

**INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA DA PARAÍBA**

**TECNOLOGIA EM SISTEMAS PARA INTERNET**

MARCELA BARRETO DE OLIVEIRA KRAMER

MATHEUS PEREIRA DE SOUSA

PABLO DE LIMA MARQUES ESTRELA

RICARDO LUIZ DUARTE PRADO

**PROJETO PARA BANCO DE DADOS RELACIONAL DE BIBLIOTECA UNIVERSITÁRIA**

JOÃO PESSOA - PB

2022

**1 INTRODUÇÃO**

1.1 CONTEXTUALIZAÇÃO DO PROJETO

O projeto em questão aborda o funcionamento de uma biblioteca universitária através de uma modelagem conceitual. O esquema é modelado na ferramenta brModelo e as informações são representadas com intuito de oferecer o máximo de detalhes possível.

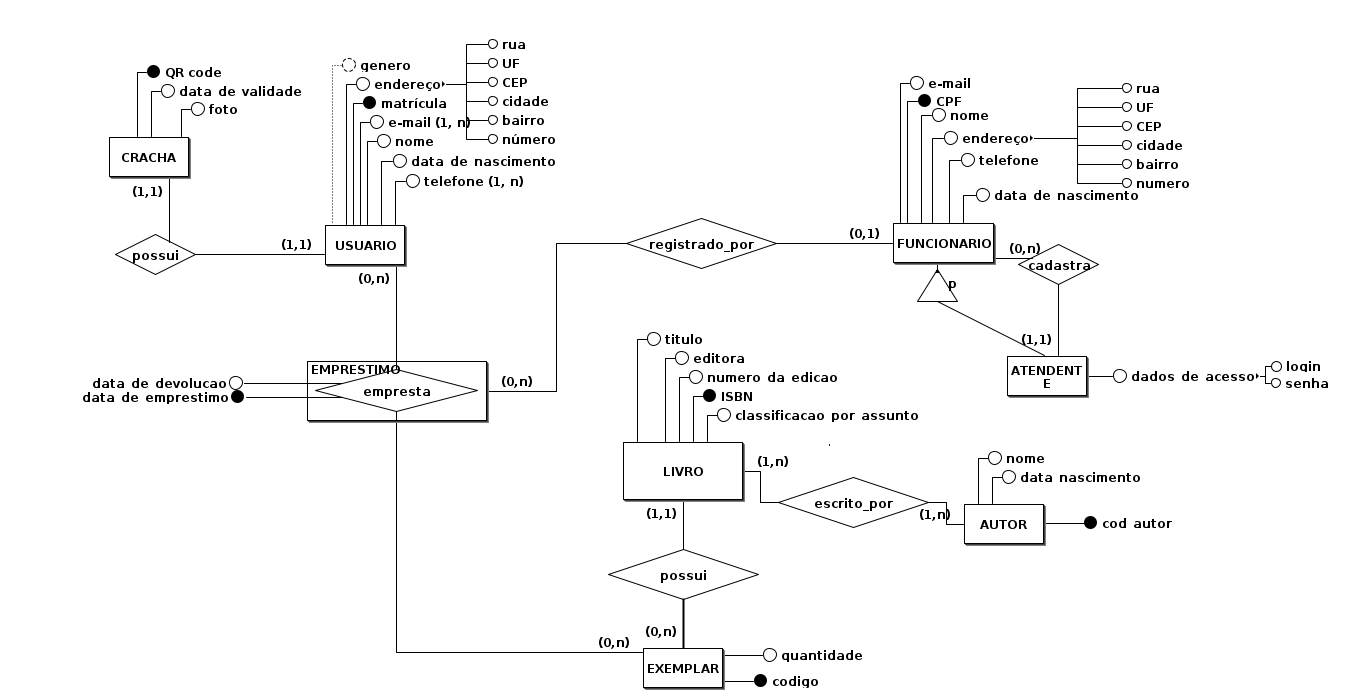
1.2 MINIMUNDO

Uma biblioteca universitária precisa melhorar o seu sistema de gerenciamento próprio. Para isso, decide reformular a organização dos seus dados da seguinte forma: um usuário é cadastrado com matrícula, nome, data de nascimento, telefones, e-mails, endereço (rua, UF, CEP, cidade, bairro e número) e gênero (opcional). Além disso, é mantido um registro do crachá dos usuários, que, por sua vez, é registrado com um QR code, data de validade e a foto do usuário. A biblioteca também armazena as informações dos livros que possui, especificando o ISBN, o título, a editora, o número da edição e a sua classificação por assunto, assim como mantém o registro dos autores desses livros, detalhando o código do autor, seu nome e sua data de nascimento. Ademais, os funcionários que trabalham na biblioteca também são registrados, nesse caso, com CPF, nome, endereço (rua, UF, CEP, cidade, bairro e número), telefone, data de nascimento e e-mail. Por fim, seguem as demais regras de negócio:

* Um usuário possui apenas um crachá, bem como um crachá só identifica um único usuário.
* Um usuário pode ou não pegar emprestado vários exemplares de livros, assim como um mesmo exemplar de livro pode ou não ser emprestado a vários usuários. Esse empréstimo — do qual são registradas as datas de empréstimo e devolução — pode ser feito online ou por intermédio de um funcionário. Logo, o empréstimo pode ser registrado ou não por um funcionário, bem como um funcionário pode registrar ou não vários empréstimos.
* Um livro pode ter ou não vários exemplares, sendo cada um desses registrado com um código identificador. No entanto, um exemplar só pode ser de um único livro e só existe caso haja esse livro correspondente.
* Um livro é escrito por um ou vários autores e um autor pode escrever um ou mais livros.
* Um funcionário pode ser especializado como atendente, possuindo, assim, dados de acesso ao sistema da biblioteca, nesse caso, login e senha. O atendente, por sua vez, é o responsável por cadastrar nenhum ou vários funcionários de maneira geral, mas um funcionário é cadastrado por um único atendente.

**2 PROJETO CONCEITUAL**

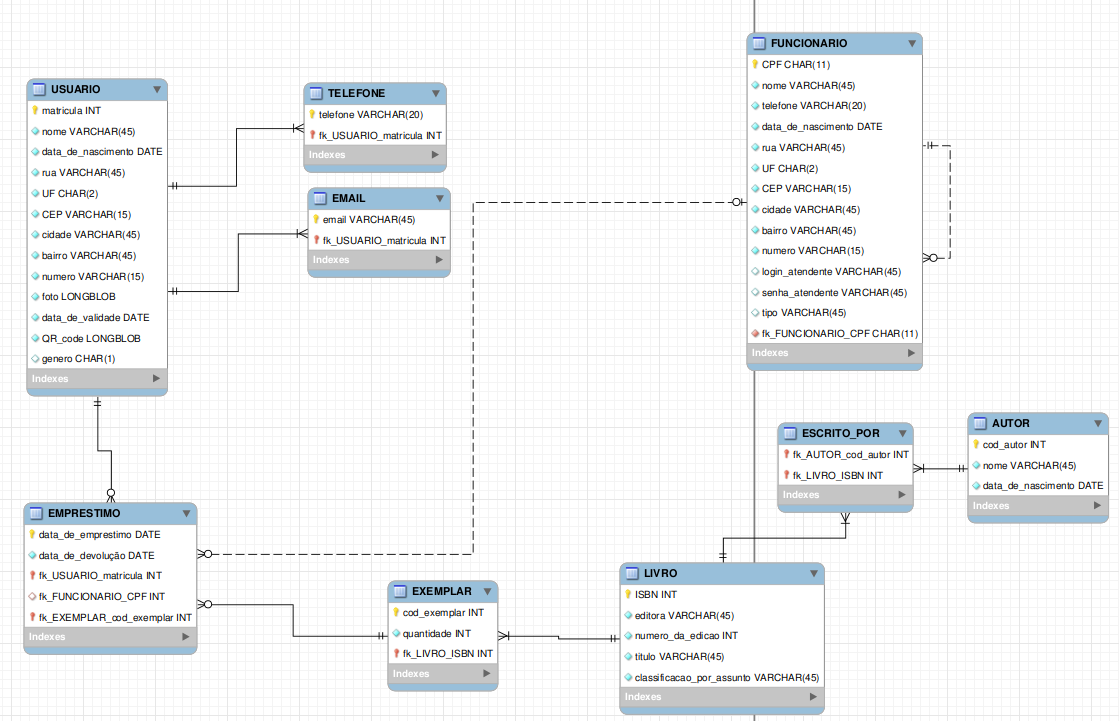
2.1 DIAGRAMA E-R

Figura 1 - Modelo conceitual da Biblioteca Universitária (brModelo)

