

```

1  /***** BD Objeto-Relacional Cinema - LBD
2  *****/
3  -- tipo para telefone vetor variável
4  DROP TYPE tipofone FORCE ;
5  CREATE OR REPLACE TYPE tipofone AS VARRAY(5) OF INTEGER ;
6  -- tipo endereço similar a uma struct
7  DROP TYPE tipoendereço FORCE;
8  CREATE OR REPLACE TYPE tipoendereço AS OBJECT
9  ( tipologradouro CHAR(10),
10 logradouro VARCHAR2(40),
11 numero SMALLINT,
12 complemento CHAR(10),
13 bairro VARCHAR2(25),
14 cidade VARCHAR2(25),
15 UF CHAR(2),
16 CEP NUMBER(8),
17 fones tipofone )
18 FINAL ;
19 -- tipo filme
20 DROP TYPE tipofilme FORCE ;
21 CREATE OR REPLACE TYPE tipofilme AS OBJECT
22 ( cod_filme SMALLINT,
23 titulo_filme VARCHAR2(50),
24 ano_lancamento SMALLINT,
25 genero VARCHAR2(15),
26 duracao SMALLINT,
27 estudio VARCHAR2(30) ) ;
28 -- tabela tipada baseada em tipofilme
29 DROP TABLE or_filme CASCADE CONSTRAINTS ;
30 CREATE TABLE or_filme OF tipofilme
31 ( PRIMARY KEY ( cod_filme),
32 UNIQUE ( titulo_filme, ano_lancamento),
33 duracao NOT NULL )
34 OBJECT IDENTIFIER IS SYSTEM GENERATED ;
35 -- populando filme
36 INSERT INTO or_filme VALUES ( 1, 'Velozes e Furiosos 7', 2014, 'Aventura', 100, 'Universal' ) ;
37 INSERT INTO or_filme VALUES ( 2, 'Minha mãe e uma peça 2', 2016, 'Comedia', 105, 'Embrafilme' ) ;
38 COMMIT ;
39 -- tipo para sala
40 DROP TYPE tiposala FORCE ;
41 CREATE OR REPLACE TYPE tiposala AS OBJECT
42 ( nome_sala CHAR(25),
43 capacidade SMALLINT,
44 situacao_sala CHAR(15) ) NOT FINAL ;
45 DESC tiposala ;
46 -- tipo sala vip
47 DROP TYPE tiposala_vip FORCE;
48 CREATE OR REPLACE TYPE tiposala_vip UNDER tiposala
49 ( categ_poltrona CHAR(15),
50 realidade_aumentada CHAR(1) ) FINAL ;
51 DESC tiposala_vip ;
52 -- tabela tipada baseada em tiposala
53 DROP TABLE or_sala CASCADE CONSTRAINTS;
54 CREATE TABLE or_sala OF tiposala
55 ( PRIMARY KEY (nome_sala),
56 CHECK ( situacao_sala IN ( 'Disponível', 'Em uso', 'Manutenção', 'Desativada' )))
57 OBJECT IDENTIFIER IS SYSTEM GENERATED ;
58 -- populando sala
59 INSERT INTO or_sala VALUES ( 'Branca', 250, 'Disponível' ) ;
60 INSERT INTO or_sala VALUES ( 'Verde', 200, 'Em uso' ) ;
61 SELECT * FROM or_sala ;
62 -- tipo horario
63 DROP TYPE tipohorario FORCE ;
64 CREATE OR REPLACE TYPE tipohorario AS OBJECT
65 ( horario NUMBER(4,2) ) FINAL ;
66 -- tabela tipada horario
67 DROP TABLE or_horario CASCADE CONSTRAINTS;
68 CREATE TABLE or_horario OF tipohorario
69 ( UNIQUE ( horario) )
70 OBJECT IDENTIFIER IS SYSTEM GENERATED ;
71 -- populando
72 INSERT INTO or_horario VALUES ( 14.00 ) ;
73 INSERT INTO or_horario VALUES ( 16.30 ) ;
74 INSERT INTO or_horario VALUES ( 19.20 ) ;
75 -- tipo para sessao
76 DROP TYPE tiposessao FORCE;
77 CREATE OR REPLACE TYPE tiposessao AS OBJECT
78 ( num_sessao INTEGER,
79 dt_sessao DATE,
80 filme REF tipofilme,
81 horario_inicio REF tipohorario,
82 publico SMALLINT ) FINAL ;
83 -- tabela tipada para sessao
84 DROP TABLE or_sessao CASCADE CONSTRAINTS ;
85 CREATE TABLE or_sessao OF tiposessao
86 ( PRIMARY KEY ( num_sessao) ,
87 dt_sessao NOT NULL)
88 OBJECT IDENTIFIER IS SYSTEM GENERATED;
89 -- sequencia para sessao
90 DROP SEQUENCE sessao_or_seq ;
91 CREATE SEQUENCE sessao_or_seq START WITH 18000 ;
92 -- finalmente populando sessao
93 INSERT INTO or_sessao VALUES ( sessao_or_seq.nextval ,
94 current_date , ( SELECT REF(f) FROM or_filme f WHERE f.cod_filme = 2),
