



CENTRO UNIVERSITÁRIO ANHANGUERA PITÁGORAS UNOPAR DE CAMPO GRANDE

CURSO: ENGENHARIA DE SOFTWARE

DISCIPLINA: LINGUAGEM DE PROGRAMAÇÃO

ATIVIDADE PRÁTICA – UNIDADE 2, AULA 4

BIBLIOTECAS E MÓDULOS EM PYTHON

Aluno: Edmar Radanovis.

RA: 2025223493

Polo: Itapira / SP-UN944038

Ano 2025 / 2º Semestre

ANHANGUERA EDUCACIONAL
ENGENHARIA DE SOFTWARE

DISCIPLINA: Linguagem de Programação

PROFESSOR: Anderson I. S. Abreu / Vanessa Matias Leite

ALUNO: Edmar Radanovis

RA: 2025223493

TÍTULO: Relatório da Aula Prática – Bibliotecas e Módulos em Python – Unidade 2 –
Aula 4

POLO: Itapira / SP-UN944038

Monte Sião, 08 de Setembro de 2025.

RESULTADOS DA ATIVIDADE PRÁTICA

Proposta:



Introdução a Linguagem Python - Bibliotecas e Módulos.

- Você foi contratado para desenvolver um sistema simples de gerenciamento de livros em uma biblioteca. O sistema deve permitir cadastrar novos livros, listar todos os livros disponíveis, buscar um livro pelo título, e gerar um gráfico com a quantidade de livros por gênero.
 - Definir a classe Livro:
 - Comece definindo a estrutura básica de um livro usando uma classe.
 - Cada livro terá atributos como título, autor, gênero e quantidade.
 - Criar a lista de livros:
 - Inicialize uma lista vazia para armazenar os livros que serão cadastrados.
 - Implementar funções para gerenciar os livros :
 - Função para cadastrar um novo livro.
 - Função para listar todos os livros.
 - Função para buscar um livro pelo título.
 - Utilizar a biblioteca Matplotlib para gerar um gráfico:
 - Instalação da Matplotlib.
 - Gerar o gráfico de quantidade de livros por gênero.

link do repositório no GitHub:

