

Guia Prático de SQL: Joins, Consultas e Funções

1. SELECT BÁSICO

```
SELECT coluna1, coluna2  
FROM tabela  
WHERE condicao;
```

O que faz: seleciona colunas de uma tabela com filtros opcionais.

Exemplos:

Buscar nome e idade dos pacientes maiores de 30 anos:
SELECT nome, idade FROM Pacientes WHERE idade > 30;

Buscar todos os funcionários:
SELECT * FROM Funcionarios;

2. JOINS (Junções de Tabelas)

2.1 INNER JOIN

```
SELECT * FROM tabela1  
INNER JOIN tabela2 ON tabela1.coluna = tabela2.coluna;
```

Traz apenas registros que existem nas duas tabelas.

Exemplo:
SELECT m.nome, t.qtd
FROM movies m
INNER JOIN tickets t ON m.id = t.movie_id;

2.2 LEFT JOIN

```
SELECT * FROM tabela1  
LEFT JOIN tabela2 ON tabela1.coluna = tabela2.coluna;
```

Traz todos os dados da tabela1, mesmo sem correspondência na tabela2 (preenche com NULL).

Exemplo:

```
SELECT d.nome, m.titulo  
FROM directors d  
LEFT JOIN movies m ON d.id = m.director_id;
```

2.3 RIGHT JOIN

```
SELECT * FROM tabela1  
RIGHT JOIN tabela2 ON tabela1.coluna = tabela2.coluna;
```

Traz todos os dados da tabela2, com dados da tabela1 (NULL se não houver).

3. FUNÇÕES DE AGREGAÇÃO

Função: COUNT

O que faz: Conta registros

Exemplo: SELECT COUNT(*) FROM Medicos;

Função: SUM

O que faz: Soma valores

Exemplo: SELECT SUM(salario) FROM Funcionarios;

Função: AVG

O que faz: Média

Exemplo: SELECT AVG(idade) FROM Pacientes;

Função: MIN

O que faz: Menor valor

Exemplo: SELECT MIN(salario) FROM Funcionarios;

Função: MAX

O que faz: Maior valor

Exemplo: SELECT MAX(salario) FROM Funcionarios;

Exemplo com filtro:

SELECT COUNT(*) FROM Medicos WHERE especialidade = 'ortopedia';

4. FILTROS COM WHERE

LIKE – nomes que começam com M:

SELECT nome FROM Medicos WHERE nome LIKE 'M%';

BETWEEN – intervalo:

SELECT * FROM Consultas WHERE hora BETWEEN '14:00:00' AND '18:00:00';

IS NULL – onde o valor é nulo:

SELECT nome FROM Medicos WHERE nroa IS NULL;

IN – dentro de uma lista:

SELECT * FROM Medicos WHERE especialidade IN ('ortopedia', 'pediatria');

5. SUBCONSULTAS

5.1 IN

Seleciona se o valor estiver presente em outra consulta:

```
SELECT nome FROM Medicos  
WHERE codm IN (SELECT codm FROM Consultas);
```

5.2 ANY / ALL

Comparação com múltiplos valores:

```
SELECT nome FROM Funcionarios  
WHERE salario > ALL (SELECT salario FROM Funcionarios WHERE cargo =  
'Recepcionista');
```

5.3 EXISTS

Verifica se existe ao menos um resultado:

```
SELECT nome FROM Medicos m  
WHERE EXISTS (SELECT * FROM Consultas c WHERE c.codm = m.codm);
```

6. GROUP BY / HAVING

Agrupamento por especialidade:

```
SELECT especialidade, COUNT(*)  
FROM Medicos  
GROUP BY especialidade;
```

Agrupamento com filtro:

```
SELECT especialidade, COUNT()  
FROM Medicos  
GROUP BY especialidade  
HAVING COUNT() > 1;
```

7. ORDER BY e LIMIT

Ordenar salários decrescentes:

```
SELECT * FROM Funcionarios  
ORDER BY salario DESC;
```

Mostrar os 3 funcionários mais bem pagos:

```
SELECT * FROM Funcionarios  
ORDER BY salario DESC  
LIMIT 3;
```

8. EXEMPLOS PRONTOS (JOINS CLÁSSICOS)

Diretores e seus filmes:

```
SELECT d.nome, m.titulo  
FROM directors d  
JOIN movies m ON d.id = m.director_id;
```

Filmes e tickets vendidos:

```
SELECT m.titulo, t.qtd  
FROM movies m  
JOIN tickets t ON m.id = t.movie_id;
```

Filmes com diretores e tickets:

```
SELECT m.titulo, d.nome, t.qtd  
FROM movies m  
JOIN directors d ON m.director_id = d.id  
JOIN tickets t ON m.id = t.movie_id;
```

Todos os diretores (com ou sem filmes):

```
SELECT d.nome, m.titulo  
FROM directors d  
LEFT JOIN movies m ON d.id = m.director_id;
```

Todos os filmes (com ou sem diretor):

```
SELECT m.titulo, d.nome  
FROM movies m  
LEFT JOIN directors d ON m.director_id = d.id;
```

9. PASSO A PASSO PARA QUALQUER EXERCÍCIO

1. Identifique as tabelas envolvidas.
2. Veja se há relação entre elas (chave estrangeira, etc.).
3. Escolha o tipo de JOIN adequado (INNER, LEFT, etc.).
4. Aplique filtros com WHERE.
5. Use funções de agregação ou GROUP BY, se necessário.
6. Use aliases (AS) para nomes mais claros.
7. Adicione ORDER BY e LIMIT, se quiser ordenar ou limitar os resultados.