

Como usar a documentação da API Java 2

Helder da Rocha
www.argonavis.com.br

- *Aprender a usar a documentação é **essencial** para quem deseja desenvolver aplicações em Java*
- *A documentação da linguagem, ferramentas e API é um download separado do SDK*
- *Para instalar a documentação Java, abra o arquivo ZIP na raiz da sua instalação Java*
 - *A documentação será instalada abaixo do subdiretório docs/ da instalação Java (\$JAVA_HOME)*
 - `%JAVA_HOME%\docs\` (ex: `c:\jdk1.4.0\docs/`)
 - `$JAVA_HOME/docs/` (ex: `/usr/java/j2sdk1.4.0/docs/`)
 - *A documentação da API Java está em*
 - `$JAVA_HOME/docs/api/index.html`
 - *Manuais, tutoriais sobre recursos da linguagem em*
 - `$JAVA_HOME/docs/index.html`

Documentação

Descrição da classe
(escolhida na janela B)

- hierarquia
- documentação detalhada, métodos, variáveis, etc.

Se um método não for encontrado na classe mostrada, procure nas superclasses (use as referências cruzadas)

Lista de pacotes

Lista de classes e interfaces do pacote escolhido na janela (A)

The screenshot shows the Java 2 Platform SE v1.3 documentation page for the `Class Frame`. The page is displayed in Microsoft Internet Explorer. The navigation menu on the left is divided into two sections: 'Packages' (A) and a list of classes/interfaces (B). The 'Packages' section lists various Java packages, and the second section lists classes and interfaces from the selected package. The main content area (C) displays the class hierarchy for `Frame`, showing its inheritance from `java.lang.Object`, `java.awt.Component`, `java.awt.Container`, `java.awt.Window`, and `java.awt.Frame`. It also lists 'All Implemented Interfaces' (Accessible, ImageObserver, MenuContainer, Serializable) and 'Direct Known Subclasses' (JFrame). The page includes navigation links for 'PREV CLASS', 'NEXT CLASS', 'FRAMES', 'NO FRAMES', 'SUMMARY', and 'DETAIL'.

Como usar a documentação (2)

Descrição de todos os pacotes da API Java

Descrição de todas as classes do pacote atual (java.awt)

Lista de pacotes que usam o pacote atual (java.awt)

Hierarquia de classes no pacote atual (java.awt)

Classes e métodos cujo uso não é mais recomendado

Índice com referências cruzadas (use para procurar métodos e campos quando não souber a classe)

[Overview](#) [Package](#) **[Class](#)** [Use](#) [Tree](#) [Deprecated](#) [Index](#) [Help](#)

[PREV CLASS](#) [NEXT CLASS](#)

[FRAMES](#) [NO FRAMES](#)

SUMMARY: [INNER](#) | [FIELD](#) | [CONSTR](#) | [METHOD](#)

DETAIL: [FIELD](#) | [CONSTR](#) | [METHOD](#)

