



## Como usar a documentação da API Java 2

*Helder da Rocha*  
[www.argonavis.com.br](http://www.argonavis.com.br)

- *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 in Microsoft Internet Explorer. The page is titled "Java 2 Platform SE v1.3 - Microsoft Internet Explorer fornecido por UOL". The address bar shows "C:\jdk1.3\docs\api\index.html".

The left sidebar (labeled A) contains a "Packages" list with links to [java.applet](#), [java.awt](#), [java.awt.color](#), [java.awt.datatransfer](#), [java.awt.dnd](#), and [java.awt.event](#). Below this is a "Font" section with links to [Font](#), [FontMetrics](#), [Frame](#), [GradientPaint](#), [Graphics](#), [Graphics2D](#), [GraphicsConfigTempl](#), [GraphicsConfiguration](#), [GraphicsDevice](#), [GraphicsEnvironment](#), [GridBagConstraints](#), [GridBagLayout](#), [GridLayout](#), [Image](#), [Insets](#), [JobAttributes](#), and [JobAttributes.Default](#).

The main content area (labeled B) shows the "Class Frame" documentation. It includes a "Overview" tab, a "Package" tab, and a "Class" tab. The "Class" tab is selected, showing the class hierarchy: [java.lang.Object](#) (parent), [java.awt.Component](#) (parent), [java.awt.Container](#) (parent), [java.awt.Window](#) (parent), and [java.awt.Frame](#) (current class). Below the hierarchy, it lists "All Implemented Interfaces: [Accessible](#), [ImageObserver](#), [MenuContainer](#), [Serializable](#)". It also lists "Direct Known Subclasses: [JFrame](#)".

The right sidebar (labeled C) shows the "Class Frame" documentation, including a "Description" section and a "Methods" section.

