JPanel

JPanel, una parte del paquete Java Swing, es un contenedor que puede almacenar un grupo de componentes. La tarea principal de JPanel es organizar los componentes, se pueden establecer varios diseños en JPanel que proporcionan una mejor organización de los componentes, sin embargo, no tiene una barra de título.

Constructores de JPanel

1. **JPanel()**: crea un nuevo panel con un diseño de flujo
2. **JPanel(LayoutManager l)**: crea un nuevo JPanel con layoutManager especificado
3. **JPanel(boolean isDoubleBuffered):** crea un nuevo JPanel con una estrategia de buffering especificada
4. **JPanel(LayoutManager l, boolean isDoubleBuffered):** crea un nuevo JPanel con layoutManager especificado y una estrategia de almacenamiento en búfer especificada

Funciones más utilizadas de JPanel

1. **add(Componente c):** agrega un componente a un contenedor especificado
2. **setLayout(LayoutManager l):** establece el diseño del contenedor en el administrador de diseño especificado
3. **updateUI()**: restablece la propiedad de la interfaz de usuario con un valor de la apariencia actual.
4. **setUI(interfaz de usuario de PanelUI):** establece la apariencia de un objeto que representa este componente.
5. **getUI()**: devuelve el objeto look and feel que representa este componente.
6. **paramString()**: devuelve una representación de cadena de este JPanel.
7. **getUIClassID()**: devuelve el nombre de la clase Look and feel que representa este componente.
8. **getAccessibleContext()**: obtiene el AccessibleContext asociado a este JPanel.

Tomemos un programa de ejemplo para ilustrar el uso de la clase JPanel agregando instantáneas de ejecución secuencial de salidas que justifiquen los siguientes conjuntos de programas de la siguiente manera:

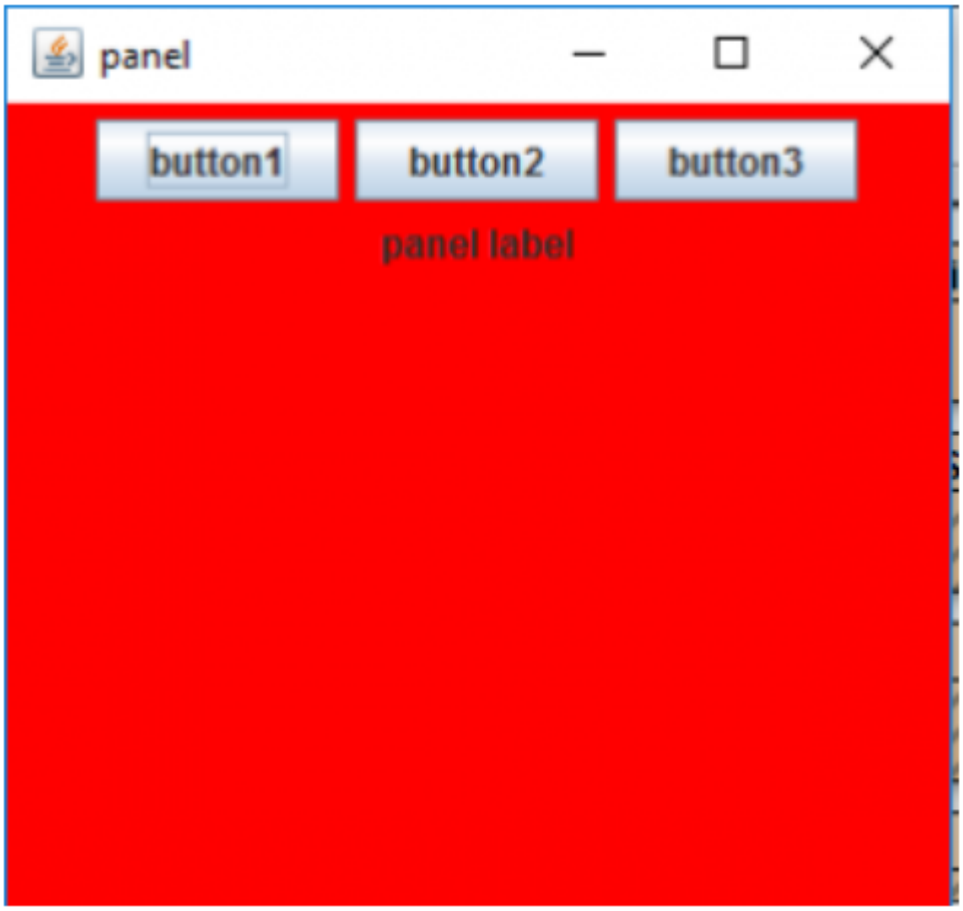
Ejemplo:

```
// Java Program to Create a Simple JPanel
// and Adding Components to it

// Importing required classes
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

// Class 1
// Helper class extending JFrame class
```

```
class solution extends JFrame {

    // JFrame
    static JFrame f;

    // JButton
    static JButton b, b1, b2;

    // Label to display text
    static JLabel l;

    // Main class
    public static void main(String[] args)
    {
        // Creating a new frame to store text field and
        // button
        f = new JFrame("panel");

        // Creating a label to display text
        l = new JLabel("panel label");

        // Creating a new buttons
        b = new JButton("button1");
        b1 = new JButton("button2");
        b2 = new JButton("button3");

        // Creating a panel to add buttons
        JPanel p = new JPanel();

        // Adding buttons and textfield to panel
        // using add() method
        p.add(b);
        p.add(b1);
        p.add(b2);
        p.add(l);

        // setbackground of panel
        p.setBackground(Color.red);

        // Adding panel to frame
        f.add(p);

        // Setting the size of frame
        f.setSize(300, 300);

        f.show();
```

```
    }
}
```

