

FACULDADE ANHANGUERA DE SÃO JOSÉ DOS CAMPOS

# DOUGLAS RICARDO DE OLIVEIRA

## MATRÍCULA: 2025164380

# SISTEMA DE GERENCIAMENTO DE BIBLIOTECA

Projeto Integrado - Curso de Ciência de Dados

SÃO JOSÉ DOS CAMPOS - SP  
2025

# FACULDADE ANHANGUERA DE SÃO JOSÉ DOS CAMPOS

## DOUGLAS RICARDO DE OLIVEIRA

MATRÍCULA: 2025164380

## PROJETO INTEGRADO: SISTEMA DE GERENCIAMENTO DE BIBLIOTECA

Projeto apresentado como requisito parcial para aprovação na disciplina Projeto Integrado, do curso de Ciência de Dados, sob orientação dos docentes da Faculdade Anhanguera.

SÃO JOSÉ DOS CAMPOS - SP  
2025

# INTRODUÇÃO

Com o avanço da tecnologia e a necessidade de modernização dos processos, este projeto propõe o desenvolvimento de um sistema de gerenciamento de biblioteca em Python, utilizando os princípios da programação orientada a objetos. O sistema tem como objetivo digitalizar e automatizar o controle de livros e usuários, promovendo maior organização e eficiência na gestão da biblioteca.

# DESENVOLVIMENTO

## Requisitos Atendidos

• Cadastro de livros com título, autor, ano e número de cópias  
• Cadastro de usuários com nome, ID e contato  
• Empréstimo e devolução de livros  
• Consulta por título, autor ou ano  
• Relatórios de livros e usuários

## Tecnologias Utilizadas

• Linguagem Python  
• Paradigma POO (Programação Orientada a Objetos)  
• Ambiente Google Colab para testes

## Estrutura do Sistema

### Classes

• Livro: representa os livros cadastrados.  
• Usuario: representa os leitores da biblioteca.  
• Biblioteca: responsável pela gestão de livros e usuários.

### Funcionalidades

• Menu interativo via console  
• Validação de erros (ex: livro indisponível, usuário inexistente)  
• Código modularizado e comentado

Além das implementações técnicas, este projeto também está alinhado com aspectos de gestão e planejamento, conforme abordado em literaturas como a Revista de Gestão e Projetos (UNINOVE, 2010) e a revista Gestão & Planejamento (FACS, 1999). A estrutura do sistema reflete práticas de organização e análise de processos que contribuem para a eficácia do serviço prestado à comunidade.

# CÓDIGO-FONTE DO SISTEMA (PYTHON)

class Livro:  
 def \_\_init\_\_(self, titulo, autor, ano, copias):  
 self.titulo = titulo  
 self.autor = autor  
 self.ano = ano  
 self.copias = copias  
 self.copias\_disponiveis = copias  
  
class Usuario:  
 def \_\_init\_\_(self, nome, id\_usuario, contato):  
 self.nome = nome  
 self.id\_usuario = id\_usuario  
 self.contato = contato  
 self.livros\_emprestados = []  
  
class Biblioteca:  
 def \_\_init\_\_(self):  
 self.livros = []  
 self.usuarios = []  
  
 def cadastrar\_livro(self, titulo, autor, ano, copias):  
 livro = Livro(titulo, autor, ano, copias)  
 self.livros.append(livro)  
  
 def cadastrar\_usuario(self, nome, id\_usuario, contato):  
 usuario = Usuario(nome, id\_usuario, contato)  
 self.usuarios.append(usuario)  
  
 def emprestar\_livro(self, id\_usuario, titulo):  
 usuario = next((u for u in self.usuarios if u.id\_usuario == id\_usuario), None)  
 livro = next((l for l in self.livros if l.titulo.lower() == titulo.lower()), None)  
 if usuario and livro and livro.copias\_disponiveis > 0:  
 livro.copias\_disponiveis -= 1  
 usuario.livros\_emprestados.append(livro.titulo)  
  
 def devolver\_livro(self, id\_usuario, titulo):  
 usuario = next((u for u in self.usuarios if u.id\_usuario == id\_usuario), None)  
 livro = next((l for l in self.livros if l.titulo.lower() == titulo.lower()), None)  
 if usuario and livro and titulo in usuario.livros\_emprestados:  
 livro.copias\_disponiveis += 1  
 usuario.livros\_emprestados.remove(titulo)  
  
 def consultar\_livros(self, termo):  
 return [l for l in self.livros if termo.lower() in l.titulo.lower()  
 or termo.lower() in l.autor.lower() or termo.lower() in str(l.ano)]  
  
 def gerar\_relatorios(self):  
 for l in self.livros:  
 print(f"{l.titulo} - {l.copias\_disponiveis}/{l.copias} disponíveis")  
 for u in self.usuarios:  
 print(f"{u.nome} (ID: {u.id\_usuario}) - Livros: {u.livros\_emprestados}")

# CAPTURA DE TELA DO FUNCIONAMENTO NO GOOGLE COLAB



# CONCLUSÃO

O sistema proposto atende às necessidades básicas de uma biblioteca comunitária, oferecendo funcionalidades fundamentais como cadastro, empréstimo e devolução de livros. A utilização da linguagem Python e da POO permitiu uma estrutura organizada e escalável. A experiência contribuiu para a aplicação prática dos conhecimentos adquiridos no curso, além de reforçar conceitos de cultura, sociedade e ética, conforme discutido nas obras de Metcalf (2015) e La Taille (2007), promovendo um desenvolvimento mais humano e responsável no uso da tecnologia.

# REFERÊNCIAS

BRASIL. Lei nº 9.610, de 19 de fevereiro de 1998. Regula os direitos autorais e dá outras providências. Disponível em: http://www.planalto.gov.br/ccivil\_03/leis/L9610.htm. Acesso em: 04 jun. 2025.  
  
ALURA. Cursos de Programação em Python. Disponível em: https://www.alura.com.br  
  
DOCUMENTAÇÃO OFICIAL DO PYTHON. Disponível em: https://docs.python.org/3/  
  
REVISTA DE GESTÃO E PROJETOS. Universidade Nove de Julho. 2010 - ISSN 2236-0972  
  
METCALF, Peter. Cultura e Sociedade. São Paulo: Saraiva, 2015.  
  
GESTÃO & PLANEJAMENTO. FACS Serviços Educacionais S.A. 1999 - ISSN 1516-9103  
  
LA TAILLE, Yves de. Moral e ética: dimensões intelectuais e afetivas. Porto Alegre: Artmed, 2007.  
  
KOCH, Ingedore Villaça; ELIAS, Vanda Maria. Ler e escrever: estratégias de produção textual. 2ª ed. - São Paulo: Contexto, 2010. Biblioteca Virtual 3.0