

TÉCNICAS DE PROGRAMACIÓN ORIENTADA A OBJETOS

14/05/2024 y 16/05/2024

UPN.EDU.PE

UNIDAD 2: RELACIONES DE CLASES DE HERENCIA SIMPLE Y MÚLTIPLE, COMPONENTES SWING Y ACCESO A DATOS



Sesión 15

- Paquete SWING
- Componentes
- Estructuras
- Ejemplos

LOGRO DE LA SESIÓN:



Al término de la sesión de aprendizaje, el estudiante entiende los conceptos de Arquitectura N Capas y componentes SWING.



AWT Y SWING

- Los componentes de Swing tienen nombres que comienzan con *J*.
 - ◆ Ejemplo: Button en AWT es JButton en Swing
- Los componentes de AWT están en el paquete `java.awt`, los de Swing en `javax.swing`.
- ```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;
```

# PAQUETES JFC/SWING

- **javax.swing.plaf**
- **javax.swing.plaf.basic**
- **javax.swing.plaf.metal**
- **javax.swing.plaf.multi**

- **javax.swing**
- **javax.swing.table**
- **javax.swing.tree**
- **javax.swing.border**
- **javax.swing.colorchooser**
- **javax.swing.filechooser**

- **javax.swing.text**
- **javax.swing.text.html**
- **javax.swing.text.html.parser**
- **javax.swing.text.rtf**

- **javax.swing.event**
- **javax.swing.undo**

# PAQUETES JFC/SWING

**Control del  
“Look & Feel”  
de Swing**

## Overview of JFC/Swing Packages

- javax.swing.plaf
- javax.swing.plaf.basic
- javax.swing.plaf.metal
- javax.swing.plaf.multi

- javax.swing
- javax.swing.table
- javax.swing.tree
- javax.swing.border
- javax.swing.colorchooser
- javax.swing.filechooser

- javax.swing.text
- javax.swing.text.html
- javax.swing.text.html.parser
- javax.swing.text.rtf

- javax.swing.event
- javax.swing.undo

**Componentes,  
incluyendo  
componentes  
complejos**

**Widgets basados en texto  
( incluyendo html/rtf )**

**Paquetes nuevos de eventos**

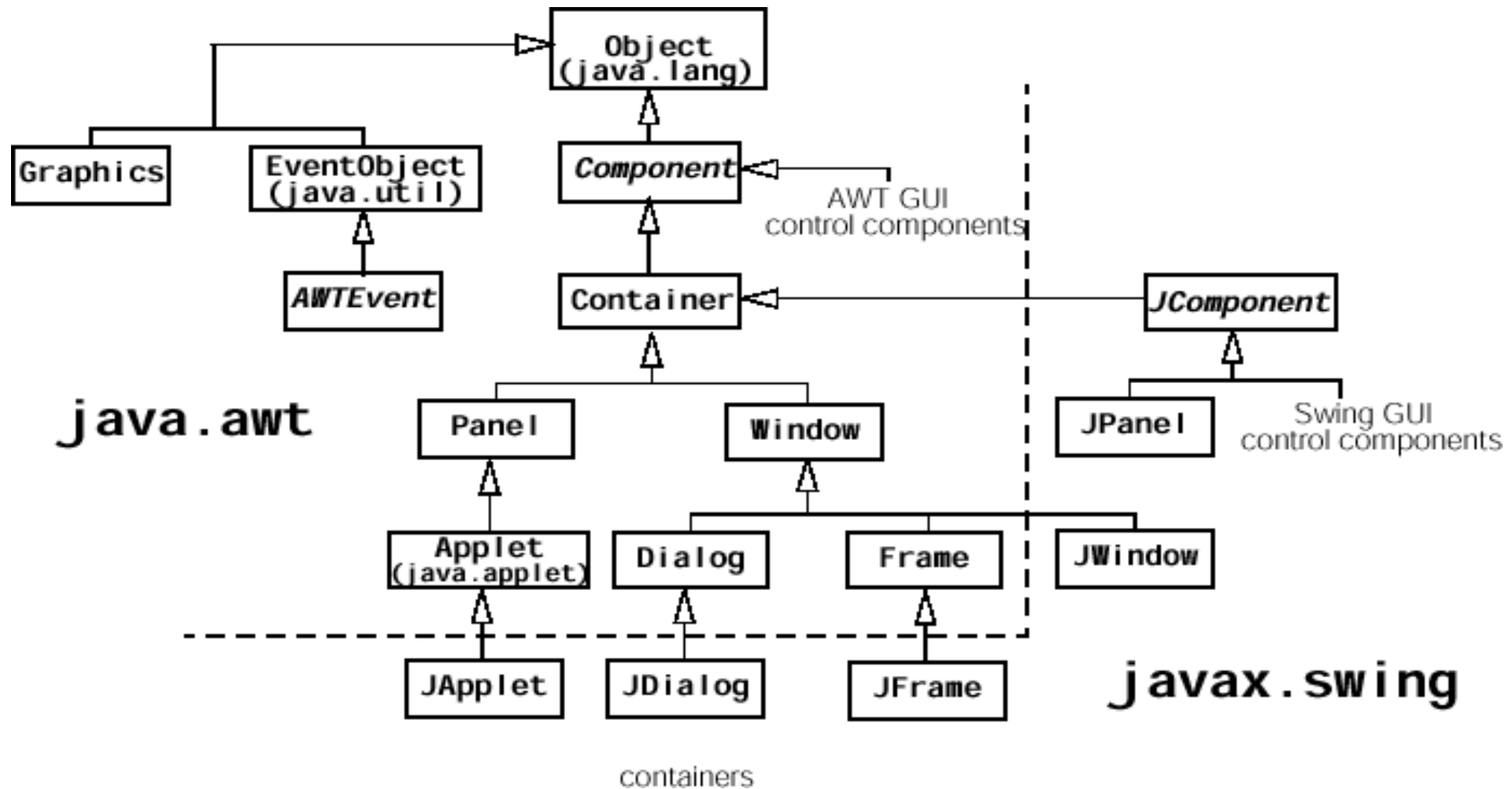


# APLICACIONES BASADAS EN GUI

El desarrollo de una aplicación basada en GUI requiere la comprensión de:

