

Aluno: Rodrigo Kapulka Franco

Disciplina: Banco de dados avançado

## DESAFIO 2 - Controle de Segurança

1, 2 e 3)

```
1  -- 1) Criação dos grupos (papéis);
2
3  CREATE OR REPLACE role role_gerente;
4  CREATE OR REPLACE role role_recepcionista;
5  CREATE OR REPLACE role role_atendente_geral;
6
7  -- 2) Designação dos privilégios para cada um dos grupos criados;
8
9  -- Gerente
10 GRANT SELECT, INSERT, UPDATE, DELETE
11 ON trabalho_dois.*
12 TO role_gerente
13 WITH GRANT OPTION;
14
15 -- Recepcionista
16 GRANT SELECT, INSERT, UPDATE, DELETE
17 ON trabalho_dois.cliente
18 TO role_recepcionista;
19
20 GRANT SELECT, INSERT, UPDATE, DELETE
21 ON trabalho_dois.reserva
22 TO role_recepcionista;
23
24 GRANT SELECT, INSERT, UPDATE, DELETE
25 ON trabalho_dois.hospedagem
26 TO role_recepcionista;
27
28 -- Atendente Geral
29 CREATE OR REPLACE VIEW view_atendente_geral
30 AS
31 SELECT c.nm_cliente, h.nr_quarto
32 FROM trabalho_dois.hospedagem AS h
33 INNER JOIN trabalho_dois.cliente AS c
34 ON h.cd_cliente = c.cd_cliente;
35
36 GRANT INSERT, UPDATE ON trabalho_dois.hospedagem_servico TO "role_atendente_geral";
37
38 GRANT SELECT ON view_atendente_geral TO "role_atendente_geral";
39
40 -- 3) Criação de, no mínimo, um usuário para cada grupo criado;
41
42 CREATE OR REPLACE USER gerente IDENTIFIED BY '12345';
43 CREATE OR REPLACE USER recepcionista IDENTIFIED BY '12345';
44 CREATE OR REPLACE USER atendente IDENTIFIED BY '12345';
45
46 GRANT "role_gerente" TO gerente;
47 SET DEFAULT role "role_gerente" FOR gerente;
48
49 GRANT "role_recepcionista" TO recepcionista;
50 SET DEFAULT role "role_recepcionista" FOR recepcionista;
51
52 GRANT "role_atendente_geral" TO atendente;
53 SET DEFAULT role "role_atendente_geral" FOR atendente;
```

#### 4) Prints testando de forma geral o controle de segurança

Gerente:

- Testando inserção e acesso a registros de uma tabela:

The screenshot shows the HeidiSQL interface with the following details:

- Query Window:**

```
1 INSERT INTO funcionario (CD_FUNCIONARIO, CD_CARGO, NM_FUNCIONARIO) VALUES (5,1,"Rodrigo Kapulka Franco");
2
3 SELECT * FROM funcionario;
```
- Table Structure (funcionario):**

CD_FUNCIONARIO	CD_CARGO	NM_FUNCIONARIO
1	1	João Gerente Fonseca
2	2	Maria Recepcionista da Silva
3	3	Carlos Atendente Geral Costa
4	4	Luiza Estagiária Souza
5	1	Rodrigo Kapulka Franco
- Database Structure:** The left sidebar shows a database named 'trabalho\_dois' containing tables: cargo, categoria, cliente, funcionario, hospedag..., quarto, reserva, servico, and view\_aten...

- Inserindo e deletando um registro em uma tabela:

The screenshot shows the HeidiSQL interface with the following details:

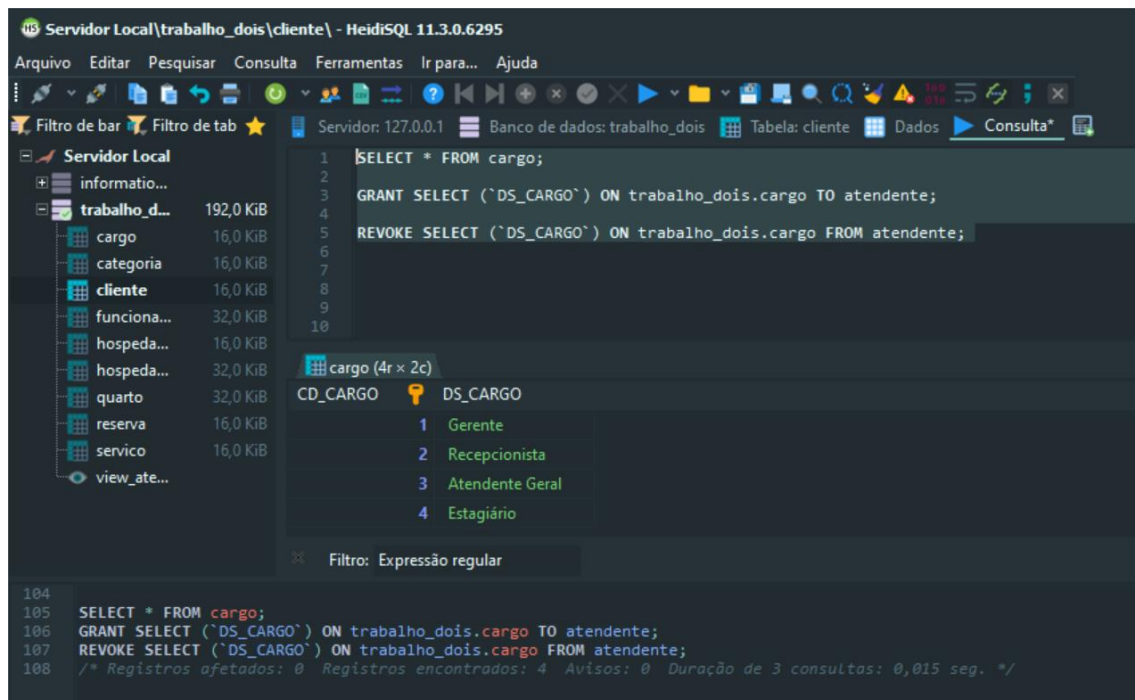
- Query Window:**

```
1
2
3 INSERT INTO cargo (cd_cargo, ds_cargo) VALUES (5, "teste");
4
5 DELETE FROM cargo
6 WHERE cd_cargo = 5;
7 SELECT * FROM cargo;
8
9
10
```
- Table Structure (cargo):**

CD_CARGO	DS_CARGO
1	Gerente
2	Recepcionista
3	Atendente Geral
4	Estagiário
- Database Structure:** The left sidebar shows a database named 'trabalho\_dois' containing tables: cargo, categoria, cliente, funciona..., hospeda..., quarto, reserva, servico, and view\_ate...
- Execution Log:**

```
114 INSERT INTO cargo (cd_cargo, ds_cargo) VALUES (5, "teste");
115 DELETE FROM cargo WHERE cd_cargo = 5;
116 SELECT * FROM cargo;
117 /* Registros afetados: 2 Registros encontrados: 4 Avisos: 0 Duração de 3 consultas: 0,000 seg. */
118
```

- Dando acesso e retirando este acesso ao “role” atendente:



The screenshot shows the HeidiSQL interface with the following SQL commands in the editor:

```
1 SELECT * FROM cargo;
2
3 GRANT SELECT (`DS_CARGO`) ON trabalho_dois.cargo TO atendente;
4
5 REVOKE SELECT (`DS_CARGO`) ON trabalho_dois.cargo FROM atendente;
6
7
8
9
10
```

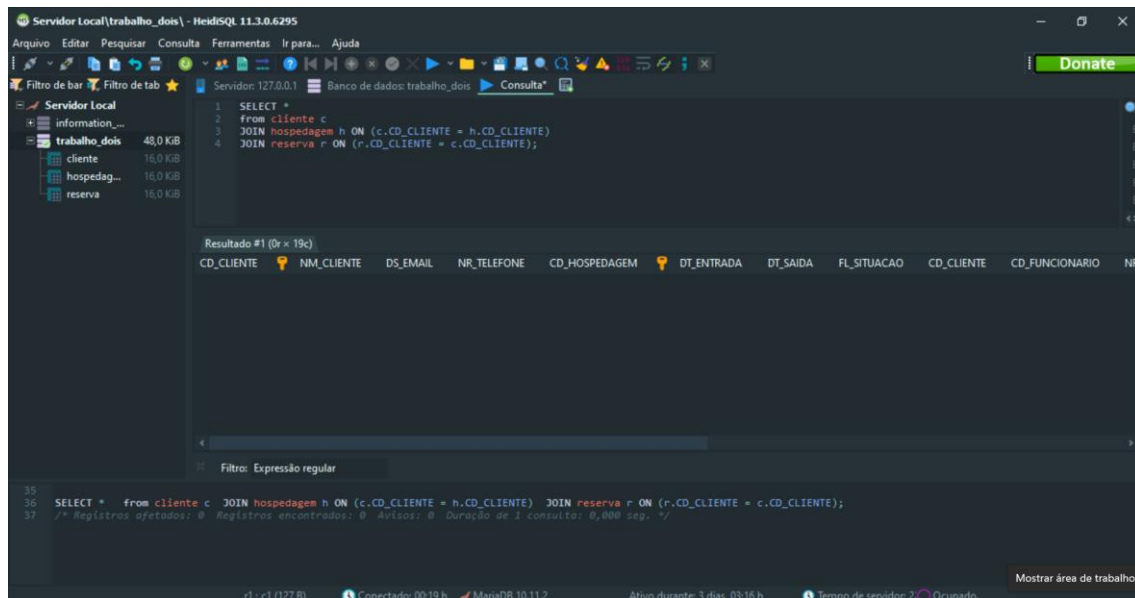
The results pane shows the structure of the 'cargo' table (4 rows x 2 columns):