**3 PROJETO LÓGICO**

3.1 DIAGRAMA RELACIONAL

Figura 2 - Modelo lógico da Biblioteca Universitária (MySQL Workbench)



3.2 DICIONÁRIO DE DADOS

TABELA AUTOR

| **Coluna** | **Tipo** | **PK** | **FK** | **Ùnico** | **NN** | **Default** | **Comentário** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| cod\_autor | INT | ✓ |  |  | ✓ |  | Código identificador do autor. |
| nome | VARCHAR(45) |  |  |  | ✓ |  | Nome do autor. |
| data\_de\_nascimento | DATE |  |  |  | ✓ |  | Data de nascimento do autor.  Não pode assumir valores futuros. |

TABELA EMAIL

| **Coluna** | **Tipo** | **PK** | **FK** | **Ùnico** | **NN** | **Default** | **Comentário** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| email | VARCHAR(45) | ✓ |  |  | ✓ |  | Endereço de e-mail usuário da biblioteca.  Garantir a unicidade da chave primária(email, fk\_USUARIO\_matricula). |
| fk\_USUARIO\_matricula | INT | ✓ | ✓ |  | ✓ |  | Referencia a tabela USUARIO. Garantir a unicidade da chave primária(email, fk\_USUARIO\_matricula). |

TABELA EMPRÉSTIMO

| **Coluna** | **Tipo** | **PK** | **FK** | **Ùnico** | **NN** | **Default** | **Comentário** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| data\_de\_emprestimo | DATE | ✓ |  |  | ✓ | TODAY | Data de empréstimo realizado. Não pode assumir valor futuro. Garante unicidade da chave primária(data\_de\_emprestimo, fk\_USUARIO\_matricula, fk\_EXEMPLAR\_cod\_exemplar). |
| data\_de\_devolução | DATE |  |  |  | ✓ |  | Data de devolução de empréstimo realizado. Não pode assumir valores passados. |
| fk\_USUARIO\_matricula | INT | ✓ | ✓ |  | ✓ |  | Referencia a tabela USUARIO. Garante unicidade da chave primária(data\_de\_emprestimo, fk\_USUARIO\_matricula, fk\_EXEMPLAR\_cod\_exemplar). |
| fk\_FUNCIONARIO\_CPF | INT |  | ✓ |  |  |  | Referencia a tabela FUNCIONARIO. |
| fk\_EXEMPLAR\_cod\_exemplar | INT | ✓ | ✓ |  | ✓ |  | Referencia a tabela EXEMPLAR. Garante unicidade da chave primária(data\_de\_emprestimo, fk\_USUARIO\_matricula, fk\_EXEMPLAR\_cod\_exemplar). |

TABELA ESCRITO\_POR

| **Coluna** | **Tipo** | **PK** | **FK** | **Ùnico** | **NN** | **Default** | **Comentário** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| fk\_AUTOR\_cod\_autor | INT | ✓ | ✓ |  | ✓ |  | Referencia a tabela AUTOR. Garantir a unicidade da chave primária(fk\_AUTOR\_cod\_autor, fk\_LIVRO\_ISBN). |
| fk\_LIVRO\_ISBN | INT | ✓ | ✓ |  | ✓ |  | Referência a tabela LIVRO. Garantir a unicidade da chave primária(fk\_AUTOR\_cod\_autor, fk\_LIVRO\_ISBN). |

TABELA EXEMPLAR

| **Coluna** | **Tipo** | **PK** | **FK** | **Ùnico** | **NN** | **Default** | **Comentário** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| cod\_exemplar | INT | ✓ |  |  | ✓ |  | Código de exemplar de livro  Garantir a unicidade da chave primária (cod\_exemplar, fk\_LIVRO\_ISBN). |
| quantidade | INT |  |  |  | ✓ | 1 | Quantidade de exemplares disponíveis.  Só pode assumir valores maiores que zero. |
| fk\_LIVRO\_ISBN | INT | ✓ | ✓ |  | ✓ |  | Referencia a tabela LIVRO.  Garantir a unicidade da chave primária (cod\_exemplar, fk\_LIVRO\_ISBN). |

