



**INSTITUTO  
FEDERAL**

Paraíba

**ANÁLISE E DESENVOLVIMENTO DE SISTEMAS**

DIOGO SANTANA FREITAS

**GESTÃO E CONTROLE DA BIBLIOTECA MUNICIPAL**

CAJAZEIRAS - PB

2023

DIOGO SANTANA FREITAS

## **GESTÃO E CONTROLE DA BIBLIOTECA MUNICIPAL**

Trabalho apresentado ao Curso Superior Tecnólogo de Análise e Desenvolvimento de Sistemas do IFPB – Instituto Federal de Educação, Ciência e Tecnologia da Paraíba Campus Cajazeiras, para a disciplina de Banco de Dados 1, ministrado pelo Prof. Dr. Fabio Gomes de Andrade.

CAJAZEIRAS - PB

2023

# Sumário

<b>1. INTRODUÇÃO.....</b>	<b>4</b>
<b>2 MODELO CONCEITUAL.....</b>	<b>5</b>
2.1 Levantamento de Requisitos.....	5
2.2 Diagrama Entidade-Relacionamento.....	7
2.3 Dicionário Conceitual de Dados.....	8
Entidades:.....	8
Relacionamentos:.....	11
2.4 Consultas.....	13
<b>3. MODELO LÓGICO.....</b>	<b>14</b>
3.1 Mapeamento Entidade-Relacionamento.....	14
3.2 Dicionário Lógico de Dados.....	15
<b>4. MODELO FÍSICO.....</b>	<b>28</b>
4.1 Scripts SQL para criação das tabelas e povoamento.....	28
4.2 Scripts de criação de índices.....	34
4.3 Scripts de criação de visões.....	34
4.4 Povoamento.....	35
4.5 Scripts das consultas.....	42

## **1. INTRODUÇÃO**

O uso de base de dados nos dias atuais se tornou indispensável para a gestão e organização de instituições, principalmente devido à necessidade de armazenar dados importantes com segurança e organização, dispensando recursos analógicos que antes eram utilizados para essa gestão, e que não possuem uma segurança muito eficaz em seu armazenamento. Dessa forma, a Biblioteca Municipal de Uiraúna com a carência de um sistema de Banco de dados que armazena os dados dos usuários e abola fichas de papéis que recorrentemente são perdidas pelos usuários que as recebem, necessita tornar possível analisar os atrasos decorrentes e transformar a gestão da biblioteca para que os livros sejam organizados de forma mais precisa e que o bibliotecário saiba da existência de livros a partir de uma consulta ao banco, necessita dessa aplicação para facilitar o dia-a-dia dos funcionários.