CD_CARGO	DS_CARGO
1	Gerente
2	Recepcionista
3	Atendente Geral
4	Estagiário

The status bar at the bottom indicates: 104 SELECT \* FROM cargo; 105 GRANT SELECT (`DS\_CARGO`) ON trabalho\_dois.cargo TO atendente; 106 REVOKE SELECT (`DS\_CARGO`) ON trabalho\_dois.cargo FROM atendente; 107 /\* Registros afetados: 0 Registros encontrados: 4 Avisos: 0 Duração de 3 consultas: 0,015 seg. \*/

Recepcionista:

- Acesso às três tabelas especificadas (elas não estão populadas):



The screenshot shows the HeidiSQL interface with the following SQL query in the editor:

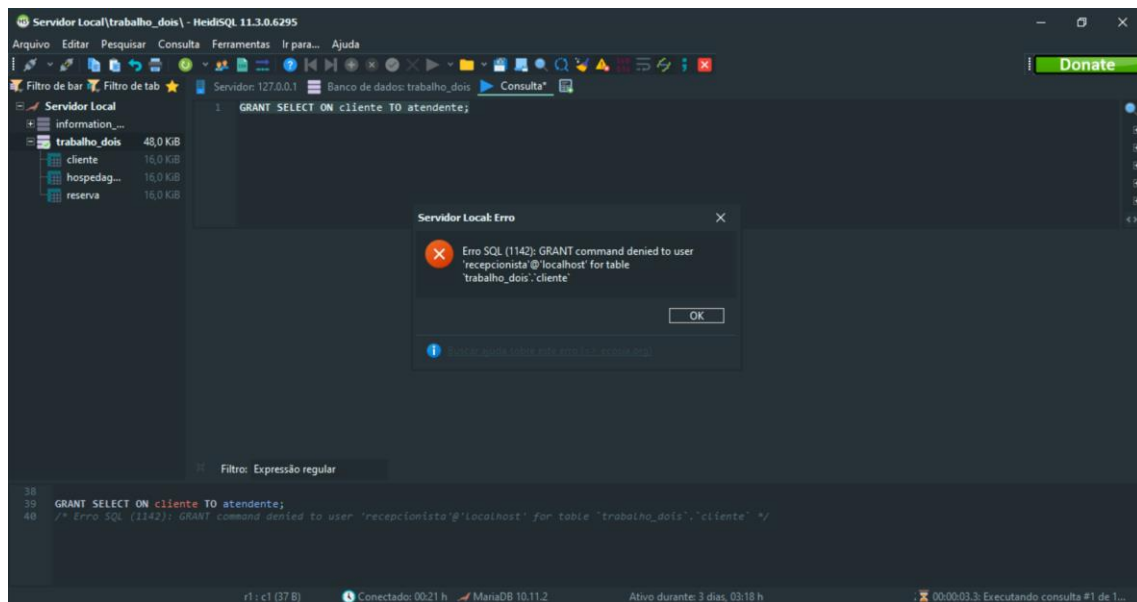
```
1 SELECT *
2 from cliente c
3 JOIN hospedagem h ON (c.CD_CLIENTE = h.CD_CLIENTE)
4 JOIN reserva r ON (r.CD_CLIENTE = c.CD_CLIENTE);
```

The results pane shows the structure of the query results (0 rows x 19 columns):

CD_CLIENTE	NM_CLIENTE	DS_EMAIL	NR_TELEFONE	CD_HOSPEDAGEM	DT_ENTRADA	DT_SAIDA	FL_SITUACAO	CD_CLIENTE	CD_FUNCIONARIO	NR
------------	------------	----------	-------------	---------------	------------	----------	-------------	------------	----------------	----

