

# Exercícios SQL

1. Crie uma tabela chamada "alunos" com os seguintes campos: id (inteiro), nome (texto), idade (inteiro) e curso (texto).

```
create table alunos (  
    id integer,  
    nome varchar(50),  
    idade integer,  
    curso varchar(50)  
);
```

2. Insira pelo menos 5 registros de alunos na tabela que você criou no exercício anterior.

```
insert into alunos (id, nome, idade, curso)  
values ('1', 'Maria', 31, 'Moda'),  
      ('2', 'João', 13, 'Artes'),  
      ('3', 'Pedro', 32, 'Moda'),  
      ('4', 'Ana', 35, 'Mecânica'),  
      ('5', 'Zé', 65, 'Jardinagem');
```

3. Consultas Básicas

Escreva consultas SQL para realizar as seguintes tarefas:

- a) Selecionar todos os registros da tabela "alunos".

```
select * from alunos;
```

- b) Selecionar o nome e a idade dos alunos com mais de 20 anos.**

```
select  
    nome,  
    idade  
from
```

```
alunos
where
    idade > 20;
```

c) Selecionar os alunos do curso de "Engenharia" em ordem alfabética.

```
select
    *
from
    alunos
where
    curso = 'Engenharia'
order by
    nome;
```

d) Contar o número total de alunos na tabela

```
select count(*) from alunos;
```

#### 4. Atualização e Remoção

a) Atualize a idade de um aluno específico na tabela.

```
update alunos
set idade = 50
where id = 4;
```

b) Remova um aluno pelo seu ID.

```
delete from alunos
where id = 3;
```

#### 5. Criar uma Tabela e Inserir Dados

Crie uma tabela chamada "clientes" com os campos: id (chave primária), nome (texto), idade (inteiro) e saldo (float). Insira alguns

registros de clientes na tabela.

```
create table clientes (  
    id serial primary key,  
    nome varchar(50),  
    idade integer,  
    saldo float  
);  
  
insert into clientes (id, nome, idade, saldo)  
values ('1', 'Maria', 31, 350.00),  
       ('2', 'João', 13, 345.00),  
       ('3', 'Pedro', 32, 234.00),  
       ('4', 'Ana', 35, 45.00),  
       ('5', 'Zé', 65, 3493.00);
```

## 6. Consultas e Funções Agregadas

Escreva consultas SQL para realizar as seguintes tarefas:

a) Selecione o nome e a idade dos clientes com idade superior a 30 anos.

```
select nome, idade from clientes  
where idade > 30;
```

b) Calcule o saldo médio dos clientes.

```
select nome, avg(saldo) from clientes  
group by nome;
```

c) Encontre o cliente com o saldo máximo.

```
select nome, saldo from clientes  
order by saldo desc
```

```
limit 1;
```

d) Conte quantos clientes têm saldo acima de 1000.

```
select count(*) from clientes  
where saldo > 1000;
```

## 7. Atualização e Remoção com Condições

a) Atualize o saldo de um cliente específico.

```
update clientes  
set saldo = 3333.45  
where id = 2;
```

b) Remova um cliente pelo seu ID.

```
delete from clientes  
where id = 5;
```

## 8. Junção de Tabelas

Crie uma segunda tabela chamada "compras" com os campos: id (chave primária), cliente\_id (chave estrangeira referenciando o id da tabela "clientes"), produto (texto) e valor (real). Insira algumas compras associadas a clientes existentes na tabela "clientes". Escreva uma consulta para exibir o nome do cliente, o produto e o valor de cada compra.

```
create table compras (  
    id_compras SERIAL primary key,  
    cliente_id INTEGER references clientes (id),  
    produto VARCHAR(50),  
    valor real  
);  
  
insert into compras (id_compras, cliente_id, produto, valor)
```

```
values  ('1', '3', 'Vassoura', 3.00),
        ('2', '1', 'Rodo', 5.00),
        ('3', '1', 'Pá', 4.00),
        ('4', '2', 'Cesto', 5.00),
        ('5', '2', 'Saco de Lixo', 3.00);

select c.nome, s.produto, s.valor
from clientes c
left join compras s on s.cliente_id = c.id
```