**BANCO DE DADOS NÃO RELACIONAL**

**DSM 3° SEMESTRE**

Giuliana M. Gralha

Larissa Sousa

Silvana Menezes

**E-COMMERCE – CRIAÇÃO DE BANCO DE DADOS PARA IMPLEMENTAÇÃO DO BI**

São Paulo

2023

**Sumário**

[**1.** **Sobre o projeto** 3](#_Toc146923181)

[**2.** **Objetivo** 4](#_Toc146923182)

[**3.** **MER** 5](#_Toc146923183)

[**4.** **Quantidade de Registros em cada Tabela** 6](#_Toc146923184)

[**5.** **Script MySQL** 7](#_Toc146923185)

[**6.** **Referências Bibliográficas** 14](#_Toc146923186)

1. **Sobre o projeto**

Nós da ManasCodes fomos contratadas para estruturar a base de dados para a empresa Venda Feliz, na plataforma e-Commerce, com pretensão de aumentar suas vendas e implantar uma solução de BI (Business Intelligence), para gerar relatórios de tomada de decisão

1. **Objetivo**

Criar o Script de criação do Banco de Dados da empresa VendasFeliz e utilizar a ferramenta web mockaroo, para geração de dados e populacionar o banco, auxiliando futuramente na implantação de BI (Business Intelligence) utilizando a ferramenta Power BI após estruturar o banco de dados para gerar relatórios de tomada de decisão moldando tais informações extraídas com a intenção de aumentar as vendas da empresa VendasFeliz.

1. **MER**

Diagrama, Esquemático

Descrição gerada automaticamente

1. **Quantidade de Registros em cada Tabela**

|  |  |
| --- | --- |
| **Nome da Tabela** | **Quantidade de Registro** |
| Cliente | 500 |
| Transportadora | 500 |
| Produtos | 200 |
| Nota Fiscal | 500 |
| Fornecedores | 100 |
| FormasPagamentos | 500 |
| Sessao | 1000 |
| Avaliacao | 500 |
| Entregas | 500 |
| Pedidos | 500 |
| Pedidos | 500 |
| Garantia | 100 |
| EstoqueFornecedor | 200 |
| DadosPedidos | 500 |
| ConsolidaNegocio | 500 |
| Pagamentos | 500 |

1. **Script MySQL**

CREATE DATABASE vendasFeliz;

USE vendasFeliz;

CREATE TABLE Fornecedores (

cnpj VARCHAR(20) NOT NULL ,

razaoSocial VARCHAR(35) NOT NULL ,

nomeFantasia VARCHAR(20) NULL ,

categoria VARCHAR(35) NULL ,

telefone VARCHAR(18) NULL ,

email VARCHAR(45) NULL ,

endereco VARCHAR(30) NULL ,

numero VARCHAR(10) NULL ,

cep VARCHAR(12) NULL ,

cidade VARCHAR(30) NULL ,

estado VARCHAR(30) NULL ,

PRIMARY KEY(cnpj));

CREATE TABLE Transportadoras (

cnpj VARCHAR(20) NOT NULL ,

nome VARCHAR(30) NULL ,

email VARCHAR(45) NULL ,

telefone VARCHAR(18) NULL ,

tipoDeEntrega VARCHAR(20) NULL,

PRIMARY KEY(cnpj));

CREATE TABLE Produtos (

idProduto INTEGER UNSIGNED NOT NULL AUTO\_INCREMENT,

nome VARCHAR(60) NOT NULL,

qtdEstoque INT NULL ,

detalheProduto TEXT NULL ,

PRIMARY KEY(idProduto));

CREATE TABLE Pedidos (

idPedidos INTEGER UNSIGNED NOT NULL ,

Produtos\_idProduto INTEGER UNSIGNED NOT NULL ,

dataPedido DATETIME NULL ,

enderecoEntrega VARCHAR(30) NULL ,

valorTotal DECIMAL(10,2) NULL ,

desconto DECIMAL(10,2) NULL,

valorTotalComDesconto DECIMAL(10,2) NULL ,

statusPedido VARCHAR(12) NULL,

PRIMARY KEY(idPedidos),

foreign key(Produtos\_idProduto)

REFERENCES Produtos(idProduto));

CREATE TABLE Clientes (

idCliente INTEGER UNSIGNED NOT NULL AUTO\_INCREMENT,

nome VARCHAR(15) NULL ,

sobrenome VARCHAR(25) NULL ,

email VARCHAR(45) NULL ,

genero VARCHAR(1) NULL ,

telefone VARCHAR(18) NULL ,

endereco VARCHAR(30) NULL ,

numero VARCHAR(10) NULL ,

complemento VARCHAR(12) NULL ,

cep VARCHAR(12) NULL ,

cidade VARCHAR(30) NULL ,

estado VARCHAR(30) NULL ,

zona varchar(10) NULL,

dataDoCadastro DATETIME NULL,

PRIMARY KEY(idCliente));

CREATE TABLE Garantia (

id INTEGER UNSIGNED NOT NULL AUTO\_INCREMENT,

fornecedores\_cnpj VARCHAR(20) NOT NULL ,

inicio DATETIME NULL ,

final DATETIME NULL ,

condicoes TEXT NULL ,

PRIMARY KEY(id),

FOREIGN KEY(fornecedores\_cnpj)

REFERENCES Fornecedores(cnpj));