## 2 MODELO CONCEITUAL

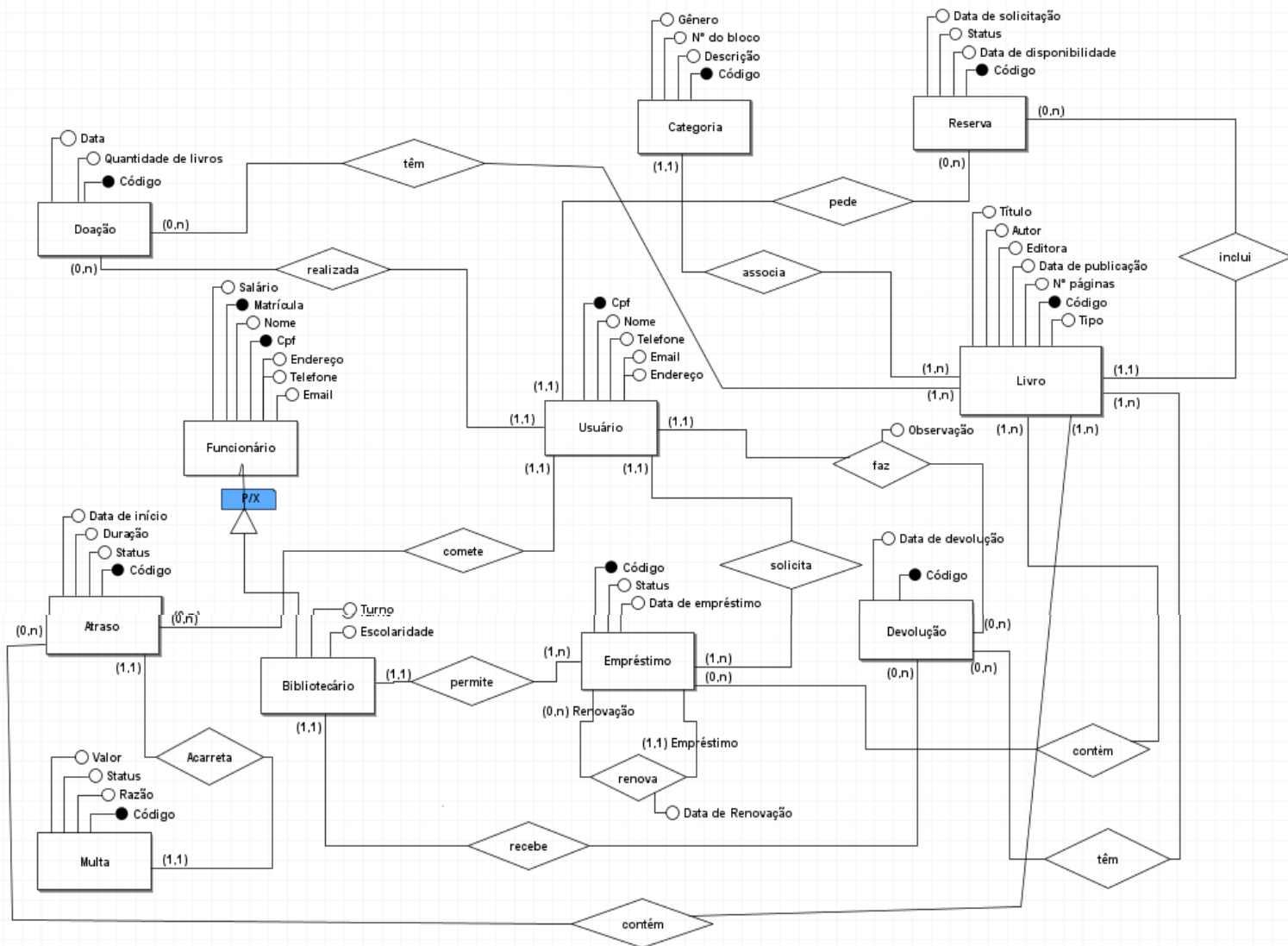
### 2.1 Levantamento de Requisitos

- A biblioteca possui alguns funcionários encarregados diariamente para atender aos usuários e alguns para manter a limpeza do ambiente. Para cada funcionário é necessário informações como nome, CPF, salário, endereço, email, telefone e matrícula. Os bibliotecários devem possuir informações específicas sobre o seu turno e a escolaridade. Além disso, o bibliotecário será responsável exclusivamente por permitir os empréstimos, e cada empréstimo precisa conter a data em que foi realizado, a quantidade de livros, o código identificador, status e os livros emprestados.
- Novos livros que aparecem na biblioteca devem ser cadastrados, para cada livro deve possuir dados como título, autor, editora, data de publicação, número de páginas e código identificador. Cada livro deve estar associado a uma categoria, e nessa categoria deve conter o gênero, número do bloco em que está localizada dentro da biblioteca e descrição.
- Os usuários que desejarem solicitar um empréstimo é preciso se cadastrar no sistema, e esse sistema irá armazenar o nome, CPF, endereço, email e telefone. Cada usuário pode em cada empréstimo levar no máximo 3 livros. Caso o prazo de entrega não seja suficiente para o uso dos livros, um empréstimo feito pode se renovar caso o usuário solicite.
- É ocorrente o recebimento de doações dentro da biblioteca, e para que fique registrado essas doações no sistema, o bibliotecário deve armazenar a data da doação do livro e a quantidade de livros. Ademais, o nome, telefone, email, cpf e o endereço do usuário que realizou a doação naquele dia.
- No sistema também é possível que quando chegar a data da devolução do livro/livros emprestados ao usuário, caso o usuário devolva até a data limite será armazenado a data da devolução, o bibliotecário que recebeu, livros emprestados e observações como a falta de algum item do livro. Caso contrário, o sistema deve adicionar um registro de atraso contendo informações como a

data de início do atraso, duração, usuário que atrasou a entrega e status. Para caso de atraso será acarretado uma multa que deve ser armazenada com o valor, a razão e o status.

- Para situações em que o usuário deseja um livro que já foi emprestado, será possível que o usuário peça uma reserva para aquele livro. Na reserva deve ter o usuário que solicitou, a data de solicitação, o status, a data em que o livro estará disponível e o livro solicitado

## 2.2 Diagrama Entidade-Relacionamento



## 2.3 Dicionário Conceitual de Dados

### Entidades:

#### Entidade Funcionário:

- Generalização da entidade Bibliotecário, armazena informações sobre os funcionários.

#### Atributo(s):

- Nome: armazena o nome completo de uma pessoa;
- CPF: armazena o CPF de uma pessoa;
- Email: armazena um ou mais emails de uma pessoa;
- Telefone: armazena um ou mais telefones de uma pessoa;
- Salário: armazena o valor do salário de um funcionário;
- Matrícula: armazena o número da matrícula de um funcionário;
- Endereço: armazena o endereço de um funcionário;

#### Entidade Bibliotecário

- Especialização da entidade Funcionário e é usada para relacionamentos específicos com outras entidades;

#### Atributo(s):

- Turno: armazena o turno de trabalho do Bibliotecário;
- Escolaridade: armazena o grau de escolaridade;

#### Entidade Usuário

- Entidade que armazena as informações de um usuário cadastrado

#### Atributo(s):

- Nome: armazena o nome completo de uma pessoa;
- CPF: armazena o CPF de uma pessoa;
- Email: armazena um ou mais emails de uma pessoa;
- Telefone: armazena um ou mais telefones de uma pessoa;
- Endereço: armazena informações sobre o endereço;



### **Entidade Livro:**

- Entidade usada para armazenar informações específicas sobre os livros;

#### **Atributo(s):**

- Título: armazena o título do livro;
- Autor: armazena informações sobre o autor do livro;
- Editora: armazena informações da editora do livro;
- Data de Publicação: armazena a data em que o livro foi publicado;
- Número de páginas: armazena a quantidade de página do livro;
- Código: armazena um número que identifica exclusivamente o livro;
- Tipo: armazena o tipo do livro, literário ou didático;

### **Entidade Empréstimo:**

- Entidade usada para armazenar informações sobre os empréstimos solicitados e concedidos na biblioteca.

#### **Atributo(s):**

- Data de realização: armazena a data em que o empréstimo foi realizado;
- Código: armazena o número identificador do empréstimo;
- Status: armazena o estado do empréstimo (rejeitado, concebido, pendente);

### **Entidade Devolução:**

- Entidade utilizada para registrar devoluções feitas por usuários.

#### **Atributo(s):**

- Data de devolução: armazena a data em que foi realizada a devolução;
- Observações: armazena uma descrição de observações feitas durante o registro da devolução;
- Código: armazena o número identificador da devolução;

### **Entidade Atraso:**

- Entidade utilizada para armazenar atrasos que ocorreram após o vencimento do prazo da devolução

**Atributo(s):**

- Data de início: armazena a data de início do atraso;
- Duração: armazena a quantidade de dias que durou o atraso;
- Status: armazena o estado atual daquele atraso;
- Código: armazena o número identificador do atraso;

**Entidade Multa:**

- Entidade que armazena recorrentes multas causadas por atraso ou infração com os livros.

**Atributo(s):**

- Valor: armazena o valor acumulado da multa;
- Razão: armazena o motivo pelo qual a multa foi aplicada;
- Status: armazena o estado da multa (rejeitado, concebido, pendente);
- Código: armazena o número identificador da multa;

**Entidade Reserva:**

- Entidade utilizada para armazenar uma reserva desejada a um livro.

**Atributo(s):**

- Data de solicitação: armazena a data em que foi solicitado o livro;
- Status: armazena o estado atual da reserva solicitada;
- Data de disponibilidade: armazena a data em que o livro solicitado estará disponível;
- Código: armazena o número identificador da reserva

**Entidade Doação:**

- Entidade registra doação feita por uma pessoa

**Atributo(s):**

- Data da doação: armazena a data em que foi realizada a doação;
- Quantidade de Livros: armazena a quantidade de livros doados;

- Código: armazena o número identificador da doação;