TABELA FUNCIONÁRIO

| **Coluna** | **Tipo** | **PK** | **FK** | **Ùnico** | **NN** | **Default** | **Comentário** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CPF | CHAR(11) | ✓ |  |  | ✓ |  | Cadastro de Pessoa Física do funcionário da biblioteca. |
| nome | VARCHAR(45) |  |  |  | ✓ |  | Nome do funcionário da biblioteca. |
| telefone | VARCHAR(20) |  |  |  | ✓ |  | Número de telefone do funcionário da biblioteca. |
| data\_de\_nascimento | DATE |  |  |  | ✓ |  | Data de nascimento do funcionário da biblioteca.  Não pode assumir valores futuros. |
| rua | VARCHAR(45) |  |  |  | ✓ |  | Logradouro do funcionário da biblioteca. |
| UF | CHAR(2) |  |  |  | ✓ |  | Unidade Federativa do funcionário da biblioteca.  Pode assumir apenas os valores: AM, AP, PA, PE, PB, PI, MA, BA, SE, CE, RJ, SP, MG, ES, RN, RS, SC, PR, TO, RR, RO, AC, DF. |
| CEP | VARCHAR(15) |  |  |  | ✓ |  | Código postal do funcionário da biblioteca. |
| cidade | VARCHAR(45) |  |  |  | ✓ |  | Cidade do funcionário da biblioteca. |
| bairro | VARCHAR(45) |  |  |  | ✓ |  | Bairro do funcionário da biblioteca. |
| numero | VARCHAR(15) |  |  |  | ✓ |  | Número do endereço do funcionário da biblioteca. |
| login | VARCHAR  (45) |  |  | ✓ |  |  | Login do atendente da biblioteca. Se o tipo do funcionário for ‘atendente’, não pode assumir valor nulo. |
| senha | VARCHAR  (45) |  |  |  |  |  | Senha do atendente da biblioteca. Se o tipo do funcionário for ‘atendente’, não pode assumir valor nulo. |
| tipo | VARCHAR(45) |  |  |  |  |  | Tipo do funcionário da biblioteca.  Quando login e senha forem valores não nulos (o usuário é atendente), o tipo possui valor ‘atendente’. |
| fk\_FUNCIONARIO\_CPF | INT |  | ✓ |  | ✓ |  | Referencia a tabela FUNCIONARIO. |

TABELA LIVRO

| **Coluna** | **Tipo** | **PK** | **FK** | **Ùnico** | **NN** | **Default** | **Comentário** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| ISBN | INT | ✓ |  |  | ✓ |  | Código identificador internacional do livro. |
| editora | VARCHAR(45) |  |  |  | ✓ |  | Nome da editora responsável pela publicação do livro. |
| numero\_da\_edicao | INT |  |  |  | ✓ | 1 | Número da edição do livro.  Só pode assumir valores maiores que zero. |
| titulo | VARCHAR(45) |  |  |  | ✓ |  | Título do livro. |
| classificacao\_por\_assunto | VARCHAR(45) |  |  |  | ✓ |  | Classificação por assunto do livro. |

TABELA TELEFONE

| **Coluna** | **Tipo** | **PK** | **FK** | **Ùnico** | **NN** | **Default** | **Comentário** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| telefone | VARCHAR(20) | ✓ |  |  | ✓ |  | Número de telefone do usuário da biblioteca.  Garantir a unicidade da chave primária(telefone, fk\_USUARIO\_matricula). |
| fk\_USUARIO\_matricula | INT | ✓ | ✓ |  | ✓ |  | Referencia a tabela USUARIO. Garantir a unicidade da chave primária(telefone, fk\_USUARIO\_matricula). |