Salida:



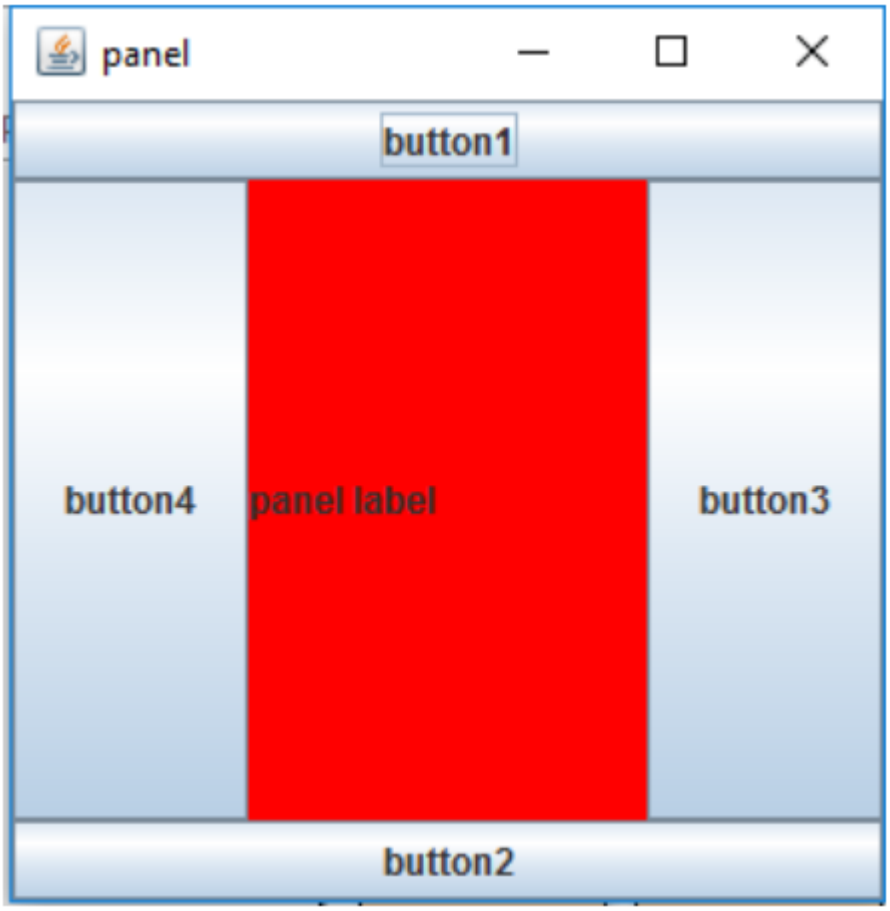
DescargarEjemplo

Ejemplo 2:

```
// Java Program to Create a JPanel with a Border Layout
// and Adding Components to It

// Importing required classes
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

// Main class
// Extending JFrame class
class solution extends JFrame {

    // JFrame
    static JFrame f;
```

```
    // JButton
    static JButton b, b1, b2, b3;

    // Label to display text
    static JLabel l;

    // Main driver method
    public static void main(String[] args)
    {
        // Creating a new frame to store text field and
        // button
        f = new JFrame("panel");

        // Creating a label to display text
        l = new JLabel("panel label");

        // Creating a new buttons
        b = new JButton("button1");
        b1 = new JButton("button2");
        b2 = new JButton("button3");
        b3 = new JButton("button4");

        // Creating a panel to add buttons
        // and a specific layout
        JPanel p = new JPanel(new BorderLayout());

        // Adding buttons and textfield to panel
        // using add() method
        p.add(b, BorderLayout.NORTH);
        p.add(b1, BorderLayout.SOUTH);
        p.add(b2, BorderLayout.EAST);
        p.add(b3, BorderLayout.WEST);
        p.add(l, BorderLayout.CENTER);

        // setbackground of panel
        p.setBackground(Color.red);

        // Adding panel to frame
        f.add(p);

        // Setting the size of frame
        f.setSize(300, 300);

        f.show();
    }
}
```

