

JComboBox - JList

Componentes de interfaz de
usuario en Java Swing

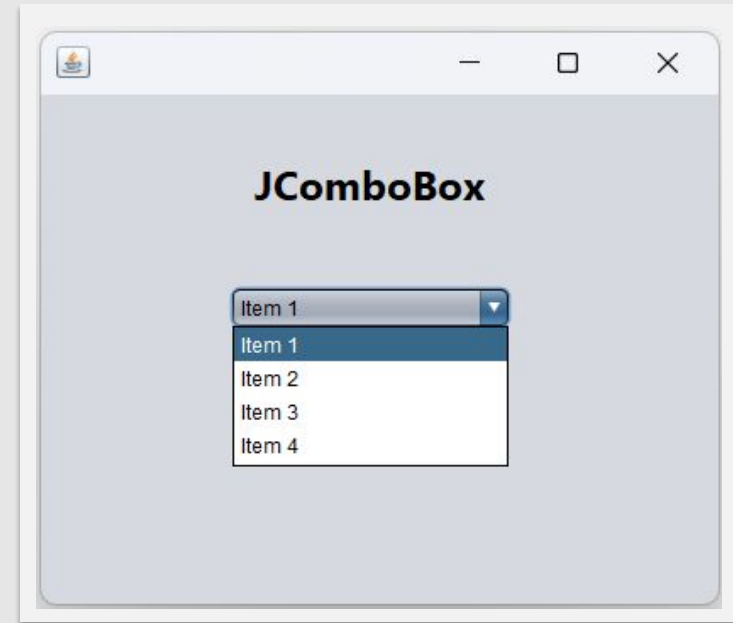
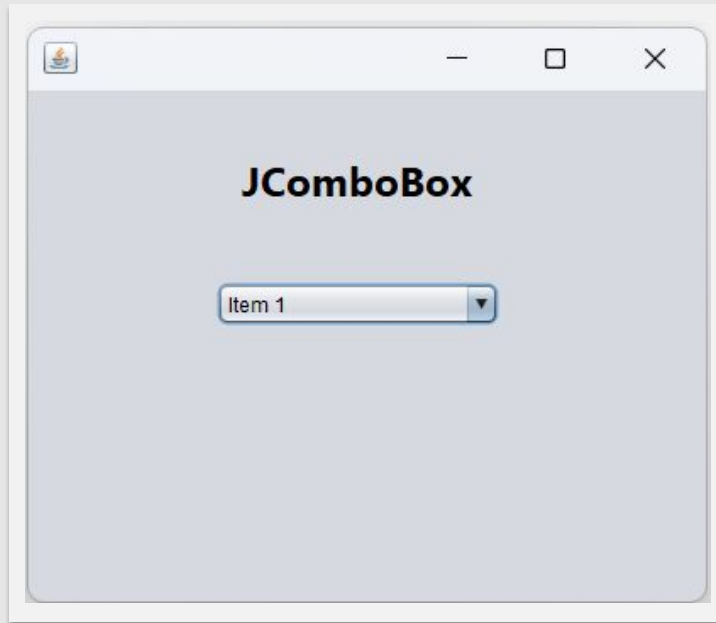
César Delgado – Jeicob Murillo

Facultad de Ingeniería – Sede Bogotá

Universidad Nacional de Colombia

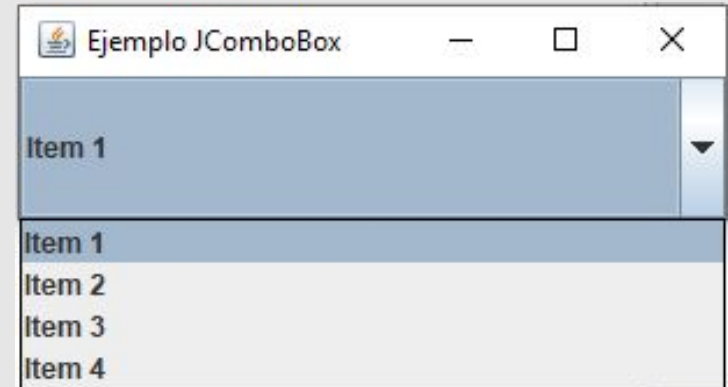
PROYECTO **CULTURAL, CIENTÍFICO Y COLECTIVO** DE NACIÓN

JComboBox - Lista Desplegable



- Componente que permite al usuario seleccionar un elemento de una lista predefinida.
- Se asemeja a una caja de selección que muestra una opción seleccionada y permite abrir una lista de opciones.

