

Tema 7: Interfaces gráficas de usuario II

Gestores de diseño (layouts)

Gestores de diseño por defecto:

- En un JFrame: BorderLayout.
- En un JPanel: FlowLayout

1. FlowLayout

Deja fluir los componentes de izquierda a derecha en el orden en que son añadidos al contenedor. Si no hay espacio en la fila de componentes, empieza debajo una nueva fila.

Por defecto los componentes aparecen centrados, comportamiento que podemos cambiar pasándole al constructor del FlowLayout las constantes:

- `new FlowLayout(FlowLayout.LEFT)`
- `new FlowLayout(FlowLayout.RIGHT)`
- `new FlowLayout(FlowLayout.CENTER)`

También podemos cambiar la separación horizontal y vertical entre componentes indicándolo al llamar al constructor. Por ejemplo, para dejar 25px en horizontal y 15px en vertical:

- `new FlowLayout(FlowLayout.LEFT, 25, 15);`

2. GridLayout

Sirve para dividir un panel en forma de rejilla con filas y columnas.



Para crearlo podemos usar dos constructores:

- `GridLayout gr = new GridLayout(filas, columnas);`
- `GridLayout gr = new GridLayout(filas, columnas, separación horizontal, separación vertical);`

Los componentes se van añadiendo por filas y columnas en el orden en que son añadidos al JFrame. La rejilla ocupa todo el panel del JFrame.

Ejemplo: Creamos una rejilla de 2 filas x 3 columnas y añadimos en cada una de ellas un botón. Los botones estarán separados entre ellos 10 px horizontal y verticalmente.

```
public GridLayout1() {
    super("Panel de tiro");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setSize(350, 200);

    setLayout(new GridLayout(2, 3, 10, 10));

    JButton btnSam1 = new JButton("SAM 1");
    JButton btnSam2 = new JButton("SAM 2");
    JButton btnTorpedo1 = new JButton("TORPEDO 1");
    JButton btnTorpedo2 = new JButton("TORPEDO 2");
    JButton btnHarpoon1 = new JButton("HARPOON 1");
    JButton btnHarpoon2 = new JButton("HARPOON 2");

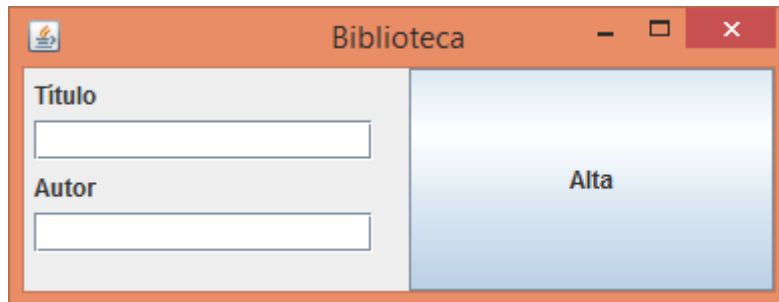
    add(btnSam1);
    add(btnTorpedo1);
    add(btnHarpoon1);
    add(btnSam2);
    add(btnTorpedo2);
    add(btnHarpoon2);
}
```

```

        setVisible(true);
    }

```

Ejemplo: Formulario para dar de alta libros en una biblioteca.



Lo que haremos es asignar un GridLayout a la ventana. Pero en la primera casilla añadimos un panel (Jpanel) al que hemos asignado un diseño FlowLayout con dos etiquetas y dos casillas de texto.

Los dos componentes del JFrame están enmarcados en rojo y verde.

```

public GridLayout2() {
    super("Biblioteca");
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    setSize(390, 150);

    setLayout(new GridLayout(1, 2, 10, 10));

    JPanel panelDatos = new JPanel();
    panelDatos.setLayout(new FlowLayout(FlowLayout.LEFT));
    JLabel lblTitulo = new JLabel("Título");
    JTextField txtTitulo = new JTextField(15);
    JLabel lblAutor = new JLabel("Autor");
    JTextField txtAutor = new JTextField(15);
    panelDatos.add(lblTitulo);
    panelDatos.add(txtTitulo);
    panelDatos.add(lblAutor);
    panelDatos.add(txtAutor);

    JButton btnAlta = new JButton("Alta");

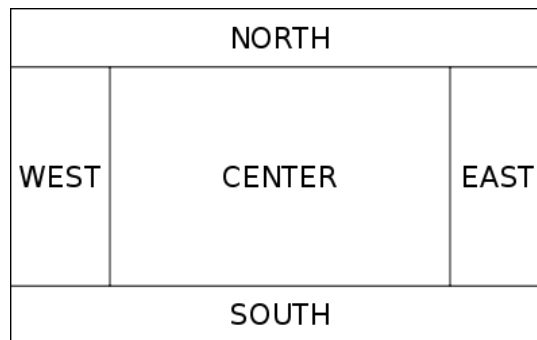
    add(panelDatos);
    add(btnAlta);

    setVisible(true);
}

