

```
1 Laboratório de SQL - Parte I: Comandos básicos.
2
3 -----
4 EXEMPLO 1
5 -----
6
7 1) CRIAÇÃO DE TABELAS E CONSULTA DE ESTRUTURA DE TABELAS
8
9 -- >>> Sintaxe MySQL -----
10
11 CREATE TABLE Pessoas
12 (
13     cod_pessoa    INT AUTO_INCREMENT,
14     nome          VARCHAR(40),
15     telefone      VARCHAR(20),
16     identidade    VARCHAR(20),
17     sexo          CHAR(1),
18     rua           VARCHAR(40),
19     numero        VARCHAR(10),
20     bairro        VARCHAR(40),
21     cep           VARCHAR(8),
22     cidade        VARCHAR(40),
23     uf            CHAR(2),
24     PRIMARY KEY(cod_pessoa)
25 );
26
27 SHOW tables;
28
29 DESCRIBE pessoas;
30
31
32 -- >>> Sintaxe POSTGRESQL -----
33
34 CREATE TABLE Pessoas
35 (
36     cod_pessoa    SERIAL PRIMARY KEY,
37     nome          VARCHAR(40),
38     telefone      VARCHAR(20),
39     identidade    VARCHAR(20),
40     sexo          CHAR(1),
41     rua           VARCHAR(40),
42     numero        VARCHAR(10),
43     bairro        VARCHAR(40),
44     cep           VARCHAR(8),
45     cidade        VARCHAR(40),
46     uf            CHAR(2)
47 );
48
49 SELECT table_name FROM information_schema.tables WHERE table_schema = 'public';
50
51 SELECT * FROM INFORMATION_SCHEMA.COLUMNS WHERE table_name = 'pessoas';
52
53
54 2) INSERÇÃO DE DADOS EM UMA TABELA
55
56 INSERT INTO Pessoas (nome, rua, numero, bairro,
57                     cidade, uf, cep, telefone, identidade,
58                     sexo)
59     VALUES ('Jose Francisco de Queiroz', 'Aquarius', '258',
60            'Vila So Jos', 'SJC', 'MG',
61            '12227320', '3911-7707', 'M-X.XXX.XXX', 'M');
62
63 INSERT INTO Pessoas (nome, rua, numero, bairro,
64                     cidade, uf, cep, telefone, identidade,
65                     sexo)
66     VALUES ('Eduardo Silva de Queiroz', 'Albino Sartori',
67            '95', 'Vila Sao Jose', 'Ouro Preto', 'MG',
68            '35400000', '3551-4243', 'M-X.23X.0X4', 'M');
69
70 INSERT INTO Pessoas (nome, rua, numero, bairro,
71                     cidade, uf, cep, telefone, identidade,
```

```
72         sexo)
73     VALUES ('Maria Antonia de Oliveira', '36', '1020',
74             'Centro', 'Ituiutaba', 'SP',
75             '36400000', '3921-1248', 'M-19.239.013', 'F');
76
77
78 INSERT INTO Pessoas (nome, rua, numero, bairro,
79                     cidade, uf, cep, telefone, identidade,
80                     sexo)
81     VALUES ('Marcia Teffolo de Oliveira', 'Pandi Calgeras',
82             '580', 'Cabeas', 'Ouro Preto', 'MT', '35400000',
83             '3551-1807', 'M-9.239.019', 'F');
84
85
86 INSERT INTO Pessoas (nome, rua, numero, bairro,
87                     cidade, uf, cep, telefone, identidade,
88                     sexo)
89     VALUES ('Joaquim Queiroz da Silva', 'Goias', '58',
90             'Centro', 'Santa Vitoria', 'GO',
91             '12327320', '4911-7707', 'M-9.239.000', 'M');
92
93
94 INSERT INTO Pessoas
95     VALUES (20, 'Maria Tereza Melo', '3561-7607', '11220333-6',
96             'M', 'Francisco Jose Longo', '180',
97             'Centro', '12227320', 'SJC', 'SP');
98
99 3) CONSULTAS DIVERSAS
100
101 SELECT * FROM pessoas;
102
103 SELECT nome, cidade FROM pessoas;
104
105 SELECT * FROM pessoas WHERE sexo = 'M';
106
107 SELECT * FROM pessoas WHERE nome LIKE '%Queiroz%';
108
109 SELECT DISTINCT cidade FROM pessoas;
110
111 SELECT * FROM pessoas ORDER BY cidade;
112