95 ( SELECT REF(h) FROM or_horario h WHERE h.horario = 16.3 ) , 98 ) ;
96 INSERT INTO or_sessao VALUES ( sessao_or_seq.nextval ,
97 current_date , ( SELECT REF(f) FROM or_filme f WHERE f.cod_filme = 1),
98 ( SELECT REF(h) FROM or_horario h WHERE h.horario = 19.2 ) , 78 ) ;
99 SELECT * FROM or_sessao ;
100 SELECT ss.num_sessao, ss.publico, ss.filme.titulo_filme, ss.horario_inicio.horario
101 FROM or_sessao ss ;
102
103 -- tipo sessao na sala
104 DROP TYPE tipo_sessao_sala FORCE ;
105 CREATE OR REPLACE TYPE tipo_sessao_sala AS OBJECT
106 ( sessao REF tiposessao) FINAL ;
107 -- tabela para tipos sessao sala
108 DROP TYPE tabela_sesoes FORCE;
109 CREATE OR REPLACE TYPE tabela_sesoes AS TABLE OF tipo_sessao_sala ;
110 -- aninhando a tabela de sessoes na sala
111 ALTER TYPE tiposala ADD ATTRIBUTE sesoes_exibidas tabela_sesoes
112 CASCADE NOT INCLUDING TABLE DATA ;
113
114 -- inserindo as sessões em uma sala específica
115 UPDATE or_sala s SET s.sesoes_exibidas = tabela_sesoes (
116 tipo_sessao_sala ( ( SELECT REF(ss) FROM or_sessao ss WHERE ss.num_sessao = 18000)),
117 tipo_sessao_sala ( ( SELECT REF(ss) FROM or_sessao ss WHERE ss.num_sessao = 18001)) )
118 WHERE s.nome_sala = 'Verde' ;
119

```

```

120 SELECT s.nome_sala, se.sessao.num_sessao, se.sessao.filme.titulo_filme
121 FROM or_sala s, TABLE ( s.sessoes_exibidas) se ;
122
123 /*****
124 atividade 6
125 *****/
126 --a) Defina um tipoartista e a respectiva tabela tipada or_artista.
127 --Preencha com duas linhas. Altere o tipofilme ( criado em sala de aula)
128 --para aninhar os artistas que participam do filme e-inclua para o filme de código 1 os artistas cadastrados anteriormente, um como ator outro como diretor
129 DROP TYPE tipoartista FORCE ;
130 CREATE OR REPLACE TYPE tipoartista AS OBJECT
131 ( nome_artista VARCHAR2(30),
132 ano_nascimento SMALLINT,
133 nacionalidade_artista VARCHAR2(20)) FINAL ;
134
135 DROP TABLE or_artista CASCADE CONSTRAINTS ;
136 CREATE TABLE or_artista OF tipoartista
137 ( UNIQUE ( nome_artista))
138 OBJECT IDENTIFIER IS SYSTEM GENERATED ;
139
140 INSERT INTO or_artista VALUES ( 'Vin Diesel', 1970, 'Americano') ;
141 INSERT INTO or_artista VALUES ( 'Paul Walker', 1972, 'Americano') ;
142 INSERT INTO or_artista VALUES ( 'Eva Mendes', 1987, 'Porto Riquenha') ;
143
144 DROP TYPE tipoparticipacao FORCE ;
145 CREATE OR REPLACE TYPE tipoparticipacao AS OBJECT
146 ( artista REF tipoartista,
147 tipo_participacao CHAR(20) ) FINAL ;
148
149 DROP TYPE tab_participantes FORCE ;
150 CREATE OR REPLACE TYPE tab_participantes AS TABLE OF tipoparticipacao ;
151 -- aninhando a tabela em filme
152 ALTER TYPE tipofilme ADD ATTRIBUTE elenco tab_participantes
153 CASCADE NOT INCLUDING TABLE DATA ;
154
155 UPDATE or_filme SET elenco = tab_participantes (
156 tipoparticipacao(( SELECT ref(a) FROM or_artista a WHERE a.nome_artista = 'Vin Diesel'),
157 'DIRECAO'),
158 tipoparticipacao(( SELECT ref(a) FROM or_artista a WHERE a.nome_artista = 'Paul Walker'),
159 'ATUACAO'),
160 tipoparticipacao(( SELECT ref(a) FROM or_artista a WHERE a.nome_artista = 'Eva Mendes'),
161 'ATUACAO') )
162 WHERE UPPER(titulo_filme) LIKE '%VELOZES%';
163
164 SELECT f.titulo_filme, e.artista.nome_artista, e.tipo_participacao
165 FROM or_filme f, TABLE(f.elenco) e ;
166
167 --b) Defina um tipoassento e a respectiva tabela or_assento referenciando o objeto sala.
