

```

1  /*
2  Banco de Dados
3  Prof.ª Mariana Meirelles de Mello
4  Semestre: 2025.2
5  Data: 21/10/2025
6
7  Aula 01: Modelo Físico de Banco de Dados - Primeiros comandos SQL
8 */
9
10 /*
11 BOAS PRÁTICAS!
12     1. Usar identação para deixar o código mais legível e claro.
13         - Uma instrução por linha
14         - Letras maiúsculas para comandos SQL e minúsculas para nomes de tabelas e
15             colunas
16     2. Nomes significativos.
17         - Não utilizar acentos, espaços e caracteres especiais.
18     3. Utilize comentários em trechos importantes e mais complexos.
19         -- comentario de linha
20 */
21 /*
22 Iremos utilizar inicialmente o "MySQL 8.0 Command Line Client - Unicode"
23 Ao utilizar o "MySQL 8.0 Command Line Client - Unicode" devemos prestar muita atenção
nos comandos que estamos escrevendo, pois se tivermos algum erro, teremos que
reescrever todo o comando que desejamos.
24
25 Em SQL para dizer ao nosso SGBD que finalizamos um comando devemos utilizar o ;
(ponto-e-vírgula). Logo, quando terminar um comando não esqueça do ;.
26 */
27
28 /*
29 Quando desejarmos visualizar algum objeto que exista no nosso SGBD devemos utilizar o
comando SHOW, seguido do objeto desejado.
30 */
31
32 -- Visualizar as bases de Dados
33 SHOW databases;
34
35 /* Retorno do comando com as databases do meu banco
36 +-----+
37 | Database      |
38 +-----+
39 | clinica       |
40 | information_schema |
41 | mysql          |
42 | performance_schema |
43 | sakila         |
44 | sys            |
45 | world          |
46 +-----+
47 */
48
49 -- Para utilizar uma base de dados, utilizamos o comando USE.
50 USE clinica;
51
52 -- Para visualizar as tabelas dessa base de dados, utilizamos o comando SHOW visto
anteriormente.
53 SHOW TABLES;
54
55 /* Retorno do comando:
56 +-----+
57 | Tables_in_clinica |
58 +-----+
59 | consulta          |
60 | medico            |
61 | paciente          |
62 +-----+
63 */
64
65 /*
66 Se precisarmos ver a estrutura dos campos de uma tabela podemos utilizar o comando
DESC de describe.

```

```

67 Sintaxe: DESC nome_tabela
68 */
69 DESC consulta;
70
71 /* Retorno do comando:
72
73 +-----+-----+-----+-----+-----+
74 | Field      | Type       | Null | Key | Default | Extra          |
75 +-----+-----+-----+-----+-----+
76 | idConsulta | int        | NO   | PRI | NULL    | auto_increment |
77 | dt_consulta | date       | NO   |     | NULL    |                 |
78 | diagnostico | varchar(200) | YES  |     | NULL    |                 |
79 | fk_idPaciente | int       | NO   | MUL | NULL    |                 |
80 | fk_idMedico | int       | NO   | MUL | NULL    |                 |
81 +-----+-----+-----+-----+-----+
82 */
83
84 /*
85 Agora vamos criar a nossa primeira base de dados de exemplo para treinarmos os
86 primeiros comandos de SQL.
87 Sempre para criar algo, seja uma base de dados, uma tabela, uma view utilizamos o
88 comando: CREATE
89
90 Sintaxe: CREATE <OBJETO> <nome>.
91
92 O nome do nosso primeiro banco será, aula01_noite.
93 */
94
95 CREATE DATABASE aula01_noite;
96
97 -- Ao visualizarmos nossas databases, a que acabamos de criar ja aparecerá no nosso
98 resultado:
99 SHOW DATABASES;
100
101 /* Retorno do comando:
102 +-----+
103 | Database           |
104 +-----+
105 | aula01_noite      |
106 | clinica            |
107 | information_schema |
108 | mysql               |
109 | performance_schema  |
110 | sakila              |
111 | sys                 |
112 | world               |
113 +-----+
114 */
115
116
117 USE aula01_noite;
118
119 -- Vamos criar a nossa primeira tabela!
120 /*
121 Sempre para criar algum objeto utilizamos o comando CREATE. Como queremos criar uma
122 tabela, utilizaremos este comando seguido de "table" e o nome que você deseja para a
123 sua tabela.
124 Sintaxe:
125
126 Estrutura genérica para criação de uma tabela. As restrições não são obrigatórias.
127 CREATE TABLE nome_tabela(
128     atributo1 tipo [restricoes],
129     atributo2 tipo [restricoes],
130     atributoN tipo [restricoes]
131 );
132
133 A nossa primeira tabela será de 'produto' e deve ter os seguintes atributos:
134     - idProduto: chave primária e de controle do SGBD

