



# **Interfaces gráficas**

**Jose Jesus García Rueda**

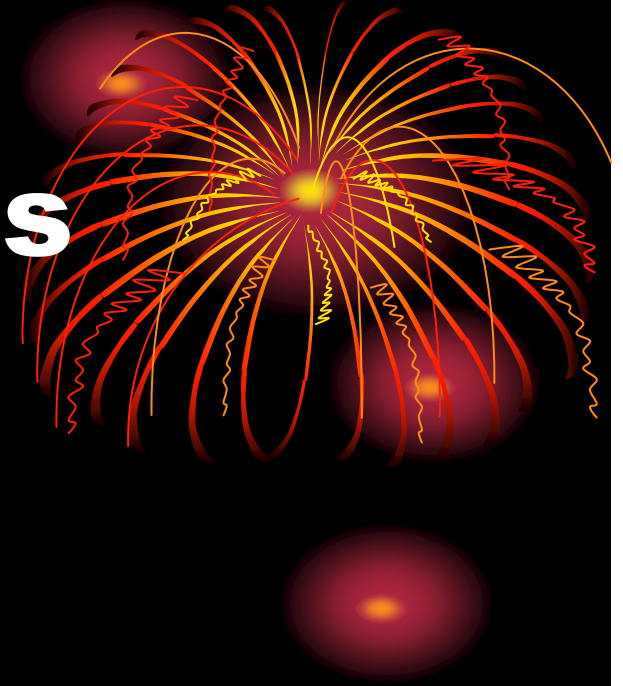
# Introducción

- **¿Qué son las interfaces gráficas?**
- **Ejemplos bien conocidos...**
- **Vamos a programar una:**
  - **Colores.java**

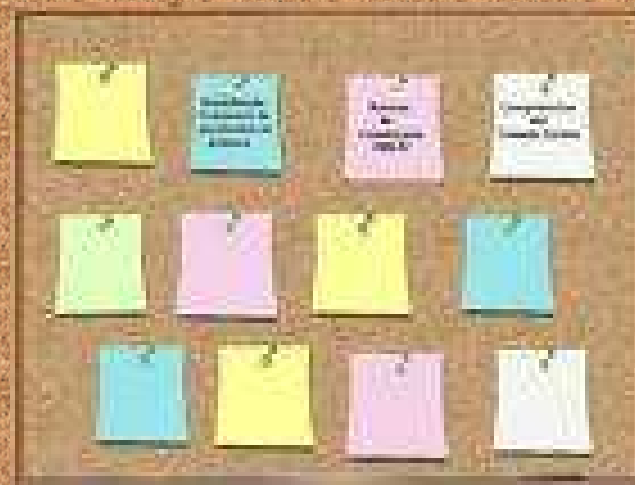


# Conceptos básicos

- **Aplicación gráfica.**
- **Contenedores.**
- **Acciones.**
- **Eventos.**
- **Elementos gráficos:**
  - **Barra de menús.**
  - **Barra de título.**
  - **Botones de minimizar y maximizar.**
  - **Botón de cerrar.**
  - **Scroll.**
  - **Marco de la ventana.**
  - **Iconos.**
  - **Botones.**
  - **Áreas de texto...**

