

Apêndice Complementar

C Script de Testes

A Listagem 1 apresenta os testes de validação referentes a DDL gerada automaticamente do modelo bancário ilustrado na Figura 6. Comentários foram inseridos de forma a identificar quais elementos ER da DDL são testados. Por exemplo, as linhas 3 a 25 testam a entidade forte GERENTE.

```
1 SET serveroutput ON;
2
3 -- Início do teste da entidade GERENTE
4 DECLARE
5     gerenteId GERENTE.id%TYPE;
6     gerenteNome GERENTE.nome%TYPE;
7 BEGIN
8     INSERT INTO GERENTE (id, nome) VALUES (null, 'Arthur') RETURNING id, nome INTO gerenteId, gerenteNome;
9
10    dbms_output.put_line(gerenteId);
11    dbms_output.put_line(gerenteNome);
12 END;
13 /
14
15 DECLARE
16     gerenteId2 GERENTE.id%TYPE;
17     gerenteNome2 GERENTE.nome%TYPE;
18 BEGIN
19     INSERT INTO GERENTE (id, nome) VALUES (1, 'Alan') RETURNING id, nome INTO gerenteId2, gerenteNome2;
20
21    dbms_output.put_line(gerenteId2);
22    dbms_output.put_line(gerenteNome2);
23 END;
24 /
25 -- Fim do teste da entidade GERENTE
26
27 -- Início do teste da entidade AGENCIA
28 DECLARE
29     agenciaCodigo AGENCIA.codigo%TYPE;
30     agenciaNome AGENCIA.nome%TYPE;
31     agenciaGerenteId AGENCIA.id_gerente%TYPE;
32 BEGIN
33     INSERT INTO AGENCIA (codigo, nome, id_gerente) VALUES (null, 'Bradesco', 1) RETURNING codigo, nome, id_gerente INTO
34         agenciaCodigo, agenciaNome, agenciaGerenteId;
35
36    dbms_output.put_line(agenciaCodigo);
37    dbms_output.put_line(agenciaNome);
38    dbms_output.put_line(agenciaGerenteId);
39 END;
40 /
41
42 DECLARE
43     agenciaCodigo2 AGENCIA.codigo%TYPE;
44     agenciaNome2 AGENCIA.nome%TYPE;
45     agenciaGerenteId2 AGENCIA.id_gerente%TYPE;
46 BEGIN
47     INSERT INTO AGENCIA (codigo, nome, id_gerente) VALUES (3646, 'Santander', null) RETURNING codigo, nome, id_gerente
48         INTO agenciaCodigo2, agenciaNome2, agenciaGerenteId2;
49
50    dbms_output.put_line(agenciaCodigo2);
51    dbms_output.put_line(agenciaNome2);
52    dbms_output.put_line(agenciaGerenteId2);
53 END;
54 /
55
56 DECLARE
57     agenciaCodigo3 AGENCIA.codigo%TYPE;
58     agenciaNome3 AGENCIA.nome%TYPE;
59     agenciaGerenteId3 AGENCIA.id_gerente%TYPE;
60 BEGIN
61     INSERT INTO AGENCIA (codigo, nome, id_gerente) VALUES (3646, 'Banco do Brasil', 1) RETURNING codigo, nome, id_gerente
62         INTO agenciaCodigo3, agenciaNome3, agenciaGerenteId3;
63
64    dbms_output.put_line(agenciaCodigo3);
65    dbms_output.put_line(agenciaNome3);
66    dbms_output.put_line(agenciaGerenteId3);
67 END;
68 /
69 -- Fim do teste da entidade AGENCIA
70
71 -- Início do teste da entidade CORRENTISTA
72 DECLARE
73     correntistaCpf CORRENTISTA.cpf%TYPE;
74     correntistaNome CORRENTISTA.nome%TYPE;
```

```

72 BEGIN
73     INSERT INTO CORRENTISTA (cpf, nome) VALUES (null, 'Joseph') RETURNING cpf, nome INTO correntistaCpf, correntistaNome;
74
75     dbms_output.put_line(correntistaCpf);
76     dbms_output.put_line(correntistaNome);
77 END;
78 /
79
80 DECLARE
81     correntistaCpf2 CORRENTISTA.cpf%TYPE;
82     correntistaNome2 CORRENTISTA.nome%TYPE;
83 BEGIN
