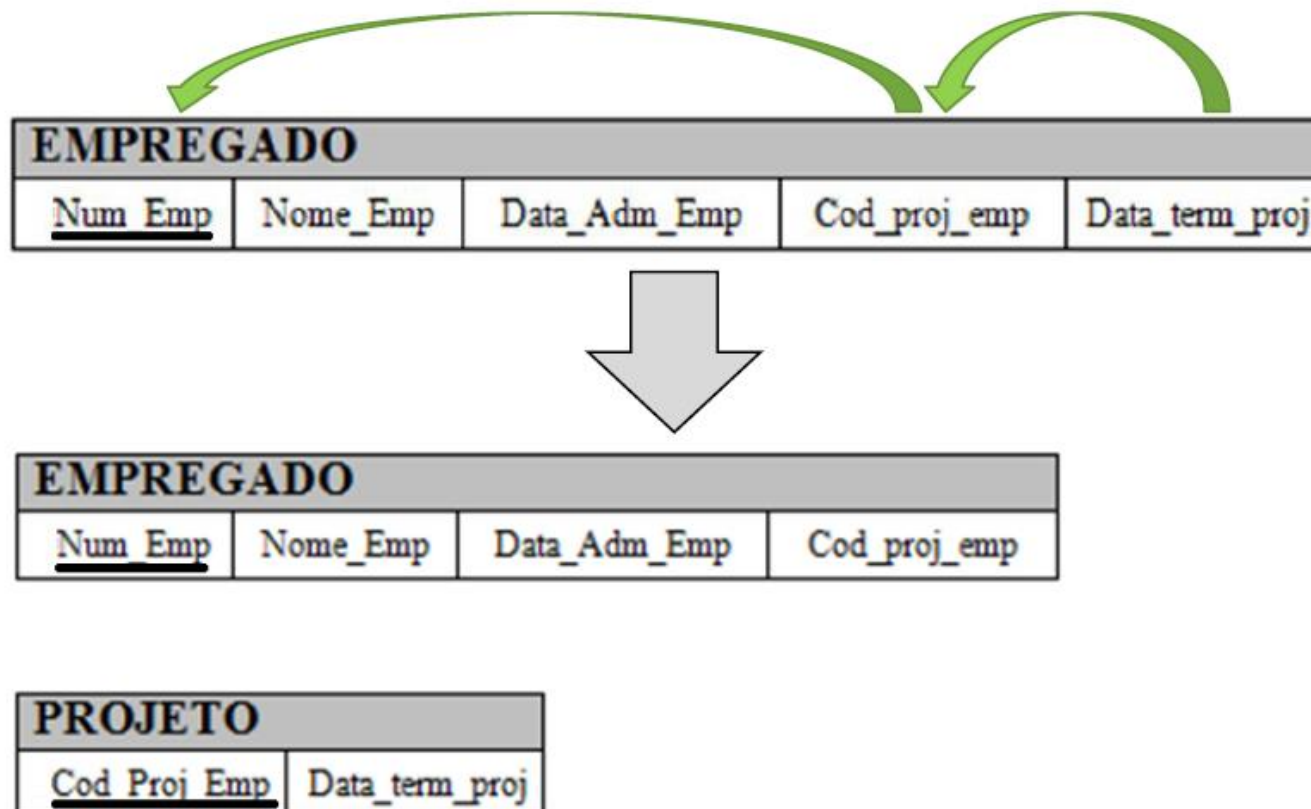


3ª Forma Normal (3 FN)

- Uma tabela está na 3 FN se ela já estiver na 2 FN.
 - Não deve existir dependência funcional transitiva entre atributos não chave.

3ª Forma Normal (3 FN)

- Solução para dependência funcional transitiva:



3ª Forma Normal (3 FN)

- A terceira forma normal se preocupa em transformar o esquema de tabela não normalizado em um esquema relacional sem tabelas aninhadas.
- Projeto_Empregado (ID, Tipo, Descricao, ID_Empregado, Nome, Setor, Salario, Data_Admissao)
- Projeto (ID, Tipo, Descricao)
- Empregado (ID, ID_Projeto, Nome, Setor, Salario, Data_Admissao)

A solid blue background featuring a variety of white geometric shapes and symbols. The elements are scattered across the frame, including: a large triangle in the upper left; two right-pointing arrows in the top center; a circle in the top right; a six-pointed star in the upper middle; a four-dot pattern in the middle right; a four-dot pattern in the middle left; a four-pointed star in the middle right; a cross in the middle right; a four-pointed star in the lower middle; a circle in the lower right; a four-dot pattern in the lower middle; a cross in the lower left; a circle in the lower left; a plus sign in the lower right; two right-pointing arrows in the bottom right; a four-pointed star in the bottom right; and a four-dot pattern in the bottom right.

