**Observações:**

* Para realização dos exercícios é necessário que você baixe o sqlite e coloque-o em sua área de trabalho. Depois você precisará abrir o cmd para executar com o nome correto do banco de dados conforme pedido.
* Pode pesquisar a vontade e conversar com seus colegas. O Google é seu amigo.
* Não reclame que não sabe. Pesquise, tente, teste e refaça! Mude sua postura de “Não sei fazer” para “Vou descobrir como se faz”. **Anote a resposta de TODOS os exercícios**. Irá te ajudar lá na frente!
* Não pule nenhum exercício achando que “já sabe”.
* Nomes de tabelas, campos e banco de dados NÃO devem possuir acentos e cedilha.
* Lembre-se que o “;” finaliza os comandos.
* Os tipos de dados basicamente são: int para inteiros, text para texto, real para números com decimais, date para datas e boolean para booleanos. Você também pode usar para campos texto o tipo varchar especificando o tamanho. Para valores monetários você pode usar o tipo real.
* **Sua missão é resolver todas as questõsqlitees**. Não pare antes disso.

1. Execute o sqlite criando um banco de dados de nome lista1.sqlite.

polly@familia-of-02:~/Downloads/SQLITE/sqlite-tools-linux-x86-3400100$ sqlite3 lista1.sqlite

1. Crie uma tabela com o nome de alunos. Deverá conter o campo código (inteiro), nome, telefone e cidade (texto). Vou te ajudar nessa: CREATE TABLE alunos (codigo int, nome text, telefone text, cidade text);

CREATE TABLE alunos (codigo integer, nome text, telefone text, cidade text);

1. Use o comando .tables para verificar se a tabela foi criada

sqlite> .schema

CREATE TABLE alunos (codigo integer, nome text, telefone text, cidade text);

1. Crie uma tabela com o nome de alunos2. Deverá conter o campo código (inteiro), nome (varchar de tamanho 200), telefone (varchar de tamanho 50)e cidade (varchar de tamanho 100). Vou te ajudar nessa também. O comando ficará assim: CREATE TABLE alunos2 (codigo int, nome varchar(200), telefone varchar(50), cidade varchar(100) );

CREATE TABLE alunos2 (codigo integer, nome varchar(200), telefone varchar(50), cidade varchar(100));

1. Crie a tabela funcionários contendo os campos nome, endereço, telefone, cidade, estado, cep, rg, cpf e salário. Coloque os tipos de dados necessários.

CREATE TABLE funcionarios (nome varchar(200), endereco varchar(200), telefone varchar(50), cidade varchar(100), estado varchar(50), cep varchar(12), rg varchar(20), cpf varchar(20), salario integer);

1. Saia do sqlite com o comando .exit.

.exit

1. Abra novamente no sqlite o banco lista1.sqlite.

sqlite3 lista1.sqlite

1. Verifique se as tabelas ainda existem com o comando .tables

sqlite> .schema

CREATE TABLE alunos (codigo integer, nome text, telefone text, cidade text);

CREATE TABLE alunos2 (codigo integer, nome varchar(200), telefone varchar(50), cidade varchar(100));

CREATE TABLE funcionarios (nome varchar(200), endereco varchar(200), telefone varchar(50), cidade varchar(100), estado varchar(50), cep varchar(12), rg varchar(20), cpf varchar(20), salario integer);

1. Crie a tabela fornecedores contendo os campos nome, endereço, telefone, cidade, estado, cep, cnpj e email. Coloque os tipos de dados necessários.

CREATE TABLE fornecedore (nome varchar(200), endereco varchar(200), telefone varchar(50), cidade varchar(100), estado varchar(50), cep varchar(12), cnpj varchar(30), email varchar(50));

sqlite> alter table fornecedore rename to fornecedores;

sqlite> .schema

1. Crie a tabela livros contendo o campo código, nome, categoria, resumo, precocusto, precovenda.

CREATE TABLE livros (codigo integer, nome varchar(200), categoria varchar(20), resumo varchar(300), precocusto integer, precovenda integer);

1. Existe uma maneira de verificar o ESQUEMA da tabela, ou seja, sua estrutura. É o comando .SCHEMA.

sqlite> .schema

CREATE TABLE alunos (codigo integer, nome text, telefone text, cidade text);

CREATE TABLE alunos2 (codigo integer, nome varchar(200), telefone varchar(50), cidade varchar(100));

CREATE TABLE funcionarios (nome varchar(200), endereco varchar(200), telefone varchar(50), cidade varchar(100), estado varchar(50), cep varchar(12), rg varchar(20), cpf varchar(20), salario integer);

CREATE TABLE livros (codigo integer, nome varchar(200), categoria varchar(20), resumo varchar(300), precocusto integer, precovenda integer);

CREATE TABLE fornecedores (nome varchar(200), endereco varchar(200), telefone varchar(50), cidade varchar(100), estado varchar(50), cep varchar(12), cnpj varchar(30), email varchar(50));

1. Crie a tabela estoque contendo o campo código, nomedoproduto, categoria, quantidade e fornecedor

sqlite> create table estoque (codigo integer, nomedoproduto varchar(100), categoria varchar(20), quantidade integer, fornecedor varchar(200));

1. Crie a tabela notas contendo os campos código, nomedoaluno, 1bim, 2bim, 3bim e 4bim

sqlite> create table notas (codigo integer, nomedoaluno varchar(100), primeirobim integer, segundobim integer, terceirobim integer, quartobim integer);