84     INSERT INTO CORRENTISTA (cpf, nome) VALUES (12345678910, 'John') RETURNING cpf, nome INTO correntistaCpf2,
85         correntistaNome2;
86
87     dbms_output.put_line(correntistaCpf2);
88     dbms_output.put_line(correntistaNome2);
89 END;
90 /
91 -- Fim do teste da entidade CORRENTISTA
92
93 -- Início do teste da entidade CORRENTE
94 DECLARE
95     correnteTxManutencao CORRENTE.tx_manutencao%TYPE;
96     correnteContaNumero CORRENTE.numero_conta%TYPE;
97 BEGIN
98     INSERT INTO CORRENTE (tx_manutencao, numero_conta) VALUES (6, null) RETURNING tx_manutencao, numero_conta INTO
99         correnteTxManutencao, correnteContaNumero;
100
101     dbms_output.put_line(correnteTxManutencao);
102     dbms_output.put_line(correnteContaNumero);
103 END;
104 /
105
106 DECLARE
107     correnteTxManutencao2 CORRENTE.tx_manutencao%TYPE;
108     correnteContaNumero2 CORRENTE.numero_conta%TYPE;
109 BEGIN
110     INSERT INTO CORRENTE (tx_manutencao, numero_conta) VALUES (5, 616279) RETURNING tx_manutencao, numero_conta INTO
111         correnteTxManutencao2, correnteContaNumero2;
112
113     dbms_output.put_line(correnteTxManutencao2);
114     dbms_output.put_line(correnteContaNumero2);
115 END;
116 /
117 -- Fim do teste da entidade CORRENTE
118
119 -- Início do teste da entidade POUPANCA
120 DECLARE
121     poupancaDiaAniversario POUPANCA.dia_aniversario%TYPE;
122     poupancaContaNumero POUPANCA.numero_conta%TYPE;
123 BEGIN
124     INSERT INTO POUPANCA (dia_aniversario, numero_conta) VALUES (23, null) RETURNING dia_aniversario, numero_conta INTO
125         poupancaDiaAniversario, poupancaContaNumero;
126
127     dbms_output.put_line(poupancaDiaAniversario);
128     dbms_output.put_line(poupancaContaNumero);
129 END;
130 /
131
132 DECLARE
133     poupancaDiaAniversario2 POUPANCA.dia_aniversario%TYPE;
134     poupancaContaNumero2 POUPANCA.numero_conta%TYPE;
135 BEGIN
136     INSERT INTO POUPANCA (dia_aniversario, numero_conta) VALUES (23, 616280) RETURNING dia_aniversario, numero_conta INTO
137         poupancaDiaAniversario2, poupancaContaNumero2;
138
139     dbms_output.put_line(poupancaDiaAniversario2);
140     dbms_output.put_line(poupancaContaNumero2);
141 END;
142 /
143 -- Fim do teste da entidade POUPANCA
144
145 -- Início do teste da entidade CONTA
146 DECLARE
147     contaNumero CONTA.numero%TYPE;
148     contaSaldo CONTA.saldo%TYPE;
149     contaCodigoAgencia CONTA.codigo_agencia%TYPE;
150 BEGIN
151     INSERT INTO CONTA (numero, saldo, codigo_agencia) VALUES (null, 3000, 3646) RETURNING numero, saldo, codigo_agencia
152         INTO contaNumero, contaSaldo, contaCodigoAgencia;
153
154     dbms_output.put_line(contaNumero);
155     dbms_output.put_line(contaSaldo);
156     dbms_output.put_line(contaCodigoAgencia);
157 END;
158 /
159
160 DECLARE
161     contaNumero2 CONTA.numero%TYPE;
162     contaSaldo2 CONTA.saldo%TYPE;
163     contaCodigoAgencia2 CONTA.codigo_agencia%TYPE;
164 BEGIN
165     INSERT INTO CONTA (numero, saldo, codigo_agencia) VALUES (616278, 3000, 3646) RETURNING numero, saldo, codigo_agencia
166         INTO contaNumero2, contaSaldo2, contaCodigoAgencia2;