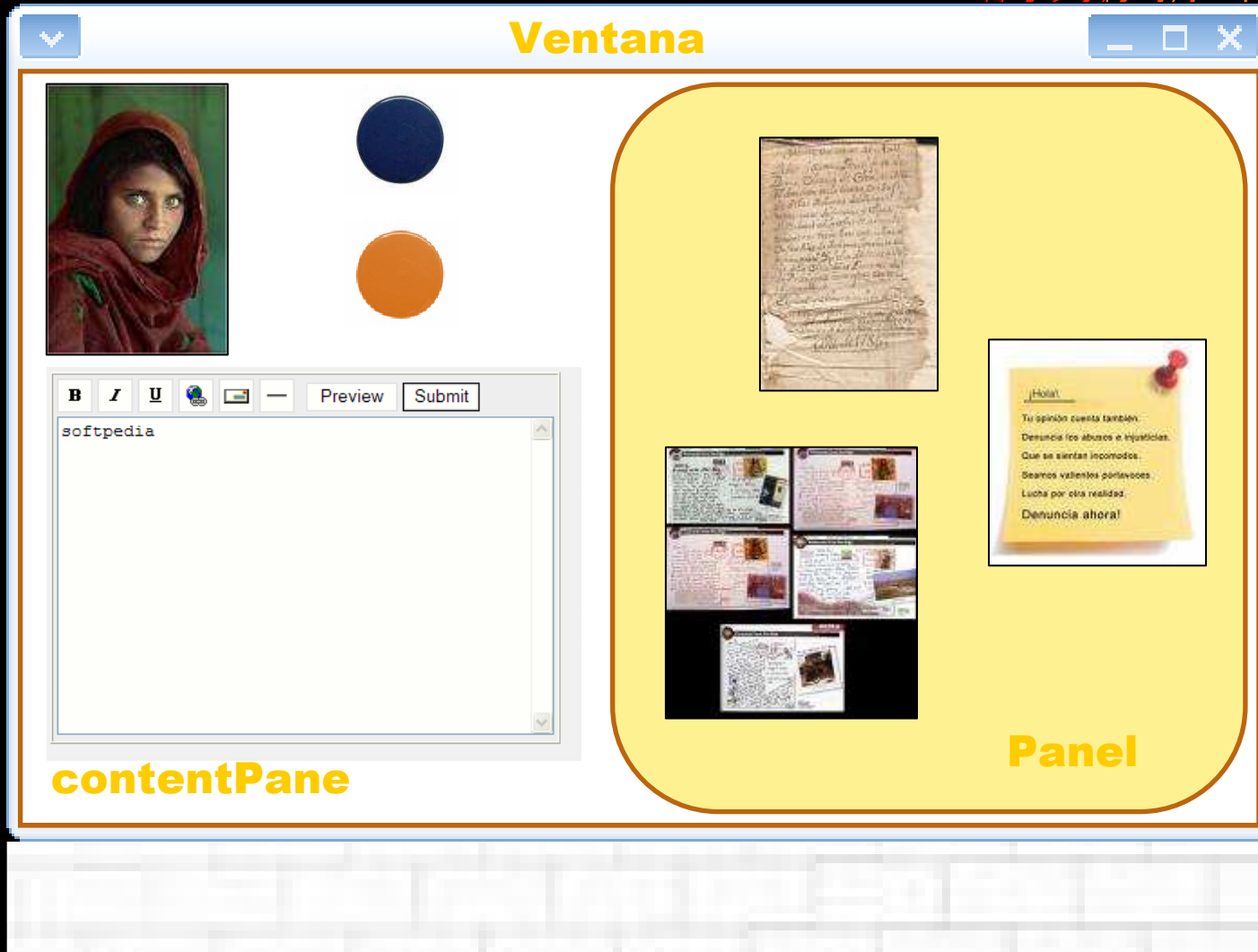


# La metáfora de la pared



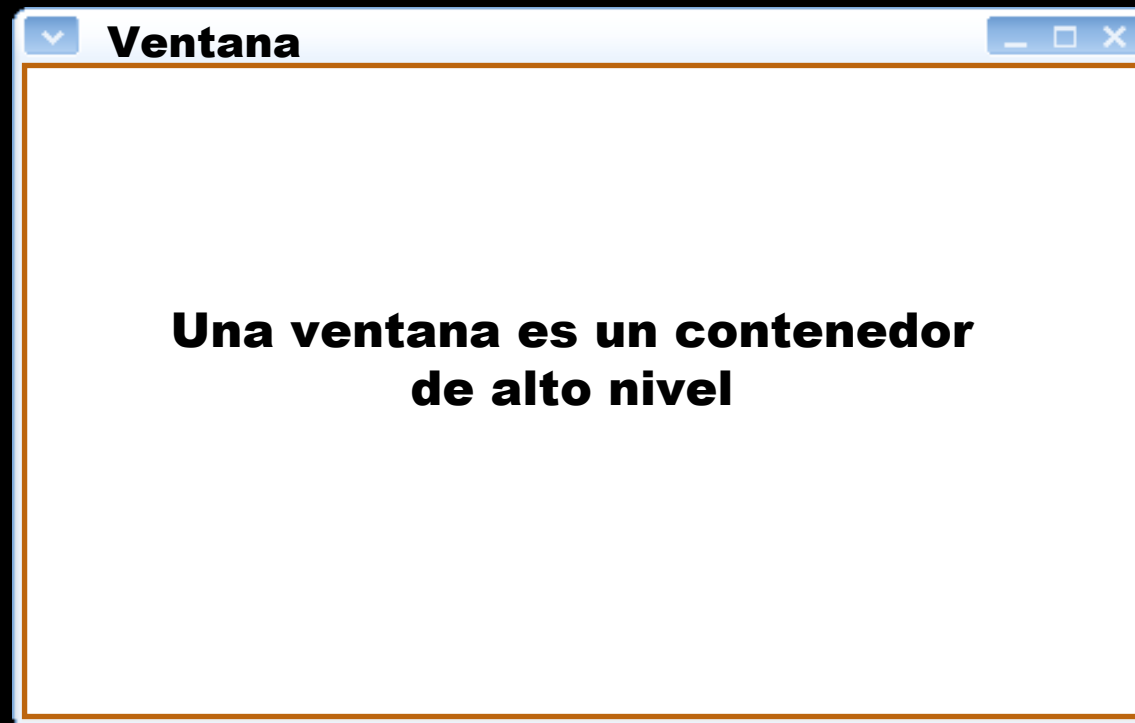


# La metáfora de la pared



# Creando la pared

- **¿Cómo crear una ventana en Java?**



# ¿Cómo se crea una ventana en Java?



Una ventana en Java no es más que una clase que hereda de JFrame, la ventana genérica

```
import javax.swing.*;
```

```
public class Ejemplo extends JFrame {
```

Las clases necesarias para construir interfaces gráficas pertenecen al paquete Swing

```
    todo que lo pone todo en marcha */
```

```
    @Override public static void main (String argv[]) {
```

```
        Ejemplo ventana = new Ejemplo();
```

```
        ventana.setSize (400, 400);
```

```
        ventana.setVisible(true);
```

```
    } // Fin del main
```

```
} // Fin de Ejemplo
```

Hay que, explícitamente, hacer visible la ventana

# Forrando la pared de corcho



- **Todo contenedor de alto nivel en Swing (como las ventanas) estará “forrado” con un PANEL RAIZ (“*content Pane*”).**
- **Sobre él se colocarán el resto de componentes de la ventana.**
  - **Incluidos otros contenedores.**





# Añadiéndolo al código



```
import javax.swing.* ;
```

```
public class Ejemplo extends JFrame {
```

```
/* El método que lo pone todo en marcha */  
public static void main (String argv[]) {
```

```
    Ejemplo ventana = new Ejemplo();
```

```