1. Crie a tabela caixa contendo os campos código, data, descrição, debito e credito.

sqlite> create table caixa (codigo integer, data date, descricao varchar(300), debito integer, credito integer);

1. Crie a tabela contasAPagar contendo os campos código, data\_conta, descrição, valor e data\_pagamento.

sqlite> create table contasAPagar (codigo integer, data\_conta date, descricao varchar(300), valor integer, data\_pagamento date);

1. Crie a tabela contasAReceber contendo os campos código, data\_conta, descrição, valor e data\_recebimento.

sqlite> create table contasAReceber (codigo integer, data\_conta date, descricao varchar(300), valor integer, data\_recebimento date);

1. Crie a tabela filmes contendo os campos código, nome, sinopse, categoria e diretor

sqlite> create table filmes (codigo integer, nome varchar(100), sinopse varchar(200), categoria varchar(20), diretor varchar(100));

1. Crie a tabela CDs contendo os campos código, nome, cantor, ano e quantidademusicas.

sqlite> create table CDs (codigo integer, nome varchar(100), cantor varchar(100), ano integer, quantidadedemusicas integer);

1. Agora iremos aprender a excluir tabelas. É muito fácil. Basta usar o comando DROP TABLE. Se quero excluir a tabela alunos o comando fica assim: **Drop table alunos;**. Faça isso então, exclua a tabela alunos.

sqlite> .schema

CREATE TABLE alunos (codigo integer, nome text, telefone text, cidade text);

CREATE TABLE alunos2 (codigo integer, nome varchar(200), telefone varchar(50), cidade varchar(100));

CREATE TABLE funcionarios (nome varchar(200), endereco varchar(200), telefone varchar(50), cidade varchar(100), estado varchar(50), cep varchar(12), rg varchar(20), cpf varchar(20), salario integer);

CREATE TABLE livros (codigo integer, nome varchar(200), categoria varchar(20), resumo varchar(300), precocusto integer, precovenda integer);

CREATE TABLE fornecedores (nome varchar(200), endereco varchar(200), telefone varchar(50), cidade varchar(100), estado varchar(50), cep varchar(12), cnpj varchar(30), email varchar(50));

CREATE TABLE estoque (codigo integer, nomedoproduto varchar(100), categoria varchar(20), quantidade integer, fornecedor varchar(200));

CREATE TABLE notas (codigo integer, nomedoaluno varchar(100), primeirobim integer, segundobim integer, terceirobim integer, quartobim integer);

CREATE TABLE caixa (codigo integer, data date, descricao varchar(300), debito integer, credito integer);

CREATE TABLE contasAPagar (codigo integer, data\_conta date, descricao varchar(300), valor integer, data\_pagamento date);

CREATE TABLE contasAReceber (codigo integer, data\_conta date, descricao varchar(300), valor integer, data\_recebimento date);

CREATE TABLE filmes (codigo integer, nome varchar(100), sinopse varchar(200), categoria varchar(20), diretor varchar(100));

CREATE TABLE CDs (codigo integer, nome varchar(100), cantor varchar(100), ano integer, quantidadedemusicas integer);

1. Use o comando .tables e veja se a tabela realmente foi excluída

sqlite> drop table alunos;

sqlite> .tables

CDs contasAPagar filmes livros

alunos2 contasAReceber fornecedores notas

caixa estoque funcionarios

1. Exclua a tabela livros.

sqlite> drop table livros;

sqlite> .tables

CDs caixa contasAReceber filmes funcionarios

alunos2 contasAPagar estoque fornecedores notas

1. Exclua a tabela contasAPagar.

sqlite> drop table contasAPagar;

sqlite> .tables

CDs caixa estoque fornecedores notas

alunos2 contasAReceber filmes funcionarios

1. Exclua também a tabela contasAReceber.

sqlite> drop table contasAReceber;

sqlite> .tables

CDs caixa filmes funcionarios

alunos2 estoque fornecedores notas

1. Agora apague a tabela filmes.

sqlite> drop table filmes;

sqlite> .tables

CDs caixa fornecedores notas

alunos2 estoque funcionarios

1. Liste as tabelas e veja se a tabela alunos2 ainda existe.

sqlite> .tables

CDs caixa fornecedores notas

alunos2 estoque funcionarios

1. Agora iremos aprender como MUDAR O NOME das tabelas. É fácil, basta usar o comando ALTER TABLE. Por exemplo se quisermos mudar o nome da tabela NOMEFEIO para NOMEBONITO o comando ficará assim: ALTER TABLE NOMEFEIO RENAME TO NOMEBONITO; - Agora que você sabe disse renomeie a tabela alunos para super\_alunos

sqlite> alter table alunos2 rename to super\_alunos;

sqlite> .tables

CDs estoque funcionarios super\_alunos

caixa fornecedores notas

1. Use o comando .tables e veja se foi alterado o nome.

sqlite> .tables

CDs estoque funcionarios super\_alunos

caixa fornecedores notas

1. Altere o nome da tabela estoque para produtos.

sqlite> alter table estoque rename to produtos;

sqlite> .tables

CDs fornecedores notas super\_alunos

caixa funcionarios produtos

1. Altere o nome da tabela notas para aprovados.

sqlite> alter table notas rename to aprovados;

sqlite> .tables

CDs caixa funcionarios super\_alunos

aprovados fornecedores produtos

1. Altere o nome da tabela aprovados para notas.

sqlite> alter table aprovados rename to notas;

sqlite> .tables

CDs fornecedores notas super\_alunos

caixa funcionarios produtos

sqlite>