Como usar a documentação

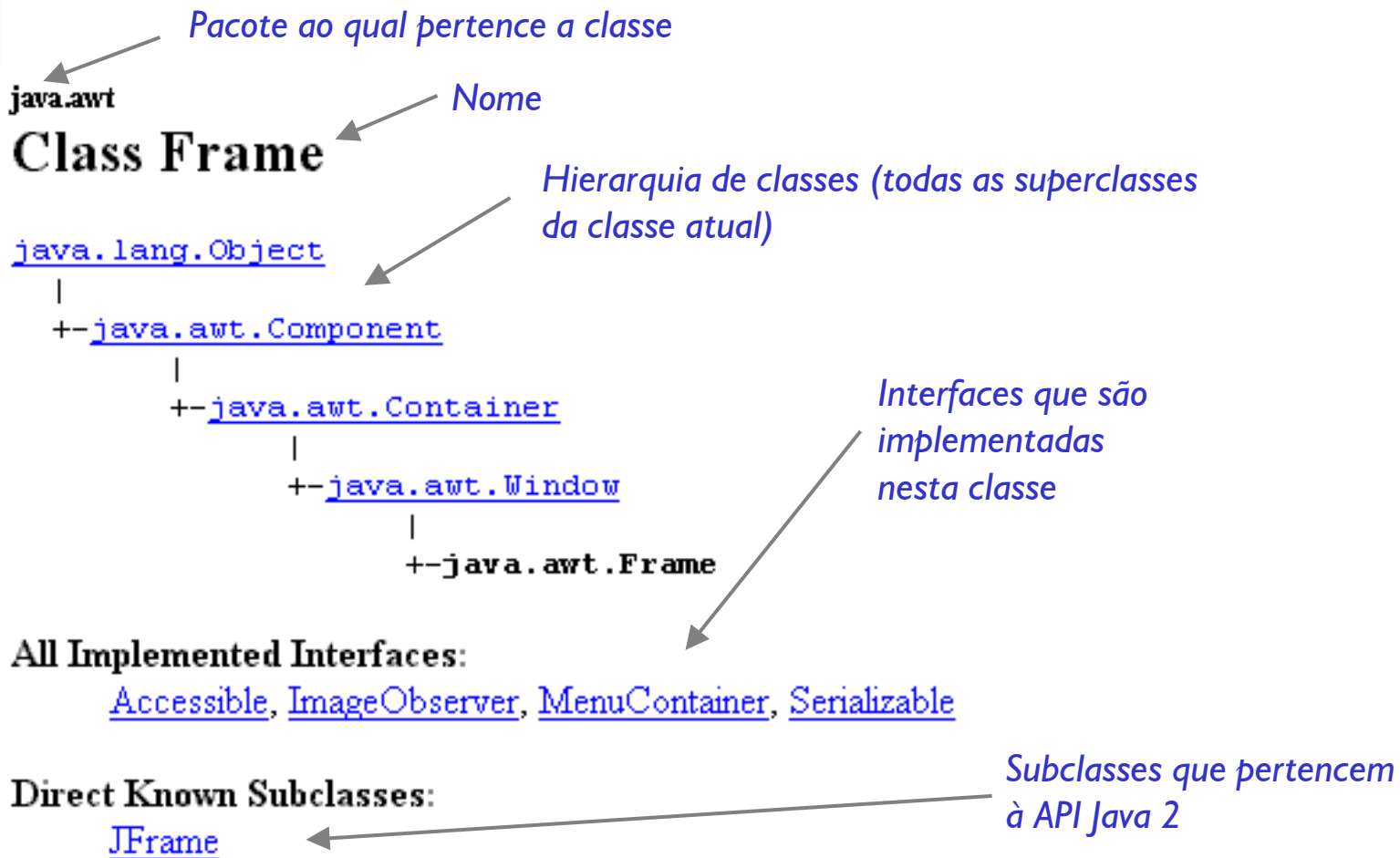
Links para esta página

- lista de classes internas
- lista de campos de dados
- lista de construtores
- lista de métodos

Links para esta página

- documentação de campos de dados
- documentação de construtores
- documentação de métodos

Como usar a documentação (3)



Como usar a documentação (4)

- *Listas de classes internas, campos de dados, métodos e construtores*

Constructor Summary	
Frame ()	Constructs a new instance of Frame that is initially invisible.
Frame (GraphicsConfiguration gc)	Create a Frame with the specified GraphicsConfiguration of a screen device.
Frame (String title)	Constructs a new, initially invisible Frame object with the specified title.
Frame (String title, GraphicsConfiguration gc)	Constructs a new, initially invisible Frame object with the specified title and a GraphicsConfiguration.

Lista contém breve descrição

Clique no nome para descrição detalhada

Tipos de retorno

Method Summary	
void	addNotify () Makes this Frame displayable by connecting it to a native screen resource.
protected void	finalize ()

Parâmetros

Documentação de um construtor

Frame

```
public Frame(String title,  
            GraphicsConfiguration gc)
```

Assinatura do construtor
(mostra tipos dos parâmetros)

Constructs a new, initially invisible Frame object with the specified title and a GraphicsConfiguration.

O que faz o construtor

Parameters:

title - the title to be displayed in the frame's border. A null value is treated as an empty string, "".

Descrição de cada parâmetro

gc - the GraphicsConfiguration of the target screen device. If gc is null, the system default GraphicsConfiguration is assumed.

Throws:

[IllegalArgumentException](#) - if gc is not from a screen device.

Exceções que o construtor pode provocar

See Also:

[Component.setSize\(int, int\)](#), [Component.setVisible\(boolean\)](#),
[GraphicsConfiguration.getBounds\(\)](#)