```

3. BorderLayout

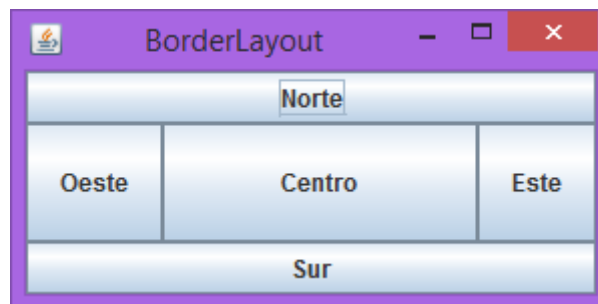
Este layout divide la ventana en cinco zonas, la más grande de las cuales es la central:



Cuando utilizamos este gestor de diseño, al añadir cada componente habrá que indicar dónde irá colocado. Por ejemplo:

```
setLayout(new BorderLayout());
add(botón, BorderLayout.NORTH);
```

Ejemplo: Colocamos un botón en cada zona del BorderLayout.



```
public BorderLayout1() {
    super("BorderLayout");
    setSize(300, 150);
    setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

    setLayout(new BorderLayout());

    JButton btnNorte = new JButton("Norte");
    JButton btnSur = new JButton("Sur");
    JButton btnEste = new JButton("Este");
    JButton btnOeste = new JButton("Oeste");
    JButton btnCentro = new JButton("Centro");

    add(btnNorte, BorderLayout.NORTH);
    add(btnSur, BorderLayout.SOUTH);
```

```

        add(btnEste, BorderLayout.EAST);
        add(btnOeste, BorderLayout.WEST);
        add(btnCentro, BorderLayout.CENTER);

        setVisible(true);
    }

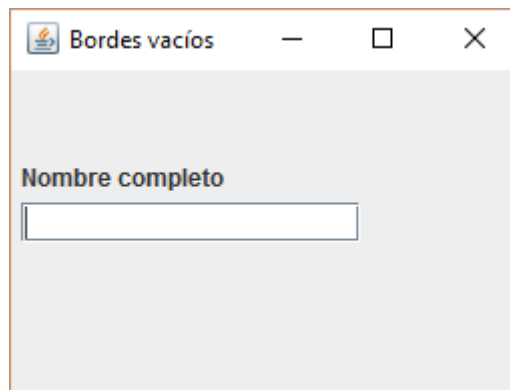
```

4.- Bordos (ampliación)

5.1 Bordos simples

Podemos añadir bordes a los paneles para ponerles un marco con título, o decorarlos, o dejar márgenes a su alrededor. Para ello usaremos la interfaz `Border` y la clase `BorderFactory`, que devuelve objetos que implementan `Border` y que pueden establecerse con el método `setBorder()` de `JPanel`.

Ejemplo: Ponemos un borde vacío al panel del `JFrame` para dejar 40 px de margen arriba de nuestros controles.



```

public class BordeVacio extends JFrame {

    public static void main(String[] args) {
        new BordeVacio();
    }

    public BordeVacio() {
        super("Bordes vacíos");
        setSize(270, 200);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        setLayout(new FlowLayout(FlowLayout.LEFT));

        JPanel panel = (JPanel) this.getContentPane();

        panel.setBorder(BorderFactory.createEmptyBorder(40, 0, 0, 0));

        JLabel lblNombre = new JLabel("Nombre completo");
        JTextField txtNombre = new JTextField(15);
    }
}

```

```

        add(lblNombre);
        add(txtNombre);

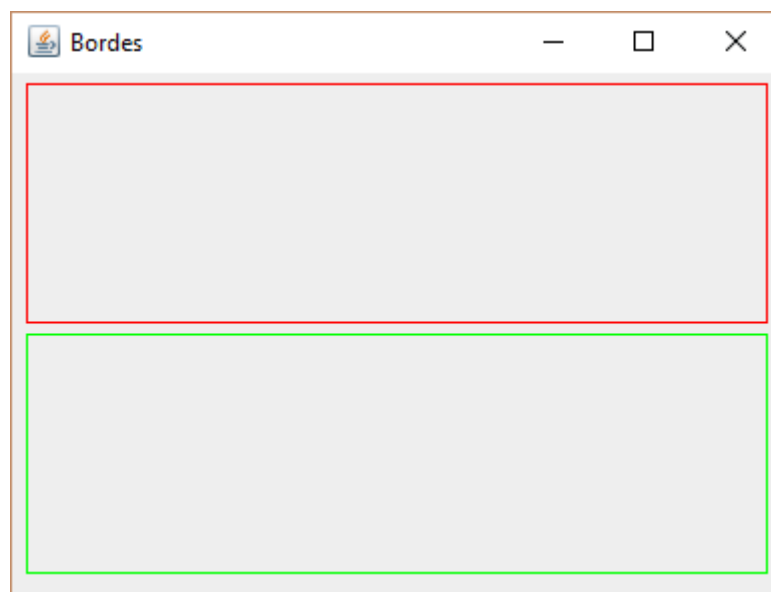
        setVisible(true);

    }

}