CREATE TABLE Sessao (

id INTEGER UNSIGNED NOT NULL AUTO\_INCREMENT,

Clientes\_idCliente INTEGER UNSIGNED NOT NULL ,

dispositivoUsado VARCHAR(10) NULL ,

inicioSessao DATETIME null,

PRIMARY KEY(id),

FOREIGN KEY(Clientes\_idCliente)

REFERENCES Clientes(idCliente));

CREATE TABLE Entregas (

idEntregas INTEGER UNSIGNED NOT NULL AUTO\_INCREMENT,

Transportadoras\_cnpj VARCHAR(20) NOT NULL ,

numeroRastreio VARCHAR(20) NULL ,

frete DECIMAL(10,2) NULL ,

PRIMARY KEY(idEntregas),

FOREIGN KEY(Transportadoras\_cnpj)

REFERENCES Transportadoras(cnpj));

CREATE TABLE Avaliacoes (

Clientes\_idCliente INTEGER UNSIGNED NOT NULL ,

comentario TEXT NULL ,

estrelas INT NULL ,

FOREIGN KEY(Clientes\_idCliente)

REFERENCES Clientes(idCliente));

CREATE TABLE EstoqueFornecedor (

Fornecedores\_cnpj VARCHAR(20) NOT NULL ,

Produtos\_idProduto INTEGER UNSIGNED NOT NULL ,

FOREIGN KEY(Fornecedores\_cnpj)

REFERENCES Fornecedores(cnpj),

FOREIGN KEY(Produtos\_idProduto)

REFERENCES Produtos(idProduto));

CREATE TABLE DadosPedidos (

Clientes\_idCliente INTEGER UNSIGNED NOT NULL ,

Pedidos\_idPedidos INTEGER UNSIGNED NOT NULL ,

PRIMARY KEY(Clientes\_idCliente, Pedidos\_idPedidos),

FOREIGN KEY(Clientes\_idCliente)

REFERENCES Clientes(idCliente),

FOREIGN KEY(Pedidos\_idPedidos)

REFERENCES Pedidos(idPedidos));

CREATE TABLE NotaFiscal (

idNotaFiscal INTEGER UNSIGNED NOT NULL,

dataNota datetime,

valorNota decimal(10,2),

PRIMARY KEY(idNotaFiscal));

CREATE TABLE FormasPagamentos (

id INTEGER UNSIGNED NOT NULL AUTO\_INCREMENT,

pix VARCHAR(10),

ted VARCHAR(10),

numeroCartao VARCHAR(20) NULL ,

dataVencimento DATE NULL ,

nomeTitular VARCHAR(20) NULL ,

codigoSeguranca VARCHAR(3) NULL ,

qtdDeParcelas INT NULL ,

juros decimal(10,2) NULL ,

bancoEmissor INT NULL ,

valorParcela numeric(10,2),

PRIMARY KEY(id));

CREATE TABLE Pagamentos (

id INTEGER UNSIGNED NOT NULL AUTO\_INCREMENT,

Pedidos\_idPedidos INTEGER UNSIGNED NOT NULL ,

Entregas\_idEntregas INTEGER UNSIGNED NOT NULL ,

NotaFiscal\_idNotaFiscal INTEGER UNSIGNED NOT NULL ,

FormasPagamentos\_id INTEGER UNSIGNED NOT NULL ,

valorPago DECIMAL(10,2) NULL ,

dataPagto DATETIME NULL ,

PRIMARY KEY(id),

FOREIGN KEY(Pedidos\_idPedidos)

REFERENCES Pedidos(idPedidos),

FOREIGN KEY(Entregas\_idEntregas)

REFERENCES Entregas(idEntregas),

FOREIGN KEY(NotaFiscal\_idNotaFiscal)

REFERENCES NotaFiscal(idNotaFiscal),

FOREIGN KEY(FormasPagamentos\_id)

REFERENCES FormasPagamentos(id));

CREATE TABLE ConsolidaNegocio (

idConsolidaNegocio INTEGER UNSIGNED NOT NULL AUTO\_INCREMENT,

Entregas\_idEntregas INTEGER UNSIGNED NOT NULL ,

Pagamentos\_id INTEGER UNSIGNED NOT NULL ,

Pedidos\_idPedidos INTEGER UNSIGNED NOT NULL ,

data DATETIME NULL,

PRIMARY KEY(idConsolidaNegocio),

FOREIGN KEY(Entregas\_idEntregas)

REFERENCES Entregas(idEntregas),

FOREIGN KEY(Pagamentos\_id)

REFERENCES Pagamentos(id),

FOREIGN KEY(Pedidos\_idPedidos)

REFERENCES Pedidos(idPedidos));

Para ter acesso ao Script inteiro e os Inserts [clique aqui.](https://github.com/Giuliana09/ProjetoBI/blob/main/BDN/Script_do_MER/V4%20-%20ScriptVedasFeliz.sql)

1. **Referências Bibliográficas**

DEVPLENO. Mockaroo - Gerando SQL/CSV de testes :). YouTube, 2017.

Disponível em:

<https://www.youtube.com/watch?v=9bjbKIGMG9s>. Acesso em: 06 de Setembro de 2023

MOCKAROO. How to generate datasets related by a foreign key using. YouTube, 2021. Disponível em:

<https://www.youtube.com/watch?v=S_oYFGhZSkQ>. Acesso em: 06 de Setembro de 2023

CODERS4FUTURE. Aprenda a usar o mockaroo, a melhor ferramenta para gerar dados. YouTube, 2022. Disponível em:

<https://www.youtube.com/watch?v=9bjbKIGMG9s>. Acesso em: 06 de Setembro de 2023