PROGRAMAÇÃO E DESENVOLVIMENTO DE BANCO DE DADOS
CURSO DE ENGENHARIA DE SOFTWARE

RELATÓRIO AULA PRÁTICA – CRIAÇÃO DE BANCO DE DADOS UTILIZANDO A LINGUAGEM SQL COM OPERAÇÕES DE MANIPULAÇÃO E ACESSO AOS DADOS.

NELSON PINHEIRO DE CARVALHO

MOGI DAS CRUZES
2023

SUMÁRIO

1 – Introdução

Criação da estrutura de um banco de dados, (tabelas), com a linguagem SQL por meio de um diagrama entidade-relacionamento pré-definido, inserindo dados no banco de dados criado e possibilitando a consulta dos dados armazenados por meio da criação de uma visão (View)

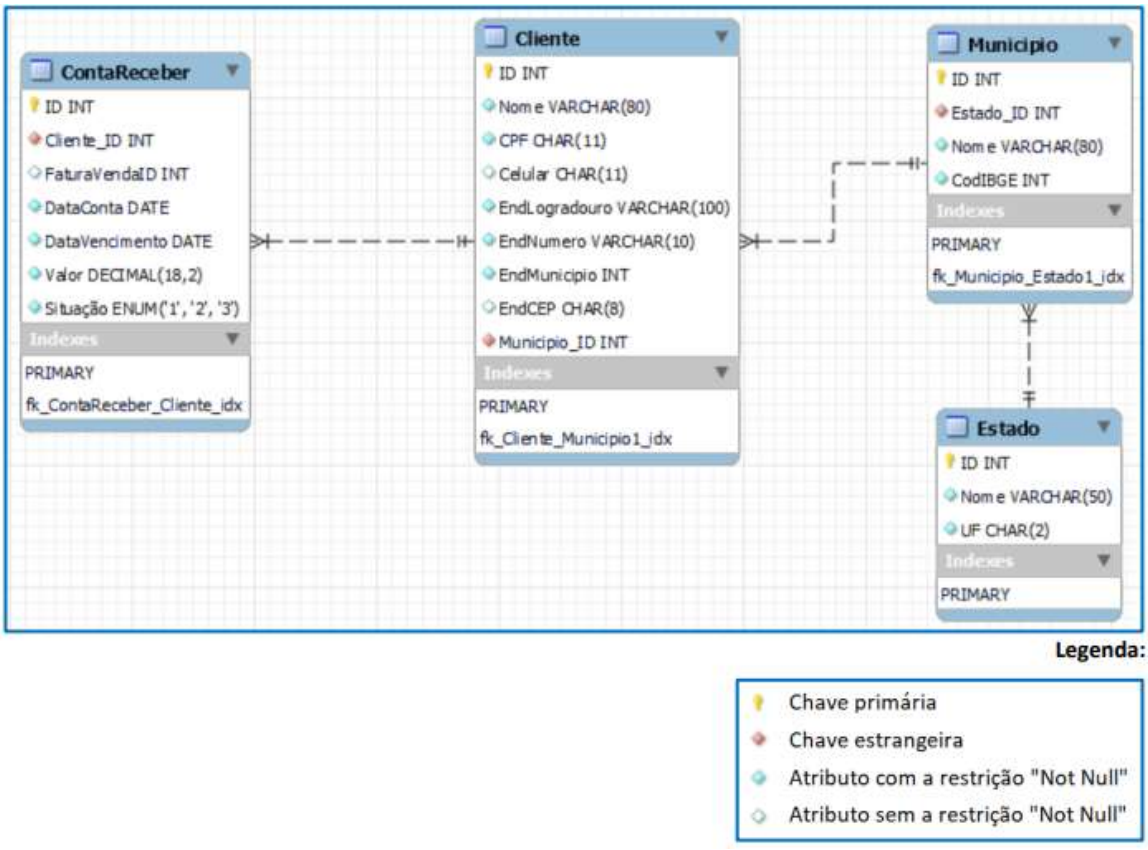
2 – Métodos

Foi utilizado o programa o MySQL Workbench (MySQL Community Server), disponibilizado de maneira gratuita no endereço: https://www.mysql.com/downloads/

3. Resultados

O trabalho foi feito obedecendo os seguintes requisitos:

- Criar um banco de dados de nome “Loja” utilizando o MySQL Server através do software MySQL Workbench.
- Adicionar todas as estruturas de dados no banco de dados criado, utilizando comandos de definição de dados, (DDL), da linguagem SQL, respeitando o modelo DER previamente definido:



- O banco de dados foi criado e foi utilizado um script para a criação das tabelas e relacionamentos, (apêndice A).
  Na criação do banco de dados foram respeitadas as seguintes regras:
  - As chaves primárias devem ser colocadas todas como autoincremento.
  - Respeite os relacionamentos, tipos, precisões e restrições de não nulo.
  - O campo “Situação” da tabela “ContaReceber” deve ser do tipo ENUM e possuir apenas os valores 1, 2 ou 3, sendo 1 – Conta registrada, 2 – Conta cancelada, 3 – Conta paga.
- Na segunda etapa foi criado um script chamado “inserir.sql”, (apêndice B), contento os comandos de manipulação (DML), com o objetivo de popular todas as tabelas existentes na base de dados com pelo menos três registros por tabela.
- Na terceira etapa, por meio dos comandos de consulta (DQL) da linguagem SQL, foi elaborado um script chamado “consulta.sql”, (apêndice C), responsável por criar uma VIEW que retorna todas as contas que ainda não foram pagas, contendo informações como ID da conta a receber, nome e CPF do cliente associado à conta, data de vencimento da conta e o valor.