```

Ejemplo: Creamos dos paneles y los añadimos al JFrame. Cada panel tiene un tamaño fijo y un borde de diferente color.



```

public class BordesLinea extends JFrame {

    public static void main(String[] args) {
        new BordesLinea();
    }

    public BordesLinea() {
        super("Bordes");
        setSize(400, 300);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        setLayout(new FlowLayout());

        // PANEL SUPERIOR
        JPanel panelSuperior = new JPanel();

        panelSuperior.setPreferredSize(new Dimension(370, 120));
        panelSuperior.setBorder(BorderFactory.createLineBorder(Color.RED));

        // PANEL INFERIOR
        JPanel panelInferior = new JPanel();
        panelInferior.setPreferredSize(new Dimension(370, 120));
    }
}

```

```

panelInferior.setBorder(BorderFactory.createLineBorder(Color.GREEN));

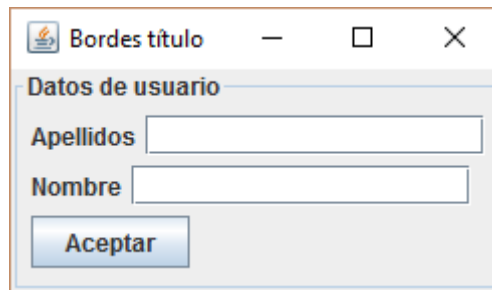
// VENTANA COMPLETA
add(panelSuperior);
add(panelInferior);

setVisible(true);
}

}

```

Ejemplo: Ponemos al panel de contenido de un JFrame un borde con título informativo, "Datos del usuario".



```

public class BordesTitulo extends JFrame {

    public static void main(String[] args) {
        new BordesTitulo();
    }

    public BordesTitulo() {
        super("Bordes título");
        setSize(260, 150);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        // PANEL DE CONTENIDO
        JPanel panel = (JPanel) this.getContentPane();
        panel.setLayout(new FlowLayout(FlowLayout.LEFT));

        panel.setBorder(BorderFactory.createTitledBorder("Datos de
usuario"));

        JLabel lblApellidos = new JLabel("Apellidos");
        JTextField txtApellidos = new JTextField(15);

        JLabel lblNombre = new JLabel("Nombre");
        JTextField txtNombre = new JTextField(15);

        JButton btnAceptar = new JButton("Aceptar");

        add(lblApellidos);
        add(txtApellidos);
        add(lblNombre);
        add(txtNombre);
        add(btnAceptar);
    }
}

```

```

        setVisible(true);
    }
}

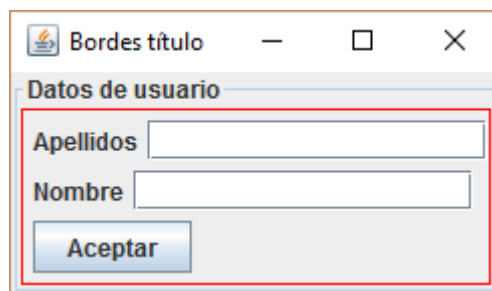
```

5.2 Bordes compuestos

Los bordes compuestos nos permiten añadir varios bordes simples (incluso también compuestos). El procedimiento consiste en crear dos bordes y asignar al panel un nuevo `CompoundBorder` al que pasamos ambos bordes como parámetros.

- `Border bordeTitulo = BorderFactory.createTitledBorder("Título del panel");`
- `Border bordeLinea = BorderFactory.createLineBorder(Color.RED);`
- `panel.setBorder(BorderFactory.createCompoundBorder(bordeTitulo, bordeLinea));`

Ejemplo: Podemos añadir al panel del ejemplo anterior un borde extra de color rojo:



```

public class BordesCompuestos extends JFrame {

    public static void main(String[] args) {
        new BordesCompuestos();
    }

    public BordesCompuestos() {
        super("Bordes título");
        setSize(260, 150);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

        // PANEL DE CONTENIDO
        JPanel panel = (JPanel) this.getContentPane();
        panel.setLayout(new FlowLayout(FlowLayout.LEFT));

        // Borde compuesto.
        Border bordeTitulo = BorderFactory.createTitledBorder("Datos de
usuario");
        Border bordeLinea = BorderFactory.createLineBorder(Color.RED);
        panel.setBorder(BorderFactory.createCompoundBorder(bordeTitulo,
bordeLinea));

        JLabel lblApellidos = new JLabel("Apellidos");

```



```
        JTextField txtApellidos = new JTextField(15);

        JLabel lblNombre = new JLabel("Nombre");
        JTextField txtNombre = new JTextField(15);

        JButton btnAceptar = new JButton("Aceptar");

        add(lblApellidos);
        add(txtApellidos);
        add(lblNombre);
        add(txtNombre);
        add(btnAceptar);

        setVisible(true);
    }
}
```