

# oBoticário

## Modelagem de Banco de Dados Relacional

### Escopo do Documento:

- Motivação
- Modelagem Lógica
- Modelagem Física
- Script
- Considerações
- Sugestões de queries
- Tecnologias utilizadas



## Sumário

Histórico do Documento.....	3
Motivação.....	4
Modelagem Lógica.....	5
Modelagem Física.....	7
Script.....	9
Considerações.....	13
Sugestões de queries.....	14
Tecnologias utilizadas.....	15



## Histórico do Documento

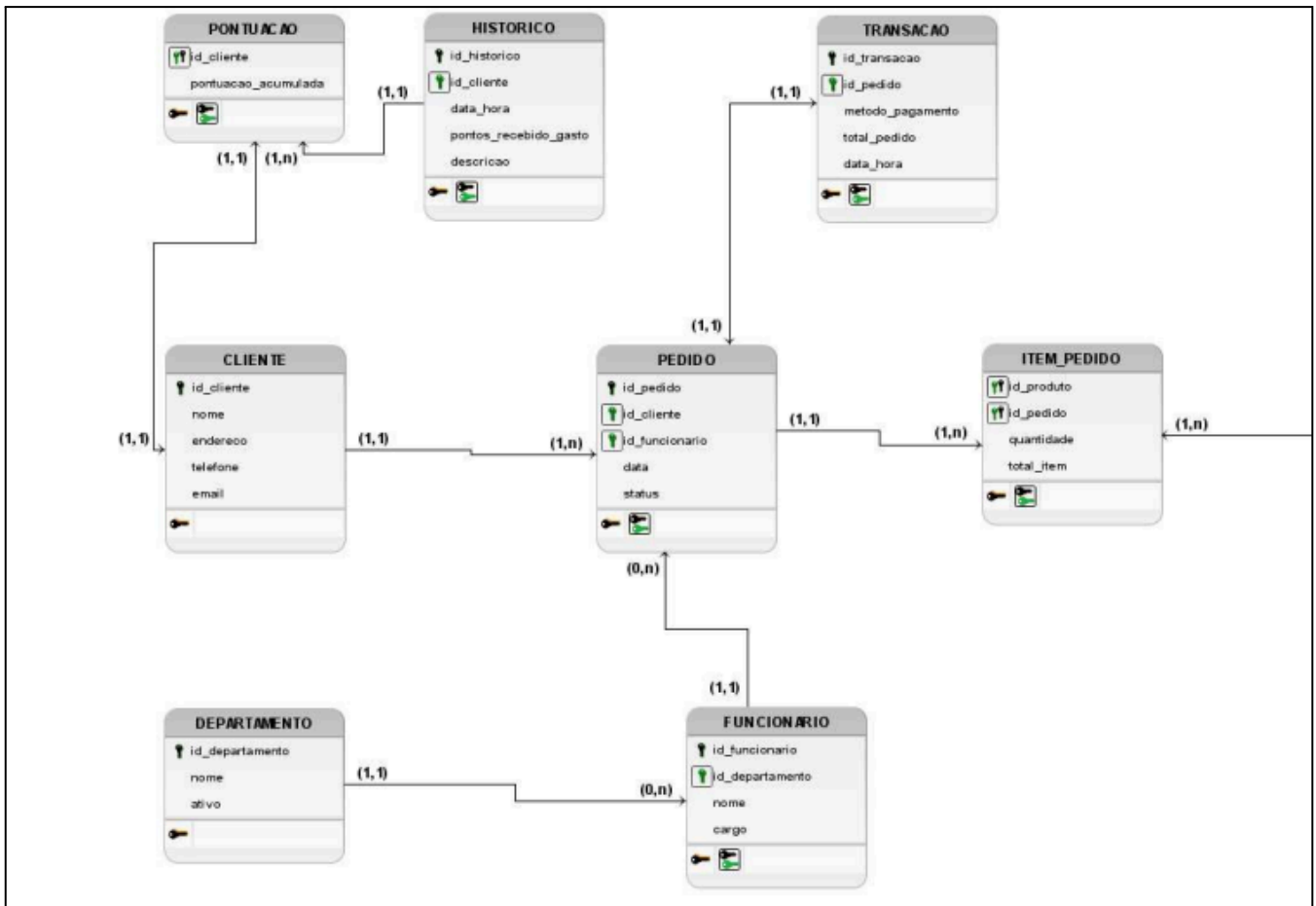
Data	Autor	Nome do Arquivo	Detalhe
25/04/2024	Pedro Lima do Nascimento	Projeto1_Boticário.pdf	Criação do Documento.