https://github.com/ed-radanovis/Eng_Software_L-P_U2-A4_09-2025.git

```
unit_two_lesson_four_libraries_and_modules.py > cadastrar_livro
1 # Sistema de Gerenciamento de Livros para Biblioteca
2 # Utilizando classes, funções e a Biblioteca matplotlib para gráficos
3
4 import matplotlib.pyplot as plt
5
6 class Livro:
7     def __init__(self, titulo, autor, genero, quantidade):
8         self.titulo = titulo
9         self.autor = autor
10        self.genero = genero
11        self.quantidade = quantidade
12
13    def __str__(self):
14        return f"Uma representação de Livro na biblioteca.\nAtributos: título, autor, gênero e quantidade disponível."
15
16    def cadastrar_livro():
17        lista_livros = []
18
19        for i in range(5):
20            livro = Livro("Nome do Livro", "Autor", "Gênero", 1)
21            lista_livros.append(livro)
22
23        biblioteca = []
24
25        def cadastrar_livro():
26            função para cadastrar um novo livro na biblioteca.
27            Solicita os dados do livro ao usuário e adiciona à lista.
28
29            print("CADASTRO DE NOVO LIVRO")
30
31            titulo = input("Digite o título do Livro: ").strip()
32            autor = input("Digite o autor do livro: ").strip()
33            genero = input("Digite o gênero do livro: ").strip()
34
35            # Validação da quantidade
36            while True:
37                try:
38                    quantidade = int(input("Digite a quantidade disponível: "))
39                    if quantidade < 0:
40                        print("A quantidade não pode ser negativa.")
41                    else:
42                        break
43                except ValueError:
44                    print("Por favor, digite um número válido.")
45
46            novo_livro = Livro(titulo, autor, genero, quantidade)
47            biblioteca.append(novo_livro)
48            print(f"\nLivro '{titulo}' cadastrado com sucesso!")
49
50        def listar_livros():
51            função para listar todos os livros cadastrados na biblioteca.
52
53            print("\n")
54            print("LISTA DE TODOS OS LIVROS")
55            print("-----")
56
57            if not biblioteca:
58                print("Nenhum livro cadastrado na biblioteca.")
59            else:
60                for livro in enumerate(biblioteca):
61                    print(f"{livro[0]}. {livro[1]}")
62
63        def buscar_livro_por_titulo():
64            função para buscar um livro pelo título (busca parcial case-insensitive).
65
66            print("\n")
67            print("BUSCA DE LIVRO POR TÍTULO")
68            print("-----")
69
70            if not biblioteca:
71                print("Nenhum livro cadastrado na biblioteca.")
72            else:
73                termo_busca = input("Digite o título ou parte do título para buscar: ").lower()
74
75                livros_encontrados = []
76
77                for livro in biblioteca:
78                    if termo_busca in livro.titulo.lower():
79                        livros_encontrados.append(livro)
80
81                if livros_encontrados:
82                    print(f"\n{len(livros_encontrados)} resultados encontrados com '{termo_busca}'")
83                    for i, livro in enumerate(livros_encontrados, 1):
84                        print(f"{i}. {livro}")
85                else:
86                    print(f"\nNenhum livro encontrado com '{termo_busca}' no título.")
87
88        def gerar_grafico_genres():
89            # Geração de gráfico de barras com a quantidade de livros por gênero.
90            # Utiliza a Biblioteca matplotlib para visualização.
91
92            print("\n")
93            print("GRÁFICO DE LIVROS POR GÊNERO")
94            print("-----")
95
96            if not biblioteca:
97                print("Nenhum livro cadastrado para gerar gráfico.")
98            else:
99                # Contagem de livros por gênero
100               contagem_genres = {}
101               for livro in biblioteca:
102                   if livro.genero in contagem_genres:
103                       contagem_genres[livro.genero] += livro.quantidade
104                   else:
105                       contagem_genres[livro.genero] = livro.quantidade
106
107               # Ordenando os gêneros
108               generos = list(contagem_genres.keys())
109               quantidades = list(contagem_genres.values())
110
111               plt.figure(figsize=(10, 6))
112               plt.bar(generos, quantidades, color=[ "#E74C3C", "#E655AA", "#E181F2FF", "#D9E9D9", "#85C999", "#3E8EAA" ])
113               plt.xlabel("Gêneros", fontweight='bold')
114               plt.ylabel("Quantidade de Livros por Gênero", fontweight='bold')
115               plt.title("Gráfico de Quantidade de Livros por Gênero", fontweight='bold')
116               plt.xticks(rotation=45, ha='right')
117
118               # Adicionando valores nas barras
119               for valor in quantidades:
120                   plt.text(i, valor + 0.1, str(valor), ha='center', va='bottom', fontweight='bold')
121
122               plt.tight_layout()
123               plt.show()
124
125               print("Gráfico gerado com sucesso!")
126
127        def menu_principal():
128            função que exibe o menu principal e gerencia as opções do sistema.
129
130            while True:
131                print("\n")
132                print("SISTEMA DE GERENCIAMENTO DE BIBLIOTECA")
133                print("-----")
134                print("1. Cadastrar novo livro")
135                print("2. Listar todos os livros")
136                print("3. Buscar livro por título")
137                print("4. Gerar gráfico de gêneros")
138                print("5. Sair do sistema")
139
140                print("-----")
141
142                opcao = input("Digite a opção desejada: ").strip()
143
144                if opcao == "1":
145                    cadastrar_livro()
146                elif opcao == "2":
147                    listar_livros()
148                elif opcao == "3":
149                    buscar_livro_por_titulo()
150                elif opcao == "4":
151                    gerar_grafico_genres()
152                elif opcao == "5":
153                    print("Obrigado por usar o sistema! Até logo! ☺")
154                    break
155                else:
156                    print("Opção inválida! Digite um número de 1 a 5.")
157
158        # Execução principal do programa
159        nome_principal()
160
161        # Exemplo de uso: Adiciona livros de exemplo para teste.
162        biblioteca.append(Livro("Dom Casmurro", "Machado de Assis", "Romance", 5))
163        biblioteca.append(Livro("O Cortiço", "Aluísio Azevedo", "Romance", 3))
164        biblioteca.append(Livro("O Alienista", "Machado de Assis", "Ficção", 2))
165        biblioteca.append(Livro("A Invenção de Hugo Cabret", "Santosh Ray", "Infantil", 6))
166        biblioteca.append(Livro("1984", "George Orwell", "Ficção Científica", 3))
167        biblioteca.append(Livro("Harry Potter e a Pedra Filosofal", "J. K. Rowling", "Infantil", 4))
168        biblioteca.append(Livro("Clean Code", "Robert Cecil Martin", "Tecnologia / Programação", 8))
169
170        print("\n\nBem-vindo ao Sistema de Gerenciamento de Biblioteca! ☺")
171        print("\n\nAlguns livros de exemplo foram adicionados para teste.")
172
173        nome_principal()
```