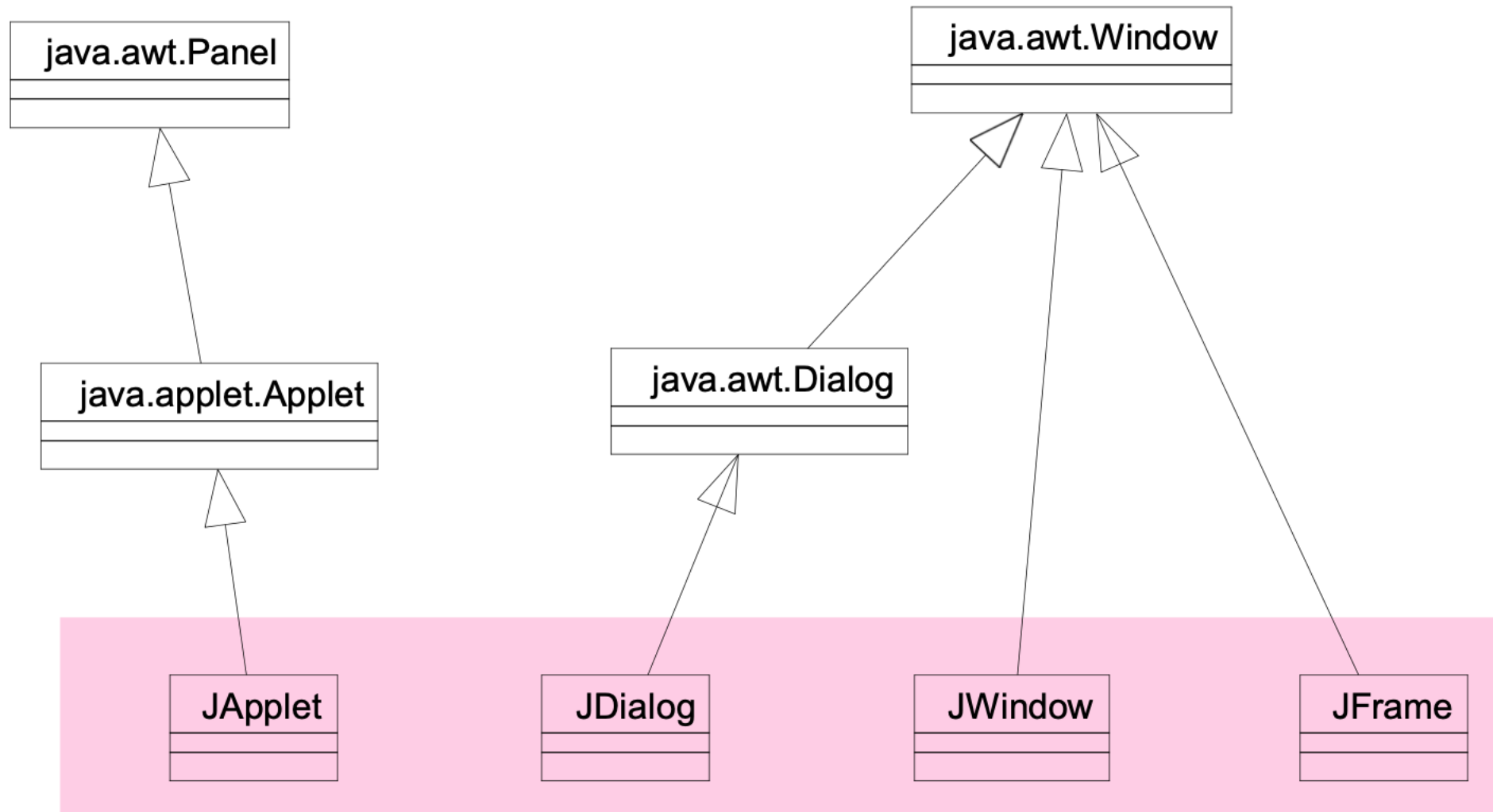
- Estructura de la jerarquía de herencia, que define el comportamiento y atributos de los componentes en la GUI de la aplicación.
- Estructura de la jerarquía de contenedores, que define cómo se disponen todos los componentes en la GUI de la aplicación.
- Manejo de eventos.

# JERARQUÍA DE HERENCIA

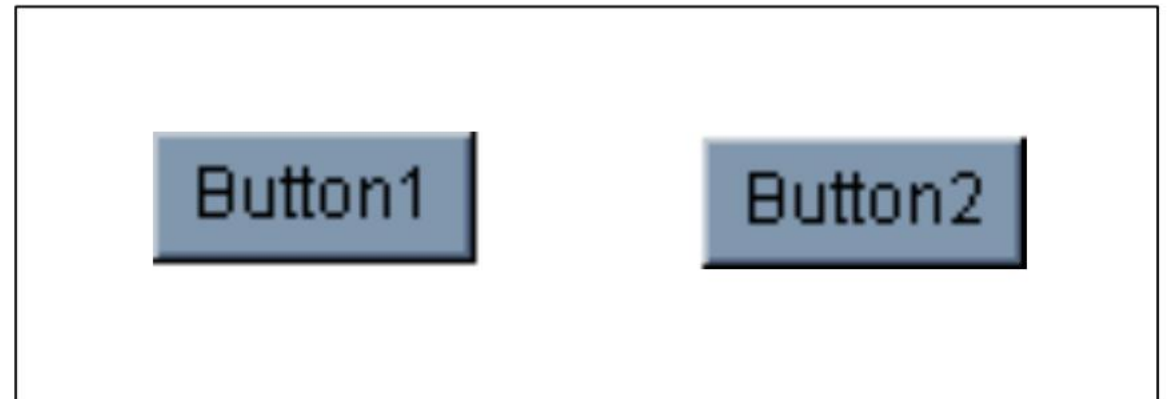




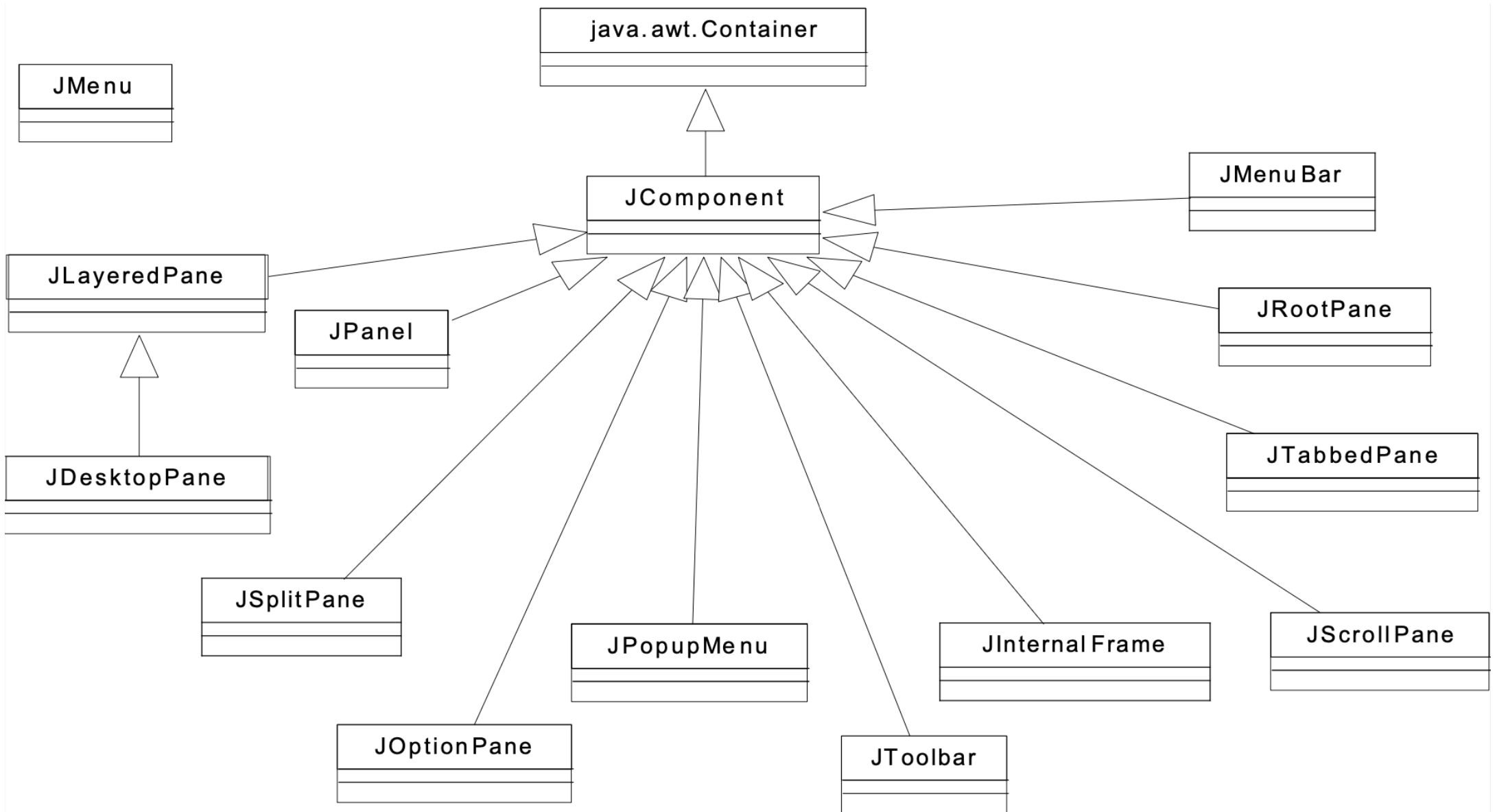
# ↑↑ HERENCIA "HEAVY"