## Motivação

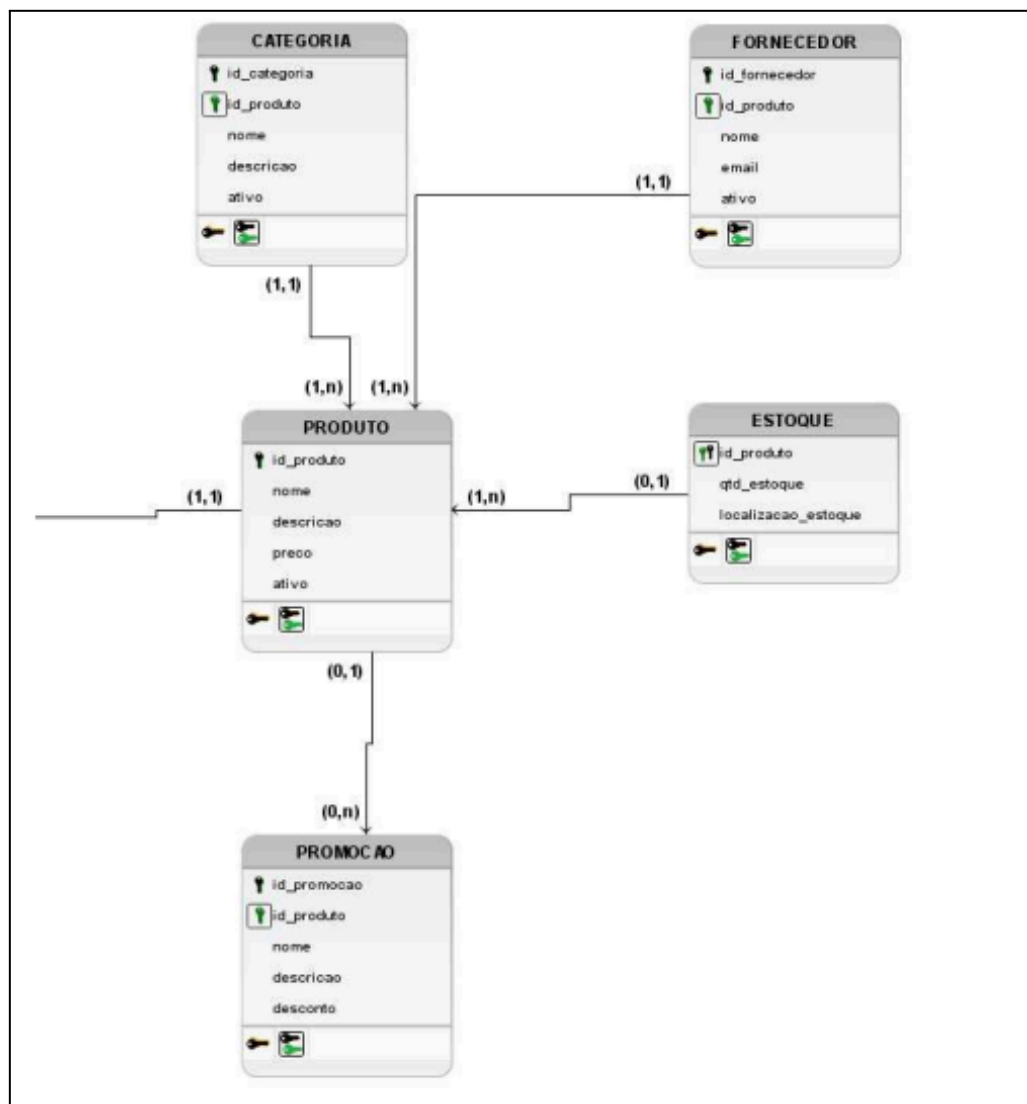
O Boticário é uma empresa reconhecida mundialmente no setor de cosméticos e perfumaria. Com uma vasta gama de produtos, desde fragrâncias até itens de cuidados com a pele e maquiagem, a empresa enfrenta desafios constantes na gestão de vendas, sistema de pontuação e controle de estoque. Este projeto visa criar uma modelagem de banco de dados que atenda às necessidades específicas dessas áreas de negócio, proporcionando uma visão clara das entidades, relacionamentos e atributos envolvidos.

