

Consultas Básicas com SELECT

As consultas básicas em SQL são realizadas com o comando SELECT, que é usado para recuperar dados de uma ou mais tabelas. A estrutura básica de uma consulta é:

```
SELECT coluna1, coluna2, ...
```

```
FROM tabela;
```

Exemplo:

Para listar todos os clientes cadastrados em uma tabela cliente:

```
SELECT cliente_cod, cliente_nome, cidade
```

```
FROM cliente;
```

Essa consulta retorna todas as linhas da tabela cliente, mostrando o código, o nome e a cidade de cada cliente.

Filtros e Ordenação

Para refinar os dados retornados por uma consulta, utilizamos filtros com a cláusula WHERE. Além disso, a ordenação dos resultados é feita com ORDER BY, que pode organizar os dados em ordem crescente (ASC) ou decrescente (DESC).

Exemplo de Filtro:

Listar clientes que moram na cidade de "Niterói":

```
SELECT cliente_nome, cidade
```

```
FROM cliente
```

```
WHERE cidade = 'Niterói';
```

Exemplo de Ordenação:

Listar agências ordenadas pelos fundos, da maior para a menor:

```
SELECT agencia_nome, fundos
```

```
FROM agencia
```

```
ORDER BY fundos DESC;
```

Neste exemplo, as agências são listadas em ordem decrescente de acordo com o valor dos fundos.

Junções (JOINS)

As junções permitem combinar dados de duas ou mais tabelas com base em uma condição de relacionamento. Os tipos mais comuns de junção incluem:

- **INNER JOIN:** Retorna apenas as linhas que têm correspondência em ambas as tabelas.
- **LEFT JOIN:** Retorna todas as linhas da tabela da esquerda, e as correspondências da tabela da direita (ou NULL se não houver correspondência).
- **RIGHT JOIN:** Retorna todas as linhas da tabela da direita e as correspondências da tabela da esquerda (ou NULL se não houver correspondência).
- **FULL JOIN:** Retorna todas as linhas quando há uma correspondência em uma das tabelas.

Exemplo de INNER JOIN:

Listar clientes e o nome da agência onde possuem uma conta:

```
SELECT cliente.cliente_nome, agencia.agencia_nome, conta.saldo
```

```
FROM cliente
```

```
JOIN conta ON cliente.cliente_cod = conta.cliente_cliente_cod
```

```
JOIN agencia ON conta.agencia_agencia_cod = agencia.agencia_cod;
```

Neste exemplo, os dados das tabelas cliente, conta e agencia são combinados para mostrar o nome do cliente, o nome da agência e o saldo da conta.

Funções de Agregação

As funções de agregação permitem calcular valores resumidos em conjunto de dados, como somas, médias, contagens, entre outros. As principais funções de agregação em incluem:

- **SUM():** Retorna a soma de um conjunto de valores.
- **AVG():** Calcula a média dos valores.
- **COUNT():** Conta o número de linhas ou valores distintos.
- **MIN() e MAX():** Retornam o menor e o maior valor de uma coluna, respectivamente.

Essas funções são frequentemente usadas em conjunto com GROUP BY, que agrupa os dados antes de aplicar a função de agregação.

Exemplo:

Calcular o saldo médio das contas:

```
SELECT AVG(saldo) AS saldo_medio  
FROM conta;
```

Esse comando calcula a média dos saldos da tabela conta.

Exemplo com GROUP BY:

Contar o número de clientes por cidade:

```
SELECT cidade, COUNT(cliente_cod) AS num_clientes  
FROM cliente  
GROUP BY cidade;
```

Aqui, os clientes são agrupados por cidade, e a contagem é realizada para cada cidade.

Subconsultas

Subconsultas são consultas aninhadas dentro de outras consultas e são úteis para realizar operações em etapas, como filtros adicionais ou cálculos intermediários. Elas podem ser usadas em várias partes de uma consulta, como WHERE, HAVING, ou mesmo SELECT.

Exemplo:

Encontrar agências com fundos acima da média de todas as agências:

```
SELECT agencia_cod, agencia_nome, fundos  
FROM agencia  
WHERE fundos > (  
    SELECT AVG(fundos)  
    FROM agencia  
);
```

Neste exemplo, a subconsulta calcula a média dos fundos de todas as agências, e a consulta externa exibe apenas as agências com fundos acima dessa média.

Operações de Atualização

Para modificar dados existentes em uma tabela, utilizamos o comando UPDATE. Este comando permite alterar valores de colunas com base em uma condição.

Estrutura Básica:

UPDATE tabela

SET coluna1 = valor1, coluna2 = valor2, ...

WHERE condição;

Exemplo:

Aumentar o saldo de todas as contas em 5%:

UPDATE conta

SET saldo = saldo * 1.05;

Este comando aplica um aumento de 5% ao saldo de todas as contas.

Exemplo com Condição:

Atualizar a cidade de um cliente específico:

UPDATE cliente

SET cidade = 'São Paulo'

WHERE cliente_cod = 5;

Neste caso, apenas o cliente com código 5 terá a cidade alterada para "São Paulo".

Exclusão de Dados

Para remover registros de uma tabela, utilizamos o comando DELETE. A exclusão pode ser feita para uma linha específica ou para várias linhas que atendem a uma condição. **Cuidado:** O comando DELETE é irreversível, portanto, deve ser usado com atenção.

Estrutura Básica:

DELETE FROM tabela

WHERE condição;

Exemplo:

Excluir todas as contas com saldo zerado:

```
DELETE FROM conta
```

```
WHERE saldo = 0;
```

Exemplo com Subconsulta:

Excluir clientes que não possuem contas:

```
DELETE FROM cliente
```

```
WHERE cliente_cod NOT IN (
```

```
    SELECT cliente_cliente_cod
```

```
    FROM conta
```

```
);
```

Neste exemplo, a subconsulta identifica os clientes que possuem contas, e a consulta principal exclui os clientes que não estão nesta lista.