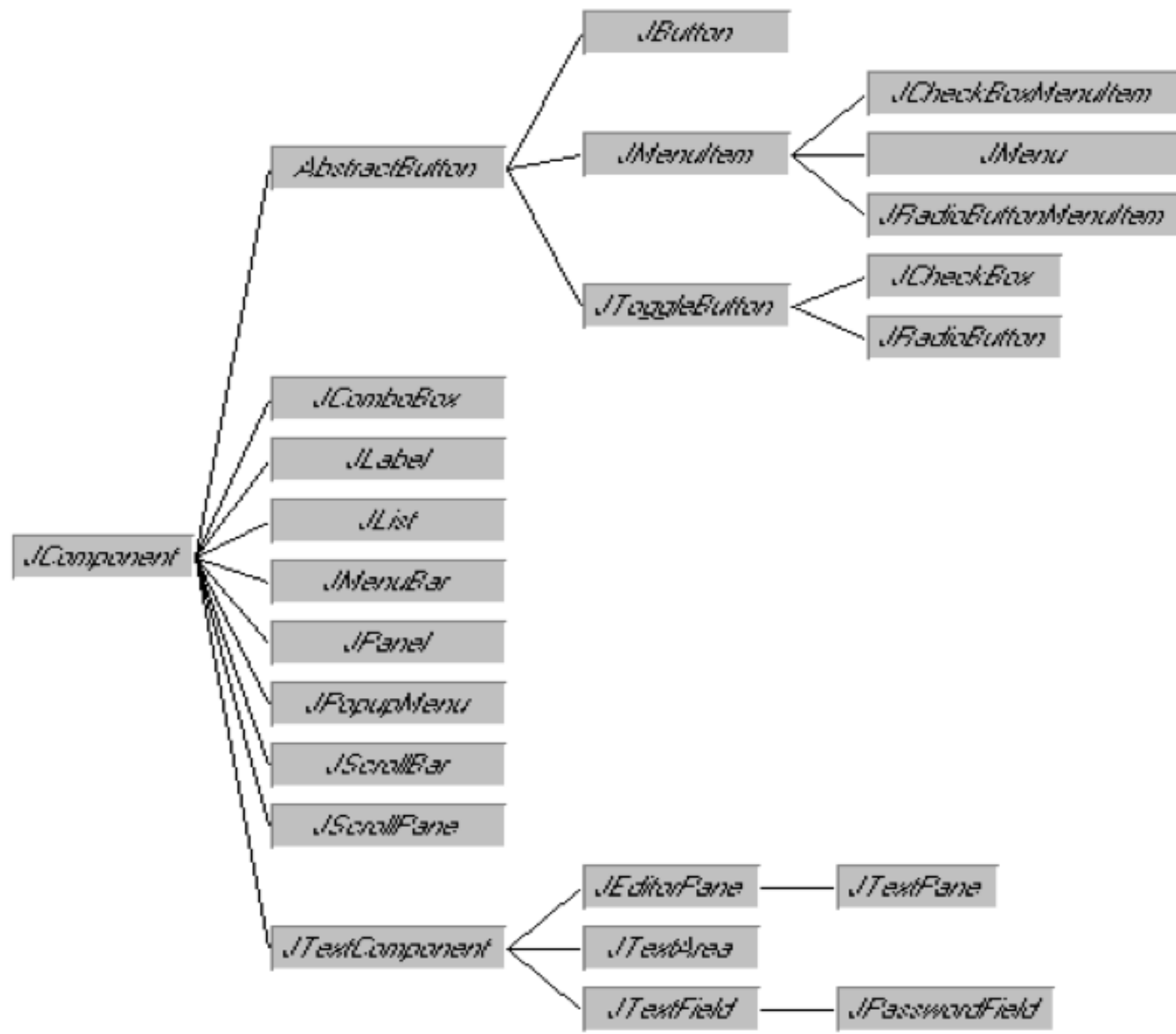
# ↑ HEAVYWEIGHT VS. LIGHTWEIGHT



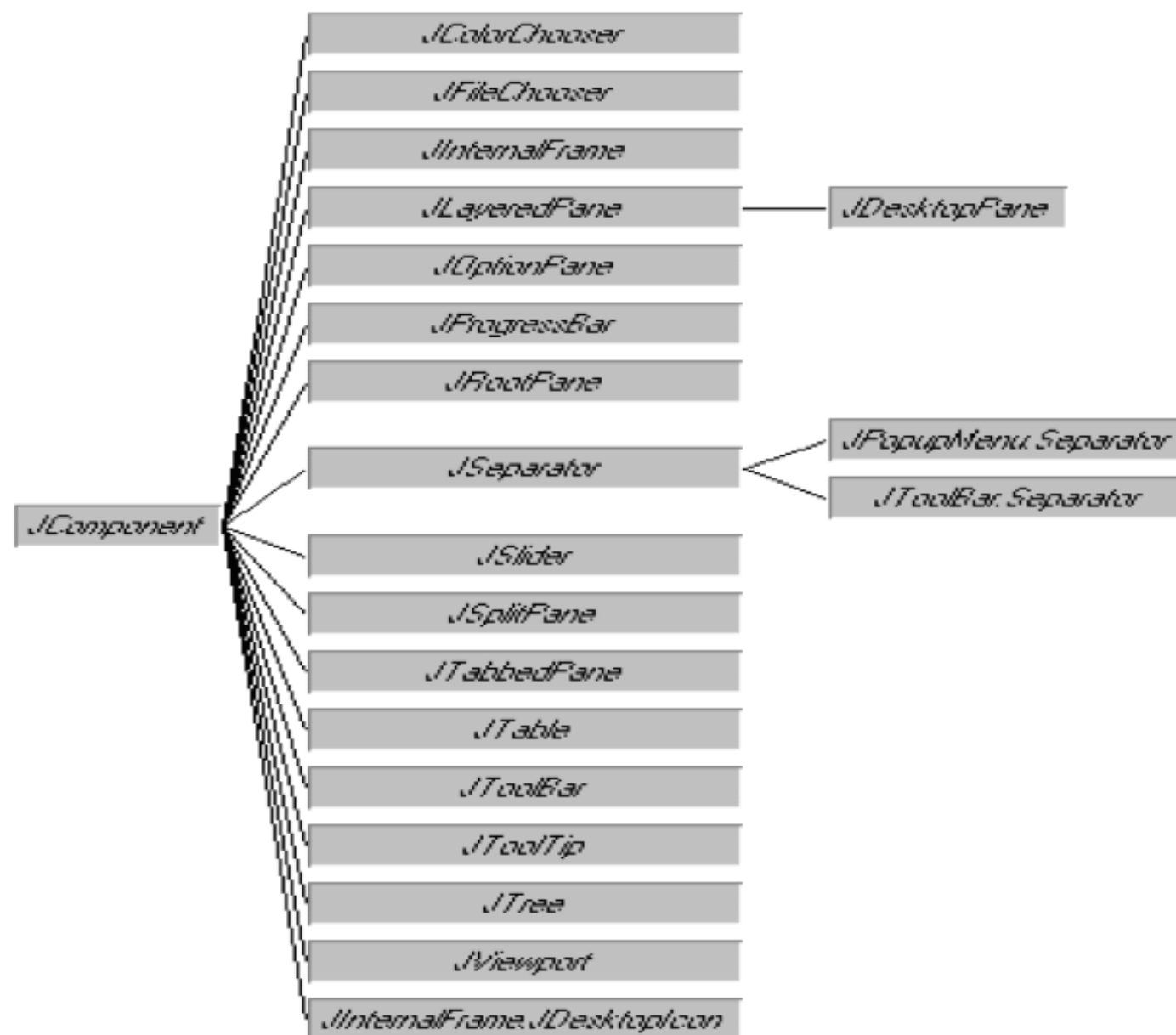
# CONTENEDORES EN SWING



# ↑↑ SIMILAR A AWT



# ↑↑ NUEVOS DE SWING





# COMPONENTES DE SWING

La clase Component (y sus subclases) proveen soporte para manejo de eventos, cambio del tamaño de un componente, control de color y fuentes, pintado.