TABELA USUÁRIO

| **Coluna** | **Tipo** | **PK** | **FK** | **Ùnico** | **NN** | **Default** | **Comentário** |
| --- | --- | --- | --- | --- | --- | --- | --- |
| matricula | INT | ✓ |  |  | ✓ |  | Matrícula do usuário da biblioteca. |
| nome | VARCHAR(45) |  |  |  | ✓ |  | Nome do usuário da biblioteca. |
| data\_de\_nascimento | DATE |  |  |  | ✓ |  | Data de nascimento do usuário da biblioteca.  Não pode assumir um valor futuro. |
| rua | VARCHAR(45) |  |  |  | ✓ |  | Logradouro do usuário da biblioteca. |
| UF | CHAR(2) |  |  |  | ✓ |  | Unidade Federativa do usuário da biblioteca.  Pode assumir apenas os valores: AM, AP, PA, PE, PB, PI, MA, BA, SE, CE, RJ, SP, MG, ES, RN, RS, SC, PR, TO, RR, RO, AC, DF. |
| CEP | VARCHAR(15) |  |  |  | ✓ |  | Código postal do usuário da biblioteca. |
| cidade | VARCHAR(45) |  |  |  | ✓ |  | Cidade do usuário da biblioteca. |
| bairro | VARCHAR(45) |  |  |  | ✓ |  | Bairro do usuário da biblioteca. |
| numero | VARCHAR(15) |  |  |  | ✓ |  | Número do endereço do usuário da biblioteca. |
| foto | LONGBLOB |  |  | ✓ | ✓ |  | BLOB(Binary Large Object) da foto do crachá do usuário da biblioteca. Garantir unicidade da chave candidata. |
| data\_de\_validade | DATE |  |  |  | ✓ |  | Data de validade do crachá do usuário da biblioteca.  Não pode assumir valores passados. |
| QR\_code | LONGBLOB |  |  | ✓ | ✓ |  | LONGBLOB(Long Binary Large Object) do código de barras bidimensional do crachá do usuário da biblioteca. É gerado a partir da matrícula e nome do usuário.  Garantir unicidade da chave candidata. |
| genero | CHAR(1) |  |  |  |  |  | Gênero do usuário da biblioteca. Pode assumir apenas os valores: 'M' ou 'F'. |

**4 SCRIPT SQL**

/\*\*\* Criação do banco de dados BibliotecaUniv \*\*\*/

CREATE DATABASE BibliotecaUniv;

/\*\*\* Utilização do banco de dados BibliotecaUniv \*\*\*/

use BibliotecaUniv;

/\*\*\* Criação da Tabela Autor \*\*\*/

create table Autor(

cod\_autor int not null ,

nome varchar(45) not null,

data\_de\_nascimento date not null,

constraint pk\_autor primary key (cod\_autor),

constraint ck\_data\_de\_nascimento\_autor check(data\_de\_nascimento < sysdate())

);

/\*\*\* Criação da Tabela Livro \*\*\*/

create table Livro(

ISBN int not null,

editora varchar(45) not null,

numero\_da\_edicao int default 1,

titulo varchar(45) not null,

classificacao\_por\_assunto varchar(45) not null,

constraint pk\_ISBN primary key (ISBN),

constraint ck\_numero\_da\_edicao check(numero\_da\_edicao > 0),

constraint ck\_ISBN\_livro check(ISBN > 0)

);

/\*\*\* Criação da Tabela Exemplar \*\*\*/

create table Exemplar(

cod\_exemplar int not null,

quantidade int not null,

ISBN\_livro int not null,

constraint pk\_exemplar primary key (cod\_exemplar,ISBN\_livro),

constraint fk\_livro\_exemplar foreign key (ISBN\_livro) references Livro (ISBN)

);

/\*\*\* Criação da Tabela Escrito\_por \*\*\*/

create table Escrito\_por(

cod\_autor int not null,

ISBN\_livro int not null,

constraint fk\_cod\_autor foreign key (cod\_autor) references Autor (cod\_autor),

constraint fk\_livro\_escrito\_por foreign key (ISBN\_livro) references Livro (ISBN),

constraint pk\_escrito\_por primary key (cod\_autor,ISBN\_livro)

);

