

## Aula: Comandos INSERT e SELECT no MySQL

### Objetivos:

1. Compreender o uso do comando **INSERT** para adicionar novos registros em tabelas.
  2. Aprender a utilizar o comando **SELECT** para consultar e recuperar dados de tabelas.
  3. Aplicar filtros e ordenações em consultas para obter resultados específicos.
- 

## 1. Comando INSERT

### 1.1 Introdução ao **INSERT**

- **Descrição:** O comando **INSERT** é utilizado para adicionar novos registros em uma tabela do banco de dados.

#### Sintaxe Básica:

sql

Copiar código

```
INSERT INTO nome_da_tabela (coluna1, coluna2, coluna3, ...)
VALUES (valor1, valor2, valor3, ...);
```

- 

#### Exemplo:

Suponha que temos uma tabela chamada **clientes**:

sql

Copiar código

```
CREATE TABLE clientes (
    id INT AUTO_INCREMENT,
    nome VARCHAR(100),
    email VARCHAR(100),
    PRIMARY KEY (id)
);
```

Para inserir um novo cliente na tabela:

sql

Copiar código

```
INSERT INTO clientes (nome, email)
VALUES ('João Silva', 'joao.silva@example.com');
```

- 

### 1.2 Inserir Múltiplos Registros

**Sintaxe:**

sql

Copiar código

```
INSERT INTO nome_da_tabela (coluna1, coluna2, coluna3, ...)
VALUES (valor1, valor2, valor3, ...),
       (valor4, valor5, valor6, ...),
       ...;
```

•

**Exemplo:**

sql

Copiar código

```
INSERT INTO clientes (nome, email)
VALUES ('Maria Oliveira', 'maria.oliveira@example.com'),
       ('Pedro Santos', 'pedro.santos@example.com');
```

•

### 1.3 Inserir Dados com Subconsulta

**Sintaxe:**

sql

Copiar código

```
INSERT INTO nome_da_tabela (coluna1, coluna2, ...)
SELECT valor1, valor2, ...
FROM outra_tabela
WHERE condição;
```

•

**Exemplo:**

Suponha que temos uma tabela `novos_clientes` com os mesmos campos da tabela `clientes`:

sql

Copiar código

```
INSERT INTO clientes (nome, email)
SELECT nome, email
FROM novos_clientes
WHERE data_admissao > '2024-01-01';
```

•

---

## 2. Comando SELECT

## 2.1 Introdução ao **SELECT**

- **Descrição:** O comando **SELECT** é usado para consultar e recuperar dados de uma ou mais tabelas.

### Sintaxe Básica:

sql

Copiar código

```
SELECT coluna1, coluna2, coluna3, ...  
FROM nome_da_tabela;
```

•

### Exemplo:

Para recuperar todos os registros da tabela **clientes**:

sql

Copiar código

```
SELECT * FROM clientes;
```

- Aqui, **\*** significa "todas as colunas".

## 2.2 Filtrar Resultados com **WHERE**

### Sintaxe:

sql

Copiar código

```
SELECT coluna1, coluna2, ...  
FROM nome_da_tabela  
WHERE condição;
```

•

### Exemplo:

Para recuperar clientes com o nome "João Silva":

sql

Copiar código

```
SELECT * FROM clientes  
WHERE nome = 'João Silva';
```

•

## 2.3 Ordenar Resultados com **ORDER BY**

### Sintaxe:

sql

Copiar código

```
SELECT coluna1, coluna2, ...  
FROM nome_da_tabela
```

```
ORDER BY coluna1 [ASC|DESC];
```

- 

**Exemplo:**

Para recuperar todos os clientes ordenados pelo nome em ordem alfabética:

sql

Copiar código

```
SELECT * FROM clientes  
ORDER BY nome ASC;
```

- 

## 2.4 Limitar Resultados com LIMIT

**Sintaxe:**

sql

Copiar código

```
SELECT coluna1, coluna2, ...  
FROM nome_da_tabela  
LIMIT número;
```

- 

**Exemplo:**

Para recuperar os primeiros 5 clientes:

sql

Copiar código

```
SELECT * FROM clientes  
LIMIT 5;
```

- 

## 2.5 Agregar Dados com Funções de Agregação

- **Funções de Agregação Comuns:**
  - `COUNT()`: Conta o número de registros.
  - `SUM()`: Soma os valores de uma coluna.
  - `AVG()`: Calcula a média dos valores de uma coluna.
  - `MAX()`: Encontra o valor máximo em uma coluna.
  - `MIN()`: Encontra o valor mínimo em uma coluna.

**Exemplo:**

Para contar o número total de clientes:

sql

Copiar código

```
SELECT COUNT(*) AS total_clientes
```

```
FROM clientes;
```

- 

---

## Exercícios Práticos

### 1. Inserir Dados:

- Crie uma tabela **produtos** e insira ao menos 3 registros com diferentes valores.

sql

Copiar código

```
CREATE TABLE produtos (  
    id INT AUTO_INCREMENT,  
    nome VARCHAR(100),  
    preco DECIMAL(10,2),  
    PRIMARY KEY (id)  
);
```

```
INSERT INTO produtos (nome, preco)  
VALUES ('Produto A', 19.99),  
      ('Produto B', 29.99),  
      ('Produto C', 39.99);
```

2.

### 3. Consulta Simples:

- Recupere todos os registros da tabela **produtos**.

sql

Copiar código

```
SELECT * FROM produtos;
```

4.

### 5. Filtragem e Ordenação:

- Recupere produtos com preço superior a 20 e ordene os resultados pelo preço em ordem decrescente.

sql

Copiar código

```
SELECT * FROM produtos  
WHERE preco > 20  
ORDER BY preco DESC;
```

6.

### 7. Limitação de Resultados:

- Recupere apenas os 2 primeiros produtos da tabela `produtos`.

sql

Copiar código

```
SELECT * FROM produtos  
LIMIT 2;
```

8.

9. **Agregação:**

- Suponha que você tenha uma tabela `categorias` e uma tabela `produtos` com uma coluna `categoria_id`. Recupere o nome da categoria e o total de produtos em cada categoria.