Un componente es un objeto de una subclase concreta. Se distinguen dos clases de componentes:

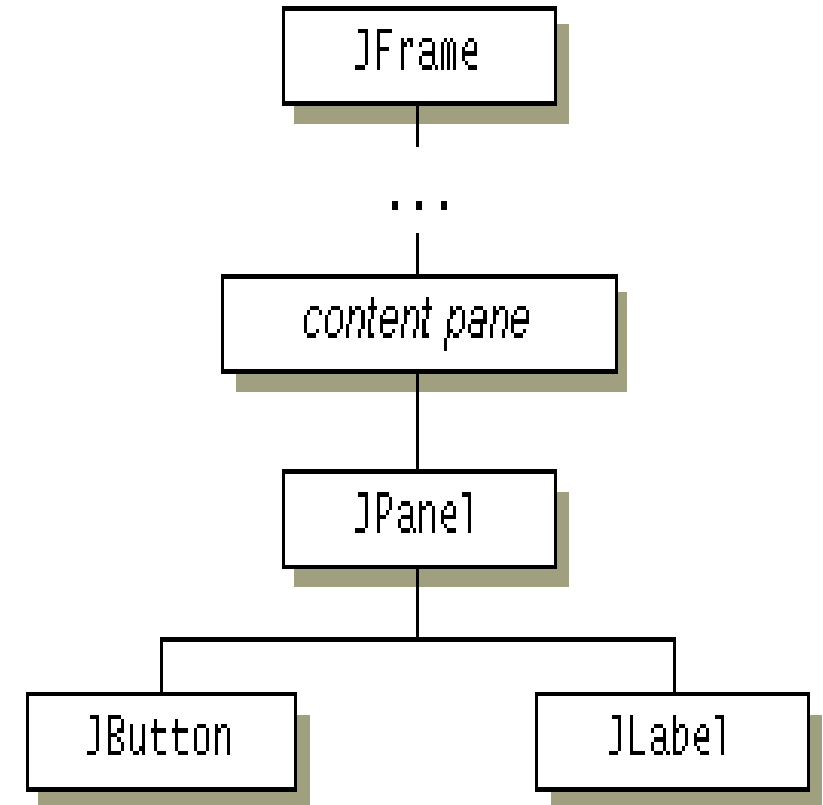
- Componentes de control de la GUI: la interacción de la GUI con el usuario se realiza a través de ellos.
- Contenedores: contienen otros componentes (u otros contenedores).



# CONTENEDORES

- Anidamiento de componentes (Jerarquía de contenedores en contraste con la Jerarquía de herencia). Cada programa Swing contiene al menos una.
- Usan un Layout Manager para determinar cómo se disponen los componentes en los contenedores.
- Swing provee 4 contenedores de alto nivel (ventana base de la GUI): JFrame, JApplet, JDialog y JWindow.
- La jerarquía está compuesta de diferentes capas.
- Cada contenedor de alto nivel contiene un contenedor intermedio conocido como “content pane”. En casi todos los programas no es necesario conocer qué hay entre el contenedor de alto nivel y el content pane.

# JERARQUÍA DE CONTENEDORES







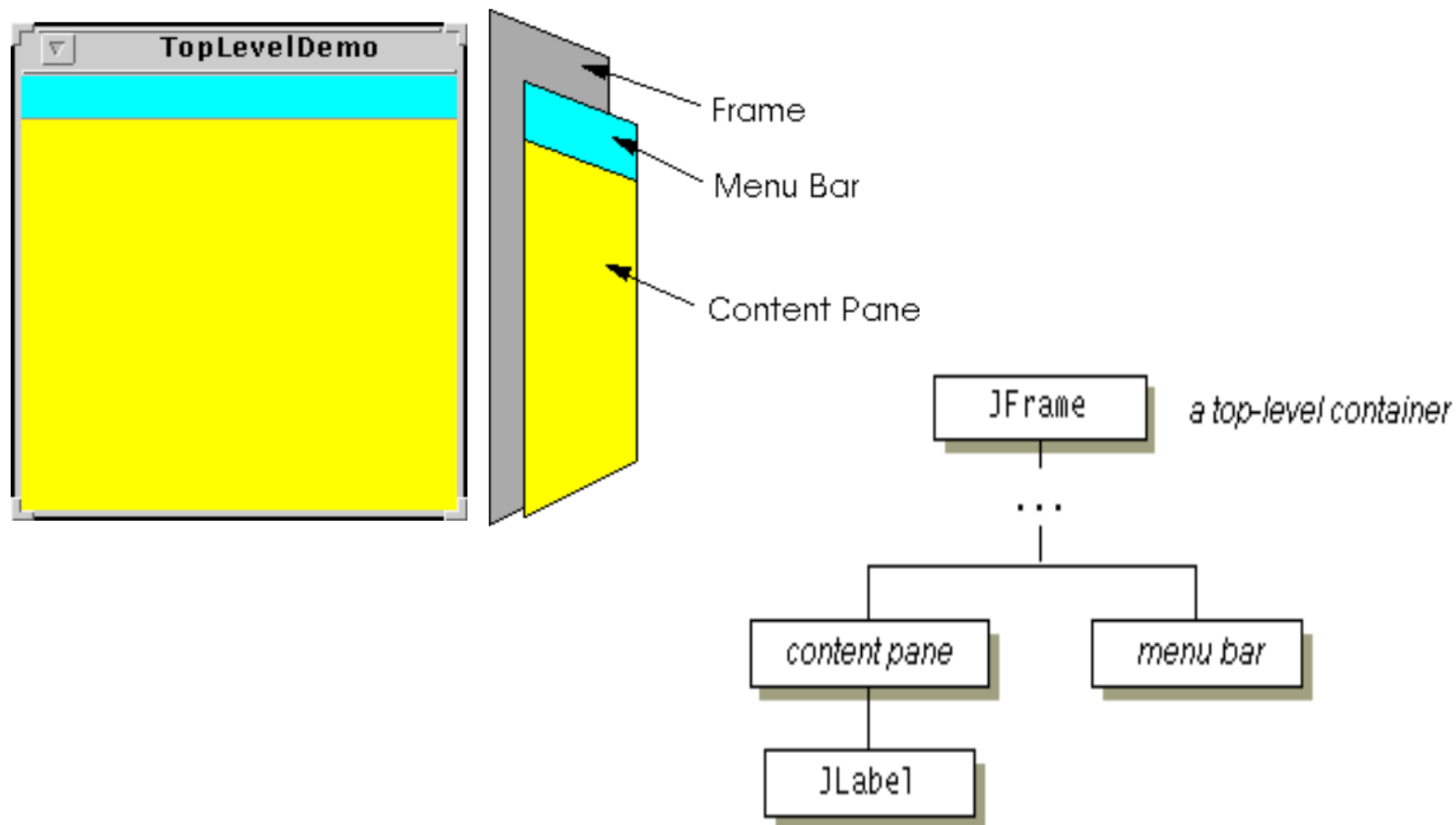
# JERARQUÍA DE CONTENEDORES

La apariencia de una GUI está determinada por:  La jerarquía de contenedores

- El Layout Manager de cada contenedor
- Las propiedades de los componentes individuales