Questão 1

- O projeto lógico abaixo está normalizado?
- aluno (id, nome, idade, status)
- disciplina (id, nome)
- escola (id, nome)
- curso (id, id_escola, nome)
- grade_curricular (id, id_curso, nome, status) aluno_disciplina(id_aluno, id_disciplina)
- grade_disciplina(id_grade, id_disciplina)

Questão 1 - Resolução

- O projeto lógico abaixo está normalizado?
- aluno (id, nome, idade, status)
- disciplina (id, nome)
- escola (id, nome)
- curso (id, id_escola, nome)
- grade_curricular (id, id_curso, nome, status) aluno_disciplina(id_aluno, id_disciplina)
- grade_disciplina(id_grade, id_disciplina)

Questão 2

- O projeto lógico abaixo está normalizado?
- veiculo (id, nome, modelo, marca, valor)
- cliente (id, nome, cpf, data_nascimento, endereco, telefone, status)
- vendedor (id, nome, comissao, data_admissao, endereco, telefone, status)
- venda (id, id_cliente, id_vendedor, data, valor_total, valor_desconto, forma_pagamento, status)
- item_venda (id, id_venda, id_veiculo, quantidade, chassi, cor, opcionais)

Questão 2 - Resolução

- marca (id, nome)
- estado (id, nome, uf)
- opcional (id, nome)
- cor (id, nome)
- modelo (id, id_marca, nome)
- veiculo (id, id_modelo, nome, valor)
- cidade (id, id_estado, nome)
- endereco (id, id_cidade, logradouro, numero, bairro, cep)
- cliente (id, id_endereco, nome, cpf, data_nascimento, status)
- telefone_cliente (id, id_cliente, numero, tipo)
- vendedor (id, id_endereco, nome, comissao, data_admissao, status)
- telefone_vendedor (id, id_vendedor, numero, tipo)
- venda (id, id_cliente, id_vendedor, data, valor_total, valor_desconto, forma_pagamento, status)
- item_venda (id, id_venda, id_veiculo, id_cor, quantidade, chassi)
- item_venda_opcional (id_item_venda, id_opcional)

Questão 3

- O projeto lógico abaixo está normalizado?
- livro (id, codigo_interno, titulo, resumo, volume, ano, edicao, editora, autores, genero, qtd_pagina, qtd_exemplar, isbn, idioma, disponibilidade, tipo_capa)
- usuario (id, nome, email, codigo_pessoa, senha)
- aluno (id, id_usuario, curso, matricula, status)
- comunidade (id, id_usuario, cpf, endereco, telefone)
- funcionario (id, id_usuario, cargo, setor, ramal, status)
- emprestimo(id, id_usuario, data_registro, data_devolucao, qtd_renovacao, status)
- livro_emprestimo (id_emprestimo, id_livro)
- reserva(id, id_usuario, data_registro, prioridade, status)
- livro_reserva (id_reserva, id_livro)
- multa(id, id_emprestimo, valor, data_vencimento, tipo_pagamento, status)

Questão 3 – Resolução

- autor (id, nome)
- editora (id, nome)
- genero (id, nome)
- idioma (id, nome)
- disponibilidade (id, nome)
- tipo_capa (id, nome)
- setor (id, nome)
- curso (id, nome)
- prioridade (id, nome)
- estado (id, nome, uf)
- tipo_pagamento (id, nome)
- cidade (id, id_estado, nome)
- livro (id, id_editora, id_idioma, id_disponibilidade, id_tipo_capa, codigo_interno, titulo, resumo, volume, ano, edicao, qtd_pagina, qtd_exemplar, isbn)
- autor_livro (id_livro, id_autor)
- genero_livro (id_livro, id_genero)
- usuario (id, codigo_pessoa, nome, email, senha)
- aluno (id, id_usuario, id_curso, matricula, status)
- comunidade (id, id_usuario, id_cidade, cpf, logradouro, bairro, numero, cep)
- comunidade_telefone (id, id_comunidade, telefone, tipo)
- funcionario (id, id_usuario, id_setor, ramal, status)
- emprestimo(id, id_usuario, data_registro, data_devolucao, qtd_renovacao, status)
- livro_emprestimo (id_emprestimo, id_livro)
- reserva(id, id_usuario, id_prioridade, data_registro, status)
- livro_reserva (id_reserva, id_livro)
- multa(id, id_emprestimo, id_tipo_pagamento, data_vencimento, valor, status)