## **Relacionamentos:**

- permite: relacionamento entre a entidade Bibliotecário e a entidade Empréstimo. Um bibliotecário pode permitir um ou vários empréstimos, já um empréstimo só pode ser permitido por um bibliotecário.
- solicita: relacionamento entidades Usuário e a Entidade Empréstimo. Um usuário pode solicitar no mínimo um empréstimo e no máximo vários, e um empréstimo só pode ser solicitado por um usuário.
- contém: relacionamento entre as entidades Empréstimo e Livro. Um empréstimo pode conter no mínimo um livro e no máximo 3, e um livro pode não estar contido em um empréstimo ou em vários empréstimos.
- renova: auto-relacionamento entre a entidade Empréstimo. Um empréstimo pode ser renovado zero ou várias vezes, um empréstimo só pode ser renovação de outro empréstimo.
- associa: relacionamento entre as entidades Categoria e Livro. Uma categoria pode associar um ou vários livros, porém um livro só pode estar associado a uma categoria.
- realizada: relacionamento entre as entidades Usuário e Doação. Uma pessoa pode realizar zero ou várias doações, mas uma doação só pode ser feita por uma pessoa.
- comete: relacionamento entre entidades Usuário e Atraso. Um usuário pode cometer zero ou vários atrasos, e um atraso pode ser cometido por um usuário.
- acarreta: relacionamento entre as entidades Atraso e Multa. Um atraso pode acarretar uma multa, e uma multa só pode ser acarretada por um atraso.
- recebe: relacionamento entre as entidades Bibliotecário e Devolução. Um bibliotecário pode receber zero ou várias devoluções, e uma devolução só pode ser recebida por um bibliotecário.
- têm: relacionamento entre as entidades Devolução e Livro. Uma devolução pode ter um a três livros, e um livro pode estar em zero ou várias devoluções.

- faz: relacionamento entre as entidades Usuário e Devolução. Um usuário pode fazer zero ou várias devoluções, e uma devolução só pode ser feita por um usuário.
- pede: relacionamento entre as entidades Usuário e Reserva. Um usuário pode fazer zero ou várias reservas, e uma reserva pode ser feita por um usuário.
- Inclui: relacionamento entre as entidades Reserva e Livro. Uma reserva pode incluir um livro, e um livro pode estar em zero ou várias reservas.
- têm: relacionamento entre as entidades Doação e Livro. Uma doação pode conter um ou vários livros e um livro pode estar em zero ou várias doações
- contém: relacionamento entre as entidades Atraso e Livro. Um atraso pode estar relacionado a um ou vários livros, e os livros podem estar em zero ou vários atrasos.

## 2.4 Consultas

- Obter todas as informações de um livro, incluindo o gênero da categoria e o número do bloco em que ela está associada
- Obter a quantidade total de livros emprestados por categoria:
- Listar o título, o código e o autor de todos os livros que não possuem editoras, mas possuem uma data de publicação e ordenar por ordem decrescente do autor
- Listar todas as informações das devoluções em que o status é de concluída e que possuem alguma observação
- Listar livros em que o sobrenome do autor possua uma determinada sequência de caracteres, em ordem crescente do autor.
- Obter bibliotecários em que o número do telefone que iniciam com uma determinada sequência de dígitos
- Obter o nome e o cpf dos usuários que possuem e pelo menos dois livros emprestados e ordenar por ordem decrescente do nome
- Obter todas as informações de usuários com multas pendentes
- Obter todas as devoluções recebidas por um determinado bibliotecário
- Obter todas as doações feitas por usuários que já realizaram um empréstimo:
- Obter todos os usuários que realizaram pelo menos uma doação e não possuem nenhum registro de atraso
- Obter os livros que foram emprestados, mas não foram reservados
- Obter a média salarial de todos os funcionários que não são bibliotecários
- Obter a soma da quantidade de todos os livros doados recebidos na biblioteca
- Obter o código, o gênero e a quantidade de livros que estão associados a cada categoria
- Obter o nome, cpf e a quantidade de empréstimos realizados pelos usuários que possuem status de pendente em que a quantidade é superior a 1

### 3. MODELO LÓGICO

#### 3.1 Mapeamento Entidade-Relacionamento

Usuário (CPF, nome, email, telefone, endereço)

Funcionário (Matrícula, CPF, nome, email, telefone, salário, endereço)

Bibliotecário (Matrícula, turno, escolaridade)

Livro (Código, título, Autor, Editora, data\_publicação, n°\_páginas, tipo, *CodCategoria*)

Categoria (Código, gênero, N°Bloco, descrição)

Empréstimo (Código, status, data\_empréstimo, *CpfUsuário*, *MatBibliotecário*)

Devolução (Código, status, data\_devolução, observação, *CpfUsuario*, *MatBibliotecario*)

Reserva (Código, data\_reserva, status, data\_disponibilidade, *CpfUsuario*, *codLivro*)

Atraso (Código, duração, status, data\_inicio, *CpfUsuario*, *CodMulta*)

Multa (Código, status, razão, valor)

Doação (Código, Data, *quant\_livros*, *CpfUsuário*)

Devolução\_Livro (*CodDevolução*, *CodLivro*)

Empréstimo\_Livro (*CodEmprestimo*, *CodLivro*)

Renovações (*codEmprestimo*, *codRenovação*, data\_Renovação)

Atraso\_Livro(*codAtraso*, *codLivro*)

Doação\_Livro (*CodDoação*. *CodLivro*)

### 3.2 Dicionário Lógico de Dados

**Usuário:** relação que armazena dados relacionados aos usuários cadastrados

Atributo	Descrição	Tipo	Domínio	Restrição
Nome	Representa o nome de um usuário cadastrado	VARCHAR(100)	VARCHAR(100)	Não nulo
CPF	Representa o cpf de um usuário cadastrado	CHARACTER(11)	CHARACTER(11)	Chave primária
Email	Representa um email de um usuário cadastrado	VARCHAR(60)	VARCHAR(60)	Não nulo
Telefone	Representa o número de telefone de um usuário cadastrado	VARCHAR(15)	VARCHAR(15)	Não nulo
Endereço	Representa o endereço de um usuário cadastrado	VARCHAR(100)	VARCHAR(100)	Não nulo

**Funcionário:** relação que armazena dados relacionados aos funcionários cadastrados

Atributo	Descrição	Tipo	Domínio	Restrição
Matrícula	Representa a matrícula de um funcionário cadastrado	VARCHAR(20)	VARCHAR(20)	Chave primária
Nome	Representa o nome de um funcionário cadastrado	VARCHAR(100)	VARCHAR(100)	Não nulo
CPF	Representa o cpf de um funcionário cadastrado	CHARACTER(11)	CHARACTER(11)	chave
Email	Representa um email de um funcionário cadastrado	VARCHAR(60)	VARCHAR(60)	Não nulo
Telefone	Representa o número de telefone de um funcionário cadastrado	VARCHAR(15)	VARCHAR(15)	Não nulo
Endereço	Representa o endereço de um funcionário cadastrado	VARCHAR(100)	VARCHAR(100)	Não nulo
Salário	Representa o salário de um funcionário cadastrado	NUMERIC(10, 2)	Números reais > 1320	Não nulo



