

Classe Livro

2.44 - Abaixo está a estrutura básica para uma classe 'Book', que pode estar localizada no projeto de exercício do livro. A estrutura básica já define dois campos e um construtor para inicializar os campos. Nesse exercício e nos próximos, você adicionará mais recursos à estrutura básica dessa classe.

Adicione dois métodos de acesso à classe - 'getAuthor' e 'GetTitle' - que retornam os campos 'author' e 'title' e como seus respectivos resultados. Teste sua classe criando algumas instâncias e chamando os métodos.

```
/**
 * A class that maintains information on a book.
 * This might form part of a larger application such
 * as a library system, for instance.
 *
 * @author (Insert your name here.)
 * @version (Insert today's date here.)
 */
class Book
{
    // The fields.
    private String author;
    private String title;

    /**
     * Set the author and title fields when this object
     * is constructed.
     */
    public Book(String bookAuthor, String bookTitle)
    {
        author = bookAuthor;
        title = bookTitle;
    }

    // Add the methods here ...
}
```

2.45 - Adicione dois métodos, 'printAuthor' e 'printTitle', à estrutura básica da classe 'Book'. Devem ser impressos os campos de autor e de título, respectivamente, na janela de terminal.

2.46 - Adicione mais um campo, 'pages', à classe 'Book' para armazenar o número de páginas. Esse deve ser do tipo 'int' e seu valor inicial deve ser passado para o único construtor, junto com as strings 'author' e 'title'. Inclua um método de acesso getPages apropriado para esse campo.

2.47 - Adicione um método, 'printDetails', à classe 'Book'. Isso deve imprimir detalhes do autor, título e páginas na janela de terminal. Você escolhe como os detalhes são formatados. Por exemplo, todos os três itens poderiam ser impressos em uma única linha ou cada um poderia ser impresso em uma linha separada. Você também pode escolher incluir algum texto explicativo para ajudar um usuário a determinar qual é o autor e qual é o título, por exemplo.

```
'Title: Robinson Crusoe, Author: Daniel Defoe, Pages: 232'
```

2.48 - Adicione mais um campo, 'refNumber', para a classe 'Book'. Esse campo pode armazenar um número de referência para uma biblioteca, por exemplo. Ele deve ser de tipo 'String' e ser iniciado com a string de comprimento zero ("") no construtor, pois seu valor inicial não é passado em um parâmetro para o construtor. Em vez disso, defina um método modificador para ele com a seguinte assinatura:

```
public void setRefNumber(String ref)
```

O corpo desse método deve atribuir o valor do parâmetro ao campo 'refNumber'. Adicione um método de acesso 'getRefNumber' correspondente para ajudá-lo a verificar se o método modificador funciona corretamente.

2.49 - Modifique seu método 'printDetails' para incluir impressão do número de referência. Entretanto, o método deve imprimir o número de referência somente se tiver sido configurado - isso é, se a string 'refNumber' tiver um comprimento diferente de zero. Se ela não tiver sido configurada, então imprima a string 'ZZZ' no lugar. Dica: Utilize uma instrução condicional cujo teste chama o método 'length' na string 'refNumber'.

2.50 - Modifique seu método modificador 'setRefNumber' de modo que configure o campo 'refNumber' somente se o parâmetro for uma string de pelo menos três caracteres. Se for inferior a três, imprima uma mensagem de erro e deixe o campo inalterado.

2.51 - Adicione um campo inteiro, 'borrowed', à classe 'Book'. Isso mantém uma contagem do número de vezes que um livro foi emprestado. Adicione um método modificador, 'borrow', à classe. Esse método deve incrementar o campo em 1 toda vez que ele for chamado. Inclua um método de acesso 'getBorrowed', que retorna o valor desse novo campo com seu resultado. Modifique 'printDetails' de modo que inclua o valor desse campo com um trecho de um texto explicativo.

--- Código fonte da classe Livro ---

```
**
* Classe Livro simula o objeto Livro
* Autor - .Zerol
* Versão - 1.0 Alpha_Beta (14/04/2006)
*/

public class Livro
{
    //Campos
    private String cAutor;
    private String cTitulo;
    private long cPaginas;
    private String cNReferencia;
    private long cEmprestimo;

    public Livro(String escritor, String livro, long
npaginas)
    {
        cAutor = escritor;
        cTitulo = livro;
        cPaginas = npaginas;
        cNReferencia = "";
        cEmprestimo = 0;
    }

    public String mostraAutor()
    {
        return cAutor;
    }

    public String mostraTitulo()
    {
        return cTitulo;
    }

    public String mostraNReferncia()
    {
        return cNReferencia;
    }

    public long mostraEmprestimo()
    {
        return cEmprestimo;
    }
}
```

```
public void inserirNumeroDeReferencia(String
vNReferencia)
{
    if (vNReferencia.length() >=3)
    {
        cNReferencia = vNReferencia;
    }
    else
    {
        System.out.println("Você deve inserir um
Número de Referência de no mínimo 3
caracteres!");
    }
}

public void inserirEmprestimo()
{
    cEmprestimo += 1;
}

public long mostraPaginas()
{
    return cPaginas;
}

public void imprimirAutor()
{
    System.out.println("O autor é: " + cAutor);
}

public void imprimirTitulo()
{
    System.out.println("O título é: " + cTitulo);
}
```

```

public void imprimirDetalhes()
{
    String vNReferencia = "";

    if (cNReferencia.length() == 0)
    {
        vNReferencia = "ZZZ";
    }
    else
    {
        vNReferencia = cNReferencia;
    }

    System.out.println("###Biblioteca RPF###");
    System.out.println("Título: " + cTitulo);
    System.out.println("Autor: " + cAutor);
    System.out.println("Páginas: " + cPaginas);
    System.out.println("Número de Referencia: " +
vNReferencia);
    System.out.println("...---...");
    System.out.println("Emprestado " + cEmprestimo + "
vezes");
    System.out.println("#####");
}
}

```