/\*\*\* Criação da Tabela Usuário \*\*\*/

create table Usuario(

matricula int not null,

nome varchar(45) not null,

data\_de\_nascimento date not null,

rua varchar(45) not null,

UF char(2) not null,

CEP varchar(15) not null,

cidade varchar(45) not null,

bairro varchar(45) not null,

numero varchar(15) not null,

foto longblob not null,

data\_de\_validade date not null,

qr\_code longblob not null,

genero char(1) null,

constraint pk\_usuario primary key (matricula),

constraint ck\_data\_de\_nascimento\_usuario check(data\_de\_nascimento < sysdate()),

constraint ck\_data\_de\_validade\_usuario check(data\_de\_validade > sysdate())

);

/\*\*\* Criação da Tabela Email \*\*\*/

create table Email(

email varchar(45) not null,

matricula\_usuario int not null,

constraint fk\_usuario\_email foreign key (matricula\_usuario) references Usuario (matricula),

constraint pk\_email primary key (matricula\_usuario,email)

);

/\*\*\* Criação da Tabela Telefone \*\*\*/

create table Telefone(

telefone varchar(45) not null,

matricula\_usuario int not null,

constraint fk\_usuario\_telefone foreign key (matricula\_usuario) references Usuario (matricula),

constraint pk\_telefone primary key (matricula\_usuario,telefone)

);

/\*\*\* Criação da Tabela Funcionario \*\*\*/

create table Funcionario(

cpf char(11) not null,

nome varchar(45) not null,

telefone varchar(20) not null,

data\_de\_nascimento date not null,

rua varchar(45) not null,

uf varchar(2) not null,

cep varchar(15) not null,

cidade varchar(45) not null,

bairro varchar(45) not null,

numero varchar(15) not null,

login\_atendente varchar(45) null,

senha\_atendente varchar(45) null,

tipo varchar(45) null,

cpf\_responsavel\_cadastro char(11) null,

constraint pk\_funcionario primary key (cpf),

constraint fk\_atendente\_responsavel foreign key (cpf\_responsavel\_cadastro) references Funcionario(cpf),

constraint uk\_login unique (login\_atendente),

constraint ck\_funcionario\_cpf check(char\_length(cpf)=11),

constraint ck\_data\_de\_nascimento\_funcionario check(data\_de\_nascimento < sysdate())

);

/\*\*\* Criação da Tabela Empréstimo \*\*\*/

create table Emprestimo(

matricula\_usuario int not null,

cpf\_funcionario char(11) null,

cod\_exemplar int not null,

data\_de\_emprestimo date not null,

data\_de\_devolucao date not null,

constraint fk\_usuario\_matricula foreign key (matricula\_usuario) references Usuario (matricula),

constraint fk\_cod\_exemplar foreign key (cod\_exemplar) references Exemplar (cod\_exemplar),

constraint fk\_cpf\_funcionario foreign key (cpf\_funcionario) references Funcionario (cpf),

constraint ck\_data\_de\_emprestimo check(data\_de\_emprestimo <= sysdate()),

constraint ck\_data\_de\_devolucao check(data\_de\_devolucao > sysdate()),

constraint pk\_emprestimo primary key (matricula\_usuario,cod\_exemplar,data\_de\_emprestimo)

);

/\* Inserindo informações na tabela autor \*/

insert into Autor(cod\_autor,nome, data\_de\_nascimento)

values(1,'Eva Heller', '1948-08-08'), /\*psicologia das cores\*/

(2, 'Andrew Stuart Tanenbaum', '1944-03-16'),/\* Sistemas Operativos Modernos\*/

(3, 'Charles Duhigg', '1974-06-06'), /\*o poder do hábito\*/

(4, 'Stephen Hawking', '1942-01-08'), /\*uma breve história do tempo\*/

(5, 'Karl Marx', '1818-05-05'),/\* O capital\*/

(6, 'Candido', '1976-06-06') ;