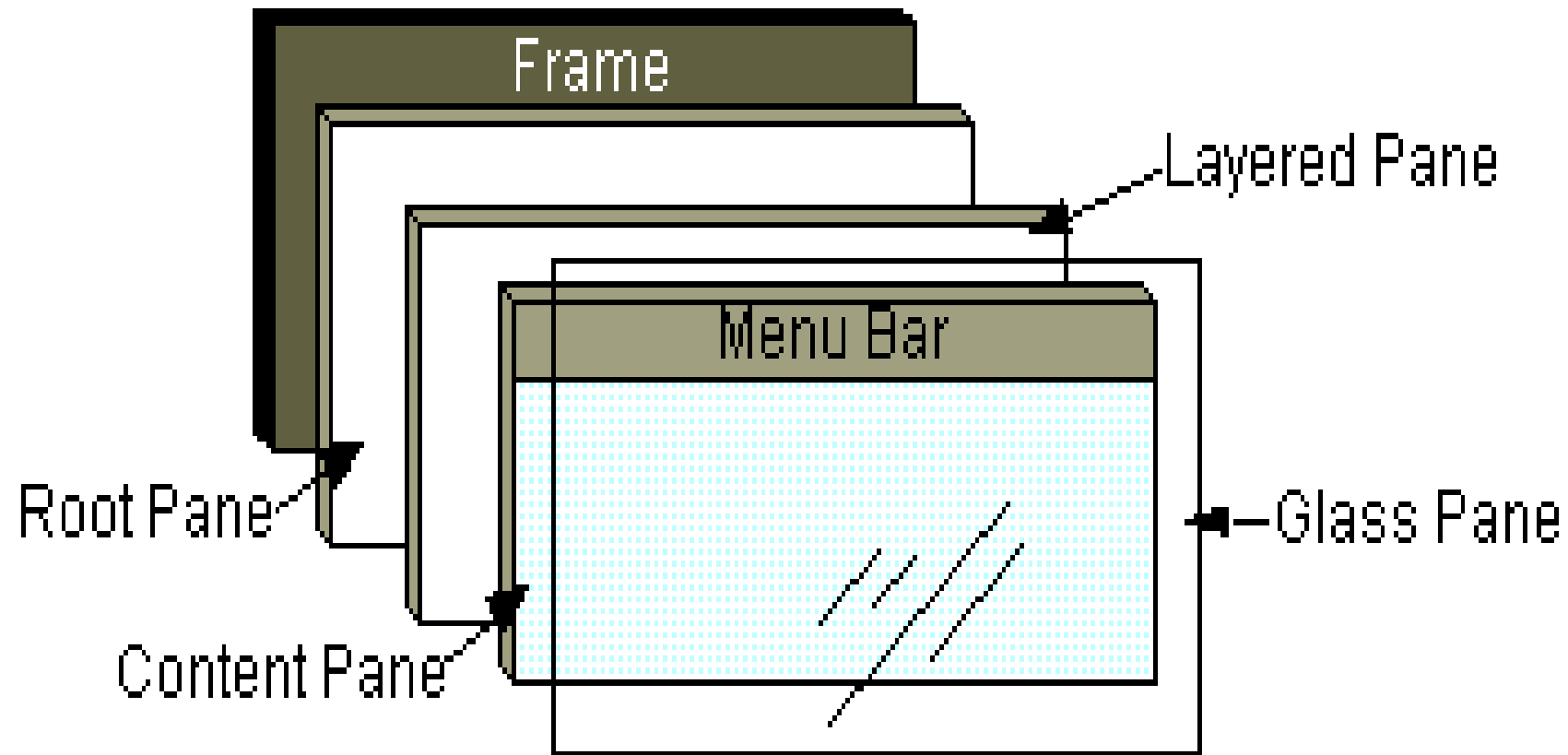
Todos estos ítems trabajan en conjunto para determinar el efecto visual final.

# ESTRUCTURA DE UN JFRAME





# ESTRUCTURA DE UN JFRAME





# ROOT PANES

- “Añadido” en forma invisible al contenedor de alto nivel. ▪Creado por Swing cuando instancia un contenedor de alto nivel.
- Maneja prácticamente todo entre el contenedor de alto nivel y los componentes atómicos.
- Tener en cuenta si necesita interceptar clicks del mouse o pintar sobre varios componentes.
- Es una instancia de JLayeredPane la que contiene la barra de menús y el content pane.

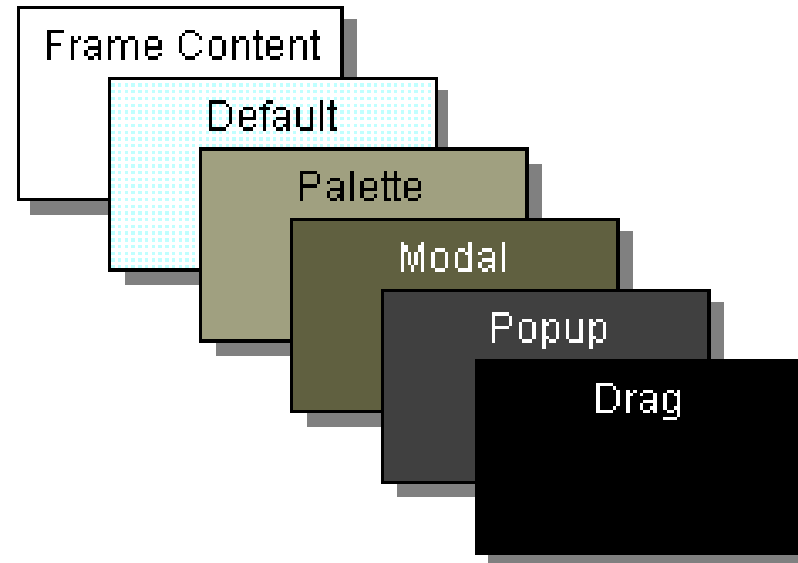


## CONTENT PANES

- Usualmente es un JPanel.
- En la mayoría de las aplicaciones Swing contiene casi todo, excepto la barra de menú.
- Debe ser creado explícitamente.

# LAYERED PANES

- Provisto por root pane pero también puede crearse.
- Contenedor con profundidad, tal que componentes que se superponen (ej:popup menus) pueden aparecer unos encima de otros (z-buffering).



# GLASS PANES

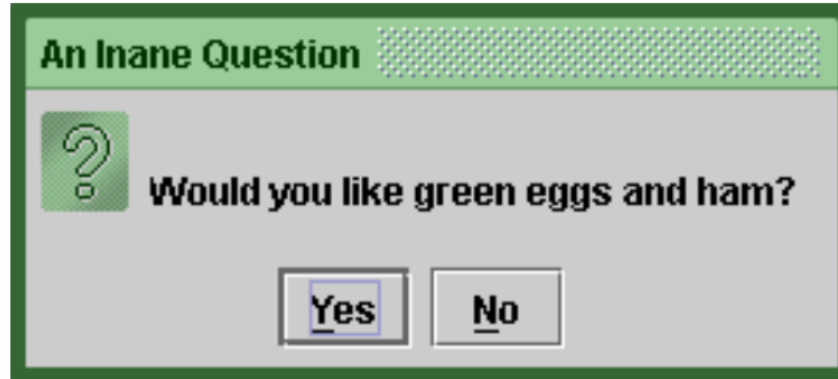
- Util para pintar o interceptar eventos (por ejemplo: bloquear todos los eventos del mouse) en un área que contenga uno o más componente.



