

Unidade: Documentação de SQL

Desenvolvimento de SQL

Se você não conhece SQL, saiba mais por meio deste curso: [SQL](#).

Criando tabelas

```
CREATE TABLE customers (id  
INTEGER PRIMARY KEY, name  
TEXT, age INTEGER, weight  
REAL);
```

Vários tipos de dados

```
CREATE TABLE customers (id  
INTEGER PRIMARY KEY, age  
INTEGER);
```

Usar chaves primárias

Veja também: [especificar padrões](#), [usar chaves estrangeiras](#). Para ver mais detalhes, consulte: [Referência de SQLite para CREATE](#).

Inserção de dados

```
INSERT INTO customers VALUES  
(73, "Brian", 33);
```

Inserção de dados

```
INSERT INTO customers (name,  
age) VALUES ("Brian", 33);
```

Inserção de dados
para colunas
nomeadas

Veja também: [Referência de SQLite para INSERT](#).

Consultando dados

```
SELECT * FROM customers;
```

Selecionar tudo

```
SELECT * FROM customers WHERE  
age > 21;
```

Filtrar com condição

```
SELECT * FROM customers WHERE  
age < 21 AND state = "NY";
```

Filtrar com várias
condições

```
SELECT * FROM customers WHERE  
plan IN ("free", "basic");
```

Filtrar com IN

```
SELECT name, age FROM  
customers;
```

Selecionar colunas
específicas

```
SELECT * FROM customers WHERE  
age > 21 ORDER BY age DESC;
```

Ordenar resultados

```
SELECT name, CASE WHEN age >
18 THEN "adult" ELSE "minor"
END "type" FROM customers;
```

Transformar com
CASE

Veja também: [filtrar com LIKE](#), [restringir com LIMIT](#), [usar ROUND](#) e [outras funções importantes](#). Para ver mais detalhes, consulte: [referência de SQLite para SELECT](#).

Agregando dados

```
SELECT MAX(age) FROM
customers;
```

Agregar funções

```
SELECT gender, COUNT(*) FROM
students GROUP BY gender;
```

Agrupamento de
dados

Veja também: [restringir resultados com HAVING](#).

Associando tabelas relacionadas

```
SELECT customers.name,
orders.item FROM customers
JOIN orders ON customers.id =
orders.customer_id;
```

Inner join

```
SELECT customers.name,
orders.item FROM customers
LEFT OUTER JOIN orders ON
customers.id =
orders.customer_id;
```

Outer join

Atualização e exclusão de dados

```
UPDATE customers SET age = 33
WHERE id = 73;
```

Atualização de dados

```
DELETE FROM customers WHERE
id = 73;
```

Exclusão de dados

Veja também: [ALTER TABLE](#).

Nossa implementação de SQL baseia-se em um dialeto popular de SQL chamado de [SQLite](#). Para executar o SQL no navegador, usamos as seguintes tecnologias:

[asm.js](#)
[Emscripten](#)
[sql.js](#)