113 SELECT * FROM pessoas ORDER BY cidade DESC;
114
115 SELECT * FROM pessoas
116 WHERE cidade IN ('Ouro Preto', 'Ituiutaba')
117
118 > Selecionar só mulheres de Ouro Preto
119 > Selecionar só moradores da Regiao Sudeste
120
121 4) ALTERAÇÃO DA ESTRUTURA DA TABELA
122
123 ALTER TABLE pessoas
124 (
125     ADD COLUMN pais char(2)
126 );
127
128 5) ALTERAÇÃO DOS REGISTROS DE UMA TABELA
129
130 UPDATE pessoas SET pais = "BR";
131
132 UPDATE pessoas SET cidade = 'Sao Jose dos Campos' WHERE cidade = 'SJC';
133
134 6) CRIANDO JUNÇÕES
135
136 -- >>> Sintaxe MySQL -----
137
138 CREATE TABLE pedidos (
139     cod_pedido INT AUTO_INCREMENT,
140     nro_pedido INT NOT NULL,
141     cod_pessoa INT NOT NULL,
142     PRIMARY KEY(cod_pedido));
```

```
143
144 -- >>> Sintaxe PostgreSQL -----
145
146 CREATE TABLE pedidos (
147   cod_pedido SERIAL PRIMARY KEY,
148   nro_pedido INT NOT NULL,
149   cod_pessoa INT NOT NULL);
150
151 INSERT INTO pedidos VALUES
152 (1,77895,3), (2,44678,3),(3,22456,1),(4,24562,1),(5,34764,15);
153
154 SELECT pessoas.nome, pessoas.cidade, pedidos.nro_pedido
155 FROM pessoas
156 INNER JOIN pedidos
157     ON pessoas.cod_pessoa=pedidos.cod_pessoa
158 ORDER BY pessoas.nome;
159
160 SELECT pessoas.nome, pessoas.cidade, pedidos.nro_pedido
161 FROM pessoas
162 LEFT JOIN pedidos
163     ON pessoas.cod_pessoa=pedidos.cod_pessoa
164 ORDER BY pessoas.nome;
165
166 SELECT pessoas.nome, pessoas.cidade, pedidos.nro_pedido
167 FROM pessoas
168 RIGHT JOIN pedidos
169     ON pessoas.cod_pessoa=pedidos.cod_pessoa
170 ORDER BY pessoas.nome;
171
172 6) REMOVENDO REGISTROS
173
174 DELETE FROM pessoas WHERE sexo = 'F';
175
176 DELETE from pessoas;
177
178 -----
179 EXEMPLO 2
180 -----
181
182 -- >>> Sintaxe MySQL -----
183
184 CREATE TABLE compras (
185   item INT(4) UNSIGNED ZEROFILL DEFAULT '0000' NOT NULL,
186   fornecedor CHAR(20)          DEFAULT ''      NOT NULL,
187   preco DOUBLE(16,2)          DEFAULT '0.00' NOT NULL,
188   PRIMARY KEY(item, fornecedor));
189
190 -- >>> Sintaxe MySQL -----
191
192 CREATE TABLE compras (
193   item INTEGER DEFAULT '0000' NOT NULL,
194   fornecedor CHAR(20) DEFAULT '' NOT NULL,
195   preco REAL DEFAULT '0.00' NOT NULL,
196   PRIMARY KEY(item, fornecedor));
197
198 INSERT INTO compras VALUES
199 (1,'Extra',3.45),
200 (1,'Pao de Acucar',3.99),
201 (2,'Extra',10.99),
202 (3,'Pao de Acucar',1.45),
203 (3,'Wall-Mart',1.69),
204 (3,'Makro',1.25),
205 (4,'Makro',19.95);
206
207 - Quantos compras foram feitas?
208 SELECT COUNT(*)
209 FROM compras;
210
211 - Quantos itens diferentes foram comprados?
212 SELECT COUNT(DISTINCT item)
213 FROM compras;
```

```
214
215 - Qual o item com maior código?
216 SELECT MAX(item) AS item
217 FROM compras;
218
219 - Quanto foi gasto em compras?
220 SELECT SUM(preco) AS total FROM compras;
221
222 - Quanto foi pago para cada fornecedor?
223 SELECT fornecedor, SUM(preco)
224 FROM compras
225 GROUP BY fornecedor;
226
227 - Quais fornecedores receberam mais que 10?
228 SELECT fornecedor, SUM(preco)
229 FROM compras
230 GROUP BY fornecedor
231 HAVING SUM(preco)>10;
232
233 - Encontre o número, fornecedor e preço do item mais caro.
234 SELECT item, fornecedor, preco
235 FROM compras
236 WHERE preco=(SELECT MAX(preco)
237             FROM compras);
238
239 - Qual é o maior preço por item?
240 SELECT item, MAX(preco) AS preco
241 FROM vendas
242 GROUP BY item;
243
244 - Para cada item, encontre o(s) fornecedor(s) com o maior preço.
