**Ejercicios de Interfaces Gráficas en Java (Swing y AWT)**

**Sección 1: Componentes básicos y diseño de interfaces**

1. **Crea una ventana con JFrame** que tenga un título personalizado y un tamaño de 500x300 píxeles. Configúrala para que cierre la aplicación al presionar el botón de cerrar.
2. **Agrega un JPanel a la ventana creada en el ejercicio anterior** y cambia su color de fondo a azul.
3. **Diseña una interfaz utilizando un FlowLayout** que contenga tres botones con los textos "Aceptar", "Cancelar" y "Salir".
4. **Crea una interfaz con un BorderLayout** y distribuye cinco botones en sus posiciones correspondientes (Norte, Sur, Este, Oeste y Centro).
5. **Implementa una interfaz con un GridLayout(3,2)** que contenga seis botones numerados del 1 al 6.

**Sección 2: Eventos en Java Swing**

1. **Crea una ventana con un botón que cambie su texto al ser presionado** (Ejemplo: si inicia con "Presióname", cambiarlo a "¡Presionado!").
2. **Desarrolla una interfaz con dos botones: "Rojo" y "Verde"**. Al presionar cada botón, el fondo del JPanel debe cambiar al color correspondiente.
3. **Implementa un contador de clics con un botón y una etiqueta**. Cada vez que se presione el botón, la etiqueta debe mostrar la cantidad de veces que se ha pulsado.
4. **Crea un campo de texto y un botón**. Al presionar el botón, el contenido del campo de texto debe mostrarse en un cuadro de diálogo (JOptionPane).
5. **Programa una interfaz con un JTextField y un botón "Borrar"**. Al presionar el botón, el campo de texto debe vaciarse.

**Sección 3: Eventos del teclado y ratón**

1. **Implementa un programa que detecte cuando el usuario presiona la tecla "Enter"** dentro de un JTextField y muestre un mensaje en la consola.
2. **Crea una interfaz que detecte los eventos de ratón sobre un JPanel** y muestre en la consola las coordenadas del puntero cuando se haga clic.
3. **Implementa un MouseListener en un JLabel**, de modo que al pasar el cursor por encima, el texto cambie de color.
4. **Programa una interfaz con un KeyListener** que cambie el color de fondo de un JPanel cuando se presione la tecla "R" (Rojo), "G" (Verde) o "B" (Azul).
5. **Crea una aplicación donde el usuario pueda mover un JLabel arrastrándolo con el ratón**.

**Sección 4: Componentes interactivos**

1. **Crea un formulario de inicio de sesión** con dos JTextField (usuario y contraseña) y un JButton para validar los datos.
2. **Implementa un JCheckBox que habilite o deshabilite un campo de texto** según su estado.
3. **Desarrolla una interfaz con un grupo de JRadioButton** que permita al usuario seleccionar su género (Masculino, Femenino, Otro).
4. **Crea una lista (JList) con una serie de nombres** y un botón que muestre en una etiqueta el nombre seleccionado.
5. **Programa un JComboBox con diferentes colores**. Al seleccionar un color, el fondo de la ventana debe cambiar a ese color.

Aquí tienes la siguiente tanda de 20 ejercicios:

**Sección 5: Menús en Java Swing**

1. **Crea una ventana con una barra de menús** que tenga un menú llamado "Archivo" con las opciones "Abrir", "Guardar" y "Salir".
2. **Desarrolla un menú con tres opciones en el menú "Archivo"**: "Nuevo", "Abrir", "Salir". Cuando se seleccione "Salir", la aplicación debe cerrarse.
3. **Implementa un menú con un JCheckBoxMenuItem** para activar o desactivar un tema oscuro en la interfaz.
4. **Crea un menú con dos opciones**: "Acelerador" que ejecute una acción cuando se presiona Ctrl+S, y "Salir" que cierre la ventana al seleccionarlo.
5. **Diseña una ventana con un menú** que contenga un submenú "Preferencias", con las opciones "Cambiar color" y "Configurar sonido".

**Sección 6: Contenedores y organizadores**

1. **Crea una ventana con un JPanel que contenga tres botones** y organízalos con un FlowLayout. Asegúrate de que el JPanel esté alineado al centro de la ventana.
2. **Diseña una ventana con un JPanel de tipo GridLayout(2, 2)** y coloca cuatro botones en las casillas del layout.
3. **Implementa un JPanel con un BorderLayout** y agrega un botón en cada una de las posiciones: Norte, Sur, Este, Oeste y Centro.
4. **Crea una ventana con un JTabbedPane** que contenga dos paneles: uno con un campo de texto y otro con un botón.
5. **Desarrolla una ventana con un JScrollPane** que contenga una lista larga de elementos (mínimo 50). Asegúrate de que la lista sea desplazable.