```

```

161 dbms_output.put_line(contaNumero2);
162 dbms_output.put_line(contaSaldo2);
163 dbms_output.put_line(contaCodigoAgencia2);
164 END;
165 /
166
167 DECLARE
168 contaNumero3 CONTA.numero%TYPE;
169 contaSaldo3 CONTA.saldo%TYPE;
170 contaCodigoAgencia3 CONTA.codigo_agencia%TYPE;
171 BEGIN
172 INSERT INTO CONTA (numero, saldo, codigo_agencia) VALUES (616279, 3000, 3646) RETURNING numero, saldo, codigo_agencia
173 INTO contaNumero3, contaSaldo3, contaCodigoAgencia3;
174 dbms_output.put_line(contaNumero3);
175 dbms_output.put_line(contaSaldo3);
176 dbms_output.put_line(contaCodigoAgencia3);
177 END;
178 /
179
180 DECLARE
181 contaNumero4 CONTA.numero%TYPE;
182 contaSaldo4 CONTA.saldo%TYPE;
183 contaCodigoAgencia4 CONTA.codigo_agencia%TYPE;
184 BEGIN
185 INSERT INTO CONTA (numero, saldo, codigo_agencia) VALUES (616280, 5000, 3646) RETURNING numero, saldo, codigo_agencia
186 INTO contaNumero4, contaSaldo4, contaCodigoAgencia4;
187 dbms_output.put_line(contaNumero4);
188 dbms_output.put_line(contaSaldo4);
189 dbms_output.put_line(contaCodigoAgencia4);
190 END;
191 /
192 -- Fim do teste da entidade CONTA
193
194 -- Inicio do teste da restricao de disjuncao
195 DECLARE
196 poupancaDiaAniversario3 POUPANCA.dia_aniversario%TYPE;
197 poupancaContaNumero3 POUPANCA.numero_conta%TYPE;
198 BEGIN
199 INSERT INTO POUPANCA (dia_aniversario, numero_conta) VALUES (23, 616279) RETURNING dia_aniversario, numero_conta INTO
200 poupancaDiaAniversario3, poupancaContaNumero3;
201 dbms_output.put_line(poupancaDiaAniversario3);
202 dbms_output.put_line(poupancaContaNumero3);
203 END;
204 /
205 -- Fim do teste da restricao de disjuncao
206
207 -- Inicio do teste da entidade CHEQUE
208 DECLARE
209 chequeNumRegistro CHEQUE.num_registro%TYPE;
210 chequeCcNumero CHEQUE.numero_conta%TYPE;
211 BEGIN
212 INSERT INTO CHEQUE (num_registro, numero_conta) VALUES (null, 616279) RETURNING num_registro, numero_conta INTO
213 chequeNumRegistro, chequeCcNumero;
214 dbms_output.put_line(chequeNumRegistro);
215 dbms_output.put_line(chequeCcNumero);
216 END;
217 /
218
219 DECLARE
220 chequeNumRegistro2 CHEQUE.num_registro%TYPE;
221 chequeCcNumero2 CHEQUE.numero_conta%TYPE;
222 BEGIN
223 INSERT INTO CHEQUE (num_registro, numero_conta) VALUES (1, 616260) RETURNING num_registro, numero_conta INTO
224 chequeNumRegistro2, chequeCcNumero2;
225 dbms_output.put_line(chequeNumRegistro2);
226 dbms_output.put_line(chequeCcNumero2);
227 END;
228 /
229
230 DECLARE
231 chequeNumRegistro3 CHEQUE.num_registro%TYPE;
232 chequeCcNumero3 CHEQUE.numero_conta%TYPE;
233 BEGIN
234 INSERT INTO CHEQUE (num_registro, numero_conta) VALUES (1, 616279) RETURNING num_registro, numero_conta INTO
235 chequeNumRegistro3, chequeCcNumero3;
236 dbms_output.put_line(chequeNumRegistro3);
237 dbms_output.put_line(chequeCcNumero3);
238 END;
239 /
240 -- Fim do teste da entidade CHEQUE
241
242 -- Inicio do teste da entidade relacionamento CARTAO
243 DECLARE
244 cartaoSenha CARTAO.senha%TYPE;
245 cartaoNumero CARTAO.numero%TYPE;
246 cartaoCpfCorrentista CARTAO.cpf_correntista%TYPE;
247 cartaoNumeroConta CARTAO.numero_conta%TYPE;
248 BEGIN
249 INSERT INTO CARTAO (senha, numero, cpf_correntista, numero_conta) VALUES (1234, 1234567890123456, null, 616279)
250 RETURNING senha, numero, cpf_correntista, numero_conta INTO cartaoSenha, cartaoNumero, cartaoCpfCorrentista,