    ventana.getContentPane().add(...);
```

```
    ventana.setSize (400, 400);  
    ventana.setVisible(true);
```

```
} // Fin del main
```

```
} // Fin de Ejemplo
```

# ¿Qué elementos puedo “colgar en el corcho”?



- **En el panel raíz podré colgar componentes del paquete Swing:**
  - **Etiquetas:** **JLabel**
  - **Botones:** **JButton**
  - **Cajas de texto:** **JTextField, JTextArea**
  - **Casillas de verificación:** **JCheckBox**
  - **Botones de opción:** **JRadioButton**
  - **Listas:** **JList**
  - **Barras de desplazamiento:** **JScrollBar**
- **Todos los componentes Swing heredan de JComponent.**

# ¿Y cómo los cuelgo?



```
 JButton boton;  
 JLabel etiqueta;
```

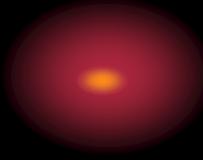
```
 public Ejemplo() {
```

```
     etiqueta = new JLabel("Una etiqueta");  
     boton = new JButton("Un botón");  
     boton.setSize(100, 70);  
     getContentPane().add(boton);  
     getContentPane().add(etiqueta);
```

```
 }
```

# PARADA TÉCNICA



- **(Aprovecharemos para echarle un vistazo a la API de Java, para saber dónde encontrar información sobre los elementos gráficos y cómo usarlos ;-)**
- 

# ¿Y cómo cuelgo “corchos dentro del corcho”?



- **Emplearemos PANELES DE CONTENIDO:** **Jpanel**
- **Son contenedores de nivel intermedio:**
  - **Simplifican la organización de la ventana.**
- **Un panel puede contener a otros paneles.**



