

Biblioteca ULBRA - Projeto de Banco de Dados

1. Nome do projeto e descrição:

Nome: Biblioteca-ULBRA -Sistema de Gestão de Biblioteca

Descrição:

Sistema para gestão do acervo físico de uma biblioteca universitária. Permite cadastrar títulos, controlar exemplares físicos, registrar empréstimos e devoluções, gerenciar reservas e gerar relatórios gerenciais.

Objetivo:

Substituir controles manuais por um repositório central, controlar a circulação de exemplares, disponibilizar consulta rápida ao catálogo e fornecer relatórios para orientar aquisições, manutenção e cobrança de pendências.

2. Propósito do banco de dados

O banco de dados armazena o catálogo de títulos, autores, editoras, categorias, exemplares físicos, cadastro de membros e funcionários, histórico de empréstimos, multas e reservas. Deve ser otimizado para consultas de busca no catálogo e para operações transacionais típicas de circulação (empréstimo e devolução).

3. Definição das collections:

autores: id, nome.

editoras: id, nome.

categorias: id, nome.

livros: id, titulo, isbn, ano, editora_id, categoria_id, autores (lista), resumo, criado_em.

exemplares: id, livro_id, codigo_de_barras, data_aquisicao, condicao, status.

membros: id, nome, email, telefone, data_registro, ativo, tipo_membro.

funcionarios: id, nome, email, cargo.

emprestimos: id, exemplar_id, membro_id, funcionario_saida_id, funcionario_entrada_id, datas, multa, status.

reservas: id, livro_id, membro_id, data_reserva, status.

4. Relacionamentos

Livros relacionados a autores (lista interna), editoras e categorias por id. Exemplares relacionados a livros. Empréstimos relacionados a exemplares, membros e funcionários. Reservas relacionadas a livros e membros.

5. Exemplos conceituais:

Autor: id 1, Machado de Assis.

Livro: possui título, isbn, ano, relação com editora e autores.

Exemplar: possui código de barras, data de aquisição e status.

Membro: dados pessoais e tipo.

Empréstimo: registro contendo exemplar, membro e datas.

Reserva: vínculo entre membro e livro.

6. Modelagem e estruturação:

Regras:

1. Empréstimo só ocorre se o exemplar estiver disponível;
2. Devolução atualiza status e calcula multa se houver atraso;
3. Reservas seguem ordem cronológica.

Índices recomendados em isbn, código de barras e campos de pesquisa.

7. Relatórios:

- 1) Livros mais emprestados.
- 2) Empréstimos por mês.
- 3) Membros com pendências.
- 4) Disponibilidade por categoria.
- 5) Reservas ativas por livro.
- 6) Tempo médio de empréstimo por tipo de membro.
- 7) Exemplares em condição ruim.
- 8) Autores mais emprestados.

8. Comandos de criação do banco:

Seleciona ou cria o banco de dados:
use biblioteca_ulbra

Cria a collection de autores:
db.createCollection("autores")

Cria a collection de editoras:
db.createCollection("editoras")

Cria a collection de categorias:
db.createCollection("categorias")

Cria a collection de livros:
db.createCollection("livros")

Cria a collection de exemplares:
db.createCollection("exemplares")

Cria a collection de membros:
db.createCollection("membros")

Cria a collection de funcionários:
db.createCollection("funcionarios")

Cria a collection de empréstimos:
db.createCollection("emprestimos")

Cria a collection de reservas:
db.createCollection("reservas")

Cria índice único para ISBN:
db.livros.createIndex({ isbn: 1 }, { unique: true })

Cria índice de texto para título e resumo:
db.livros.createIndex({ titulo: "text", resumo: "text" })

Cria índice único para código de barras:
db.exemplares.createIndex({ codigo_de_barras: 1 }, { unique: true })

Cria índice para exemplares por livro:
db.exemplares.createIndex({ livro_id: 1 })

Cria índice para empréstimos por membro:
db.emprestimos.createIndex({ membro_id: 1 })

Cria índice para empréstimos por exemplar:
db.emprestimos.createIndex({ exemplar_id: 1 })