168 DROP TYPE tipoassento FORCE ;
169 CREATE OR REPLACE TYPE tipoassento AS OBJECT
170 ( numero_assento SMALLINT ,
171 fileira_assento CHAR(1),
172 nome_sala CHAR(20),
173 sala REF tiposala ,
174 tipo_assento CHAR(15) ) FINAL ;
175
176 -- tabela tipada assento
177 DROP TABLE or_assento CASCADE CONSTRAINTS ;
178 CREATE TABLE or_assento OF tipoassento
179 (PRIMARY KEY ( nome_sala, fileira_assento, numero_assento),
180 tipo_assento NOT NULL )
181 OBJECT IDENTIFIER IS SYSTEM GENERATED ;
182
183 --d) Inclua três assentos em cada sala cadastrada na aula.
184
185 INSERT INTO or_assento VALUES ( 1, 'A', 'Branca', ( SELECT REF(s) FROM or_sala s WHERE UPPER(s.nome_sala) LIKE '%BRANCA%' ), 'Normal') ;
186 INSERT INTO or_assento VALUES ( 2, 'A', 'Branca', ( SELECT REF(s) FROM or_sala s WHERE UPPER(s.nome_sala) LIKE '%BRANCA%' ), 'Normal') ;
187 INSERT INTO or_assento VALUES ( 3, 'A', 'Branca', ( SELECT REF(s) FROM or_sala s WHERE UPPER(s.nome_sala) LIKE '%BRANCA%' ), 'Normal') ;
188 INSERT INTO or_assento VALUES ( 1, 'A', 'Verde', ( SELECT REF(s) FROM or_sala s WHERE UPPER(s.nome_sala) LIKE '%VERDE%' ), 'Normal') ;
189 INSERT INTO or_assento VALUES ( 1, 'B', 'Verde', ( SELECT REF(s) FROM or_sala s WHERE UPPER(s.nome_sala) LIKE '%VERDE%' ), 'Normal') ;
190 INSERT INTO or_assento VALUES ( 1, 'C', 'Verde', ( SELECT REF(s) FROM or_sala s WHERE UPPER(s.nome_sala) LIKE '%VERDE%' ), 'Normal') ;
191 INSERT INTO or_assento VALUES ( 1, 'C', 'Verde', ( SELECT REF(s) FROM or_sala s WHERE UPPER(s.nome_sala) LIKE '%VERDE%' ), 'Normal') ;
192
193 SELECT * FROM or_assento ;
194 SELECT a.*, a.sala.nome_sala, a.sala.capacidade
195 FROM or_assento a ;
196
197 --d) Defina um tipoingresso que referencia o assento
198 DROP TYPE tipoingresso FORCE ;
199 CREATE OR REPLACE TYPE tipoingresso AS OBJECT
200 ( num_ingresso INTEGER ,
201 assento REF tipoassento,
202 tipo_ingresso CHAR(15) ,
203 valor_pago NUMBER(10,2) ) FINAL ;
204
205 --e) Aninhe os ingressos na sessão, de tal forma que ao consultar a sessão os ingressos vendidos venham juntos
206 DROP TYPE tab_ingresso FORCE ;
207 CREATE OR REPLACE TYPE tab_ingresso AS TABLE OF tipoingresso ;
208
209 ALTER TYPE tiposessao ADD ATTRIBUTE ingressos tab_ingresso
210 CASCADE NOT INCLUDING TABLE DATA ;
211
212 SELECT * FROM or_sessao ;
213 DESC tipoingresso ;
214 SELECT a.*, a.sala.nome_sala, a.sala.capacidade
215 FROM or_sessao a ;
216
217 --f) Inclua três ingressos para alguma sessão criada em sala de aula
218 UPDATE or_sessao ss SET ss.ingressos = tab_ingresso (
219 tipoingresso ( 1, ( SELECT REF(a) FROM or_assento a WHERE a.numero_assento = 1 AND a.fileira_assento = 'A'
220 AND UPPER(a.sala.nome_sala) LIKE '%BRANCA%' ), 'INTEIRA', 35 ) )
221 WHERE ss.num_sessao = 18000 ;
222 -- atualizando 18001
223 UPDATE or_sessao ss SET ss.ingressos = tab_ingresso
224 ( tipoingresso ( 2, ( SELECT REF(a) FROM or_assento a WHERE a.numero_assento = 1
225 AND a.fileira_assento = 'A' AND UPPER(a.sala.nome_sala) LIKE '%VERDE%' ), 'MEIA', 17.5 ) ,
226 tipoingresso ( 3, ( SELECT REF(a) FROM or_assento a WHERE a.numero_assento = 1
227 AND a.fileira_assento = 'B' AND UPPER(a.sala.nome_sala) LIKE '%VERDE%' ), 'MEIA', 17.5 ) ,
228 tipoingresso ( 4, ( SELECT REF(a) FROM or_assento a WHERE a.numero_assento = 1