**Bibliotecário:** relação que armazena informações específicas dos bibliotecários.

Atributo	Descrição	Tipo	Domínio	Restrição
Matrícula	Representa a matrícula de um bibliotecário cadastrado	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia a tabela Funcionário
Turno	Representa o turno de trabalho de um bibliotecário cadastrado	VARCHAR(30)	VARCHAR(30)	Não nulo
Escolaridade	Representa o grau de escolaridade de um bibliotecário	VARCHAR(50)	VARCHAR(50)	Não nulo

**Livro:** relação que armazena informações específicas dos livros

Atributo	Descrição	Tipo	Domínio	Restrição
----------	-----------	------	---------	-----------

Código	Representa a matrícula de um funcionário cadastrado	VARCHAR(20)	VARCHAR(20)	Chave primária
Título	Representa o título de um livro cadastrado	VARCHAR(100)	VARCHAR(100)	Não nulo
Autor	Representa o nome do autor de um livro	VARCHAR(100)	VARCHAR(100)	Não nulo
Editora	Representa o nome da editora do livro	VARCHAR(80)	VARCHAR(80)	
Data_publicação	Representa a data em que o livro foi publicado	DATE	DATE	
Tipo	Representa o tipo do livro (literário ou didático)	VARCHAR(50)	VARCHAR(50)	Não nulo
Número de páginas	Representa a quantidade de páginas do livro	INT	INT	Não nulo
Cod_Categoria	Representa o código da categoria do livro em que o livro está presente	VARCHAR(20)	VARCHAR(20)	Chave estrangeira que referencia código na relação Categoria

**Categoria:** relação que armazena especificas dos bibliotecários

Atributo	Descrição	Tipo	Domínio	Restrição
Código	Representa o código de uma categoria	VARCHAR(20)	VARCHAR(20)	Chave primária
Gênero	Representa o gênero em que a categoria está associada	VARCHAR(100)	VARCHAR(100)	Não nulo
N°_bloco	Representa o número do bloco em que uma categoria está situada	INT	INT	Não nulo
Descrição	Representa uma descrição sobre uma categoria	VARCHAR(300)	VARCHAR(300)	

**Empréstimo:** relação que armazena os empréstimos realizados.

Atributo	Descrição	Tipo	Domínio	Restrição
Código	Representa o código de um empréstimo realizado	VARCHAR(20)	VARCHAR(20)	Chave primária
Status	Representa o status de um empréstimo solicitado	VARCHAR(20)	VARCHAR(20)	Não nulo
Data_Empréstimo	Representa a data em que o empréstimo foi realizado	DATE	DATE	Não nulo

CpfUsuário	Representa o cpf do usuário que solicitou o empréstimo	CHARACTER(11)	CHARACTER(11)	Não nulo Chave estrangeira que referencia cpf na relação Usuário
MatBibliotecário	Representa a matrícula do bibliotecário que concebeu o empréstimo	VARCHAR(20)	VARCHAR(20)	Não nulo Chave estrangeira que referencia matrícula na relação Bibliotecário

Devolução: relação que armazena as devoluções realizadas.				
Atributo	Descrição	Tipo	Domínio	Restrição
Código	Representa o código de uma devolução realizada	VARCHAR(20)	VARCHAR(20)	Chave primária
Status	Representa o status de uma devolução realizada	VARCHAR(20)	VARCHAR(20)	Não nulo
Data_Devolução	Representa a data em que o empréstimo foi realizado	DATE	DATE	Não nulo
Observação	Representa uma eventual observação sobre a situação do livro que foi devolvido	VARCHAR(80)	VARCHAR(80)	
CpfUsuário	Representa o cpf do usuário que realizou a devolução	CHARACTER(11)	CHARACTER(11)	Chave estrangeira que referencia cpf na relação Usuário

MatBibliotecário	Representa a matrícula do bibliotecário que recebeu a devolução	VARCHAR(20)	VARCHAR(20)	Chave estrangeira que referencia matrícula na relação Bibliotecário
------------------	---	-------------	-------------	---

<b>Reserva:</b> relação que armazena as reservas solicitadas.				
<b>Atributo</b>	<b>Descrição</b>	<b>Tipo</b>	<b>Domínio</b>	<b>Restrição</b>
Código	Representa o código de uma reserva solicitada	VARCHAR(20)	VARCHAR(20)	Chave primária
Status	Representa o status de uma reserva solicitada	VARCHAR(20)	VARCHAR(20)	Não nulo
Data_Reserva	Representa a data em que a reserva foi solicitada	DATE	DATE	Não nulo
Data_Disponibilidade	Representa a data de disponibilidade do livro que está contido na reserva	DATE	DATE	Não nulo
CpfUsuário	Representa o cpf do usuário que realizou a reserva	CHARACTER(11)	CHARACTER(11)	Não nulo Chave estrangeira que referencia cpf na relação Usuário

MatBibliotecário	Representa a matrícula do bibliotecário que registrou a reserva	VARCHAR(20)	VARCHAR(20)	Não nulo Chave estrangeira que referencia matrícula na relação Bibliotecário
codLivro	Representa o código do livro que será reservado	VARCHAR(20)	VARCHAR(20)	Não nulo Chave estrangeira que referencia código na relação Livro

**Atraso:** relação que armazena os atrasos cometidos.

Atributo	Descrição	Tipo	Domínio	Restrição
Código	Representa o código de um atraso cometido	VARCHAR(20)	VARCHAR(20)	Chave primária
Status	Representa o status de uma devolução realizada	VARCHAR(20)	VARCHAR(20)	Não nulo
Duração	Representa a quantidade de dias em que perdurou o atraso	INT	INT	Não nulo
Data_inicio	Representa a data que iniciou o atraso	DATE	DATE	Não nulo
CpfUsuário	Representa o cpf do usuário que atrasou a entrega	CHARACTER(11)	CHARACTER(11)	Chave estrangeira que referencia cpf na relação Usuário

CodMulta	Representa o código da multa relacionada ao atraso	VARCHAR(20)	VARCHAR(20)	Chave estrangeira que referencia código na relação Multa
----------	--	-------------	-------------	--

**Multa:** relação que armazena as multas.

Atributo	Descrição	Tipo	Domínio	Restrição
Código	Representa o código de uma multa	VARCHAR(20)	VARCHAR(20)	Chave primária
Status	Representa o status de uma multa	VARCHAR(20)	VARCHAR(20)	Não nulo
Razão	Representa uma razão além de atrasos que levaram a multa	VARCHAR(200)	VARCHAR(200)	
Valor	Representa o valor da multa	NUMERIC(5, 2)	NUMERIC(5, 2)	Não nulo