=> Figura 1: Print da tela com o código.

```

File Edit Selection View Go ... < > U2_A4_BIBLIOTECAS_E_MODULOS_EM_PYTHON 08 ... x
unit_two_lesson_four_libraries_and_modules.py > listar_livros
25 def cadastrar_livro():
    PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS Python + ... | x
    & C:/Users/edrad/AppData/Local/Programs/Python/Python311/python.exe "c:/Users/edrad/OneDrive/Area de Trabalho/ed/Anhanguera_Eng_Software/2º semestre/5º-Linguagem_de_Programacao/sttvl/dados/U2_A4_BIBLIOTECAS_E_MODULOS_EM_PYTHON/unit_two_lesson_four_libraries_and_modules.py"
    Bem-vindo ao Sistema de Gerenciamento de Biblioteca
    Alguns livros de exemplo foram adicionados para teste.

    SISTEMA DE GERENCIAMENTO DE BIBLIOTECA
    1. Cadastrar novo livro
    2. Listar todos os livros
    3. Buscar livro por título
    4. Gerar gráfico por gênero
    5. Sair do sistema
    Digite a opção desejada: 1
    CADASTRO DE NOVO LIVRO
    Digite o título do livro: Introdução à Computação Usando Python
    Digite o autor do livro: André Luiz P. P. Markovic
    Digite o gênero do livro: Tecnologia / Programação
    Digite a quantidade disponível: 10
    Livro 'Introdução à Computação Usando Python' cadastrado com sucesso!
    SISTEMA DE GERENCIAMENTO DE BIBLIOTECA
    1. Cadastrar novo livro
    2. Listar todos os livros
    3. Buscar livro por título
    4. Gerar gráfico por gênero
    5. Sair do sistema
    Digite a opção desejada: 2
    LISTA DE TODOS OS LIVROS
    1. 'Dom Casmurro' por Machado de Assis | Gênero: Romance | Disponível: 5
    2. 'Cidade dos Livros' por J. R. R. Tolkien | Gênero: Romance | Disponível: 1
    3. 'Iracema' por José de Alencar | Gênero: Romance | Disponível: 1
    4. 'Aventuras de Dom Quixote' por M. de Cervantes | Gênero: Romance | Disponível: 2
    5. 'O Pequeno Príncipe' por Antoine de Saint-Exupéry | Gênero: Infantil | Disponível: 6
    6. '1984' por George Orwell | Gênero: Ficção Científica | Disponível: 8
    7. 'O Hobbit' por J. R. R. Tolkien | Gênero: Romance | Disponível: 4
    8. 'Clean Code' por Robert Cecil Martin | Gênero: Tecnologia / Programação | Disponível: 8
    9. 'Introdução à Computação Usando Python' por Ljubomir Perkovic | Gênero: Tecnologia / Programação | Disponível: 10
    SISTEMA DE GERENCIAMENTO DE BIBLIOTECA
    1. Cadastrar novo livro
    2. Listar todos os livros
    3. Buscar livro por título
    4. Gerar gráfico por gênero
    5. Sair do sistema
    Digite a opção desejada: 3
    BUSCA DE LIVRO POR TÍTULO
    Digite o título ou parte do título para buscar: Clean Architecture
    × Nenhum livro encontrado com 'clean architecture' no título.

    SISTEMA DE GERENCIAMENTO DE BIBLIOTECA
    1. Cadastrar novo livro
    2. Listar todos os livros
    3. Buscar livro por título
    4. Gerar gráfico por gênero
    5. Sair do sistema
    Digite a opção desejada: 3
    BUSCA DE LIVRO POR TÍTULO
    Digite o título ou parte do título para buscar: clean co
    × Livros encontrados com 'clean co'
    'Clean Code' por Robert Cecil Martin | Gênero: Tecnologia / Programação | Disponível: 8
    SISTEMA DE GERENCIAMENTO DE BIBLIOTECA
    1. Cadastrar novo livro
    2. Listar todos os livros
    3. Buscar livro por título
    4. Gerar gráfico por gênero
    5. Sair do sistema
    Digite a opção desejada: 4
    × Opção inválida! Digite um número de 1 a 5.
    SISTEMA DE GERENCIAMENTO DE BIBLIOTECA
    1. Cadastrar novo livro
    2. Listar todos os livros
    3. Buscar livro por título
    4. Gerar gráfico por gênero
    5. Sair do sistema
    Digite a opção desejada: 4
    GRÁFICO DE LIVROS POR GÊNERO
    Figure1
    Quantidade de Livros por Gênero
    18
    12
    6
    3
    4
    2
    0.0 2.5 5.0 7.5 10.0 12.5 15.0 17.5
    Romance Ficção Infantil Ficção Científica Fábula Tecnologia / Programação
    Gêneros
    In 57, Col 8 Spaces:2 UFT-8 CRLF ( Python 3.13.7 Go Live Quokka Prettier