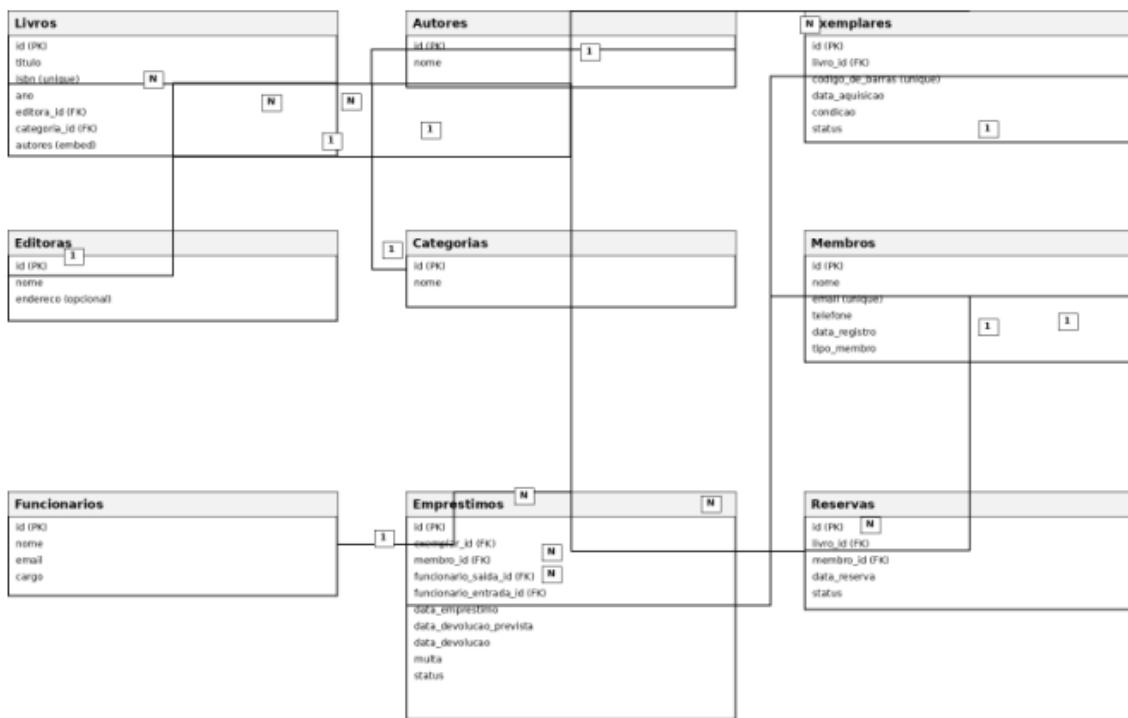
Cria índice por data de empréstimo:
db.emprestimos.createIndex({ data_emprestimo: 1 })

Cria índice para reservas por livro:
db.reservas.createIndex({ livro_id: 1 })

Cria índice para reservas por membro:
db.reservas.createIndex({ membro_id: 1 })

Cria índice único para e-mail de membros:
db.membros.createIndex({ email: 1 }, { unique: true })

9. Modelo Conceitual:



Relacionamento	Cardinalidade
Livros – Autores	N : N
Livros – Editoras	1 : N
Livros – Categorias	1 : N
Livros – Exemplares	1 : N
Exemplares – Empréstimos	1 : N
Membros – Empréstimos	1 : N
Funcionários – Empréstimos (saída)	1 : N
Funcionários – Empréstimos (entrada)	1 : N
Livros – Reservas	1 : N
Membros – Reservas	1 : N

9. Modelo Lógico:

1. autores

- id (PK) -inteiro
- nome -string
- biografia -string

- nacionalidade -string
- data_nascimento -date

2. **editoras**

- id (PK)
- nome -string
- endereco -string
- contato -string

3. **categorias**

- id (PK)
- nome -string

4. **livros**

- id (PK)
- titulo -string
- isbn -string
- ano -inteiro
- editora_id (FK) -referência para editoras.id
- categoria_id (FK) -referência para categorias.id
- autores -array embedado contendo objetos com id_autor e
- resumo -string
- criado_em -date
- ativo -boolean

5. **exemplares**

- id (PK)
- livro_id (FK) -referência para livros.id
- codigo_de_barras -string
- data_aquisicao -date
- condicao -string
- status -string

6. membros

- id (PK)
- nome -string
- email -string
- telefone -string
- data_registro -date
- ativo -boolean
- tipo_membro -string

7. funcionarios

- id (PK)
- nome -string
- email -string
- cargo -string

8. emprestimos

- id (PK)
- exemplar_id (FK) -referência para exemplares.id

- membro_id (FK) -referência para membros.id
- funcionario_saida_id (FK) -referência para funcionarios.id
- funcionario_entrada_id (FK) - referência para funcionarios.id
- data_emprestimo - date
- data_devolucao_prevista - date
- data_devolucao - date
- multa - number
- status - string

9. reservas

- id (PK)
- livro_id (FK) - referência para livros.id
- membro_id (FK) - referência para membros.id
- data_reserva - date
- status - string (ativa / cumprida / cancelada / expirada)