# Jerarquía de paneles

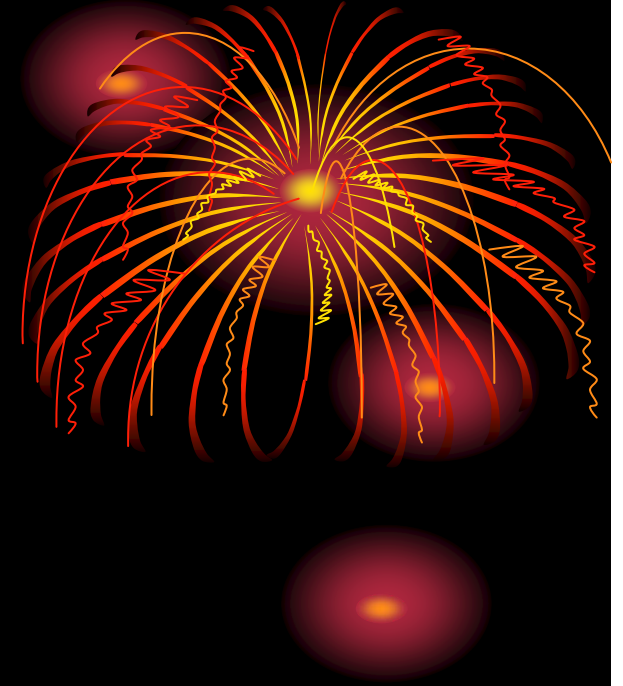


**B** *I* U Preview Submit

softpedia



# Ejemplo de panel



```
 JButton boton;  
 JLabel etiqueta;  
 JPanel panel;
```

```
 public Ejemplo() {
```

```
     panel = new JPanel();  
     getContentPane().add(panel);
```

```
     etiqueta = new JLabel("Una etiqueta");  
     boton = new JButton("Un botón");  
     boton.setSize(100, 70);  
     panel.add(boton);  
     panel.add(etiqueta);
```

```
 }
```

# ¿Cómo se coloca todo lo visto?



- **O bien por coordenadas...**
  - `etiqueta.setBounds(100, 70, 50, 50);`
- **...o bien mediante ADMINISTRADORES DE DISEÑO:**
  - Son como plantillas para colocar los elementos gráficos.
  - Se asocian a los paneles.
  - Aquí veremos tres tipos.
- **Para usar coordenadas hay primero que anular el layout:**
  - `panel.setLayout(null)`
- **¡Para usar los layouts hay que importar `java.awt.*`!!**

# Ejemplo con FlowLayout

**JButton boton;**  
**JLabel etiqueta;**  
**JButton otroBoton;**  
**JPanel panel;**

```
public Ejemplo() {
```

```
    panel = new JPanel();  
    getContentPane().add(panel);
```

```
    etiqueta = new JLabel("Una etiqueta");  
    boton = new JButton("Un botón");  
    otroBoton = new JButton("Otro Botón");  
    panel.add(boton);  
    panel.add(etiqueta);  
    panel.add(otroBoton);
```

```
}
```



**Coloca los elementos  
en línea, uno detrás de otro**

**¡FlowLayout viene por defecto!  
(En los paneles)**

# Ejemplo con GridLayout

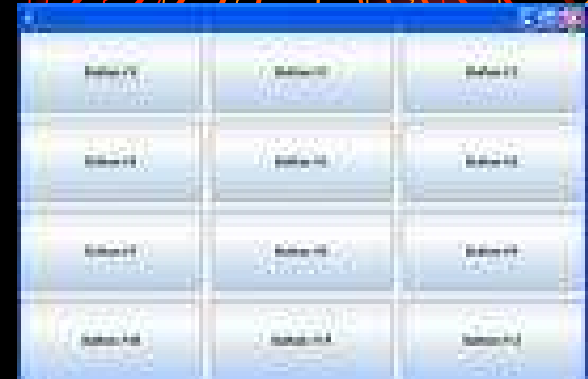
```
public Ejemplo() {
```

```
    panel = new JPanel();  
    panel.setLayout(new GridLayout(2, 2));  
    getContentPane().add(panel);
```

```
    etiqueta = new JLabel("Una etiqueta");  
    boton = new JButton("Un botón");  
    otroBoton = new JButton("Otro Botón");  
    panel.add(boton);  
    panel.add(etiqueta);  
    panel.add(otroBoton);
```

```
}
```

**Coloca los elementos en cuadrícula**





# Ejemplo con BorderLayout

Viene por defecto en los contenedores de nivel superior

```
public Ejemplo() {
```

```
    panel = new JPanel();  
    panel.setLayout(new BorderLayout());  
    getContentPane().add(panel);
```

```
    etiqueta = new JLabel("Una etiqueta");  
    boton = new JButton("Un botón");  
    otroBoton = new JButton("Otro Botón");  
    panel.add(boton, BorderLayout.SOUTH);  
    panel.add(etiqueta, BorderLayout.WEST);  
    panel.add(otroBoton, BorderLayout.NORTH);
```

```
}
```

Divide el contenedor en 5 secciones:  
*norte, sur, este, oeste y centro*



