



GESTÃO DE CONDOMÍNIOS

PROJETO DE BANCO DE DADOS

EQUIPE



LEONARDO



LUCAS



MATEUS



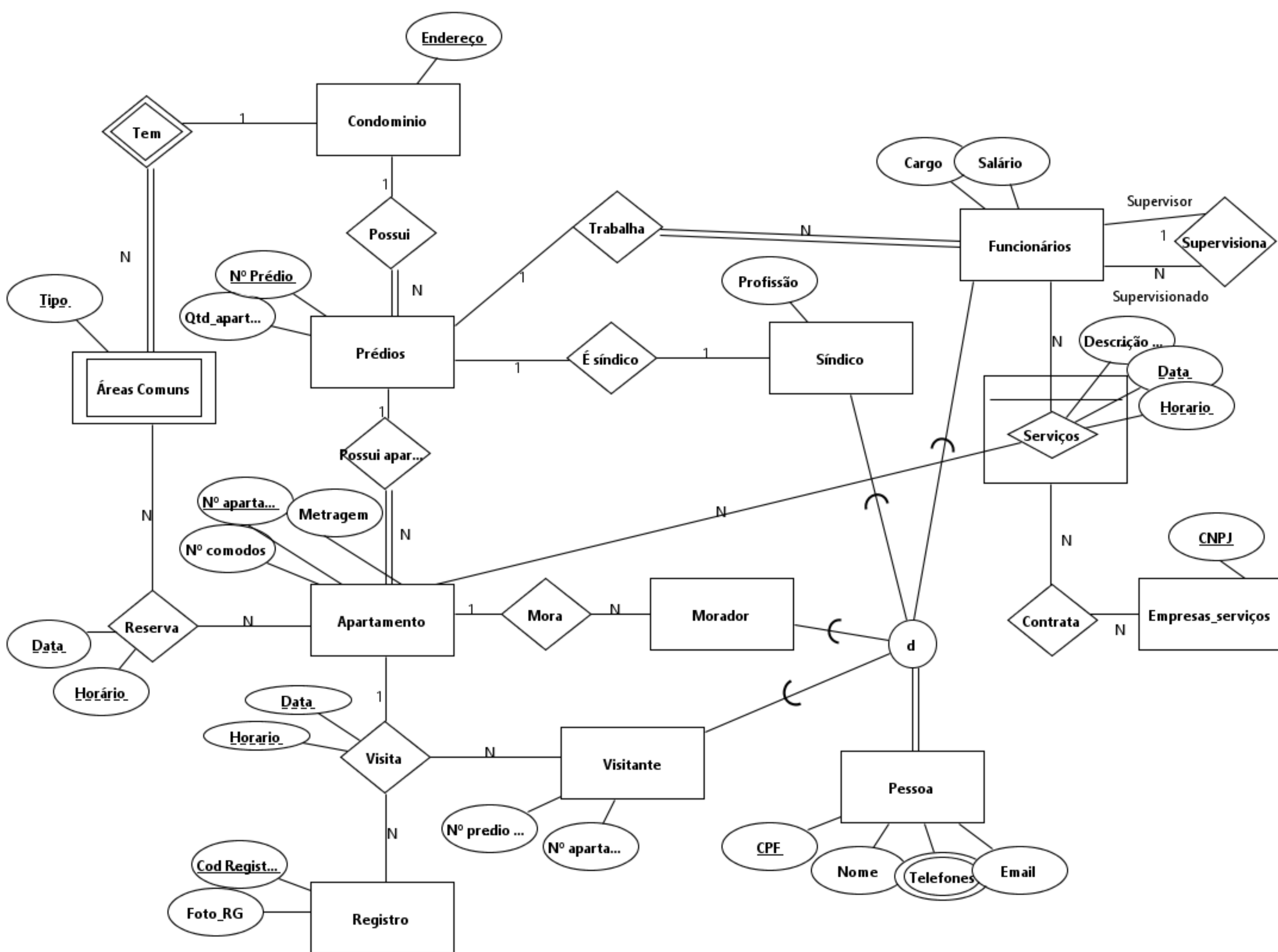
MAIARA



MAÍRA



RENATA



GROUP BY/HAVING

número de serviços realizados por cada funcionário

```
SELECT F.CPF, COUNT(*) AS Numero_de_Serviços
FROM funcionarios F
JOIN servicos S ON F.CPF = S.CPF
GROUP BY F.CPF
HAVING COUNT(*) > 1;
```

CPF	NUMERO_DE_SERVIÇOS
12345678901	2

JUNÇÃO INTERNA

Quantos funcionários existem por prédios?

```
SELECT p.num_predio, p.quantidade_apartamentos, COUNT(f.cpf) AS total_funcionarios
FROM predios p
INNER JOIN funcionarios f ON p.num_predio = f.num_predio
GROUP BY p.num_predio, p.quantidade_apartamentos;
```

NUM_PREDIO	QUANTIDADE_APARTAMENTOS	TOTAL_FUNCIONARIOS
1	10	2
2	8	3

JUNÇÃO EXTERNA

Quais são os apartamentos que não têm moradores?

```
select a.num_apartamento  
from apartamentos a  
left join moradores m on a.num_apartamento = m.num_apartamento  
where m.num_apartamento is null;
```

NUM_APARTAMENTO
107
108
109
206
105
110
104
204
203
103

SEMI JUNÇÃO

Listar todos os apartamentos que têm serviços contratados

```
SELECT A.num_apartamento  
FROM apartamentos A  
WHERE EXISTS (  
    SELECT *  
    FROM servicos S  
    WHERE A.num_apartamento = S.num_apartamento  
);
```

