

# **Sistema de Biblioteca**

Ana Karolina Rosa Lourenço da Silva - 323211803  
Versão 1.0

# Sumário

Table of contents



# Índice Hierárquico

## Hierarquia de Classes

Esta lista de hierarquias está parcialmente ordenada (ordem alfabética):

Biblioteca .....	5
Item .....	7
Livro.....	10
Revista.....	13
Principal .....	12

# Índice dos Componentes

## Lista de Classes

Aqui estão as classes, estruturas, uniões e interfaces e suas respectivas descrições:

<b>Biblioteca</b>	.....	55
<b>Item</b>	.....	77
<b>Livro</b>	.....	1010
<b>Principal</b>	.....	1212
<b>Revista</b>	.....	1313

# Índice dos Arquivos

## Lista de Arquivos

Esta é a lista de todos os arquivos e suas respectivas descrições:

<b>C:/wamp64/www/A3/A3/src/main/java/Biblioteca.java</b>	5
<b>C:/wamp64/www/A3/A3/src/main/java/Item.java</b>	7
<b>C:/wamp64/www/A3/A3/src/main/java/Livro.java</b>	10
<b>C:/wamp64/www/A3/A3/src/main/java/Principal.java</b>	12
<b>C:/wamp64/www/A3/A3/src/main/java/Revista.java</b>	13

# Classes

## Referência da Classe Biblioteca

### Métodos Públicos

- **Biblioteca ()**
  - void **inserirItem** (Item item)
  - void **alterarItem** (int index, Item item)
  - void **removerItem** (int index)
  - void **listarItens** ()
- 

### Construtores e Destrutores

#### Biblioteca.Biblioteca ()

```
6          {
7      itens = new ArrayList<>();
8  }
```

---

### Documentação das funções

#### void Biblioteca.alterarItem (int index, Item item)

Altera item na biblioteca.

##### Parâmetros

<i>index</i>	Índice do item alterado na lista.
<i>item</i>	Novo item com as alterações.

```
27          {
28      if (index >= 0 && index < itens.size()) {
29          itens.set(index, item);
30      } else {
31          System.out.println("Índice inválido.");
32      }
33  }
```

#### void Biblioteca.inserirItem (Item item)

Insere um novo item na biblioteca.

##### Parâmetros

<i>item</i>	Novo item inserido.
-------------	---------------------

```
16          {
17      itens.add(item);
18  }
```

#### void Biblioteca.listarItens ()

```
49          {
50      for (Item item : itens) {
51          item.exibirDetalhes();
52      }
53  }
```

### **void Biblioteca.removerItem (int index)**

Remove item da biblioteca.

#### **Parâmetros**

<i>index</i>	Índice do item removido na lista.
--------------	-----------------------------------

```
41         {
42         if (index >= 0 && index < itens.size()) {
43             itens.remove(index);
44         } else {
45             System.out.println("Índice inválido.");
46         }
47     }
```

---

**A documentação para essa classe foi gerada a partir do seguinte arquivo:**

- C:/wamp64/www/A3/A3/src/main/java/**Biblioteca.java**



## Referência da Classe Item

Diagrama de hierarquia da classe Item:



### Métodos Públicos

- **Item** (String **titulo**, String **autor**, int **ano**)
- abstract void **exibirDetalhes** ()
- String **getTitulo** ()
- void **setTitulo** (String **titulo**)
- String **getAutor** ()
- void **setAutor** (String **autor**)
- int **getAno** ()
- void **setAno** (int **ano**)

### Atributos Protegidos

- String **titulo**
- String **autor**
- int **ano**

---

## Construtores e Destrutores

### Item.Item (String titulo, String autor, int ano)

Construtor que inicializa os atributos básicos do **Item**.

#### Parâmetros

<i>titulo</i>	Título do item.
<i>autor</i>	Autor do item.
<i>ano</i>	Ano de publicação do item.

```
14                                     {
15     this.titulo = titulo;
16     this.autor = autor;
17     this.ano = ano;
18 }
```

---

## Documentação das funções

### abstract void Item.exibirDetalhes () [abstract]

Reimplementado por **Livro** (p.10) e **Revista** (p.13).