```

```

133     - nome: campo obrigatório
134     - preço: campo obrigatório
135     - dt_validade
136 */
137
138 CREATE TABLE produto(
139     idProduto INT PRIMARY KEY AUTO_INCREMENT,
140     nome VARCHAR(100) NOT NULL,
141     preço FLOAT(8,2) NOT NULL,
142     dt_validade DATE
143 );
144
145 /*
146 PRIMARY KEY - é a cláusula utilizada para definir um atributo como chave primária
147 NOT NULL - define que aquele atributo não pode receber valor nulo, ou seja, temos que
148 obrigatoriamente definir um valor ao inserir dados na tabela para aquele atributo
149 AUTO_INCREMENT - é a cláusula utilizada para passar o controle do incremento da chave
150 primária para o SGBD.
151 */
152 /*
153 Visualizar as tabelas que existem na nossa base de dados através do comando SHOW.
154 */
155 SHOW TABLES;
156
157 /* Retorno do comando:
+-----+
| Tables_in_aula01_noite |
+-----+
| produto                 |
+-----+
*/
158
159
160
161
162
163
164 -- Visualizar a estrutura da nossa tabela.
165 DESC produto;
166
167
168 /* Retorno do comando:
+-----+-----+-----+-----+-----+-----+
| Field      | Type       | Null | Key  | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| idProduto  | int        | NO   | PRI   | NULL    | auto_increment |
| nome        | varchar(100) | NO   |        | NULL    |                |
| preço       | float(8,2)  | NO   |        | NULL    |                |
| dt_validade | date       | YES  |        | NULL    |                |
+-----+-----+-----+-----+-----+-----+
*/
169
170
171
172
173
174
175
176
177
178 /*
179 Após criarmos uma tabela, pode ser necessário alguma alteração na sua estrutura. Para
180 alterar campos de uma tabela, utilizamos o comando ALTER TABLE juntamente com alguma
181 das cláusulas abaixo:
182
183 CHANGE - permite alterar o nome do campo, tipo e restrições
184 MODIFY - pode fazer tudo o que o CHANGE executa, menos renomear um campo
185 ADD - adiciona um novo campo a tabela já criada
186
187 Vamos realizar algumas alterações na tabela que acabamos de criar.
188 */
189
190 -- Renomear a tabela para 'produto' para 'produtos':
191
192 ALTER TABLE produto
193     RENAME TO produtos;
194
195
196 /*Retorno do comando:
+-----+
| Tables_in_aula01_noite |
+-----+
| produtos                 |
+-----+
*/
197
198
199
200
201

```

```

202
203 -- Observe que agora o nome da nossa tabela é 'produtoS'.
204
205 /*
206 Para alterar o nome de um atributo devemos usar a cláusula CHANGE junto com o ALTER
207 TABLE
208 Sintaxe:
209 ALTER TABLE <nome_da_tabela>
210   CHANGE 'nome_atual' 'novo_nome' <tipo> <restricoes>;
211
212 Alterando o nome do atributo 'preco' para 'valor':
213 */
214
215 ALTER TABLE produtos
216   CHANGE preco valor FLOAT(8,2) NOT NULL;
217
218 DESC produtos;
219
220 /* Retorno do comando:
221 +-----+-----+-----+-----+-----+
222 | Field      | Type       | Null | Key  | Default | Extra          |
223 +-----+-----+-----+-----+-----+
224 | idProduto  | int        | NO   | PRI   | NULL    | auto_increment |
225 | nome        | varchar(100) | NO   |       | NULL    |                 |
226 | valor       | float(8,2)  | NO   |       | NULL    |                 |
227 | dt_validade| date       | YES  |       | NULL    |                 |
228 +-----+-----+-----+-----+-----+
229 */
230
231 /*
232 Alterar o tipo de um atributo.
233 Podemos utilizar o CHANGE ou o MODIFY. Vamos ver a sintaxe com o MODIFY.
234
235 Sintaxe:
236 ALTER TABLE <nome_da_tabela>
237   MODIFY <atributo> <tipo> <restricoes>;
238
239 *Restrições, se houver.
240 Não podemos utilizar o MODIFY para renomear um atributo, apenas o CHANGE.
241
242 Vamos alterar o tipo do atributo 'valor' para 'decimal'
243 */
244
245 ALTER TABLE produtos
246   MODIFY valor DECIMAL(7,2) NOT NULL;
247
248 DESC produtos;
249
250 /* Retorno do comando:
251 +-----+-----+-----+-----+-----+
252 | Field      | Type       | Null | Key  | Default | Extra          |
253 +-----+-----+-----+-----+-----+
254 | idProduto  | int        | NO   | PRI   | NULL    | auto_increment |
255 | nome        | varchar(100) | NO   |       | NULL    |                 |
256 | valor       | decimal(7,2)  | NO   |       | NULL    |                 |
257 | dt_validade| date       | YES  |       | NULL    |                 |
258 +-----+-----+-----+-----+-----+
259 */
260
261 /*
262 Vou deixar aqui o exemplo de como fazer a alteração acima utilizando o CHANGE no
263 lugar do MODIFY.
264 (Não precisa executar, é apenas um exemplo para vocês verem a sintaxe utilizando
265 outra cláusula para executar a mesma tarefa.)
266
267 ALTER TABLE produtos
268   CHANGE valor valor DECIMAL(7,2) NOT NULL;
269 */
270
271
272 -- Até a próxima aula!

```