**Doação:** relação que armazena dados sobre as doações realizadas.

Atributo	Descrição	Tipo	Domínio	Restrição
----------	-----------	------	---------	-----------

Código	Representa o código de uma devolução realizada	VARCHAR(20)	VARCHAR(20)	Chave primária
Quant_Livros	Representa a quantidade de livros doados	INT	números inteiros > 0	Não nulo
Data_Doação	Representa a data em que a doação foi realizada	DATE	DATE	Não nulo

CpfUsuário	Representa o cpf do usuário que realizou a doação	CHARACTER(11)	CHARACTER(11)	Chave estrangeira que referencia cpf na relação Usuário
------------	---	---------------	---------------	---

**Devolução\_Livro:** relação que armazena informações sobre a devolução e os livros contidos nela..

Atributo	Descrição	Tipo	Domínio	Restrição
CodDevolução	Representa o código uma devolução realizada	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia código na relação Devolução
CodLivro	Representa o código do livro contido em uma devolução	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia código na relação Livro



**Empréstimo\_Livro:** relação que armazena informações sobre empréstimos e os livros contidos neles.

Atributo	Descrição	Tipo	Domínio	Restrição
CodEmpréstimo	Representa o código uma empréstimo realizada	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia código na relação Empréstimo
CodLivro	Representa o código do livro contido em um empréstimo	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia código na relação Livro

**Doação\_Livro:** relação que armazena informações sobre a doações e os livros contidos nela..

Atributo	Descrição	Tipo	Domínio	Restrição
CodDoação	Representa o código de uma doação realizada	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia código na relação Doação

CodLivro	Representa o código do livro contido em uma deoação	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia código na relação Livro
----------	---	-------------	-------------	--

**Atraso\_Livro:** relação que armazena informações sobre a doações e os livros contidos nela.

Atributo	Descrição	Tipo	Domínio	Restrição
CodAtraso	Representa o código de um atraso cometido	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia código na relação Atraso
CodLivro	Representa o código do livro contido em um atraso	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia código na relação Livro

**Renovações:** relação que armazena informações sobre as renovações de um empréstimo.

Atributo	Descrição	Tipo	Domínio	Restrição
CodEmpréstimo	Representa o código de um empréstimo realizado	VARCHAR(20)	VARCHAR(20)	Chave primária  Chave estrangeira que referencia código na relação Empréstimo
CodRenovação	Representa o código da renovação de um empréstimo	VARCHAR(20)	VARCHAR(20)	Chave primária
Data_Renovação	Representa a data de renovação de um empréstimo	DATE	DATE	Não nulo

## 4. MODELO FÍSICO

### 4.1 Scripts SQL para criação das tabelas

- Criação da relação Usuário

```
create table Usuario (  
    CPF CHARACTER(11),  
    Nome VARCHAR(100) NOT NULL,  
    Email VARCHAR(60) NOT NULL,  
    Telefone VARCHAR(15) NOT NULL,  
    Endereco VARCHAR(100) NOT NULL,  
    CONSTRAINT Usuario_cpf_pk PRIMARY KEY(CPF)  
);
```

- Criação da relação Funcionário

```
create table Funcionario (  
    Matricula VARCHAR(20),  
    Nome VARCHAR(100),  
    CPF CHARACTER(11) UNIQUE NOT NULL,  
    Email VARCHAR(60),  
    Telefone VARCHAR(15) NOT NULL,  
    Endereco VARCHAR(100) NOT NULL,  
    Salario NUMERIC(10, 2) NOT NULL,  
    CONSTRAINT Funcionario_matricula_pk PRIMARY KEY(Matricula),  
    CONSTRAINT Funcionario_salario_check CHECK(Salario >= 1320)  
);
```

- Criação da relação Bibliotecário

```
create table Bibliotecario (  
    Matricula VARCHAR(20),  
    Turno VARCHAR(30) NOT NULL,  
    Escolaridade VARCHAR(50) NOT NULL,  
    CONSTRAINT Bibliotecario_matricula_pk PRIMARY  
KEY(Matricula),  
    CONSTRAINT Bibliotecario_matricula_fk FOREIGN KEY(Matricula)
```

```
REFERENCES Funcionario(Matricula)  
)
```

- Criação da relação Categoria

```
create table Categoria(  
    codigo varchar(20),  
    genero varchar(100) not null,  
    num_bloco int not null,  
    descricao varchar(300),  
    constraint categoria_codigo_pk primary key(codigo)  
)
```

- Criação da relação Livro

```
create table Livro(  
    codigo varchar(20),  
    titulo varchar(100) not null,  
    autor varchar(100) not null,  
    editora varchar(80),  
    data_publicacao date,  
    tipo varchar(50) not null,  
    num_paginas int not null,  
    cod_categoria varchar(20),  
    constraint livro_codigo_pk primary key(codigo),  
    constraint livro_cod_categoria_fk foreign key (cod_categoria)  
    references Categoria (codigo)  
)
```

- Criação da relação Empréstimo

```
create table Emprestimo (  
    codigo varchar(20),  
    status varchar(20) not null,  
    data_emprestimo date not null,  
    cpf_usuario character(11) not null,  
    mat_bibliotecario varchar(20) not null,
```

```

        constraint emprestimo_codigo_pk primary key(codigo),
        constraint emprestimo_usuario_fk foreign key(cpf_usuario)
references Usuario(CPF),
        constraint emprestimo_bibliotecario_fk foreign
key(mat_bibliotecario) references Bibliotecario(matricula)
)

```

- Criação da relação Devolução

```

create table Devolucao (
    codigo varchar(20),
    status varchar(20) not null,
    data_devolucao date not null,
    observacao varchar(80),
    cpf_usuario character(11) not null,
    mat_bibliotecario varchar(20) not null,
    constraint devolucao_codigo_pk primary key(codigo),
    constraint devolucao_usuario_fk foreign key(cpf_usuario)
references Usuario(CPF),
    constraint devolucao_bibliotecario_fk foreign
key(mat_bibliotecario) references Bibliotecario(matricula)
)

```

- Criação da relação Multa

```

create table Multa (
    codigo varchar(20),
    status varchar(20) not null,
    razao varchar(200),
    valor numeric(5, 2) not null,
    constraint multa_codigo_pk primary key(codigo)
);