NUM_APARTAMENTO
101
202

ANTI-JUNÇÃO

Cpf, nome, email de pessoa que não é nem funcionário nem morador

```
SELECT cpf, nome, email
FROM pessoas
WHERE cpf NOT IN (SELECT cpf FROM moradores)
AND cpf NOT IN (SELECT cpf FROM funcionarios);
```

CPF	NOME	EMAIL
56789012345	Paulo Fernando	pf@example.com
67890123456	Lula	fol@example.com
87890123421	Cris	cris@example.com

SUBCONSULTA DO TIPO ESCALAR

Cpf, nome, email de pessoa que não é

```
SELECT metragem  
FROM apartamentos A  
WHERE metragem = (  
    SELECT MAX(metragem)  
    FROM apartamentos  
);
```

METRAGEM
100.2

SUBCONSULTA DO TIPO LINHA

Numero de moradores em cada apartamento

```
SELECT a.num_apartamento,  
       (SELECT COUNT(*)  
        FROM moradores m  
        WHERE m.num_apartamento = a.num_apartamento) AS num_moradores  
FROM apartamentos a;
```

NUM_APARTAMENTO	NUM_MORADORES
101	1
102	0
103	0
104	0

SUBCONSULTA DO TIPO TABELA

CPF, E-mail dos moradores dos prédios

```
select p.nome, p.email  
from pessoas p  
where p.cpf in (select m.cpf from moradores m);
```

NOME	EMAIL
Davi	davi@example.com
Bianca	bianca@example.com
Ana Santos	ana@example.com
João Pedro Pedrosa	joaopedrosa@example.com
Maria Mariana de Jesus	mariamariana@example.com

OPERAÇÃO DE CONJUNTO

CPF das pessoas que não são funcionários

```
SELECT p.cpf  
FROM pessoas p  
MINUS  
SELECT f.cpf  
FROM funcionarios f;
```

CPF
11111111111
12344321567
22222222222
45678901234
56789012345

TRIGGER




Verifica se ainda é possível inserir um apartamento no prédio

```
CREATE OR REPLACE TRIGGER trigger_verificar_limite_apartamentos
BEFORE INSERT ON apartamentos
FOR EACH ROW
DECLARE
    total_apartamentos INT;
    limite_apartamentos INT;
BEGIN
    SELECT COUNT(*) INTO total_apartamentos
    FROM apartamentos
    WHERE num_predio = :NEW.num_predio;

    SELECT quantidade_apartamentos INTO limite_apartamentos
    FROM predios
    WHERE num_predio = :NEW.num_predio;

    IF total_apartamentos >= limite_apartamentos THEN
        RAISE_APPLICATION_ERROR(-20001, 'A adição do novo apartamento ultrapassa o limite permitido no prédio.');
```



```
    END IF;
END;
/
```

TRIGGER




Verifica se ainda é possível inserir um apartamento no prédio

```
193 INSERT INTO apartamentos (num_apartamento, num_comodos, metragem, num_predio) VALUES (209, 3, 80.5, 2);
194
195
```

ORA-20001: A adição do novo apartamento ultrapassa o limite permitido no prédio. ORA-06512: at "SQL_QXIQZLLGPYIFXFMRVOKAMPMHS.TRIGGER_VERIFICAR_LIMITE_APARTAMENTOS", line 15
ORA-06512: at "SYS.DBMS_SQL", line 1721

More Details: <https://docs.oracle.com/error-help/db/ora-20001>



PROCEDURE

atualização dos dados
do
morador

```
1 v CREATE OR REPLACE PROCEDURE atualizar_dados_moradores (  
2     p_cpf VARCHAR2,  
3     p_nome VARCHAR2,  
4     p_email VARCHAR2,  
5     p_num_apartamento NUMBER  
6 )  
7 AS  
8     v_num_moradores NUMBER;  
9 v BEGIN  
10  
11     SELECT COUNT(*)  
12     INTO v_num_moradores  
13     FROM moradores  
14     WHERE cpf = p_cpf;  
15  
16 v     IF v_num_moradores = 0 THEN  
17         DBMS_OUTPUT.PUT_LINE('CPF não encontrado.');18         RETURN;  
19     END IF;  
20  
21 v     UPDATE pessoas  
22     SET nome = p_nome,  
23         email = p_email  
24     WHERE cpf = p_cpf;  
25  
26 v     UPDATE moradores  
27     SET num_apartamento = p_num_apartamento  
28     WHERE cpf = p_cpf;  
29  
30     COMMIT; -- Confirmar a transação  
31 v EXCEPTION  
32     WHEN OTHERS THEN  
33         ROLLBACK;  
34         RAISE;  
35 END;  
36 /
```

PROCEDURE

```
37
38 select * from moradores;
39 select*from pessoas;
40 EXECUTE atualizar_dados_moradores('1111111111', 'João da Silva', 'joo@example.com', 106);
41 |
```

07090123421	Cr 13	Cr 13@example.com	
11111111111	João da Silva	joo@example.com	

```
37
38 select * from moradores;
39 select*from pessoas;
40 EXECUTE atualizar_dados_moradores('1111111111', 'João da Silva', 'joo@example.com', 106);
41
```

CPF	NUM_APARTAMENTO
11111111111	106

The background is a light beige color. It features several abstract geometric elements: a large yellow lightning bolt shape pointing downwards, centered behind the text; various circles in shades of pink, red, blue, and teal, some solid and some outlined; and a series of concentric white circles on the left side. The text 'DÚVIDAS?' is written in a bold, black, sans-serif font, centered horizontally and partially overlaid by the yellow lightning bolt.

DÚVIDAS?