**Sección 7: Componentes avanzados**

1. **Crea una ventana con un JList que permita seleccionar múltiples elementos**. Usa un ListSelectionModel para configurar la selección múltiple.
2. **Implementa una aplicación que contenga un JComboBox con diferentes frutas**. Al seleccionar una fruta, un mensaje debe mostrar el nombre de la fruta seleccionada.
3. **Crea una ventana con un JTextArea** que permita ingresar texto, y un botón "Limpiar" que borre todo el contenido del área de texto.
4. **Programa una interfaz con un JPasswordField** para ingresar una contraseña y un botón "Mostrar" que muestre el texto de la contraseña.
5. **Desarrolla una ventana que contenga un JTextArea** para escribir notas y un botón para guardar esas notas en un archivo de texto.

**Sección 8: Diseño y aspecto visual**

1. **Crea una ventana con un JLabel** que contenga un mensaje, y cambia el color y tamaño de la fuente.
2. **Diseña una ventana con un JButton** y configura su borde y fondo con diferentes colores.
3. **Implementa un formulario con JTextField, JPasswordField y un botón "Enviar"**. Establece el borde de cada campo de texto para que tenga un aspecto más visual.
4. **Crea un JRadioButton que cambie el color de fondo de la ventana** al seleccionar una opción (rojo, verde, azul).
5. **Desarrolla una interfaz con un JCheckBox que permita mostrar u ocultar un texto en la ventana**.

**Sección 9: Integración de componentes con eventos**

1. **Crea una interfaz que tenga un JTextField, un JButton y un JLabel**. Cuando el usuario ingrese texto en el campo de texto y presione el botón, el texto debe mostrarse en la etiqueta.
2. **Implementa una ventana con un JComboBox** que contenga los días de la semana. Al seleccionar un día, un mensaje debe mostrar el día seleccionado.
3. **Desarrolla una interfaz con un JCheckBox que, al seleccionarlo, active o desactive un JButton**.
4. **Crea una ventana con un JList y un JButton**. Al seleccionar un ítem de la lista y presionar el botón, el nombre del ítem debe mostrarse en un cuadro de mensaje.
5. **Programa una interfaz que contenga un JTextArea**. Al escribir algo en el área de texto y presionar un botón, se debe mostrar el contenido en un cuadro de diálogo.

**Sección 10: Finalización de aplicaciones**

1. **Crea una interfaz con un botón "Cerrar"** que cierre la ventana de la aplicación al hacer clic.
2. **Desarrolla una ventana con un JButton que, al ser presionado,** muestre una confirmación de si realmente se quiere cerrar la aplicación.
3. **Programa una ventana que contenga un botón "Salir"**. Al presionarlo, la aplicación debe cerrarse de forma limpia, mostrando un mensaje de despedida antes de cerrarse.
4. **Implementa una aplicación con una ventana principal que se cierra con un JFrame.EXIT\_ON\_CLOSE** al hacer clic en el botón de cerrar.
5. **Crea una ventana que se cierre al presionar una tecla determinada (por ejemplo, Esc).**

Aquí tienes la última parte de los ejercicios:

**Sección 11: Uso avanzado de contenedores y componentes**

1. **Crea una ventana con un JPanel que contenga un formulario de contacto** con campos de texto para nombre, dirección de correo electrónico y mensaje. Añade un botón "Enviar" que muestre los datos ingresados en un cuadro de mensaje.
2. **Implementa una interfaz con un JList y un JComboBox**. Al seleccionar un valor del JComboBox, se deben mostrar en la lista sólo los elementos que coincidan con la selección.
3. **Desarrolla una interfaz con un JTabbedPane que contenga tres paneles**: uno con un JLabel, otro con un JTextField, y otro con un JButton.
4. **Crea una ventana con un JPanel que contenga un GridLayout(4, 2)** con 8 botones. Añade un JCheckBox que permita activar o desactivar el panel (habilitar/deshabilitar todos los botones).
5. **Programa una interfaz con un JTextArea y un JScrollPane** que permita al usuario agregar texto largo, mostrando el área de desplazamiento de forma automática.

**Sección 12: Interacción con archivos y almacenamiento**

1. **Crea una interfaz con un JTextArea y un botón "Guardar"**. Al presionar el botón, el texto del área de texto debe guardarse en un archivo de texto en el disco.
2. **Desarrolla una ventana con un JButton y un JTextField**. Cuando se presione el botón, el texto en el campo debe guardarse en un archivo XML.
3. **Programa una interfaz con un JTextArea, un JScrollPane y un botón "Cargar"**. Al presionar el botón, el contenido de un archivo de texto se debe cargar en el área de texto.
4. **Crea una ventana con un JComboBox que permita seleccionar un archivo desde el sistema de archivos** y mostrar el contenido del archivo seleccionado en un JTextArea.
5. **Desarrolla una aplicación con un JButton que, al ser presionado,** muestre el contenido de un archivo de texto en una ventana emergente (JOptionPane).