JComboBox - Estructura



```
//Ejemplo para exposición Jeicob Murillo y César Delgado

import javax.swing.JComboBox;

public class JFramePractice {

    public static void main(String[] args) {

        // Crear una matriz de objetos para usar como elementos en JComboBox
        Object[] items = {"Item 1", "Item 2", "Item 3", "Item 4"};

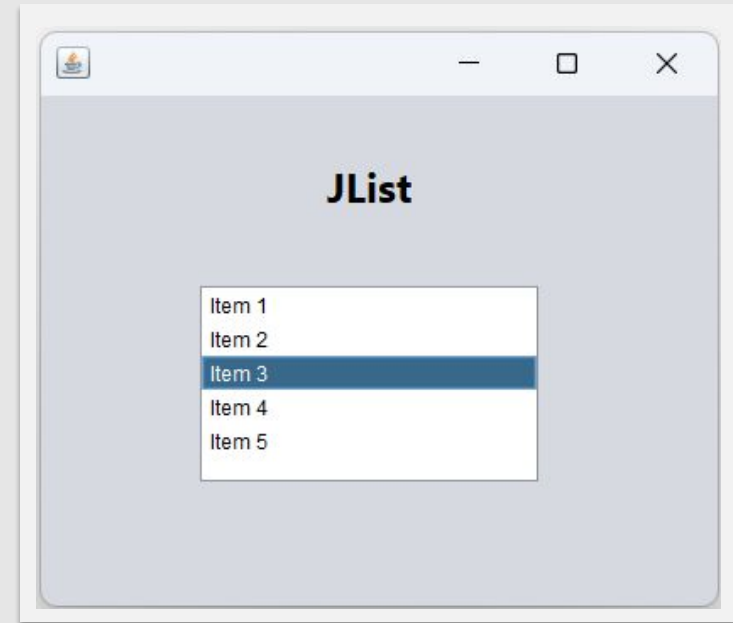
        // Crear un JComboBox y pasar la matriz de objetos
        JComboBox<Object> comboBox = new JComboBox<>(items);

        // Agregar JComboBox al contenedor (por ejemplo, un panel o un marco)
        // panel.add(comboBox);

    }

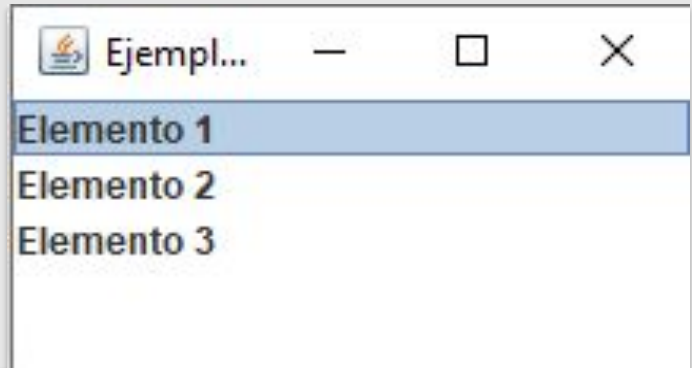
}
```

JList - Lista de Elementos



- Es un componente que muestra una lista de elementos que el usuario puede seleccionar.
- Es útil cuando se necesita mostrar una lista de elementos completa y permitir selecciones múltiples o simples.

JList - Lista de Elementos



```
import javax.swing.DefaultListModel;
import javax.swing.JList;

public class JFramePractice {

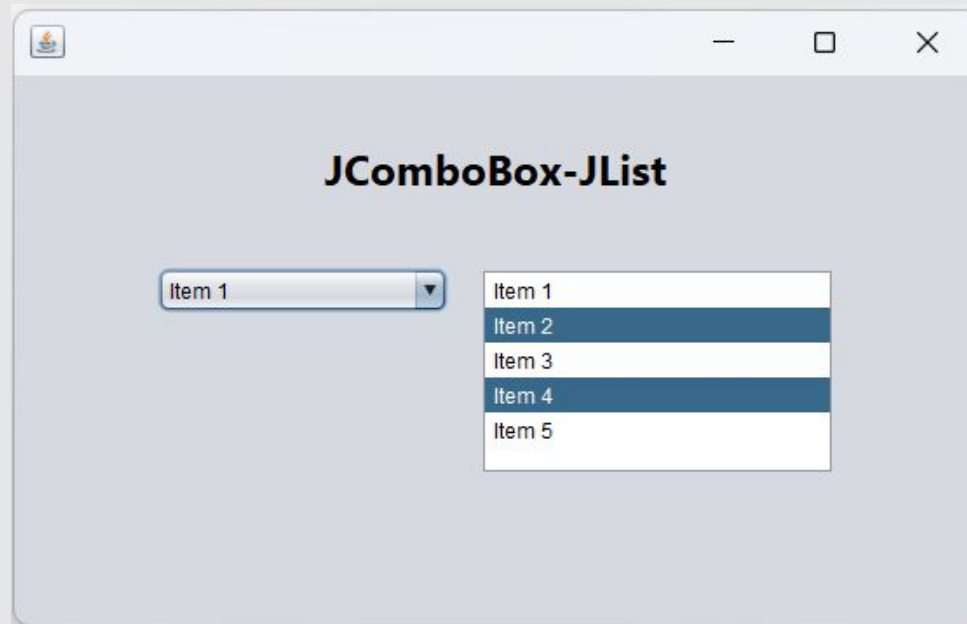
    public static void main(String[] args) {
        // Crear un modelo de lista para almacenar los elementos de la lista
        DefaultListModel<String> listModel = new DefaultListModel<>();

        // Agregar elementos al modelo de lista
        listModel.addElement(element: "Elemento 1");
        listModel.addElement(element: "Elemento 2");
        listModel.addElement(element: "Elemento 3");

        // Crear una instancia de JList y pasar el modelo de lista
        JList<String> list = new JList<>(dataModel: listModel);

        // Agregar JComboBox al contenedor (por ejemplo, un panel o un marco)
        // panel.add(list);
    }
}
```

Ventajas



JComboBox

- Ahorro de Espacio: Ideal para conservar espacio en la interfaz de usuario al mostrar solo la selección actual.
- Selección Única: Permite al usuario elegir una opción de una lista.
- Lista Predefinida: Útil cuando la lista de opciones es conocida y limitada.

JList

- Lista Completa: Muestra una lista completa de elementos.
- Selección Múltiple: Permite a los usuarios seleccionar varios elementos.
- Personalización: Puede personalizarse con un modelo de datos personalizado.

Gracias

Universidad Nacional de Colombia

PROYECTO **CULTURAL, CIENTÍFICO Y COLECTIVO** DE NACIÓN