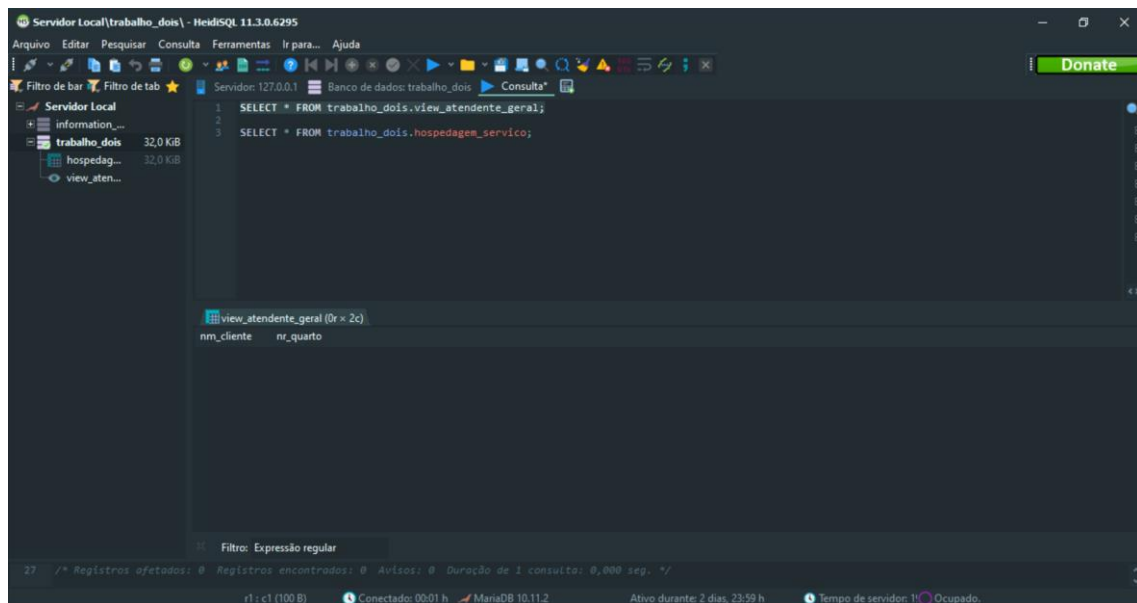
The status bar at the bottom indicates: 35 SELECT \* from cliente c JOIN hospedagem h ON (c.CD\_CLIENTE = h.CD\_CLIENTE) JOIN reserva r ON (r.CD\_CLIENTE = c.CD\_CLIENTE); 36 /\* Registros afetados: 0 Registros encontrados: 0 Avisos: 0 Duração de 1 consulta: 0,000 seg. \*/

- Tentativa de dar alguma permissão a outra “role”:

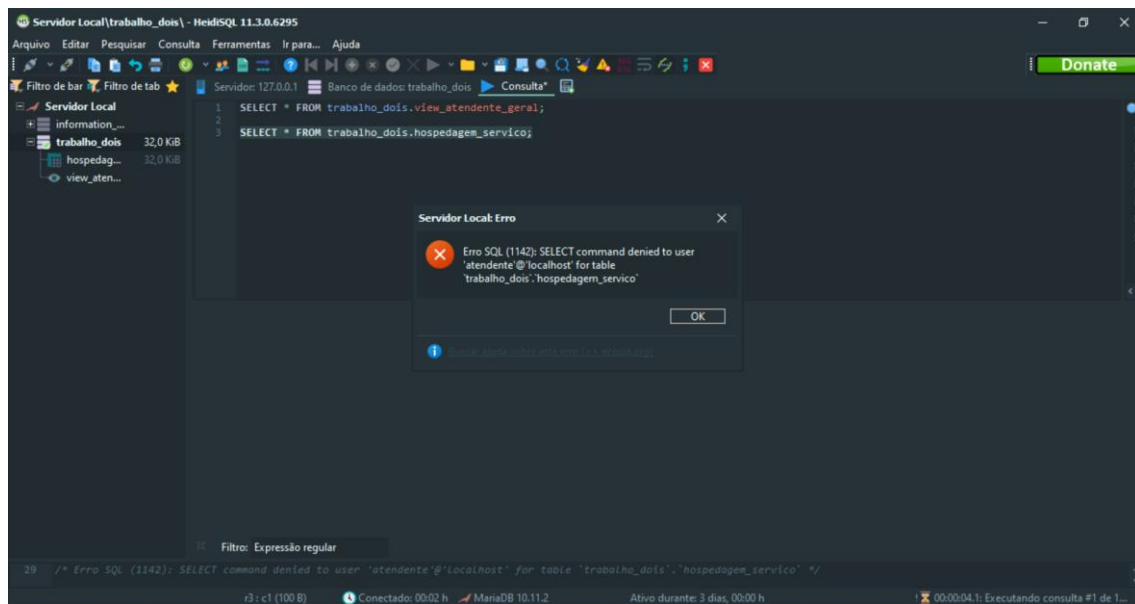


Atendente Geral:

- Acesso (select) a view com as informações do usuário.



- Acesso negado a visualização dos registros na tabela “hospedagem\_serviço”:



- Inserção de um registro na tabela “hospedagem\_serviço”:

