



## Tema1. Introducción al Diseño de Interfaces con Swing

---



## Contenidos

---

- **JFC**
- **AWT**
- **Swing**
  - **Estructura básica de Swing**
  - **Componentes**
  - **Contenedores**
  - **Componentes Básicos implicados en la construcción de IGU**





JFC

- JFC es la abreviatura de *Java Foundation Classes* y representa un conjunto de clases que ayudan en la construcción de interfaces de usuario
- Está constituido por 5 librerías:
  - **AWT.** Permite la construcción de IGU para todas las versiones del JDK
  - **Swing.** Basada en la anterior, permite la construcción de IGU más avanzadas que las construidas con AWT.
  - **Accesibility.** Proporciona asistencia a usuarios que tienen problemas con los interfaces de usuario tradicionales. Todos los componentes Swing soportan accesibilidad
  - **2D API.** Incorpora clases para implementar diferentes formas, fuentes, colores, etc.
  - **Drag and Drop.** Permiten implementar el drag and drop sobre diferentes elementos de las aplicaciones



# 1

## Abstract Window Toolkit (AWT)





## AWT - Características

- Permite la construcción de Interfaces Gráficas de Usuario (IGU) que pueden ejecutarse en cualquier sistema y para todas las versiones del JDK (Java Development Kit)
- Muestra los componentes (botones, etiquetas, campos de texto) utilizando (por defecto) el aspecto de la plataforma local



## AWT – Elementos implicados

- En una aplicación construida con AWT se distinguen:
  - Componentes
  - Contenedores
  - Eventos
  - Administradores de Diseño ( layouts)
- Los **componentes** (botones, etiquetas, etc.) de la IGU se organizan en **contenedores** y esperan hasta que el usuario ejecuta alguna acción sobre ellos (**evento**)
- A su vez los contenedores son componentes y pueden por tanto ser agrupados en otros contenedores
- Por lo general en un contenedor habrá varios componentes y para situarlos de forma conveniente pueden usarse los **administradores de diseño**





## AWT – Elementos implicados

- **En resumen, las clases del paquete AWT pueden clasificarse en :**
  - **Gráficos**
  - **Componentes**
  - **Administradores de Diseño ( layouts)**
  - **Manipuladores de eventos**
  - **Manipuladores de imágenes**
- **Toda esta funcionalidad se encuentra en los paquetes:**
  - **java.awt.\*;**
  - **java.awt.event.\*;**



---

## 2 Swing



# Swing

- Swing es la biblioteca de clases más importante de JFC
- Surge en 1997 debido a que AWT presentaba muchas limitaciones a la hora de construir aplicaciones de cierta entidad
- La mayoría de la funcionalidad que necesitamos se encuentra en el paquete:
  - **javax.swing**
- En Swing, los componentes
  - Están escritos en Java
  - Proporcionan un Look&Feel (L&F) consistente entre plataformas
  - Se pueden usar sobre cualquier plataforma que soporte Java 1.1 o posteriores
- A partir de JDK 1.2 Swing forma parte de la distribución estándar de Java (no hay que hacer nada especial para emplear sus componentes)



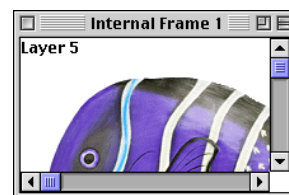
# Swing – Diferencias con AWT (I)

Swing permite cambiar el Look&Feel de la aplicación en tiempo de ejecución

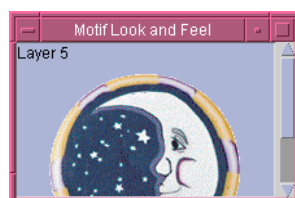
Metal



Mac OS



Motif



Windows





## Swing – Diferencias con AWT (II)

- Swing permite la construcción de IGU para aplicaciones de cierta entidad
- Swing incorpora un mayor número de componentes que AWT: tablas, árboles, sliders, spinners, barras de progreso, frames internos y componentes de texto. También permite añadirles bordes a los componentes.
- Los componentes Swing pueden tener tooltips colocados sobre ellos
- Swing permite ligar eventos de teclado a componentes



# 3

## Estructura Básica de Swing





## Estructura Básica de Swing

- La estructura básica de Swing se basa en
  - Componentes y
  - Contenedores
- La interacción con el usuario y el sistema se realiza por medio de eventos



## Componentes

- Los componentes son objetos que permiten al usuario interactuar con la aplicación
- Un componente está formado por
  - Propiedades
  - Métodos
  - Eventos
- Los componentes (botones, comboboxes, listas,...) representan los elementos básicos para la construcción de los programas

Ok



## Propiedades

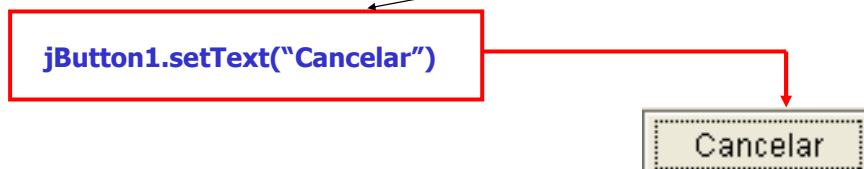
- Representan un valor o estado asociado al componente
- Las propiedades son atributos y algo más... Ya qué:
  - No es solamente una variable cuyo valor se puede leer o escribir directamente
  - Normalmente tiene asociado una acción que se ejecuta con la modificación de la propiedad

### Componente JButton



## Propiedades (II)

- Las propiedades tienen un valor antes de que se use el componente
- El valor de una propiedad puede cambiarse en tiempo de ejecución



- Los resultados de modificar una propiedad son inmediatos al desencadenarse la acción ligada a la propiedad





## Métodos

- Los métodos de un componente funcionan tal y como lo haría cualquier método de una clase de Java
- Cada método proporciona un comportamiento de un componente, ya que obligan al componente a realizar una acción
- Ej.

```
TextArea1.setText (" Un texto de prueba")
```



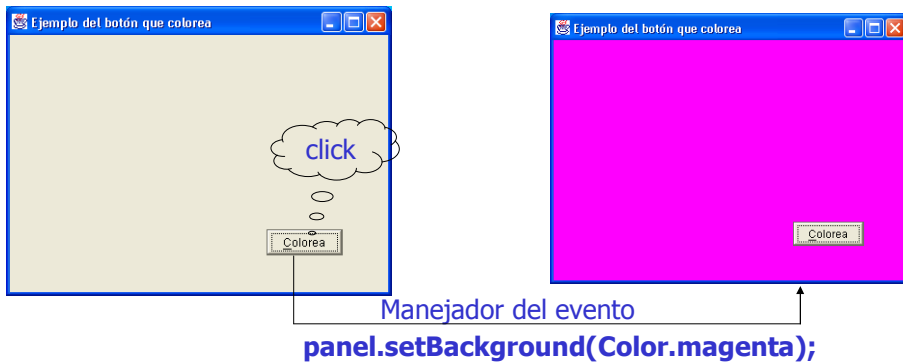
## Eventos o Sucesos

- Un evento es una señal, externa o interna a la aplicación, que produce la ejecución de un bloque de código que se escribe como un método.
- Ejemplos
  - Pulsar el ratón
  - Llamar a un método
  - Modificar el valor de una propiedad
- El programador tiene que escribir el código adecuado para responder a cada evento (manejo de eventos)



## Manejo de Eventos

- Ejemplo:



## Contenedores

- Los componentes no se encuentran aislados, sino **agrupados dentro de contenedores**.
- Los contenedores
  - Se utilizan para **recoger y gestionar otros componentes**
  - También son componentes
- Generalmente, en los contenedores no se emplean posiciones fijas, sino que los componentes están situados con una **disposición controlada (layouts)**





## Principales Tipos de Contenedores

- Marco (JFrame)
  - Ventana que no está contenida dentro de otra ventana
  - Tiene borde, título, menú de control, botones para maximizar y minimizar, controles para redimensionar y puede contener barra de menús
  - Suele ser el contenedor principal para una aplicación con IGU
- Cuadro de Diálogo (JDialog)
  - Generalmente se emplean para recoger datos del usuario y mostrar mensaje de advertencia
  - Tiene que tener un componente del que se derive
  - No puede contener barra de menús
  - Pueden ser modales o no modales



## Principales tipos de contenedores (II)

- Otros
  - Panel (JPanel)
    - Contenedor sencillo de la IU que se emplea para agrupar otros componentes como botones, etiquetas, campos de texto, etc.
    - Los paneles están incrustados en otros elementos como JFrame o JDialog.
  - Panel de Pestañas (JTabbedPane)
    - Contenedor compuesto por pestañas; el usuario puede cambiar de pestaña con tan solo hacer click en una de ella. Al utilizar este contenedor, primero se le añaden los JPanel necesarios y luego el resto de componentes dentro de estos.



## Componentes Básicos

Implicados en la construcción de una aplicación con IGU

- Con Swing una ventana es representada por un contenedor de alto nivel (normalmente **JFrame**)
- Sin embargo, los componentes no se añaden directamente sobre dicho contenedor
- Es necesario un panel (**JPanel**), llamado panel de contenido, sobre el que se sitúan los componentes
- Este panel está contenido en el JFrame.
- Sobre este panel se sitúan el resto de componentes