Salida:



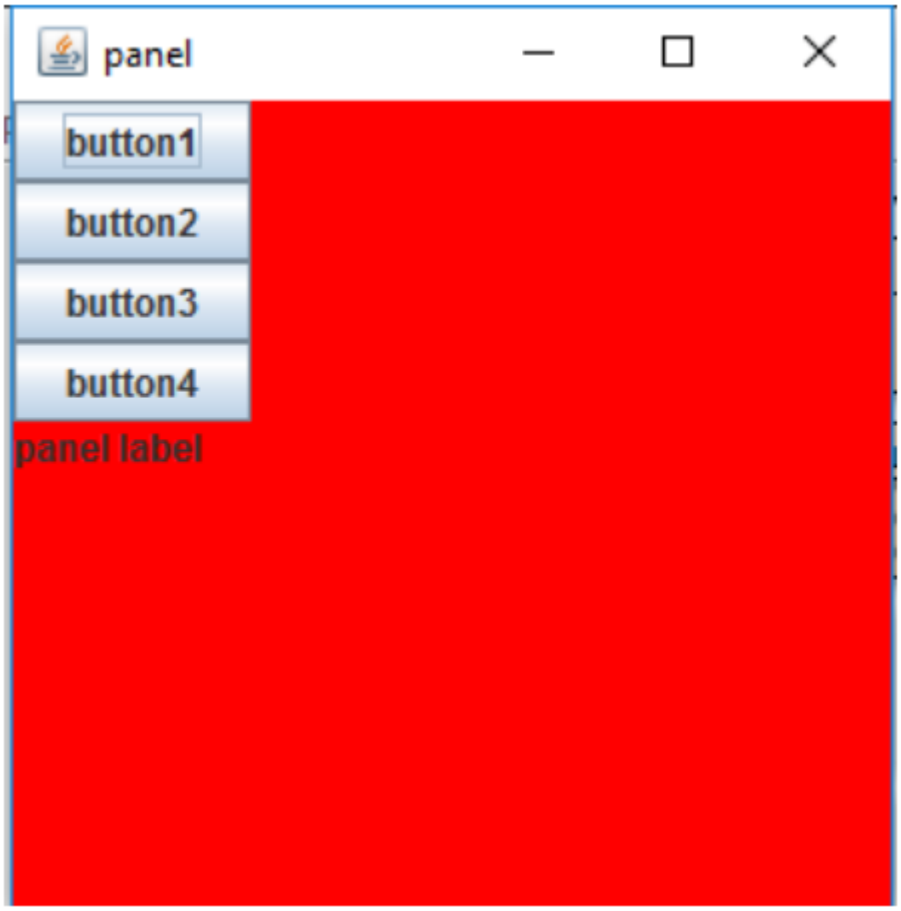
DescargarEjemplo

Ejemplo 3:

```
// Java Program to Create a JPanel with a Box layout
// and Adding components to it

// Importing required classes
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

// Main class
// Extending JFrame class
class solution extends JFrame {

    // JFrame
    static JFrame f;

    // JButton
    static JButton b, b1, b2, b3;
```

```
    // Label to display text
    static JLabel l;

    // Main drive method
    public static void main(String[] args)
    {
        // Creating a new frame to store text field and
        // button
        f = new JFrame("panel");

        // Creating a label to display text
        l = new JLabel("panel label");

        // Creating a new buttons
        b = new JButton("button1");
        b1 = new JButton("button2");
        b2 = new JButton("button3");
        b3 = new JButton("button4");

        // Creating a panel to add buttons and
        // textfield and a layout
        JPanel p = new JPanel();

        // Setting box layout
        p.setLayout(new BoxLayout(p, BoxLayout.Y_AXIS));

        // Adding buttons and textfield to panel
        p.add(b);
        p.add(b1);
        p.add(b2);
        p.add(b3);
        p.add(l);

        // Setting background of panel
        p.setBackground(Color.red);

        // Adding panel to frame
        f.add(p);

        // Setting the size of frame
        f.setSize(300, 300);

        f.show();
    }
}
```

Salida:



Bibliografía:

- Extraído de <<Java Swing - JPanel con ejemplos>>. Consultado el 28 de noviembre del 2022