O desafio completo está presente nesse [link](#).

## Modelagem Lógica



Modelo Lógico - Parte 1.

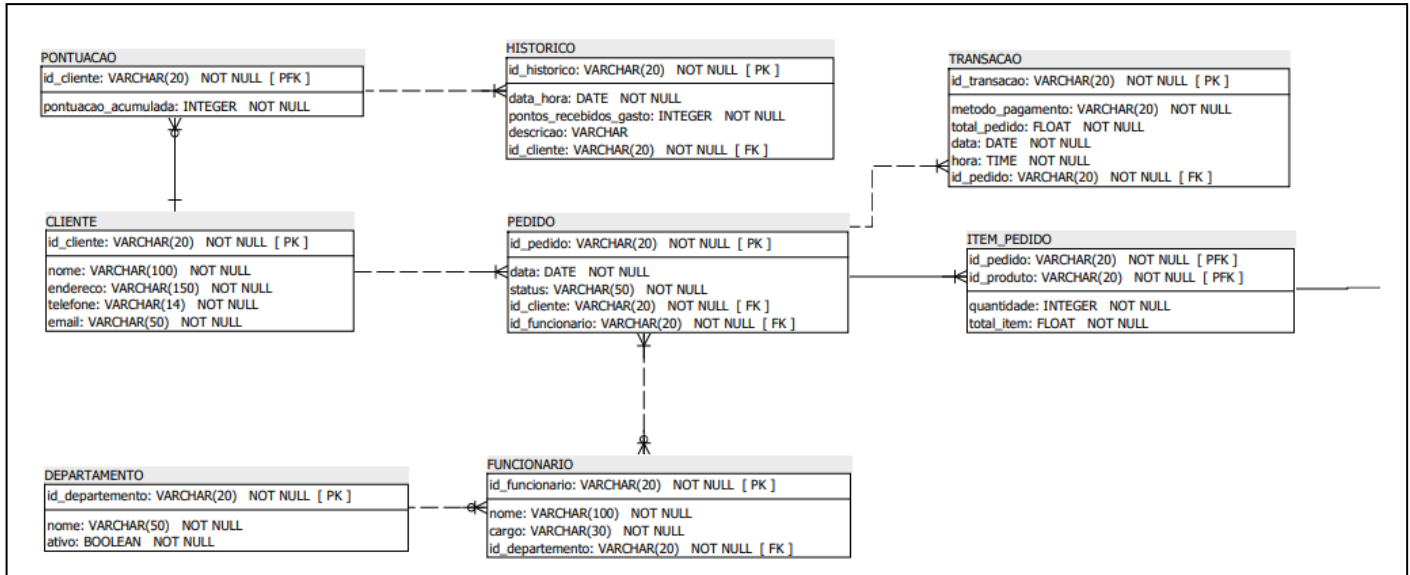


*Modelo Lógico - Parte 2.*

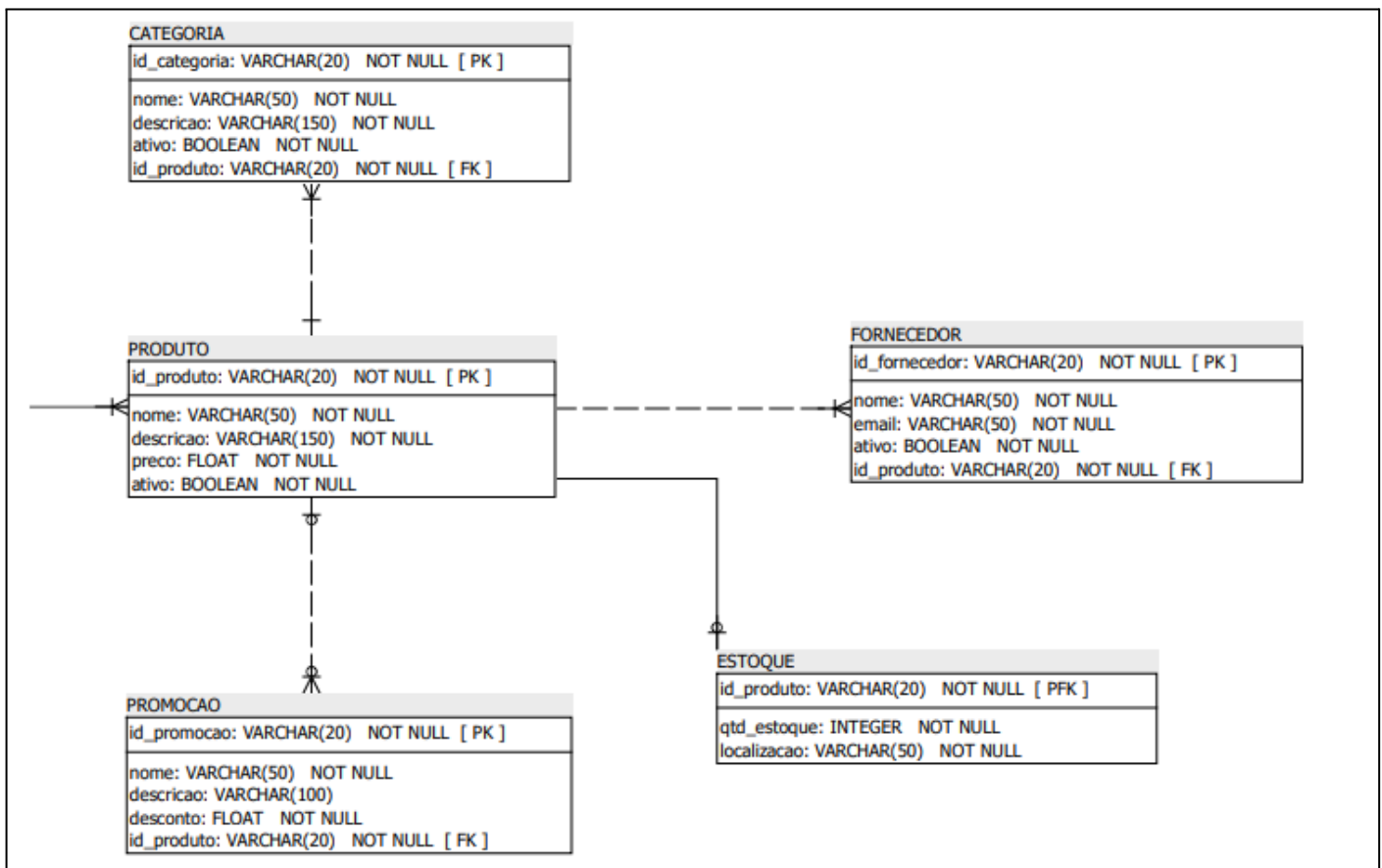
A fim de permitir uma melhor visualização do diagrama o mesmo foi dividido em duas partes. A conexão entre as duas partes da imagem dá-se pelo relacionamento entre as entidades “ITEM\_PEDIDO” e “PEDIDO”.

A visualização do modelo inteiro está disponível através deste [link](#).

## Modelagem Física



Modelagem Física - Parte 1



Modelagem Física - Parte 2

Assim como no capítulo anterior, a imagem relativa à modelagem física foi dividida em duas partes. Da mesma forma, a ligação entre as imagens dá-se pelo relacionamento entre “ITEM\_PEDIDO” e “PEDIDO”.

A visualização do modelo inteiro está disponível através deste [link](#).