```

- Criação da relação Atraso

```

create table Atraso (
    codigo varchar(20),

```

```

status varchar(20) not null,
duracao int not null,
data_inicio date not null,
cpf_usuario character(11) not null,
cod_multa varchar(20) not null,
constraint atraso_codigo_pk primary key(codigo),
constraint atraso_usuario_fk foreign key(cpf_usuario) references
Usuario(CPF),
constraint atraso_multa_fk foreign key(cod_multa) references
Multa(codigo)
);

```

- Criação da relação Reserva

```

create table Reserva (
    codigo varchar(20),
    status varchar(20) not null,
    data_reserva date not null,
    data_disponibilidade date not null,
    cpf_usuario character(11) not null,
    mat_bibliotecario varchar(20) not null,
    cod_livro varchar(20) not null,
    constraint reserva_codigo_pk primary key(codigo),
    constraint reserva_usuario_fk foreign key(cpf_usuario) references
Usuario(CPF),
    constraint reserva_bibliotecario_fk foreign key(mat_bibliotecario)
references Bibliotecario(matricula),
    constraint reserva_livro_fk foreign key(cod_livro) references
Livro(codigo)
);

```

- Criação da relação Doação

```

create table Doacao (
    codigo varchar(20),
    quant_livros int not null,
    data_doacao date not null,
    cpf_usuario character(11) not null,

```

```
constraint doacao_codigo_pk primary key(codigo),
constraint doacao_usuario_fk foreign key(cpf_usuario) references
Usuario(CPF)
)
```

- Criação da relação Empréstimo\_Livro

```
create table Emprestimo_Livro (
    cod_emprestimo varchar(20),
    cod_livro varchar(20),
    constraint emprestimo_livro_pk primary key(cod_emprestimo,
cod_livro),
    constraint emprestimo_livro_codLivro_fk foreign key(cod_livro)
references Livro(codigo),
    constraint emprestimo_livro_codEmprestimo_fk foreign
key(cod_emprestimo) references Emprestimo(codigo)
);
```

- Criação da relação Devolução\_Livro

```
create table Devolucao_Livro (
    cod_devolucao varchar(20),
    cod_livro varchar(20),
    constraint devolucao_livro_pk primary key(cod_devolucao,
cod_livro),
    constraint devolucao_livro_codLivro_fk foreign key(cod_livro)
references Livro(codigo),
    constraint devolucao_livro_codDevolucao_fk foreign
key(cod_devolucao) references Devolucao(codigo)
);
```



- Criação da relação Doação\_Livro

```
create table Doacao_Livro (  
    cod_doacao varchar(20),  
    cod_livro varchar(20),  
    constraint doacao_livro_pk primary key(cod_doacao, cod_livro),  
    constraint doacao_livro_codLivro_fk foreign key(cod_livro)  
references Livro(codigo),  
    constraint doacao_livro_codDoacao_fk foreign key(cod_doacao)  
references Doacao(codigo)  
)
```

- Criação da relação Atraso\_Livro

```
create table Atraso_Livro (  
    cod_atraso varchar(20),  
    cod_livro varchar(20),  
    constraint atraso_livro_pk primary key(cod_atraso, cod_livro),  
    constraint atraso_livro_codLivro_fk foreign key(cod_livro)  
references Livro(codigo),  
    constraint atraso_livro_codAtraso_fk foreign key(cod_atraso)  
references Atraso(codigo)  
);
```

- Criação da relação Renovações

```
create table Renovacoes (  
    cod_emprestimo varchar(20),  
    cod_renovacao varchar(20),  
    data_renovacao date not null,  
    constraint renovacoes_pk primary key(cod_emprestimo,  
cod_renovacao),  
    constraint renovacoes_codEmprestimo_fk foreign  
key(cod_emprestimo) references Emprestimo(codigo)
```

```
);
```

## 4.2 Scripts de criação de índices

```
create index indice_bibliotecario on Bibliotecario(matricula);
create index indice_usuario on Usuario(cpf);
create index indice_funcionario on Funcionario(matricula);
create index indice_categoria on Categoria(codigo);
create index indice_livro on Livro(codigo);
create index indice_emprestimo on Emprestimo(codigo);
create index indice_devolucao on Devolucao(codigo);
create index indice_reserva on Reserva(codigo);
create index indice_multa on Multa(codigo);
create index indice_atraso on Atraso(codigo);
create index indice_doacao on Doacao(codigo);
create index indice_emprestimo_livro on Emprestimo_Livro(cod_emprestimo);
create index indice_devolucao_livro on Devolucao_Livro(cod_devolucao);
create index indice_doacao_livro on Doacao_Livro(cod_doacao);
create index indice_atraso_livro on Atraso_Livro(cod_atraso);
create index indice_renovacoes on Renovacoes(cod_emprestimo);
```

## 4.3 Scripts de criação de visões

- Visão que obtém todas as multas que possuem status de pendente

```
create view multas_pendentes as select codigo
from Multa
where status = 'pendente';
```

- Visão que obtém as informações de todos os atrasos pendentes e todas as informações dos usuários relacionadas a elas.

```
create view usuarios_atraso_pendentes as select U.*, A.*  
from Usuario U inner join Atraso A on U.cpf = A.cpf_usuario  
where A.status = 'pendente';
```

- Visão que obtém todos os empréstimos e todas as informações dos livros relacionados a ele.

```
create view visao_livro_emprestimo_livro as select *  
from Emprestimo_Livro EL, Livro L  
where EL.cod_livro = L.codigo;
```

#### 4.4 Povoamento

- Relação Funcionário

```
insert into Funcionario (matricula, nome, Cpf, email, telefone, endereco,  
salario)  
values ('1111-1', 'Kamila', '12314588929', 'kamila@gmail.com', '9992678900',  
'Rua Bernado Nogueira', 2000),  
('1111-2', 'Veneranda', '15646797851', 'veneranda@gmail.com',  
'992867456', 'Rua Constantino da Silva', 1400),
```

```
('1111-3', 'Patrícia', '22287698134', 'patricia@gmail.com', '9654326765',  
'Rua Cunha Da Costa', 1400),  
('1111-4', 'Kauan', '12365378900', 'kauan@gmail.com', '995678912', 'Rua  
Bernado Nogueira', 1400),  
('1111-5', 'Viviane', '75565988856', 'vivi@gmail.com', '987654990',  
'Centro', 2500),  
('1111-6', 'Rafael', '98765432100', 'rafael@gmail.com', '9998887776',  
'Avenida das Flores', 1800),  
('1111-7', 'Juliana', '45678901234', 'juliana@gmail.com', '9888776655', 'Rua  
dos Amigos', 1600);
```

