

# Criando um Banco de Dados para uma Escola

Este guia passo a passo mostrará como criar um banco de dados simples para uma escola, usando o SQL. Abordaremos a criação do banco de dados, tabelas, inserção de dados e consultas (queries) para recuperar informações relevantes.

# Criando o Banco de Dados

Começaremos criando o banco de dados chamado "escola". O comando a ser utilizado varia de acordo com o sistema de gerenciamento de banco de dados (SGBD) que você está usando. Aqui estão exemplos para alguns dos SGBDs mais populares:

- MySQL: `CREATE DATABASE escola;`
- PostgreSQL: `CREATE DATABASE escola;`
- SQLite: `CREATE DATABASE escola;`

Após executar este comando, você terá um banco de dados chamado "escola" pronto para armazenar informações sobre seus alunos e suas notas.

# Criando a Tabela "Aluno"

Agora vamos criar a tabela "aluno" para armazenar dados sobre cada aluno. Essa tabela terá quatro campos: id\_aluno, nome, idade e email.

O campo id\_aluno será a chave primária, garantindo a unicidade de cada registro de aluno. Os demais campos armazenarão o nome, idade e endereço de email do aluno.

O comando para criar essa tabela no MySQL é:

```
CREATE TABLE aluno (  
  id_aluno INT PRIMARY KEY,  
  nome VARCHAR(255),  
  idade INT,  
  email VARCHAR(255)  
);
```

Você pode adaptar o tamanho dos campos VARCHAR de acordo com suas necessidades. Por exemplo, se você precisar armazenar emails mais longos, poderá aumentar o tamanho do campo email.

# Criando a Tabela "Nota"

Agora vamos criar a tabela "nota" para armazenar as notas dos alunos. Essa tabela terá quatro campos: id\_nota, id\_aluno, disciplina e nota.

O campo id\_nota será a chave primária. O campo id\_aluno será uma chave estrangeira, referenciando a tabela "aluno" e garantindo a integridade dos dados, pois um aluno pode ter várias notas. Os campos disciplina e nota armazenarão o nome da disciplina e a nota obtida pelo aluno, respectivamente.

O comando para criar essa tabela no MySQL é:

```
CREATE TABLE nota (  
  id_nota INT PRIMARY KEY,  
  id_aluno INT,  
  disciplina VARCHAR(255),  
  nota DECIMAL(4,2),  
  FOREIGN KEY (id_aluno) REFERENCES aluno(id_aluno)  
);
```

O tipo DECIMAL(4,2) para o campo nota permite armazenar valores decimais com 2 casas decimais, como 8.50 ou 7.00, que são típicos para notas.

# Inserindo Dados nas Tabelas

Agora vamos inserir alguns dados nas tabelas "aluno" e "nota" para termos dados para consultar. Começaremos pela tabela "aluno".

Para inserir os dados dos alunos João Silva, Maria Oliveira e Carlos Souza, utilizaremos o comando INSERT INTO no MySQL:

```
INSERT INTO aluno (id_aluno, nome, idade, email) VALUES  
(1, 'João Silva', 20, 'joao.silva@example.com'),  
(2, 'Maria Oliveira', 22, 'maria.oliveira@example.com'),  
(3, 'Carlos Souza', 19, 'carlos.souza@example.com');
```

Lembre-se de que você pode modificar os valores de idade e email de acordo com seus dados.

# Inserindo Dados na Tabela "Nota"

Agora vamos inserir as notas dos alunos na tabela "nota". Para inserir as notas de João, Maria e Carlos, utilizaremos o comando INSERT INTO no MySQL:

```
INSERT INTO nota (id_nota, id_aluno, disciplina, nota) VALUES  
(1, 1, 'Matemática', 8.5),  
(2, 1, 'História', 7.0),  
(3, 2, 'Matemática', 9.0),  
(4, 2, 'História', 6.5),  
(5, 3, 'Matemática', 5.5),  
(6, 3, 'História', 8.0);
```

Lembre-se de que você pode modificar os valores de nota e disciplina de acordo com seus dados.

# Consultas (Queries) no Banco de Dados

Agora que temos dados em nosso banco de dados, podemos começar a realizar consultas para recuperar as informações que precisamos.

Para mostrar todos os alunos, podemos usar o comando SELECT no MySQL:

```
SELECT * FROM aluno;
```

Para mostrar todas as notas, podemos usar o comando SELECT no MySQL:

```
SELECT * FROM nota;
```

Para mostrar o nome dos alunos e todos os dados de nota, podemos usar o comando SELECT com JOIN no MySQL:

```
SELECT a.nome, n.disciplina, n.nota  
FROM aluno a  
JOIN nota n ON a.id_aluno = n.id_aluno;
```

# Consultas (Queries) com Filtros

Podemos usar a cláusula WHERE para filtrar os resultados das nossas consultas.

Para mostrar o nome e idade dos alunos e todos os dados de nota, MAS DESDE QUE A IDADE DO ALUNO SEJA >=20, podemos usar o comando SELECT com JOIN e WHERE no MySQL:

```
SELECT a.nome, a.idade, n.disciplina, n.nota  
FROM aluno a  
JOIN nota n ON a.id_aluno = n.id_aluno  
WHERE a.idade >= 20;
```

Com essas consultas, você pode recuperar informações específicas do banco de dados e utilizá-las para análise e tomada de decisão.