

Utilização de Operações de SQL para o Gerenciamento de Tabelas de um Banco de Dados

Lucas Teixeira Gonçalves¹

¹Centro de Ciências Computacionais – Universidade Federal do Rio Grande (FURG)
Caixa Postal 474 – 96.201-900 – Rio Grande – RS – Brazil

lucas.teixeira@furg.br

Resumo. *Este trabalho mostra a utilização de operações de operações da linguagem SQL para o gerenciamento de um modelo relacional previamente proposto. O modelo utilizado será o de um sistema gerenciador de banco de dados para reserva de passagens da impera Fictious Airlines.*

1. Tabelas Utilizadas

Foram utilizadas tabelas do modelo relacional desenvolvido para a Fictious Airlines em um projeto anterior. Primeiramente, serão apresentadas as tabelas originais e, em outra seção, serão utilizadas as tabelas apresentadas para a demonstração de operações SQL [Codd 1970].

Tabela 1. Tabela armazenando os usuários e suas senhas

USUÁRIO	
LOGIN	SENHA
puf3zin	noia123noia
bela_teixeira	farofa40
ninooff	puf1m4r4v1lh0z0

Tabela 2. Tabela armazenando o código dos vôos e a quantidade de assentos disponíveis

VÔO	
CÓDIGO VÔO	ASSENTOS DISPONÍVEIS
201	320
202	7
208	82

Tabela 3. Tabela armazenando o código das aeronaves e o tipo das aeronaves

AERONAVE	
CÓDIGO AERONAVE	TIPO
100	Boeing 737
101	Airbus A-320
102	Airbus A-380

Tabela 4. Tabela que armazena os passageiros

PASSAGEIRO	
CPF	NOME
03173766083	Lucas Teixeira Gonçalves
76823382049	Isabel Cristina de Freitas Teixeira
23894872018	Eliza Alves Nunes

Tabela 5. Tabela que armazena os dados necessários das passagens

PASSAGEM			
CÓDIGO PASSAGEM	CPF PASSAGEIRO	ASSENTO	VALIDADE
A23	76823382049	D05	30/05/16
A26	03173766893	D06	12/06/16
A27	23894872018	D07	12/06/16

Tabela 6. Tabela que armazena os dados necessários dos trechos

TRECHO				
CÓDIGO TRECHO	HORA	DATA	ORIGEM	DESTINO
T01	16:00	26/04	POA	CGH
T02	18:30	26/04	CWB	GIG
T03	02:10	27/04	GIG	POA

Tabela 7. Tabela que organiza a relação entre usuários e passagens

USUÁRIO x PASSAGEM	
LOGIN USUÁRIO	CÓDIGO PASSAGEM
lucast	A23
lucast	A27
bela.teixeira	A26

Tabela 8. Tabela que organiza a ligação entre os vôos e os trechos

VÔO x TRECHO	
CÓDIGO VÔO	CÓDIGO TRECHO
201	T01
201	T03
202	T02

Tabela 9. Tabela que organiza a ligação entre as passagens e os trechos

PASSAGEM x TRECHO	
CÓDIGO PASSAGEM	CÓDIGO TRECHO
A23	T02
A27	T02
A27	T03
A26	T03

Tabela 10. Tabela que armazena os limites de cada classe para cada aeronave

CLASSE			
CÓDIGO AERONAVE	CLASSE	PRIMEIRO ASSENTO	ÚLTIMO ASSENTO
100	Executiva	A01	F06
100	Econômica	G01	L06
101	Executiva	A01	J08
101	Econômica	K01	Q08

Tabela 11. Tabela mostrando o os assentos disponíveis em cada aeronave

MAPA ASSENTOS	
CÓDIGO AERONAVE	ASSENTO
101	A2
101	A3
101	A4
101	A6
102	A5
102	F6
102	G3

Portanto, pode ser utilizado o seguinte modelo relacional descritivo:

```
Usuário(login, senha)
Vôo(Vcódigo, qtd_disponiveis)
Aeronave(Acódigo, tipo)
Passageiro(POcpf, nome)
Passagem(PMcódigo, POcpf, validade, assento)
    POcpf referencia Passageiro
Trecho(Tcódigo, Vcódigo, origem, destino, hora, data)
    Vcódigo referencia Vôo
UsuárioPassagem(login, PMcódigo)
    login referencia Usuário
    PMcódigo referencia Passagem
VôoTrecho(Vcódigo, Tcódigo)
    Vcódigo referencia Vôo
    Tcódigo referencia Trecho
PassagemTrecho(PMcódigo, Tcódigo)
    PMcódigo referencia Passagem
    Tcódigo referencia Trecho
Classes(Acódigo, classe, primeiro_assento, ultimo_assento)
    Acódigo referencia Aeronave
MapaAssentos(Acódigo, assento)
    Acódigo referencia Aeronave
```