```

=> Figura 2: Print da tela do terminal executando as funções propostas.

Lógica Utilizada no Sistema de Gerenciamento de Biblioteca

O sistema foi desenvolvido seguindo uma abordagem modular e orientada a objetos, implementando os seguintes componentes principais:

Estrutura do Sistema:

- **Classe Livro:** Define a estrutura básica dos livros com atributos como título, autor, gênero e quantidade disponível, utilizando o método `__str__` para representação textual.
- **Lista Centralizada:** Utiliza uma lista global biblioteca para armazenamento de todos os objetos Livro cadastrados.
- **Funções Modulares:**
 - `cadastrar_livro()`: Implementa validação de dados e tratamento de exceções para entrada numérica.
 - `listar_livros()`: Exibe todos os livros de forma organizada com enumeração.
 - `buscar_livro_por_titulo()`: Realiza busca case-insensitive com correspondência parcial.
 - `gerar_grafico_generos()`: Processa dados e gera visualização com `matplotlib`.

Lógica do Gráfico:

- **Agrupamento por Gênero:** Percorre todos os livros somando as quantidades por categoria.
- **Transformação de Dados:** Converte dicionário de contagens para listas separadas (gêneros e quantidades).
- **Visualização:** Utiliza mapa de cores personalizado com 6 tons distintos para melhor distinção visual.

Personalização Visual:

- Cores hexadecimais específicas.
- Formatação de rótulos e valores nas barras.
- Ajuste de layout para melhor legibilidade.

 **Fluxo do Programa:** Menu interativo → Validação de opções →

Processamento específico → Retorno ao menu principal, garantindo usabilidade e tratamento de erros.

 **Tratamento de Erros:**

- Validação de entrada numérica na quantidade de livros.
- Verificação de lista vazia antes de operações críticas.
- Prevenção de erros de índice e tipo.

O sistema demonstra o uso eficiente de classes, funções, estruturas de dados e bibliotecas externas para criar uma aplicação completa e funcional.