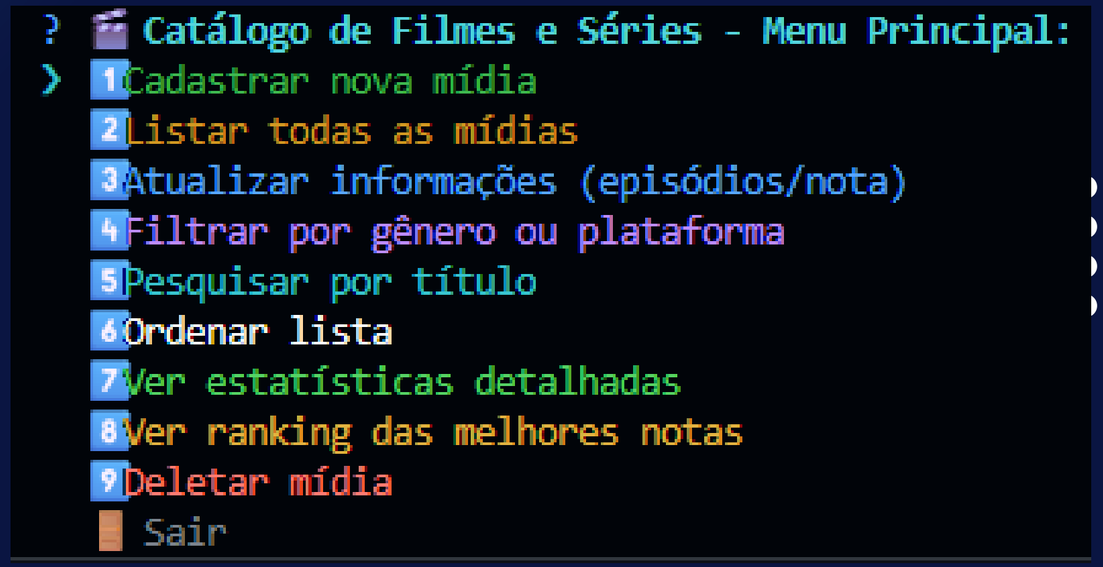
TRABALHO FINAL: PROJETO INTEGRADOR

Feito por: Kaléu

O QUE O PROJETO FAZ?

O Catálogo de Filmes e Séries, feito em Node.js. É um aplicativo interativo que roda direto no terminal, onde é possível cadastrar, listar e gerenciar filmes e séries de forma prática e visual.



ESTRUTURA E OBJETIVO

{} dados.json

O objetivo foi criar um sistema simples, mas funcional, que não precisa de banco de dados. Ele usa um arquivo chamado dados.json para salvar tudo automaticamente. A ideia é poder registrar o que eu assisto e acompanhar as notas e informações de cada mídia.

TECNOLOGIAS UTILIZADAS

Eu usei o Node.js como base do projeto, o pacote @inquirer/prompts para criar os menus interativos, o chalk para deixar o terminal colorido, e o fs/promises para ler e salvar os dados no JSON.