```

```

250      cartaoNumeroConta;
251      dbms_output.put_line(cartaoSenha);
252      dbms_output.put_line(cartaoNumero);
253      dbms_output.put_line(cartaoCpfCorrentista);
254      dbms_output.put_line(cartaoNumeroConta);
255  END;
256  /
257
258  DECLARE
259      cartaoSenha2 CARTAO.senha%TYPE;
260      cartaoNumero2 CARTAO.numero%TYPE;
261      cartaoCpfCorrentista2 CARTAO.cpf_correntista%TYPE;
262      cartaoNumeroConta2 CARTAO.numero_conta%TYPE;
263  BEGIN
264      INSERT INTO CARTAO (senha, numero, cpf_correntista, numero_conta) VALUES (1234, 1234567890123456, 12345678910, null)
        RETURNING senha, numero, cpf_correntista, numero_conta INTO cartaoSenha2, cartaoNumero2, cartaoCpfCorrentista2,
        cartaoNumeroConta2;
265
266      dbms_output.put_line(cartaoSenha2);
267      dbms_output.put_line(cartaoNumero2);
268      dbms_output.put_line(cartaoCpfCorrentista2);
269      dbms_output.put_line(cartaoNumeroConta2);
270  END;
271  /
272
273  DECLARE
274      cartaoSenha3 CARTAO.senha%TYPE;
275      cartaoNumero3 CARTAO.numero%TYPE;
276      cartaoCpfCorrentista3 CARTAO.cpf_correntista%TYPE;
277      cartaoNumeroConta3 CARTAO.numero_conta%TYPE;
278  BEGIN
279      INSERT INTO CARTAO (senha, numero, cpf_correntista, numero_conta) VALUES (1234, 1234567890123456, 12345678910, 616279)
        RETURNING senha, numero, cpf_correntista, numero_conta INTO cartaoSenha3, cartaoNumero3, cartaoCpfCorrentista3,
        cartaoNumeroConta3;
280
281      dbms_output.put_line(cartaoSenha3);
282      dbms_output.put_line(cartaoNumero3);
283      dbms_output.put_line(cartaoCpfCorrentista3);
284      dbms_output.put_line(cartaoNumeroConta3);
285  END;
286  /
287  -- Fim do teste da entidade relacionamento CARTAO
288
289  -- Início do teste da atributo multivalorado MOVIMENTACOES
290  DECLARE
291      movimentacoesNConta CONTA_MOVIMENTACOES.numero_conta%TYPE;
292      movimentacoesId CONTA_MOVIMENTACOES.pk_movimentacoes%TYPE;
293      movimentacoesMovimentacoes CONTA_MOVIMENTACOES.movimentacoes%TYPE;
294  BEGIN
295      INSERT INTO MOVIMENTACOES (numero_conta, pk_movimentacoes, movimentacoes) VALUES (null, 1, 'Deposito 300.00')
        RETURNING numero_conta, pk_movimentacoes, movimentacoes INTO movimentacoesNConta, movimentacoesId,
        movimentacoesMovimentacoes;
296
297      dbms_output.put_line(movimentacoesNConta);
298      dbms_output.put_line(movimentacoesId);
299      dbms_output.put_line(movimentacoesMovimentacoes);
300  END;
301  /
302
303  DECLARE
304      movimentacoesNConta2 CONTA_MOVIMENTACOES.numero_conta%TYPE;
305      movimentacoesId2 CONTA_MOVIMENTACOES.pk_movimentacoes%TYPE;
306      movimentacoesMovimentacoes2 CONTA_MOVIMENTACOES.movimentacoes%TYPE;
307  BEGIN
308      INSERT INTO MOVIMENTACOES (numero_conta, pk_movimentacoes, movimentacoes) VALUES (616279, null, 'Deposito 300.00')
        RETURNING numero_conta, pk_movimentacoes, movimentacoes INTO movimentacoesNConta2, movimentacoesId2,
        movimentacoesMovimentacoes2;
309
310      dbms_output.put_line(movimentacoesNConta2);
311      dbms_output.put_line(movimentacoesId2);
312      dbms_output.put_line(movimentacoesMovimentacoes2);
313  END;
314  /
315
316  DECLARE
317      movimentacoesNConta3 CONTA_MOVIMENTACOES.numero_conta%TYPE;
318      movimentacoesId3 CONTA_MOVIMENTACOES.pk_movimentacoes%TYPE;
319      movimentacoesMovimentacoes3 CONTA_MOVIMENTACOES.movimentacoes%TYPE;
320  BEGIN
321      INSERT INTO MOVIMENTACOES (numero_conta, pk_movimentacoes, movimentacoes) VALUES (616279, 1, 'Deposito 300.00')
        RETURNING numero_conta, pk_movimentacoes, movimentacoes INTO movimentacoesNConta3, movimentacoesId3,
        movimentacoesMovimentacoes3;
322
323      dbms_output.put_line(movimentacoesNConta3);
324      dbms_output.put_line(movimentacoesId3);
325      dbms_output.put_line(movimentacoesMovimentacoes3);
326  END;
327  /
328  -- Fim do teste da atributo multivalorado MOVIMENTACOES

```

Listagem 1. Script de Teste