### int Item.getAno ()

Obtém o ano de publicação do item.

#### Retorna

O ano de publicação do item.

```
64                                     {
65     return ano;
66 }
```

### String Item.getAutor ()

Obtém o autor do item.

#### Retorna

O autor do item.

```
46         {  
47             return autor;  
48         }
```

### String Item.getTitulo ()

Obtém o título do item.

#### Retorna

O título do item.

```
28         {  
29             return titulo;  
30         }
```

### void Item.setAno (int ano)

Define o ano de publicação do item.

#### Parâmetros

<i>ano</i>	Novo ano de publicação para o item.
------------	-------------------------------------

```
73         {  
74             this.ano = ano;  
75         }
```

### void Item.setAutor (String autor)

Define o autor do item.

#### Parâmetros

<i>autor</i>	Novo autor para o item.
--------------	-------------------------

```
55         {  
56             this.autor = autor;  
57         }
```

### void Item.setTitulo (String titulo)

Define o título do item.

#### Parâmetros

<i>titulo</i>	Novo título para o item.
---------------	--------------------------

```
37         {  
38             this.titulo = titulo;  
39         }
```

## Atributos

**int** Item.ano [protected]

**String** Item.autor [protected]

**String** Item.titulo [protected]

---

**A documentação para essa classe foi gerada a partir do seguinte arquivo:**

- C:/wamp64/www/A3/A3/src/main/java/**Item.java**

## Referência da Classe Livro

Diagrama de hierarquia da classe Livro:



### Métodos Públicos

- **Livro** (String **titulo**, String **autor**, int **ano**, String genero)
- String **getGenero** ()
- void **setGenero** (String genero)
- void **exibirDetalhes** ()

### Métodos Públicos herdados de Item

- **Item** (String **titulo**, String **autor**, int **ano**)
- String **getTitulo** ()
- void **setTitulo** (String **titulo**)
- String **getAutor** ()
- void **setAutor** (String **autor**)
- int **getAno** ()
- void **setAno** (int **ano**)

### Outros membros herdados

#### Atributos Protegidos herdados de Item

- String **titulo**
- String **autor**
- int **ano**

---

## Construtores e Destrutores

### Livro.Livro (String titulo, String autor, int ano, String genero)

Construtor que inicializa os atributos de um **Livro**.

#### Parâmetros

<i>titulo</i>	Título do livro.
<i>autor</i>	Autor do livro.
<i>ano</i>	Ano de publicação do livro.
<i>genero</i>	Gênero do livro.

```
13                                     {
14     super(titulo, autor, ano);
15     this.genero = genero;
16 }
```

---

## Documentação das funções

### void Livro.exibirDetalhes ()

Reimplementa **Item** (p.7).

```
39                                     {
```

```

40         System.out.println("Livro: " + titulo + ", Autor: " + autor + ", Ano: " +
ano + ", Gênero: " + genero);
41     }

```

### String Livro.getGenero ()

Obtém o gênero do livro.

#### Retorna

O gênero do livro.

```

24         {
25             return genero;
26     }

```

### void Livro.setGenero (String genero)

Define o gênero do livro.

#### Parâmetros

<i>genero</i>	Novo gênero para o livro.
---------------	---------------------------

```

32         {
33             this.genero = genero;
34     }

```

---

**A documentação para essa classe foi gerada a partir do seguinte arquivo:**

- C:/wamp64/www/A3/A3/src/main/java/**Livro.java**

## Referência da Classe Principal

### Membros públicos estáticos

- static void **main** (String[] args)

---

### Descrição detalhada

#### Autor

anaka

---

### Documentação das funções

#### static void Principal.main (String[] args) [static]

Inicialização do sistema da biblioteca.