229 AND a.fileira_assento = 'C' AND UPPER(a.sala.nome_sala) LIKE '%VERDE%' ), 'INTEIRA', 35 ) )
230 WHERE ss.num_sessao = 18001 ;
231
232 /**** Atividade 7 ****/
233 -- a) Inclua mais um funcionário em duas etapas, primeiro com os dados sem as estruturas aninhadas de fone e escala de trabalho,
234 -- depois atualize incluindo 3 fones e duas escalas
235 INSERT INTO or_funcionario (num_funcional, nome_func, sexo_func, dt_admissao, num_ctps)
236 VALUES ( 20, 'Maria Aparecida Santos', 'F', '11/11/2015', '231A') ;
237
238 ALTER TYPE tipofuncionario ADD ATTRIBUTE fonefunc tipofone
239 CASCADE NOT INCLUDING TABLE DATA ;

```

```

240
241 UPDATE or_funcionario SET fonefunc = tipofone( 945388665, 922654321, 970089365 )
242 WHERE num_funcional = 20 ;
243
244 UPDATE or_funcionario SET escala_atendimento = tab_trabalho (
245 tipotrabalho ( (SELECT REF(e) FROM or_escalas WHERE e.num_escalas = 1),
246               current_date - 2, current_date +5, 'Caixa') )
247 WHERE num_funcional = 20 ;
248
249 -- b) Mostre o elenco dos filmes que serão exibidos hoje :
250 -- nome filme, duração, nome artista, nacionalidade e o tipo de participação
251 SELECT f.titulo_filme, f.duracao, e.artista.nome_artista, e.artista.nacionalidade_artista, e.tipo_participacao
252 FROM or_filme f, TABLE (f.elenco) e, or_sessao ss
253 WHERE TO_CHAR(ss.dt_sessao,'DD/MM/YYYY') = TO_CHAR(current_date, 'DD/MM/YYYY')
254 AND f.titulo_filme = ss.filme.titulo_filme ;
255
256 DESC or_artista ;
257
258 -- c) Mostre os ingressos vendidos para as sessões de hoje : numero sessão, data , hora inicio, titulo filme, numero ingresso,
259 -- tipo ingresso, valor pago, numero assento, fileira, sala e tipo de assento e capacidade da sala
260 desc or_sessao ;
261 desc tipoingresso ;
262 desc or_assento ;
263
264 SELECT ss.num_sessao, ss.dt_sessao, ss.horario_inicio.horario, ss.filme.titulo_filme ,
265 ig.num_ingresso, ig.tipo_ingresso, ig.valor_pago, ig.assento.numero_assento, ig.assento.fileira_assento, ig.assento.nome_sala,
266 ig.assento.sala.capacidade
267 FROM or_sessao ss, TABLE (ss.ingressos) ig
268 WHERE TO_CHAR(ss.dt_sessao,'DD/MM/YYYY') = TO_CHAR(current_date, 'DD/MM/YYYY') ;
269
270 desc or_filme ;
271
272 -- d) Mostre os funcionários que trabalham nas sessões de hoje : nome funcionário, função, dia da semana,
273 -- hora início escala, hora término escala, início e término da escala
274 SELECT DISTINCT f.nome_func, ea.funcao, ea.escala.dia_semana, ea.escala.hora_inicio.horario, ea.escala.hora_termino.horario,
275 ea.dia_inicio, ea.dia_termino
276 FROM or_funcionario f, or_sessao ss, TABLE ( f.escalas_atendimento) ea
277 WHERE ss.dt_sessao BETWEEN ea.dia_inicio AND ea.dia_termino
278 AND ss.horario_inicio.horario BETWEEN ea.escala.hora_inicio.horario AND ea.escala.hora_termino.horario ;
279
280 DESC or_sessao ;
281
282 SELECT * FROM or_funcionario ;
283 SELECT f.nome_func, fn.*
284 FROM or_funcionario f, TABLE ( f.fonefunc) fn;
285 SELECT f.nome_func, ea.*
286 FROM or_funcionario f, TABLE ( f.escalas_atendimento) ea ;
287 SELECT f.nome_func, ea.escala.dia_semana, ea.escala.hora_inicio.horario, ea.escala.hora_termino.horario,
288 ea.dt_ini_trabalho, ea.funcao
289 FROM or_funcionario f, TABLE ( f.escalas_atendimento) ea;

```