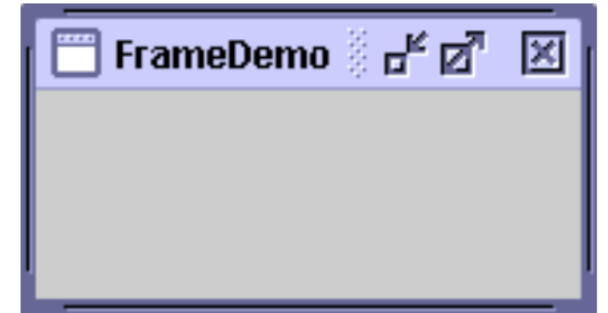
# CONTENEDORES DE ALTO NIVEL



Applet



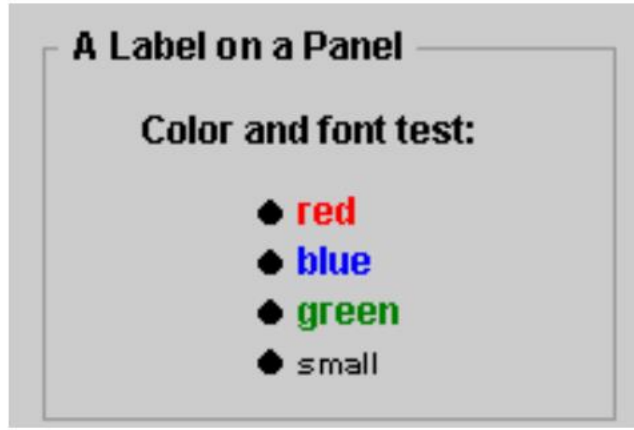
Dialog



Frame



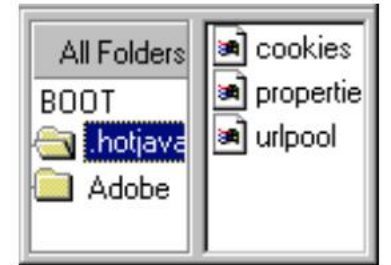
# CONTENEDORES GENERALES



Panel



Scroll Pane



SplitPane

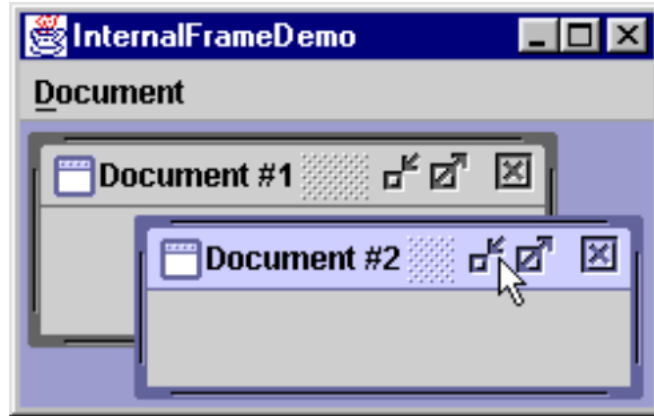


TabbedPane

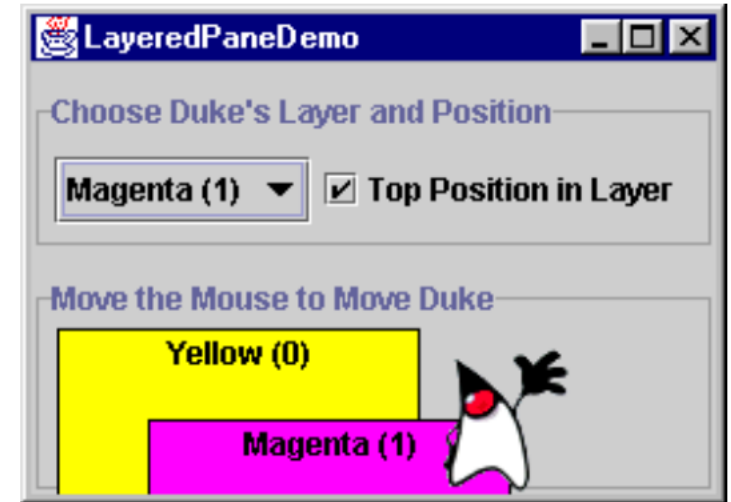


Toolbar

# CONTENEDORES ESPECIALES

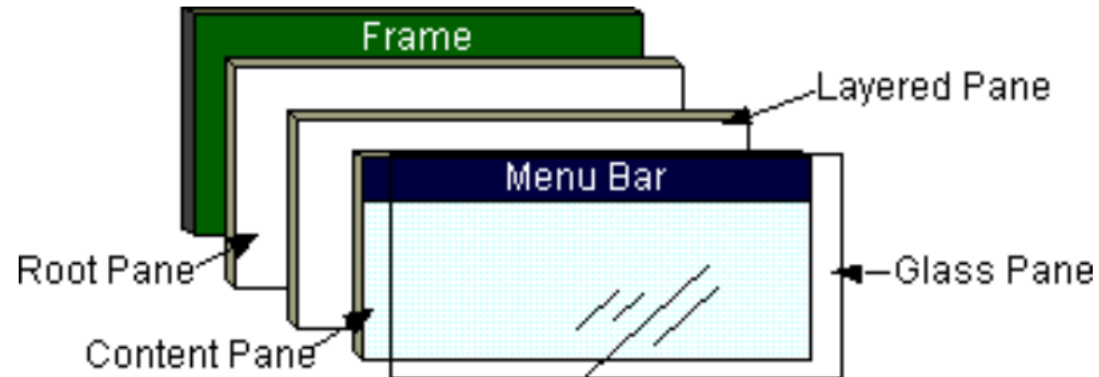


InternalFrame



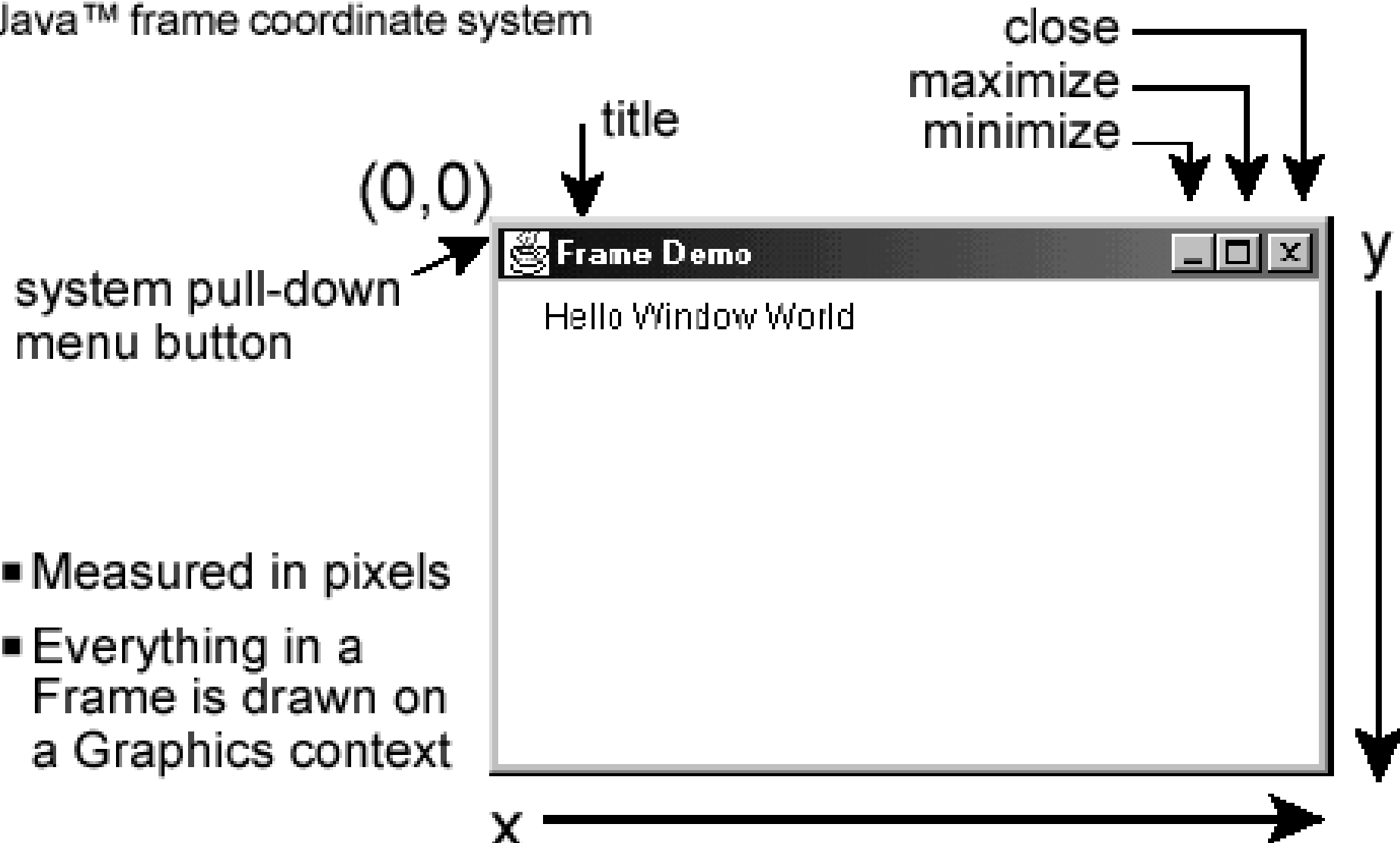
LayeredPane

Root Pane



# JFRAME

Java™ frame coordinate system



- Measured in pixels
- Everything in a Frame is drawn on a Graphics context

# ALGUNOS MÉTODOS DE JFRAME

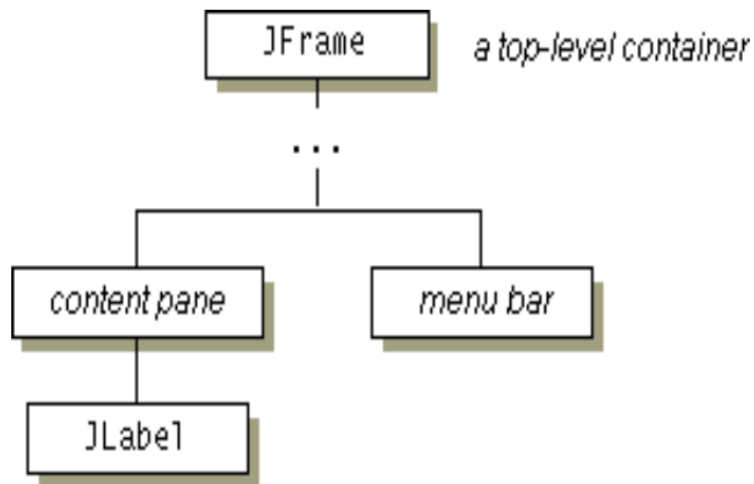
- Para añadir componentes al content pane:

```
myFrameInstance.getContentPane().add(myComponent);
```

Usual  
"this"

Requerido por  
Instancias de  
*JFrame*,  
*JDialog* y  
*JInternalFrame*

Ej.:  
"myLabel"



# ALGUNOS MÉTODOS DE JFRAME

- Para construir una ventana con un título y mostrarla:

```
JFrame theWindow = new JFrame("Graffiti");
theWindow.show();
//idem theWindow.setVisible(true);
```



- Para determinar su tamaño:

```
theWindow.setJMenuBar(cyanMenuBar);
theWindow.setSize(220, 100);//o mejor
theWindow.pack();
```



# NUEVA FUNCIONALIDAD DE JFRAME

- `setDefaultCloseOperation(int)` es la más importante:
  - ◆ `DO_NOTHING_ON_CLOSE`
  - ◆ `HIDE_ON_CLOSE` (default) oculta el frame cuando el usuario lo cierra pero no se deshace de los recursos del sistema asociados (puedo volver a mostrar).
  - ◆ `DISPOSE_ON_CLOSE` oculta el frame y llama al método `dispose()`, para liberar recursos.