## Script

```
1
2 CREATE TABLE Tb_PRODUTO (
3     id_produto VARCHAR(20) NOT NULL,
4     nome VARCHAR(50) NOT NULL,
5     descricao VARCHAR(150) NOT NULL,
6     preco REAL NOT NULL,
7     ativo BOOLEAN DEFAULT TRUE NOT NULL,
8     CONSTRAINT tb_produto_pk PRIMARY KEY (id_produto)
9 );
10
11
12 CREATE TABLE Tb_PROMOCAO (
13     id_promocao VARCHAR(20) NOT NULL,
14     nome VARCHAR(50) NOT NULL,
15     descricao VARCHAR(100),
16     desconto REAL NOT NULL,
17     id_produto VARCHAR(20) NOT NULL,
18     CONSTRAINT tb_promocao_pk PRIMARY KEY (id_promocao)
19 );
20
21
22 CREATE TABLE Tb_CATEGORIA (
23     id_categoria VARCHAR(20) NOT NULL,
24     nome VARCHAR(50) NOT NULL,
25     descricao VARCHAR(150) NOT NULL,
26     ativo BOOLEAN DEFAULT TRUE NOT NULL,
27     id_produto VARCHAR(20) NOT NULL,
28     CONSTRAINT tb_categoria_pk PRIMARY KEY (id_categoria)
29 );
30
31
32 CREATE TABLE Tb_FORNECEDOR (
33     id_fornecedor VARCHAR(20) NOT NULL,
34     nome VARCHAR(50) NOT NULL,
35     email VARCHAR(50) NOT NULL,
36     ativo BOOLEAN DEFAULT TRUE NOT NULL,
37     id_produto VARCHAR(20) NOT NULL,
38     CONSTRAINT tb_fornecedor_pk PRIMARY KEY (id_fornecedor)
39 );
40
41
42 CREATE TABLE Tb_ESTOQUE (
43     id_produto VARCHAR(20) NOT NULL,
44     qtd_estoque INTEGER NOT NULL,
45     localizacao VARCHAR(50) NOT NULL,
46     CONSTRAINT tb_estoque_pk PRIMARY KEY (id_produto)
47 );
48
49
50 CREATE TABLE Tb_DEPARTAMENTO (
51     id_departamento VARCHAR(20) NOT NULL,
52     nome VARCHAR(50) NOT NULL,
53     ativo BOOLEAN DEFAULT TRUE NOT NULL,
54     CONSTRAINT tb_departamento_pk PRIMARY KEY (id_departamento)
55 );
56
57
58 CREATE TABLE Tb_FUNCIONARIO (
59     id_funcionario VARCHAR(20) NOT NULL,
60     nome VARCHAR(100) NOT NULL,
61     cargo VARCHAR(30) NOT NULL,
62     id_departamento VARCHAR(20) NOT NULL,
63     CONSTRAINT tb_funcionario_pk PRIMARY KEY (id_funcionario)
64 );
65
66
67 CREATE TABLE Tb_CLIENTE (
68     id_cliente VARCHAR(20) NOT NULL,
69     nome VARCHAR(100) NOT NULL,
70     endereco VARCHAR(150) NOT NULL,
71     telefone VARCHAR(14) NOT NULL,
72     email VARCHAR(50) NOT NULL,
73     CONSTRAINT tb_cliente_pk PRIMARY KEY (id_cliente)
```

Script de Criação - Parte 1

```

74 );
75
76
77 CREATE TABLE Tb_PEDIDO (
78     id_pedido VARCHAR(20) NOT NULL,
79     data DATE NOT NULL,
80     status VARCHAR(50) NOT NULL,
81     id_cliente VARCHAR(20) NOT NULL,
82     id_funcionario VARCHAR(20) NOT NULL,
83     CONSTRAINT tb_pedido_pk PRIMARY KEY (id_pedido)
84 );
85
86
87 CREATE TABLE Tb_ITEM_PEDIDO (
88     id_pedido VARCHAR(20) NOT NULL,
89     id_produto VARCHAR(20) NOT NULL,
90     quantidade INTEGER NOT NULL,
91     total_item REAL NOT NULL,
92     CONSTRAINT tb_item_pedido_pk PRIMARY KEY (id_pedido, id_produto)
93 );
94
95
96 CREATE TABLE Tb_TRANSACAO (
97     id_transacao VARCHAR(20) NOT NULL,
98     metodo_pagamento VARCHAR(20) NOT NULL,
99     total_pedido REAL NOT NULL,
100     data DATE NOT NULL,
101     hora TIME NOT NULL,
102     id_pedido VARCHAR(20) NOT NULL,
103     CONSTRAINT tb_transacao_pk PRIMARY KEY (id_transacao)
104 );
105
106
107 CREATE TABLE Tb_PONTUACAO (
108     id_cliente VARCHAR(20) NOT NULL,
109     pontuacao_acumulada INTEGER DEFAULT 0 NOT NULL,
110     CONSTRAINT tb_pontuacao_pk PRIMARY KEY (id_cliente)
111 );
112
113
114 CREATE TABLE Tb_HISTORICO (
115     id_historico VARCHAR(20) NOT NULL,
116     data_hora DATE NOT NULL,
117     pontos_recebidos_gasto INTEGER NOT NULL,
118     descricao VARCHAR,
119     id_cliente VARCHAR(20) NOT NULL,
120     CONSTRAINT tb_historico_pk PRIMARY KEY (id_historico)
121 );
122
123
124 ALTER TABLE Tb_ITEM_PEDIDO ADD CONSTRAINT tb_produto_tb_item_pedido_fk
125 FOREIGN KEY (id_produto)
126 REFERENCES Tb_PRODUTO (id_produto)
127 ON DELETE NO ACTION
128 ON UPDATE NO ACTION
129 NOT DEFERRABLE;
130
131 ALTER TABLE Tb_ESTOQUE ADD CONSTRAINT tb_produto_tb_estoque_fk
132 FOREIGN KEY (id_produto)
133 REFERENCES Tb_PRODUTO (id_produto)
134 ON DELETE NO ACTION
135 ON UPDATE NO ACTION
136 NOT DEFERRABLE;
137
138 ALTER TABLE Tb_FORNECEDOR ADD CONSTRAINT tb_produto_tb_fornecedor_fk
139 FOREIGN KEY (id_produto)
140 REFERENCES Tb_PRODUTO (id_produto)
141 ON DELETE NO ACTION
142 ON UPDATE NO ACTION
143 NOT DEFERRABLE;
144
145 ALTER TABLE Tb_CATEGORIA ADD CONSTRAINT tb_produto_tb_categoria_fk
146 FOREIGN KEY (id_produto)

