

# Trabalho 2

Vamos recriar as funcionalidades do primeiro trabalho (gerenciar estoque da biblioteca), porém dessa vez iremos adicionar interatividade através do terminal utilizando `prompt-sync`, e ao invés de salvarmos os dados na memória, iremos utilizar a biblioteca `fs` nativa do NodeJS para alterar um arquivo JSON onde iremos armazenar nossos livros.

Teremos duas "entidades" em nossa aplicação, para cada uma iremos desenvolver funcionalidades básicas de Criação, Leitura, Edição e Exclusão (CRUD). Essas entidades serão os livros e as categorias.

A estrutura do livro que iremos utilizar dessa vez contará com os seguintes campos:

```
{
  "id": 1,
  "titulo": "",
  "autor": "",
  "ano": 2024,
  "categorias": []
}
```

A estrutura de uma categoria é a seguinte:

```
{
  "id": 1,
  "nome": ""
}
```

## Funcionalidades

A aplicação deverá possuir um menu interativo, onde o usuário irá escolher uma opção através do número (similar a menus de WhatsApp). As opções do menu serão as seguintes:

- 1 - Criar Livro
- 2 - Editar Livro
- 3 - Buscar Livro por Nome
- 4 - Buscar Livro por Categoria
- 5 - Excluir Livro
- 6 - Criar Categoria
- 7 - Editar Categoria

8 - Visualizar Categorias

9 - Excluir Categoria

## **Cr terios de Avalia  o**

Assim como no trabalho anterior, irei considerar a l gica utilizada, nomea  es de fun   es e vari  veis, e a indenta   o do c digo.