  - ◆ `EXIT_ON_CLOSE`, cierra la aplicación (`System.exit(0)`)

# HELLO WORLD EN SWING

```
import javax.swing.*;

public class HelloWorldSwing {

 public static void main(String[] args) {// crear un nuevo frame

 JFrame frame = new JFrame("HelloWorldSwing");

 // crear una etiqueta y añadir al frame

 JLabel label = new JLabel("Hello World");
 frame.getContentPane().add(label);

 // especifica la operación de cierre de la ventana

 frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

 // muestra el frame en pantalla

 frame.pack(); frame.setVisible(true);
 }
}
```





# COMPONENTES ATÓMICOS

- Componentes de tipo JPanel almacenan otros objetos de la GUI tales como: botones, etiquetas, campos de texto, etc. Estos objetos gráficos son considerados componentes atómicos, puesto que no pueden almacenar otros objetos de la GUI.

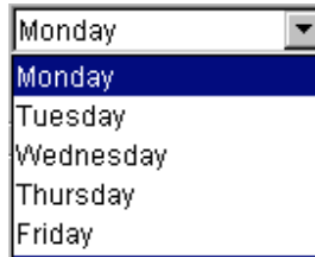




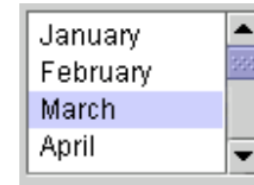
# COMPONENTES ATÓMICOS BÁSICOS



Buttons



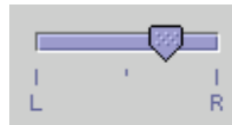
Combo Box



List



Menu



Slider



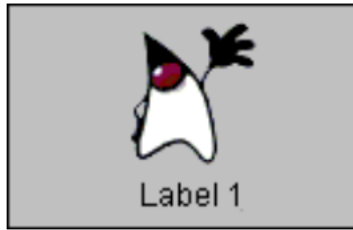
Spinner



Text Fields



# COMPONENTES ATÓMICOS NO EDITABLES



Label



Progress Bar

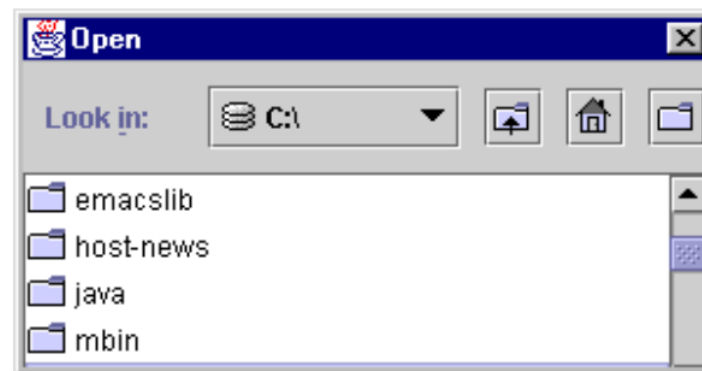


Tool tip

# ↑ OTROS COMPONENTES ATÓMICOS



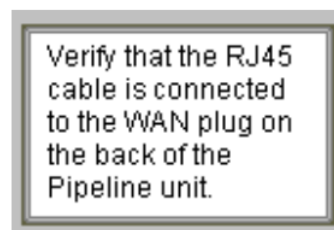
Color Chooser



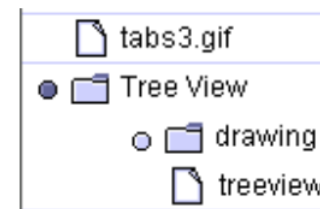
File Chooser

| First Na... | Last Name |
|-------------|-----------|
| Mark        | Andrews   |
| Tom         | Ball      |
| Alan        | Chung     |
| Jeff        | Dinkins   |