Nestas tabelas foram realizadas as seguintes operações:

```
mysql> SELECT * FROM flights WHERE disponiveis >= 50;
+-----+-----+
| vcodigo | disponiveis |
+-----+-----+
|      201 |          320 |
|      208 |           82 |
+-----+-----+
2 rows in set (0.00 sec)
```

Figura 1.

```
mysql> SELECT * FROM flights WHERE disponiveis >= 50 and vcodigo != 201;
+-----+-----+
| vcodigo | disponiveis |
+-----+-----+
|      208 |          82 |
+-----+-----+
1 row in set (0.00 sec)
```

Figura 2.

```
mysql> SELECT * FROM map WHERE assento = "A2" or assento = "F6";
+-----+-----+
| acodigo | assento |
+-----+-----+
|      101 | A2      |
|      102 | F6      |
+-----+-----+
2 rows in set (0.00 sec)
```

Figura 3.

```
mysql> SELECT * FROM tickets WHERE not validade = '2016-05-30';
+-----+-----+-----+-----+
| pmcodigo | cpf          | assento | validade |
+-----+-----+-----+-----+
| A26      | 03173766083 | D06     | 2016-06-12 |
| A27      | 23894872018 | D07     | 2016-06-12 |
+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

Figura 4.

```
mysql> SELECT cpf FROM passenger;
+-----+
| cpf      |
+-----+
| 03173766083 |
| 76823382049 |
| 23809487201 |
+-----+
3 rows in set (0.00 sec)
```

Figura 5.

```
mysql> SELECT tcodigo, origem, destino FROM trecho WHERE not horario = '2016-04-27 02:10:00';
+-----+-----+-----+
| tcodigo | origem | destino |
+-----+-----+-----+
| T01     | POA    | CGH     |
| T02     | CWB    | GIG     |
+-----+-----+-----+
2 rows in set (0.00 sec)
```

Figura 6.

```
mysql> SELECT pmcodigo FROM tickets WHERE cpf = "76823382049" UNION SELECT pmcodigo FROM ticketsTrecho WHERE tcodigo = "T02";
+-----+
| pmcodigo |
+-----+
| A23      |
| A27      |
+-----+
2 rows in set (0.00 sec)
```

Figura 7.

```
mysql> SELECT tcodigo FROM flightsTrecho WHERE vcodigo = "201" UNION SELECT tcodigo FROM ticketsTrecho WHERE pmcodigo = "A27";
```

tcodigo
T01
T03
T02

3 rows in set (0.00 sec)

Figura 8.

```
mysql> (SELECT tcodigo, disponiveis FROM flights, flightsTrecho);
```

tcodigo	disponiveis
T01	320
T01	7
T01	82
T03	320
T03	7
T03	82
T02	320
T02	7
T02	82

9 rows in set (0.00 sec)

Figura 9.

```
mysql> SELECT * FROM flights, trecho;
```

vcodigo	disponiveis	tcodigo	horario	origem	destino
201	320	T01	2016-04-26 16:00:00	POA	CGH
202	7	T01	2016-04-26 16:00:00	POA	CGH
208	82	T01	2016-04-26 16:00:00	POA	CGH
201	320	T02	2016-04-26 18:30:00	CWB	GIG
202	7	T02	2016-04-26 18:30:00	CWB	GIG
208	82	T02	2016-04-26 18:30:00	CWB	GIG
201	320	T03	2016-04-27 02:10:00	GIG	POA
202	7	T03	2016-04-27 02:10:00	GIG	POA
208	82	T03	2016-04-27 02:10:00	GIG	POA

9 rows in set (0.00 sec)

Figura 10.

Referências

Codd, E. F. (1970). A relational model of data for large shared data banks. *Communications of the ACM*, 13(6):377–387.