245 SELECT item, fornecedor, preco
246 FROM compras v1
247 WHERE preco=(SELECT MAX(v2.preco)
248             FROM compras v2
249             WHERE v1.item = v2.item);
250
251 - Aumente em 10% o valor das compras acima de 5
252 UPDATE compras
253     SET preco = preco+ preco*0.1
254     WHERE preco > 5;
255
256
257 -----
258 EXEMPLO 3
259 -----
260
261 CREATE TABLE cidades (
262     nome      VARCHAR(80) PRIMARY KEY,
263     x         REAL,
264     y         REAL
265 );
266
267 CREATE TABLE tempo (
268     cidade    VARCHAR(80) REFERENCES cidades(nome),
269     temp_bx   INTEGER,
270     temp_al   INTEGER,
271     prcp      REAL,
272     data      DATE
273 );
274
275 INSERT INTO cidades VALUES ('Sao Jose dos Campos',-23.2, -43.6);
276 INSERT INTO cidades VALUES ('Campos do Jordao',-22.12, -42.5);
277
278 INSERT INTO tempo VALUES ('Sao Jose dos Campos', 13, 21, 0.25, '2010-06-13');
279 INSERT INTO tempo VALUES ('Sao Jose dos Campos', 11, 21, 0.30, '2010-06-14');
280 INSERT INTO tempo VALUES ('Sao Jose dos Campos', 11, 21, 0.27, '2010-06-15');
281 INSERT INTO tempo VALUES ('Sao Jose dos Campos', 13, 21, 0.25, '2010-06-16');
282 INSERT INTO tempo VALUES ('Sao Jose dos Campos', 19, 24, 0.22, '2010-06-17');
283
284 INSERT INTO tempo VALUES ('Campos do Jordao', 4, 11, 0.21, '2010-06-13');
```

```
285 INSERT INTO tempo VALUES ('Campos do Jordao', 0, 9, 0.32, '2010-06-14');
286 INSERT INTO tempo VALUES ('Campos do Jordao', -1, 8, 0.17, '2010-06-15');
287 INSERT INTO tempo VALUES ('Campos do Jordao', 5, 11, 0.15, '2010-06-16');
288 INSERT INTO tempo VALUES ('Campos do Jordao', 3, 14, 0.22, '2010-06-17');
289
290 INSERT INTO tempo VALUES ('Taubate', 9, 22, 0.11, '2010-06-13');
291 // observar o erro no comando acima!
292
293 INSERT INTO cidades VALUES ('Taubate', -21.12, -42.3);
294 INSERT INTO tempo VALUES ('Taubate', 9, 22, 0.11, '2010-06-13');
295 INSERT INTO tempo VALUES ('Taubate', 11, 25, 0.12, '2010-06-14');
296 INSERT INTO tempo VALUES ('Taubate', 12, 27, 0.17, '2010-06-15');
297 INSERT INTO tempo VALUES ('Taubate', 7, 24, 0.25, '2010-06-16');
298 INSERT INTO tempo VALUES ('Taubate', 8, 24, 0.12, '2010-06-17');
299
300 SELECT cidade, (temp_bx+temp_al)/2 AS temp_med, data
301 FROM tempo;
302
303 SELECT *
304 FROM tempo, cidades
305 WHERE cidade = nome;
306
307 SELECT tempo.cidade, tempo.temp_bx, tempo.temp_al,
308        tempo.prcp, tempo.data, cidades.x, cidades.y
309 FROM tempo, cidades
310 WHERE cidades.nome = tempo.cidade;
311
312 SELECT *
313        FROM tempo INNER JOIN cidades ON (tempo.cidade = cidades.nome);
314
315 SELECT *
316        FROM tempo LEFT OUTER JOIN cidades ON (tempo.cidade = cidades.nome);
317
318 SELECT W1.cidade, W1.temp_bx AS low, W1.temp_al AS alta,
319        W2.cidade, W2.temp_bx AS baixa, W2.temp_al AS alta
320 FROM tempo W1, tempo W2
321 WHERE W1.temp_bx < W2.temp_bx
322        AND W1.temp_al > W2.temp_al;
323
324 SELECT max(temp_bx)
325 FROM tempo;
326
327 SELECT cidade FROM tempo
328        WHERE temp_bx = (SELECT max(temp_bx) FROM tempo);
329
330 CREATE VIEW myview AS
331        SELECT cidade, temp_bx, temp_al, prcp, data, x, y
332        FROM tempo, cidades
333        WHERE cidade = nome;
334
335 SELECT * FROM myview;
336
337 -----
338 EXEMPLO 4
339 -----
340
341 CREATE TABLE focos
342 (
343        nro                INTEGER PRIMARY KEY,
344        lat                REAL,
345        lon                REAL,
346        latGMS             VARCHAR(20),
347        lonGMS             VARCHAR(20),
348        data               VARCHAR(20),
349        hora               VARCHAR(20),
350        satelite           VARCHAR(20),
351        municipio          VARCHAR(40),
352        estado             VARCHAR(2),
353        vegetacao          VARCHAR(50),
354        suscetibilidade    VARCHAR(50),
355        precipitacao       REAL,
```

```
356     numdiassemchuva      INTEGER,
357     risco                 REAL,
358     persistencia         REAL
359 );
360
361 COPY focos FROM 'c:/fire_spot_sp.csv' CSV;
362
```