Table



Text



Tree

# UNIDAD 2: RELACIONES DE CLASES DE HERENCIA SIMPLE Y MÚLTIPLE, COMPONENTES SWING Y ACCESO A DATOS



## Sesión 16

- REPASO / EVALUACIÓN FINAL

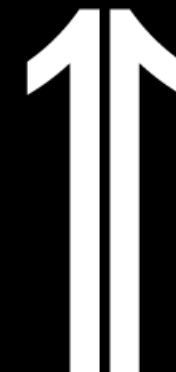
# APLICANDO LO APRENDIDO



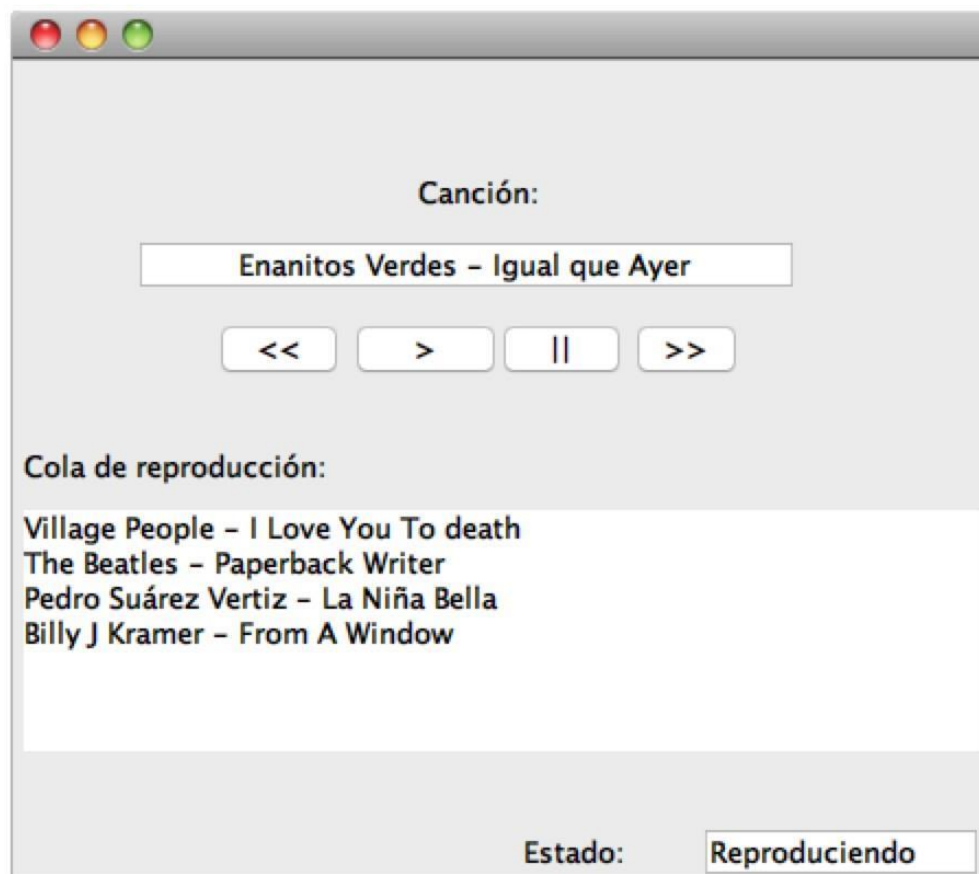
- Veremos ejemplos de casos donde se aplique el paradigma POO.
- Partiremos del análisis de casos, la elaboración de los diagramas de casos correspondientes y la codificación.
- Utilizaremos los estándares y buenas prácticas revisados durante el curso.

# EJEMPLOS DE LO APRENDIDO

## CASO 1



### IMPLEMENTACIÓN DE UN REPRODUCTOR DE AUDIO



# EJEMPLOS DE LO APRENDIDO

## CASO 2



### SISTEMA DE CONTROL PARA UNA SALA DE CINE

The interface is a window with a title bar containing three colored buttons (red, yellow, green). It features a 'Sala:' label followed by a dropdown menu. Below this is a grid of 12 seat selection buttons arranged in four rows and three columns. The first row contains buttons for seats 1, 2, and 3; the second for 4, 5, and 6; the third for 7, 8, and 9; and the fourth for 10, 11, and 12. To the right of the grid is a 'Precio por entrada' label and a text input field showing 'S/. 20,00'. Below the grid is a 'Liquidar' button. At the bottom of the window is a text area containing the following messages: 'Se ha seleccionado el asiento 5', 'Se ha seleccionado el asiento 1', 'Se ha seleccionado el asiento 11', followed by three dashes, 'Se ha realizado la liquidación.', 'El total es: S/. 60,00', another set of three dashes, and 'Datos guardados en el archivo correctamente'.

Sala:

2 3

4  6

7 8 9

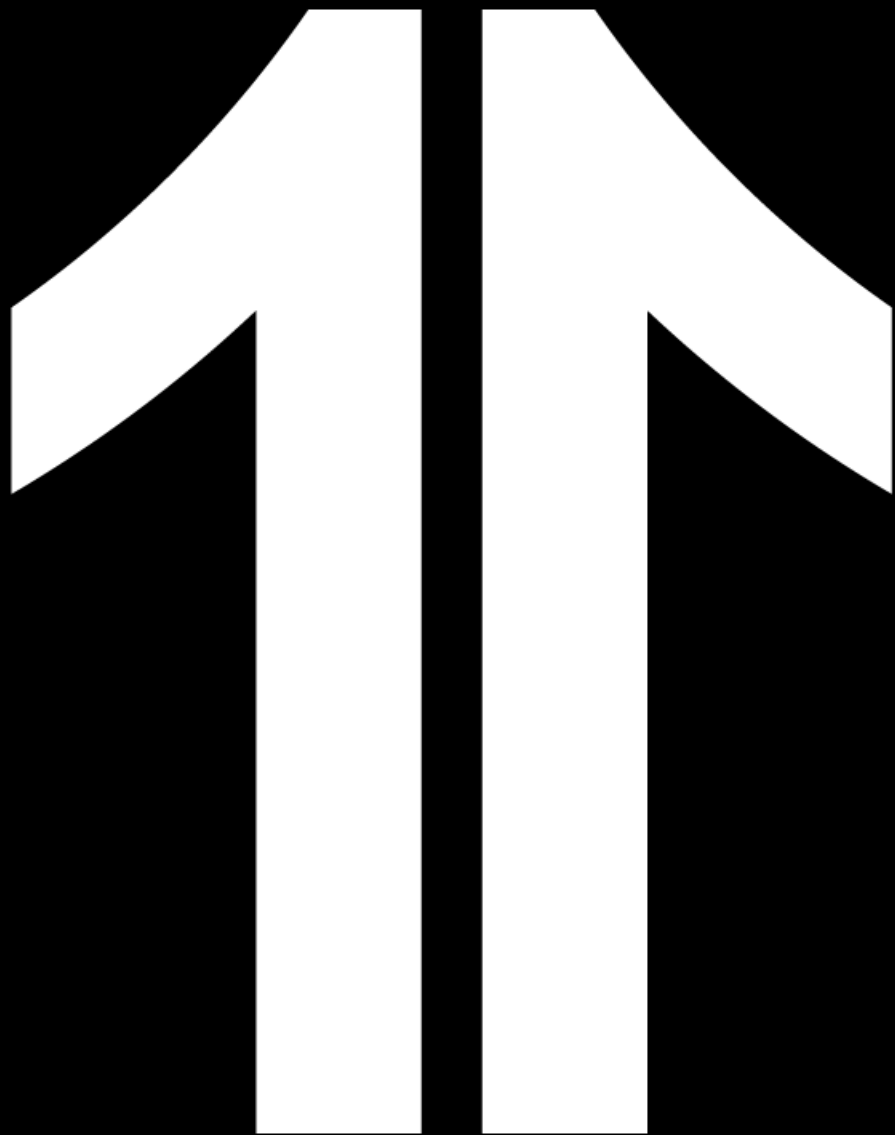
10  12

Precio por entrada

S/. 20,00

Liquidar

Se ha seleccionado el asiento 5  
Se ha seleccionado el asiento 1  
Se ha seleccionado el asiento 11  
---  
Se ha realizado la liquidación.  
El total es: S/. 60,00  
---  
Datos guardados en el archivo correctamente



**GRACIAS**



