

**MODELAGEM DE UMA BASE DE DADOS  
PARA A EMPRESA “BonifácioTec”**

*Por: Alegria Bonifácio  
Ciências da computação | 3º Ano*

## **I. LEVANTAMENTO DE REQUISITOS**

A empresa **BonifácioTec** vende computadores e impressoras. Para isso, ela precisa criar uma Base de Dados para gerir clientes, pedidos, produtos, funcionários e fornecedores.

### **1.1. Requisitos funcionais**

- Cadastrar clientes com seus dados pessoais;
- Registrar funcionários que fazem os atendimentos;
- Gerenciar produtos disponíveis para venda;
- Permitir que os clientes façam pedidos;
- Controlar fornecedores;
- Registrar as compras feitas pela empresa junto aos fornecedores.

## **II. ENTIDADES E SEUS ATRIBUTOS**

Com base nos requisitos, identifiquei as seguintes entidades:

### **1) Cliente:**

- ID\_Cliente (Identificador único)
- Nome
- Email
- Telefone
- Endereço

### **2) Funcionário:**

- ID\_Funcionário
- Nome
- Cargo
- Telefone

### **3) Produto:**

- ID\_Produto
- Nome\_Produto
- Preço
- Categoria (Computador ou Impressora)
- Estoque

#### **4) Pedido:**

- ID\_Pedido (Identificador único)
- Data\_Pedido
- ID\_Cliente (Relacionamento com Cliente)
- ID\_Funcionario (Relacionamento com Funcionário)

#### **5) Produto\_Pedido (Relacionamento entre Pedido e Produto):**

- ID\_Item\_Pedido
- ID\_Pedido
- ID\_Produto
- Quantidade
- Preço

#### **6) Fornecedor:**

- ID\_Fornecedor (Identificador único)
- Nome
- Telefone
- Endereço

#### **7) Produto\_Fornecedor (Relacionamento entre Produto e Fornecedor):**

- ID\_Produto\_Fornecedor
- ID\_Produto (Relacionamento com Produto)
- ID\_Fornecedor (Relacionamento com Fornecedor)
- Preço\_Compra

### **III. NORMALIZAÇÃO**

Fiz a normalização, para evitar redundâncias e inconsistências nos dados.

#### **a) Primeira Forma Normal (1FN):**

- Garanti que todos os atributos sejam atômicos (sem listas ou dados repetitivos dentro de um campo).
- Crie tabelas auxiliares para eliminar repetições. Como por exemplo: Produto\_Pedido, que evita repetição de produtos dentro de um pedido.

**b) Segunda Forma Normal (2FN):**

- Todos os atributos dependem totalmente da chave primária.
- Por exemplo: Na tabela Produto\_Pedido, Quantidade e Preço, dependem apenas de ID\_Pedido e ID\_Produto juntos, e não separadamente.

**c) Terceira Forma Normal (3FN):**

- Removi dependências transitivas.
- Por exemplo: As tabelas Fornecedor e Produto estão separados corretamente, sem depender de outras tabelas desnecessárias.

## **IV. DER - DIAGRAMA ENTIDADE RELACIONAMENTO**

### **1.1. Relacionamentos:**

- Um Cliente pode fazer vários Pedidos (1-N).
- Um Pedido pertence a um Cliente (N1).
- Um Pedido pode ter vários Produtos (N-M) - tabela Produto\_Pedido.
- Um Funcionário pode atender vários Pedidos (1-N).
- Um Fornecedor pode fornecer vários Produtos (1-N).
- Um Produto pode ser comprado de vários Fornecedores (N-M) - tabela Produto\_Fornecedor

## **V. MODELO LÓGICO**

### **5.1. Tabelas:**

```
CREATE TABLE Cliente (  
    ID_Cliente INT PRIMARY KEY AUTO_INCREMENT,  
    Nome VARCHAR(100) NOT NULL,  
    Email VARCHAR(100) UNIQUE NOT NULL,  
    Telefone VARCHAR(15),  
    Endereço TEXT  
);
```

```

CREATE TABLE Funcionario (
  ID_Funcionario INT PRIMARY KEY AUTO_INCREMENT,
  Nome VARCHAR(100) NOT NULL,
  Cargo VARCHAR(50),
  Telefone VARCHAR(15)
);

CREATE TABLE Pedido (
  ID_Pedido INT PRIMARY KEY AUTO_INCREMENT,
  Data_Pedido DATE NOT NULL,
  ID_Cliente INT NOT NULL,
  ID_Funcionario INT NOT NULL,
  FOREIGN KEY (ID_Cliente) REFERENCES Cliente (ID_Cliente),
  FOREIGN KEY (ID_Funcionario) REFERENCES Funcionario(ID_Funcionario)
);

CREATE TABLE Produto (
  ID_Produto INT PRIMARY KEY AUTO_INCREMENT,
  Nome_Produto VARCHAR (100) NOT NULL,
  Preço DECIMAL (10,2) NOT NULL,
  Categoria ENUM ('Computador', 'Impressora') NOT NULL,
  Estoque INT NOT NULL
);

CREATE TABLE Produto_Pedido (
  ID_Item_Pedido INT PRIMARY KEY AUTO_INCREMENT,
  ID_Pedido INT NOT NULL,
  ID_Produto INT NOT NULL,
  Quantidade INT NOT NULL,
  Preço DECIMAL (10,2) NOT NULL,
  FOREIGN KEY (ID_Pedido) REFERENCES Pedido (ID_Pedido),
  FOREIGN KEY (ID_Produto) REFERENCES Produto (ID_Produto)
);

CREATE TABLE Fornecedor (
  ID_Fornecedor INT PRIMARY KEY AUTO_INCREMENT,
  Nome VARCHAR(100) NOT NULL,
  Telefone VARCHAR(15),
  Endereço TEXT
);

```

```
CREATE TABLE Produto_Fornecedor (  
  ID_Produto_Fornecedor INT PRIMARY KEY AUTO_INCREMENT,  
  ID_Produto INT NOT NULL,  
  ID_Fornecedor INT NOT NULL,  
  Preço_Compra DECIMAL (10,2) NOT NULL,  
  FOREIGN KEY (ID_Produto) REFERENCES Produto (ID_Produto),  
  FOREIGN KEY (ID_Fornecedor) REFERENCES Fornecedor (ID_Fornecedor)  
)
```