```

## Script de Criação - Parte 2

```
147 REFERENCES Tb_PRODUTO (id_produto)
148 ON DELETE NO ACTION
149 ON UPDATE NO ACTION
150 NOT DEFERRABLE;
151
152 ALTER TABLE Tb_PROMOCAO ADD CONSTRAINT tb_produto_tb_promocao_fk
153 FOREIGN KEY (id_produto)
154 REFERENCES Tb_PRODUTO (id_produto)
155 ON DELETE NO ACTION
156 ON UPDATE NO ACTION
157 NOT DEFERRABLE;
158
159 ALTER TABLE Tb_FUNCIONARIO ADD CONSTRAINT tb_departamento_tb_funcionario_fk
160 FOREIGN KEY (id_departamento)
161 REFERENCES Tb_DEPARTAMENTO (id_departamento)
162 ON DELETE NO ACTION
163 ON UPDATE NO ACTION
164 NOT DEFERRABLE;
165
166 ALTER TABLE Tb_PEDIDO ADD CONSTRAINT tb_funcionario_tb_pedido_fk
167 FOREIGN KEY (id_funcionario)
168 REFERENCES Tb_FUNCIONARIO (id_funcionario)
169 ON DELETE NO ACTION
170 ON UPDATE NO ACTION
171 NOT DEFERRABLE;
172
173 ALTER TABLE Tb_PONTUACAO ADD CONSTRAINT tb_cliente_tb_pontuacao_fk
174 FOREIGN KEY (id_cliente)
175 REFERENCES Tb_CLIENTE (id_cliente)
176 ON DELETE NO ACTION
177 ON UPDATE NO ACTION
178 NOT DEFERRABLE;
179
180 ALTER TABLE Tb_PEDIDO ADD CONSTRAINT tb_cliente_tb_pedido_fk
181 FOREIGN KEY (id_cliente)
182 REFERENCES Tb_CLIENTE (id_cliente)
183 ON DELETE NO ACTION
184 ON UPDATE NO ACTION
185 NOT DEFERRABLE;
186
187 ALTER TABLE Tb_TRANSACAO ADD CONSTRAINT tb_pedido_tb_transacao_fk
188 FOREIGN KEY (id_pedido)
189 REFERENCES Tb_PEDIDO (id_pedido)
190 ON DELETE NO ACTION
191 ON UPDATE NO ACTION
192 NOT DEFERRABLE;
193
194 ALTER TABLE Tb_ITEM_PEDIDO ADD CONSTRAINT tb_pedido_tb_item_pedido_fk
195 FOREIGN KEY (id_pedido)
196 REFERENCES Tb_PEDIDO (id_pedido)
197 ON DELETE NO ACTION
198 ON UPDATE NO ACTION
199 NOT DEFERRABLE;
200
201 ALTER TABLE Tb_HISTORICO ADD CONSTRAINT tb_pontuacao_tb_historico_fk
202 FOREIGN KEY (id_cliente)
203 REFERENCES Tb_PONTUACAO (id_cliente)
204 ON DELETE NO ACTION
205 ON UPDATE NO ACTION
206 NOT DEFERRABLE;
```

*Script de Criação - Parte 3*

Os Scripts de criação das tabelas também foram divididos em 3 imagens, a fim de melhorar a visualização.