# Como usar a documentação (2)

*Descrição de todos os pacotes da API Java*

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

*Hierarquia de classes no pacote atual (java.awt)*

*Classes e métodos cujo uso não é mais recomendado*

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

*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*

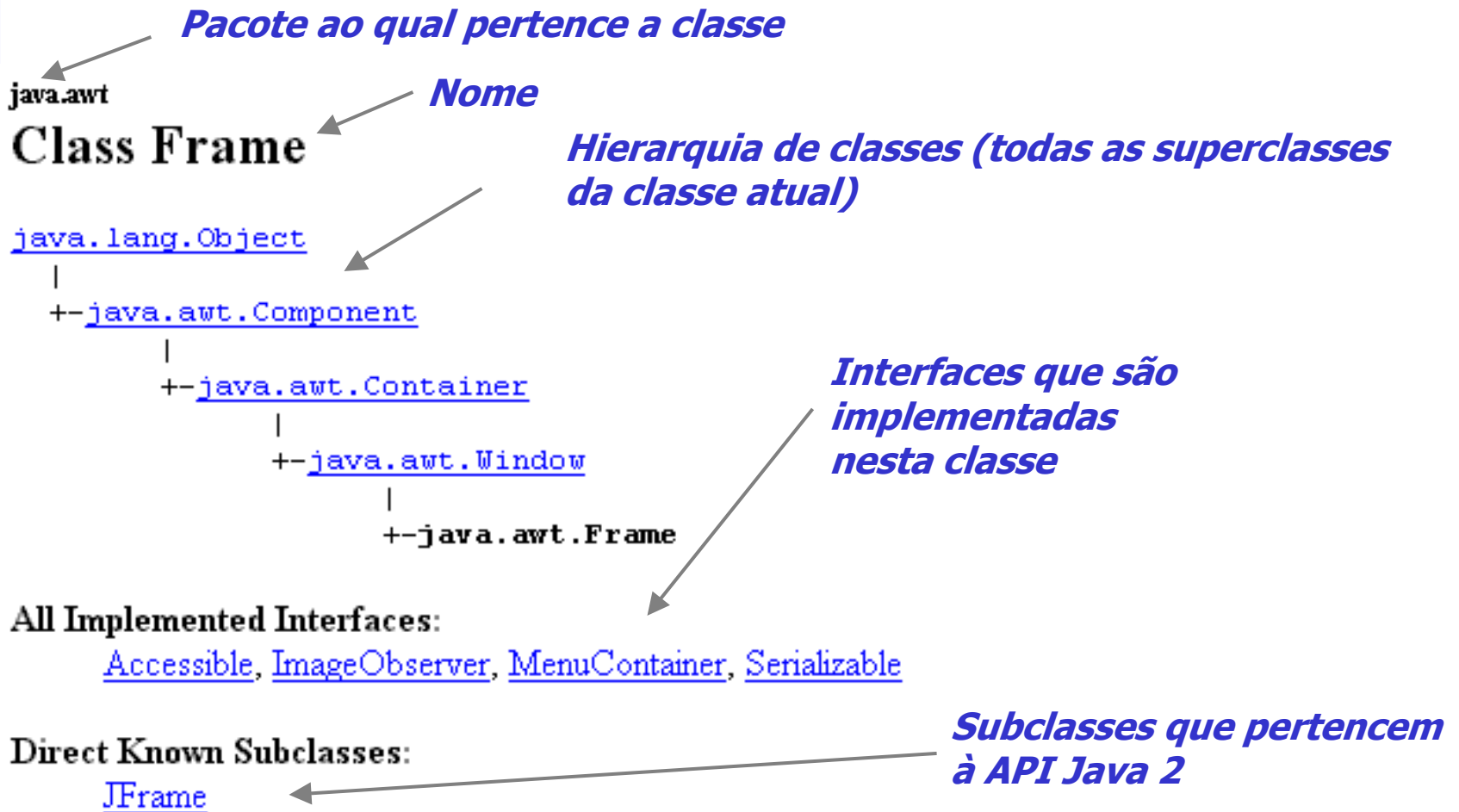
*Links para esta página*

- 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   |  |
|---|--|
| <a href="#">Frame()</a>                                       | Constructs a new instance of Frame that is initially invisible.  |
| <a href="#">Frame(GraphicsConfiguration gc)</a>               | Create a Frame with the specified GraphicsConfiguration of a screen device.                              |
| <a href="#">Frame(String title)</a>                           | Constructs a new, initially invisible Frame object with the specified title.                             |
| <a href="#">Frame(String title, GraphicsConfiguration gc)</a> | 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           | <a href="#">addNotify()</a><br>Makes this Frame displayable by connecting it to a native screen resource. |
| protected void | <a href="#">finalize()</a>  |

*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, "".

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

*Descrição de cada parâmetro*

### Throws:

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

*Exceções que o construtor pode provocar*

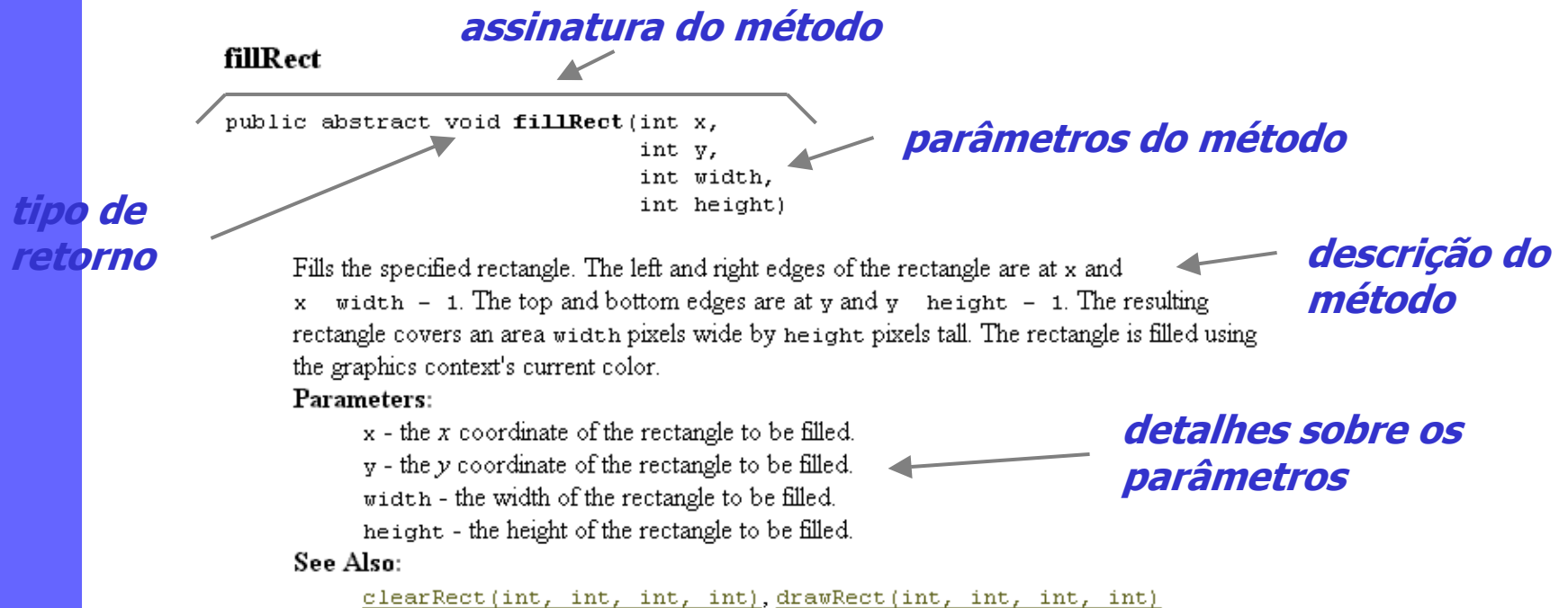
### See Also:

[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



- Como usar o método acima ( `Graphics.fillRect()` )
  - `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



# Exercício 1: documentação

- a) Crie duas classes
  - uma classe que estenda `javax.swing.JFrame` (esta classe irá fornecer a interface gráfica)
  - uma outra classe, executável (contendo `main`) para iniciar a classe gráfica
- b) A classe derivada de `JFrame` deve ter um construtor e ...
  - o construtor deve definir o título da janela (use `super()` )
  - deve definir o tamanho (`setSize`): 300x300, e tornar a janela visível (`setVisible`) - procure em `Component`
- c) ... sobrepor o método `paint(Graphics g)` herdado de `java.awt.Component`
  - método `paint` deve usar contexto gráfico da janela (objeto `g`) para
  - mudar a cor atual de `g` (para vermelho, por exemplo) (`setColor`)
  - 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