/\* Inserindo informações na tabela livro\*/

insert into Livro(ISBN, editora, numero\_da\_edicao, titulo, classificacao\_por\_assunto)

values(1, 'Pearson' , 4, 'Sistemas Operativos Modernos', 'tecnologia'),

(2, 'olhares', 4, 'Psicologia das cores', 'designer' ),

(3, 'objetiva', 2, 'O poder do hábito', 'psicologia'),

(4, 'intrinsace', 13, 'Uma Breve História do Tempo', 'física'),

(5, 'Veneta', null, 'O capital','sociologia' ),

(6, 'Pearson', 1, 'O senhor dos aneis', 'Fantasia'),

(7, 'Rocco', 1, 'Harry Potter e a preda filosonal', 'Fantasia'),

(8, 'Veneta', 5 , 'Jogos Vorazes', 'Ficcao'),

(9, 'Rocco', 4, 'O pequeno Principe', 'Fantasia'),

(10, 'Presença', 1,'Cartas de amor aos mortos', 'Tecnologia');

/\* Inserindo informações na tabela Exemplar\*/

insert into Exemplar(cod\_exemplar, quantidade, ISBN\_livro)

values(02946, 20, 1), /\*Psicologia das cores\*/

(01298, 10, 1),/\*Sistemas Operativos Modernos\*/

(68460, 40, 2),/\*O poder do hábito\*/

(47420, 21, 3), /\*Uma breve história do tempo\*/

(07213, 13, 4);/\*O capital\*/

/\* Inserindo informações na tabela Escrito\_por\*/

insert into Escrito\_por(cod\_autor, ISBN\_livro)

values(1, 1), /\*Psicologia das cores\*/

(2, 2),/\*Sistemas Operativos Modernos\*/

(3, 3),/\*O poder do hábito\*/

(4, 4),/\*Uma breve história do tempo\*/

(5, 5),/\*O capital\*/

(5, 6),

(5, 7),

(2, 8),

(3, 9),

(1, 10);

/\* O comando abaixo exemplifica o procedimento para a inserção de imagens

rodar esse comando: select @@GLOBAL.secure\_file\_priv;

substituir 'nome\_do\_arquivo' pelo resultado do comando anterior: select load\_file('/var/lib/mysql-files/{nome\_do\_arquivo}');

\*/

/\* Inserindo informações na tabela Usuario \*/

insert into Usuario(matricula, nome, data\_de\_nascimento, rua, UF, CEP, cidade, bairro, numero, foto, data\_de\_validade, qr\_code)

values(2022001, 'Vanessa Silva', '2000-10-31', 'Avenida Coremas','PB', '58013-430', 'João Pessoa','Centro','561' , load\_file('/var/lib/mysql-files/imagem.png'), '2024-12-31', load\_file('/var/lib/mysql-files/imagem.png')),

(2022002, 'Rita Clara', '2001-11-06', 'Rua Comerciário Antônio Manoel de Sousa','PB', '58071-585', 'João Pessoa','Cristo Redentor','20' , load\_file('/var/lib/mysql-files/imagem.png'), '2022-12-31', load\_file('/var/lib/mysql-files/imagem.png')),

(2022003, 'Laís Epifanio Machado', '2002-10-10', 'Rua Osvaldo Travassos Campos','PB', '58080-540', 'João Pessoa','Ernani Sátiro','16' , load\_file('/var/lib/mysql-files/imagem.png'), '2023-12-31', load\_file('/var/lib/mysql-files/imagem.png')),

(2022004, 'Cleiton Bernadino', '2001-11-08', 'Conjunto Jacinto Medeiros','PB', '58026-080', 'João Pessoa','Treze de Maio','120' , load\_file('/var/lib/mysql-files/imagem.png'), '2022-12-31', load\_file('/var/lib/mysql-files/imagem.png')),

(2022005, 'Raimundo de Moraes', '2004-03-23', 'Rua Guadalupe','PB', '58079-806', 'João Pessoa','Grotão', '777', load\_file('/var/lib/mysql-files/imagem.png'), '2024-12-31', load\_file('/var/lib/mysql-files/imagem.png'));

/\*Inserindo informações na tabela email\*/

insert into Email(email, matricula\_usuario)

values('Silva.Vanessa@academico.ifpb.edu.br', 2022001),

('Silva.Rita@academico.ifpb.edu.br', 2022002),

('Lais.Machado@academico.ifpb.edu.br', 2022003),

('Cleiton.Bernadito@academico.ifpb.edu.br', 2022004),

('Mores.Santos@academico.ifpb.edu.br', 2022005);