- Relação Bibliotecário

```
insert into Bibliotecario (matricula, turno, escolaridade)  
values ('1111-1', 'tarde', 'superior completo'),  
('1111-4', 'manha', 'ensino medio completo'),  
('1111-5', 'noite', 'superior completo'),  
('1111-2', 'manha', 'superior completo'),  
('1111-3', 'tarde', 'ensino medio completo');
```

- Relação Usuário

```
insert into Usuario (cpf, nome, email, telefone, endereco)  
values ('12377654229', 'Diogo', 'diogo@gmail.com', '991234567', 'Rua Senhor  
Nogueira'),  
('67896542987', 'Ana Clara', 'ana@gmail.com', '99976543', 'Rua das  
Flores'),  
('99278906544', 'Gabriela', 'gabi@gmail.com', '992654378', 'Rua Cunha
```

da Costa'),

('11177659321', 'Misael', 'misael@gmail.com', '9937654312', 'Rua  
Constatino da Silva'),

('76549875433', 'Estephany', 'estephany@gmail.com', '991886654', 'Rua  
Marielle Franco');

- Relação Livro

```
insert into Livro (codigo, titulo, autor, editora, cod_categoria, tipo,
data_publicacao, num_paginas)
values ('6558380544', 'A biblioteca da meia-noite', 'Matt Haig', 'Bertrand Brasil',
'23467', 'literario', '27/07/2021', 308),
      ('8535932569', 'Pessoas normais', 'Sally Rooney', 'Companhia das
Letras', '23467', 'literario', '30/09/2019', 264),
      ('8584391509', 'Os sete maridos de Evelyn Hugo', 'Taylor Jenkis', null,
'23467', 'literario', '21/08/2019', 360),
      ('9780292724', 'Laços de Família', 'Clarice Lispector', 'Rocco Digital',
'76541', 'literario', '19/06/2020', 135),
      ('8535931805', 'Os contos', 'Lygia Fagundes Telles', 'Companhia das
Letras', '76541', 'literario', '28/11/2018', 762),
      ('6555874600', 'Sentimento do Mundo', 'Carlos Drummond de
Andrade', 'Record', '12678', 'literário', '18/04/2022', 93),
      ('8547209824', 'Historia Global, Brasil e Geral', 'Gilberto Cotrim',
'Saraiva', '76542', 'didático', '03/12/2019', 832),
      ('6555875232', 'A rosa do povo', 'Carlos Drummond', 'Recorde', '12678',
'literario', null, 238 ),
      ('6588490365', 'Antes que o café esfrie', 'Toshikazu Kawaguchi', null,
'76541', 'literario', '15/04/2022', 208),
      ('8525064141', 'O livro da História', 'Rafael Longo', null, '76542',
'didático', null, 352);
```

- Relação Categoria

```
insert into Categoria (codigo, genero, num_bloco, descricao)
values ('23467', 'Romance', 1, '...'),
      ('12678', 'Poema', 2, '...'),
      ('98753', 'Novela', 3, '...'),
      ('76541', 'Conto', 4, '...'),
      ('98765', 'Drama histórico', 5, '...'),
      ('76542', 'História', 6, '...');
```

- Relação Empréstimo

```
insert into Empréstimo (codigo, status, data_emprestimo, cpf_usuario,
mat_bibliotecario)
values ('11191', 'pendente', '20/12/2022', '99278906544', '1111-3'),
      ('11192', 'pendente', '09/12/2022', '76549875433', '1111-5'),
      ('11193', 'concluido', '15/12/2022', '12377654229', '1111-1'),
      ('11194', 'pendente', '15/12/2022', '67896542987', '1111-4'),
      ('11198', 'concluido', '10/12/2022', '99278906544', '1111-4'),
      ('11196', 'concluido', '12/12/2022', '11177659321', '1111-3'),
      ('11199', 'concluido', '17/12/2022', '76549875433', '1111-1'),
      ('11200', 'concluido', '23/11/2022', '12377654229', '1111-5'),
      ('11201', 'pendente', '05/01/2023', '67896542987', '1111-4'),
      ('11202', 'concluido', '08/01/2023', '99278906544', '1111-2'),
      ('11203', 'concluido', '10/01/2023', '12377654229', '1111-5'),
      ('11204', 'pendente', '12/01/2023', '11177659321', '1111-2'),
      ('11205', 'pendente', '15/01/2023', '76549875433', '1111-1');
```

- Relação Empréstimo\_Livro

```
insert into Empréstimo_Livro (cod_emprestimo, cod_livro)
values ('11191', '9780292724'),
      ('11192', '8584391509'),
      ('11193', '8584391509'),
      ('11194', '6555874600'),
      ('11198', '8547209824'),
      ('11196', '9780292724'),
      ('11199', '6555874600'),
      ('11200', '8584391509');
```

- Relação Devolução

```
insert into Devolucao (codigo, status, data_devolucao, observacao,
cpf_usuario, mat_bibliotecario)
values ('91234', 'concluida', '28/12/2023', '...' , '12377654229', '1111-1'),
      ('91764', 'concluida', '20/12/2023', null, '99278906544', '1111-1'),
      ('91334', 'concluida', '28/12/2023', null, '12377654229', '1111-4'),
      ('91094', 'concluida', '23/12/2023', '...' , '76549875433', '1111-5'),
      ('91774', 'concluida', '10/12/2023', '...' , '11177659321', '1111-3');
```

- Relação Devolucao\_Livro

```
insert into Devolucao_Livro (cod_devolucao, cod_livro)
values ('91234', '6555874600'),
      ('91764', '9780292724'),
      ('91334', '8584391509'),
      ('91094', '8547209824'),
```

```
('91774', '8584391509');
```

- Relação Doação

```
insert into Doacao (codigo, quant_livros, data_doacao, cpf_usuario)
values ('64576', 2, '08/09/2023', '76549875433'),
      ('64366', 1, '12/10/2023', '11177659321'),
      ('64987', 1, '15/10/2023', '67896542987'),
      ('64332', 1, '17/08/2023', '67896542987'),
      ('64637', 1, '15/09/2023', '12377654229');
```

- Relação Doação\_Livro

```
insert into doacao_livro (cod_doacao, cod_livro)
values ('64576', '9780292724'),
      ('64576', '6555874600'),
      ('64366', '6558380544'),
      ('64987', '8547209824'),
      ('64332', '8535932569'),
      ('64637', '8584391509');
```

