

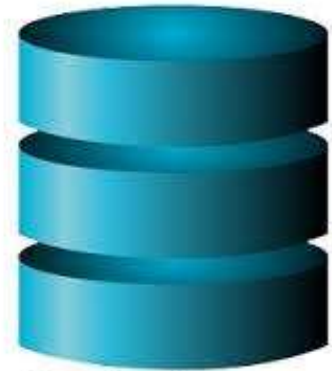


JavaScript

Banco de Dados



SQLite



Database



Abra o VsCode;

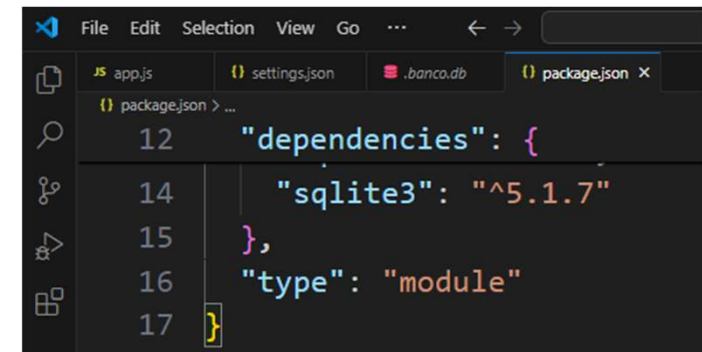
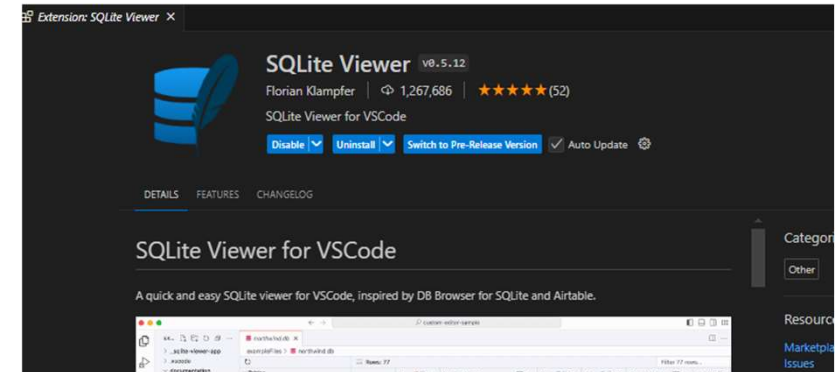
<<diretório local “pasta nova”>>

1 - *npm init -y;*

2 - *npm install sqlite3;*

3 - *npm install sqlite;*

O código abaixo é um exemplo de como criar e popular uma tabela de usuários em um banco de dados SQLite utilizando Node.js:





Importação de Módulos

```
import { open } from 'sqlite';  
import sqlite3 from 'sqlite3';
```

Essas duas linhas importam os módulos necessários para trabalhar com o banco de dados SQLite. O módulo `sqlite` fornece uma interface para interagir com o banco de dados, enquanto o módulo `sqlite3` é o driver que permite que o Node.js se comunique com o banco de dados.



Função criarEPopularTabelaUsuarios

```
async function criarEPopularTabelaUsuarios(nome, sobrenome){  
  // ...  
}
```

Essa função é responsável por criar a tabela de usuários e popular com os dados fornecidos. Ela é uma função assíncrona, o que significa que ela retorna uma promessa que é resolvida quando a operação é concluída.



Abertura do Banco de Dados

```
const db = await open({  
    filename: './banco.db',  
    driver: sqlite3.Database,  
});
```

Essa linha abre o banco de dados SQLite localizado no arquivo ./banco.db. O objeto open é uma função que retorna uma promessa que é resolvida quando o banco de dados é aberto com sucesso. O objeto driver é setado para sqlite3.Database, que é o driver que permite que o Node.js se comunicar com o banco de dados.



Criação da Tabela de Usuários

```
await db.run(
  'CREATE TABLE IF NOT EXISTS usuarios (id INTEGER PRIMARY KEY,
  nome TEXT, sobrenome TEXT)'
);
```

Essa linha cria a tabela de usuários se ela não existir. A instrução CREATE TABLE é usada para criar a tabela, e a cláusula IF NOT EXISTS garante que a tabela não seja criada se ela já existir. A tabela tem três colunas: id, nome e sobrenome



Inserção de Dados na Tabela de Usuários

```
await db.run(  
  'INSERT INTO usuarios (nome, sobrenome) VALUES (?,?)', [  
    nome,  
    sobrenome,  
  ]);
```

Essa linha insere os dados fornecidos na tabela de usuários. A instrução INSERT INTO é usada para inserir os dados, e os valores são passados como parâmetros para evitar ataques de SQL injection.



Fechamento do Banco de Dados

await db.close();

Essa linha fecha o banco de dados após as operações serem concluídas.

criarEPopularTabelaUsuarios('seu nome', 'Sobrenome');

Essa linha transporta a informação dentro do parêntese para a tabela



Execute o código;

<<diretório local “pasta nova”>>

..

4 – node ./app