/\* Inserindo informações na tabela telefone\*/

insert into Telefone(telefone, matricula\_usuario)

values('98854-1234', 2022001),

('94002-8922', 2022002),

('92034-9933', 2022003),

('91234-5678', 2022004),

('98765-4321', 2022005);

/\* Insert de funcionario \*/

insert into Funcionario(cpf, nome, telefone, data\_de\_nascimento, rua, uf, cep, cidade, bairro,numero, login\_atendente, senha\_atendente, tipo, cpf\_responsavel\_cadastro)

value('10274829351','Caetano Veloso', '98867-1235', '1975-10-05', 'Avenida Coremas','PB', '58013-430', 'João Pessoa','Centro','565' , null, null, null, null),

('15624829361','Alcione', '99357-1251', '1955-11-13', 'Rua Guadalupe','PB', '58079-806', 'João Pessoa','Grotão', '777', null, null, null, '10274829351'),

('12924862965','Renata Costa', '98887-1235', '1975-03-05', 'Rua Comerciário Antônio Manoel de Sousa','PB', '58071-585', 'João Pessoa','Cristo Redentor','18', null, null, null, '10274829351'),

('10252049367','Catarina Rios', '98868-3215', '1985-10-10', 'Conjunto Jacinto Medeiros','PB', '58026-080', 'João Pessoa','Treze de Maio','122', null, null, null, '10274829351'),

('41274829954','Felipe Neto', '98867-1235', '1984-01-11', 'Rua Osvaldo camara Travassos Campos','PB', '58080-540', 'João Pessoa','Ernani Sátiro','19', null, null, null, '10274829351'),

('41274829953','Eliana', '98867-1235', '1984-06-11', 'Rua da areia Travassos Campos','PB', '58080-540', 'João Pessoa','Ernani Sátiro','193', null, null, null, '10274829351'),

('41274829951','Edemberg', '98867-1235', '1994-05-11', 'Rua Osvaldo Travassos Campos','PE', '58080-510', 'Recife','Boa Viagem','21', 'edemberg', '123456', 'atendente', '10274829351');

/\*Inserindo informações na tabela empréstimo\*/

insert into Emprestimo(matricula\_usuario, cpf\_funcionario, cod\_exemplar,data\_de\_emprestimo, data\_de\_devolucao)

value(2022001, null,02946, '2022-11-20', '2022-12-21'), /\* Vanessa fez um empréstimo livro "Psicologia das cores"\*/

(2022002, '15624829361' ,02946, '2022-11-30', '2022-12-30'), /\* Rita fez um empréstimo livro "Psicologia das cores"\*/

(2022001, '15624829361' ,47420, '2022-11-06', '2022-12-25'), /\* Laís fez um empréstimo livro "Uma breve história do tempo"\*/

(2022003, '10274829351' ,07213, '2022-11-20', '2022-12-26'), /\* Cleiton fez um empréstimo livro "O capital"\*/

(2022004, '10274829351' ,01298, '2022-11-28', '2022-12-28'); /\* Raimundo fez um empréstimo livro "Sistemas Operativos Modernos"\*/

/\* selecionando todos os emprestimos que nao tiveram um funcionário envolvido \*/

select \*

from Emprestimo

where cpf\_funcionario is null;

/\* selecionando todos os exemplares cujo a quantidade está entre 11 e 30 \*/

select \*

from Exemplar

where quantidade between 11 and 30;

/\* selecionando a quantidade de funcionarios por cep, agrupando-os \*/

select cep, count(\*) as quantidade

from Funcionario

group by cep;

/\* selecionando o Funcionario mais velho dentre os que possuem cep iniciando com 58080 \*/

select min(data\_de\_nascimento)

from Funcionario

where cep like '58080%';

/\* selecionando todos os autores e ordenando do mais novo pro mais velho pela data de nascimento \*/

select \*

from Autor

order by data\_de\_nascimento desc;

/\* selecionando os livros que possuem edição diferente de 1 e 4 \*/

select \*

from Livro

where numero\_da\_edicao not in (1,4);

/\* Selecionando os autores que possuem mais que um registro de livro escrito\*/

select A.nome as autor

from Autor A

join Escrito\_por E

on A.cod\_autor = E.cod\_autor

group by A.nome

Having count(A.nome) > 1;