

# Simulador de Biblioteca – Projeto em C#

## Console com JSON

---

### Objetivo

Desenvolver um sistema de biblioteca funcional em C# (aplicação console), utilizando arquivos `.json` para persistência de dados, capaz de:

- Cadastrar e listar livros e usuários
- Controlar empréstimos e devoluções de livros
- Gerar relatórios simples no console

Este projeto é voltado para aprendizado de:

- Estruturação de código em camadas
  - Manipulação de arquivos JSON com `System.Text.Json`
  - Prática com entrada e saída no console
- 

### Estrutura de Pastas Sugerida

```
BibliotecaConsole/  
├─ Program.cs  
├─ Models/  
│   ├── Livro.cs  
│   ├── Usuario.cs  
│   └─ Emprestimo.cs  
├─ Services/  
│   ├── LivroService.cs  
│   ├── UsuarioService.cs  
│   └─ EmprestimoService.cs  
└─ Data/  
    ├── livros.json  
    ├── usuarios.json  
    └─ emprestimos.json
```

---

# Requisitos Técnicos

- [.NET SDK 6.0+](#)
  - Usar a biblioteca `System.Text.Json` (já incluída no .NET 6+)
- 

## Funcionalidades Recomendadas

### Livros

- ☒ Cadastrar livro
- ☒ Listar todos os livros
- ☒ Listar livros disponíveis

### Usuários

- ☒ Cadastrar usuário
- ☒ Listar usuários

### Empréstimos

- ☒ Realizar empréstimo
  - ☒ Realizar devolução
  - ☒ Listar empréstimos ativos
  - ☒ Ver histórico de empréstimos por usuário
- 

## Divisão em Camadas

Camada	Responsabilidade
<code>Models</code>	Representações de dados (POCOs)
<code>Services</code>	Manipulação dos arquivos JSON + regras de negócio
<code>Program.cs</code>	Entrada do usuário, menus e chamadas aos serviços

---

## Menu Principal - Exemplo

==== Sistema de Biblioteca ====

1. Cadastrar usuário
  2. Cadastrar livro
  3. Listar livros disponíveis
  4. Emprestar livro
  5. Devolver livro
  6. Listar empréstimos ativos
  7. Histórico de empréstimos por usuário
  0. Sair
- Escolha uma opção: \_
- 

## Exemplo de Uso

### Empréstimo:

ID do usuário: 2

ID do livro: 5

Livro "Dom Casmurro" emprestado com sucesso para o usuário Maria.

Data do empréstimo: 01/07/2025

### Devolução:

ID do empréstimo: 3

Livro "Dom Casmurro" devolvido com sucesso em 05/07/2025.

---

## Estrutura dos Arquivos JSON (Exemplos)

### livros.json

```
[
  {
    "Id": 1,
    "Titulo": "Dom Casmurro",
    "Autor": "Machado de Assis",
    "Disponivel": true
  }
]
```

```
}  
]
```

## usuarios.json

```
[  
  {  
    "Id": 1,  
    "Nome": "Maria"  
  }  
]
```

## emprestimos.json

```
[  
  {  
    "Id": 1,  
    "UsuarioId": 1,  
    "LivroId": 1,  
    "DataEmprestimo": "2025-07-01",  
    "DataDevolucao": null  
  }  
]
```

---

## Dicas e Boas Práticas

- Use `File.Exists()` para checar se o arquivo existe antes de ler
- Use `File.ReadAllText()` e `File.WriteAllText()` com `JsonSerializer`
- Ao salvar, sempre substitua o conteúdo inteiro do JSON
- Crie métodos auxiliares para carregar e salvar listas de objetos
- Commit no Git com mensagens claras a cada avanço importante

---

## Critérios de Conclusão

- Estrutura do projeto organizada conforme sugerido
- Todas funcionalidades básicas implementadas

- Manipulação funcional dos arquivos JSON
- Interface de console com menus claros
- Código testado e funcionando