#### Parâmetros

<i>args</i>	Argumentos de linha de comando.
-------------	---------------------------------

```
13                                     {
14     Biblioteca biblioteca = new Biblioteca();
15     Scanner scanner = new Scanner(System.in);
16     int opcao;
17
18     do {
19         System.out.println("\nMenu de Opções:");
20         System.out.println("1. Inserir");
21         System.out.println("2. Alterar");
22         System.out.println("3. Remover");
23         System.out.println("4. Listar");
24         System.out.println("5. Sair");
25         System.out.print("Escolha uma opção: ");
26         opcao = scanner.nextInt();
27         scanner.nextLine();
28
29         switch (opcao) {
30             case 1:
31                 inserirItem(biblioteca, scanner);
32                 break;
33             case 2:
34                 alterarItem(biblioteca, scanner);
35                 break;
36             case 3:
37                 removerItem(biblioteca, scanner);
38                 break;
39             case 4:
40                 biblioteca.listarItens();
41                 break;
42             case 5:
43                 System.out.println("Saindo do sistema...");
44                 break;
45             default:
46                 System.out.println("Opção inválida.");
47         }
48     } while (opcao != 5);
49 }
```

---

A documentação para essa classe foi gerada a partir do seguinte arquivo:

- C:/wamp64/www/A3/A3/src/main/java/**Principal.java**

## Referência da Classe Revista

Diagrama de hierarquia da classe Revista:



### Métodos Públicos

- **Revista** (String **titulo**, String **autor**, int **ano**, int **edicao**)
- int **getEdicao** ()
- void **setEdicao** (int **edicao**)
- void **exibirDetalhes** ()

### Métodos Públicos herdados de Item

- **Item** (String **titulo**, String **autor**, int **ano**)
- String **getTitulo** ()
- void **setTitulo** (String **titulo**)
- String **getAutor** ()
- void **setAutor** (String **autor**)
- int **getAno** ()
- void **setAno** (int **ano**)

### Outros membros herdados

#### Atributos Protegidos herdados de Item

- String **titulo**
- String **autor**
- int **ano**

---

## Construtores e Destrutores

### Revista.Revista (String titulo, String autor, int ano, int edicao)

Construtor que inicializa os atributos de uma **Revista**.

#### Parâmetros

<i>titulo</i>	Título da revista.
<i>autor</i>	Autor da revista.
<i>ano</i>	Ano de publicação da revista.
<i>edicao</i>	Edição da revista.

```
11                                     {
12     super(titulo, autor, ano);
13     this.edicao = edicao;
14 }
```

---

## Documentação das funções

### void Revista.exibirDetalhes ()

Reimplementa **Item** (p.7).

```
38                                     {
```

```

39         System.out.println("Revista: " + titulo + ", Autor: " + autor + ", Ano: "
+ ano + ", Edição: " + edicao);
40     }

```

### **int Revista.getEdicao ()**

Obtém a edição da revista.

#### **Retorna**

A edição da revista.

```

20         {
21             return edicao;
22     }

```

### **void Revista.setEdicao (int edicao)**

Define a edição da revista.

#### **Parâmetros**

<i>edicao</i>	Nova edição para a revista.
---------------	-----------------------------

```

28         {
29             this.edicao = edicao;
30     }

```

---

**A documentação para essa classe foi gerada a partir do seguinte arquivo:**

- C:/wamp64/www/A3/A3/src/main/java/**Revista.java**



# Arquivos

## Referência do Arquivo

**C:/wamp64/www/A3/A3/src/main/java/Biblioteca.java**

## Componentes

`class Biblioteca`

## **Referência do Arquivo**

**C:/wamp64/www/A3/A3/src/main/java/Item.java**

## **Componentes**

`class Item`

## **Referência do Arquivo**

**C:/wamp64/www/A3/A3/src/main/java/Livro.java**

## **Componentes**

`class Livro`

## **Referência do Arquivo**

**C:/wamp64/www/A3/A3/src/main/java/Principal.java**

## **Componentes**

`class Principal`

## **Referência do Arquivo**

**C:/wamp64/www/A3/A3/src/main/java/Revista.java**

## **Componentes**

class **Revista**