O arquivo de criação dos scripts encontra-se disponível no [link](#).

## Considerações

No projeto em questão, é crucial considerar o desempenho do banco de dados ao definir índices e chaves primárias. A escolha cuidadosa das colunas para índices deve priorizar consultas frequentes e operações de manipulação de dados, buscando equilibrar a otimização das consultas com o impacto nas operações de inserção, atualização e exclusão. Ao selecionar chaves primárias, é fundamental escolher colunas que garantam a unicidade e integridade dos registros. Implementar índices e chaves primárias de forma estratégica no modelo físico e lógico pode melhorar significativamente o desempenho global do banco de dados.

No capítulo a seguir, veremos algumas sugestões de queries baseadas no nosso projeto.

## Sugestões de queries

Na tabela 1, vamos sugerir algumas queries que poderiam ser implementadas no nosso projeto, visando o estudo de eficiência e projeção de programação.

Sugestões de Queries	
Consultar todos os produtos ativos	<pre>SELECT * FROM Tb_PRODUTO WHERE ativo = TRUE;</pre>
Consultar todos os pedidos de um cliente específico	<pre>SELECT * FROM Tb_PEDIDO WHERE id_cliente = 'ID_DO_CLIENTE';</pre>
Consultar o total de itens vendidos em cada pedido	<pre>SELECT id_pedido, COUNT(*) AS total_itens FROM Tb_ITEM_PEDIDO GROUP BY id_pedido;</pre>
Consultar a quantidade em estoque de um produto específico	<pre>SELECT qtd_estoque FROM Tb_ESTOQUE WHERE id_produto = 'ID_DO_PRODUTO';</pre>
Consultar todos os funcionários de um departamento específico	<pre>SELECT * FROM Tb_FUNCIONARIO WHERE id_departamento = 'ID_DO_DEPARTAMENTO';</pre>
Consultar o histórico de pontos de um cliente específico	<pre>SELECT * FROM Tb_HISTORICO WHERE id_cliente = 'ID_DO_CLIENTE';</pre>
Consultar todos os produtos de uma categoria específica	<pre>SELECT * FROM Tb_PRODUTO INNER JOIN Tb_CATEGORIA ON Tb_PRODUTO.id_produto = Tb_CATEGORIA.id_produto WHERE Tb_CATEGORIA.id_categoria = 'ID_DA_CATEGORIA';</pre>
Consultar todas as promoções ativas	<pre>SELECT * FROM Tb_PROMOCAO WHERE ativo = TRUE;</pre>
Consultar todos os pedidos feitos em uma data específica	<pre>SELECT * FROM Tb_PEDIDO WHERE data = 'DATA_ESPECIFICA';</pre>
Consultar o total de transações realizadas em um determinado período	<pre>SELECT COUNT(*) AS total_transacoes FROM Tb_TRANSACAO WHERE data BETWEEN 'DATA_INICIAL' AND 'DATA_FINAL';</pre>
Consultar os produtos fornecidos por um determinado fornecedor	<pre>SELECT * FROM Tb_PRODUTO INNER JOIN Tb_FORNECEDOR ON Tb_PRODUTO.id_produto = Tb_FORNECEDOR.id_produto WHERE Tb_FORNECEDOR.id_fornecedor = 'ID_DO_FORNECEDOR';</pre>

Tabela 1 - Sugestões de Queries

## Tecnologias utilizadas

Neste projeto algumas tecnologias foram utilizadas. Tanto para elaboração dos modelos físico e lógico, quanto para este documento. A seguir, uma lista com indicando essas tecnologias:

Br Modelo

SQL Power Architect

Notepad ++

Google Docs

GitHub