Métodos relacionados

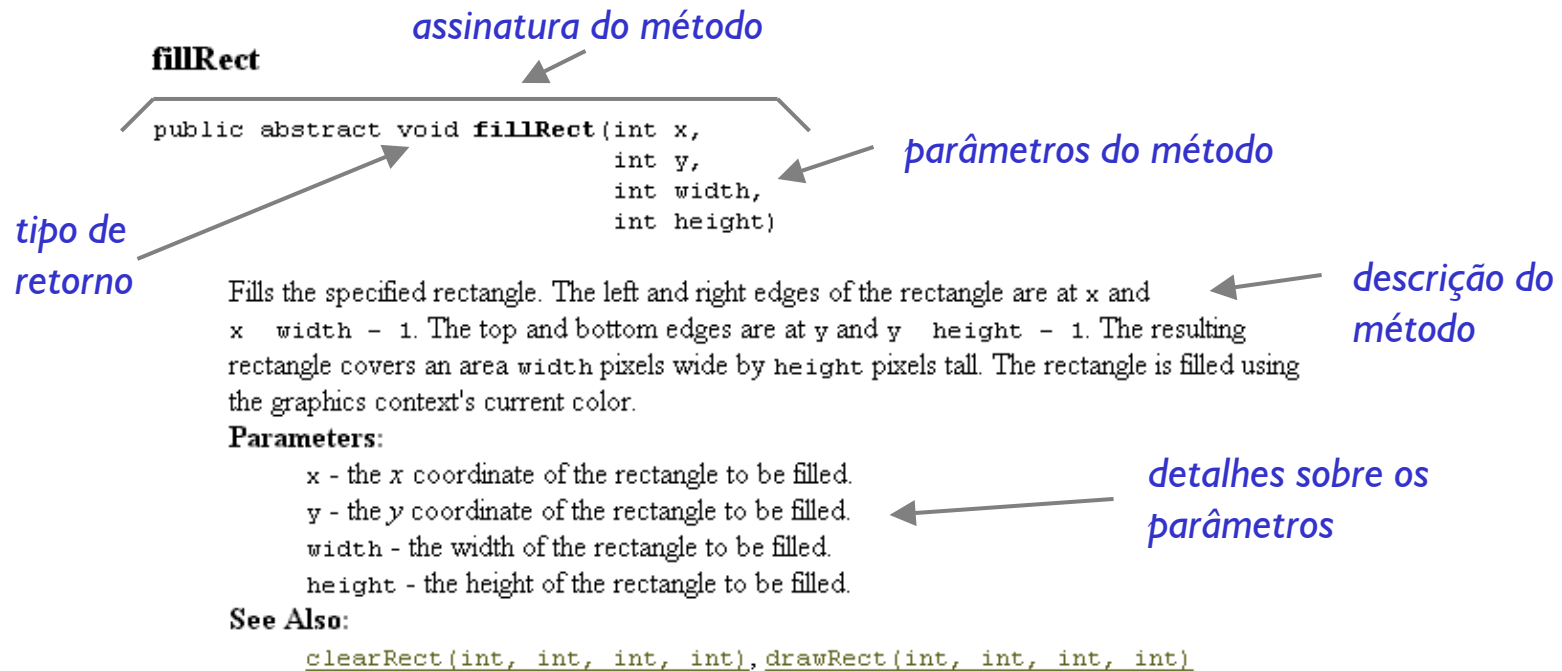
- Alguns exemplos de como usar o construtor acima

```
Frame f = new Frame("Título da Janela", null);
```

```
class MinhaJanela extends Frame {  
    public MinhaJanela(String titulo) {  
        super(titulo, null);
```

...

Documentação de um método



- Para chamar o método acima (`fillRect()` da classe `java.awt.Graphics`)
`g.fillRect(25, 50, 100, 200);` // *g: referência Graphics*
 - *Desenha um retângulo preenchido com a cor atual do contexto gráfico, com seu canto superior esquerdo na posição $x:25$ e $y:50$, com 100 pixels de largura por 200 de altura*
- *Para sobrepôr, repita a assinatura do método e forneça sua implementação*

Exercício 1: documentação

a) Crie duas classes

- Uma classe deve estender `javax.swing.JFrame` (esta classe irá fornecer a interface gráfica)
- Outra classe, executável (contendo `main`) para iniciar a classe gráfica.

b) Crie um construtor na classe derivada de JFrame

- O construtor deve definir o título da janela (use `super()` com os argumentos correspondentes ou descubra um método que faça isto)
- Deve definir o tamanho (`setSize`): 300x300, e tornar a janela visível (`setVisible`) - procure em `java.awt.Component`

c) Sobreponha o método `paint(Graphics g)` herdado de `java.awt.Component`

- Veja a assinatura correta na documentação. `paint()` é chamada pelo sistema automaticamente para pintar o contexto gráfico da aplicação
- Método `paint` deve usar contexto gráfico da janela (objeto `g`) para mudar a cor atual de `g` (para vermelho, por exemplo) (`setColor`) e desenhar, em `g`, um círculo (`fillOval`), tendo a metade da largura da janela (150), e posicionado no centro.

Ex 2: geração de documentação

- a) use comentários de documentação para descrever a classe que você criou, o construtor e o método `paint()`
- Use comentários `/** ... */` ANTES dos métodos, construtores e classes
 - Use `@param nome descrição` para descrever os parâmetros
 - `@param` é um comando do javadoc. Pode vir no início da linha (o início da linha pode ter espaços ou asteriscos `*`). Exemplo:

```
/** @param mensagem Texto contendo descrição... */
```
- b) Rode o javadoc passando a classe como argumento
- > `javadoc NomeDaClasse`
 - Navegue e explore os documentos HTML gerados
- c) Crie uma target no Ant para gerar documentação de todo o projeto (veja no capítulo 8 um resumo dos tags do Ant).

Curso J100: Java 2 Standard Edition

Revisão 17.0

© 1996-2003, Helder da Rocha
(helder@acm.org)

 argonavis.com.br