4. Conclusão

O banco de dados foi criado com sucesso, respeitando todas as diretrizes. Também foram criados os scripts de inserção de dados, (apêndice B), e da criação e consulta da VIEW, (apêndice C).

5. Apêndices

APÊNDICE A – CRIAÇÃO DE TABELAS

```
Create table if not exists Estado (
Id INT auto_increment primary key,
Nome VARCHAR(50) not null unique,
UF CHAR(2) not null unique
);

CREATE TABLE if not exists Municipio (
Id INT auto_increment primary key,
Estado_Id INT,
Nome VARCHAR(80) not null,
CodIBGE INT not null unique,
constraint fk_Id_Estado
foreign key (Estado_Id) references estado (Id)
);

CREATE TABLE Cliente (
Id INT auto_increment PRIMARY KEY,
Nome VARCHAR(80) not null,
CPF CHAR(11) not null unique,
Celular CHAR(11) not null unique,
EndLogradouro VARCHAR(100) not null,
EndNumero VARCHAR(10) not null,
EndMunicipio INT not null,
EndCEP CHAR(8) not null,
Municipio_Id INT not null,
constraint fk_Id_Municipio
foreign key (Municipio_Id) references municipio (Id)
);

create table if not exists ContaReceber (
Id INT auto_increment PRIMARY KEY,
Cliente_ID INT not null,
FaturaVendaId INT not null unique,
DataConta DATE not null,
DataVencimento DATE not null,
Valor DECIMAL(18,2) not null,
Situação ENUM('1','2','3') not null,
constraint fk_Id_Cliente
foreign key (Cliente_ID)
references Cliente (Id)
);
```

APÊNDICE B – INSERIR.SQL

```
insert ignore into estado(Nome, UF)
values
('São Paulo', 'SP'),
('Rio de Janeiro', 'RJ'),
('Espirito Santo', 'ES');

insert ignore into municipio(Estado_Id, Nome, CodIBGE)
values
('1', 'Mogi das Cruzes', '3530607'),
('1', 'Suzano', '3552502'),
('1', 'Arujá', '3503901'),
('2', 'Niterói', '3303302'),
('2', 'Resende', '3304201'),
('2', 'Volta Redonda', '3306305'),
('3', 'Vitória', '3205309'),
('3', 'Guarapari', '3202405'),
('3', 'Linhares', '3203205');

insert ignore into cliente(
Nome, CPF, Celular, EndLogradouro, EndNumero, EndMunicipio, EndCEP, Municipio_Id)
values
('Guilherme', '11111111111', '99999-9999', 'Rua hum', '100', '1', '08888000', '1'),
('Gabriel', '22222222222', '88888-8888', 'Rua dois', '200', '2', '09999000', '2'),
('Nelson', '33333333333', '77777-7777', 'Rua três', '300', '3', '07777000', '3'),
('Tatiane', '44444444444', '22222-2222', 'Rua quatro', '400', '3', '06666000', '3'),
('Débora', '55555555555', '33333-3333', 'Rua cinco', '500', '2', '05555000', '2'),
('Vilma', '66666666666', '44444-4444', 'Rua seis', '600', '2', '07777000', '1');

insert ignore into contareceber(
Cliente_ID, FaturaVendaId, DataConta, DataVencimento, Valor, Situação)
values
('1', '1001', '2023-05-20', '2023-06-20', '100.00', '1'),
('2', '1002', '2023-05-21', '2023-06-21', '200.00', '2'),
('3', '1003', '2023-05-22', '2023-06-22', '300.00', '3'),
('4', '1004', '2023-05-23', '2023-06-23', '400.00', '1'),
('5', '1005', '2023-05-24', '2023-06-24', '500.00', '2'),
('6', '1006', '2023-05-25', '2023-06-25', '600.00', '1');
```

APÊNDICE C – CONSULTA.SQL

```
CREATE OR REPLACE view v_SemPagar as
SELECT CR.Id as 'CODIGO',
        C.Nome as 'NOME DO CLIENTE',
        C.CPF,
   date_format(CR.DataVencimento, '%d/%m/%Y') as 'Data do Vencimento',
        CR.Valor
FROM contareceber CR
JOIN cliente C ON CR.Cliente_ID = C.Id
WHERE CR.Situação = '1';

select * from v_SemPagar;
```

Mogi das Cruzes, 31 de maio de 2023

NELSON PINHEIRO DE CARVALHO