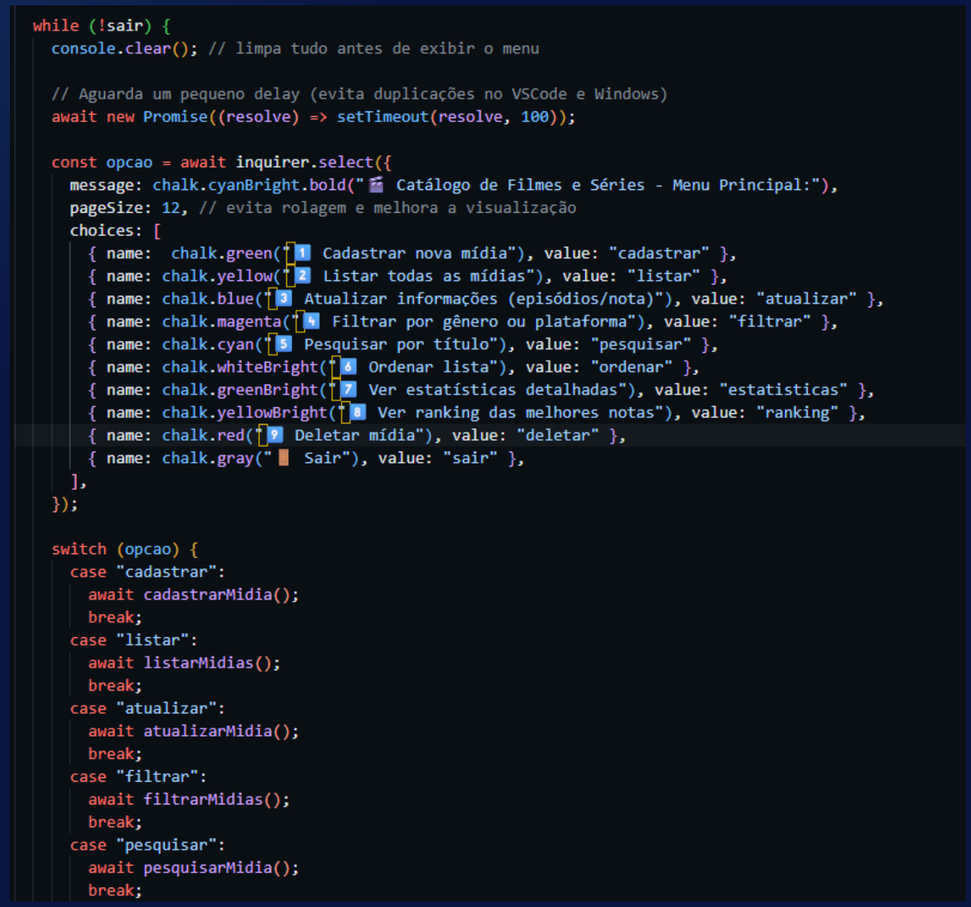
Foi interessante aprender como tudo se conecta de forma assíncrona.

```
import fs from "fs/promises";
import * as inquirer from "@inquirer/prompts";
import chalk from "chalk";


const DB_FILE = "dados.json";
```

DIFICULDADES

- Async/Await: entender como esperar respostas do usuário e salvar dados na ordem certa.
- Inquirer: montar menus interativos sem repetir opções no terminal.
- Validação: tratar notas e números digitados com vírgula ou texto errado.
- Git: resolver problemas de “changes” criando um .gitignore.
- Diferença entre filmes e séries: ajustar a lógica para atualizar episódios ou notas.



```
while (!sair) {
  console.clear(); // limpa tudo antes de exibir o menu

  // Aguarda um pequeno delay (evita duplicações no VSCode e Windows)
  await new Promise((resolve) => setTimeout(resolve, 100));

  const opcao = await inquirer.select({
    message: chalk.cyanBright.bold("🎬 Catálogo de Filmes e Séries - Menu Principal:"),
    pageSize: 12, // evita rolagem e melhora a visualização
    choices: [
      { name:  chalk.green("1 Cadastrar nova mídia"), value: "cadastrar" },
      { name: chalk.yellow("2 Listar todas as mídias"), value: "listar" },
      { name: chalk.blue("3 Atualizar informações (episódios/nota)"), value: "atualizar" },
      { name: chalk.magenta("4 Filtrar por gênero ou plataforma"), value: "filtrar" },
      { name: chalk.cyan("5 Pesquisar por título"), value: "pesquisar" },
      { name: chalk.whiteBright("6 Ordenar lista"), value: "ordenar" },
      { name: chalk.greenBright("7 Ver estatísticas detalhadas"), value: "estatisticas" },
      { name: chalk.yellowBright("8 Ver ranking das melhores notas"), value: "ranking" },
      { name: chalk.red("9 Deletar mídia"), value: "deletar" },
      { name: chalk.gray("0 Sair"), value: "sair" },
    ],
  });

  switch (opcao) {
    case "cadastrar":
      await cadastrarMidia();
      break;
    case "listar":
      await listarMidias();
      break;
    case "atualizar":
      await atualizarMidia();
      break;
    case "filtrar":
      await filtrarMidias();
      break;
    case "pesquisar":
      await pesquisarMidia();
      break;
```

OBRIGADO PELA SUA ATENÇÃO!