

Aula 2

Created	@August 7, 2025 6:10 PM
Data	@August 7, 2025
Status	Não iniciado
Tipo	Aula

Desafio Final do Módulo – Sistema de Gestão de Biblioteca

Objetivo

Criar um sistema completo de **gestão de biblioteca** utilizando **Programação Orientada a Objetos em JavaScript**, aplicando todos os conceitos vistos ao longo do módulo:

- Abstração de entidades reais
- Classes e objetos
- Construtor, atributos e métodos
- Encapsulamento, getters e setters
- Herança e polimorfismo
- Prototype
- Boas práticas de programação

Contexto

A escola precisa de um sistema para gerenciar sua biblioteca. Ele deve permitir o **cadastro, edição, exclusão e listagem** de livros, autores e usuários, além de permitir que usuários façam **empréstimos e devoluções**.

Requisitos Obrigatórios

1. Modelagem UML Simplificada

- Criar um diagrama UML com as entidades do sistema e seus relacionamentos.

2. Classes Obrigatórias

- **Livro** (atributos como título, autor, ano, gênero, disponível/não disponível)
- **Autor** (nome, nacionalidade, ano de nascimento)
- **Usuario** (nome, matrícula, histórico de empréstimos)
- **Biblioteca** (responsável por gerenciar os cadastros e operações)

3. Funcionalidades Mínimas

- Cadastro, edição e exclusão de livros, autores e usuários
- Empréstimo e devolução de livros
- Validação para não permitir empréstimo de livro indisponível
- Histórico de empréstimos de cada usuário

4. Uso de Encapsulamento

- Pelo menos 2 atributos privados em alguma classe com métodos getters e setters

5. Uso de Herança e Polimorfismo

- Criar pelo menos uma classe que herde de outra (ex.: **UsuarioAluno** e **UsuarioProfessor**)
- Métodos sobrescritos para alterar comportamento de forma polimórfica

Requisitos Extras (Desafio Avançado)

- Criar interface HTML para manipular o sistema (listagens, formulários, botões)
- Persistir os dados usando **LocalStorage**
- Filtrar e pesquisar livros pelo título ou autor
- Criar relatórios (ex.: livros mais emprestados, usuários com mais empréstimos)

Entrega

- Código fonte em um repositório no GitHub
- Arquivo `README.md` explicando:
 - Objetivo do projeto
 - Como rodar
 - Estrutura das classes
- UML em formato imagem ou PDF