- Relação Reserva

```
insert into Reserva (codigo, status, data_reserva, data_disponibilidade,
cpf_usuario, mat_bibliotecario, cod_livro)
values ('29787', 'pendente', '20/12/2022' , '27/12/2022', '11177659321', '1111-1',
```



```
'6555874600'),
    ('29768', 'concluida', '05/12/2022', '08/12/2022', '99278906544', '1111-2',
'6555874600'),
    ('29732', 'pendente', '19/12/2022', '22/12/2022', '99278906544', '1111-4',
'8535932569'),
    ('29709', 'concluida', '19/11/2022', '21/11/2022', '12377654229', '1111-5',
'6558380544'),
    ('29721', 'concluida', '03/11/2022', '12/11/2022', '12377654229', '1111-3',
'8535932569');
```

- Relação Multa

```
insert into Multa (codigo, status, razao, valor)
values ('33301', 'pendente', 'Atraso na devolução', 0.5),
    ('33302', 'pendente', 'Danos ao livro', 0.25),
    ('33303', 'concluido', 'Perda do livro', 0.50),
    ('33304', 'pendente', 'Atraso na devolução', 0.75),
    ('33305', 'concluido', 'Danos ao livro', 1.00);
```

- Relação Atraso

```
insert into Atraso (codigo, duracao, status, data_inicio, Cpf_usuario,
cod_multa)
values ('22201', 5, 'pendente', '28/12/2022', '99278906544', '33301'),
    ('22202', 3, 'pendente', '17/12/2022', '12377654229', '33302'),
    ('22203', 2, 'concluido', '23/11/2022', '12377654229', '33303'),
    ('22204', 4, 'pendente', '04/01/2023', '67896542987', '33304'),
    ('22205', 2, 'concluido', '16/01/2023', '99278906544', '33305');
```

- Relação Atraso\_Livro

```
insert into Atraso_Livro (cod_atraso, cod_livro)
values ('22201', '6558380544'),
      ('22202', '8535932569'),
      ('22203', '8584391509'),
      ('22204', '9780292724'),
      ('22205', '8535931805');
```

- Relação Renovações

```
insert into Renovacoes (cod_emprestimo, cod_Renovacao, data_Renovacao)
values ('11191', '22211', '27/12/2022'),
      ('11193', '22212', '22/12/2022'),
      ('11196', '22213', '20/12/2022'),
      ('11199', '22214', '24/12/2022'),
      ('11202', '22215', '10/01/2023');
```

#### 4.5 Scripts das consultas

- Consulta de todos os usuários que realizaram pelo menos uma doação e não possuem registro de atraso - Consulta envolvendo operações com conjuntos

```
select U.*
from Usuario U, Doacao D
where U.cpf = D.cpf_usuario
intersect
select U.*
from Usuario U
where U.cpf not in (select cpf_usuario from Atraso)
```

- Consulta para obter todos os livros que foram emprestados, mas não possuem registro de reserva - Consulta envolvendo operação com conjuntos

```
select L.*  
from Livro L, Emprestimo_Livro EL  
where L.codigo = EL.cod_livro  
except  
select L.*  
from Livro L, Reserva R  
where L.codigo = R.cod_livro
```

- Listar o título, o código e o autor de todos os livros que não possuem editoras, mas possuem uma data de publicação e ordenar por ordem decrescente do autor - Consulta envolvendo comparação com valores nulos

```
select codigo, titulo, autor  
from Livro  
where editora is null AND data_publicacao is not null;
```

- Listar todas as informações das devoluções em que o status é de concluída e que possuem alguma observação - Consulta envolvendo comparação com valores nulos

```
select *  
from Devolucao  
where status = 'concluida' AND observacao is not null;
```

- Obter todas as informações de usuários com multas pendentes - Consulta envolvendo aninhamento

```
select *  
from Usuario  
where cpf in (  
    select cpf_usuario as cpf  
    from Atraso  
    where cod_multa in (select codigo from multas_pendentes));
```

- Obter o nome e o cpf dos usuários que possuem e pelo menos dois livros emprestados e ordenar por ordem decrescente do nome - Consulta envolvendo aninhamento

```
select cpf, nome  
from Usuario  
where cpf in  
    (select E.cpf_usuario  
    from Emprestimo E, Emprestimo_Livro EL  
    where E.codigo = EL.cod_emprestimo  
    group by E.cpf_usuario  
    having count(*) >= 2)  
order by nome desc;
```

- Obter o código, o gênero e a quantidade de livros que estão associados a cada categoria - Consulta envolvendo agrupamento

```
select Cat.codigo, Cat.genero, count(*) as quant_livros  
from Categoria Cat, Livro L
```

```
where L.cod_categoria = Cat.codigo  
group by Cat.codigo, Cat.genero;
```

- Obter o nome, cpf e a quantidade de empréstimos realizados pelos usuários que possuem status de pendente em que a quantidade é superior a 1 - Consulta envolvendo agrupamento com filtro

```
select U.cpf, U.nome, count(*) as quant_pendentes  
from Usuario U, Emprestimo E  
where U.cpf = E.cpf_usuario AND E.status = 'pendente'  
group by U.cpf, U.nome  
having count(*) >= 2;
```

- Listar livros em que o sobrenome do autor possua uma determinada sequência de caracteres, em ordem decrescente do autor - Consulta envolvendo busca por substrings

```
select *  
from Livro  
where autor like '%Lispector';  
order by nome asc;
```

- Obter bibliotecários em que o número do telefone que iniciam com uma determinada sequência de dígitos - Consulta envolvendo busca por substrings

```
select F.*  
from Funcionario F natural inner join Bibliotecario B  
where F.telefone like '992%';
```

- Obter todas as informações dos livros, incluindo o gênero da categoria e o número do bloco em que ela está associada - Consulta envolvendo junção

```
select L.*, Cat.genero, num_bloco  
from Livro L inner join Categoria Cat on L.cod_categoria = Cat.codigo;
```

- Consulta para obter a quantidade total de livros emprestados por categoria -  
Consulta envolvendo junção

```
select Cat.genero, count(*) as livros_emprestados  
from Categoria Cat inner join visao_livro_emprestimo_livro VLEL on  
Cat.codigo = VLEL.cod_categoria  
group by Cat.genero;
```

- Obter todas as doações feitas por usuários que já realizaram um empréstimo -  
Consulta envolvendo aninhamento correlacionado

```
select *  
from Doacao  
where exists (select cpf_usuario from Emprestimo);
```

- Obter a soma da quantidade de todos os livros doados recebidos na biblioteca -  
Consulta envolvendo função agregada

```
select sum(quant_livros) as livros_doados  
from Doacao
```

- Obter a média salarial de todos os funcionários que não são bibliotecários -  
Consulta envolvendo função agregada

```
select avg(salario) as media_salarial  
from Funcionario F, Bibliotecario B
```

where  $F.matricula \leftrightarrow B.matricula$